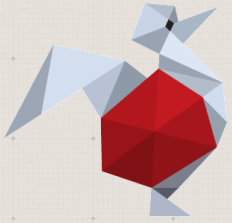
# Praktyczny Wireshark

tomasz.turba@securitum.pl

tturba

tturba



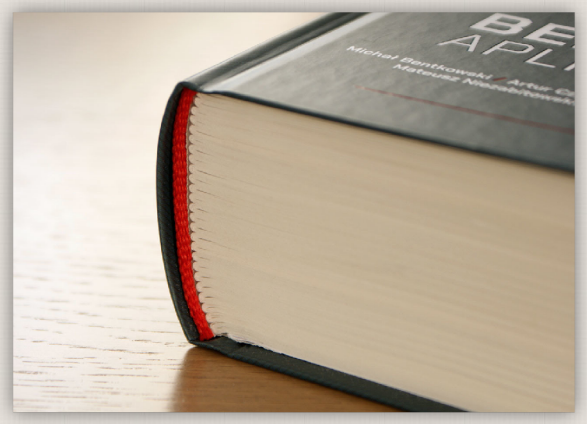
sekurak.pl

szkolenia

testy/audyty bezpieczeństwa

sekurak.pl

securitum



ksiazka.sekurak.pl

mega
sekurak hacking party

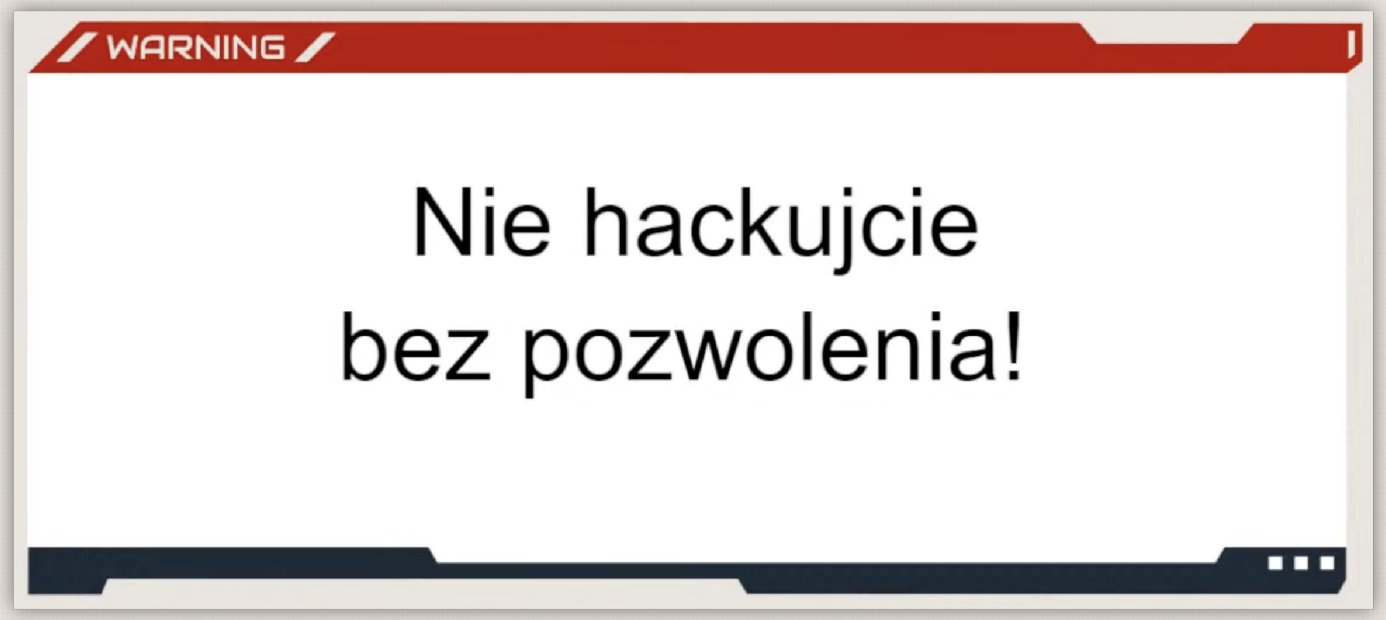
Tomasz Turba

CISS, CCNP, CCSP, RHCE, MSDST, ABW, ISO27001

tturba

tturba

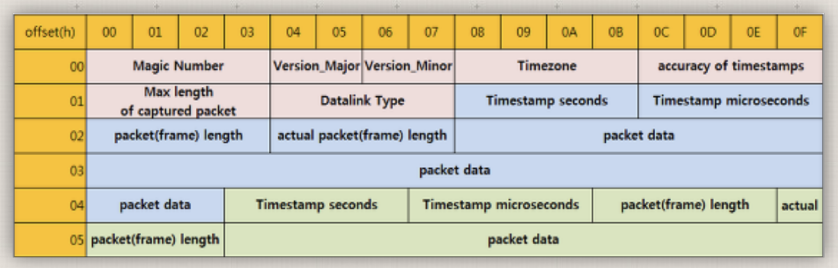
0111001100001

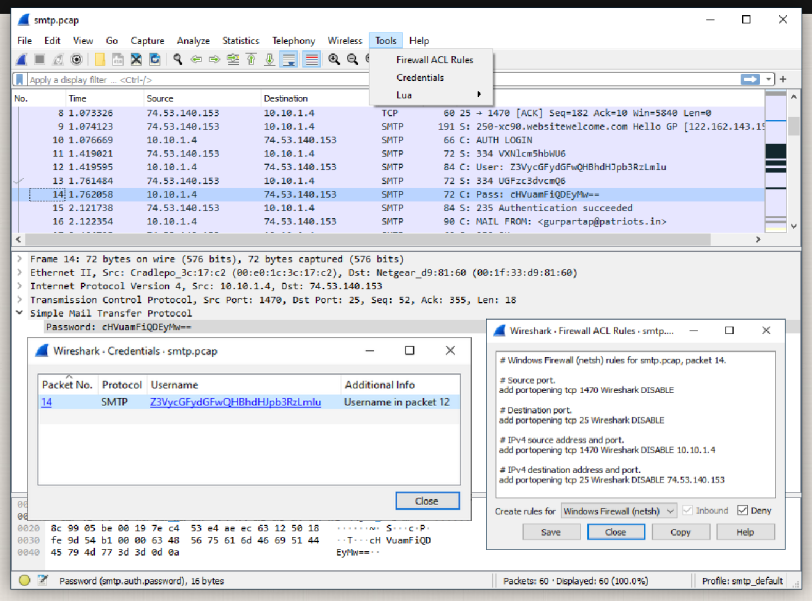


