Wireshark

Network Sniffer (ang. program "węszący" sieć) jest to program lub sprzęt komputerowy służący do przechwytywania i zapisywania ruchu sieciowego. Pozwala szczegółowo zapoznać się z zawartością przesyłanych pakietów poprzez ich dekodowanie. Wykorzystywany jest głównie do diagnostyki niezawodności i wydajności sieci. Jednym z najpopularniejszych rozwiązań tego typu jest program Wireshark, rozwijany od 1998 roku na zasadach licencji GNU GPL.

IP

Protokół komunikacyjny przeznaczony dla Internetu (ang. Internet Protocol), jest protokołem wartstwy sieci modelu OSI, który stanowi podstawę struktury komunikacyjnej Internetu. Obecnie ciągle stosowany jest protokół w wersji 4 (IPv4), natomiast sukcesywnie wypierany jest przez swojego następcę, wersję 6 (IPv6).
Długość nagłówka IPv4 wynosi od 20 do 60 bajtów.

<table>
<tr><th>0-3</th><th>4-7</th><th>8-13</th><th>14-15</th><th>16-18</th><th>19-31</th></tr>
<tr><td>Wersja</td><td>Długość nagłówka</td><td>Usługi zróżnicowane</td><td>ECN</td><td colspan="2">Całkowita długość</td></tr>
<tr><td colspan="4">Numer identyfikacyjny</td><td>Flagi</td><td>Przesunięcie</td></tr>
<tr><td colspan="2">Czas życia</td><td colspan="2">Protokół warstwy wyższej</td><td colspan="2">Suma kontrolna nagłówka</td></tr>
<tr><td colspan="6">Adres źródłowy IP</td></tr>
<tr><td colspan="6">Adres docelowy IP</td></tr>
<tr><td colspan="5">Opcje IP</td><td>Wypełnienie</td></tr>
<tr><td colspan="6">Dane</td></tr>
</table>

Budowa nagłówka IPv4

- **Wersja** – pole opisujące wersję protokołu.
- **Długość nagłówka** – długość nagłówka IP wyrażona w 32-bitowych słowach; minimalna długość nagłówka to 5.
- **Usługi zróżnicowane** – Pierwsze trzy bity pola Usługi zróżnicowane informują o priorytecie (111 to najwyższy, a 000 - zwyczajny priorytet). Kolejne trzy bity, oznaczają ważność poszczególnych parametrów: D - małe opóźnienie (ang. delay), T - duża przepustowość (ang. throughput) i R - wysoka niezawodność (ang. reliability).
- **ECN** – jeśli ustawiony na wartość 1, informuje o przeciążeniu bufora
- **Całkowita długość pakietu** – długość całego datagramu IP (nagłówek oraz dane); minimalna długość to 576 bajtów, natomiast maksymalna to 65535 bajty.
- **Numer identyfikacyjny** – numer identyfikacyjny, wykorzystywany podczas fragmentacji do określenia przynależności pofragmentowanych datagramów.
- **Flagi** – flagi wykorzystywane podczas fragmentacji datagramów.
- **Przesunięcie** – w przypadku fragmentu większego datagramu pole to określa miejsce danych w oryginalnym datagramie;
- **Czas życia** – czas życia datagramu. Zgodnie ze standardem liczba przeskoków przez jaką datagram znajduje się w obiegu.
- **Protokół warstwy wyższej** – informacja o protokole warstwy wyższej, który jest przenoszony w polu danych datagramu IP.
- **Suma kontrolna nagłówka** – suma kontrolna nagłówka pakietu, pozwalająca stwierdzić czy został on poprawnie przesłany, sprawdzana i aktualizowana przy każdym przetwarzaniu nagłówka.
- **Adres źródłowy i adres docelowy** – pola adresów nadawcy i odbiorcy datagramu IP.
- **Opcje** – niewymagane pole opcji, opisujące dodatkowe zachowanie pakietów IP
- **Wypełnienie** – opcjonalne pole wypełniające nagłówek do wielkości będącej wielokrotnością 32.

TCP

Protokół kontroli transmisji (ang. Transmission Control Protocol), jest to połączeniowy i niezawodny protokół komunikacyjny warstwy transportowej modelu OSI. Stanowi część powszechnie stosowanego stosu TCP/IP.

Nagłówek TCP składa się conajmniej z pięciu 32 bitowych słów, co łącznie daje 160 bitów. Dodatkowo zawierać może pole Opcje o zmiennej długości będącej wielokrotnością 8 bitów.

