Narzędzie Wireshark

Nie zawsze źle działająca sieć lokalna, łącze bezprzewodowe bądź sieć szkieletowa operatora muszą być przyczyną efektu spowolnienia dostępu do zasobów sieciowych. Bardzo często zdarza się, że wpływ na to mogą mieć aplikacje działające w tle zarówno na naszym komputerze, jak i na czyimś komputerze podłączonym do naszej sieci. Innym powodem może być atak na nasz komputer w celu wyłudzenia naszych danych (tzw. DNS spoof). Oczywiście może to być także po prostu zgubienie naszych pakietów (serwer docelowy jest wyłączony, nie odpowiada itp.).

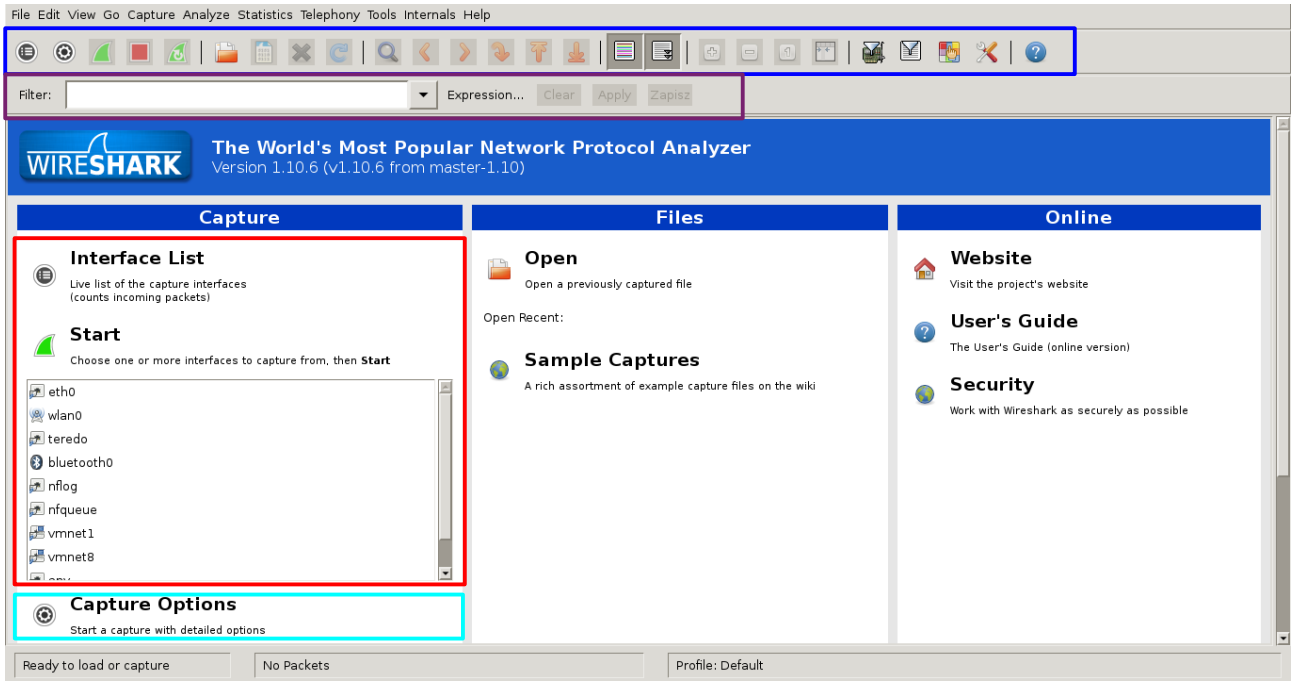
Jakiekolwiek by nasze intencje, JAKO ADMINISTRATORÓW SIECI, nie były, możemy przeanalizować działanie logicznej sieci lokalnej – kto, gdzie i co wysyła oraz co dobiera (i od kogo). Najpopularniejszym narzędziem do tego celu jest **Wireshark**, który bez najmniejszego problemu jest w stanie przechwycić WSZYSTKIE pakiety w obrębie danej sieci, posegregować je ze względu na pochodzenie, sprawdzić adres sprzętowy, z którego został wysłany pakiet itp. Wszystko, co potrzebne jest do podsłuchu to karta sieciowa (LAN/WLAN) działająca w trybie mieszanym (promiscous).

UWAGA! Narzędzia tego typu można wykorzystywać jedynie będąc uprawnioną osobą administrującą/konserwującą sieć lokalną/intranet. Jeżeli narzędzie będzie wykorzystywane przez użytkownika sieci to administrator w niektórych wypadkach może taką osobę odłączyć od sieci/pociągnąć ją do odpowiedzialności karnej! Program pozwala bowiem na przechwytywanie DANYCH TEKSTOWYCH obecnych w pakietach – pozwala on w tym kontekście na szpiegowanie działalności użytkowników (np. przechwycenie haseł, treści wiadomości itp.)

Poza Wireshark istnieją także inne programy do szpiegowania, jak chociażby Cain & Abel, Capsa Network Analyzer czy chociażby potężny Ettercap (stosowany również jako program do ataku typu człowiek w środku!). Dlaczego więc Wireshark? Ponieważ jest najprostszym z wyżej wymienionych, posiada rozbudowane opcje analizy sieci oraz działa niemal na wszystkich systemach operacyjnych (Cain and Abel działa jedynie na Windows, Capsa jest płatna a Ettercap najlepiej działa w systemach Unix/Linux).

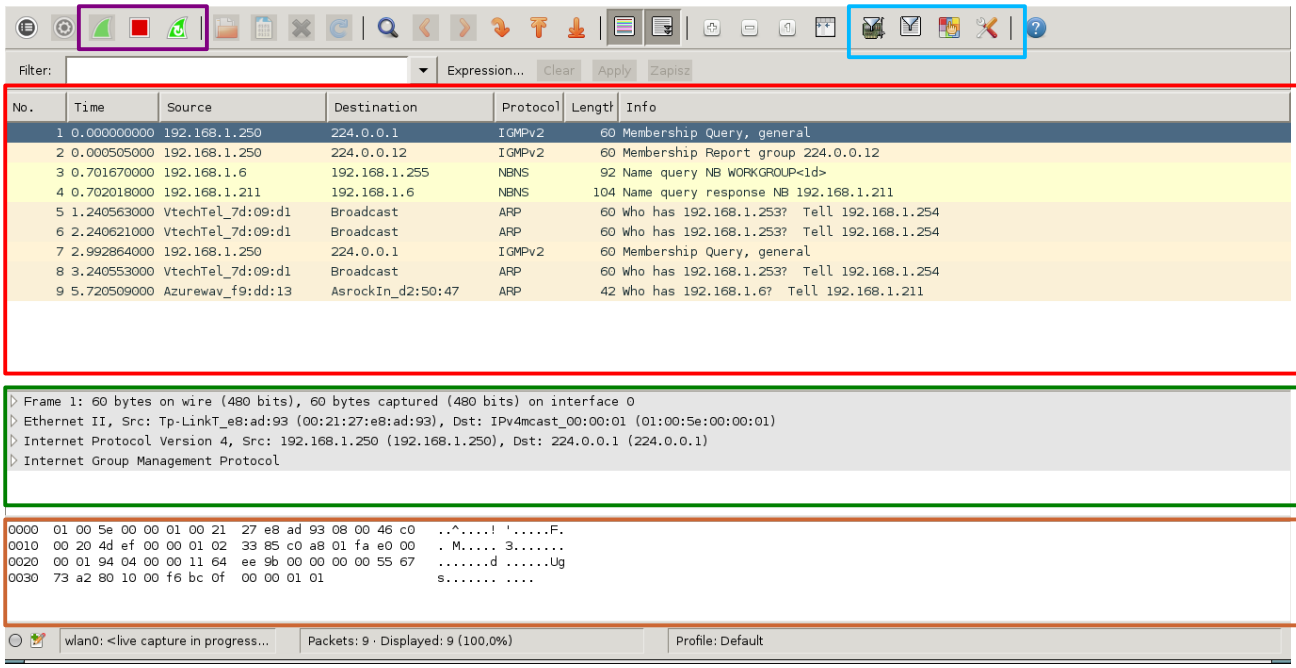
1. Podstawy użytkowania.

Program Wireshark ZAWSZE musi zostać uruchomiony z uprawnieniami administratora systemu. W innym wypadku nie będzie on miał możliwości przełączenia karty sieciowej w tryb mieszany, co z kolei skutecznie uniemożliwi podsłuch sieci – będzie możliwe jedynie podsłuchanie pakietów skierowanych do nas (a i tak mogą być z tym problemy – szczególnie z pakietami innymi niż TCP/UDP).
Po włączeniu programu będziemy mieli taki oto wygląd:



- NIEBIESKA ramka (pierwsza od góry) otacza pasek narzędziowy. Znajdują się na nim najważniejsze skróty do działań w programie, takie jak włączenie/wyłączenie rejestrowania (przechwytywania) pakietów, możliwość otworzenia wcześniej przechwyconej sesji (np. w celu przeanalizowania działania sieci), możliwość włączenie/wyłączenia kolorowania składni, włączenie/wyłączenie automatycznego przesuwania listy przechwyconych pakietów do najnowszego, edycja opcji programu, itd. WSZYSTKIE opcje posiadają swoją etykietę – wystarczy najechać na daną opcję kursorem i poczekać ok. 3 sekund, a program wyświetli nam informację na temat danej opcji jako podpowiedź
- FIOLETOWA ramka (druga od góry) to pasek wyrażeń filtrujących wyświetlanie pakietów; dzięki odpowiedniej składni (w dalszej części przykłady zastosowania). Program dysponuje także przykładowymi filtrami (wystarczy kliknąć przycisk **Filter:**); GDYBY zdarzyło się, że pasek ten zniknie to można go przywrócić poprzez menu View->Filter Toolbar
- CZERWONA ramka zawiera listę dostępnych w systemie interfejsów sieciowych (kablowe, bezprzewodowe, wirtualne mosty, karty maszyn wirtualnych itd.). Zanim zaczniemy nowe przechwytywanie trzeba wybrać jeden z tych interfejsów (kliknięcie lewym przyciskiem myszy, musi zostać podświetlony) po czym klikamy przycisk Start (zielona płetwa rekina)
- JASNONIEBIESKA ramka (ostatnia) pozwala na edycje opcji przechwytywania. Przykładowo możemy dodać nowy wirtualny interfejs (tutaj nazwany jako potok), który może zbierać dane np. z dwóch lub więcej interfejsów fizycznych bądź zbierać tylko określone dane na określonym interfejsie. Ponadto można wybrać opcje rozwiązywania nazw (domyślnie zaznaczone są zawsze dowiązania z adresem MAC oraz warstwy transportowej; można dodatkowo uruchomić rozwiązywanie nazw na poziomie warstwy sieciowej).

Po uruchomieniu nasłuchu zmieni się środek naszego okna. Od teraz będziemy mieli w nim dostępne informacje na temat pakietów wędrujących po wskazanej przez nas sieci (sieci, do której należy interfejs). Przykładowa zawartość okna analizy pakietów:

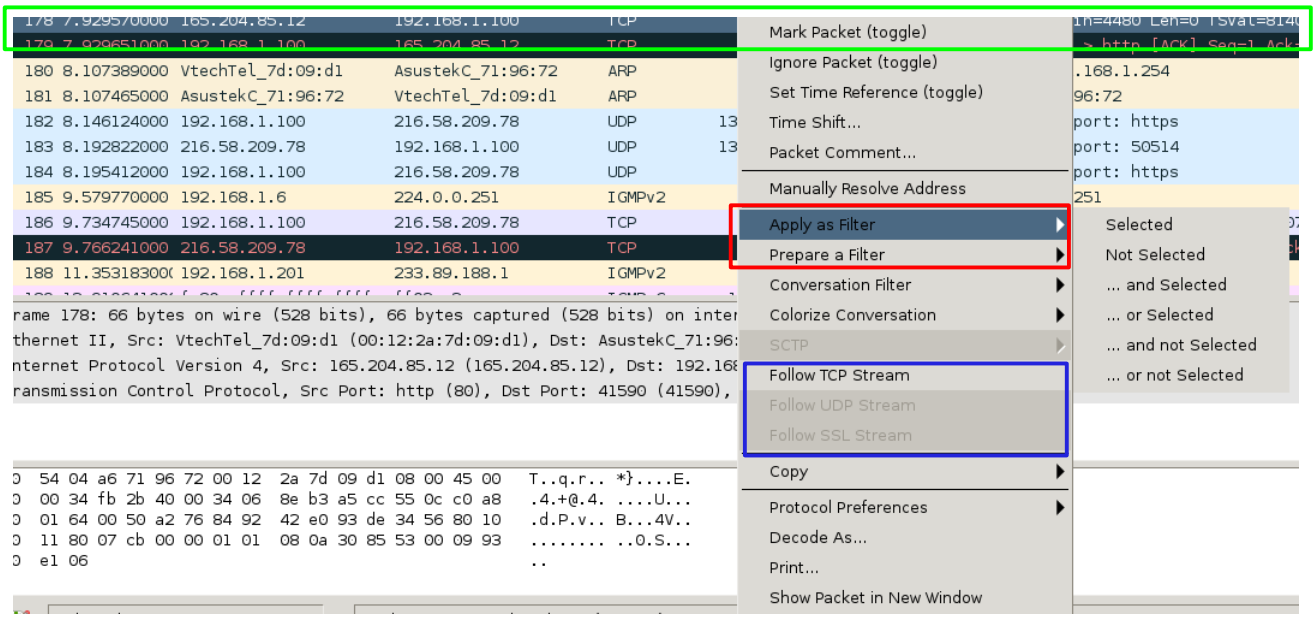


Czerwona ramka zawiera aktualnie przechwycone pakiety danych. Mamy liczbę przechwyconych pakietów, czas przechwycenia (liczony od rozpoczęcia przechwytywania), źródło pakietu (kto wysłał zapytanie – może być IP lub adres MAC), cel (do kogo jest adresowany), protokół (przeważnie nazwa, np. TCP), długość (ilość przesłanych bajtów w ramce) oraz informacje (co dany pakiet zrobił).
Zielona ramka zawiera informacje przenoszone w danym pakiecie – zarówno surowe nagłówki jak i treść właściwa (dane). Dzięki temu możemy dowiedzieć się jak wygląda struktura pakietu (czy nie został zdeformowany), czy jest poprawnie adresowany oraz czy jego treść nie jest uszkodzona.
Pomarańczowa ramka zawiera tylko zawartość pakietu (same dane) w formie szesnastkowej oraz tekstowej.

W fioletowej ramce znajdują się dosyć istotne przyciski odpowiadające (w kolejności):
- włączenie przechwytywania pakietów w trybie rzeczywistym
- zatrzymanie przechwytywania pakietów
- zatrzymanie przechwytywania, wyrzucenie dotychczas przechwyconych pakietów oraz ponowne uruchomienie przechwytywania

Jasnoniebieska ramka zawiera 4 dosyć użyteczne skróty, takie jak:
- edycja filtrów przechwytywania – mamy możliwość szybkiej edycji/dodawania filtrów pakietów (np. wyświetlanie tylko źródeł z adresami IP)
- edycja/zastosowanie filtrów – okno bardzo podobne do poprzedniego; różni się tym, iż można w nim także ustawić interesujący nas filtr
- zmień kolorowanie – pozwala na zmianę kolorów przypisanych dla poszczególnych pakietów (jeżeli nie pasują nam domyślne)
- edycja ustawień – można zmieniać ustawienia programu (jeżeli tego potrzebujemy)

Każdy pakiet, niezależnie czy klikniemy na nim prawym przyciskiem myszy w oknie listy pakietów (czerwona ramka), czy też w jego zawartość (zielona ramka), posiada szereg dodatkowych opcji, które możemy wykorzystać w celu np. prześledzenia jego wpływu na uzyskanie informacji jakiego rodzaju akcji dokonał:



Pakiet, którego dotyczy opis (dla którego zostało otwarte menu) to ten w ZIELONEJ ramce.

CZERWONA ramka skupia w sobie opcje pozwalające na stworzenie szybkiego filtrowania pakietów (zaznaczony, inne niż zaznaczony, i zaznaczony, lub zaznaczony, i nie zaznaczony, i lub niezaznaczony). Jako filtr może być wybrany np. adres źródłowy, adres docelowy czy protokół. Proszę pamiętać iż trzeba KLIKNĄĆ NA ODPOWIEDNIĄ KOLUMNĘ by stworzyć odpowiedni filtr. Opcje pozwalają na natychmiastowe wykorzystanie filtrowania bądź na utworzenie jego szablonu (nie będzie zastosowany od razu).

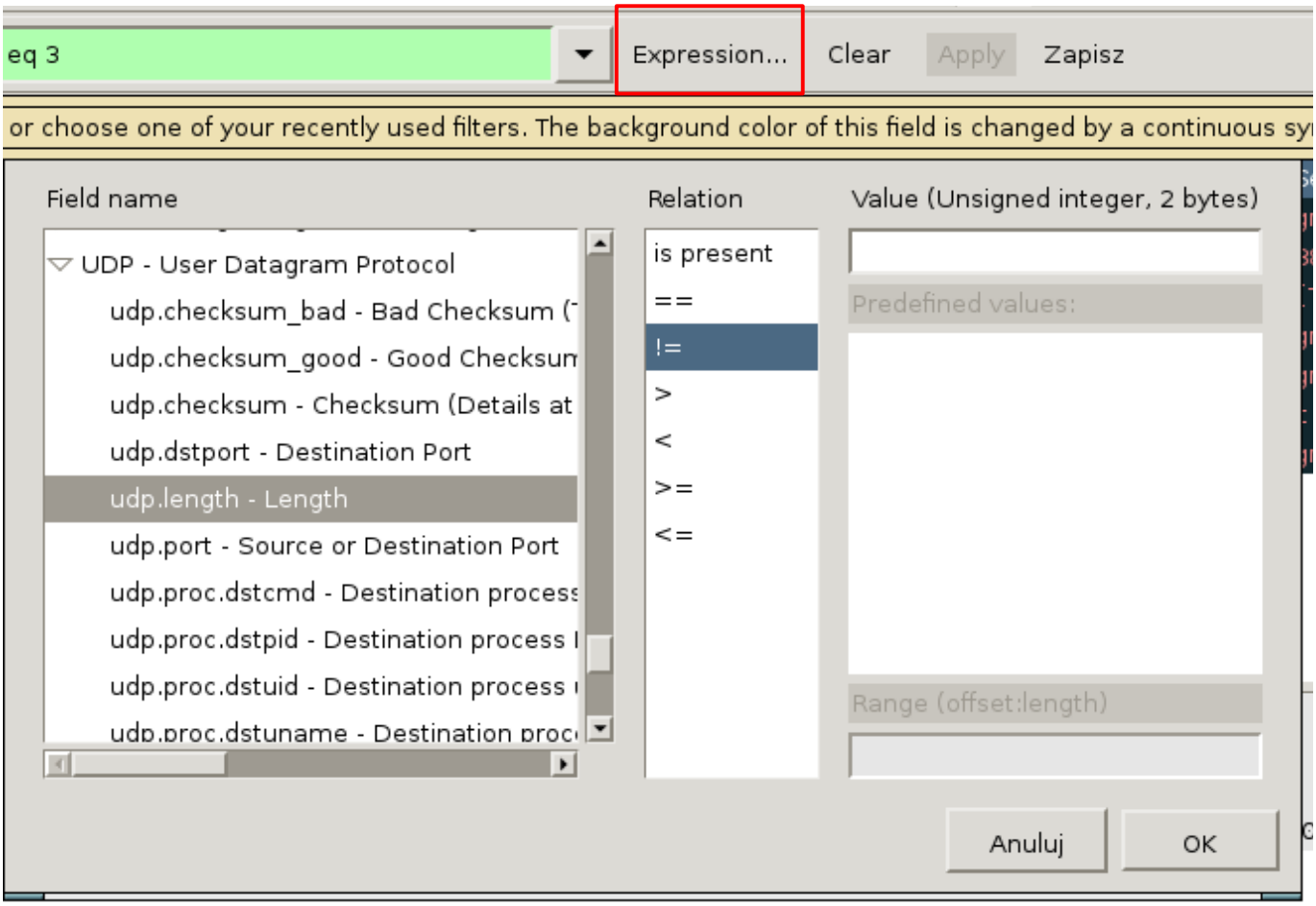
NIEBIESKA ramka pozwala na podążanie opcję podążania za wskazanym pakietem. Każdy pakiet posiada w systemie swój określony strumień, przez który wymieniane są kolejne pakiety. Strumień przeważnie istnieje przy danej transmisji, po czym zostaje zamknięty. Wireshark może dzięki temu wyświetlić tylko pakiety należące do danego strumienia. Mało z tego, pozwala także na złożenie pełnego dialogu pomiędzy klientem a serwerem, wyświetlając zarówno nagłówki danego protokołu jak i zawartość poszczególnych pakietów (oraz dodatkowe informacje, jeżeli takowe istnieją).

Klikając dwukrotnie lewym przyciskiem myszy na danym pakiecie z listy Wireshark otwiera nowe okno, w którym swobodnie możemy przejrzeć jego szczegółową zawartość.

Po prześledzeniu pakietów możemy zapisać wszystkie zebrane informacje do pliku (File->Save). Pozwala to na późniejsze otworzenie takiej sesji podsłuchiwania celem późniejszej analizy działania sieci.

2. Tworzenie własnych filtrów/wyrażeń filtrujących

W przypadku kiedy zaczniemy zbierać bardzo dużą ilość pakietów przejrzenie efektu podsłuchu stałoby się bardzo żmudnym zajęciem. Dlatego w program został wbudowany bardzo elastyczny i zwinny sposób na odfiltrowywanie interesujących nas w danej chwili danych. Pakiety można filtrować w zasadzie w dowolny sposób: można wybierać tylko określone adresy IP, określone adresy domenowe, adresy MAC, protokoły itp. Chociaż do codziennej pracy wykorzystuje się przeważnie ok. 10 łączących się ze sobą wyrażeń, program oferuje ich znacznie więcej. W celu przejrzenia biblioteki wyrażeń można przywołać okno edytora wyrażeń filtrujących:



Okno wyrażeń otwiera się poprzez kliknięcie zaznaczonego w czerwonej ramce przycisku. W kolumnie Field name możemy wybrać dowolną właściwość, po której nastąpi filtracja naszych pakietów. Na zrzucie powyżej otwarta została gałąź protokołu UDP, a wybrana została jego właściwość mówiąca o wielkości pakietu (ile bajtów zawiera jego pole danych).
Druga kolumna pozwala na wybranie relacji (porównania) wartości wskazanej właściwości względem wartości, jaką podaje się w trzeciej kolumnie. Jeżeli w tej chwili w trzeciej kolumnie (Value) wpisana zostałaby wartość 80 to ukryte zostałyby wyświetlone wszystkie pakiety UDP poza tymi, których wielkość równa jest 80.
Poniższa tabela tłumaczy poszczególne oznaczenia relacji (druga kolumna na zrzucie):

Skrót literowy (można stosować!)	Oznaczenie symboliczne	Przykład użycia
eq	==	Równy, np. ip.addr == 192.168.1.1
ne	!=	Nie równy, np. ip.dst != 192.168.1.1
gt	>	Większy, np. tcp.length > 80 (czyli 81 i więcej)
lt	<	Mniejszy, np. frame.len < 120 (czyli 119 i mniej)
ge	>=	Większy bądź równy, np. tcp.length > 80
le	<=	Mniejszy, np. frame.len < 120

Prócz prostych filtrów można także tworzyć bardziej złożone, łączone. Tabela poniżej pokazuje spoiwa takich wyrażeń wraz z przykładami:

Skrót literowy	Oznaczenie	Przykład
and	&&	Mnożenie logiczne, np. ip.addr == 192.168.1.1 && tcp

or	||	Suma logiczna, np. tcp || udp
xor	^^	Logiczne lub, np. ip.src == 192.168.1.1 ^^ ip.dst == 192.168.1.1
not	!	Logiczne zaprzeczenie, np. !(dns)

Wireshark pozwala ponadto na tworzenie pseudo wyrażeń regularnych. Pozwalają one np. na podanie tylko fragmentu adresu MAC źródła/celu. Najlepszym sposobem będzie przedstawić na przykładach:

a) eth.src[2] == 78

oznacza to, że filtr weźmie pod uwagę WSZYSTKIE adresy MAC posiadające na drugiej pozycji 78, czyli np. 00:78:AD:7F:AA:12, 00:78:45:7F:00:12, EA:78:AD:7F:AA:45

b) eth.src[0:2] == EF:2B

oznacza, że filtr weźmie pod uwagę adresy MAC, które będą posiadały z przodu wskazane subsekcje

c) eth.src[4:] == FG:AA

oznaczam, że wzięte pod uwagę zostaną tylko elementy adresu znajdujące się po 4 elemencie. Ponieważ adres MAC posiada 6 elementów (sekcji) toteż w powyższym przykładzie można było podać jedynie 2 sekcje.

d) eth.src[:4] == 13:00:00:FF

odwrotna sytuacja do poprzedniej, 2 z 6 sekcji zostają pominięte (należy więc podać 4)

e) eth.src[1-3] == EF:2B:45

określa ile sekcji ma zostać podanych (inne rozwiązanie do przykładu b)

f) eth.src[1-3,6] == EF:2B:45:FF

połączenie kilku metod; podajemy pierwsze 3 sekcję oraz ostatnią. Od tego momentu brane będą pod uwagę adresy MAC EF:2B:45:*:*:FF, gdzie * oznacza dowolny zakres pól

INFORMACJA: Jeżeli chcemy wybrać w protokole np. WSZYSTKIE adresy IP POZA WSKAZANYM to nie powinniśmy tworzyć filtru

ip.addr != 192.168.1.1

Zamiast tego należy utworzyć filtr:

!(ip.addr == 192.168.1.1)

Dzięki czemu zostaną wybrane wszystkie pakiety poza tymi, w których znajdzie się wskazany adres.

ZADANIA:
1. Jak realnie podsłuchać przy pomocy Wireshark wszystkie pakiety w danej sieci (niezależnie od ilości przełączników oraz punktów dostępowych).
2. Proszę podsłuchać protokół HTTP. Czy dane przesyłane ze stron internetowych mogą być odczytane?
3. Proszę sprawdzić jak wygląda sprawa z autoryzacją poprzez HTTP Basic Authorization (czyli strona z hasłem po stronie serwera WWW).
4. Jak wygląda sprawa z programem pocztowym wysyłającym/odbierającym pocztę? Czy możliwe jest podejrzenie hasła do poczty?
5. Jak wygląda przesyłanie danych z formularzy ze stron WWW? Czy można podejrzeć ich zawartość i, co gorsza przesyłane w ten sposób hasła?
6. Jak wygląda sprawa z serwerami FTP? Czy możliwe jest uzyskanie loginu i hasła do wskazanego FTP?
7. Czy można podsłuchać transmisję SSL?

ZADANIA WYKONUJEMY SAMODZIELNIE! W razie potrzeby instalujemy odpowiednie usługi/zakładamy konta w darmowych usługach! NIE KOPIUJEMY NICZEGO Z INTERNETU!

ŹRÓDŁA:
http://www.howtogeek.com/104278/how-to-use-wireshark-to-capture-filter-and-inspect-packets/
http://cs.gmu.edu/~astavrou/courses/ISA_674_F12/Wireshark-Tutorial.pdf
https://www.wireshark.org/docs/wsug_html_chunked/ChWorkBuildDisplayFilterSection.html
http://wiki.wireshark.org/CaptureSetup/Ethernet
http://bzyczek.kis.p.lodz.pl/pliki/pask/pask_lab_02_wireshark.pdf