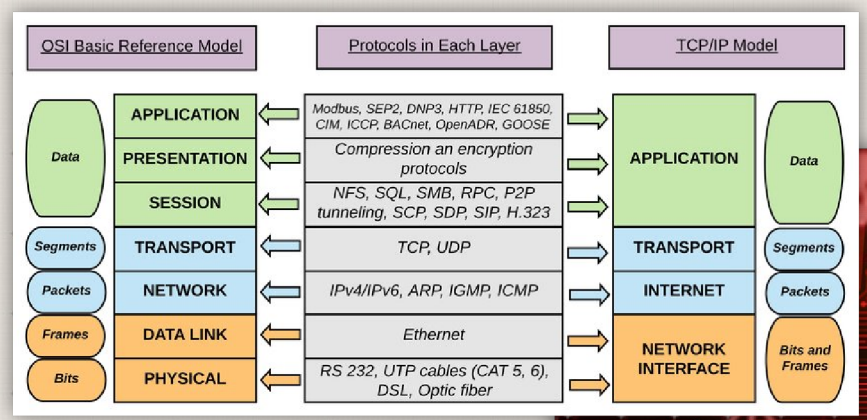

Prezentacja oraz szkolenie wyłącznie w celach edukacyjnych

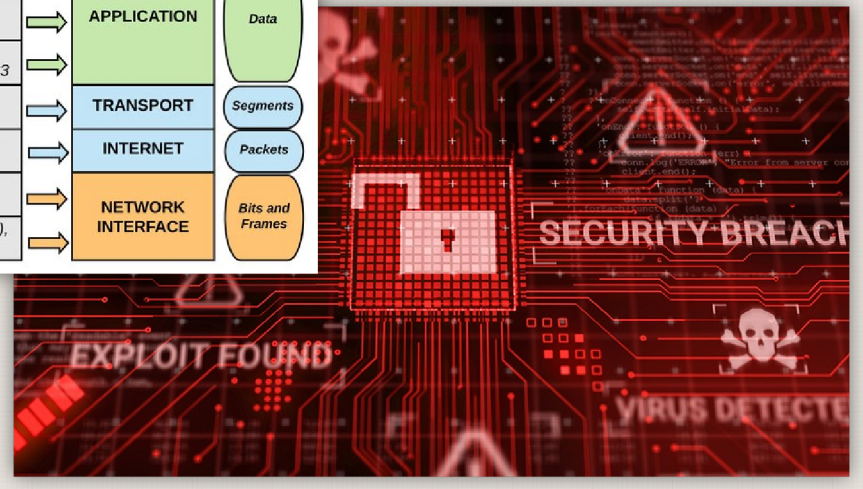
# Agenda

1. Wireshark
2. Komunikacja sieciowa
3. PCAP
4. Rodzaje filtrów
5. Ustawienia programu
6. Rekonesans malware
7. Rekonesans stealera + tls fingerprint
8. SSL decipher
9. ARP poison (MITM) discovery



# Analiza ruchu sieciowego

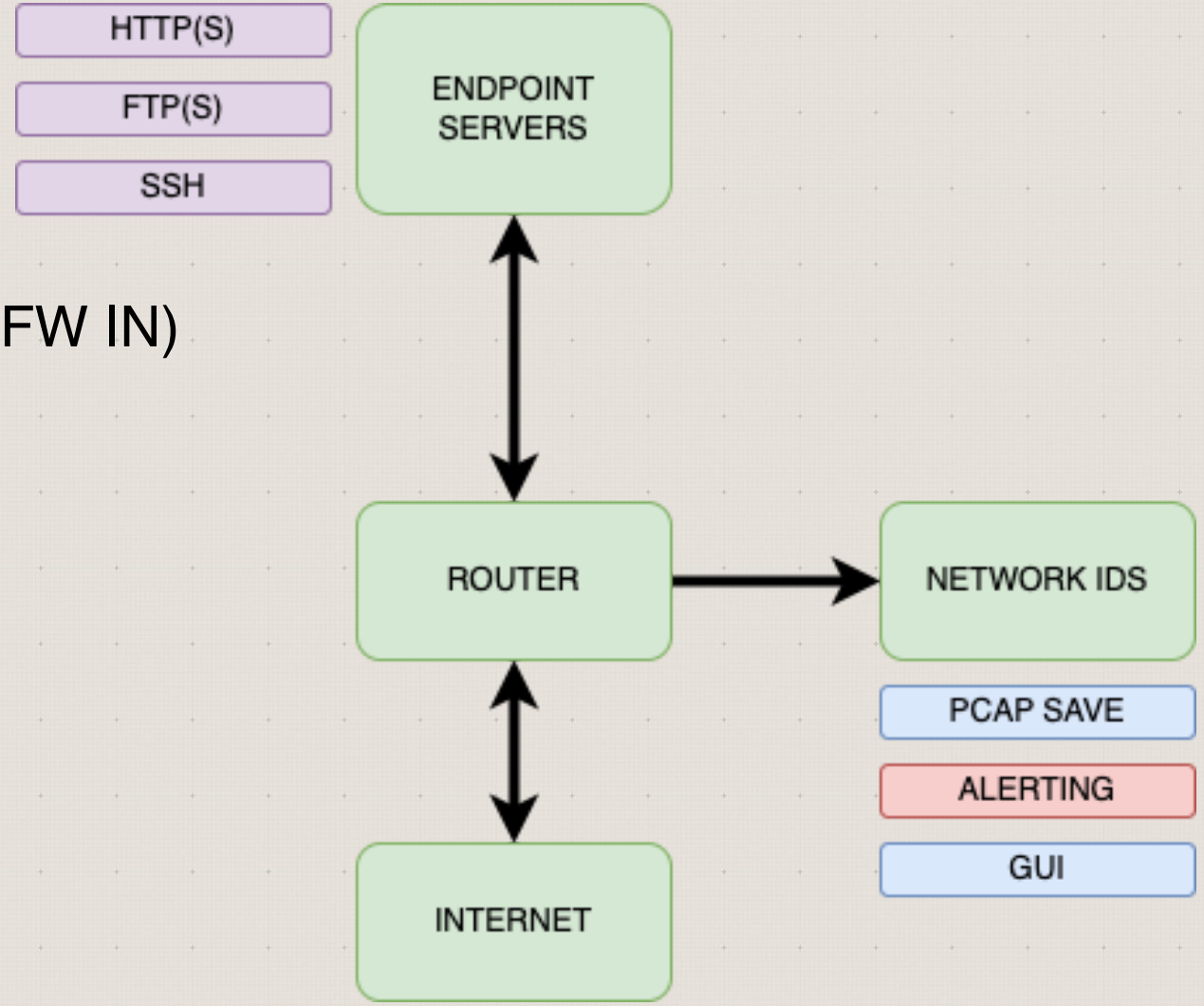
IPS / IDS / EDR / XDR

Inspekcja ruchu podejrzanego (FW IN)