<table>
<tr><th>0-3</th><th>4-9</th><th>10-15</th><th>16-31</th></tr>
<tr><td colspan="3">Port źródłowy</td><td>Port docelowy</td></tr>
<tr><td colspan="4">Numer sekwencji</td></tr>
<tr><td colspan="4">Numer potwierdzenia (jeśli flaga ACK jest ustawiona)</td></tr>
<tr><td>Długość nagłówka</td><td>Zarezerwowane</td><td>Flagi</td><td>Szerokość okna</td></tr>
<tr><td colspan="3">Suma kontrolna</td><td>Wskaźnik priorytetu (jeśli URG jest ustawiona)</td></tr>
<tr><td colspan="4">Opcje</td></tr>
</table>

Budowa nagłówka TCP

Najważniejsze cechy protokołu:

- działa w trybie klient-serwer
- wykorzystuje procedury do nawiązania i zakończenia połączenia
- połączenie sterowane jest przy pomocy flag
- gwarantuje dostarczenie wszystkich pakietów z zachowaniem kolejności, bez duplikatów

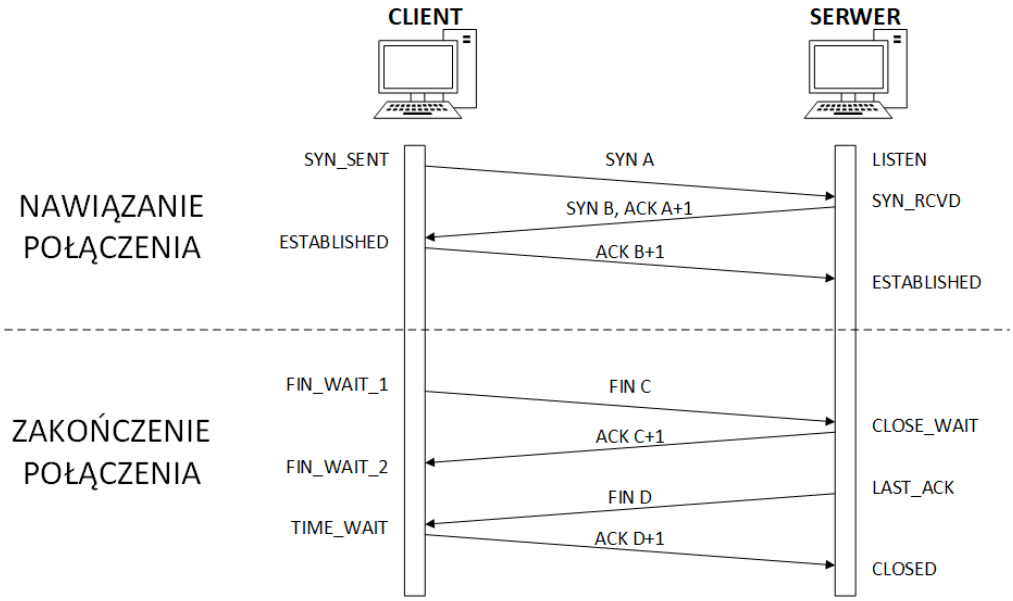
Flagi:

- NS – (ang. Nonce Sum) jednobitowa suma wartości flag ECN (ECN Echo, Congestion Window Reduced, Nonce Sum) weryfikująca ich integralność
- CWR – (ang. Congestion Window Reduced) flaga potwierdzająca odebranie powiadomienia przez nadawcę, umożliwia odbiorcy zaprzestanie wysyłania echa.
- ECE – (ang. ECN-Echo) flaga ustawiana przez odbiorcę w momencie otrzymania pakietu z ustawioną flagą CE
- URG – informuje o istotności pola "Priorytet"
- ACK – informuje o istotności pola "Numer potwierdzenia"
- PSH – wymusza przesłanie pakietu
- RST – resetuje połączenie (wymagane ponowne uzgodnienie sekwencji)
- SYN – synchronizuje kolejne numery sekwencyjne
- FIN – oznacza zakończenie przekazu danych

Mechanizm nawiązania połączenia

Jedną z najważniejszych cech protokołu sterowania transmisją jest obecność mechanizmów nawiązania i zakończenia połączenia. Nawiązanie połączenia jest opartę o procedurę zwaną *three-way handshake*. Ustanowienia połączenia wygląda następująco:

1. Klient wysyła segment SYN wraz z inicjującym numerem sekwencji np. liczbą 100 (symbol A)
2. Serwer odpowiada wysyłając segment SYN ze swoim numerem sekwencji (symbol B), a także potwierdza otrzymanie segmentu od klienta wysyłając ACK z numerem A+1.
3. Klient wysyła potwierdzenie ACK z numerem B+1 odebrania segmentu SYN od serwera.



Procedury nawiązania i zakończenia połączenia

UDP

Protokół pakietów użytkownika (ang. User Datagram Protocol) jest bezpołączeniowym protokołem komunikacyjnym warstwy transportowej modelu OSI. W przeciwieństwie do protokołu TCP nie gwarantuje dostarczenia wszystkich pakietów, ani zachowania kolejności. W zamian za to oferuje szybszą transmisję oraz mniejszy narzut danych. Nagłówek UDP składa się z 4 pól po 16 bitów.

