Hackowanie etyczne

Przegląd protokołów

Protokoły są często omawiane lub przedstawiane jako pojedynczy element lub element sieci i sposobu, w jaki dane przemieszczają się z jednego miejsca do drugiego. Na przykład TCP/IP (Transmission Control Protocol/Internet Protocol) to dwa oddzielne protokoły, które są ze sobą połączone, aby przesyłać dane z punktu A do punktu B. Istnieją inne protokoły zaprojektowane do obsługi określonych funkcji i komunikacji, ale żaden nie dorównuje temu, co TCP/IP robi dla komunikacji i ogólnego działania sieci oraz internetu jako całości. Dlaczego więc omawiamy protokoły i co mają one wspólnego z hakowaniem etycznym? Większość ludzi myśli, że hakowanie polega na hasłach lub lukach w aplikacjach, ale protokoły również mogą być nadużywane. Czasami ich podstawowe zachowanie pozwala atakującym uzyskać poszukiwane informacje.Przyjrzymy się bliżej, aby dowiedzieć się, dlaczego nadużywanie protokołów jest czymś, co należy zrozumieć i mieć na uwadze. Omówimy następujące główne tematy:

• Omówienie protokołów komunikacyjnych

• Ataki hakerskie, luki w zabezpieczeniach i wykrywanie ataków na protokoły

• Przegląd ataków i wykrywania za pomocą protokołu internetowego w wersji 6 (IPv6)

Omówienie protokołów komunikacyjnych

W tej sekcji omówimy przede wszystkim trzy protokoły komunikacyjne: TCP, UDP i ICMP. Istnieje wiele innych protokołów, które można by omówić; jednak te trzy są częścią wszystkich serwerów i stacji roboczych. Zanim jednak omówimy je szczegółowo, zacznijmy od przyjrzenia się modelowi Open System Interconnection (OSI), który definiuje znaczną część działania protokołów.

Wprowadzenie do modelu OSI

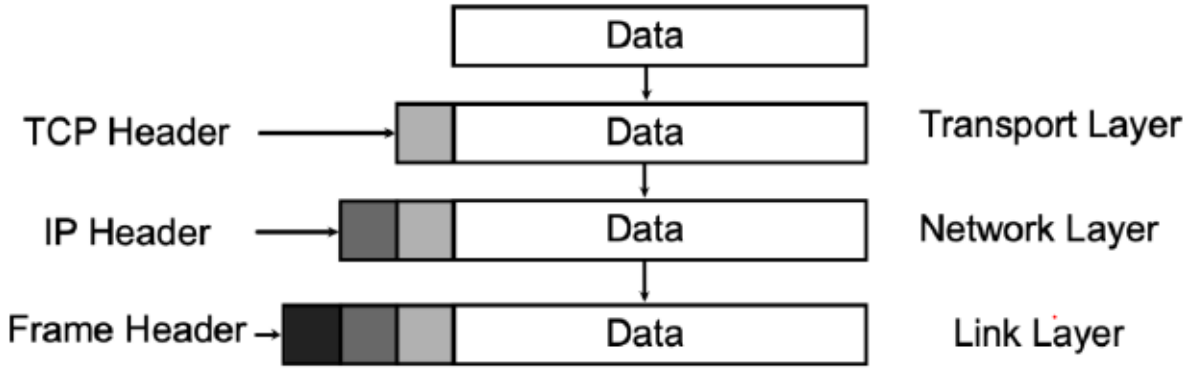
Model OSI to platforma umożliwiająca ustanowienie wspólnej metody komunikacji między systemami. Podstawową strukturę ilustruje poniższy rysunek:

OSI Model

Layer 7	Application	
Layer 6	Presentation	HTTP, FTP, telnet
Layer 5	Session	
Layer 4	Transport	TCP, UDP
Layer 3	Network	IP, ICMP, IGMP
Layer 2	Link	ARP, RARP
Layer 1	Physical	

Model składa się z siedmiu warstw, w których nadawca przetwarza warstwy w odwrotnej kolejności, zaczynając od warstwy 7 i dodając kolejne warstwy, aż do warstwy 1, w której transmisja danych jest

przenoszona od nadawcy do odbiorcy. Po otrzymaniu danych przez odbiorcę, proces przebiega w kolejności od warstwy 1, przechodząc przez kolejne warstwy, aż do warstwy 7, gdzie dane są wyświetlane w aplikacji. Przyjrzyjmy się bliżej przeznaczeniu każdej z tych warstw:



Teraz, gdy omówiliśmy nagłówki IP i TCP oraz sposób, w jaki model OSI kapsułkuje różne warstwy, aby ułatwić komunikację, przyjrzyjmy się pokrótce, w jaki sposób tworzona jest komunikacja, zanim przejdziemy do tego, w jaki sposób atakujący wykorzystują protokół TCP/IP i jak on działa.

Wprowadzenie do protokołu IP

Część protokołu IP protokołu TCP/IP to protokół warstwy sieciowej. Jako mechanizm routingu jest on używany wraz z innymi protokołami, takimi jak protokół UDP (User Datagram Protocol), protokół ICMP (Internet Control Message Protocol) i protokół IGMP (Internet Group Management Protocol), oprócz protokołu TCP. Protokół IP oferuje narzędzia do enkapsulacji informacji na potrzeby routingu między lokalizacjami źródłowymi i docelowymi za pomocą 32-bitowych adresów IP, a także zapewnia obsługę błędów, fragmentację, wygasanie danych i kontrolę określonych ustawień protokołu. Kluczowymi czynnikami protokołu IP są transmisja i dostarczanie danych. Ponieważ dostarczanie danych nie jest gwarantowane, protokół IP wdrożył kilka mechanizmów wspomagających dostarczanie danych za pośrednictwem następujących usług:

• Tabele tras IP: Aby przesyłać datagramy, lepiej znane jako pakiety, z jednego miejsca do drugiego, trasy zawarte w tabeli tras są konsultowane w celu znalezienia najlepszej ścieżki, w oparciu o dane adresowe źródła i przeznaczenia.

• Enkapsulacja: Przed transmisją w warstwie fizycznej, ruch warstwy wyższej jest enkapsulowany w pakiety IP, a po odebraniu przez interfejs sieciowy, ruch kierowany do wyższej warstwy protokołu jest dekapsulowany. Dla porównania, jest to bardzo podobne do kompresji pliku. Źródło kompresuje lub kompresuje plik i wysyła go dalej. Odbiorca musi rozpakować zawartość, aby ją odczytać.

• Usługi fragmentacji i ponownego składania pakietów: Gdy warstwa IP odbiera komunikat do przesłania, jest on analizowany poprzez obliczenie rozmiaru pakietu i porównanie go z maksymalnym rozmiarem jednostki transferu (MTU). Jeśli całkowity rozmiar jest większy niż MTU, warstwa IP ustawi flagę fragmentacji i podzieli wiadomość na fragmenty, zwane fragmentami. Host odbierający będzie odpowiedzialny za ponowne złożenie wiadomości we właściwej kolejności.

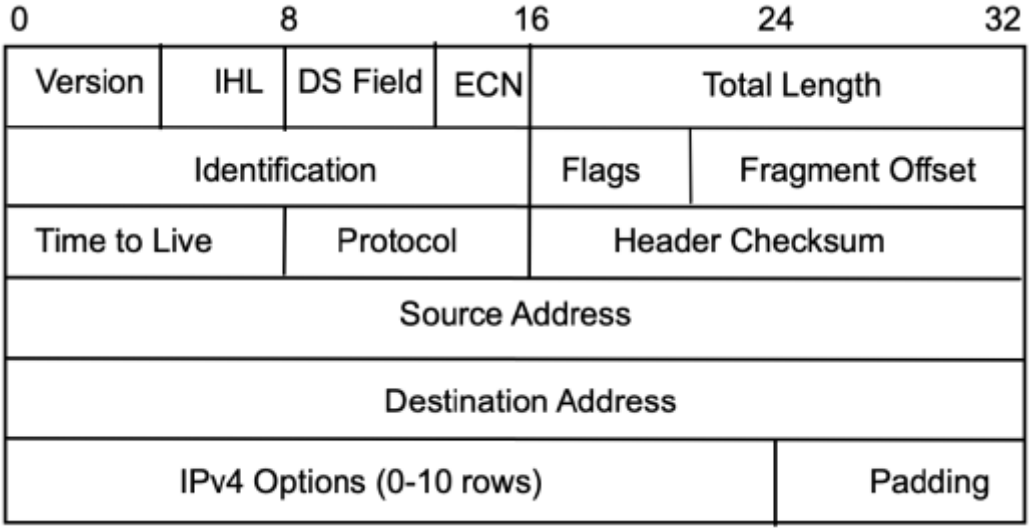
• Usługi interpretacji adresów IP: Ta funkcja ułatwia analizę adresu IP, adresując jego podstawowe sekcje, w tym sieć, podsieć i komponenty hosta. Jest ona używana w połączeniu z tablicą routingu do podejmowania inteligentnych decyzji dotyczących routingu w komunikacji sieciowej.

• Usługi rozgłoszeniowe i multicastowe: Usługi te zapewniają rozpoznawanie nazw poprzez rozgłaszanie w podsieci, w której uczestniczy host, oraz multicast, gdzie wiele hostów może komunikować się jednocześnie z jednego hosta.

• Obsługa błędów: Chociaż nie zapewnia niezawodnej komunikacji, opiera się na sprawdzaniu błędów za pomocą sum kontrolnych.

• Typ kwalifikacji usługi: Umożliwia routing pakietów przez urządzenia pośredniczące w oparciu o określone warunki, takie jak opóźnienie sieci, wykorzystanie pasma i niezawodność. Innymi słowy, wykorzystuje zebrane informacje do znalezienia najlepszej ścieżki routingu.

