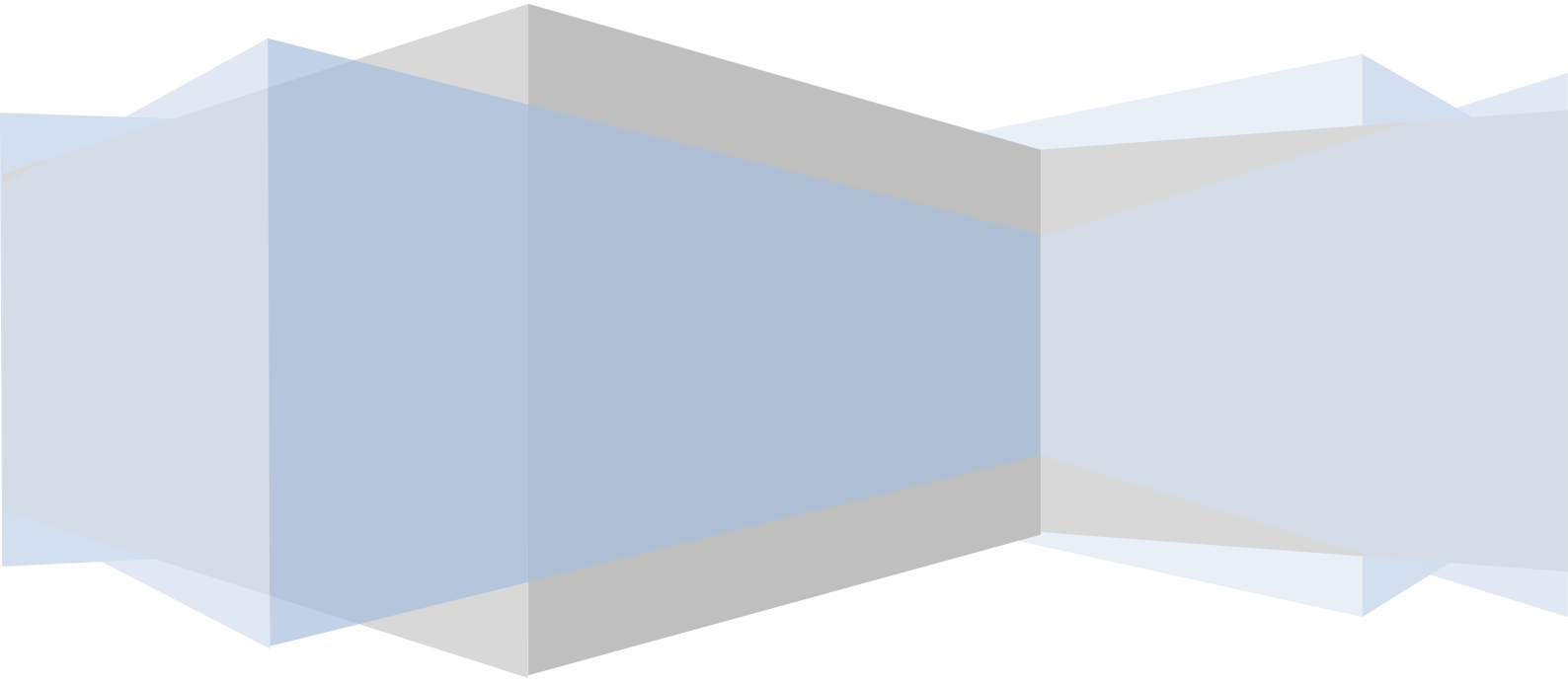
Katedra Inżynierii Komputerowej
Politechnika Częstochowska