0-15	16-31
Port źródłowy	Port docelowy
Długość datagramu	Suma kontrolna

Budowa nagłówka UDP

ARP

Protokół ARP (ang. Address Resolution Protocol) umożliwia przekształcanie adresów warstwy sieciowej (warstwa 3. modelu OSI) na adresy warstwy łącza danych (warstwa 2. modelu OSI). We współczesnych sieciach Ethernet (IEEE 802.3) sprowadza się to najczęściej do translacji adresu IPv4 na adres fizyczny MAC.

0-7	8-15	16-31
Typ warstwy fizycznej		Typ protokołu wyższej warsty
Długość adresu sprzętowego	Długość protokołu wyższej warstwy	Operacja
Adres sprzętowy źródła		
Adres protokołu wyższej warstwy źródła		
Adres sprzętowy przeznaczenia		
Adres protokołu wyższej warstwy przeznaczenia		

Budowa pakietu ARP

- Typ warstwy fizycznej (HTYPE) – 16 bitów opisujących typ protokołu warstwy fizycznej, przykład: 1 – Ethernet

- Typ protokołu wyższej warstwy (PTYPE) – 16 bitów opisujących typ protokołu warstwy wyższej, przykłady: IPv4 – 0x0800; ARP – 0x0806; IPv6 – 0x86DD
- Długość adresu sprzętowego (HLEN) – 8 bitow opisujących długość adresu sprzętowego w bajtach
- Długość protokołu wyższerj warstwy (PLEN) – 8 bitów opisujących długość adresu warstwy wyższej podana w bajtach
- Operacja (OPER) – 8 bitów opisujących kod operacji ARP, przykładowe kody: 1 – zapytanie; 2 – odpowiedź; 3 – zapytanie odwrotne; 4 – odpowiedź odwrotna
- Adres sprzętowy źródła (SHA) – 32 bity przedstawiające adres sprzętowy nadawcy
- Adres protokołu wyższej warstwy (SPA) – 32 bity przedstawiające adres nadawcy protokołu warstwy wyższej
- Adres sprzętowy źródła (THA) – 32 bity przedstawiające adres sprzętowy odbiorcy
- Adres protokołu wyższej warstwy (TPA) – 32 bity przedstawiające adres odbiorcy protokołu warstwy wyższej

Protokół ARP działa w następujący sposób:

- Host A chce przesłać pakiet do hosta B o adresie IP_B, jednak nie zna jego adresu MAC
- Host A rozgłasza do całej podsieci pakiet z pytaniem o adres MAC urządzenia o adresie IP_B
- Pytanie otrzymują wszystkie urządzenia, natomiast odpowiada tylko host B, który rozpoznał swój adres
- Host B przesyła swój adres MAC bezpośrednio do hosta A

ICMP

Internetowy protokół komunikatów kontrolnych (ang. Internet Control Message Protocol) jest protokołem warstwy sieciowej modelu OSI wykorzystywanym w diagnostyce sieci oraz trasowaniu. Umożliwia on przesyłanie między urządzeniami sieciowymi informacji o błędach w funkcjonowaniu sieci IP. Protokół ICMP jest wykorzystywany przez takie programy jak ping, czy traceroute.

0-7	8-15	16-23	24-31
Typ	Kod	Suma kontrolna	
Dane (opcjonalnie)			

Budowa pakietu ICMP

Wybrane typy wiadomości:
0 – Echo Reply (odpowiedź na ping)
3 – Destintation Unreachable
8 – Echo Request (ping)
9 – Router Advertisement
11 – Time Exceeded
17 – Address Mask Request (żądanie maski adresowej)
18 – Address Mask Reply (zwrot maski adresowej)
30 – Traceroute

0-7	8-15	16-23	24-31
Typ	Kod	Suma kontrolna	
Identyfikator		Numer sekwencji	
Dane (opcjonalnie)			

Budowa pakietu ICMP Echo Request i Echo Reply

W przypadku pakietów ICMP Echo Request i Echo Reply w sekcji Dane dodatkowo pojawiają się dodatkowe wartość: identyfikator (16 bitów) i numer sekwencji (16 bitów). Służą one do oznaczania żądań w przypadku, gdy nadawca wysyła kilka pakietów Echo Request.

DNS

System nazw domenowych (ang. Domain Name System) wykorzystuje do wymiany danych z systemem serwerów dedykowany protokół warstwy aplikacji. Jest on transportowany przeważnie w pakietach UDP.

