Mastering Filtering

SYN-bit
deep traffic analysis

Sake Blok
Relational therapist for computer systems
sake.blok@SYN-bit.nl

Agenda

- Using the filter bar
- Right-click / Drag & drop filtering
- Filtering from dialogs
- Display filter buttons
- Multiple occurrences of fields
- Multiple protocol layers
- Further reading / viewing
- Q&A

About me…

EURONET INTERNET

ABN·AMRO

ion ip

SYN-bit
deep traffic analysis


SYN-bit
deep traffic analysis
Application and network troubleshooting
Protocol and packet analysis
Training (Wireshark, TCP, SSL)
www.SYN-bit.nl

Using the filter bar

- Filter on...
  - protocol presence
  - field presence
- Using basic operators
  - == != < > <= >=
- Set filters
  - tcp.port in {80, 8000..8099}
- String filter
  - http.cookie contains "PHPSESSION"
- Combine with and, or, not, (...)

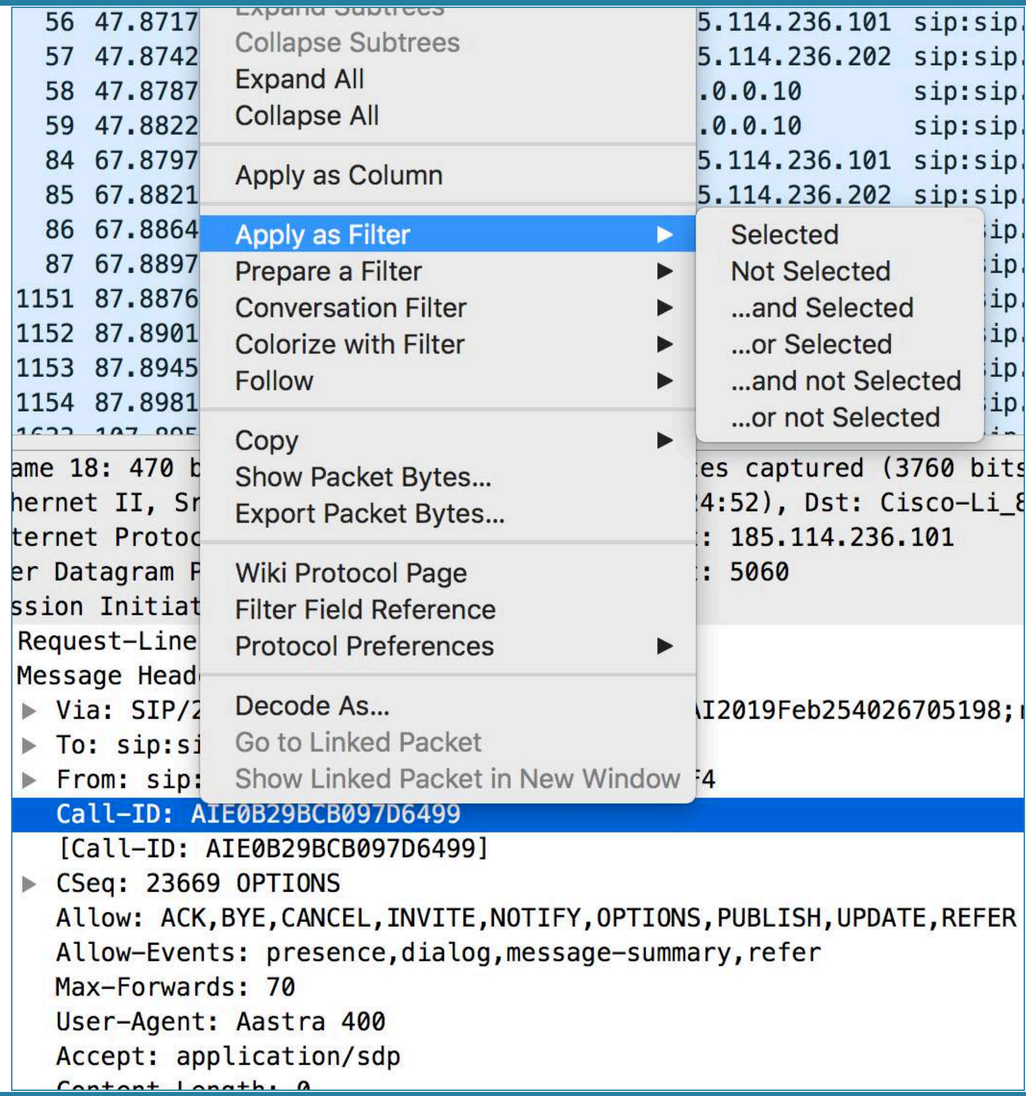
DEMO 1

Filter operators

C-like	English	Description	Example
==	eq, ***any_eq***	Equals	ip.addr == 192.168.1.1
!=	ne, ***all_ne***	(All) Not equals	ip.addr != 192.168.1.1
===	***all_eq***	***All equals***	***ip.addr === 192.168.1.1***
!==	***any_ne***	***Any not equals***	***ip.addr !== 192.168.1.0/24***
>	gt	Greater than	dns.time > 0.1
<	lt	Less than	frame.len < 60
>=	ge	Greater than or equal to	frame.number >= 1000
<=	le	Less than or equal to	frame.number <= 2000
	contains	Protocol, field or slice contains	http.host contains "amazon"
~	matches	Matches regular expression	http.host matches "^www\\..*\\.nl$"
&	bitwise_and	Bitwise and on the field	tcp.flags&7, ***tcp.flags&18==2***
	in	Set membership	tcp.port in {443 80..89} ***tcp.port in {443, 80..89}***
!	not	Logical NOT	not ip.addr == 192.168.1.1
||	or	Logical OR	udp.port==53 or tcp.port==53
&&	and	Logical AND	ip.addr == 10.0.0.1 and udp.port==5060
( )		Group expressions together	ip.addr == 10.0.0.1 and (icmp or dns)

Right-click / Drag & drop filtering

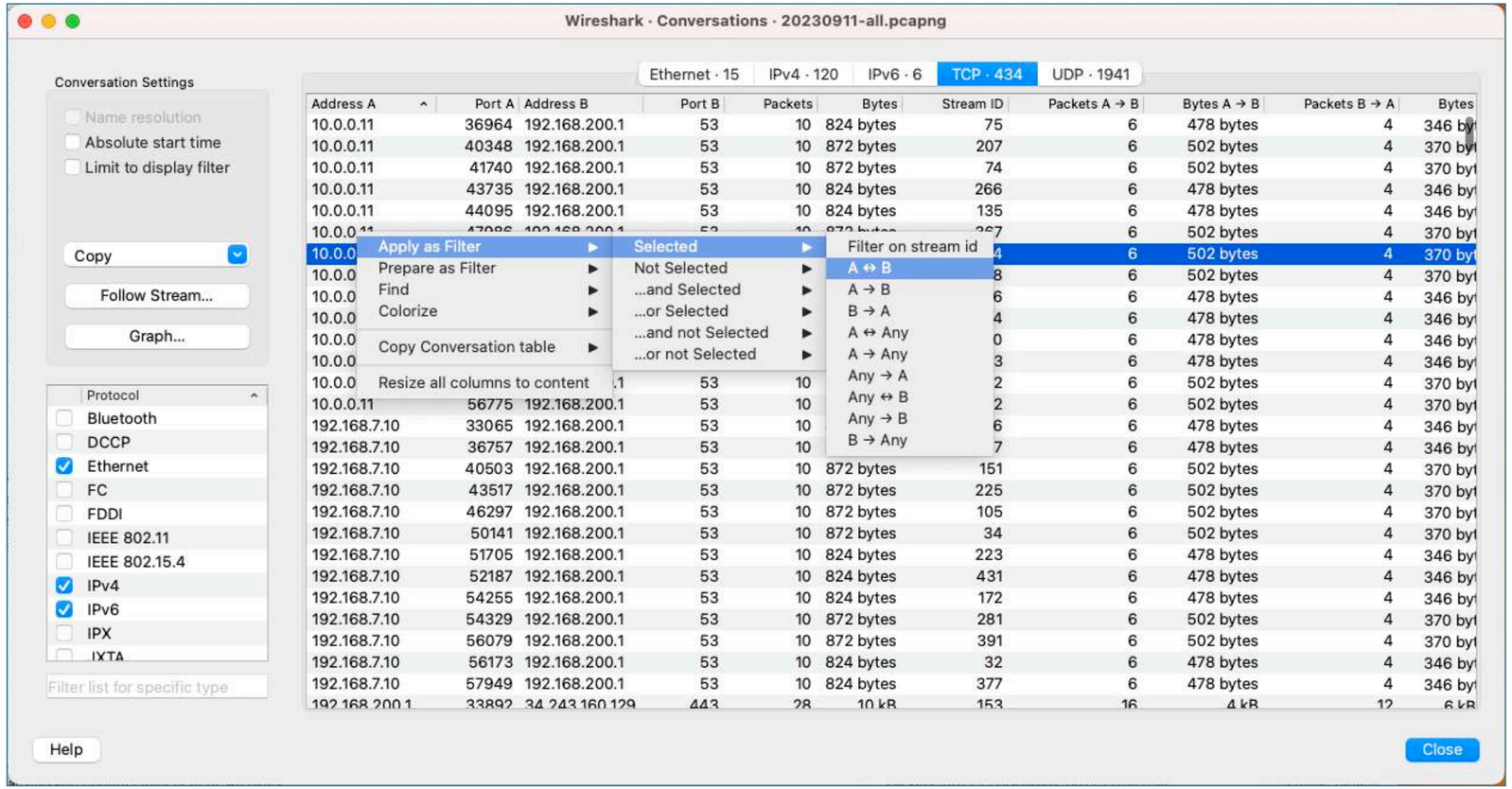
- Right-click on any item in...
  - Packet list
  - Packet details
- Or drag & drop any item to the filter bar
- Replace filter
  - ...or extend it
- Apply filter (default)
  - ... or prepare (hold shift before drop)



DEMO 2

Filtering from dialogs

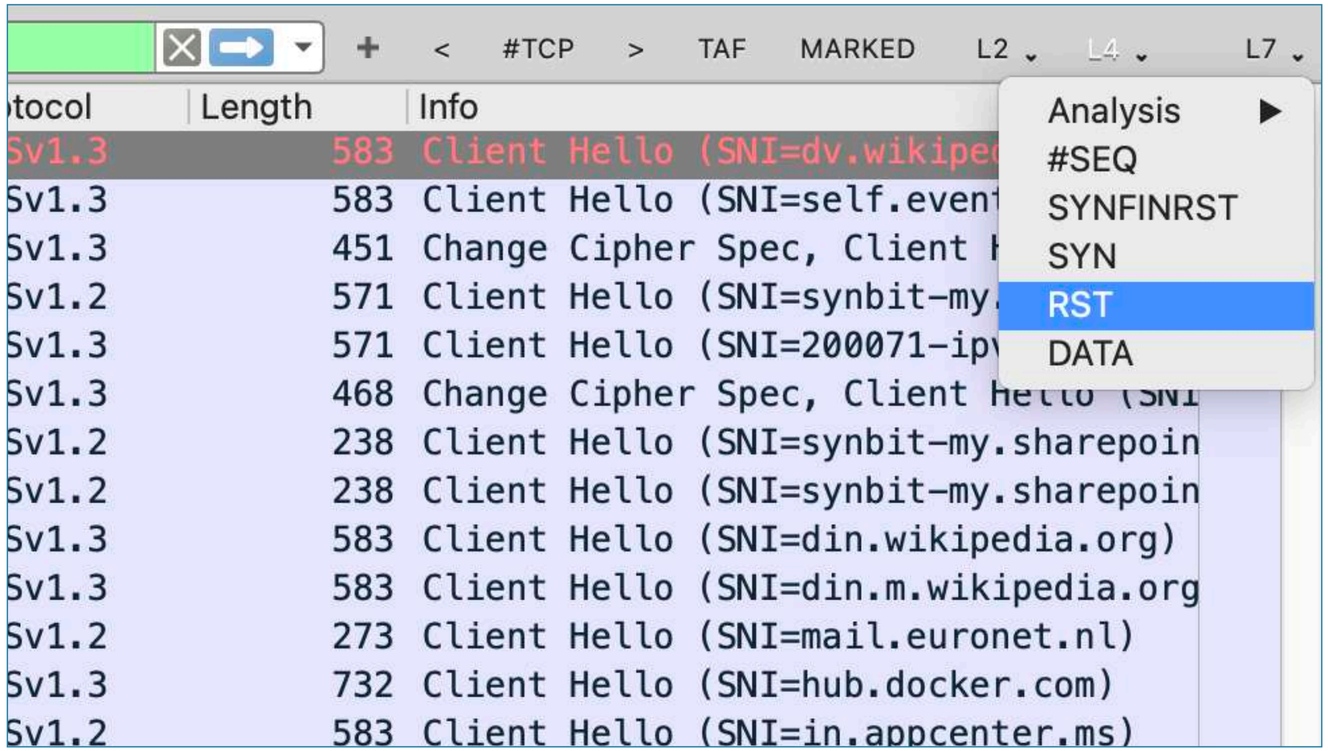
- Rightclick on items in...
  - Endpoints
  - Conversations
  - Protocol hierarchy
  - etc.



DEMO 3

Display filter buttons

- Handy 1 click filter actions
  - use + next to filter bar to create
- ...or more with dropdowns
  - use // in the filter button name
- Field references are powerful
  - tcp.stream == ${tcp.stream}
  - ip.addr == ${dns.a}



DEMO 4

Multiple occurences of fields

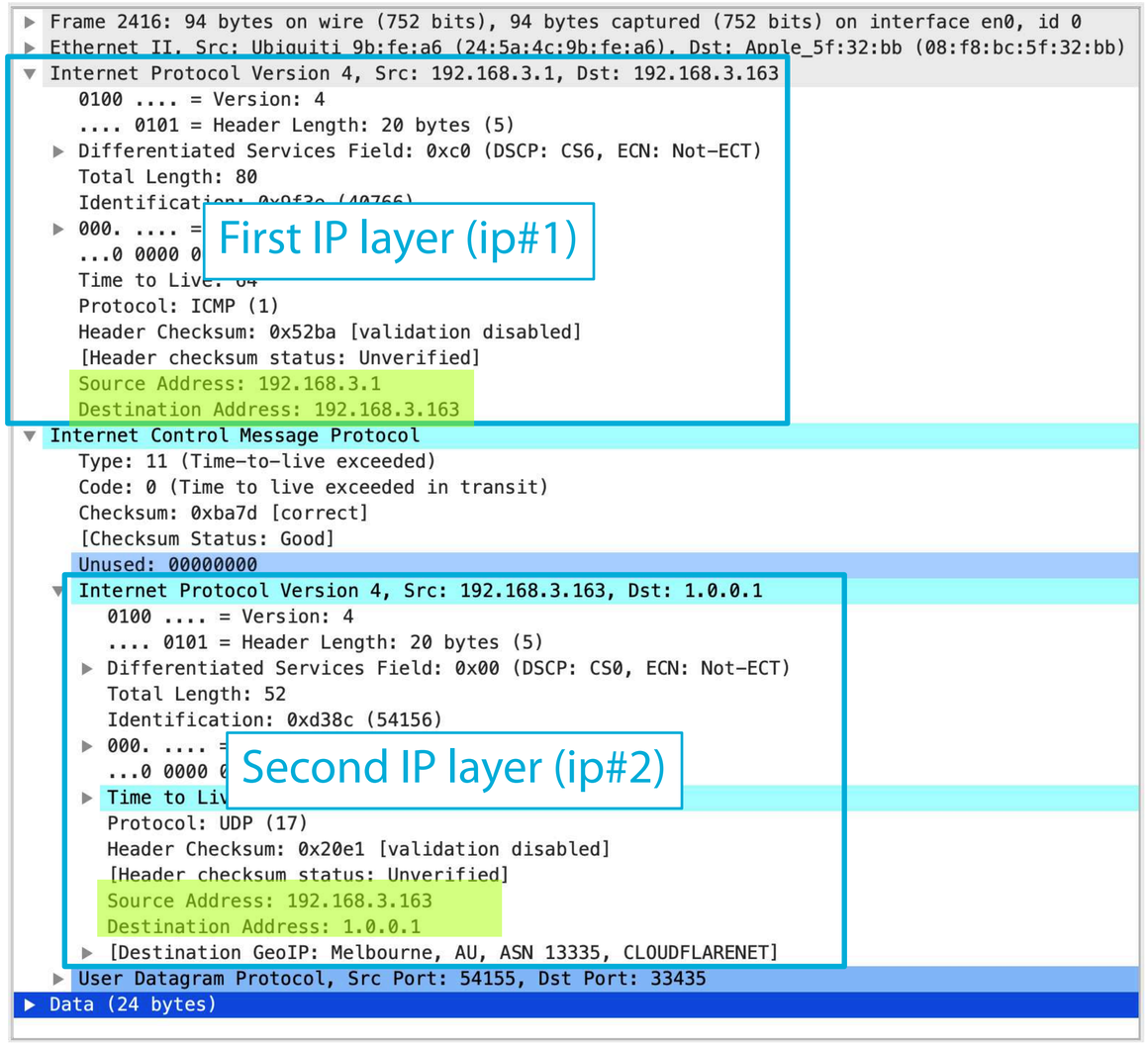
- Implied **any** on *all* operators
  - ip.addr == 192.168.1.1
- ...except for implied **all** for !=
  - ip.addr != 192.168.1.1
- Implied **any** on !==
  - ip.dsfield.dscp !== 0
- Overrule with explicit **all** or **any**
  - all tcp.port >= 1024
  - all udp.port == 53

DEMO 5

Multiple protocol layers

- Specify individual protocol layers with #n
  - ip.addr#2 == 192.168.1.1
  - vlan.id#2 == 10
  - vlan.id#-1 == 10

Filtering on specific protocol layers

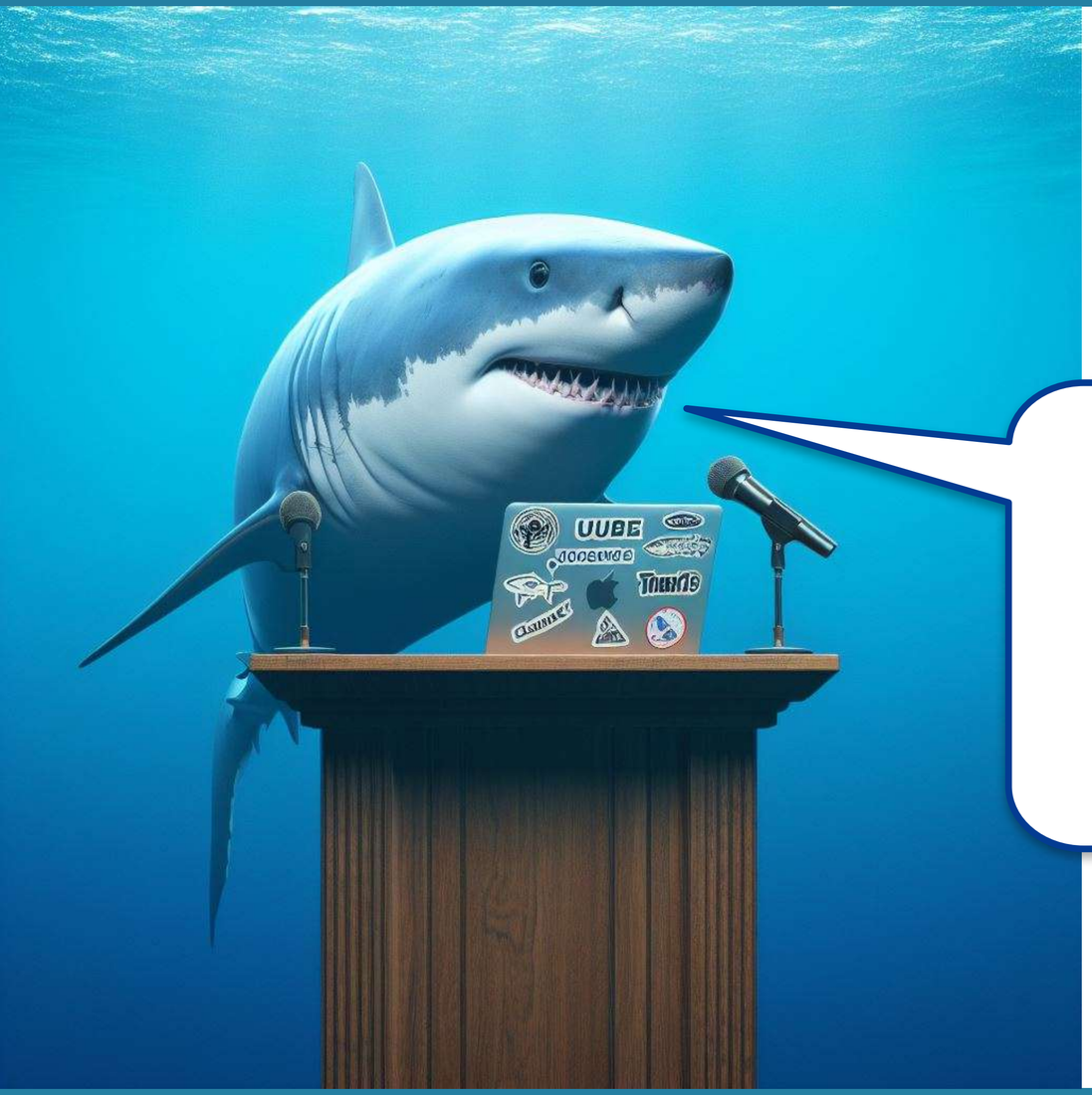


Field	192.168.3.1	192.168.3.163	1.0.0.1
ip.addr	✔	✔	✔
ip.addr#1	✔	✔	
ip.addr#2		✔	✔
ip.src	✔	✔	
ip.src#1	✔		
ip.src#2		✔	
ip.dst		✔	✔
ip.dst#1		✔	
ip.dst#2			✔

DEMO 6

Further reading / viewing

- Wireshark Users Guide
  - https://www.wireshark.org/docs/wsug_html_chunked/ChWorkDisplayFilterSection.html
- Wireshark Wiki
  - https://wiki.wireshark.org/DisplayFilters
- Wireshark display filter reference
  - https://www.wireshark.org/docs/dfref/
- Sharkfest retrospective pages
  - https://sharkfest.wireshark.org/retrospective/



Time for Q & A

FIN/ACK/FIN/ACK

Still questions?

sake.blok@SYN-bit.nl