



SharkFest '17 Europe

Custom LUA dissectors to the rescue in RCA

How to write a LUA dissector
for a proprietary protocol to
assist in troubleshooting

9 november 2017



Sake Blok

relational therapist for computer systems

sake.blok@SYN-bit.nl



SYN-bit
deep traffic analysis



Hands-on files:

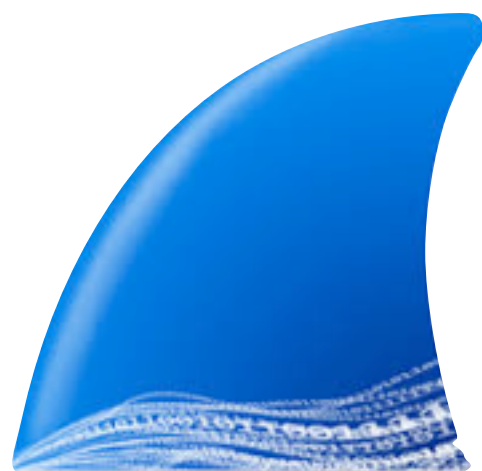


<http://www.SYN-bit.nl/files/sf17eu-lua.zip>





About me...





SYN-bit
deep traffic analysis

Application and network troubleshooting

Protocol and packet analysis

Training (Wireshark, TCP, SSL)

www.SYN-bit.nl



Houston, we have a problem...

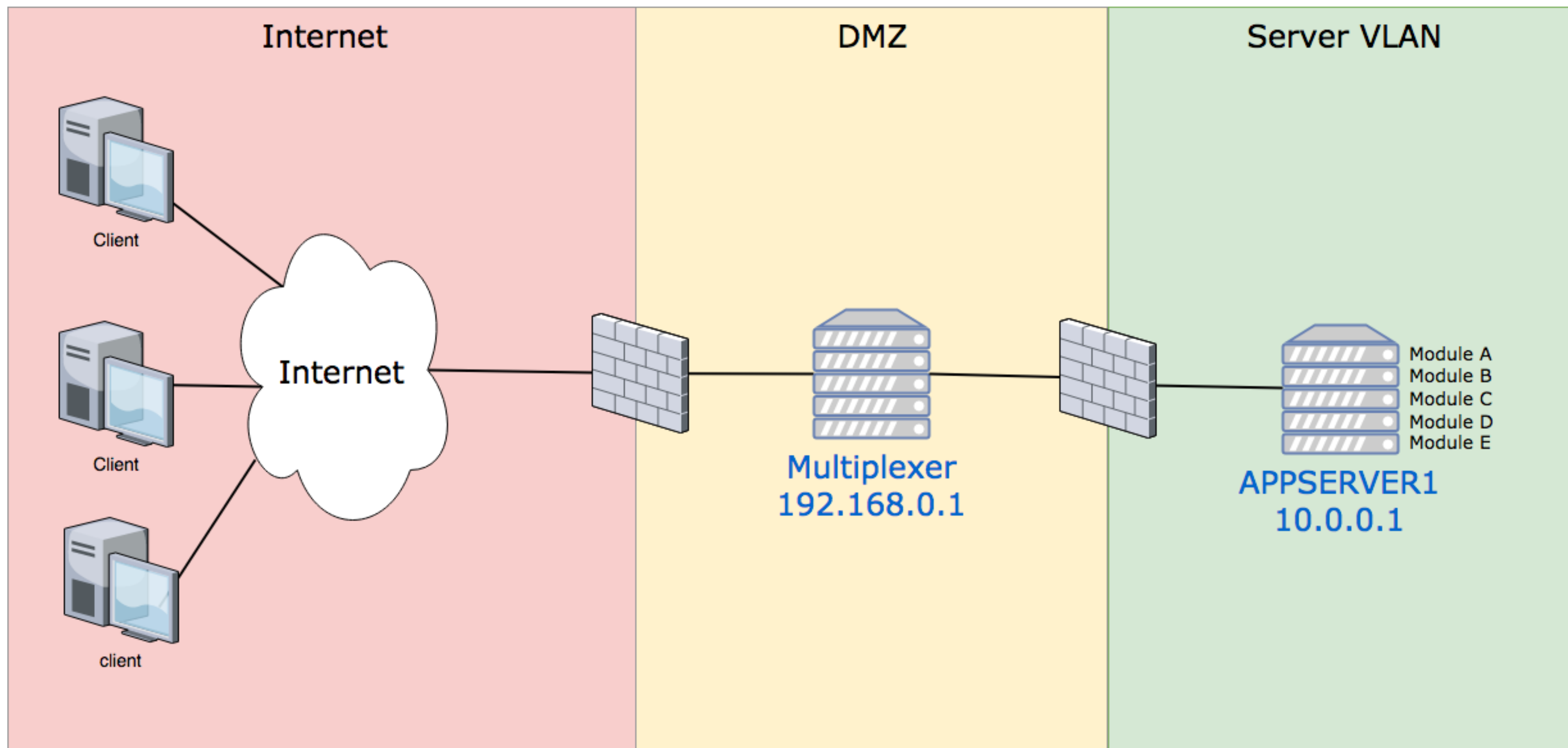


- Trading platform
- Connections being reset
- Logfiles point to network
- Software supplier blames the network (duh!)
- Custom protocol, so just TCP analysis possible
- What now?