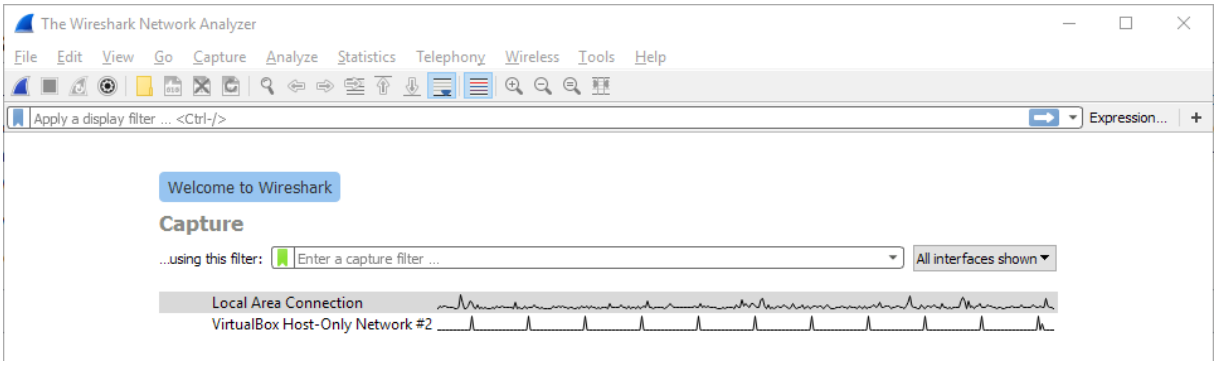
<table>
<tr><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td></tr>
<tr><td colspan="16">ID</td></tr>
<tr><td>QR</td><td colspan="4">OPCODE</td><td>AA</td><td>TC</td><td>RD</td><td>RA</td><td colspan="3">Z</td><td colspan="4">RCODE</td></tr>
<tr><td colspan="16">QDCOUNT</td></tr>
<tr><td colspan="16">ANCOUNT</td></tr>
<tr><td colspan="16">NSCOUNT</td></tr>
<tr><td colspan="16">ARCOUNT</td></tr>
</table>

Format nagłówka wiadomości DNS

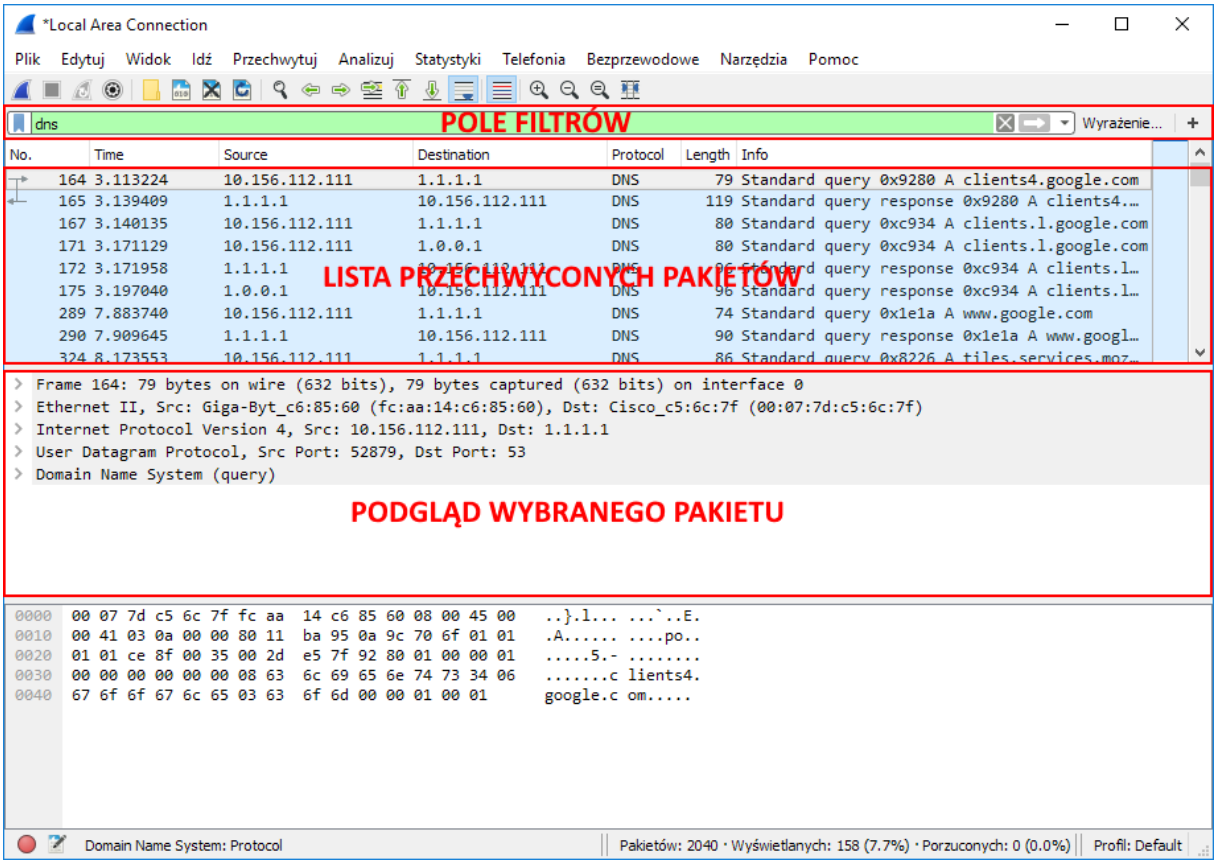
- **ID** –identyfikator tworzony przez program wysyłający zapytanie; serwer przepisuje ten identyfikator do swojej odpowiedzi, dzięki czemu możliwe jest jednoznaczne powiązanie zapytania i odpowiedzi
- **QR** – określa, czy komunikat jest zapytaniem (0) czy odpowiedzią (1)
- **OPCODE** – określa rodzaj zapytania wysyłanego od klienta, jest przypisywany przez serwer do odpowiedzi. Wartości: 0 – QUERY (standardowe zapytanie); 1 – IQUERY (zapytanie zwrotne);
  2 – STATUS (pytanie o stan serwera).
- **AA** – oznacza, że odpowiedź jest autorytatywna.
- **TC** – oznacza, że odpowiedź nie zmieściła się w jednym pakiecie UDP i została obcięta.
- **RD** – oznacza, że klient żąda rekurencji – pole to jest kopiowane do odpowiedzi
- **RA** – bit oznaczający, że serwer obsługuje zapytania rekurencyjne
- **Z** – zarezerwowane do przyszłego wykorzystania.
- **RCODE** – kod odpowiedzi. Przyjmuje wartości:
  - 0 – brak błędu,
  - 1 – błąd formatu – serwer nie potrafił zinterpretować zapytania,
  - 2 – błąd serwera – wewnętrzny błąd serwera,
  - 3 – błąd nazwy – nazwa domenowa podana w zapytaniu nie istnieje,
  - 4 – nie zaimplementowano – serwer nie obsługuje typu otrzymanego zapytania,
  - 5 – odrzucono – serwer odmawia wykonania określonej operacji, np. transferu strefy,
- **QDCOUNT** – określa liczbę wpisów w sekcji zapytania
- **ANCOUNT** – określa liczbę rekordów zasobów w sekcji odpowiedzi
- **NSCOUNT** – określa liczbę rekordów serwera w sekcji zwierzchności
- **ARCOUNT** – określa liczbę rekordów zasobów w sekcji dodatkowej