Skoro już wiemy, jak działa adres IP i jakie usługi się w nim znajdują, zobaczmy, jak on wygląda. Przyjrzyjmy się logicznemu układowi adresu IP i jego polom:



Po przejrzeniu przechwyconego pakietu i zaobserwowaniu nagłówka IP, będzie on wyglądał następująco:

```
0000  60 38 e0 8c d0 c4 3c 22  fb 6f 70 11 08 00 45 00   `8····<" ·op···E·
0010  00 34 00 00 40 00 40 06  a1 af c0 a8 64 a0 11 fd   ·4··@·@· ····d···
0020  61 cf f5 6c 00 50 cd 97  f3 05 92 a9 ed 20 80 10   a··l·P·· ····· ··
0030  00 44 29 f8 00 00 01 01  08 0a 49 9c 4a a0 f2 76   ·D)····· ··I·J··v
0040  f6 93                                              ··
```

Powyższy rysunek przedstawia szesnastkową (heksadecymalną) reprezentację pakietu. Nagłówek IP został przedstawiony na podstawie elementów nagłówka przedstawionych na rysunku 9.3, pokazując, które wartości odpowiadają poszczególnym polom. Podział pól nagłówka IP na komponenty przedstawia się następująco:

• 4: Wersja

• 5: Długość nagłówka internetowego

• 00: Zróżnicowane usługi

• 00 34: Długość całkowita

• 00 00: Identyfikacja

• 40: Flagi

• 00: Przesunięcie fragmentu

• 40: Czas życia

• 06: Protokół

• a1 af: Suma kontrolna nagłówka

• c0 a8 64 a0: Adres źródłowy

• 11 fd 61 cf: Adres docelowy

Teraz, gdy omówiliśmy nagłówek IP, który stanowi część większości komunikacji sieciowej, przyjrzyjmy się drugiej połowie komunikacji IP, czyli części TCP protokołu TCP/IP.

Wprowadzenie do TCP

Model OSI, który omówimy później w tej sekcji, składa się z siedmiu warstw. Warstwa 4 modelu jest zarezerwowana dla protokołów transportowych, takich jak TCP, i zapewnia niezawodną usługę transportu zorientowanego na połączenie. Funkcje TCP zdefiniowano w dokumencie RFC 9293, https://datatracker.ietf.org/doc/html/rfc9293. Zawarty w tym dokumencie protokół TCP charakteryzuje się następującymi głównymi cechami:

• Niezawodność połączenia: klienci i serwery TCP uruchamiają i kończą sesje TCP w określonej kolejności, zmniejszając ryzyko utraty pakietów po nawiązaniu połączenia i gwarantując bezproblemowe zakończenie sesji.

• Kolejność i sekwencjonowanie pakietów: pakiety TCP otrzymują unikalne numery zwane numerami sekwencyjnymi. Dzieje się tak, ponieważ pakiety mogą docierać w niewłaściwej kolejności z powodu routingu. Po dotarciu pakietów do komputera hosta segmenty są ponownie składane z wykorzystaniem numerów sekwencyjnych we właściwej kolejności. Po zakończeniu segment jest przesyłany do aplikacji warstwy wyższej.

• Retransmisja segmentu: Czasami pakiety są tracone podczas transmisji. Może to być spowodowane przekroczeniem limitu czasu pakietów lub błędami sieciowymi. Jeśli host wysyłający nie otrzyma potwierdzenia (ACK) pakietu, uznaje się go za utracony i natychmiast retransmituje. Retransmisje są obliczane na podstawie czasu odbioru każdego segmentu wysłanego przez TCP.

• Sprawdzanie segmentu: TCP weryfikuje zarówno dane w pakietach TCP, jak i nagłówek TCP. Odbywa się to za pomocą pola sumy kontrolnej; wszelkie segmenty, które zmienią się podczas transportu, nie przejdą przez sumę kontrolną i zostaną wyeliminowane.

• Kontrola przepływu wysyłania i odbierania: TCP ma możliwość powiadomienia zdalnego hosta o ilości danych, jaką host może zaakceptować w danym momencie. Zapobiega to przeciążeniu hosta danymi. Zarządzanie przepływem jest kontrolowane za pomocą pola maksymalnego rozmiaru segmentu (MSS), znanego również jako rozmiar segmentu wiadomości, w nagłówku.

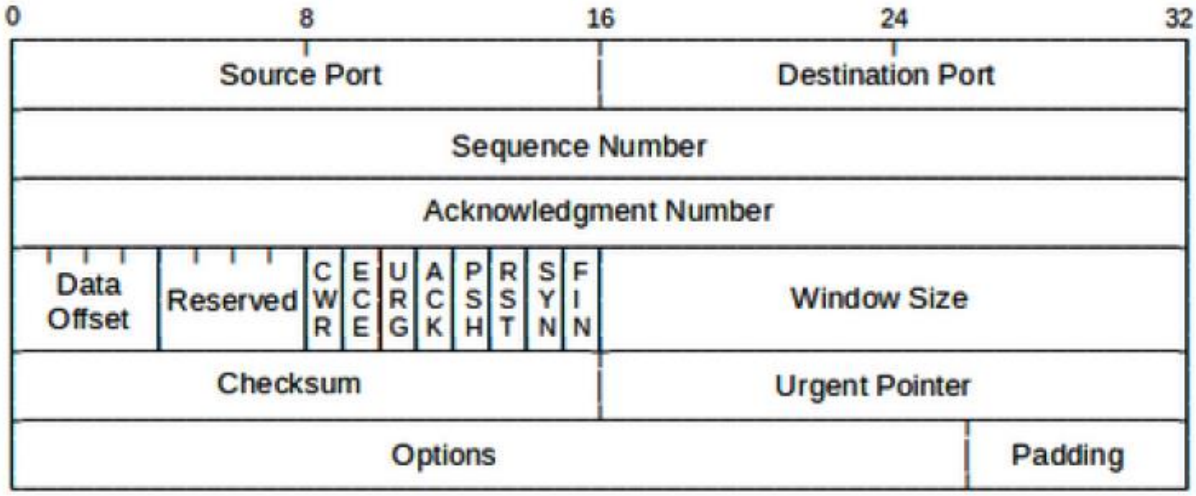
• Multipleksowanie połączeń sesji: Ta funkcja umożliwia wielu niezależnym aplikacjom nawiązywanie połączeń i utrzymywanie ich w oddzielnych konfiguracjach. Aby każda sesja była odrębna od pozostałych, przypisuje się jej unikalny numer portu. Porty te, znane jako porty eteryczne, można obserwować w systemie Windows za pomocą polecenia netstat, jak pokazano na poniższym rysunku:

```
C:\Users\vagrant>netstat -an

Active Connections

  Proto  Local Address          Foreign Address        State
  TCP    0.0.0.0:21             0.0.0.0:0              LISTENING
  TCP    0.0.0.0:22             0.0.0.0:0              LISTENING
  TCP    0.0.0.0:80             0.0.0.0:0              LISTENING
  TCP    0.0.0.0:135            0.0.0.0:0              LISTENING
  TCP    0.0.0.0:445            0.0.0.0:0              LISTENING
  TCP    192.168.100.213:139    0.0.0.0:0              LISTENING
  TCP    192.168.100.213:9300   192.168.100.213:49181  ESTABLISHED
  TCP    192.168.100.213:9300   192.168.100.213:49182  ESTABLISHED
  TCP    192.168.100.213:9300   192.168.100.213:49183  ESTABLISHED
  TCP    192.168.100.213:9300   192.168.100.213:49184  ESTABLISHED
  TCP    192.168.100.213:9300   192.168.100.213:49185  ESTABLISHED
  TCP    192.168.100.213:9300   192.168.100.213:49186  ESTABLISHED
  TCP    192.168.100.213:9300   192.168.100.213:49187  ESTABLISHED
  TCP    192.168.100.213:9300   192.168.100.213:49188  ESTABLISHED
  TCP    192.168.100.213:9300   192.168.100.213:49189  ESTABLISHED
  TCP    192.168.100.213:9300   192.168.100.213:49190  ESTABLISHED
  TCP    192.168.100.213:9300   192.168.100.213:49191  ESTABLISHED
  TCP    192.168.100.213:9300   192.168.100.213:49192  ESTABLISHED
  TCP    192.168.100.213:9300   192.168.100.213:49193  ESTABLISHED
  TCP    192.168.100.213:49181  192.168.100.213:9300   ESTABLISHED
  TCP    192.168.100.213:49182  192.168.100.213:9300   ESTABLISHED
  TCP    192.168.100.213:49183  192.168.100.213:9300   ESTABLISHED
  TCP    192.168.100.213:49184  192.168.100.213:9300   ESTABLISHED
  TCP    192.168.100.213:49185  192.168.100.213:9300   ESTABLISHED
  TCP    192.168.100.213:49186  192.168.100.213:9300   ESTABLISHED
  TCP    192.168.100.213:49187  192.168.100.213:9300   ESTABLISHED
  TCP    192.168.100.213:49188  192.168.100.213:9300   ESTABLISHED
  TCP    192.168.100.213:49189  192.168.100.213:9300   ESTABLISHED
  TCP    192.168.100.213:49190  192.168.100.213:9300   ESTABLISHED
  TCP    192.168.100.213:49191  192.168.100.213:9300   ESTABLISHED
  TCP    192.168.100.213:49192  192.168.100.213:9300   ESTABLISHED
  TCP    192.168.100.213:49193  192.168.100.213:9300   ESTABLISHED
  TCP    192.168.100.213:49270  52.202.51.185:80       ESTABLISHED
```

Teraz, gdy wiemy już nieco więcej o protokole TCP, zobaczmy, jak wygląda jego logiczny układ:



Po przejrzeniu przechwyconego pakietu nagłówek TCP, który znajduje się o jedną warstwę niżej niż nagłówek IP, będzie wyglądał następująco:

```
0000  60 38 e0 8c d0 c4 3c 22  fb 6f 70 11 08 00 45 00
0010  00 40 00 00 40 00 40 06  51 2b c0 a8 64 a0 48 1a
0020  7c 2a e3 16 01 bb 27 20  f3 b6 00 00 00 00 b0 02
0030  ff ff 76 4a 00 00 02 04  05 b4 01 03 03 06 01 01
0040  08 0a f5 2c e2 4e 00 00  00 00 04 02 00 00
```

Powyższy rysunek przedstawia szesnastkową reprezentację pakietu. Nagłówek TCP został przedstawiony na podstawie elementów nagłówka pokazanych na rysunku 9.6, pokazując, które wartości odpowiadają poszczególnym polom. Rozłóżmy pola nagłówka TCP na ich składniki:

• e3 16: Port źródłowy (58134)

• 01 bb: Port docelowy (443)

• 27 20 f3 b6: Numer sekwencyjny (656470966)

• 00 00 00 00: (0)

• b0 02: Flagi (SYN)

• ff ff: Okno (65535)

• 76 4a: Suma kontrolna

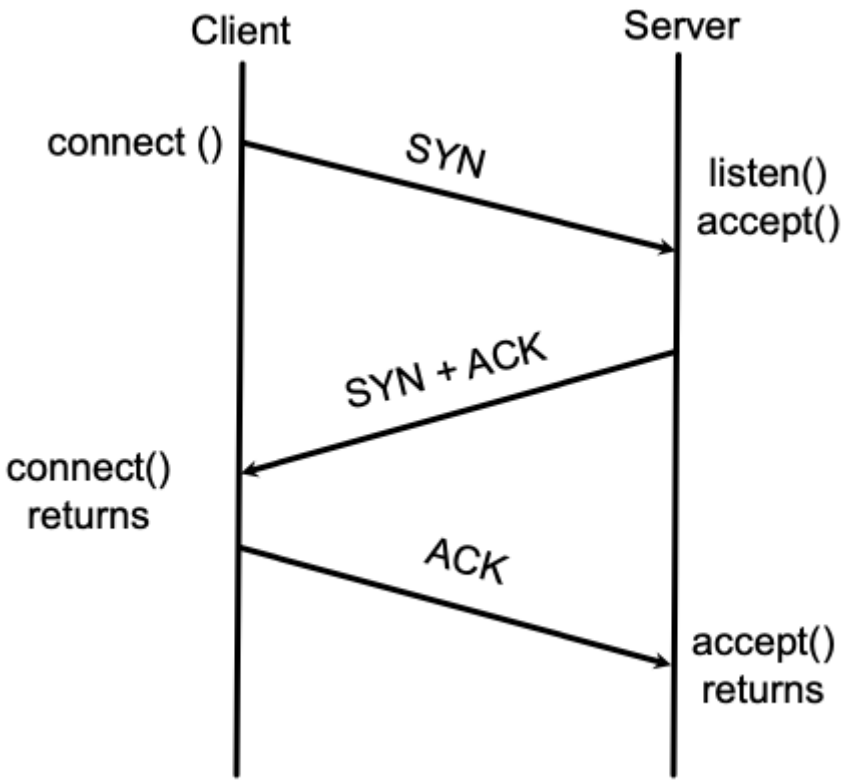
• 00 00: Pilny wskaźnik (0)

• 02 04 05 b4 01 03 03 06 01 01 08 0a f5 2c e2 4e 00 00 00 00 04 02 00 00: Opcje

Teraz, gdy poznaliśmy nagłówki IP i TCP, przyjrzyjmy się, jak komunikacja jest inicjowana poprzez potrójne uzgadnianie

Uzgadnianie trójstronne

Uzgadnianie trójstronne to sposób rozpoczynania komunikacji opartej na protokole TCP. Podczas nawiązywania połączenia system najpierw inicjuje połączenie TCP, wysyłając segment TCP SYN (synchronizacji) do określonego portu docelowego na serwerze docelowym. W większości przypadków porty, z którymi próbuje się nawiązać połączenie, są powiązane z usługą działającą na serwerze. Dodatkowo, określone usługi są przypisane do określonych portów; porty te są często nazywane portami dobrze znanymi. Jeśli serwer może zaakceptować żądanie, wysyła odpowiedź synchronizacji/potwierdzenia (SYN/ACK) na adres IP klienta i port źródłowy (co kończy proces określany jako półotwarte połączenie TCP). Po zakończeniu konfiguracji połączenia TCP nawiązywane jest w pełni otwarte połączenie TCP, a klient inicjujący odpowiada potwierdzeniem ACK, aby rozpocząć sesję. Przyjrzyjmy się diagramowi sekwencji uzgadniania trójstronnego:



Po utworzeniu sesji każdy host będzie śledził transmisje segmentów za pomocą numerów sekwencyjnych, sum kontrolnych, flag i rozmiaru okna. Zmniejsza to utratę pakietów, chroni przepływ pakietów poprzez zapobieganie niepotrzebnym retransmisjom i precyzyjnie sekwencjonuje pakiety w momencie ich dotarcia do każdego komputera odbiorczego. Przykład trzyetapowego uzgadniania z przechwyceniem pakietu przedstawiono na poniższym rysunku:

Source	Destination	Protocol	Info
192.168.100.160	142.250.189.132	TCP	52845 → 443 [SYN] Seq=0
142.250.189.132	192.168.100.160	TCP	443 → 52845 [SYN, ACK]
192.168.100.160	142.250.189.132	TCP	52845 → 443 [ACK] Seq=1

W tym momencie dane mogą być przesyłane w obie strony, o ile istnieją dane do przesłania. Może to być faktyczny transfer danych lub prosta interakcja ze stroną internetową na serwerze. Po zakończeniu komunikacji klient wysyła flagę zakończenia (FIN), po czym serwer potwierdza ją flagą FIN i ACK, a następnie zamyka sesję po stronie serwera, zwalniając zasoby przydzielone sesji do innych zadań. Przyjrzyjmy się teraz innemu protokołowi komunikacyjnemu – UDP.

UDP

UDP działa niemal zupełnie odwrotnie niż TCP. Ze względu na sposób działania TCP programiści odkryli, że jest on kosztowny pod względem ruchu sieciowego i powolny z powodu narzutu wymaganego w TCP dla zapewnienia niezawodności. Programiści potrzebowali czegoś prostego i szybkiego, gdzie narzut TCP nie byłby wymagany. Dodatkowo, akceptowalna byłaby sytuacja, gdyby niektóre pakiety nie dotarły do celu, podczas gdy cała wiadomość zachowałaby swoją integralność, na przykład strumień wideo. W tym przykładzie, po usunięciu jednej lub dwóch klatek, obraz nadal wyglądałby tak samo. Protokół UDP jest tak prosty, że nie ma o nim wiele do powiedzenia, ale oto podstawowe kroki jego działania:

1. Transfer danych w warstwie wyższej: aplikacja wysyła komunikat do usługi UDP.

2. Kapsułkowanie komunikatu: komunikat jest opakowywany lub kapsułkowany.

3. Wysyłanie komunikatu: komunikat jest przekazywany do warstwy IP w celu transmisji.

Poniższy rysunek przedstawia logiczny układ pakietu UDP i rozmieszczenie granic bajtów między poszczególnymi polami:

Jak widać, jest to bardzo proste w porównaniu z protokołem TCP. Poniższy rysunek pokazuje, jak może wyglądać pakiet UDP po przechwyceniu go za pomocą narzędzia do przechwytywania pakietów, takiego jak Wireshark:

```
0000  60 38 e0 8c d0 c4 3c 22  fb 6f 70 11 08 00 45 00
0010  00 3f d6 50 00 00 40 11  5a 6b c0 a8 64 a0 c0 a8
0020  64 01 d6 9c 00 35 00 2b  24 d8 d1 a4 01 00 00 01
0030  00 00 00 00 00 00 06 65  76 73 6e 73 35 06 69 64
0040  72 69 76 65 03 63 6f 6d  00 00 01 00 01
```

Podział pól nagłówka UDP na komponenty przedstawia się następująco:

• d6 9c: Port źródłowy (54940)

• 00 35: Port docelowy (53)

• 00 2b: Długość (43)

• 24 d8: Suma kontrolna

• d1 – 01: Dane

Najważniejszą zaletą protokołu UDP jest jego szybkość, lekkość i doskonałe możliwości komunikacji, w której utracony pakiet lub brakująca sekwencja nie wpływają na ogólną komunikację. Do popularnych usług UDP należą DNS (Domain Name Service), DHCP (Dynamic Host Configuration Protocol) oraz usługi strumieniowe. Przejdźmy teraz do wspólnego protokołu dla wszystkich maszyn – ICMP (Internet Control Message Protocol).

ICMP

ICMP to protokół dostarczania komunikatów o błędach. Jest to bardzo prosty protokół typu żądanie-odpowiedź, który charakteryzuje się następującymi unikalnymi cechami:

• Nie używa portów.

• Nie rozpoznaje relacji klient-serwer.

• Nie gwarantuje dostarczenia.

• Może wykonywać transmisje rozgłoszeniowe.

Jak więc działa protokół ICMP? Po pierwsze, każdy host nasłuchuje żądań ICMP i jest on wykorzystywany do zgłaszania błędów, takich jak brak dostępu do hosta, zablokowanie dostępu administratora i przekroczenie czasu transmisji. Poniższy rysunek przedstawia przykład braku dostępu do hosta:

```
C:\Users\vagrant>ping 4.2.2.1

Pinging 4.2.2.1 with 32 bytes of data:
Reply from 192.168.255.2: Destination host unreachable.
Reply from 192.168.255.2: Destination host unreachable.
Reply from 192.168.255.2: Destination host unreachable.
Reply from 192.168.255.2: Destination host unreachable.

Ping statistics for 4.2.2.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

C:\Users\vagrant>_
```

Jak sama nazwa wskazuje, host jest niedostępny, ale skąd pochodzi komunikat? Może on pochodzić z jednego z dwóch miejsc:

• Jeśli host znajduje się w tym samym segmencie sieci co maszyna żądająca, odpowiedź zostanie wygenerowana przez maszynę żądającą.