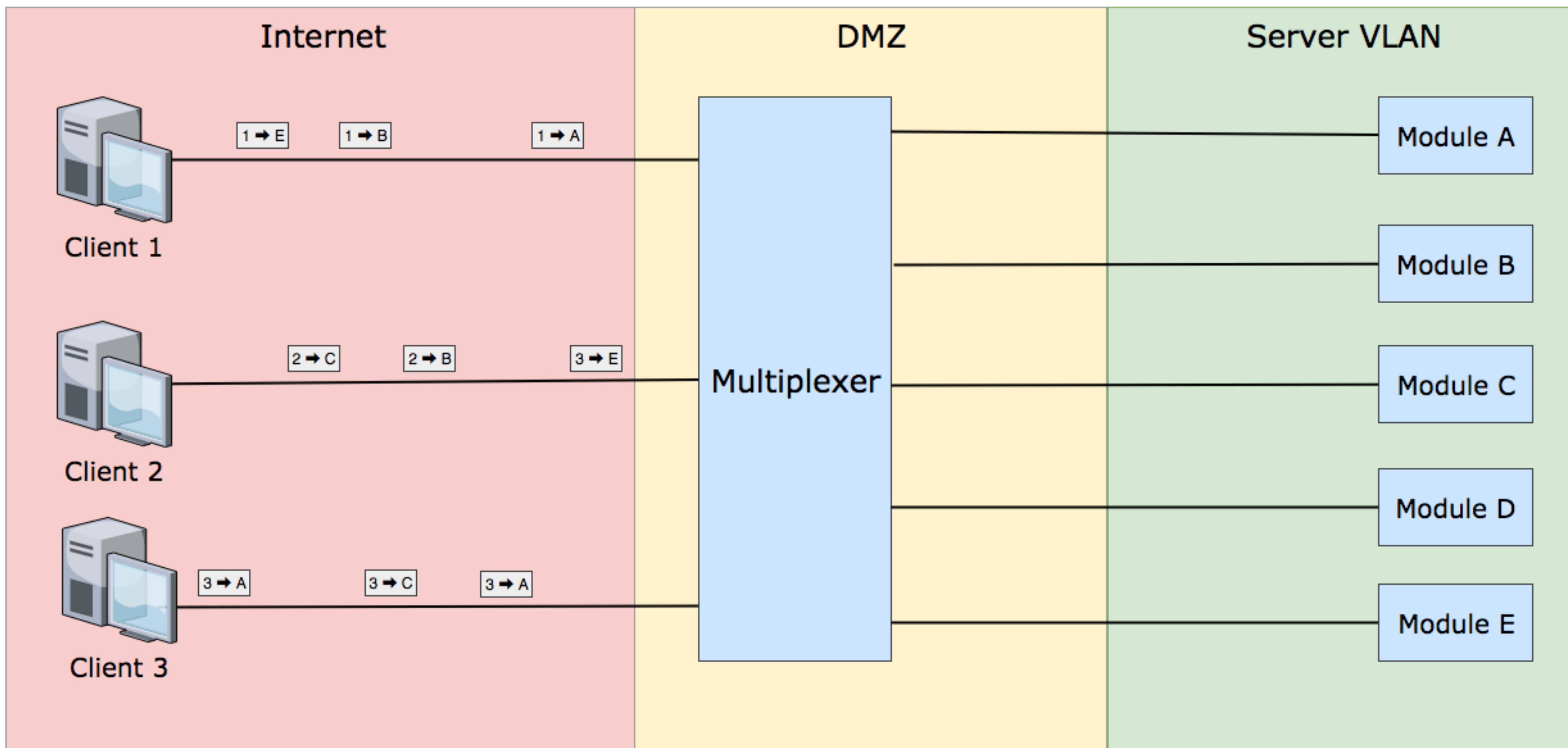


So how does the application work?