Wireshark

Wireshark jest to sniffer sieci służący do monitorowania ruchu sieciowego. Umożliwia przechwytywanie pakietów docierających do karty sieciowej. Obsługuje wiele różnych protokołów sieciowych. Program jest rozpowszechniany na zasadzie Open Source.



Okno powitalne aplikacji Wireshark



Widok główny programu Wireshark

Filtry

Ze względu na dużą ilość przechwytywanych pakietów przydatnym narzędziem mogą być filtry. W programie Wireshark istnieją dwa rodzaje filtrów: Capture Filters oraz Display Filters.

CaptureFilter służy do definiowania jakie pakiety będą przechwytywane przez program.

Natomiast Display Filters służą do filtrowania przechwyconych pakietów.

Przykładowe filtry:

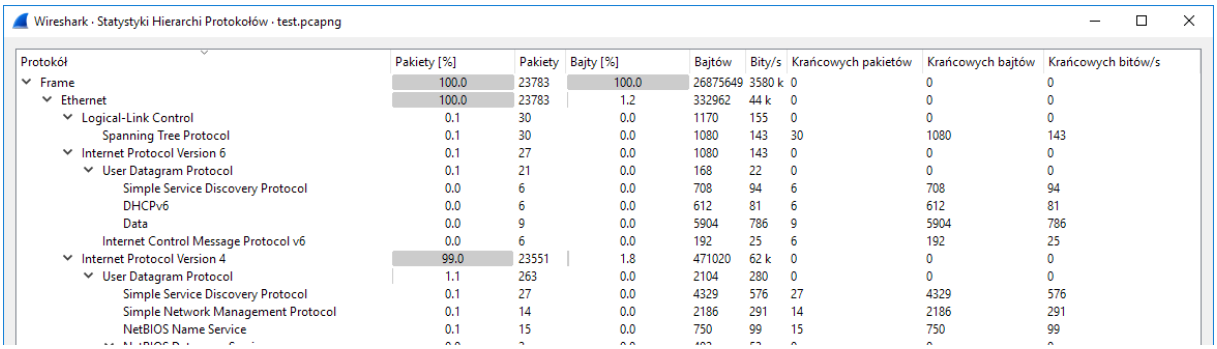
Display Filter	Wyświetlane pakiety
`dns`	pakiety zawierające protokół DNS
`dns.qry.name == www.example.com`	pakiety DNS zawierające zapytanie o domenę www.example.com
`ip.dst == 1.2.3.4 and tcp`	pakiety z adresem odbiorcy 1.2.3.4 zawierających protokół TCP
`ip.dst == 1.2.3.4 and http.request.method == GET`	pakiety z adresem odbiorcy 1.2.3.4 zawierających żądanie HTTP GET
`ip.dst = 1.2.3.4 and icmp.type == 8`	pakiety zawierające wiadomość ICMP ECHO Request z adresem odbiorcy 1.2.3.4
`ip.src == 1.2.3.4 and icmp.type == 0`	pakiety zawierające wiadomość ICMP Echo Reply z adresem nadawcy 1.2.3.4
`ftp or ftp-data`	pakiety związanych z transmisją opartą o protokół FTP (**`ftp`** to połączenie kontrolne, **`ftp-data`** to transmisja danych)
`ftp.request.command == USER`	pakiety protokołu FTP z komendą wysyłającą nazwę użytkownika
`ftp.request.command == PASS`	pakiety protokołu FTP z komendą wysyłającą hasło użytkownika
`ftp.request.command == PWD`	pakiety protokołu FTP z komendą zwracającą aktualny katalog po stronie serwera
`ftp.request.command == MLSD`	pakiety protokołu FTP z komendą zwracającą zawartość aktualnego katalogu po stronie serwera