• Jeśli host znajduje się w innej sieci, do której żądanie będzie musiało zostać skierowane, komunikat o błędzie zostanie wysłany z routera. Ten komunikat o błędzie może w rzeczywistości oznaczać, że host jest niedostępny lub że występuje problem z siecią między maszyną żądającą a hostem, a komunikat o błędzie „host niedostępny” jest wynikiem tego problemu.

Przyjrzyjmy się teraz różnicom między dwoma omówionymi protokołami komunikacyjnymi.

Porównanie TCP i UDP

TCP i UDP to dwa główne protokoły warstwy transportowej, które ułatwiają komunikację między urządzeniami w sieci. Kluczowa różnica polega na ich naturze połączeniowej i bezpołączeniowej. TCP jest zorientowany połączeniowo, zapewniając niezawodne i uporządkowane dostarczanie danych poprzez nawiązanie połączenia między nadawcą a odbiorcą. Wykorzystuje mechanizmy takie jak kontrola przepływu, wykrywanie błędów i retransmisja utraconych pakietów, aby zagwarantować dokładne przesyłanie informacji. Ta niezawodność sprawia, że TCP nadaje się do aplikacji, w których integralność danych jest kluczowa, takich jak przesyłanie plików, przeglądanie stron internetowych i poczta e-mail, ponieważ utrzymuje stan. Jednak ta niezawodność ma swoją cenę; połączenia te wymagają większego narzutu i mogą obniżać niezawodność serwerów przy przeciążonych zasobach i obsłudze dużej liczby połączeń. Z kolei UDP jest bezpołączeniowy i koncentruje się na szybkim dostarczaniu danych, bez narzutu związanego z nawiązywaniem połączenia i odzyskiwaniem danych po błędzie. Chociaż UDP poświęca część niezawodności na rzecz szybkości, dobrze nadaje się do aplikacji czasu rzeczywistego, takich jak strumieniowanie głosu i wideo, gry online i rozwiązywanie

nazw domen (DNS). UDP jest lżejszy, co czyni go idealnym wyborem w scenariuszach, w których można tolerować sporadyczną utratę pakietów, a nacisk kładziony jest na niskie opóźnienia. Wybór między TCP a UDP zależy od specyficznych wymagań aplikacji, równoważąc potrzebę niezawodności z potrzebą szybszej transmisji danych. Poniższa tabela przedstawia różnice:

TCP : UDP

Połączony : Bezpołączeniowy

Stanowy :

Strumień bajtów : Pakiet/datagram

Uporządkowane dostarczanie danych : Brak gwarancji sekwencji

Niezawodny : Stratny

Bezbłędny : Odrzucane pakiety błędów

Uzgadnianie : Brak uzgadniania

Kontrola przepływu : Brak kontroli przepływu

Stosunkowo wolny : Stosunkowo szybki

Punkt-punkt : Obsługuje multicast

Przykłady: Hypertext Transfer Protocol (HTTP), HTTP Secure (HTTPS), File Transfer Protocol (FTP) i Simple Mail Transfer

Protocol (SMTP) -

Przykłady: DNS, DHCP, Voice Over IP (VoIP) i strumieniowanie wideo

Przyjrzyjmy się pokrótce niektórym portom używanym przez wspomniane aplikacje.

Znane porty

Wcześniej, w sekcji „Uzgadnianie trójstronne”, wspomnieliśmy o znanych portach, więc przyjrzyjmy się im bliżej. Koncepcja znanych portów polega na ustanowieniu lub wyznaczeniu konkretnych numerów portów dla popularnych usług, takich jak port 80 dla usług sieciowych HTTP. Komputery mogą obsługiwać teoretycznie maksymalnie 65 535 portów. Organizacja IANA (Internet Assigned Numbers Authority) zarezerwowała pierwsze 1024 porty dla popularnych usług. Zrobiono to, aby zminimalizować ryzyko związane z typowymi działaniami sieciowymi. Poniżej przedstawiono niektóre z najczęściej spotykanych portów:

• Transfer danych FTP: 20

• Kontrola poleceń FTP: 21

• Secure Shell (SSH): 22

• Telnet: Usługa zdalnego logowania i niezaszyfrowane wiadomości tekstowe: 23

• Routing poczty e-mail SMTP: 25

• Usługa DNS: 53

• HTTP używany w sieci WWW: 80

• Protokół czasu sieciowego (NTP): 123

• Protokół dostępu do wiadomości internetowych (IMAP) do zarządzania pocztą cyfrową: 143

• Protokół SNMP (Simple Network Management Protocol): 161

• HTTPS i HTTP przez TLS/SSL: 443

Zrozumienie ataków protokołowych

Teraz, gdy znamy już protokoły typowe dla większości, jeśli nie wszystkich, maszyn uczestniczących w sieci opartej na protokole I, przyjrzyjmy się niektórym możliwym atakom, zaczynając od ataków TCP.

Ataki TCP

Jak wspomniano wcześniej, TCP to podstawowy protokół komunikacyjny, który umożliwia niezawodną komunikację sieciową. Jednak sama natura jego konstrukcji sprawia, że jest on podatny na różnego rodzaju ataki wykorzystujące luki w zabezpieczeniach protokołu. Ataki TCP mogą naruszyć poufność, integralność i dostępność danych, stwarzając poważne zagrożenie dla systemów sieciowych. Ataki te mogą obejmować zarówno proste exploity, jak i zaawansowane techniki stosowane przez cyberprzestępców i hakerów. Zrozumienie typowych ataków TCP jest kluczowe dla administratorów sieci i specjalistów ds. bezpieczeństwa, aby wdrożyć skuteczne środki zaradcze i chronić integralność komunikacji sieciowej. Przyjrzyjmy się niektórym z tych ataków:

• Zatrucie pamięci podręcznej ARP: Zatrucie ARP to proces polegający na powiązaniu przez cel nieprawidłowego adresu MAC (Media Access Control) z adresem IP. Adres MAC to adres sprzętowy interfejsu sieciowego. Protokół ARP służy do wysyłania i odpowiadania na komunikaty identyfikacyjne hosta na tym samym przełączniku w sieci. Jednym z możliwych zastosowań jest atak typu „odmowa usługi", w którym ofiara musi skojarzyć bramę z nieprawidłowym adresem MAC. Co więcej, możliwe jest skażenie pamięci podręcznych dwóch celów, tak aby każdy z nich powiązał adres IP drugiego z adresem MAC atakującego. Możliwe jest wykorzystanie tego w ataku typu „man-in-the-middle" (MITM) lub innych atakach polegających na przechwytywaniu sesji.

• Skanowanie TCP SYN, zwane również atakiem SYN flood: W tym przypadku wysyłany jest pakiet TCP z flagą SYN w nagłówku, wskazujący na żądanie połączenia. Na ten pakiet wysyłany jest pakiet SYN/ACK, potwierdzający żądanie synchronizacji. Zasoby systemowe serwera są teraz przydzielane w celu przygotowania się na połączenie przychodzące. Następnie obiekt docelowy oczekuje na ostatnie potwierdzenie ACK, które nigdy nie następuje, aby nawiązać połączenie. Nadużywanie tego procesu może doprowadzić do wyczerpania zasobów serwera, gdy żądania nowych połączeń są wielokrotnie wysyłane, a żadne z nich nie zostaje nawiązane.

• Skanowanie TCP FIN: To kolejny sposób, w jaki atakujący mogą wykorzystać flagi TCP. W tym ataku atakujący wysyła pakiet FIN zamiast pakietu SYN. Gdy host odbierający otrzyma pakiet FIN, gdy wcześniej nie nawiązano żadnej sesji, host wyśle odpowiedź Reset (RST) i ujawni obecność hosta pod tym adresem. Niektóre filtry pakietów, w tym zapory sieciowe, wykrywają ten atak i mogą nie zezwolić na ukończenie tej techniki.

• TCP RST: Flaga RST w pakietach TCP może być ustawiona tak, aby wskazywała na konieczność zamknięcia połączenia. Nie jest to to samo, co typowe zamknięcie połączenia, które sugerowałby pakiet TCP FIN. Pakiet RST jest oczywiście przydatny dla atakującego, ponieważ może proaktywnie zamknąć połączenie.

• Skanowanie TCP connect(): W tym przypadku następuje pełne połączenie po zakończeniu pełnego trzyetapowego uzgadniania. Chociaż niekoniecznie jest to atak, jest ono wykorzystywane do wykrywania sieci.

• Atak odbiciowy: Ataki odbiciowe ukrywają źródłowy adres IP atakującego, wykorzystując usługi TCP/UDP stron trzecich do odbijania ruchu atakującego do wybranej ofiary. W ataku odbiciowym atakujący wysyłają sfałszowane pakiety do usługi reflektora, podszywając się pod adres IP docelowej ofiary, pośrednio zalewając ją pakietami odpowiedzi generowanymi przez reflektory. Ponieważ usługa reflektora wykorzystywana w tym ataku może być dowolnym legalnym serwerem, który może odpowiadać na żądania, przeciwdziałanie takim atakom może być bardzo trudne.