Filtry i statystyki w analizatorach protokołów

Laboratorium Podstaw sieci komputerowych



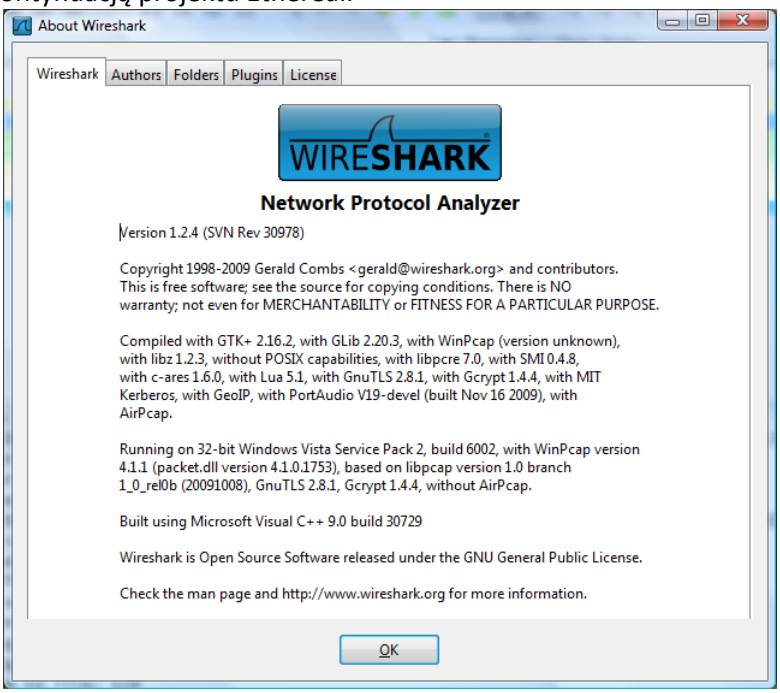
Cel ćwiczenia

Celem ćwiczenia jest zapoznanie się z wybranymi funkcjami analizatorów protokołów.

Informacje wstępne

Analizator Wireshark

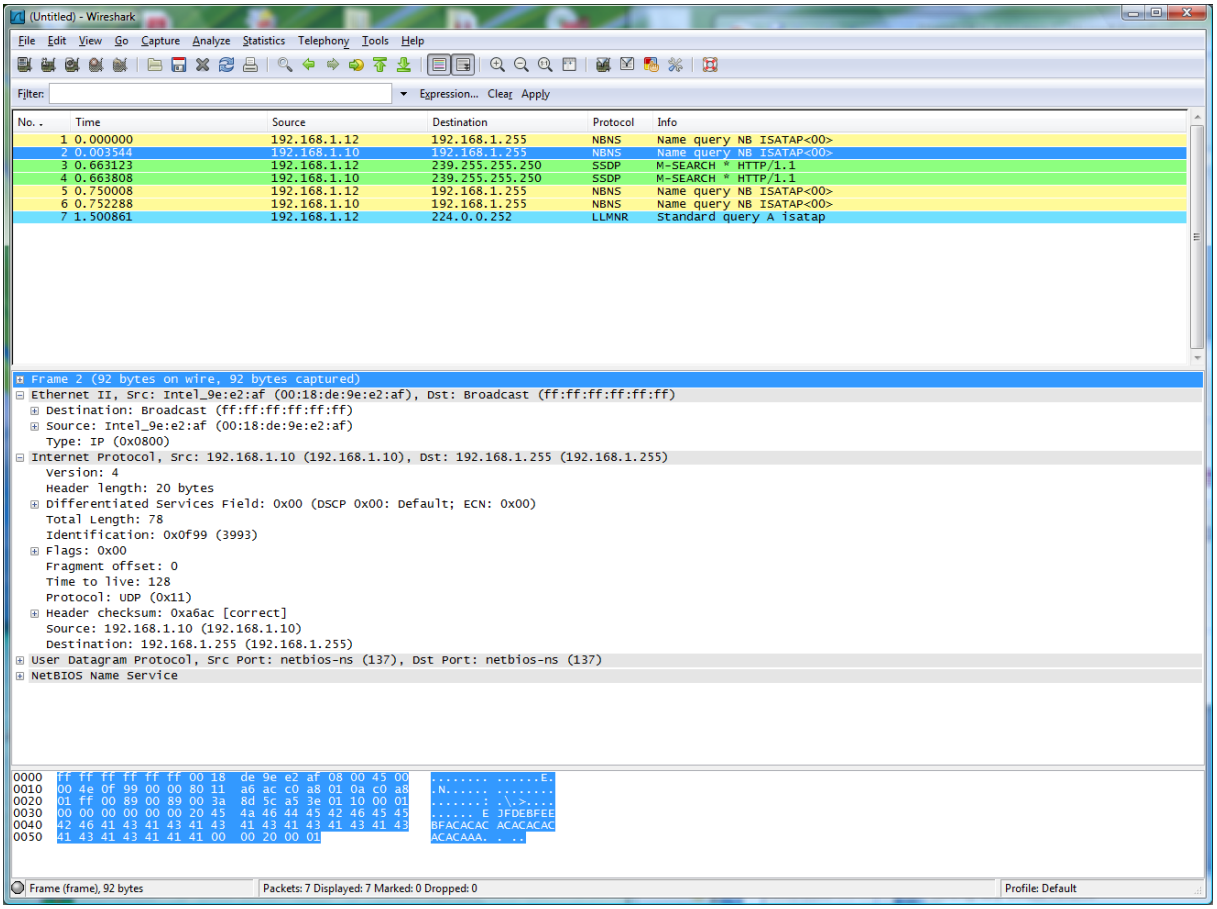
Analizator Wireshark jest programowym analizatorem protokołów. Dzięki olbrzymim i stale rosnącym możliwością i dostępności na licencji GNU jest najpopularniejszym tego typu programem na świecie. Jest on kontynuacją projektu Ethereal.