Więcej:
https://wiki.wireshark.org/DisplayFilters
https://wiki.wireshark.org/CaptureFilters#Capture_filter_is_not_a_display_filter

Statystyki
Program Wireshark umożliwia przeprowadzenie analizy statystycznej przechwyconego ruchu. Aby wyświetlić raport należy wybrać rodzaj analizy z menu Statystyki.

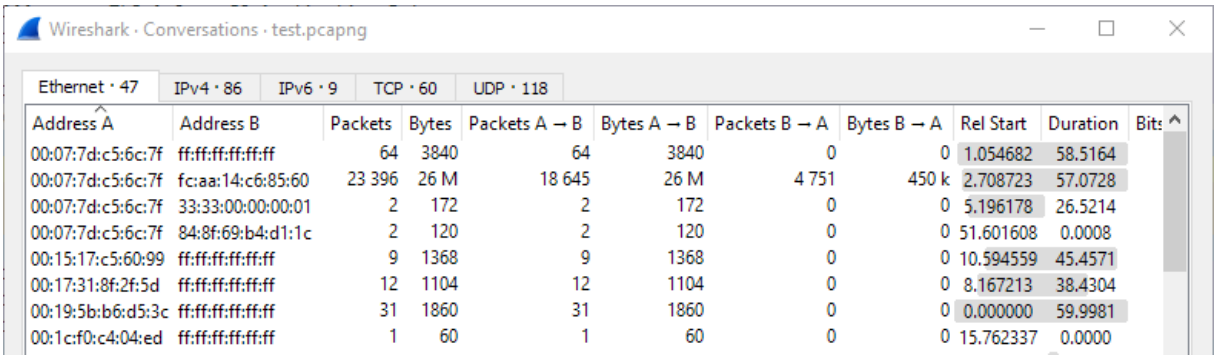
Przykładowe analizy to:

Statystyki -> Hierarchia Protokołów: przedstawia procentowy udział protokołów biorących udział w przechwyconym ruchu sieciowym



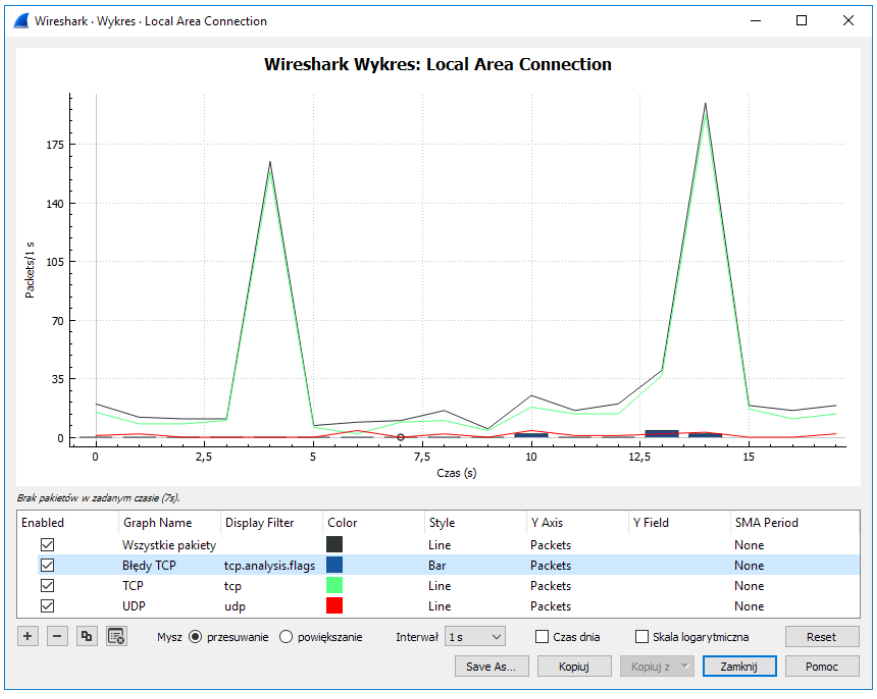
Protokół	Pakiety [%]	Pakiety	Bajty [%]	Bajtów	Bity/s	Krańcowych pakietów	Krańcowych bajtów	Krańcowych bitów/s
Frame	100.0	23783	100.0	26875649	3580 k	0	0	0
Ethernet	100.0	23783	1.2	332962	44 k	0	0	0
Logical-Link Control	0.1	30	0.0	1170	155	0	0	0
Spanning Tree Protocol	0.1	30	0.0	1080	143	30	1080	143
Internet Protocol Version 6	0.1	27	0.0	1080	143	0	0	0
User Datagram Protocol	0.1	21	0.0	168	22	0	0	0
Simple Service Discovery Protocol	0.0	6	0.0	708	94	6	708	94
DHCPv6	0.0	6	0.0	612	81	6	612	81
Data	0.0	9	0.0	5904	786	9	5904	786
Internet Control Message Protocol v6	0.0	6	0.0	192	25	6	192	25
Internet Protocol Version 4	99.0	23551	1.8	471020	62 k	0	0	0
User Datagram Protocol	1.1	263	0.0	2104	280	0	0	0
Simple Service Discovery Protocol	0.1	27	0.0	4329	576	27	4329	576
Simple Network Management Protocol	0.1	14	0.0	2186	291	14	2186	291
NetBIOS Name Service	0.1	15	0.0	750	99	15	750	99