Alertowanie

Selekcjonowanie

Monitoring

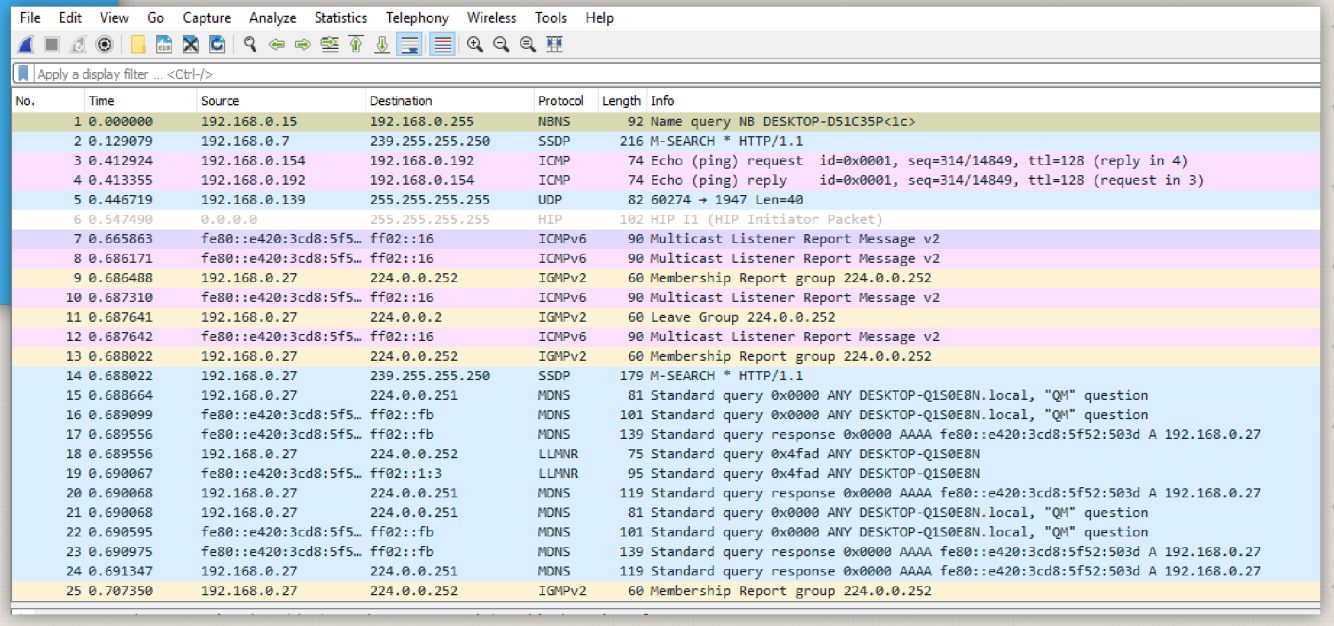


# Wireshark


2 tryby działania:
- ✓online
- ✓offline

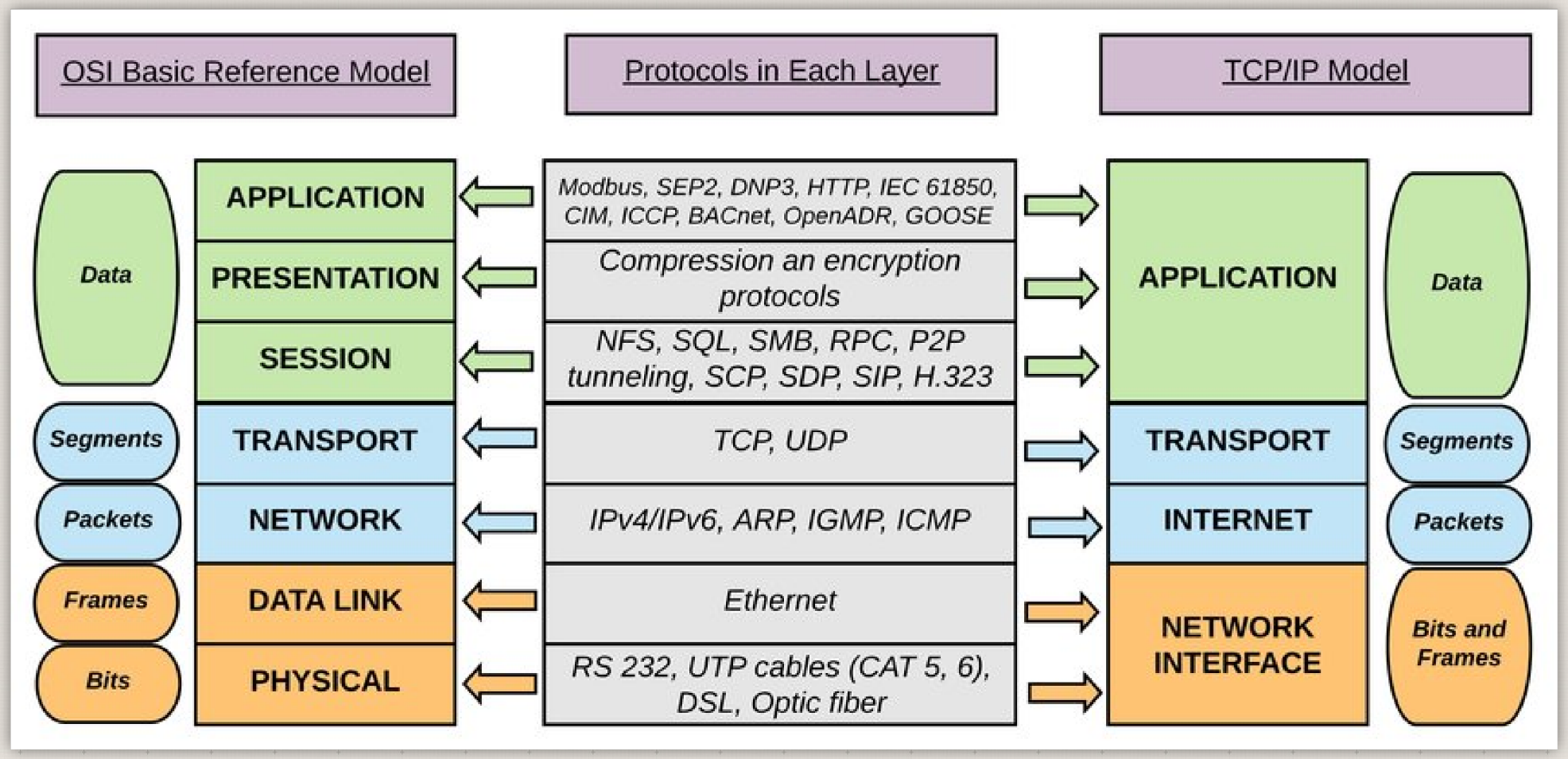
2 tryby chwytania:
- ✓promisc
- ✓monitor

https://www.wireshark.org/




0111001100001

# Komunikacja sieciowa




Źródło: researchgate.com

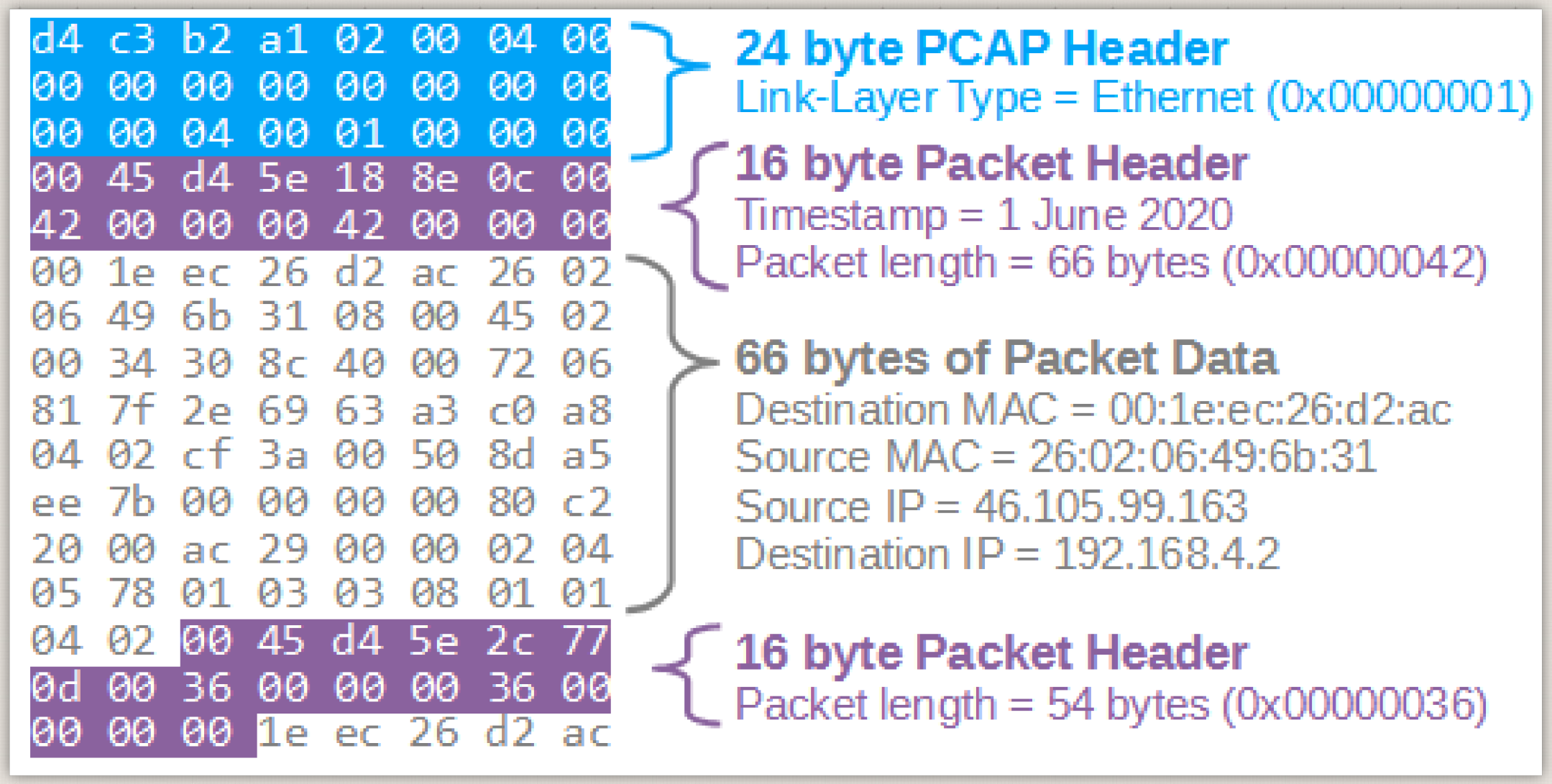
# PCAP file format

| offset(h) | 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 0A | 0B | 0C | 0D | 0E | 0F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 00 | Magic Number | | | | Version_Major | | Version_Minor | | Timezone | | | | accuracy of timestamps | | | |
| 01 | Max length of captured packet | | | | Datalink Type | | | | Timestamp seconds | | | | Timestamp microseconds | | | |
| 02 | packet(frame) length | | | | actual packet(frame) length | | | | packet data | | | | | | | |
| 03 | packet data | | | | | | | | | | | | | | | |
| 04 | packet data | | | | Timestamp seconds | | | | Timestamp microseconds | | | | packet(frame) length | | | actual |
| 05 | packet(frame) length | | | | packet data | | | | | | | | | | | |

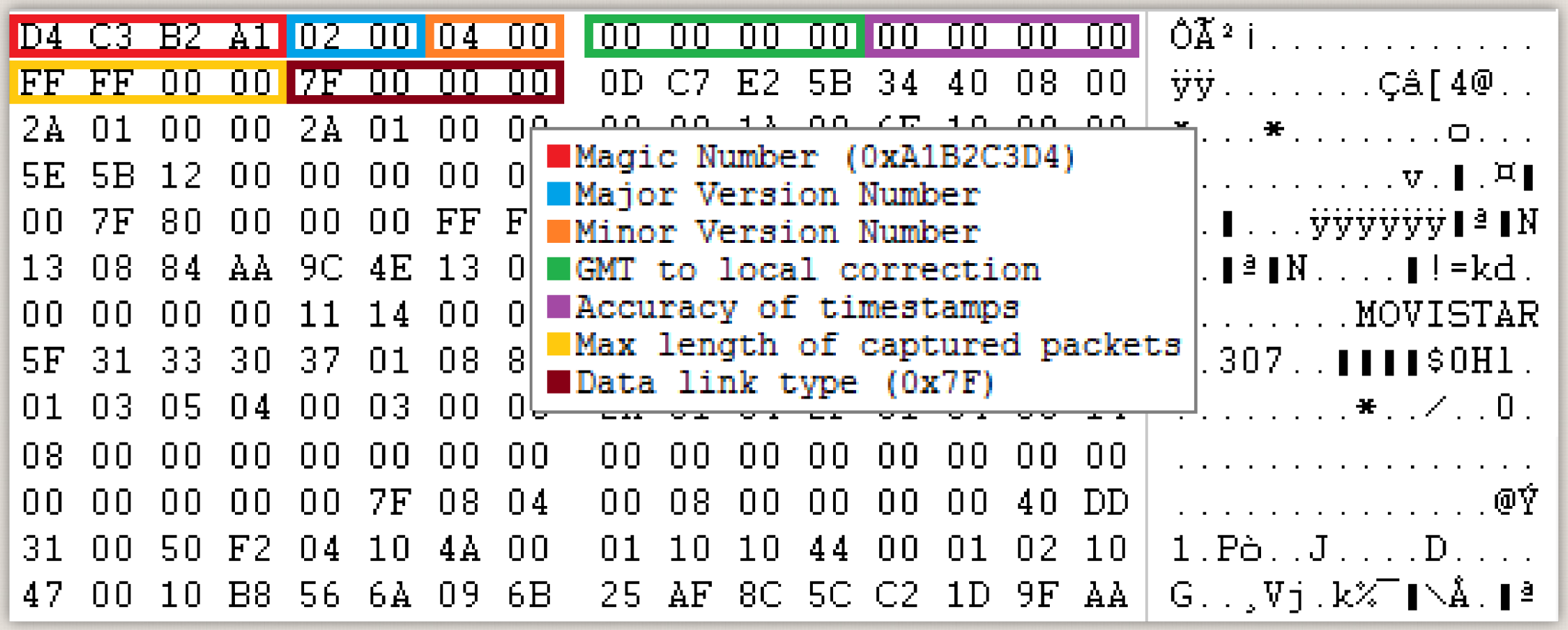
```
Offset(h)  00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
00000000   D4 C3 B2 A1 02 00 04 00 00 00 00 00 00 00 00 00
00000010   FF FF 00 00 01 00 00 00 00 00 00 00 00 00 00 00
00000020   16 02 00 00 16 02 00 00 00 10 F3 25 13 52 00 1B
```

https://github.com/the-tcpdump-group/libpcap

# PCAP file format



# PCAP file format



# Rodzaje filtrów w narzędziach

**Capture Filters** (tcpdump, tshark) - on the fly

udp and host 10.0.0.1

host 192.168.0.1

port 80

ether host 00:ff:11:22:33:ff

icmp

arp

tcp[13] == 0x12 (SYN-ACK)

dns.flags.response == 0

dst host 8.8.8.8

# Rodzaje filtrów w narzędziach

**Display Filters** (wireshark, tshark) - deep

ip.addr == 192.168.0.1

tcp.port == 80

ip.proto == 6 (TCP)

ip.src = 10.0.0.1 && http.user_agent contains "Mozilla"

frame.len == 100

http.response.status_code == 200

http.request.method == "POST"

tcp.flags.syn == 1

# Przykładowe filtry

udp dst port 53

udp src port 53

udp port 53

dst host sekurak.pl

(tcp and host sekurak.pl) or udp port 53

frame.len > 10

ip.addr == 129.10.13.37/16

https://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xhtml

eth.src[1-2] == 00:83

tcp.dst[0:1] == 6 && tcp.dst[2:3] == 29 xor udp.src[1] == 42

# Cheat sheets

## Wireshark Display Filters · Part 1

packetlife.net

### Ethernet

| | | |
|---|---|---|
| eth.addr | eth.len | eth.src |
| eth.dst | eth.lg | eth.trailer |
| eth.ig | eth.multicast | eth.type |

### IEEE 802.1Q

| | | |
|---|---|---|
| vlan.cfi | vlan.id | vlan.priority |
| vlan.etype | vlan.len | vlan.trailer |

### IPv4

| | |
|---|---|
| ip.addr | ip.fragment.overlap.conflict |
| ip.checksum | ip.fragment.toolongfragment |
| ip.checksum_bad | ip.fragments |
| ip.checksum_good | ip.hdr_len |
| ip.dsfield | ip.host |
| ip.dsfield.ce | ip.id |
| ip.dsfield.dscp | ip.len |
| ip.dsfield.ect | ip.proto |
| ip.dst | ip.reassembled_in |
| ip.dst_host | ip.src |
| ip.flags | ip.src_host |
| ip.flags.df | ip.tos |
| ip.flags.mf | ip.tos.cost |
| ip.flags.rb | ip.tos.delay |
| ip.frag_offset | ip.tos.precedence |
| ip.fragment | ip.tos.reliability |
| ip.fragment.error | ip.tos.throughput |
| ip.fragment.multipletails | ip.ttl |
| ip.fragment.overlap | ip.version |

### IPv6

| | |
|---|---|
| ipv6.addr | ipv6.hop_opt |
| ipv6.class | ipv6.host |
| ipv6.dst | ipv6.mipv6_home_address |
| ipv6.dst_host | ipv6.mipv6_length |
| ipv6.dst_opt | ipv6.mipv6_type |
| ipv6.flow | ipv6.nxt |
| ipv6.fragment | ipv6.opt.pad1 |
| ipv6.fragment.error | ipv6.opt.padn |
| ipv6.fragment.more | ipv6.plen |
| ipv6.fragment.multipletails | ipv6.reassembled_in |
| ipv6.fragment.offset | ipv6.routing_hdr |
| ipv6.fragment.overlap | ipv6.routing_hdr.addr |
| ipv6.fragment.overlap.conflict | ipv6.routing_hdr.left |
| ipv6.fragment.toolongfragment | ipv6.routing_hdr.type |
| ipv6.fragments | ipv6.src |
| ipv6.fragment.id | ipv6.src_host |
| ipv6.hlim | ipv6.version |

### ARP

| | |
|---|---|
| arp.dst.hw_mac | arp.proto.size |
| arp.dst.proto_ipv4 | arp.proto.type |
| arp.hw.size | arp.src.hw_mac |
| arp.hw.type | arp.src.proto_ipv4 |
| arp.opcode | |

### TCP

| | |
|---|---|
| tcp.ack | tcp.options.qs |
| tcp.checksum | tcp.options.sack |
| tcp.checksum_bad | tcp.options.sack_le |
| tcp.checksum_good | tcp.options.sack_perm |
| tcp.continuation_to | tcp.options.sack_re |
| tcp.dstport | tcp.options.time_stamp |
| tcp.flags | tcp.options.wscale |
| tcp.flags.ack | tcp.options.wscale_val |
| tcp.flags.cwr | tcp.pdu.last_frame |
| tcp.flags.ecn | tcp.pdu.size |
| tcp.flags.fin | tcp.pdu.time |
| tcp.flags.push | tcp.port |
| tcp.flags.reset | tcp.reassembled_in |
| tcp.flags.syn | tcp.segment |
| tcp.flags.urg | tcp.segment.error |
| tcp.hdr_len | tcp.segment.multipletails |
| tcp.len | tcp.segment.overlap |
| tcp.nxtseq | tcp.segment.overlap.conflict |
| tcp.options | tcp.segment.toolongfragment |
| tcp.options.cc | tcp.segments |
| tcp.options.ccecho | tcp.seq |
| tcp.options.ccnew | tcp.srcport |
| tcp.options.echo | tcp.time_delta |
| tcp.options.echo_reply | tcp.time_relative |
| tcp.options.md5 | tcp.urgent_pointer |
| tcp.options.mss | tcp.window_size |
| tcp.options.mss_val | |

### UDP

| | | |
|---|---|---|
| udp.checksum | udp.dstport | udp.srcport |
| udp.checksum_bad | udp.length | |
| udp.checksum_good | udp.port | |

### Operators

| | |
|---|---|
| eq or == | |
| ne or != | |
| gt or > | |
| lt or < | |
| ge or >= | |
| le or <= | |

### Logic

| | |
|---|---|
| and or && | Logical AND |
| or or \|\| | Logical OR |
| xor or ^^ | Logical XOR |
| not or ! | Logical NOT |
| [n] [...] | Substring operator |

by Jeremy Stretch v2.0

## Wireshark Display Filters · Part 2

packetlife.net

### Frame Relay

| | |
|---|---|
| fr.becn | fr.de |
| fr.chdlctype | fr.dlci |
| fr.control | fr.dlcore_control |
| fr.control.f | fr.ea |
| fr.control.ftype | fr.fecn |
| fr.control.n_r | fr.lower_dlci |
| fr.control.n_s | fr.nlpid |
| fr.control.p | fr.second_dlci |
| fr.control.s_ftype | fr.snap.oui |
| fr.control.u_modifier_cmd | fr.snap.pid |
| fr.control.u_modifier_resp | fr.snaptype |
| fr.cr | fr.third_dlci |
| fr.dc | fr.upper_dlci |

### PPP

| | |
|---|---|
| ppp.address | ppp.direction |
| ppp.control | ppp.protocol |

### MPLS

| | |
|---|---|
| mpls.bottom | mpls.oam.defect_location |
| mpls.cw.control | mpls.oam.defect_type |
| mpls.cw.res | mpls.oam.frequency |
| mpls.exp | mpls.oam.function_type |
| mpls.label | mpls.oam.ttsi |
| mpls.oam.bip16 | mpls.ttl |

### ICMP

| | | |
|---|---|---|
| icmp.checksum | icmp.ident | icmp.seq |
| icmp.checksum_bad | icmp.mtu | icmp.type |
| icmp.code | icmp.redir_gw | |

### DTP

| | | |
|---|---|---|
| dtp.neighbor | dtp.tlv_type | vtp.neighbor |
| dtp.tlv_len | dtp.version | |

### VTP

| | |
|---|---|
| vtp.code | vtp.vlan_info.802_10_index |
| vtp.conf_rev_num | vtp.vlan_info.isl_vlan_id |
| vtp.followers | vtp.vlan_info.len |
| vtp.md | vtp.vlan_info.mtu_size |
| vtp.md5_digest | vtp.vlan_info.status.vlan_susp |
| vtp.md_len | vtp.vlan_info.tlv_len |
| vtp.seq_num | vtp.vlan_info.tlv_type |
| vtp.start_value | vtp.vlan_info.vlan_name |
| vtp.upd_id | vtp.vlan_info.vlan_name_len |
| vtp.upd_ts | vtp.vlan_info.vlan_type |
| vtp.version | |

### ICMPv6

| | |
|---|---|
| icmpv6.all_comp | icmpv6.option.name_type.fqdn |
| icmpv6.checksum | icmpv6.option.name_x501 |
| icmpv6.checksum_bad | icmpv6.option.rsa.key_hash |
| icmpv6.code | icmpv6.option.type |
| icmpv6.comp | icmpv6.ra.cur_hop_limit |
| icmpv6.haad.ha_addrs | icmpv6.ra.reachable_time |
| icmpv6.identifier | icmpv6.ra.retrans_timer |
| icmpv6.option | icmpv6.ra.router_lifetime |
| icmpv6.option.cga | icmpv6.recursive_dns_serv |
| icmpv6.option.length | icmpv6.type |
| icmpv6.option.name_type | |

### RIP

| | | |
|---|---|---|
| rip.auth.passwd | rip.ip | rip.route_tag |
| rip.auth.type | rip.metric | rip.routing_domain |
| rip.command | rip.netmask | rip.version |
| rip.family | rip.next_hop | |

### BGP

| | |
|---|---|
| bgp.aggregator_as | bgp.mp_reach_nlri_ipv4_prefix |
| bgp.aggregator_origin | bgp.mp_unreach_nlri_ipv4_prefix |
| bgp.as_path | bgp.multi_exit_disc |
| bgp.cluster_identifier | bgp.next_hop |
| bgp.cluster_list | bgp.nlri_prefix |
| bgp.community_as | bgp.origin |
| bgp.community_value | bgp.originator_id |
| bgp.local_pref | bgp.type |
| bgp.mp_nlri_tnl_id | bgp.withdrawn_prefix |

### HTTP

| | |
|---|---|
| http.accept | http.proxy_authorization |
| http.accept_encoding | http.proxy_connect_host |
| http.accept_language | http.proxy_connect_port |
| http.authbasic | http.referer |
| http.authorization | http.request |
| http.cache_control | http.request.method |
| http.connection | http.request.uri |
| http.content_encoding | http.request.version |
| http.content_length | http.response |
| http.content_type | http.response.code |
| http.cookie | http.server |
| http.date | http.set_cookie |
| http.host | http.transfer_encoding |
| http.last_modified | http.user_agent |
| http.location | http.www_authenticate |
| http.notification | http.x_forwarded_for |
| http.proxy_authenticate | |

by Jeremy Stretch v2.0

https://packetlife.net/media/library/13/Wireshark_Display_Filters.pdf

0111001100001

# Cheat sheets

https://github.com/tturba/wireshark

```
#Filtruj pakiety zawierające tekst "POST" w polu danych
  data-text-lines == "POST"
#Filtruj pakiety ze specyficznym ciasteczkiem
  http.cookie contains "SESSIONID="
#Filtruj metody GET do konkretnego hosta
  tcp contains "GET" && http.host == "sekurak.pl"
#Filtruj zapytania DNS dla konkretnej domeny
  udp contains "DNS" && dns.qry.name == "sekurak.pl"
#Filtruj User-Agent "Mozilla" dla konkretnego adresu IP
  ip.src = 10.0.0.1 && http.user_agent contains "Mozilla"
#Filtruj tylko odpowiedzi HTTP = 200
  http.response.status_code == 200
#Pokaż tylko flagi TCP SYN
  tcp.flags.syn == 1
#Filtruj pakiety SYN-ACK
  tcp[13] == 0x12
```

# Wireshark - UI tweak

1. Tryby działania
2. Tweaks:
   - ✓Time display format
   - ✓Coloring rules
   - ✓Capture options / Profile
   - ✓Column preferences: STREAM, HOST
   - ✓Help > Sample captures

# malware.pcap

Statystyki IP + HTTP.host

↓

Metoda GET / POST?

↓

/*.exe --> MZ

↓

0d 0a 0d 0a / CLRF

↓

checksum

VIRUSTOTAL

https://virustotal.com

PacketTotal
Simple, free, high-quality PCAP analysis

https://packettotal.com/

ANY RUN
INTERACTIVE MALWARE ANALYSIS

https://any.run

# stealer.pcap

Statystyki IPv4?

↓

Konwersacje TCP?

↓

Metoda GET / POST?

↓

Stream [x]

↓

TLS wersja 0x030x + JA3

↓

Export HTTP


https://malware-traffic-analysis.net/training-exercises.html

pw: infected

# ssl.pcap

Windows:

set SSLKEYLOGFILE =%USERPROFILE%/ssl.log

Windows PS:

$env:SSLKEYLOGFILE= "env:USERPROFILE\ssl.log"

Linux:

export SSLKEYLOGFILE="/home/sekurak/ssl.log"


tshark CLI:

#tshark -o 'tls.keylog_file:logfile.pms' -r capture.pcap

# ssl.pcap

SSL decipher?

↓

SSLKEYLOGFILE $ENV

↓

frame contains "sekurak"

↓

Edit > Preferences > Protocols > TLS

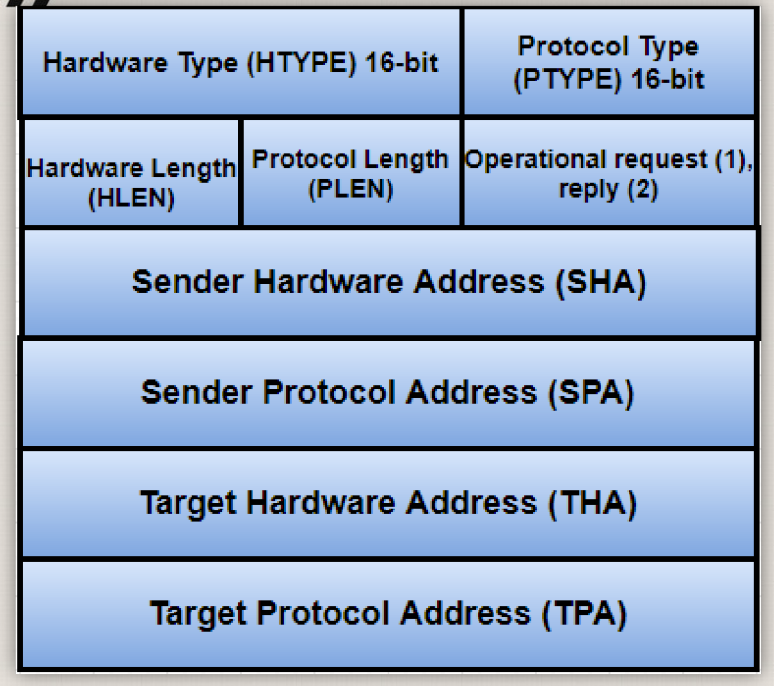
↓

Metoda GET / POST?

↓

TCP Reassembly + Uncompressed entity body

# mitm.pcap :: ARP header




```
Command Prompt
Microsoft Windows [Version 10.0.19041.388]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\User A>arp -a

Interface: 172.16.55.5 --- 0x14
  Internet Address      Physical Address      Type
  172.16.55.1           7a-4f-43-36-82-65     dynamic
  172.16.55.255         ff-ff-ff-ff-ff-ff     static
  224.0.0.22            01-00-5e-00-00-16     static
  224.0.0.251           01-00-5e-00-00-fb     static
  224.0.0.252           01-00-5e-00-00-fc     static
  239.255.255.250       01-00-5e-7f-ff-fa     static
  255.255.255.255       ff-ff-ff-ff-ff-ff     static

C:\Users\User A>
```

```
root@kali:~# arp -e
Address                  HWtype  HWaddress           Flags Mask            Iface
machine2                 ether   08:00:27:27:d6:c7   C                     eth0
10.0.2.3                 ether   08:00:27:e5:fd:ed   C                     eth0
_gateway                 ether   52:54:00:12:35:00   C                     eth0
root@kali:~# arp -s 10.0.2.4 08:00:27:AD:87:B3
root@kali:~# arp -e
Address                  HWtype  HWaddress           Flags Mask            Iface
machine1                 ether   08:00:27:ad:87:b3   CM                    eth0
machine2                 ether   08:00:27:27:d6:c7   C                     eth0
10.0.2.3                 ether   08:00:27:e5:fd:ed   C                     eth0
_gateway                 ether   52:54:00:12:35:00   C                     eth0
root@kali:~#
```

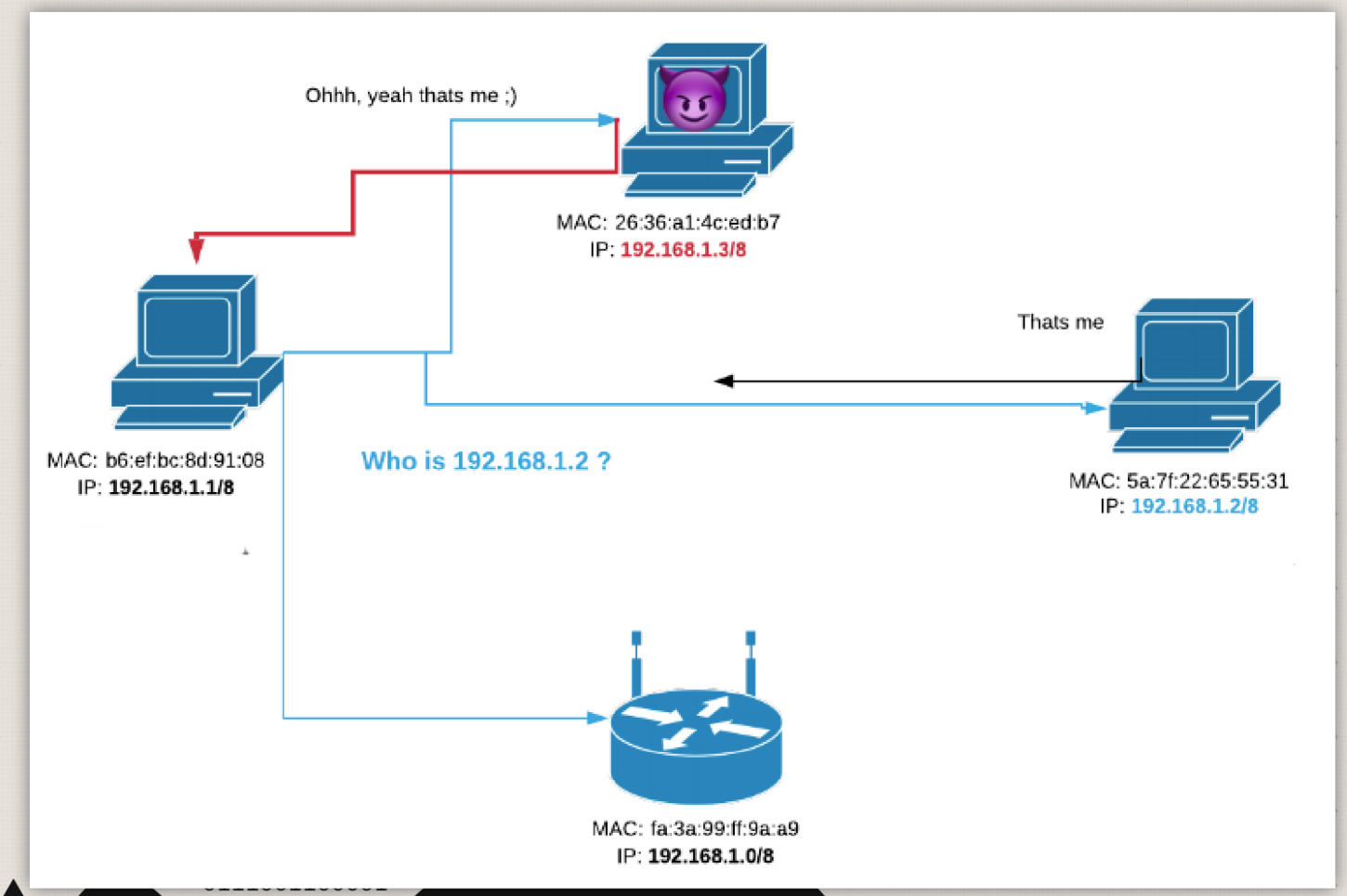
# mitm.pcap :: ARP poison vs spoof

ARP poison --> korupcja tablicy MAC adresów na minimum jednym hoście

ARP spoofing --> impersonifikacja, że jest się hostem

W praktyce?

ARP poison = ARP spoofing



# mitm.pcap

arp active catching

↓

mitm{[b]ettercap}

↓

arp reply - opcode = 2

↓

arp.duplicate-address-detected?

↓

arp.src.proto_ipv4 == Sender IP address  && (arp.opcode == 2)

↓

sca.py + arp.opcode == 1?

# Podsumowanie

- ISO/TCP model && PCAP
- Wireshark - UI/UX -> tcp.stream && http.host
- Discovery malware -> http.request.method == "GET" / "POST"
- Discovery stealer -> Follow TCP/HTTP/UDP Stream [x] && JA3 fingerprint
- SSLKeyLogFile -> Preferences -> TLS
- (arp.src.proto_ipv4 == $IP) && (arp.opcode == 2) && !(arp.src.hw_mac == $MAC)

# Wanna see a Bumblebee?

https://sekurak.pl/bumblebee-loader-nowa-droga-do-przejecia-domeny-active-directory/

https://github.com/tturba/wireshark

UWAGA

Wszelkie analizy komunikacji sieciowej możliwej do zreplikowania należy wykonywać w odseparowanym, zamkniętym i bezpiecznym środowisku

# Pytania?
# Dziękuję za uwagę

Hasło do certyfikatu? Po wypełnieniu ankiety:

https://forms.gle/Hcvc3yxPKfCt2tUn9

Konkurs?

tomasz.turba@securitum.pl

tturba tturba



tomasz.turba@securitum.pl
tturba
tturba