• Atak typu teardrop: To kolejny przykład ataku typu „odmowa usługi" (DoS). Odnosi się on do ataku, który próbuje uniemożliwić dostęp do zasobu komputera poprzez zalewanie sieci lub serwera żądaniami i danymi. W przypadku wystąpienia luki w zabezpieczeniach protokołu TCP/IP atakujący może dostarczyć do serwera docelowego pofragmentowane pakiety, których serwer może nie być w stanie ponownie złożyć, co prowadzi do przeciążenia.

To tylko kilka przykładów ataków TCP. Przyjrzyjmy się teraz protokołowi UDP i sprawdźmy, jakie rodzaje ataków można wykorzystać przeciwko niemu.

Ataki UDP

Ze względu na naturę protokołu UDP i sposób jego działania, ataki te są zazwyczaj klasyfikowane jako ataki typu DoS lub rozproszona odmowa usługi (DDoS), zazwyczaj związane z usługą obsługującą komunikację UDP, taką jak VoIP. Klasycznym przykładem jest DNS. Istnieje wiele ataków na DNS, ale najczęściej spotykane to ataki typu DNS amplification, które w rzeczywistości tworzą sytuację DoS. Ataki te obejmują:

• Atak z amplifikacją DNS: Polega on na tym, że atakujący używa otwartych serwerów DNS do wysłania żądania, zazwyczaj dużego, takiego jak informacje o całej domenie, wysyłając odpowiedź na adres podszywającej się ofiary. Żądanie jest wzmacniane, co prowadzi do awarii usługi dla celu.

• Atak typu VoIP flood: W tym ataku celem jest system VoIP. Wysyłanych jest wiele żądań VoIP, które przeciążają system w próbie udzielenia odpowiedzi.

• Atak typu NTP flood: W tym przypadku atak wysyła żądania do publicznie dostępnych serwerów NTP, a celem jest podszywający się adres ofiary.

• Atak typu UDP flood: To najprostszy z ataków; w tym przypadku atakujący lub atakujący wysyła wystarczająco dużą liczbę pakietów pozornych, aby przekroczyć możliwości serwera w zakresie przetwarzania i reagowania.

Istnieje wiele narzędzi, które umożliwiają eksploatację i testowanie protokołu UDP. Jednym z takich narzędzi jest UDP Unicorn, które umożliwia przeprowadzanie testów typu UDP flood. Można je znaleźć na stronie Source Forge pod adresem https://sourceforge.net/projects/udpunicorn/. Przyjrzyjmy się teraz naszemu ostatniemu obszarowi ataków – atakom ICMP.

Ataki ICMP

Podobnie jak dwa pozostałe protokoły, protokół ICMP może być nadużywany na kilka sposobów. Jedną z metod jest wykorzystanie go do celów rozpoznawczych, co zostało omówione w rozdziale 3. Inne rzeczywiste ataki obejmują:

• Atak typu smurf ICMP: W tym ataku żądanie echa ICMP jest kierowane na adres rozgłoszeniowy ze zmodyfikowanym lub sfałszowanym adresem źródłowym. W praktyce oznacza to wysłanie żądania ping do każdego hosta w podsieci, z prośbą o odpowiedź od każdego hosta na podrobiony adres źródłowy, co stwarza sytuację DoS.

• Atak przekierowania ICMP: Przekierowywanie ICMP jest zazwyczaj zadaniem routerów sieciowych lub węzłów niebędących hostami. Atakujący może jednak generować je za pomocą określonego komunikatu, podobnie jak w przypadku pakietów protokołu rozpoznawania adresów (ARP). Cel otrzymuje polecenie zmiany swojej tabeli routingu za pomocą przekierowania ICMP, z typem ICMP 5 i kodem 0. Atakujący może to wykorzystać w ataku DoS, wymuszając przepływ ruchu przez węzeł, który jest wyłączony lub znajduje się pod jego kontrolą.

• Atak typu „source-quench": Atak typu „source-quench" wykorzystuje komunikat ICMP typu 4 i kod 0, który jest sygnałem spowalniającym transmisję, co w efekcie spowalnia maszynę. Zamiast całkowicie odciąć lub zablokować połączenie, może to spowolnić cel, co stanowiłoby bardziej ukryty atak.

• Atak tunelowania ICMP: W tej sytuacji atakujący wykorzystują protokół do maskowania złośliwej aktywności poprzez wstrzykiwanie ukrytych danych do pakietu ICMP. Są one niezwykle trudne do wykrycia.

Istnieje kilka narzędzi, które mogą wykonywać tego typu operacje, w tym icmptunnel autorstwa Dhavala Kapila, który hermetyzuje ruch w pakietach echa ICMP w celu ominięcia portali uwierzytelniających i zapór sieciowych lub ustanowienia szyfrowanego kanału komunikacyjnego.

To tylko kilka sposobów, w jakie atakujący mogą nadużyć protokołu ICMP. Badacz Ofir Arkin opublikował obszerny artykuł na ten temat zatytułowany „Użycie ICMP w skanowaniu". Jego kopię można znaleźć pod adresem https://github.com/PacktPublishing/Hands-On-Ethical-Hacking-Tactics. Teraz, gdy omówiliśmy niektóre sposoby, w jakie atakujący mogą wykorzystać podstawowe protokoły systemu operacyjnego, przyjrzyjmy się pokrótce protokołowi IPv6.

Przegląd IPv6

IPv6 to najnowsza iteracja protokołu internetowego, zaprojektowana w celu rozwiązania ograniczeń związanych z przestrzenią adresową IPv4. IPv6 rozwiązuje ten problem dla urządzeń podłączonych do internetu, oferując znacznie rozszerzoną przestrzeń adresową. Podczas gdy IPv4 opiera się na adresach 32-bitowych, co skutkuje ograniczoną pulą unikalnych adresów, IPv6 wykorzystuje 128-bitowy schemat adresowania, umożliwiając astronomicznie dużą liczbę unikalnych kombinacji. To rozszerzenie nie tylko rozwiązuje problem zbliżającego się wyczerpania adresów IPv4, ale także wprowadza ulepszenia w zakresie wydajności sieci, funkcji bezpieczeństwa i uproszczonej konfiguracji sieci. Wdrażanie protokołu IPv6 było powolnym procesem, nie tylko dla producentów, ale także dla inżynierów sieciowych, którzy mieli wątpliwości co do jego obsługi i implementacji. Ta luka w wiedzy stworzyła atakującym pewne możliwości do wykorzystania. Przyjrzyjmy się, jak konfigurowany jest protokół IPv6, a następnie omówmy niektóre problemy bezpieczeństwa związane z jego użyciem.

Konfiguracja i konfiguracja protokołu IPv6

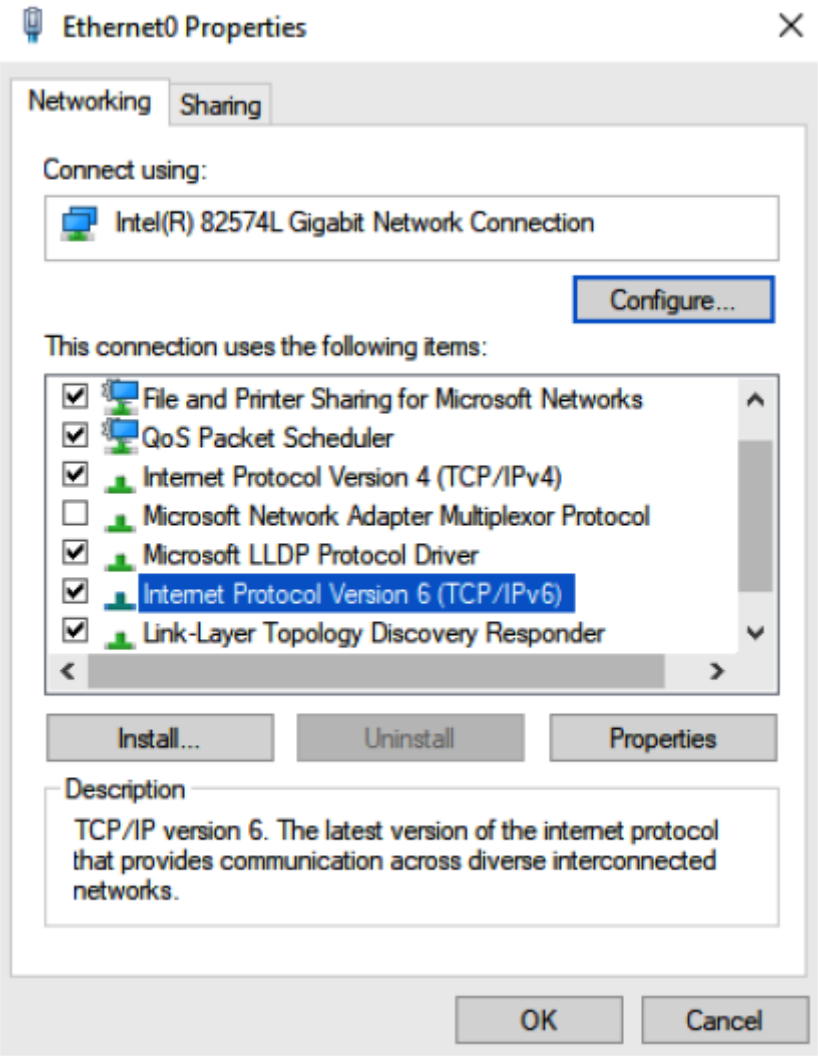
Konfiguracja i konfiguracja protokołu IPv6 są zasadniczo takie same jak w przypadku adresowania IPv4. Aby wykonać tę operację w systemie Windows, wykonaj następujące kroki:

1. Otwórz Centrum sieci i udostępniania w Panelu sterowania.

2. Kliknij opcję Zmień ustawienia karty sieciowej.

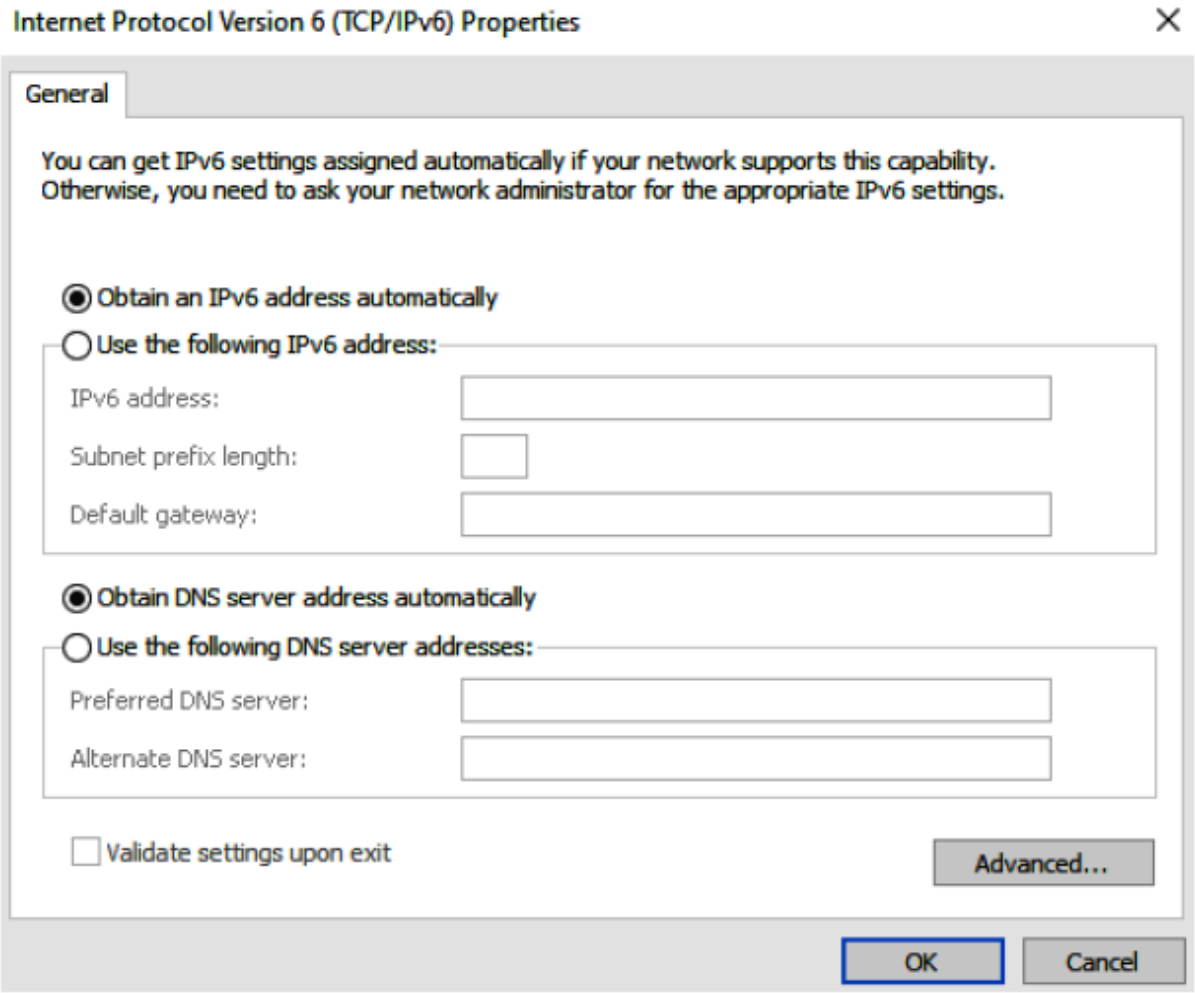
3. Znajdź interesujący Cię interfejs sieciowy, kliknij go prawym przyciskiem myszy i wybierz opcję Właściwości.

4. Następnie znajdź wpis Protokół internetowy w wersji 6 (TCP/IPv6) i upewnij się, że jest on zaznaczony:



5. Zaznacz wpis i kliknij przycisk Właściwości. Spowoduje to przejście do ekranu konfiguracji.

6. Otworzy się nowy ekran umożliwiający dalszą konfigurację, gdzie można wprowadzić statyczne wpisy lub użyć serwera DHCP do przypisania. Przykładowy ekran konfiguracji IPv6 w systemie Windows można zobaczyć na poniższym zrzucie ekranu:



7. Po skonfigurowaniu interfejs będzie obsługiwał sieć IPv6. Aby potwierdzić konfigurację protokołu IPv6, otwórz wiersz poleceń i wpisz polecenie ipconfig, które potwierdzi zmianę konfiguracji dodatkowym wpisem IPv6, jak pokazano na poniższym zrzucie ekranu:

```
C:\Users\vagrant>ipconfig

Windows IP Configuration


Ethernet adapter Local Area Connection 3:

   Connection-specific DNS Suffix  . :
   IPv4 Address. . . . . . . . . . . : 192.168.255.2
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 192.168.255.1

Tunnel adapter isatap.{07247C5D-9A29-4907-8E19-300CD9B7B524}:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :
```

```
C:\Users\vagrant>ipconfig

Windows IP Configuration


Ethernet adapter Local Area Connection 3:

   Connection-specific DNS Suffix  . :
   Link-local IPv6 Address . . . . . : fe80::6143:5c00:a4dc:e1f7
   IPv4 Address. . . . . . . . . . . : 192.168.255.2
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 192.168.255.1

Tunnel adapter isatap.{07247C5D-9A29-4907-8E19-300CD9B7B524}:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :

C:\Users\vagrant>_
```

Wszystkie najpopularniejsze funkcje Nmapa obsługują protokół IPv6. Protokół IPv6 jest obsługiwany poprzez skanowanie ping, skanowanie portów, wykrywanie wersji oraz silnik skryptowy Nmapa. Składnia polecenia pozostaje taka sama jak zwykle, z dodatkiem opcji -6. Oczywiście, jeśli podasz adres zamiast nazwy hosta, musisz użyć terminologii IPv6. Nazwy hostów są zalecane, ponieważ adres może być trudny do zapamiętania i wpisania, na przykład Fe80::6143:5c00:a4dc:e1f7. Jedynym sygnałem z wyników skanowania portów, że protokół IPv6 jest dostępny, jest adres IPv6 w wierszu „inne adresy"; w przeciwnym razie wynik skanowania będzie wyglądał jak każdy inny. Chociaż protokół IPv6 nie zyskał oczekiwanego rozpowszechnienia, jest on wykorzystywany w krajach rozwijających się na całym świecie w przypadku zasobów narażonych na działanie Internetu. Protokół IPv6 może być również widoczny wewnątrz sieci; większość systemów operacyjnych ma włączoną obsługę protokołu IPv6 w ramach domyślnej instalacji. W wielu przypadkach pracownicy IT nie są świadomi działania protokołu IPv6 w środowisku i w związku z tym nie monitorują go pod kątem obecności ruchu sieciowego (IR). W ten sposób cała ta przestrzeń pozostaje dostępna do wykorzystania. Jak więc atakujący może wykorzystać IPv6? Jak wspomniano wcześniej, wiele technik stosowanych w przypadku IPv4 nadal ma zastosowanie; istnieją jednak narzędzia, które wykorzystują IPv6 w swoich atakach. Przyjrzyjmy się temu:

• Organizacja Hacker's Choice https://www.thc.org/ stworzyła zestaw narzędzi do ataków IPv6. Zestaw zawiera ponad 20 narzędzi opracowanych specjalnie do pracy w środowiskach IPv6. Niektóre z nich, takie jak Parasite6, to narzędzia do podszywania się pod sąsiadów (ICMPv6 neighbor solicitation/advertisement spoofers), służące do przeprowadzania ataków MITM. Innym narzędziem jest detect-new-ip6, które wykrywa dołączenie nowych urządzeń IPv6 do sieci i z ich poziomu można rozpocząć skanowanie. Zestaw narzędzi jest przechowywany w serwisie GitHub i można go znaleźć tutaj: https://github.com/vanhauser-thc/thc-ipv6.

• Nie zapominajmy o Metasploit; po wyszukaniu IPv6 w konsoli Metasploit, otrzymasz co najmniej 64 wpisy związane z tym tematem. Zdecydowana większość z nich dotyczy dostarczania danych i wiązania z adresami IPv6. Jest jednak co najmniej osiem wpisów dotyczących exploitów i wstrzyknięć. Poniższy rysunek przedstawia część wsparcia dla IPv6 w Metasploit:

```
msf6 > search exploit IPv6

Matching Modules
================

   #  Name                                                    Rank    Check  Description
   -  ----                                                    ----    -----  -----------
   0  exploit/freebsd/local/ip6_setpktopt_uaf_priv_esc        great   Yes    FreeBSD ip6_setpktopt Use-After-Free Privilege Escalation
   1  payload/windows/patchupdllinject/bind_ipv6_tcp          normal  No     Windows Inject DLL, Bind IPv6 TCP Stager (Windows x86)
   2  payload/windows/patchupdllinject/bind_ipv6_tcp_uuid     normal  No     Windows Inject DLL, Bind IPv6 TCP Stager with UUID Support (Windows x86)
   3  payload/windows/patchupdllinject/reverse_ipv6_tcp       normal  No     Windows Inject DLL, Reverse TCP Stager (IPv6)
   4  payload/windows/peinject/bind_ipv6_tcp                  normal  No     Windows Inject PE Files, Bind IPv6 TCP Stager (Windows x86)
   5  payload/windows/peinject/bind_ipv6_tcp_uuid             normal  No     Windows Inject PE Files, Bind IPv6 TCP Stager with UUID Support (Windows x86)
   6  payload/windows/peinject/reverse_ipv6_tcp               normal  No     Windows Inject PE Files, Reverse TCP Stager (IPv6)
   7  payload/windows/x64/peinject/bind_ipv6_tcp              normal  No     Windows Inject Reflective PE Files, Windows x64 IPv6 Bind TCP Stager
   8  payload/windows/x64/peinject/bind_ipv6_tcp_uuid         normal  No     Windows Inject Reflective PE Files, Windows x64 IPv6 Bind TCP Stager with UUID Support

Interact with a module by name or index. For example info 8, use 8 or use payload/windows/x64/peinject/bind_ipv6_tcp_uuid

msf6 >
```

W obliczu tego typu ataków przyjrzyjmy się, co mogą zrobić obrońcy, aby chronić swoje systemy.

Obrona sieci IPv4

Istnieje wiele praktyk, które obrońcy mogą zastosować w celu ochrony sieci IPv4, ponieważ ten typ sieci istnieje już od dawna. Obrona sieci IPv4 zaczyna się od wielopłaszczyznowego podejścia:

1. Zapory sieciowe: Wdrażanie zapór sieciowych w celu filtrowania i kontrolowania ruchu przychodzącego i wychodzącego w oparciu o z góry określone reguły bezpieczeństwa pomaga zapobiegać nieautoryzowanemu dostępowi i blokować złośliwe pakiety.

2. Listy kontroli dostępu (ACL): Listy ACL są używane na routerach do definiowania i egzekwowania zasad, regulujących dostęp do zasobów sieciowych na podstawie adresów IP, portów i protokołów, kontrolując w ten sposób, kto może komunikować się z określonymi zasobami sieciowymi.

3. Systemy wykrywania włamań (IDS) i systemy zapobiegania włamaniom (IPS): System IDS monitoruje ruch sieciowy pod kątem podejrzanej aktywności lub znanych wzorców ataków, podczas gdy system IPS może aktywnie blokować lub zapobiegać wykrytym zagrożeniom w czasie rzeczywistym.

4. Segmentacja sieci: Podział sieci na mniejsze, odizolowane segmenty lub podsieci pomaga ograniczyć naruszenia i wpływ potencjalnych ataków poprzez kontrolowanie przepływu ruchu między różnymi częściami sieci.

5. Wirtualne sieci prywatne (VPN): Wdrożenie sieci VPN umożliwia bezpieczny zdalny dostęp do sieci poprzez szyfrowanie ruchu między zdalnymi użytkowników/urządzeń i sieci korporacyjnej, chroniąc poufność i integralność danych.

6. Translacja adresów sieciowych (NAT): NAT ukrywa wewnętrzne adresy IP przed sieciami zewnętrznymi, zapewniając dodatkową warstwę bezpieczeństwa poprzez zaciemnienie wewnętrznej struktury sieci i utrudniając atakującym identyfikację potencjalnych celów.

7. Podsłuchiwanie DHCP: Podsłuchiwanie DHCP ogranicza ryzyko ataków z użyciem nieautoryzowanych serwerów DHCP lub innych ataków opartych na DHCP poprzez inspekcję komunikatów DHCP, weryfikując w ten sposób legalność serwerów DHCP i zapobiegając nieautoryzowanemu przypisywaniu adresów IP urządzeniom w sieci.

8. Rozszerzenia zabezpieczeń DNS (DNSSEC): Jest to sposób na zapewnienie autentyczności odpowiedzi DNS, uniemożliwiający atakującym manipulowanie żądaniami i odpowiedziami w trakcie przesyłania.

9. Bezpieczne protokoły routingu: Stosowanie bezpiecznych protokołów routingu, takich jak Secure Neighbor Discovery (SEND) i Border Gateway Protocol Security (BGPsec), pomaga chronić przed przechwytywaniem tras, podszywaniem się i innymi atakami związanymi z routingiem.

10. Regularne zarządzanie poprawkami: Zapewnienie terminowej instalacji poprawek bezpieczeństwa i aktualizacji dla urządzeń sieciowych. Systemy operacyjne i aplikacje pomagają w usuwaniu znanych luk w zabezpieczeniach i ograniczaniu ryzyka ich wykorzystania przez atakujących.

11. Szyfrowanie: Wdrożenie protokołów szyfrowania, takich jak Internet Protocol Security (IPsec), w celu zabezpieczenia kanałów komunikacyjnych między urządzeniami sieciowymi a punktami końcowymi zwiększa poufność i integralność danych, szczególnie w przypadku poufnych informacji przesyłanych przez sieci publiczne.

12. Monitorowanie i rejestrowanie sieci: Ciągłe monitorowanie ruchu sieciowego, zdarzeń bezpieczeństwa i logów systemowych umożliwia wczesne wykrywanie anomalii, podejrzanych działań i potencjalnych incydentów bezpieczeństwa, ułatwiając szybką reakcję i działania łagodzące.

Wdrażając te zabezpieczenia IPv4 w połączeniu z kompleksowymi politykami i praktykami bezpieczeństwa, organizacje mogą skutecznie ograniczać różne zagrożenia i poprawiać ogólny poziom bezpieczeństwa swoich sieci. Wreszcie, wszystkie te zabezpieczenia można zastosować w sieciach IPv6. Przyjrzyjmy się różnicom występującym w sieciach IPv6.

Obrona sieci IPv6

Co mogą zrobić obrońcy, aby chronić swoje systemy? Cóż, w przeszłości można było zaczynać od granicy i stopniowo ją zmniejszać. Problem polega na tym, że obecnie granica jest mniej zdefiniowana niż kiedyś, ze względu na pracę zdalną, wirtualizację i technologie chmurowe. Znacznie trudniej jest określić, gdzie kończy się internet, a zaczyna infrastruktura, za której ochronę odpowiadasz. W przypadku sieci obsługujących IPv6 problem ten pogłębia nie tylko brak obsługi hosta, ale także routery i zapory sieciowe obsługujące jednocześnie IPv4 i IPv6. Taka konfiguracja jest również znana jako konfiguracja dwustosowa, w której IPv6 jest powiązane z tym samym interfejsem co IPv4. W wielu przypadkach personel IT nie jest świadomy drugiej konfiguracji i pozostawia ją w stanie domyślnym. Co gorsza, dane telemetryczne nie są gromadzone ani monitorowane pod kątem złośliwej aktywności. Obrona sieci korzystających z protokołu IPv6 wymaga wdrożenia kombinacji najlepszych praktyk, środków bezpieczeństwa i strategii monitorowania. Wiele standardowych najlepszych praktyk nadal ma zastosowanie do protokołu IPv6, podobnie jak w przypadku protokołu IPv4, w tym:

1. Znajomość protokołu IPv6, jego działania, implementacji i sposobów monitorowania.

2. Planowanie adresów IPv6:

3. Opracuj kompleksowy plan adresów IPv6, aby zminimalizować ryzyko skanowania i enumeracji adresów.

4. Konfiguracja zapory sieciowej:

5. Wdróż zapory sieciowe z kontrolą stanu i bez niej, aby filtrować przychodzący i wychodzący ruch IPv6.

6. Ogranicz niepotrzebne usługi i porty, aby zminimalizować powierzchnię ataku.

7. Zabezpiecz konfiguracje routerów:

8. Regularnie aktualizuj oprogramowanie układowe routerów, aby łatać znane luki w zabezpieczeniach.

9. Wyłącz niepotrzebne usługi i protokoły na routerach.

10. Włącz bardzo restrykcyjne zasady dostępu do konfiguracji routerów.

11. Korzystaj z systemów wykrywania i zapobiegania włamaniom (IDPS):

12. Używaj systemu IDPS zgodnego z IPv6 do monitorowania ruchu sieciowego.

13. Skonfiguruj reguły alertów w oparciu o znane wzorce złośliwych ataków IPv6.

14. Oceny bezpieczeństwa IPv6:

15. Przeprowadzaj regularne oceny bezpieczeństwa, zarówno automatyczne, jak i ręczne, w celu wykrycia luk w zabezpieczeniach.

Łącząc te środki, organizacje mogą znacząco poprawić poziom bezpieczeństwa swoich sieci IPv6. Regularna aktualizacja i dostosowywanie tych strategii w oparciu o pojawiające się zagrożenia i luki w zabezpieczeniach ma kluczowe znaczenie dla utrzymania bezpieczeństwa sieci.

Laboratorium

W tym laboratorium przeprowadzimy atak DoS na maszynę z systemem Windows Metasploitable i przeprowadzimy analizę pliku przechwytywania pakietów (PCAP).