Statystyki -> Konwersacje: przedstawia ilość danych/pakietów wymienionych między poszczególnymi hostami. Dane są posortowane według protokołów.



Address A	Address B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration	Bits
00:07:7d:c5:6c:7f	ff:ff:ff:ff:ff:ff	64	3840	64	3840	0	0	1.054682	58.5164	
00:07:7d:c5:6c:7f	fc:aa:14:c6:85:60	23 396	26 M	18 645	26 M	4 751	450 k	2.708723	57.0728	
00:07:7d:c5:6c:7f	33:33:00:00:00:01	2	172	2	172	0	0	5.196178	26.5214	
00:07:7d:c5:6c:7f	84:8f:69:b4:d1:1c	2	120	2	120	0	0	51.601608	0.0008	
00:15:17:c5:60:99	ff:ff:ff:ff:ff:ff	9	1368	9	1368	0	0	10.594559	45.4571	
00:17:31:8f:2f:5d	ff:ff:ff:ff:ff:ff	12	1104	12	1104	0	0	8.167213	38.4304	
00:19:5b:b6:d5:3c	ff:ff:ff:ff:ff:ff	31	1860	31	1860	0	0	0.000000	59.9981	
00:1c:f0:c4:04:ed	ff:ff:ff:ff:ff:ff	1	60	1	60	0	0	15.762337	0.0000	

Statystyki -> Wykres: pozwala wizualnie przedstawić częstotliwość transmisji pakietów. Pozwala tworzyć serie danych wykorzystując pole Display Filters



Literatura:

Hunt, Craig; TCP/IP : administracja sieci. Warszawa : Oficyna Wydaw. READ ME, 1996.
Blank, Andrew G, Podstawy TCP/IP / Andrew G. Blank ; przekł. z jęz. ang. Grzegorz Kowalski, Warszawa : Mikom, 2005.
Chappell, Laura, Wireshark Network Analysis, The Officiajl Wireshark Cerified Network Analyst Study Guide, Second Edition, 2012

Scenariusz nr 1

Sprzęt:
Komputer PC (System operacyjny Windows 7)

Oprogramowanie:
Wireshark

Wykonanie ćwiczenia:

1. Uruchomić program Wireshark
2. Pole capture filter zostawić puste
3. Wybrać interfejs wykorzystywany do połączenia z siecią poprzez drukrotne kliknięcie
4. Wykonać następujące czynności:
    a. uruchomić przeglądarkę i wejść na stronę www: http://.....
    b. uruchomić linię poleceń (cmd.exe) i wykonać ping do adresu: .....
    c. wykonać połączenie z serwerem ftp: ftp://.....
5. Po wykonaniu wybranych połączeń należy zakończyć przechwytywanie pakietów
6. Wykorzystując stworzony zapis ruchu sieciowego należy wykonać następujące operacje:
    a. wykonać zrzut ekranu przedstawiający żądanie i odpowiedź DNS dla domen ustalonych w punktach 4a, 4b i 4c, przykładowe zrzuty ekranu przedstawiono na ostatniej stronie
    b. na podstawie odpowiedzi z serwera DNS określić adresy IP powiązane z domenami ustalonymi w punktach 4a i 4b
    c. wykonać zrzut ekranu przedstawiający pakiety odpowiedzialne za nawiązanie połączenia TCP (tzw. Three-way handshake) z domeną ustaloną w punkcie 4a
    d. dla połączenia z punktu 4a wykonać zrzut ekranu przedstawiający żądanie HTTP GET oraz odpowiedź na to żądanie
    e. wykonać zrzut ekranu pakietów ICMP Echo Request i Echo Reply powiązanych z wykonanym poleceniem ping do adresu z punktu 4b
    f. wykonać zrzuty ekranu pakietów zawierających początkową fazę komunikacji z serwerem ftp: wysłanie loginu (+odpowiedź), wysłanie hasła (+odpowiedź), żądanie nazwy aktualnego katalogu po stronie serwera (+odpowiedź), żądanie o zawartości aktualnego katalogu po stronie serwera (+odpowiedź)