Okno programu składa się z kilku elementów ulokowanych pod sobą. Są to:

- menu,
- pasek narzędzi,
- pole filtru wyświetlania,
- lista przechwyconych ramek,
- szczegóły wskazanej ramki,
- treść bajtów ramki,
- pasek stanu.

Widoczność wymienionych elementów można włączać i wyłączać.



Przechwytywanie rozpoczynamy np. wywołując listę interfejsów, po czym, przy wybranym, naciskamy klawisz **Start**, bądź **Options** i **Start** w kolejnym oknie dialogowym. Ta druga opcja pozwala na ustawienie szczegółowych opcji i filtru przechwytywania.

Bardzo ważnym elementem pracy z analizatorem protokołu jest korzystanie z filtrów przechwytywania i wyświetlania. W programie Wireshark filtry tworzy się w oparciu o poniższe zasady.

Filtry

Filtry – operatory relacji

eq, ==	równe
ne, !=	różne
gt, >	większe niż
lt, <	mniejsze niż
ge, >=	większe niż lub równe
le, <=	mniejsze niż lub równe

Filtry – operatory logiczne

and, &&	Logical AND
or, ||	Logical OR
not, !	Logical NOT

Filtry - protokoły

arp, dns, tcp, udp, ip, ipv6, irc, idp, ipx, http, pop, smtp, ftp, gnutella, image-jfif, kerberos, l2tp, netlogon, smb...

Filtry – pola protokołów (eth)

eth.addr ==
eth.dst ==
eth.len ==
eth.src ==
eth.trailer ==
eth.type ==

Filtry – pola protokołów (IP)

ip.dst eq www.mit.edu
ip.src == 192.168.1.1
ip.addr == 129.111.0.0/16
ip.fragment ==
ip.id ==
ip.len ==
ip.ttl ==

Filtry – pola protokołów (TCP)

tcp.port == 80
tcp.dstport ==
tcp.srcport ==
tcp.ack == numer potwierdzenia
tcp.flags == flaga 8-bit
tcp.flags.reset {ack, syn, fin, }
tcp.len == ???
tcp.window_size ==

Filtry – pola protokołów (UDP)

udp.checksum
udp.checksum_bad
udp.dstport
udp.length
udp.port
udp.srcport

Filtry – pola protokołów (HTTP)

http.cookie ==
http.host ==

Filtry – pola protokołów (echo)

echo.data ==
echo.request
echo.response

Przykłady:

Ruch dla usługi telnet konkretnego hosta
tcp.port==23 and host==10.0.0.5

Ruch dla usługi telnet bez konkretnego hosta
tcp.port==23 and not host==10.0.0.5

Przebieg ćwiczeń

Przechwycić możliwie największą ilość ramek z sieci ethernet. W tym celu należy uruchomić przechwytywanie ruchu a następnie uruchomić różne programy korzystające z zasobów sieciowych i programów diagnostycznych, np. przeglądarkę internetową, klienta poczty e-mail, ftp, ping, nslookup, net, przeszukiwanie zasobów sieciowych explorerem windows.

Wśród przechwyconych pakietów należy korzystając z filtru wyświetlania wybrać te, które

- kierowane są tylko do naszego komputera,
- wysyłane są z naszego komputera,
- wykorzystują protokół arp,
- wykorzystują protokół icmp,
- wykorzystują protokół ftp,
- wykorzystują protokół pop,
- wykorzystują protokół smtp,
- wykorzystują protokół arp i zostały wysłane z naszego komputera,
- inne zaproponowane przez prowadzącego.

Wykreślić statystyki ruchu w sieci (domenie kolizyjnej) z podziałem na wybrane protokoły (http, udp, tcp, smb, icmp, arp itp.).

Sprawozdanie

Studenci pracują w parach. Każda para dokonuje badanie dostarczonych przez prowadzącego przewodów i przygotowuje protokół zawierający wyniki pomiarów. Dla każdego przewodu należy podać jego długość, rodzaj wady oraz wskazanie której lub których par i wtyczek wada dotyczy. Protokół stanowi podstawę do wykonania sprawozdania.