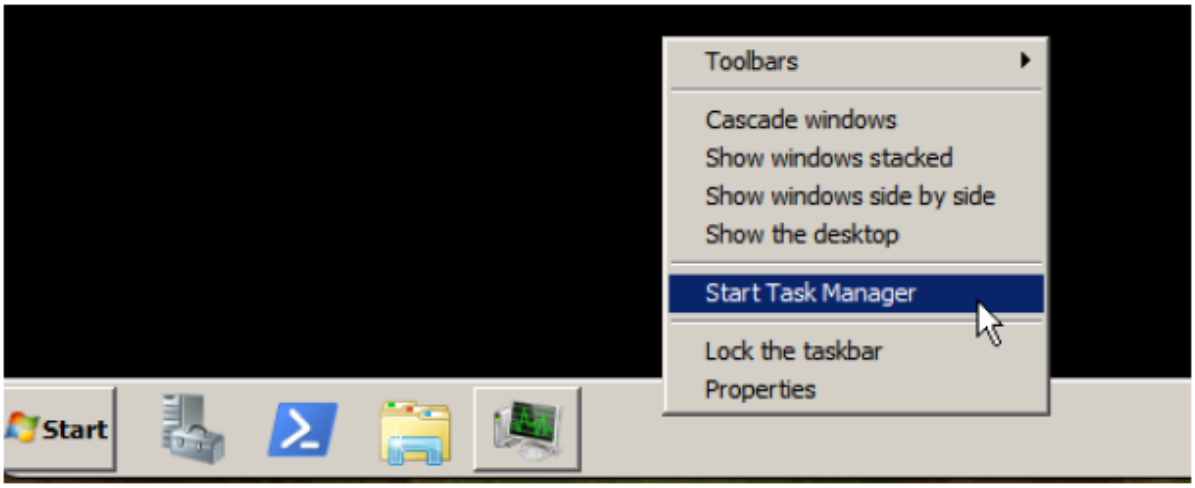
Ćwiczenie 1

W tym ćwiczeniu przeprowadzimy atak DoS na nasz system Windows za pomocą ataku SYN flood:

1. Uruchom Kali Linux i system Windows Metasploitable.

2. Zaloguj się na komputerze z systemem Windows i otwórz Menedżera zadań.

3. Możesz to zrobić, klikając prawym przyciskiem myszy pasek u dołu ekranu, po prawej stronie przycisku Start:



4. Po załadowaniu kliknij kartę „Sieć”, aby wyświetlić wykres ruchu sieciowego.

5. Teraz wróć do Kali i zaloguj się, aby przeprowadzić atak.

6. Otwórz wiersz poleceń i wpisz następujące polecenie:

sudo hping3 -c 15000 -d 500 -S -w 64 -p 80 --flood --rand-source

192.168.255.2

W ten sposób zostanie załadowana aplikacja o nazwie hping3, która jest generatorem pakietów. Następnie przekazujemy parametry ataku:

• -c 15000: To jest liczba wysyłanych pakietów

• -d 500: To jest rozmiar wysyłanych pakietów w bajtach

• -S: Wyślij jako pakiet SYN

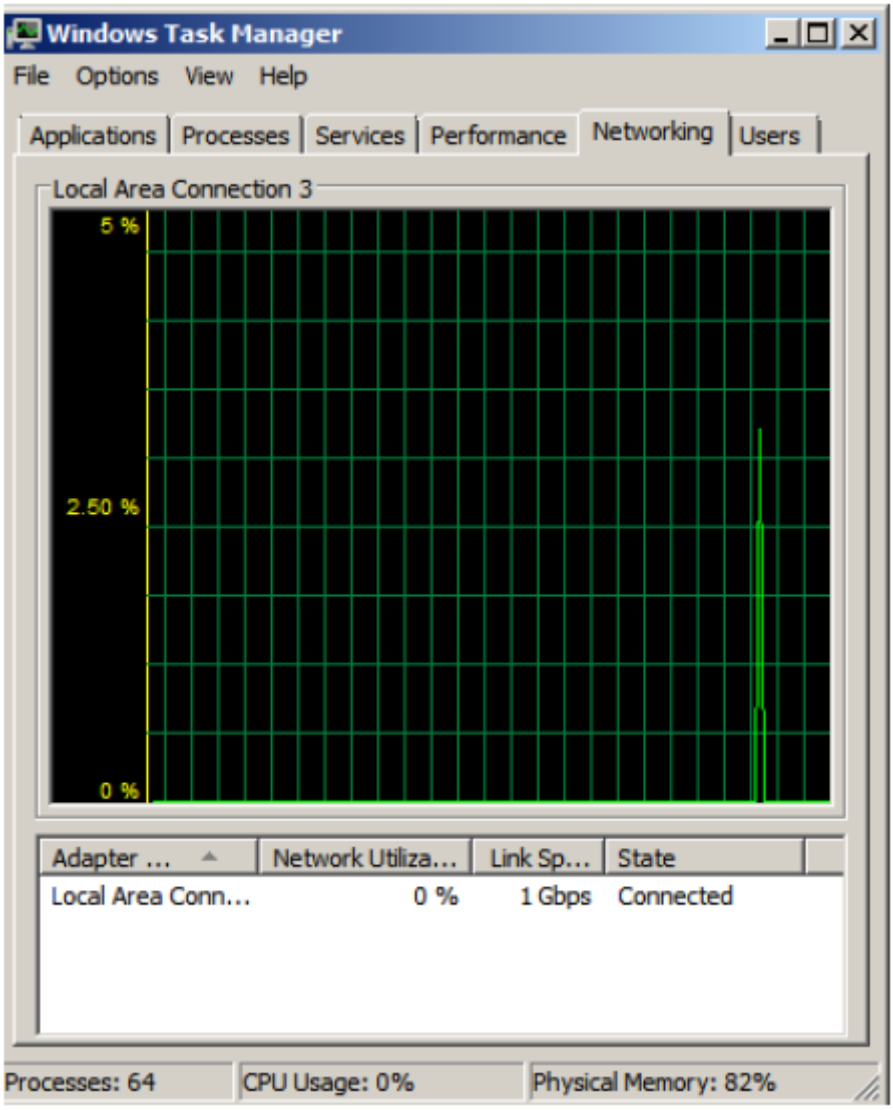
• -w 64: To jest rozmiar okna TCP

• -p 80: Port, do którego należy wysłać pakiet

• --flood: Zapełnia interfejs

• --rand-source: Generuje fałszywe adresy źródłowe

Uruchomienie tego polecenia na komputerze z systemem Windows spowoduje jego natychmiastowe zablokowanie. Uruchom polecenie przez około 30 sekund na komputerze z systemem Kali i użyj kombinacji klawiszy Ctrl + C, aby je zatrzymać. Wróć do komputera z systemem Windows i przejrzyj wykres; zobaczysz, że odnotowano gwałtowny wzrost ruchu. Będzie on wyglądał mniej więcej tak, jak na poniższym rysunku:



To ćwiczenie jest ukończone; tutaj zobaczyłeś, jak przeprowadzić atak DoS na maszynę za pomocą ataku SYN Flood. Poznaj inne luki w zabezpieczeniach sieciowych dostępne w HPING3. Dość dobrą ściągawkę dotyczącą korzystania z HPING3 można znaleźć tutaj: https://cheatography.com/myke670/cheatsheets/hping3/.

To kończy ćwiczenie 1.

Ćwiczenie 2

W tym ćwiczeniu przeanalizujemy plik przechwytywania pakietów (PCAP). Możesz użyć swojej instancji Kali Linux lub dowolnego komputera z zainstalowanym Wireshark. Pobierz plik sample.pcap z repozytorium GitHub dla tego rozdziału pod adresem https://github.com/PacktPublishing/Hands-On-Ethical-Hacking-Tactics. Załaduj plik PCAP do Wiresharka i odpowiedz na następujące pytania:

1. Ile pakietów zawiera przechwytywanie?

2. Znajdź i zapisz wszystkie żądania domen DNS i wszystkie zwrócone adresy IP.

3. Zapisz pełne żądania URI dla wszystkich żądań GET.

4. Jakiego rodzaju pakietem jest pakiet o numerze 1400?

5. Jaki jest referencyjny znacznik czasu pakietu o numerze 1404?

6. Wyeksportuj plik app_logo.png i znajdź skrót MD5 pliku.

7. Wyeksportuj plik SD-Booster-v 1-eng.html, a następnie zapisz w nim adresy e-mail i kontakty z Twittera.

To laboratorium jest ukończone; przeanalizowałeś plik PCAP i odpowiedziałeś na pytania dotyczące aktywności komputera z perspektywy sieci TCP/IP.

Podsumowanie

W tym rozdziale omówiono podstawowe elementy protokołów. Celem było wyjaśnienie ich mechanizmów i funkcjonalności, a także sposobu działania. Zrozumienie działania protokołu może również pomóc w lepszym zrozumieniu zagrożeń i ataków na wszystkich poziomach modelu OSI. Niezależnie od pełnionych ról, specjaliści ds. bezpieczeństwa muszą zrozumieć nie tylko podstawy protokołu TCP/IP, ale także to, w jaki sposób ataki mogą je wykorzystać i jak to wygląda. W następnym rozdziale przyjrzymy się bliżej niektórym z tych typów ataków, zagłębiając się w temat złośliwego oprogramowania.

Ocena

1. Wszystkie poniższe protokoły komunikacyjne to protokoły komunikacyjne, Z WYJĄTKIEM czego?

A. ICMP

B. TCP

C. SMS

D. UDP

2. IPv4 używa _____ schematu adresowania bitów

A. 16-

B. 32-

C. 64-

D. 128-

3. Co oznacza skrót OSI?

A. Infrastruktura systemu operacyjnego

B. Incydent systemu otwartego

C. Internet standardu otwartego

D. Połączenia systemów otwartych

4. Co to jest polecenie używane do wyświetlania aktywnych połączeń?

A. NetBeui

B. NetBIOS

C. Netstat

D. NetOP

5. Przez co TCP inicjuje połączenia?

A. Czas życia (Time-to-live)

B. Routeadd

C. Wygaszenie źródła (Source quench)

D. Trójetapowe uzgadnianie

6. Jaki protokół jest używany do przekazywania warunków błędu? A. UDP

B. SMTP

C. ICMP

D. SNMP

7. Który z poniższych NIE należy do dobrze znanych portów zdefiniowanych przez IANA?

A. 80

B. 123

C. 443

D. 8080

8. Jaki jest jeden z ataków wykorzystujących protokoły UDP?

A. Atak typu SYN flood

B. Atak typu DNS Amplification

C. Atak typu source-quench

D. Ping-of-Death

9. Jaka jest teoretyczna maksymalna liczba dostępnych portów?

A. 25 252

B. 65 535

C. 128 000

D. 1 684 256

10. Prawda czy fałsz – protokół IPv6 jest obecnie powszechnie wdrażany w sieciach? A. Prawda

B. Fałsz

Odpowiedzi

1. C

2. B

3. D

4. C

5. D

6. C

7. D

8. B

9. B

10. B