UWAGA! W przypadku braku pakietów DNS związanych z wykonanymi połączeniami należy uruchomić linię poleceń, następnie wpisać `ipconfig /flushdns`, a następnie powtórzyć przechwytywanie pakietów.

Wyniki pomiarów:

a) Przedstawić zrzuty ekranów dla podpunktów 6a-f
b) Przedstawić w formie tabelarycznej:
    - zawartość nagłówków IP, UDP, DNS wybranych żądanie do serwera DNS (punkt 61).
    - zawartość nagłówków IP, UDP, DNS odpowiedzi DNS z wyszczególnieniem poszczególnych protokołów (IP, UDP, DNS).
    - zawartość nagłówków IP i TCP pakietów odpowiedzialnych za nawiązanie połączenia TCP (tzw. Three-way handshake) (punkt 6c)
    - zawartość nagłówków IP, TCP i HTTP przedstawiających żądanie HTTP GET oraz odpowiedź (punkt 6d)
    - zawartość nagłówków ICMP Echo Request i Echo Reply (punkt 6e)
c) Na podstawie pakietów DNS napisać jakie adresy IP są przypisane do badanych domen
d) Opisać co zawierała odpowiedź na żądanie HTTP GET
e) Na podstawie analizy pakietów FTP ocenić bezpieczeństwo korzystania z tego protokołu
f) Przedstawić wnioski z wykonanego ćwiczenia

Scenariusz nr 2

Sprzęt:
Komputer PC (System operacyjny Windows 7)

Oprogramowanie:
Wireshark

Wykonanie ćwiczenia:

1. Uruchomić program Wireshark
2. Wybrać interfejs wykorzystywany do połączenia z siecią
3. Rozpocząć przechwytywanie pakietow
4. Zakończyć przechwytywanie pakietów, kiedy ich liczba przekroczy 10000.
5. Wybrać 10 różnych pakietów, a następnie wykonać ich zrzuty ekranu (patrz: przykład na ostatniej stronie)

Wyniki pomiarów:

a) Wybrane pakiety należy przedstawić w formie tabelarycznej przedstawiając nagłówki poszczególnych protokołów od warstwy 3 wzwyż.
b) Opisać do czego są wykorzystywane protokoły przesłane w w wybranych pakietach
c) Na podstawie przechwyconego ruchu sieciowego przedstawić następujące statystyki ruchu w sieci lokalnej:
   - zaprezentować procentowy udział protokołów opartych o IPv4 w przechwyconym ruchu
   - przedstawić 5 konwersacji TCP, podczas których przesłano największą ilość danych (przedstawić liczbę danych w bajtach)
   - przedstawić 5 konwersacji UDP, podczas których przesłano największą ilość danych (przedstawić liczbę danych w bajtach)
   - stworzyć wykres częstotliwości transmitowanych pakietów w czasie dla wybranych w punkcie a. 10 protokołów
d) Podsumować przeprowadzone ćwiczenie

PRZYKŁADY

Przykładowa prezentacja pakietu ICMP Echo Request:

IP (warstwa 3)					
0-3	**4-7**	**8-13**	**14-15**	**16-18**	**19-31**
Wersja: 4	Dł.nag: 20	DSF: 0	ECN:0	Długość całk: 60	
Nr ident: 23671				Flagi: 0	Przesunięcie
TTL: 128		Prot: ICMP		Suma kontr. nagł: 0x9eb7	
Adres źródłowy: 10.156.112.111					
Adres docelowy: 172.217.23.174					
Opcje IP					Wypełnienie

ICMP (warstwa 3)			
0-7	**8-15**	**16-23**	**24-31**
Typ: 8	Kod: 0	Suma kontrolna: 0x4d46	
Identyfikator: 1		Numer sekwencji: 21	
dane:61:62:63:64:65:66:67:68:69:6a:6b:6c:6d:6e:6f:70:71:72:73:74:75:76:77:61:62:63:64:65:66:67:68:69			

Przykładowy zrzut ekranu przedstawiający pakiet ICMP Echo Request: