WYDANIE III

PRAKTYCZNA ANALIZA PAKIETÓW

WYKORZYSTANIE NARZĘDZIA WIRESHARK DO ROZWIĄZYWANIA PROBLEMÓW ZWIĄZANYCH Z SIECIĄ

CHRIS SANDERS

no starch press

Helion

Tytuł oryginału: Practical Packet Analysis: Using Wireshark to Solve Real-World Network Problems, 3rd Edition

Tłumaczenie: Robert Górczyński

ISBN: 978-83-283-3696-4

Wydawnictwo HELION
ul. Kościuszki 1c, 44-100 GLIWICE
tel. 32 231 22 19, 32 230 98 63
e-mail: *helion@helion.pl*
WWW: *http://helion.pl* (księgarnia internetowa, katalog książek)

Pliki z przykładami omawianymi w książce można znaleźć pod adresem:
ftp://ftp.helion.pl/przyklady/panpa3.zip

Drogi Czytelniku!
Jeżeli chcesz ocenić tę książkę, zajrzyj pod adres
http://helion.pl/user/opinie/panpa3
Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

Printed in Poland.

Spis treści

4

Praca z przechwyconymi pakietami

PO KRÓTKIM WPROWADZENIU DO NARZĘDZIA WIRESHARK PRZEDSTAWIONYM W ROZDZIALE 3. JESTEŚ GOTOWY DO ROZPOCZĘCIA PRZECHWYTYWANIA I ANALIZY PAKIETÓW. W TYM ROZDZIALE DOWIESZ się, jak pracować z plikami zawierającymi przechwycone dane, z przechwyconymi pakietami oraz z formatami wyświetlania czasu. Omówione zostaną także bardziej zaawansowane opcje dotyczące przechwytywania pakietów, a ponadto zagłębimy się w świat filtrów.

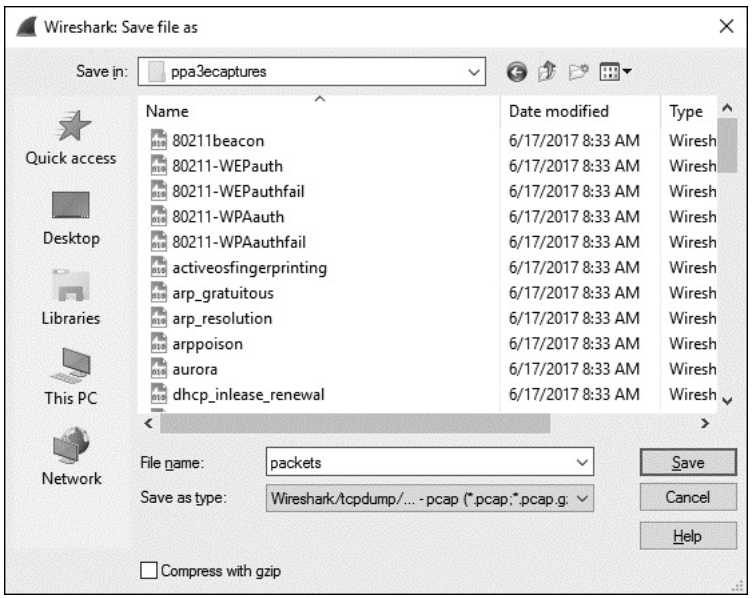
Praca z plikami zawierającymi przechwycone dane

Podczas przeprowadzania analizy pakietu przekonasz się, że znaczna jej część następuje już po przechwyceniu danych. Najczęściej w różnym czasie przeprowadzasz kilka operacji przechwytywania i zapisywania danych, a następnie analizujesz jednocześnie wszystkie zebrane w ten sposób dane. Narzędzie Wireshark

umożliwia zapisywanie plików z przechwyconymi danymi, które będziesz mógł później przeanalizować. Oczywiście masz możliwość łączenia ze sobą wielu takich plików.

Zapis i eksport plików zawierających przechwycone dane

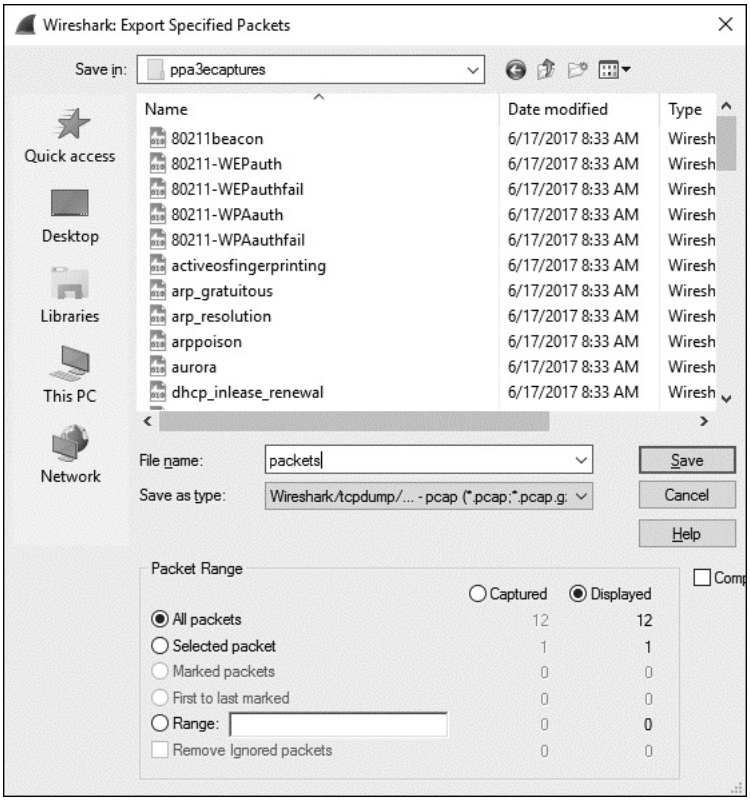
Aby zapisać przechwycony pakiet, wybierz opcję menu *File/Save As*. Na ekranie powinno wyświetlić się okno dialogowe *Save As* (zob. rysunek 4.1). W oknie tym możesz podać katalog, w którym zostanie zapisany plik, oraz określić format pliku. Jeżeli nie podasz formatu pliku, narzędzie Wireshark użyje domyślnego formatu pliku dla przechwyconych danych — *.pcap*.



Rysunek 4.1. Za pomocą okna dialogowego Save As możesz zapisywać przechwycone pakiety

Jedną z najmocniejszych stron okna dialogowego *Save As* jest możliwość zapisu określonego zakresu pakietu. W tym celu wybierz opcję *File/Export Specified Packets*. Na ekranie zostanie wyświetlone okno dialogowe, które pokazałem na rysunku 4.2. To doskonały sposób na zmniejszenie przerośniętych plików zawierających przechwycone dane pakietu. Możesz więc zapisywać pakiety pochodzące jedynie z podanego zakresu, pakiety oznaczone lub pakiety widoczne po zastosowaniu określonego filtru wyświetlania. (Pakiety oznaczone i filtry zostaną omówione w dalszej części rozdziału).

Dane przechwycone przez narzędzie Wireshark możesz eksportować do wielu różnych formatów w celu wyświetlania danych w innych mediach lub przeanalizowania zebranych danych w innych narzędziach. Dostępne formaty to między innymi: zwykły tekst, PostScript, CSV (ang. *comma separated values* — wartości



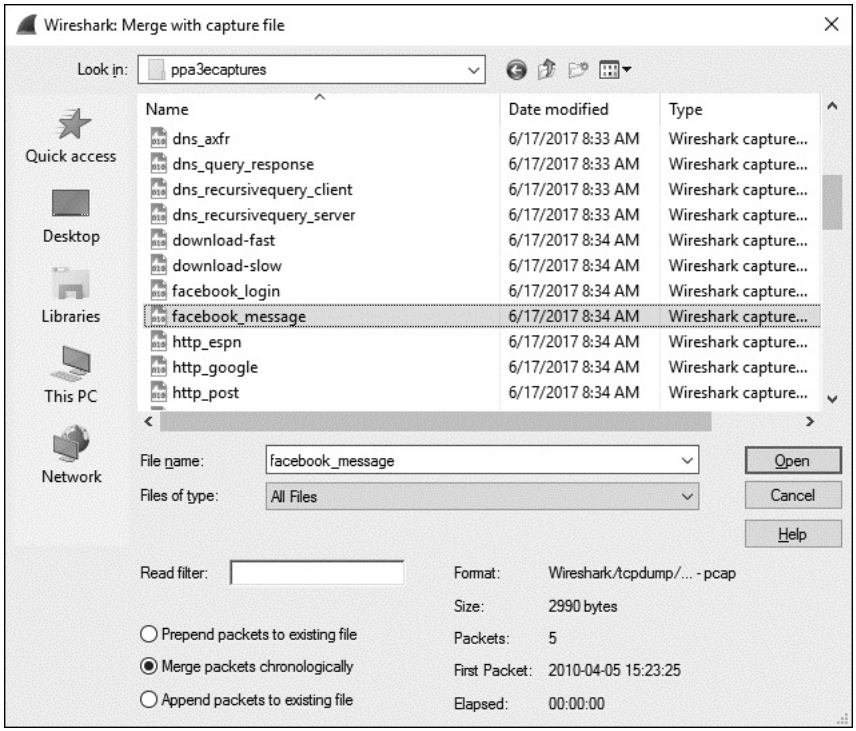
Rysunek 4.2. Okno dialogowe Export Specified Packets umożliwia większą kontrolę nad zapisywaniem przechwyconych pakietów

rozdzielone przecinkami) oraz XML. Aby wyeksportować przechwycone dane pakietu, wybierz opcję menu *File/Export Packet Dissections*, a następnie wskaż format pliku, w którym mają zostać zapisane dane. Na ekranie wyświetli się okno dialogowe *Save As* zawierające opcje związane z wybranym formatem.

Łączenie plików zawierających przechwycone dane

Pewne rodzaje analizy wymagają połączenia ze sobą wielu plików zawierających przechwycone dane. Jest to praktyka często stosowana podczas porównywania dwóch strumieni danych lub łączenia strumieni tego samego ruchu sieciowego, które zostały przechwycone oddzielnie.

Aby połączyć ze sobą pliki zawierające przechwycone dane, otwórz jeden z nich, a następnie wybierz opcję menu *File/Merge*. Na ekranie wyświetli się okno dialogowe zatytułowane *Merge with capture file* (zob. rysunek 4.3). W oknie tym wskaż plik, który ma zostać połączony z już otwartym plikiem, a następnie wybierz metodę połączenia plików. Plik wybrany w oknie dialogowym możesz umieścić przed już otworzonym plikiem, dołączyć go na końcu bądź połączyć oba pliki chronologicznie na podstawie ich znaczników czasu.



Rysunek 4.3. Okno dialogowe Merge with capture file pozwala łączyć dwa pliki zawierające przechwycone dane

Praca z pakietami

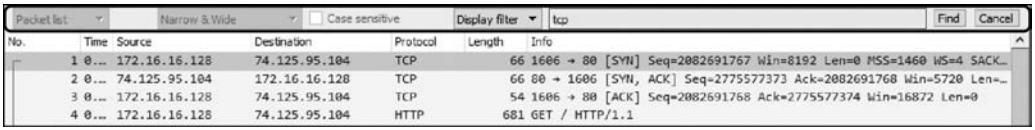
Może się zdarzyć sytuacja, w której wykorzystywana będzie ogromna liczba pakietów. W przypadku wzrostu liczby pakietów do rzędu tysięcy lub nawet milionów musisz mieć możliwość efektywnego poruszania się pomiędzy nimi. Narzędzie Wireshark umożliwia wyszukiwanie i oznaczanie pakietów spełniających określone kryteria. Pakiety można również wydrukować.

Wyszukiwanie pakietów

Aby wyszukać pakiety spełniające określone kryteria, musisz przejść do paska wyszukiwania *Find Packet* (zob. rysunek 4.4) poprzez naciśnięcie klawiszy *Ctrl+F*. Pasek wyszukiwania zostanie wyświetlony między panelami Filter i Packet List.

Pokazany na rysunku 4.4 pasek wyszukiwania zawiera trzy następujące opcje pozwalające na wyszukiwanie pakietów:

- **Display filter**. W tej opcji możesz podać filtr oparty na wyrażeniu. Dzięki temu filtrowi zostaną wyszukane jedynie pakiety spełniające zdefiniowane tutaj wyrażenie. Użyłem tej opcji na rysunku 4.4.



Rysunek 4.4. Wyszukiwanie pakietów w narzędziu Wireshark na podstawie określonych kryteriów — w pokazanym przypadku pakiety dopasowane do wyrażenia filtru tcp w opcji Display filter

- **Hex value**. Ta opcja powoduje wyszukanie pakietów zawierających podaną wartość szesnastkową.
- **String**. Ta opcja powoduje wyszukanie pakietów zawierających podany ciąg tekstowy. Istnieje możliwość wskazania panelu, w którym zostanie przeprowadzona operacja wyszukiwania, a także określenia, czy wielkość liter ma znaczenie.

Powyższe rodzaje wyszukiwania przedstawiono w tabeli 4.1.

Tabela 4.1. Dostępne rodzaje operacji wyszukiwania pakietów

Rodzaj wyszukiwania	Przykłady

filtr	`not ip`
	`ip.addr==192.168.0.1`
	`arp`

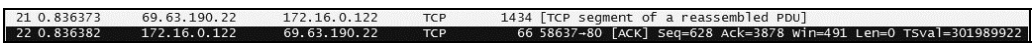
wartość szesnastkowa	`00ff`
	`ffff`
	`00abb1f0`

ciąg tekstowy	`StacjaRobocza1`
	`UżytkownikB`
	`domena`

Po wybraniu odpowiednich opcji w polu tekstowym należy wprowadzić kryteria wyszukiwania, a następnie nacisnąć przycisk *Find* w celu znalezienia pierwszego pakietu spełniającego zdefiniowane kryteria. Aby znaleźć kolejny pakiet dopasowany do kryteriów, trzeba ponownie nacisnąć przycisk *Find* lub klawisze *Ctrl+N*, natomiast przejście do poprzedniego znalezionego pakietu następuje po naciśnięciu klawiszy *Ctrl+B*.

Oznaczanie pakietów

Po znalezieniu pakietów spełniających zdefiniowane kryteria można je oznakować. Przykładowo: pakiety możesz oznakować, aby mieć możliwość ich oddzielnego zapisania lub szybkiego odszukania na podstawie koloru. Oznaczone pakiety wyróżniają się białym tekstem na czarnym tle, jak pokazano na rysunku 4.5.



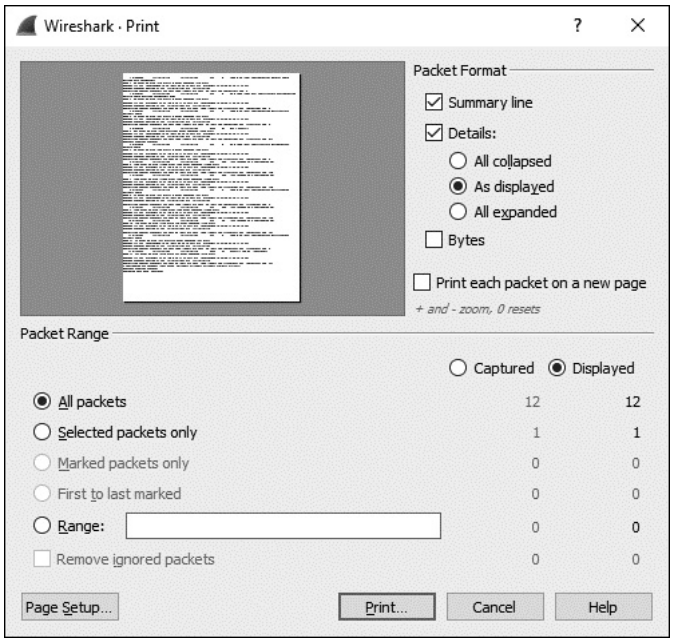
Rysunek 4.5. Oznaczony pakiet jest podświetlony na ekranie. Na rysunku widać, że pakiet pierwszy został oznaczony — ma czarne tło i biały tekst

W celu oznaczenia pakietu kliknij go prawym przyciskiem myszy w panelu *Packet List*, a następnie wybierz opcję *Mark Packet* z rozwijanego menu lub naciśnij klawisze *Ctrl+M*. Natomiast usunięcie zaznaczenia pakietu następuje po ponownym wybraniu wspomnianej opcji lub po ponownym naciśnięciu klawiszy *Ctrl+M*. W przechwyconych danych możesz oznaczyć dowolną liczbę pakietów. Do poruszania się do przodu i do tyłu po oznaczonych pakietach służą klawisze odpowiednio *Shift+Ctrl+N* i *Shift+Ctrl+B*.

Wydruk pakietów

Wprawdzie większość analizy pakietów przeprowadza się na ekranie komputera, ale zdarzają się sytuacje, gdy trzeba wydrukować przechwycone dane. Sam często drukuję pakiety i umieszczam je na biurku; w ten sposób mogę bardzo szybko sprawdzić ich zawartość podczas przeprowadzania innych analiz. Możliwość zapisu pakietów do pliku w formacie PDF również jest bardzo wygodna, zwłaszcza podczas przygotowywania raportów.

Aby wydrukować przechwycone pakiety, wyświetl okno dialogowe *Print* poprzez wybór opcji menu *File/Print*. Na ekranie pojawi się pokazane na rysunku 4.6 okno dialogowe *Print*.



Rysunek 4.6. Okno dialogowe Print umożliwia wydruk wskazanych pakietów

Podobnie jak w przypadku okna dialogowego *Export Specified Packets*, także tutaj można wybrać wydruk jedynie określonego zakresu pakietów, pakietów oznaczonych lub wyświetlanych na ekranie jako wynik działania filtru. Ponadto

masz możliwość wyboru panelu (jednego z trzech głównych paneli okna Wireshark), z którego będzie wydrukowany pakiet. Po zaznaczeniu wszystkich opcji naciśnij przycisk *Print*.

Konfiguracja formatu wyświetlania czasu i odniesień

Czas ma istotne znaczenie zwłaszcza podczas przeprowadzania analizy pakietu. Dla wszystkich zdarzeń zachodzących w sieci czas jest ważny, a Twoim zadaniem jest analiza trendów i opóźnień w sieci w niemal każdym pliku zawierającym przechwycone dane. Twórcy narzędzia Wireshark zdają sobie sprawę ze znaczenia czasu, dlatego dostarczyli wiele powiązanych z nim opcji konfiguracyjnych. W tym podrozdziale skoncentrujemy się na formacie wyświetlania czasu oraz na odniesieniach czasu do pakietu.

Format wyświetlania czasu

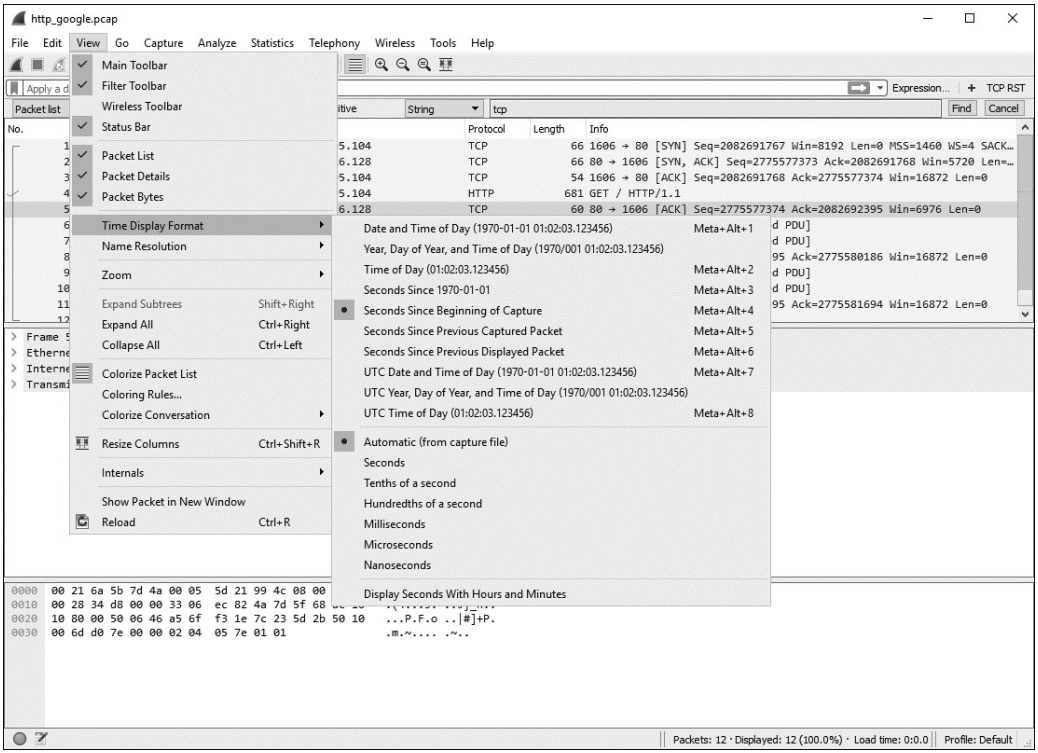
Każdy pakiet przechwycony przez narzędzie Wireshark ma znacznik czasu, który jest mu przypisany przez system operacyjny. Wireshark ma możliwość wyświetlenia zarówno bezwzględnego znacznika czasu, wskazującego dokładny moment przechwycenia danego pakietu, jak również czasu, który upłynął od ostatniego przechwyconego pakietu, a także od początku i końca operacji przechwytywania.

Opcje związane z wyświetlaniem czasu znajdują się w menu głównym zatytułowanym *View*. Pokazana na rysunku 4.7 grupa *Time Display Format* umożliwia wybór formatu wyświetlania czasu oraz dokładność czasu.

Wspomniane opcje pozwalają na wybór różnych ustawień związanych z wyświetlaniem czasu, między innymi daty i godziny, użycie czasu UTC, wyrażenie czasu w sekundach od początku epoki, w sekundach od chwili rozpoczęcia przechwytywania pakietów (to jest ustawienie domyślne), w sekundach od ostatnio przechwyconego pakietu itd.

Dokładność czasu pozwala na wybór automatycznego lub ręcznego ustawienia dokładności czasu. W przypadku trybu automatycznego format jest pobierany z pliku zawierającego przechwycone pakiety. Z kolei ręczne ustawienie pozwala na wybór sekundy, milisekundy, mikrosekundy. Obie opcje będziemy modyfikować w dalszej części książki, więc powinieneś się teraz z nimi zapoznać.

WSKAZÓWKA *Podczas porównywania danych pakietów pochodzących z różnych urządzeń upewnij się, że oba są synchronizowane z tym samym źródłem czasu. W szczególności ma to znaczenie podczas przeprowadzania analizy śledczej lub rozwiązywania problemów. Aby mieć pewność, że synchronizacja urządzeń jest prawidłowa, możesz wykorzystać protokół NTP (ang.* network time protocol*). Podczas analizy pakietów pochodzących z urządzeń znajdujących się w różnych strefach czasowych rozważ użycie czasu UTC zamiast czasu lokalnego, co pomoże uniknąć zamieszania w trakcie dokumentacji i prezentacji wyników przeprowadzonej analizy.*



Rysunek 4.7. Dostępnych jest kilka formatów wyświetlania czasu

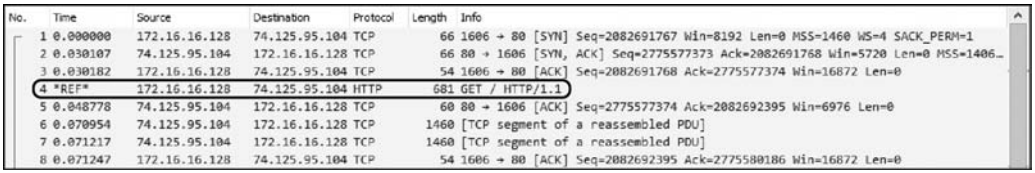
Odniesienie czasu do pakietu

Funkcja odniesienia czasu do pakietu pozwala skonfigurować określony pakiet w taki sposób, aby kolejne obliczenia dotyczące czasu były przeprowadzane względem danego pakietu. Ta funkcja jest wyjątkowo użyteczna podczas analizy wielu kolejnych zdarzeń, które są wywoływane gdzieś w środku pliku zawierającego przechwycone dane.

Aby ustawić odniesienie czasu do określonego pakietu, należy w pierwszej kolejności zaznaczyć pakiet w panelu *Packet List*, a następnie kliknąć go prawym przyciskiem myszy i wybrać opcję *Set/Unset Time Reference*. Usunięcie odniesienia czasu do pakietu następuje po zaznaczeniu pakietu w panelu Packet List i naciśnięciu klawiszy *Ctrl+T*.

Po włączeniu funkcji odniesienia czasu do określonego pakietu kolumna *Time* w panelu *Packet List* będzie zawierała ciąg tekstowy `*REF*` (zob. rysunek 4.8).

Włączenie opcji odniesienia czasu do danego pakietu jest użyteczne tylko wtedy, gdy format wyświetlania czasu przechwyconych danych jest zdefiniowany jako czas wyświetlany względem początku tych danych. Wszelkie inne ustawienia spowodują otrzymanie nieprzewidywalnych wyników oraz utworzenie bardzo mylących znaczników czasu.

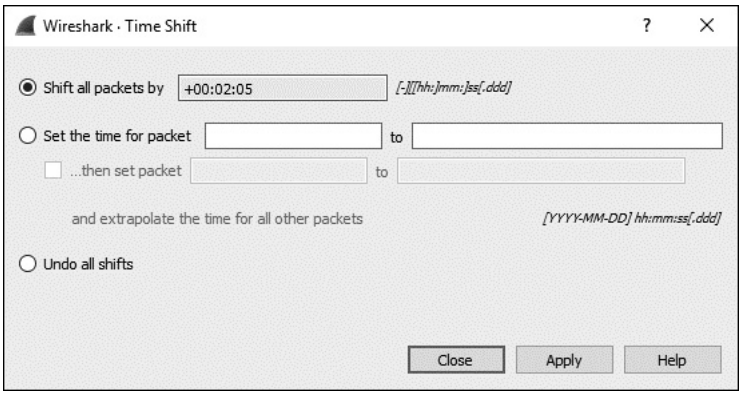


Rysunek 4.8. Pakiet wraz z włączonym odniesieniem czasu względem wskazanego pakietu

Przesunięcie czasu

W niektórych przypadkach możesz mieć pakiety pochodzące z wielu źródeł, które nie zostały zsynchronizowane z tym samym źródłem czasu. Tego rodzaju sytuacja najczęściej zdarza się podczas analizy plików zawierających przechwycone pakiety z dwóch lokalizacji współdzielących ten sam strumień danych. Wprawdzie większość administratorów dąży do zapewnienia synchronizacji wszystkich urządzeń znajdujących się w sieci, ale mimo wszystko często zdarza się kilkusekundowe przesunięcie czasu między określonymi typami urządzeń. Wireshark oferuje możliwość przesunięcia czasu pakietu, aby zniwelować tę różnicę podczas analizy.

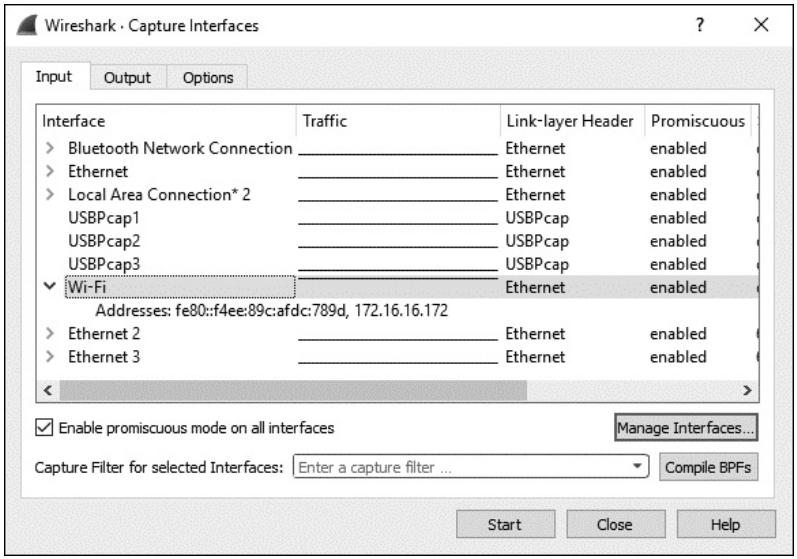
W celu przesunięcia czasu dla pakietu lub pakietów wybierz opcję *Edit/Time Shift* lub naciśnij klawisze *Ctrl+Shift+T*. W wyświetlonym oknie dialogowym *Time Shift* będziesz mógł określić zakres przesunięcia czasu dla całego pliku zawierającego przechwycone pakiety bądź też dla jedynie wybranych pakietów. W przykładzie pokazanym na rysunku 4.9 zdecydowałem się przesunąć o dwie minuty i pięć sekund czas każdego pakietu znajdującego się w przechwyconych danych.



Rysunek 4.9. Okno dialogowe Time Shift

Konfiguracja opcji przechwytywania danych

Wyjątkowo prosty proces przechwytywania danych został przedstawiony w rozdziale 3. W pokazanym na rysunku 4.10 oknie dialogowym *Capture Interfaces* narzędzie Wireshark oferuje znacznie więcej opcji związanych z przechwytywaniem



Rysunek 4.10. Okno dialogowe Capture Interfaces

danych. Wyświetlenie tego okna dialogowego następuje po wybraniu opcji *Capture/Options*.

Okno dialogowe *Capture Options* ma więcej wodotrysków, niż będziesz w stanie wykorzystać, umożliwiających jak największą elastyczność podczas przechwytywania pakietów. Opcje zostały zgrupowane w trzech kartach, które teraz omówię.

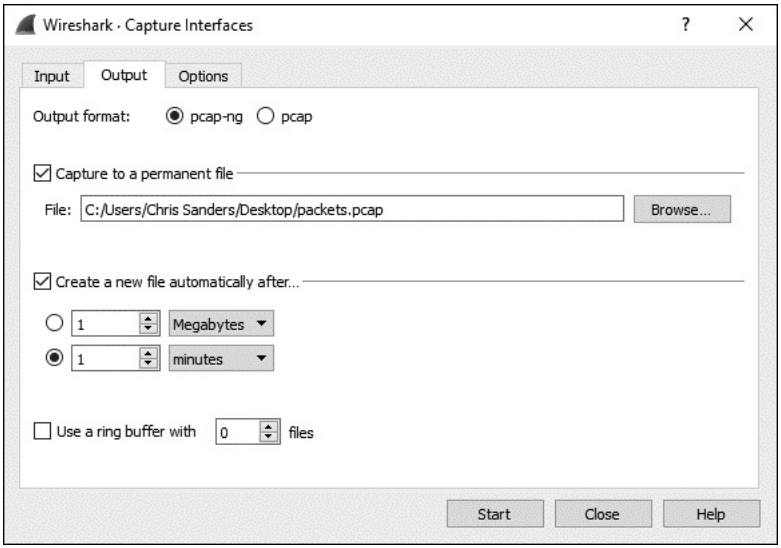
Karta Input

Głównym przeznaczeniem karty *Input* jest wyświetlenie wszystkich interfejsów, które można wykorzystać do przechwytywania pakietów, oraz dostarczenie pewnych informacji podstawowych o poszczególnych interfejsach. Obejmuje to czytelną dla człowieka nazwę interfejsu podawaną przez system operacyjny, wykres ruchu sieciowego pokazujący przepustowość tego interfejsu oraz dodatkowe opcje konfiguracyjne, takie jak stan trybu mieszanego i wielkość bufora. Ostatnia kolumna po prawej stronie (niepokazana na rysunku 4.10) wyświetla dane dotyczące zastosowanego filtru, użyciem których zajmiemy się w dalszej części rozdziału.

W tej karcie możesz kliknąć praktycznie dowolną opcję i zmienić jej wartość. Na przykład w celu wyłączenia trybu mieszanego w interfejsie należy kliknąć pole w kolumnie *Promiscuous*, a następnie włączyć lub wyłączyć ten tryb za pomocą odpowiedniej opcji w rozwijanym menu.

Karta Output

W pokazanej na rysunku 4.11 karcie *Output* możesz skonfigurować automatyczne przechowywanie przechwyconych pakietów w pliku zamiast najpierw przechwytywania danych, a dopiero później ich zapisywania w pliku. Dostępne tutaj opcje



Rysunek 4.11. Karta Output w oknie dialogowym Capture Interfaces

oferują dużą elastyczność w zarządzaniu sposobem zapisu pakietów. Dane mogą być zapisywane w jednym pliku, w zestawie plików, a nawet można użyć bufora cyklicznego (ang. *ring buffer*) do zarządzania liczbą tworzonych plików. Do bufora cyklicznego wrócimy za chwilę. Włączenie opcji zapisu danych do pliku (lub plików) wymaga podania pełnej ścieżki dostępu w polu *File*. Ewentualnie można kliknąć przycisk *Browse…*, wybrać katalog, a następnie podać nazwę dla pliku.

Podczas przechwytywania ogromnego ruchu sieciowego lub przeprowadzania długotrwałej operacji przechwytywania danych użyteczne jest utworzenie zestawu plików. Zestaw ten to grupa wielu plików, każdy z nich zawiera dane spełniające określony warunek. Aby wykorzystać zestaw plików, należy użyć opcji *Create a new file automatically after…*.

Przy zarządzaniu zestawem plików narzędzie Wireshark korzysta z różnych wyzwalaczy opartych na wielkości pliku lub na warunku dotyczącym czasu. Włączenie tych wyzwalaczy wymaga zaznaczenia pola wyboru znajdującego się obok opcji opartych na wielkości pliku lub na czasie, a także podania wartości oraz jednostki powodującej aktywację wyzwalacza. Przykładowo: możesz zdefiniować wyzwalacz tworzący nowy plik po przechwyceniu każdego 1 MB danych lub po upływie minuty przechwytywania danych (zob. rysunek 4.12).

Opcja *Use a ring buffer with…* pozwala na określenie liczby plików utworzonych w grupie; po jej przekroczeniu Wireshark rozpocznie nadpisywanie plików. Pojęcie *bufora cyklicznego* ma wiele znaczeń w informatyce. W narzędziu Wireshark oznacza zestaw plików, gdzie po zapisaniu ostatniego pliku rozpocznie się nadpisywanie pierwszego, kiedy pojawią się kolejne dane konieczne do zachowania. Innymi słowy, ta opcja jest wykorzystywana przez narzędzie Wireshark do zastosowania metody FIFO (ang. *first in, first out* — pierwszy na wejściu,

Name	Date modified	Type	Size
intervalcapture_00001_20151009141804	10/9/2017 2:19 PM	File	172 KB
intervalcapture_00002_20151009141904	10/9/2017 2:20 PM	File	25 KB
intervalcapture_00003_20151009142004	10/9/2017 2:21 PM	File	3,621 KB
intervalcapture_00004_20151009142104	10/9/2017 2:22 PM	File	52 KB
intervalcapture_00005_20151009142204	10/9/2017 2:23 PM	File	47 KB
intervalcapture_00006_20151009142304	10/9/2017 2:24 PM	File	37 KB

Rysunek 4.12. Zestaw plików utworzonych przez narzędzie Wireshark w odstępie jednej minuty

pierwszy na wyjściu) podczas zapisu wielu plików. Możesz zaznaczyć tę opcję i zdefiniować maksymalną liczbę plików używanych przez bufor cykliczny. Przykładowo: możesz zdecydować się na użycie zestawu plików do zapisu przechwytywanych danych i określić tworzenie nowego pliku co godzinę, a maksymalną liczbę plików ustalić na 6. W takim przypadku po utworzeniu ostatniego, szóstego pliku bufor cykliczny rozpocznie nadpisywanie pierwszego pliku, zamiast utworzyć siódmy. W ten sposób na dysku twardym będzie się znajdowało maksymalnie sześć plików zawierających przechwycone dane (w tym przypadku z sześciu ostatnich godzin) i nadal będzie zachowana możliwość zapisu nowych danych.

Ponadto w karcie *Output* można również określić, kiedy ma być używany format pliku *.pcapng*. Jeżeli przechwycone pakiety zamierzasz analizować za pomocą narzędzia, które nie potrafi przetwarzać plików *.pcapng*, wtedy lepiej zdecyduj się na tradycyjny format *.pcap*.

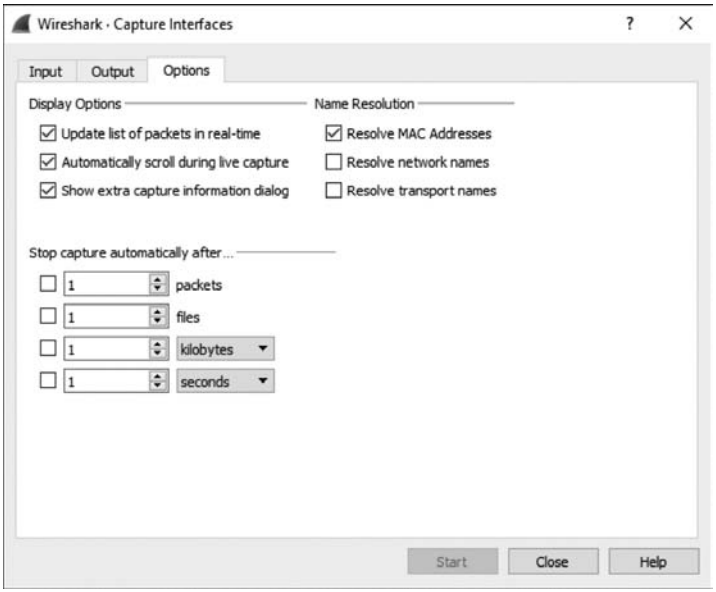
Karta Options

Karta *Options* zawiera wiele innych opcji związanych z przechwytywaniem pakietów. Opcje te zgrupowano w sekcjach *Display Options*, *Name Resolution* i *Stop capture automatically after…* (rysunek 4.13).

Sekcja Display Options

Sekcja *Display Options* określa sposób wyświetlania pakietów po ich przechwyceniu. Działanie opcji *Update list of packets in real-time* (uaktualniaj listę pakietów w czasie rzeczywistym) jest oczywiste; ponadto może być ona połączona z opcją *Automatically scrolling during live capture* (automatyczne przewijanie w panelu *Live Capture*). Po włączeniu obu opcji na ekranie wyświetlą się wszystkie przechwycone pakiety, przy czym przechwytywane pakiety będą wyświetlane natychmiast.

OSTRZEŻENIE *Połączenie opcji* Update list of packets in real-time *i* Automatically scrolling during live capture *może spowodować znaczne obciążenie procesora podczas przechwytywania dużych ilości danych. Jeżeli nie masz szczególnego powodu do wyświetlania pakietów w czasie rzeczywistym, najlepiej wyłącz obie opcje.*



Rysunek 4.13. Karta Options w oknie dialogowym Capture Interfaces

Opcja *Show extra capture information* pozwala na wyświetlenie lub ukrycie małego okna pokazującego liczbę oraz wartość procentową pakietów przechwyconych dla danego protokołu. Dane są posortowane według protokołów. Z reguły wyświetlam to okno, ponieważ nie korzystam z omówionej wcześniej opcji *Automatically scrolling during live capture*.

Sekcja Name Resolution

Opcje w tej sekcji umożliwiają włączenie automatycznego określania nazw MAC (warstwa 2.), sieci (warstwa 3.) i transportu (warstwa 4.) dla przechwytywanych danych. Szczegółowe omówienie określania nazw w narzędziu Wireshark oraz wad tego procesu zostanie przedstawione w rozdziale 5.

Sekcja Stop Capture

Sekcja *Stop capture automatically after…* pozwala zatrzymać trwającą operację przechwytywania danych po wystąpieniu określonego wyzwalacza. Podobnie jak w przypadku zestawu plików, także tutaj wyzwalacz może opierać się na wielkości pliku, odstępach czasu, jak również na liczbie pakietów. Te opcje możesz wykorzystywać w połączeniu z omówionymi wcześniej opcjami dotyczącymi zestawu plików w karcie *Output*.

Używanie filtrów

Filtry pozwalają dokładnie wskazać dane, które chcesz przeanalizować. Ujmując rzecz najprościej: filtr to wyrażenie definiujące kryteria dołączania pakietów do przechwyconych danych lub usuwania pakietów z tych danych. Jeżeli dane zawierają nieinteresujące Cię pakiety, możesz utworzyć odpowiedni filtr powodujący pozbycie się tych pakietów. Jeśli natomiast chcesz otrzymywać wyłącznie określone pakiety, wystarczy utworzyć filtr pokazujący jedynie interesujące Cię pakiety.

Narzędzie Wireshark oferuje dwa podstawowe rodzaje filtrów:

- *Filtr przechwytywania* zostaje zdefiniowany na początku operacji przechwytywania danych i zawiera tylko te pakiety, które wskazano do dołączenia w danym wyrażeniu.
- *Filtr wyświetlania* zostanie zastosowany względem istniejącego zestawu przechwyconych pakietów w celu ukrycia niepożądanych lub wyświetlenia interesujących Cię pakietów na podstawie określonego wyrażenia.

W pierwszej kolejności zapoznamy się z filtrami przechwytywania.

Filtry przechwytywania

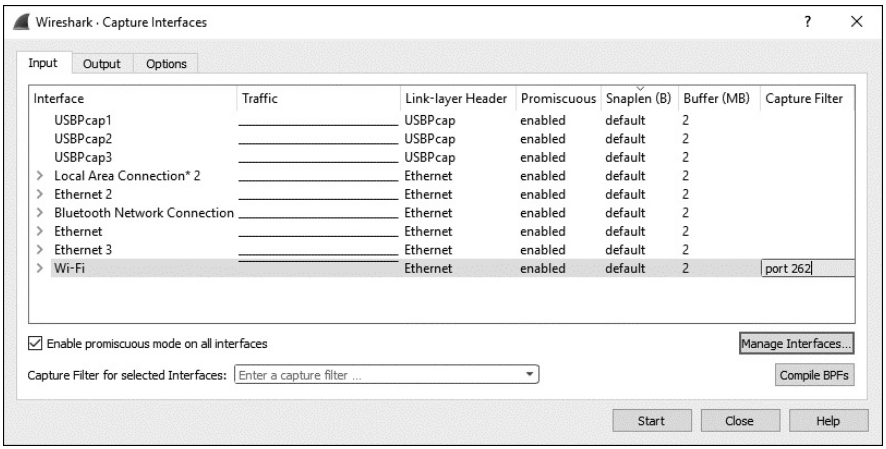
Filtry przechwytywania są używane w faktycznym procesie przechwytywania pakietów w celu ograniczenia liczby pakietów, które zostaną poddane analizie. Jednym z podstawowych powodów używania filtru przechwytywania jest zachowanie maksymalnej wydajności działania. Jeżeli wiesz, że nie będziesz analizował określonych form ruchu sieciowego, możesz odfiltrować jego dane za pomocą filtru przechwytywania. W ten sposób zaoszczędzisz nieco mocy procesora, która musiałaby zostać wykorzystana do przechwycenia nieinteresujących Cię pakietów.

Możliwość utworzenia własnych filtrów przechwytywania jest bardzo użyteczna w przypadku obsługi ogromnych ilości danych. Proces analizy można znacznie przyspieszyć poprzez zagwarantowanie, że patrzysz tylko na te pakiety, które mają związek z rozwiązywanym problemem.

Prosty przykład użycia filtru przechwytywania to sytuacja, w której przechwytujesz ruch z serwera sieciowego o wielu rolach. Przypuśćmy, że rozwiązujesz problem z usługą udostępnianą na porcie 262. Jeżeli analizowany serwer udostępnia także wiele innych usług na różnych portach, to wyszukanie i przeanalizowanie ruchu przepływającego jedynie przez port 262 będzie samo w sobie już wymagającym zadaniem. Aby przechwycić jedynie ruch przepływający przez port 262, możesz użyć filtru przechwytywania. W tym celu przejdź do omówionego wcześniej okna dialogowego *Capture Interfaces* i wykonaj następujące kroki:

1. Wybierz opcję menu *Capture/Options* znajdującą się obok nazwy interfejsu, z którego chcesz przechwycić dane. Na ekranie zostanie wyświetlone okno dialogowe *Capture Interfaces*.
2. Wybierz interfejs, z którego będą przechwytywane pakiety, a następnie w ostatniej kolumnie znajdującej się po prawej stronie wskaż filtr przechwytywania.

3. Filtr przechwytywania możesz zastosować poprzez podanie odpowiedniego wyrażenia w polu tekstowym znajdującym się obok przycisku *Capture Filter*. W omawianym przykładzie interesuje nas tylko ruch przepływający przez port 262, zatem w polu tym wpisujemy `port 262`, jak pokazano na rysunku 4.14. (Wprowadzone tutaj wyrażenie zostanie dokładnie omówione w kolejnej sekcji). Kolor komórki powinien się zmienić na szary, co wskazuje wprowadzenie prawidłowego wyrażenia. Jeżeli wpiszesz niepoprawne wyrażenie, komórka zmieni kolor na czerwony.



Rysunek 4.14. Zdefiniowanie filtru przechwytywania w oknie dialogowym Capture Interfaces

4. Po zdefiniowaniu filtru wystarczy nacisnąć przycisk *Start* rozpoczynający przechwytywanie pakietów.

Po zebraniu odpowiedniej wielkości próbki danych zobaczysz, że próbka zawiera jedynie dane ruchu sieciowego przepływającego przez port 262. Dzięki temu możesz znacznie efektywniej przeprowadzić analizę tych danych.

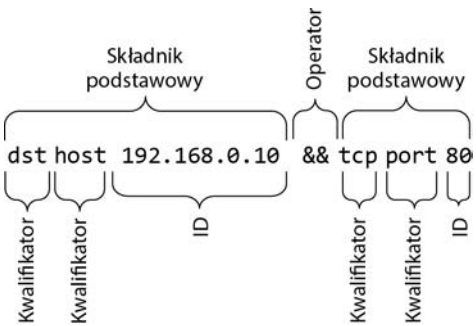
Przechwytywanie i składnia BPF

Filtry przechwytywania są stosowane przez libpcap/WinPcap i używają składni BPF (ang. *berkeley packet filter*). Składnia ta jest stosowana w wielu aplikacjach typu sniffer pakietów najczęściej z powodu wykorzystywania przez te aplikacje bibliotek libpcap/WinPcap, które pozwalają na stosowanie składni BPF. Znajomość składni BPF ma więc znacznie krytyczne, jeśli chcesz zagłębić się w sieć na poziomie pakietów.

Filtr utworzony z użyciem składni BPF jest nazywany *wyrażeniem*, a każde wyrażenie składa się z co najmniej jednego *składnika podstawowego*. Z kolei składniki składają się z co najmniej jednego *kwalifikatora* (kwalifikatory wymieniono w tabeli 4.2) wraz z identyfikatorem, jak pokazano na rysunku 4.15.

Tabela 4.2. Kwalifikatory składni BPF

Kwalifikator	Opis	Przykłady
typ	określa nazwę lub numer identyfikatora, do którego się odwołuje	`host`, `net`, `port`
kierunek	określa kierunek transmisji do urządzenia o podanej nazwie lub identyfikatorze albo od takiego urządzenia	`src`, `dst`
protokół	ogranicza dopasowanie do konkretnego protokołu	`ether`, `ip`, `tcp`, `udp`, `http`, `ftp`



Rysunek 4.15. Prosty filtr przechwytywania

Biorąc pod uwagę komponenty wyrażenia, kwalifikator `dst host` i identyfikator `192.168.0.10` tworzą postać składnika podstawowego. Taki składnik jest wyrażeniem, które spowoduje przechwycenie ruchu sieciowego pochodzącego jedynie z adresu IP 192.168.0.10.

W celu łączenia składników i tworzenia bardziej zaawansowanych wyrażeń możesz wykorzystać operatory logiczne. Poniżej wymieniono trzy operatory logiczne dostępne podczas tworzenia wyrażeń:

- operator konkatenacji `AND` (`&&`);
- operator alternatywy `OR` (`||`);
- operator negacji `NOT` (`!`).

Przykładowo: poniższe wyrażenie spowoduje przechwycenie ruchu sieciowego pochodzącego z adresu IP 192.168.0.10 oraz z portu 80 lub do tego portu:

```
src 192.168.0.10 && port 80
```

Filtr nazwy komputera i adresu

Większość tworzonych przez Ciebie filtrów będzie dotyczyła danego urządzenia sieciowego lub grupy urządzeń. W zależności od sytuacji filtrowanie może opierać się na adresie MAC urządzenia, adresie IPv4, IPv6 lub nazwie komputera DNS.

Przykładowo: chcesz się dowiedzieć, jaki ruch sieciowy przepływa przez określony komputer podczas komunikacji z serwerem znajdującym się w danej

sieci. Dla serwera możesz więc utworzyć filtr, używając kwalifikatora `host`. Tak przygotowany filtr będzie przechwytywał cały ruch sieciowy związany z adresem IPv4 interesującego Cię komputera:

```
host 172.16.16.149
```

Jeżeli w sieci używasz protokołu IPv6, to użyty w kwalifikatorze `host` filtr musi opierać się na adresie IPv6, jak przedstawiono poniżej:

```
host 2001:db8:85a3::8a2e:370:7334
```

W kwalifikatorze `host` można także użyć filtru opartego na nazwie komputera, na przykład:

```
host serwertestowy2
```

Jeśli masz obawy, że adres IP interesującego Cię komputera może ulec zmianie, możesz przygotować filtr również na podstawie adresu MAC urządzenia, podając kwalifikator `ether`:

```
ether host 00-1a-a0-52-e2-a0
```

Kwalifikatory kierunku transmisji danych są bardzo często używane w połączeniu z powyższymi przykładami w celu przechwytywania ruchu przychodzącego do określonego komputera lub wychodzącego z niego. Przykładowo: aby przechwycić jedynie ruch przychodzący do danego komputera, można użyć kwalifikatora `src`:

```
src host 172.16.16.149
```

Aby przechwycić jedynie dane opuszczające serwer o adresie 172.16.16.149 i przeznaczone dla danego komputera, możesz użyć kwalifikatora `dst`:

```
dst host 172.16.16.149
```

Kiedy nie podajesz kwalifikatora typu (`host`, `net` lub `port`) wraz ze składnikiem podstawowym, domyślnie zakłada się, że został użyty kwalifikator `host`. Dlatego poniższe wyrażenie jest odpowiednikiem zaprezentowanego w poprzednim przykładzie:

```
dst 172.16.16.149
```

Filtry portów

Oprócz filtrowania na podstawie komputerów można przeprowadzić filtrowanie na podstawie portów używanych w pakietach. Filtrowanie na podstawie portów można wykorzystać do filtrowania na podstawie usług i aplikacji używających standardowych portów. Poniżej przedstawiono prosty filtr przechwytujący jedynie ruch przepływający przez port 8080:

```
port 8080
```

W celu przechwycenia całego ruchu sieciowego poza przepływającym przez port 8080 można wykorzystać następujące wyrażenie:

```
!port 8080
```

Filtr portu można połączyć z kwalifikatorem kierunku transmisji danych. Przykładowo: aby przechwycić jedynie ruch sieciowy przychodzący do serwera WWW nasłuchującego na standardowym porcie HTTP 80, należy użyć kwalifikatora `dst`:

```
dst port 80
```

Filtry protokołów

Filtry protokołów umożliwiają filtrowanie pakietów na podstawie określonych protokołów. Są wykorzystywane w celu dopasowania protokołów innych niż warstwy aplikacji, przy czym te protokoły nie mogą być zdefiniowane poprzez podanie określonego portu. Dlatego jeżeli chcesz zobaczyć jedynie ruch sieciowy ICMP, możesz użyć następującego filtru:

```
icmp
```

Aby zobaczyć cały ruch sieciowy poza IPv6, należy użyć filtru:

```
!ip6
```

Filtry pola protokołu

Prawdziwa potęga składni BPF kryje się w możliwości przeanalizowania każdego bajta nagłówka protokołu w celu utworzenia szczegółowych filtrów opartych na tych danych. Omówione w tej sekcji filtry zaawansowane umożliwiają pobieranie określonej liczby bajtów z pakietu rozpoczynającego się we wskazanym położeniu.

Przykładowo: chcesz przeprowadzić filtrowanie na podstawie pola typu nagłówka ICMP. Pole to znajduje się na początku pakietu, czyli jego pozycja wynosi 0.

Aby określić konkretne położenie w pakiecie, należy podać konkretną pozycję, używając do tego nawiasu kwadratowego umieszczonego obok kwalifikatora protokołu — w omawianym przykładzie to `icmp[0]`. Wartością zwrotną będzie jednobajtowa liczba całkowita, względem której możemy przeprowadzić operację porównania. Na przykład aby pobrać jedynie pakiety ICMP określające, że pakiet nie dotarł do celu (typ 3), w wyrażeniu filtru należy użyć operatora równości, co przedstawiono poniżej:

```
icmp[0] == 3
```

W celu przeanalizowania jedynie pakietów ICMP przedstawiających żądania `echo` (typ 8) lub odpowiedzi na nie (typ 0) należy użyć dwóch składników podstawowych wraz z operatorem `OR`:

```
icmp[0] == 8 || icmp[0] == 0
```

Przedstawione powyżej filtry działają doskonale, ale przeprowadzają filtrowanie jedynie na podstawie jednobajtowych informacji pochodzących z nagłówka pakietu. Na szczęście można również określić wielkość danych zwracanych przez wyrażenie filtru poprzez jej podanie w nawiasie kwadratowym tuż po wartości określającej pozycję. Obie liczby muszą być rozdzielone dwukropkiem.

Przykładowo: chcemy utworzyć filtr przechwytujący wszystkie pakiety ICMP, które nie dotarły do celu — są oznaczone jako typ 3 i kod 1. To jednobajtowe pola umieszczone obok siebie w pozycji 0 nagłówka pakietu. Naszym celem jest więc utworzenie filtru sprawdzającego dwa bajty danych znajdujące się na początku nagłówka pakietu (pozycja wynosi 0) i porównanie ich względem wartości szesnastkowej 0301 (typ 3, kod 1). Wyrażenie ma więc następującą postać:

```
icmp[0:2] == 0x0301
```

Bardzo często zdarza się przechwytywanie jedynie pakietów TCP wraz z ustawioną opcją RST. Szczegółowe omówienie protokołu TCP znajdziesz w rozdziale 8. Teraz musisz jedynie wiedzieć, że opcje pakietu TCP są umieszczone w pozycji 13. To interesujące pole, ponieważ jako pole opcji ma wielkość jednego bajta, a poszczególne opcje są identyfikowane za pomocą pojedynczych bitów w tym bajcie. Jak wyjaśnię dokładnie w dodatku B, każdy bit w bajcie przedstawia pewną liczbę o podstawie 2. Wspomniany bit zawierający opcję jest określony przez numer bit przedstawiający bit, tak więc pierwszy to 1, drugi to 2, trzeci to 4, czwarty to 8 itd. W pakiecie TCP można ustawić jednocześnie wiele opcji, co oznacza brak możliwości efektywnego filtrowania za pomocą prostego wyrażenia `tcp[13]`, ponieważ ten bit RST mógł zostać ustawiony z różnych powodów.

Dlatego konieczne jest dokładne wskazanie w bajcie położenia, które ma zostać przeanalizowane. W tym celu do składnika należy dołączyć znak `&` i podać położenie tego składnika. Opcja RST jest przedstawiana za pomocą bitu o liczbie 4.

Opcja RST jest bitem przedstawiającym numer 4 w bajcie, a ustawienie temu bitowi wartości 4 wskazuje na ustawienie opcji RST. Gotowy filtr ma następującą postać:

```
tcp[13] & 4 == 4
```

Aby zobaczyć wszystkie pakiety wraz z ustawioną opcją PSH, która w omawianym bajcie jest przedstawiona za pomocą bitu znajdującego się w położeniu 8, filtr powinien mieć postać:

```
tcp[13] & 8 == 8
```

Przykładowe wyrażenia filtrów przechwytywania danych

Przekonasz się, że sukces lub porażka podczas analizy pakietów bardzo często zależy od Twoich możliwości w dziedzinie tworzenia filtrów odpowiednich do danej sytuacji. W tabeli 4.3 wymieniono kilka przykładowych filtrów przechwytywania danych, których używam najczęściej.

Tabela 4.3. Najczęściej używane filtry przechwytywania danych

Filtr	Opis
`tcp[13] & 32 == 32`	pakiety TCP wraz z ustawioną opcją URG
`tcp[13] & 16 == 16`	pakiety TCP wraz z ustawioną opcją ACK
`tcp[13] & 8 == 8`	pakiety TCP wraz z ustawioną opcją PSH
`tcp[13] & 4 == 4`	pakiety TCP wraz z ustawioną opcją RST
`tcp[13] & 2 == 2`	pakiety TCP wraz z ustawioną opcją SYN
`tcp[13] & 1 == 1`	pakiety TCP wraz z ustawioną opcją FIN
`tcp[13] == 18`	pakiety TCP SYN-ACK
`ether host 00:00:00:00:00:00` (adres zastąp swoim adresem MAC)	ruch do podanego adresu MAC oraz z tego adresu
`!ether host 00:00:00:00:00:00` (adres zastąp swoim adresem MAC)	ruch, który nie przychodzi do podanego adresu MAC oraz nie wychodzi z niego
`broadcast`	tylko ruch rozgłaszający
`icmp`	tylko ruch ICMP
`icmp[0:2] == 0x0301`	urządzenie docelowe ICMP jest niedostępne, komputer jest niedostępny
`ip`	tylko ruch IPv4
`ip6`	tylko ruch IPv6
`udp`	tylko ruch UDP

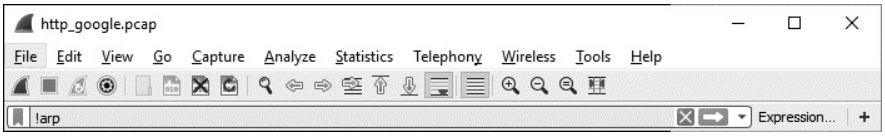
Filtry wyświetlania

Filtr wyświetlania to ten, który po zastosowaniu względem pliku zawierającego przechwycone dane nakazuje narzędziu Wireshark wyświetlenie jedynie pakietów spełniających kryteria tego filtru. Filtr wyświetlania można zdefiniować w polu *Filter* znajdującym się nad panelem *Packet List*.

Filtry wyświetlania są używane częściej niż filtry przechwytywania danych, ponieważ pozwalają filtrować pakiety bez rzeczywistego pominięcia pozostałych danych zebranych w pliku. W ten sposób, jeśli będziesz musiał powrócić do początkowego zbioru zebranych danych, wystarczy po prostu usunąć wyrażenie filtru. Tego rodzaju filtry mają też znacznie większe możliwości dzięki istniejącej dla narzędzia Wireshark obszernej biblioteki dekoderów pakietów.

Filtr wyświetlania możesz wykorzystać do ukrycia nieistotnego w danej chwili ruchu sieciowego zebranego w pliku przechwyconych danych. Przykładowo: możesz ukryć ruch pakietów ARP w panelu *Packet List*, kiedy te pakiety nie mają żadnego związku z aktualnie rozwiązywanym problemem. Jednak ponieważ pakiety ARP mogą być użyteczne później, lepszym rozwiązaniem jest ich tymczasowe ukrycie zamiast trwałego usunięcia.

Aby odfiltrować wszystkie pakiety ARP w oknie przechwytywania, po prostu umieść kursor w polu tekstowym *Filter* znajdującym się na górze panelu *Packet List*, a następnie wprowadź wyrażenie `!arp`, które spowoduje ukrycie wszystkich pakietów ARP w panelu *Packet List* (zob. rysunek 4.16). Usunięcie filtru następuje po naciśnięciu przycisku *X*. Natomiast w celu zapisania filtru do późniejszego użycia kliknij przycisk +.

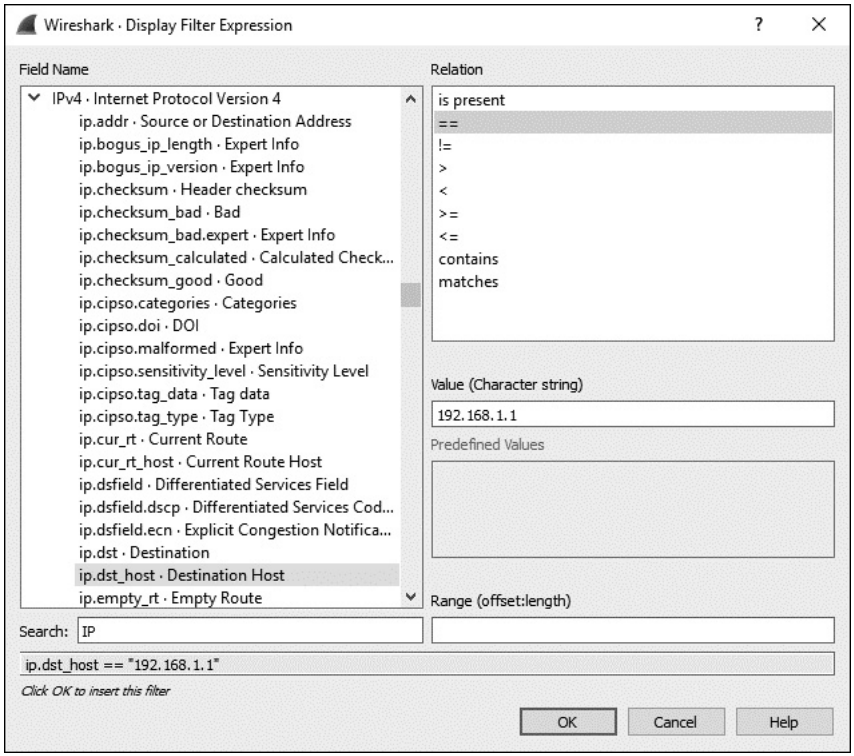


Rysunek 4.16. Utworzenie filtru wyświetlania za pomocą pola Filter znajdującego się nad panelem Packet List

Mamy dwa sposoby na zastosowanie filtrów wyświetlania. Pierwszy polega na bezpośrednim użyciu filtru za pomocą odpowiedniej składni, jak pokazałem w omawianym przykładzie. Drugi polega na wykorzystaniu okna dialogowego *Display Filter Expression* do iteracyjnego utworzenia filtru. To jest łatwiejsza metoda, gdy dopiero zaczynasz używanie filtrów. Dokładnie omówię teraz obie metody, począwszy od łatwiejszej.

Okno dialogowe Filter Expression

Pokazane na rysunku 4.17 okno dialogowe *Display Filter Expression* znacznie ułatwia początkującym użytkownikom narzędzia Wireshark tworzenie filtrów przechwytywania danych i filtrów wyświetlania. Aby wyświetlić to okno, należy kliknąć przycisk *Expression* na pasku narzędziowym *Filter*.



Rysunek 4.17. Okno dialogowe Display Filter Expression umożliwia łatwe tworzenie filtrów w narzędziu Wireshark

Po lewej stronie okna dialogowego znajdują się wszystkie dostępne do użycia protokoły. W tych polach można określić wszystkie możliwe kryteria filtru. Aby utworzyć filtr, wykonaj następujące kroki:

1. W celu wyświetlenia kryteriów związanych z danym protokołem rozwiń ten protokół, klikając symbol strzałki znajdujący się obok jego nazwy. Po znalezieniu szukanego kryterium, na którym będzie oparty filtr, kliknij je w celu zaznaczenia.
2. Następnie określ, w jaki sposób wybrane kryterium będzie zależało od zdefiniowanej dla niego wartości. Dostępne opcje to: równy, większy niż, mniejszy niż itd.
3. Utwórz wyrażenie filtru poprzez podanie wartości kryterium, która będzie miała związek z wybranym polem. Tę wartość możesz zdefiniować sam lub możesz wybrać jedną ze zdefiniowanych w narzędziu Wireshark.
4. Gotowy filtr zostanie wyświetlony na dole ekranu. Po przygotowaniu filtru kliknij przycisk *OK*; spowoduje to wstawienie go do paska narzędziowego *Filter*.

Okno dialogowe *Display Filter Expression* to doskonała pomoc dla początkujących użytkowników. Po nabyciu pewnej wprawy przekonasz się, że ręczne tworzenie wyrażeń filtrów znacznie zwiększa ich efektywność. Składnia wyrażenia filtru wyświetlania jest bardzo prosta i daje ogromne możliwości.

Struktura składni wyrażenia filtru (trudniejszy sposób)

Kiedy nabędziesz nieco większej wprawy w pracy z narzędziem Wireshark, zaczniesz bezpośrednio w oknie głównym używać składni filtru wyświetlania, aby w ten sposób przyspieszyć pracę. Na szczęście składnia filtrów wyświetlania opiera się na standardowym schemacie i jest dość łatwa. W większości przypadków jest to schemat oparty na protokole stosującym format *protokół.funkcja.podfunkcja*, o czym się możesz przekonać w oknie dialogowym *Display Filter Expression*. Teraz przedstawię kilka przykładów.

Filtry przechwytywania lub wyświetlania danych najczęściej będziesz wykorzystywał do przeprowadzania filtrowania na podstawie danego protokołu. Załóżmy, że rozwiązujesz problem związany z TCP, więc w pliku zawierającym przechwycone dane chcesz widzieć tylko ruch sieciowy TCP. W takim przypadku prosty filtr `tcp` jest idealnym rozwiązaniem.

Spójrzmy jednak na to z innej strony. Wyobraź sobie, że w trakcie procesu usuwania problemu związanego z TCP bardzo często używasz polecenia `ping`, generując w ten sposób znaczną ilość ruchu sieciowego ICMP. Ruch ICMP możesz ukryć w pliku zawierającym przechwycone dane poprzez użycie wyrażenia filtru o postaci `!icmp`.

Operatory porównania umożliwiają porównywanie wartości. Przykładowo: podczas usuwania problemów w sieciach TCP/IP bardzo często zachodzi potrzeba wyświetlenia wszystkich pakietów odwołujących się do konkretnego adresu IP. Operator porównania (`==`) pozwala na utworzenie filtru wyświetlającego wszystkie pakiety powiązane z adresem IP, na przykład 192.168.0.1:

```
ip.addr==192.168.0.1
```

Załóżmy, że chcesz wyświetlić tylko te pakiety, których wielkość jest mniejsza niż 128 bajtów. W takim przypadku można użyć operatora „mniejszy lub równy" (`<=`) w celu przygotowania następującego wyrażenia filtru:

```
frame.len <= 128
```

Operatory porównania wykorzystywane w narzędziu Wireshark zostały wymienione w tabeli 4.4.

Operatory logiczne pozwalają łączyć wiele wyrażeń filtrów w pojedyncze wyrażenie, co znacznie zwiększa efektywność działania filtru.

Przykładowo: chcemy wyświetlić pakiety wysyłane tylko do dwóch adresów IP. W tym celu możemy użyć operatora `OR` do utworzenia pojedynczego wyrażenia filtru, które będzie wyświetlało pakiety zawierające jeden ze zdefiniowanych adresów:

Tabela 4.4. Operatory porównania stosowane w wyrażeniach filtrów narzędzia Wireshark

Operator	Opis
==	równość
!=	nierówność
>	większy niż
<	mniejszy niż
>=	większy lub równy
<=	mniejszy lub równy

```
ip.addr==192.168.0.1 or ip.addr==192.168.0.2
```

Operatory logiczne wykorzystywane w narzędziu Wireshark zostały wymienione w tabeli 4.5.

Tabela 4.5. Operatory logiczne stosowane w wyrażeniach filtrów narzędzia Wireshark

Operator	Opis
and	obydwa warunki muszą przyjąć wartość true
or	jeden z warunków musi przyjąć wartość true
xor	jeden i tylko jeden warunek może przyjąć wartość true
not	żaden z warunków nie może przyjąć wartości true

Przykładowe wyrażenia filtrów wyświetlania

Koncepcje związane z tworzeniem wyrażeń filtrów są całkiem proste, ale czasem podczas rozwiązywania różnych problemów trzeba używać kilku określonych słów kluczowych i operatorów. W tabeli 4.6 wymieniono filtry wyświetlania, z których najczęściej korzystam. Pełną listę filtrów wyświetlania w narzędziu Wireshark znajdziesz w dokumentacji dostępnej na stronie http://www.wireshark.org/docs/dfref/.

Zapis filtrów

Kiedy rozpoczniesz tworzenie ogromnej liczby filtrów przechwytywania i wyświetlania danych, przekonasz się, że pewne z nich są często wykorzystywane. Na szczęście filtru nie musisz wpisywać za każdym razem, gdy chcesz go użyć, ponieważ narzędzie Wireshark pozwala zapisywać filtry i później je wykorzystywać. Aby zapisać samodzielnie przygotowany filtr przechwytywania danych, wykonaj następujące kroki.

1. Wybierz opcję menu *Capture/Capture Filters* w celu wyświetlenia okna dialogowego *Capture Filter*.
2. Utwórz nowy filtr, klikając przycisk + znajdujący się po lewej stronie wyświetlonego okna dialogowego.

Tabela 4.6. Najczęściej używane filtry wyświetlania

Filtr	Opis
`!tcp.port==3389`	wyłącznie ruch RDP
`tcp.flags.syn==1`	pakiety TCP wraz z ustawioną opcją SYN
`tcp.flags.reset==1`	pakiety TCP wraz z ustawioną opcją RST
`!arp`	wyłącznie ruch ARP
`http`	cały ruch HTTP
`tcp.port==23 || tcp.port 21`	ruch administracyjny w postaci zwykłego tekstu (Telnet lub FTP)
`smtp || pop || imap`	ruch poczty elektronicznej w postaci zwykłego tekstu (SMTP, POP lub IMAP)

3. W polu *Filter Name* podaj nazwę filtru.
4. W polu *Filter String* podaj rzeczywiste wyrażenie filtru.
5. Kliknij przycisk *OK* w celu zapisania wyrażenia filtru na liście.

Aby zapisać samodzielnie przygotowany filtr wyświetlania danych, wykonaj następujące kroki.

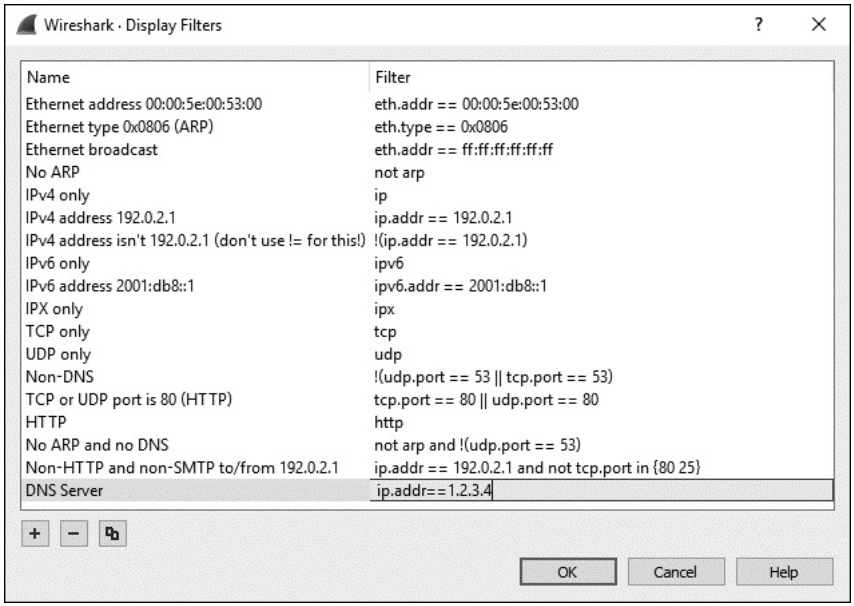
1. Wpisz wyrażenie filtru w pasku *Filter* wyświetlanym nad panelem *Packet List* w oknie głównym programu. Następnie kliknij przycisk *zakładki* znajdujący się po lewej stronie paska.
2. Kliknij opcję *Save this Filter*, a lista zapisanych filtrów zostanie wyświetlona w oddzielnym oknie dialogowym. Możesz w nim podać nazwę filtru, a następnie kliknąć przycisk *OK* (rysunek 4.18).

Dodanie filtrów wyświetlania do paska narzędzi

Jeżeli masz często wykorzystywane filtry, jednym z najłatwiejszych sposobów ich użycia jest umieszczenie ich na pasku *Filter* wyświetlanym tuż nad panelem *Packet List*. W tym celu wykonaj wymienione poniżej kroki.

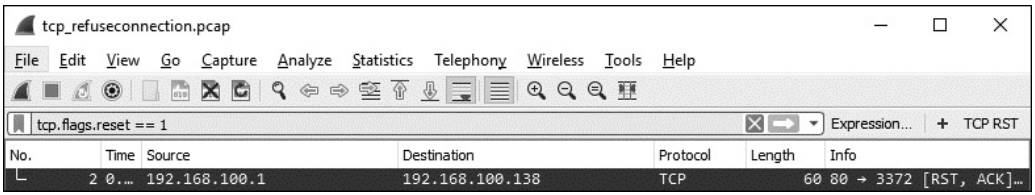
1. Wpisz wyrażenie filtru w pasku *Filter* wyświetlanym nad panelem *Packet List* w oknie głównym programu. Następnie kliknij przycisk + znajdujący się po prawej stronie paska.
2. Pod paskiem *Filter* zostanie wyświetlony nowy pasek, w którym będzie można podać nazwę filtru w polu *Label* (rysunek 4.19). Ta etykieta będzie używana do przedstawienia filtru na pasku narzędzi. Po jej podaniu kliknij przycisk *OK* — na pasku *Filter* pojawi się utworzony skrót do wprowadzonego wyrażenia filtru.

Jak możesz zobaczyć na rysunku 4.20, utworzyłem skrót prowadzący do filtru, który pozwala na szybkie wyświetlenie wszystkich pakietów TCP wraz z włączoną opcją RST. Filtry dodane do paska narzędzi są zapisywane w profilu konfiguracyjnym (więcej informacji na temat tych profili przedstawiłem w rozdziale 3.).



Rysunek 4.18. Okno dialogowe Display Filter umożliwia zapis wyrażeń filtrów bezpośrednio z paska narzędzi okna głównego Wireshark

Rysunek 4.19. Dodanie do paska Filter skrótu prowadzącego do wyrażenia filtru



Rysunek 4.20. Filtrowanie za pomocą skrótu utworzonego na pasku narzędzi

Tego rodzaju filtry stanowią więc potężne narzędzie zwiększające możliwości w zakresie wyszukiwania problemów w pakietach przechwyconych w różnych scenariuszach.

Wireshark zawiera wiele wbudowanych filtrów, które są doskonałymi przykładami, pokazującymi, jak powinien wyglądać filtr. Te domyślne filtry i strony pomocy w dokumentacji narzędzia Wireshark będziesz prawdopodobnie wykorzystywać podczas tworzenia własnych filtrów. Filtrów będziemy używać w wielu przykładach przedstawionych w książce.

Skorowidz

E

F

G

H

I

J

K

R

S

Ś

T

U

W

Z

Ź

Ż

SPRAWDŹ, CO W PAKIECIE PISZCZY!

Przechwytywanie pakietów i badanie ich zawartości może kojarzyć się z szemraną działalnością domorosłych hakerów i włamywaczy. Okazuje się jednak, że analiza pakietów jest jednym ze skuteczniejszych narzędzi do rozwiązywania problemów z siecią. O ile samo przechwycenie pakietu, na przykład za pomocą popularnego narzędzia Wireshark, zasadniczo nie sprawia problemu, o tyle zbadanie zawartości tego pakietu i wykorzystanie zdobytej wiedzy do poprawy działania sieci bywa sporym wyzwaniem nawet dla doświadczonych administratorów.

Ta książka jest niezwykle praktycznym, przystępnie napisanym podręcznikiem, który znakomicie ułatwia zrozumienie tego, co się dzieje w sieci, i podjęcie adekwatnych działań w celu poprawy jej funkcjonowania. Niniejsze, trzecie wydanie zostało przejrzane i zaktualizowane, uwzględniono w nim również najnowszą wersję narzędzia Wireshark (2.x). Opisano sposób wykorzystywania przechwyconych danych do rozwiązywania problemów sieci. Gruntownie omówiono protokoły IPv6 i SMTP, znalazł się tu także nowy rozdział, opisujący narzędzia tshark i tcpdump. Działają one na poziomie powłoki i służą do analizy pakietów.

Najważniejsze zagadnienia ujęte w książce:

- badanie komunikacji sieciowej w czasie rzeczywistym
- wykrywanie przyczyn problemów z siecią
- wyodrębnianie plików z pakietów
- działanie złośliwego kodu na poziomie pakietów
- generowanie raportów i statystyk dotyczących ruchu sieciowego

Chris Sanders — jest ekspertem w dziedzinie bezpieczeństwa informacji. Pochodzi z Mayfield w stanie Kentucky. Pracował dla wielu agencji rządowych i wojskowych, a także dla kilku firm z listy Fortune 500. W ramach współpracy z Departamentem Obrony USA rozwijał modele dostawcy usług sieciowych oraz narzędzia wywiadowcze. Założył fundację, której celem jest rozwój zaawansowanych technologii informatycznych na obszarach wiejskich. W wolnych chwilach ogląda mecze koszykówki, grilluje i spędza czas na plaży. Wraz z żoną Ellen mieszka w Charleston w stanie Karolina Południowa.

Helion
księgarnia internetowa
http://helion.pl
zamówienia telefoniczne
0 801 339900
0 601 339900
Informatyka w najlepszym wydaniu

Helion SA
ul. Kościuszki 1c, 44-100 Gliwice
tel.: 32 230 98 63
e-mail: helion@helion.pl
http://helion.pl

Sprawdź najnowsze promocje:
http://helion.pl/promocje
Książki najchętniej czytane:
http://helion.pl/bestsellery
Zamów informacje o nowościach:
http://helion.pl/nowosci

ISBN 978-83-283-3696-4
9 788328 336964
cena: 69,00 zł

no starch press