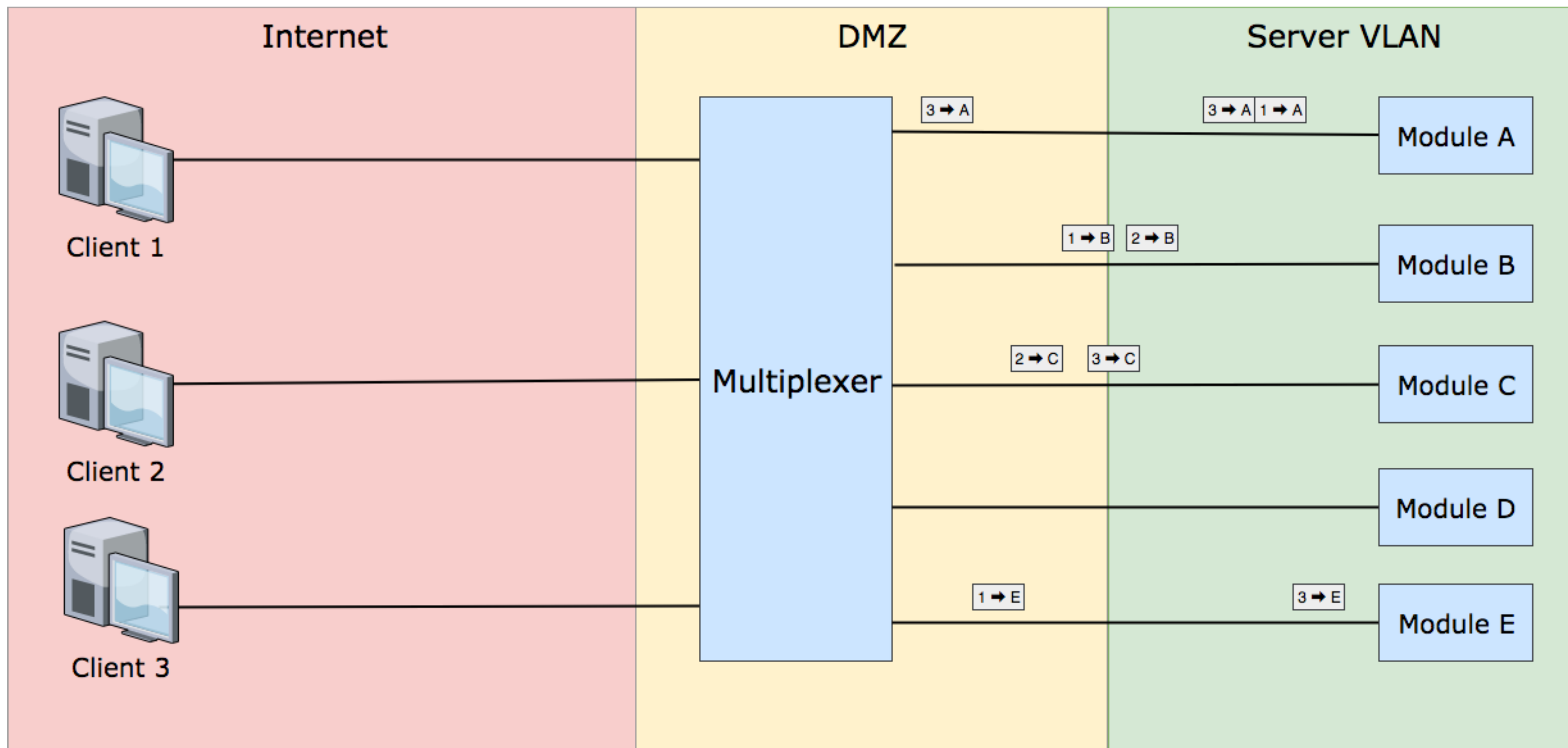


PDU's coming in



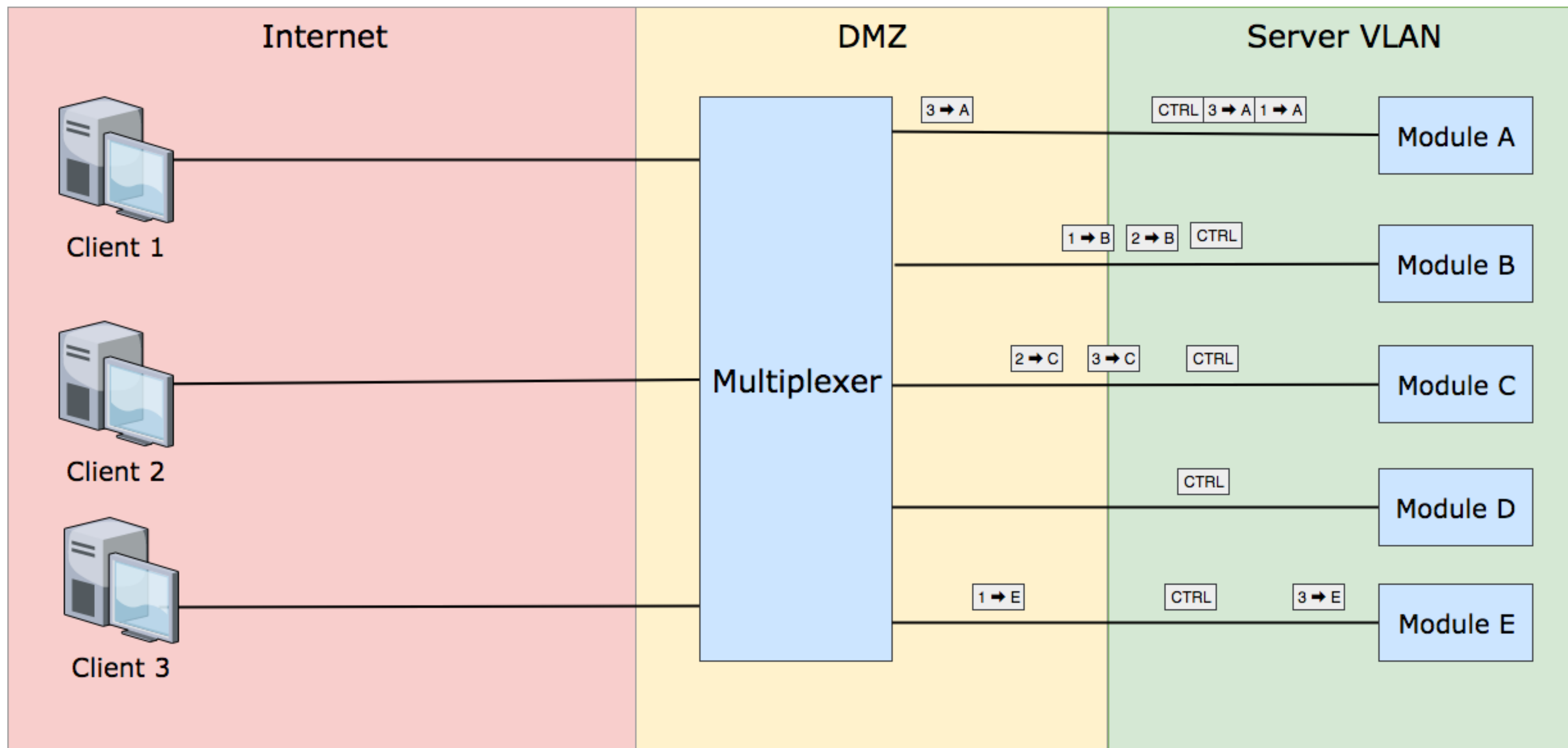


PDU's going out





... plus Control PDU's





Trader module Log



```
2014-06-05 09:27:51,883 NOTICE - Creating clientConnection: 2654 user: XXXXXX0020 on: Trader
2014-06-05 09:28:57,779 NOTICE - Creating clientConnection: 2655 user: XXXXXX0019 on: Trader
2014-06-05 09:30:12,755 NOTICE - Closing clientConnection: 2652 from: Trader
2014-06-05 09:30:48,043 NOTICE - Creating clientConnection: 2656 user: XXXXXX0013 on: Trader
2014-06-05 09:31:58,978 ERROR - No response from multiplexer: 192.168.0.1:2107 within 1 minutes 30 seconds. Terminating connection.
2014-06-05 09:31:58,978 NOTICE - Shutdown connection from: Trader
2014-06-05 09:31:58,978 WARN - ModuleServer Trader lost connection to multiplexer on 192.168.0.1:2107
2014-06-05 09:37:40,252 NOTICE - Creating clientConnection: 1129 user: APPSERVER1 on: Trader
2014-06-05 09:42:03,556 WARN - Connection lost in receive loop: Trader
2014-06-05 09:42:03,556 NOTICE - Shutdown connection from: Trader
2014-06-05 09:42:03,556 WARN - ModuleServer Trader lost connection to multiplexer on 192.168.0.1:2107
2014-06-05 09:43:54,147 NOTICE - Creating clientConnection: 2003 user: XXXXXX0016 on: Trader
```





Multiplexer Log



```
2014-06-05 09:34:41,006 NOTICE - Logging in clientConnection: 2678 user: XXXXXX0009 to: MarketConfiguration
2014-06-05 09:34:41,006 NOTICE - Logging in clientConnection: 2678 user: XXXXXX0009 to: FinancialModule
2014-06-05 09:34:52,816 NOTICE - Disabling client logins on port: '443'
2014-06-05 09:34:52,816 NOTICE - Client accept loop stopped
2014-06-05 09:34:53,362 NOTICE - Logging in clientConnection: 2671 user: XXXXXX0000 to: BackOffice
2014-06-05 09:34:53,580 NOTICE - Logging in clientConnection: 2671 user: XXXXXX0000 to: Trader
2014-06-05 09:34:53,580 NOTICE - Logging in clientConnection: 2671 user: XXXXXX0000 to: MarketConfiguration
2014-06-05 09:34:53,580 NOTICE - Logging in clientConnection: 2671 user: XXXXXX0000 to: FinancialModule
2014-06-05 09:34:56,622 NOTICE - Logging in clientConnection: 2679 user: XXXXXX0021 to: BackOffice
2014-06-05 09:34:56,825 NOTICE - Logging in clientConnection: 2679 user: XXXXXX0021 to: Trader
2014-06-05 09:34:56,825 NOTICE - Logging in clientConnection: 2679 user: XXXXXX0021 to: MarketConfiguration
2014-06-05 09:34:56,825 NOTICE - Logging in clientConnection: 2679 user: XXXXXX0021 to: FinancialModule
2014-06-05 09:35:01,458 sss 127.000.000.001 Operator 2014-06-05 09:35:01.458 Multiplexer stopped.
Stopped logging 2014-06-05 09:35:01,458
Started logging 2014-06-05 09:35:16,060
2014-06-05 09:35:16,060 sss 127.000.000.001 Operator 2014-06-05 09:35:16.060 Multiplexer started.
2014-06-05 09:35:32,924 NOTICE - Received start id: 2001
2014-06-05 09:35:32,940 NOTICE - Received security parameters
2014-06-05 09:41:26,864 sss 127.000.000.001 Operator 2014-06-05 09:41:26.864 Multiplexer stopped.
Stopped logging 2014-06-05 09:41:26,864
Started logging 2014-06-05 09:42:46,004
2014-06-05 09:42:46,004 sss 127.000.000.001 Operator 2014-06-05 09:42:46.004 Multiplexer started.
2014-06-05 09:42:50,107 NOTICE - Received start id: 2001
2014-06-05 09:42:50,138 NOTICE - Received security parameters
2014-06-05 09:43:28,531 NOTICE - Enabling client logins on port: '443'
2014-06-05 09:43:28,531 NOTICE - Client accept loop started
2014-06-05 09:43:53,647 NOTICE - Logging in clientConnection: 2003 user: XXXXXX0016 to: BackOffice
2014-06-05 09:43:53,865 NOTICE - Logging in clientConnection: 2003 user: XXXXXX0016 to: Trader
```





Timeline



```
2014-06-05 09:31:58,978 ERROR - No response from multiplexer: 192.168.0.1:2107 within 1 minutes 30 seconds. Terminating connection.
2014-06-05 09:31:58,978 WARN - ModuleServer Trader lost connection to multiplexer on 192.168.0.1:2107

2014-06-05 09:34:52,816 NOTICE - Disabling client logins on port: '443'
2014-06-05 09:34:52,816 NOTICE - Client accept loop stopped

2014-06-05 09:35:01,458 sss          127.000.000.001      Operator      2014-06-05 09:35:01.458      Multiplexer stopped.
2014-06-05 09:35:16,060 sss          127.000.000.001      Operator      2014-06-05 09:35:16.060      Multiplexer started.
2014-06-05 09:35:32,924 NOTICE - Received start id: 2001

2014-06-05 09:41:26,864 sss          127.000.000.001      Operator      2014-06-05 09:41:26.864      Multiplexer stopped.
2014-06-05 09:42:46,004 sss          127.000.000.001      Operator      2014-06-05 09:42:46.004      Multiplexer started.
2014-06-05 09:42:50,107 NOTICE - Received start id: 2001
2014-06-05 09:43:28,531 NOTICE - Enabling client logins on port: '443'
2014-06-05 09:43:28,531 NOTICE - Client accept loop started

2014-06-05 09:43:53,647 NOTICE - Logging in clientConnection: 2003 user: XXXXXX0016 to: BackOffice
```





Packet timestamps don't match



| No. | Time | Delta | Source | Destination | Protocol | Length | Info |
|-------|-----------------|----------|-------------|-------------|----------|--------|---|
| 31362 | 09:31:49.982437 | 0.000559 | 10.0.0.1 | 192.168.0.1 | TCP | 60 | 1195 → 2107 [ACK] Seq=21251242 Ack=397458 Win=7747840 Len=0 |
| 31363 | 09:31:50.158695 | 0.176258 | 192.168.0.1 | 10.0.0.1 | TCP | 1434 | [TCP ZeroWindow] 2107 → 1195 [ACK] Seq=397458 Ack=21251242 Win=0 Len=1380 |
| 31364 | 09:31:50.158699 | 0.000004 | 192.168.0.1 | 10.0.0.1 | TCP | 1434 | [TCP ZeroWindow] 2107 → 1195 [ACK] Seq=398838 Ack=21251242 Win=0 Len=1380 |
| 31365 | 09:31:50.158703 | 0.000004 | 192.168.0.1 | 10.0.0.1 | TCP | 1434 | [TCP ZeroWindow] 2107 → 1195 [ACK] Seq=400218 Ack=21251242 Win=0 Len=1380 |
| 31366 | 09:31:50.158707 | 0.000004 | 192.168.0.1 | 10.0.0.1 | TCP | 1434 | [TCP ZeroWindow] 2107 → 1195 [ACK] Seq=401598 Ack=21251242 Win=0 Len=1380 |
| 31367 | 09:31:50.158711 | 0.000004 | 192.168.0.1 | 10.0.0.1 | TCP | 875 | [TCP ZeroWindow] 2107 → 1195 [PSH, ACK] Seq=402978 Ack=21251242 Win=0 Len=821 |
| 31368 | 09:31:50.159241 | 0.000530 | 10.0.0.1 | 192.168.0.1 | TCP | 60 | 1195 → 2107 [ACK] Seq=21251242 Ack=403799 Win=7741440 Len=0 |
| 31369 | 09:31:53.888822 | 3.729581 | 10.0.0.1 | 192.168.0.1 | TCP | 65 | 1170 → 2107 [PSH, ACK] Seq=148616 Ack=180900 Win=130816 Len=11 |
| 31370 | 09:31:54.091036 | 0.202214 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 → 1170 [ACK] Seq=180900 Ack=148627 Win=130816 Len=0 |
| 31371 | 09:31:54.170051 | 0.079015 | 10.0.0.1 | 192.168.0.1 | TCP | 65 | 1170 → 2107 [PSH, ACK] Seq=148627 Ack=180900 Win=130816 Len=11 |
| 31372 | 09:31:54.371839 | 0.201788 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 → 1170 [ACK] Seq=180900 Ack=148638 Win=130816 Len=0 |
| 31373 | 09:31:54.888059 | 0.516220 | 10.0.0.1 | 192.168.0.1 | TCP | 65 | 1170 → 2107 [PSH, ACK] Seq=148638 Ack=180900 Win=130816 Len=11 |
| 31374 | 09:31:55.105074 | 0.217015 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 → 1170 [ACK] Seq=180900 Ack=148649 Win=130816 Len=0 |
| 31375 | 09:31:59.226292 | 4.121218 | 10.0.0.1 | 192.168.0.1 | TCP | 65 | 1170 → 2107 [PSH, ACK] Seq=148649 Ack=180900 Win=130816 Len=11 |
| 31376 | 09:31:59.441887 | 0.215595 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 → 1170 [ACK] Seq=180900 Ack=148660 Win=130816 Len=0 |
| 31377 | 09:32:03.343767 | 3.901880 | 10.0.0.1 | 192.168.0.1 | TCP | 66 | 1592 → 2107 [SYN] Seq=0 Win=8192 Len=0 MSS=1380 WS=256 SACK_PERM=1 |
| 31378 | 09:32:03.344095 | 0.000328 | 192.168.0.1 | 10.0.0.1 | TCP | 66 | 2107 → 1592 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1 |
| 31379 | 09:32:03.344383 | 0.000288 | 10.0.0.1 | 192.168.0.1 | TCP | 60 | 1592 → 2107 [ACK] Seq=1 Ack=1 Win=131072 Len=0 |
| 31380 | 09:32:03.344877 | 0.000494 | 10.0.0.1 | 192.168.0.1 | TCP | 63 | 1592 → 2107 [PSH, ACK] Seq=1 Ack=1 Win=131072 Len=9 |
| 31381 | 09:32:03.349625 | 0.004748 | 192.168.0.1 | 10.0.0.1 | TCP | 71 | 2107 → 1170 [PSH, ACK] Seq=180900 Ack=148660 Win=130816 Len=17 |
| 31382 | 09:32:03.544694 | 0.195069 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 → 1592 [ACK] Seq=1 Ack=10 Win=131072 Len=0 |

| No. | Time | Delta | Source | Destination | Protocol | Length | Info |
|-------|-----------------|----------|-------------|-------------|----------|--------|--|
| 31488 | 09:32:12.351357 | 0.000004 | 192.168.0.1 | 10.0.0.1 | TCP | 1434 | [TCP ZeroWindow] 2107 → 1195 [ACK] Seq=440839 Ack=21251242 Win=0 Len=1380 |
| 31489 | 09:32:12.351362 | 0.000005 | 192.168.0.1 | 10.0.0.1 | TCP | 1434 | [TCP ZeroWindow] 2107 → 1195 [ACK] Seq=442219 Ack=21251242 Win=0 Len=1380 |
| 31490 | 09:32:12.351366 | 0.000004 | 192.168.0.1 | 10.0.0.1 | TCP | 1214 | [TCP ZeroWindow] 2107 → 1195 [PSH, ACK] Seq=443599 Ack=21251242 Win=0 Len=1160 |
| 31491 | 09:32:12.351912 | 0.000546 | 10.0.0.1 | 192.168.0.1 | TCP | 60 | 1195 → 2107 [RST, ACK] Seq=21251242 Ack=405179 Win=0 Len=0 |





The Blaming Game



- Zero Window on Multiplexer

- ▶ But why does the Trader module report it did not receive a response while it is still receiving lots of data?

- TCP/RST from APPSERVER1

- ▶ But not at the time of the log message!

- ~~The network is the problem~~

- ▶ But what is? And how to convince the software vendor?
- ▶ Client (APPSERVER1) or server (Multiplexer) side problem?





LUA to the rescue...



- Different LUA dissection types
- Skeleton Dissector
- Parsing the PDU header
- Reassembly
- Starting point:
<https://wiki.wireshark.org/Lua>





LUA in Wireshark



- Lua can be used to write dissectors, post-dissectors and taps.
 - ▶ Although it's possible to write **dissectors in Lua**, Wireshark dissectors are written in C, as C is several times faster than Lua. Lua is ok for prototyping dissectors, during Reverse Engineering you can use your time for finding out how things work instead of compiling and debugging your C dissector.
 - ▶ **Post-dissectors** are dissectors meant to run after every other dissector has run. They can add items to the dissection tree so they can be used to create your own extensions to the filtering mechanism.
 - ▶ **Taps** are used to collect information after the packet has been dissected.
- So for our purpose, a normal Lua dissector is the best fit

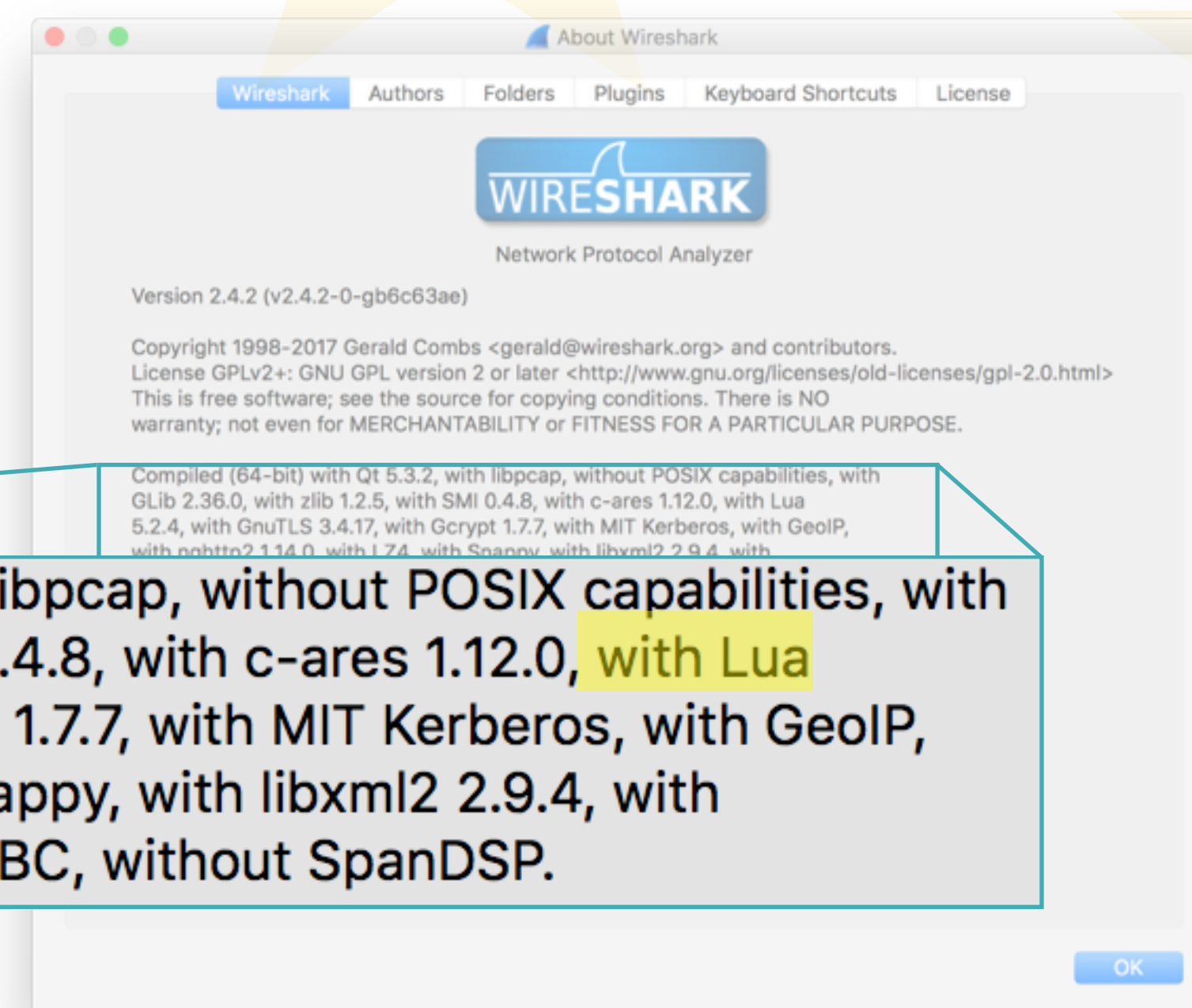




LUA supported?



- LUA has shipped with the Windows version of Wireshark since 0.99.4.
- Availability on other platforms varies.
 - ▶ Check: "Help -> About Wireshark"



Compiled (64-bit) with Qt 5.3.2, with libpcap, without POSIX capabilities, with GLib 2.36.0, with zlib 1.2.5, with SMI 0.4.8, with c-ares 1.12.0, with Lua 5.2.4, with GnuTLS 3.4.17, with Gcrypt 1.7.7, with MIT Kerberos, with GeolP, with nghttp2 1.14.0, with LZ4, with Snappy, with libxml2 2.9.4, with QtMultimedia, without AirPcap, with SBC, without SpanDSP.



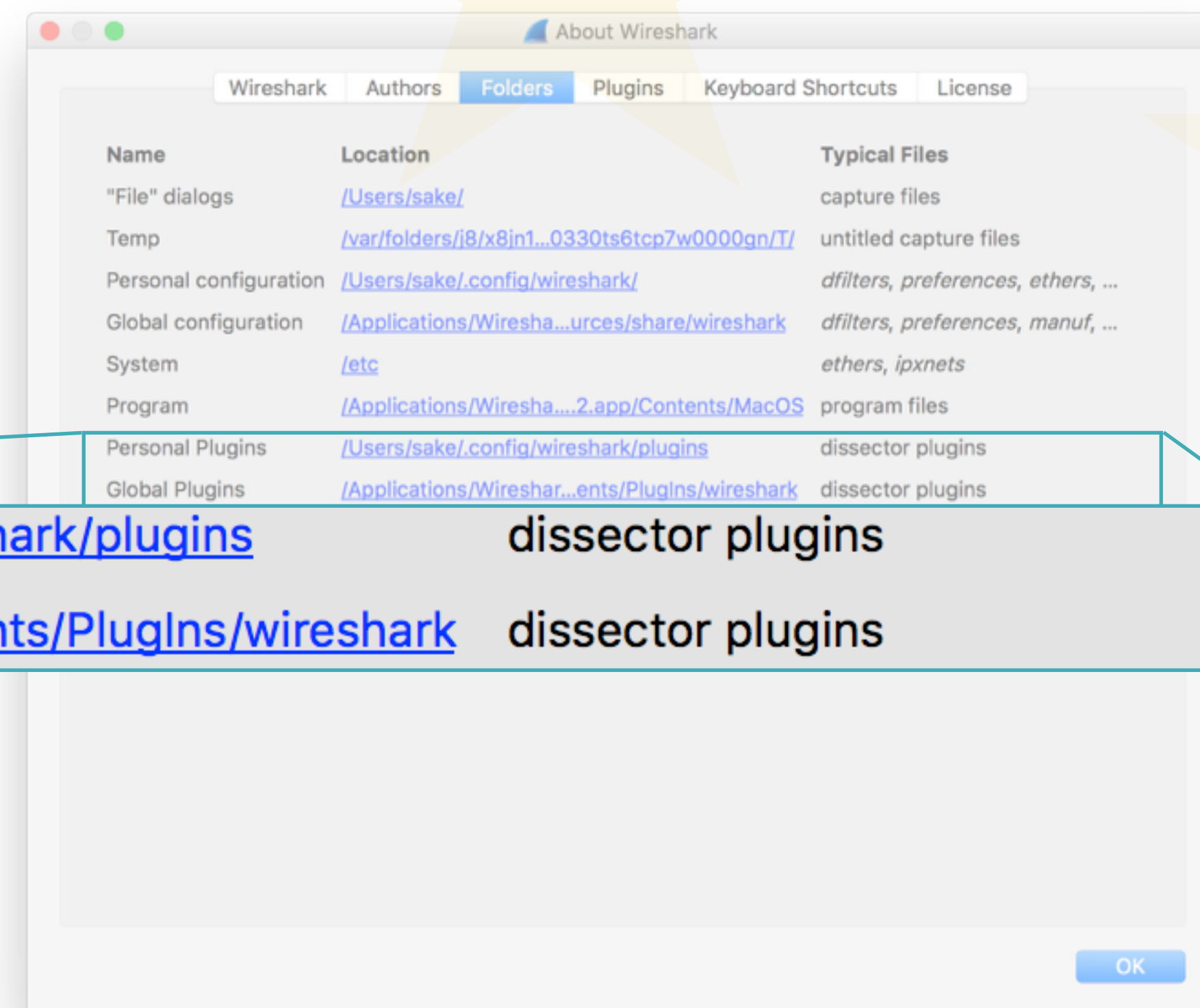


How to execute?



- CLI option: `-X lua_script:<script-file>`
- In the plugin directories
 - ▶ Need extension `.lua`

| | | |
|------------------|---|-------------------|
| Personal Plugins | /Users/sake/.config/wireshark/plugins | dissector plugins |
| Global Plugins | /Applications/Wireshar...ents/Plugins/wireshark | dissector plugins |





Hello World!



```
sake@MacSake:~$ cat ~/.config/wireshark/plugins/hello.lua
-- hello.lua
-- Lua's implementation of D. Ritchie's hello world program.
print("hello world!")
sake@MacSake:~$
```

```
sake@MacSake:~$ tshark -r icmp.pcap -c 1
  1   0.000000  172.16.0.34 → 8.8.8.8      ICMP 98 Echo (ping) request  id=0xfa0e, seq=0/0, ttl=64
sake@MacSake:~$
```

```
sake@MacSake:~$ tshark -r icmp.pcap -c 1 -X lua_script:hello.lua
hello world!
  1   0.000000  172.16.0.34 → 8.8.8.8      ICMP 98 Echo (ping) request  id=0xfa0e, seq=0/0, ttl=64
sake@MacSake:~$
```

```
sake@MacSake:~$ mv hello.lua ~/.config/wireshark/plugins/
sake@MacSake:~$ tshark -r icmp.pcap -c 1
hello world!
  1   0.000000  172.16.0.34 → 8.8.8.8      ICMP 98 Echo (ping) request  id=0xfa0e, seq=0/0, ttl=64
sake@MacSake:~$
```

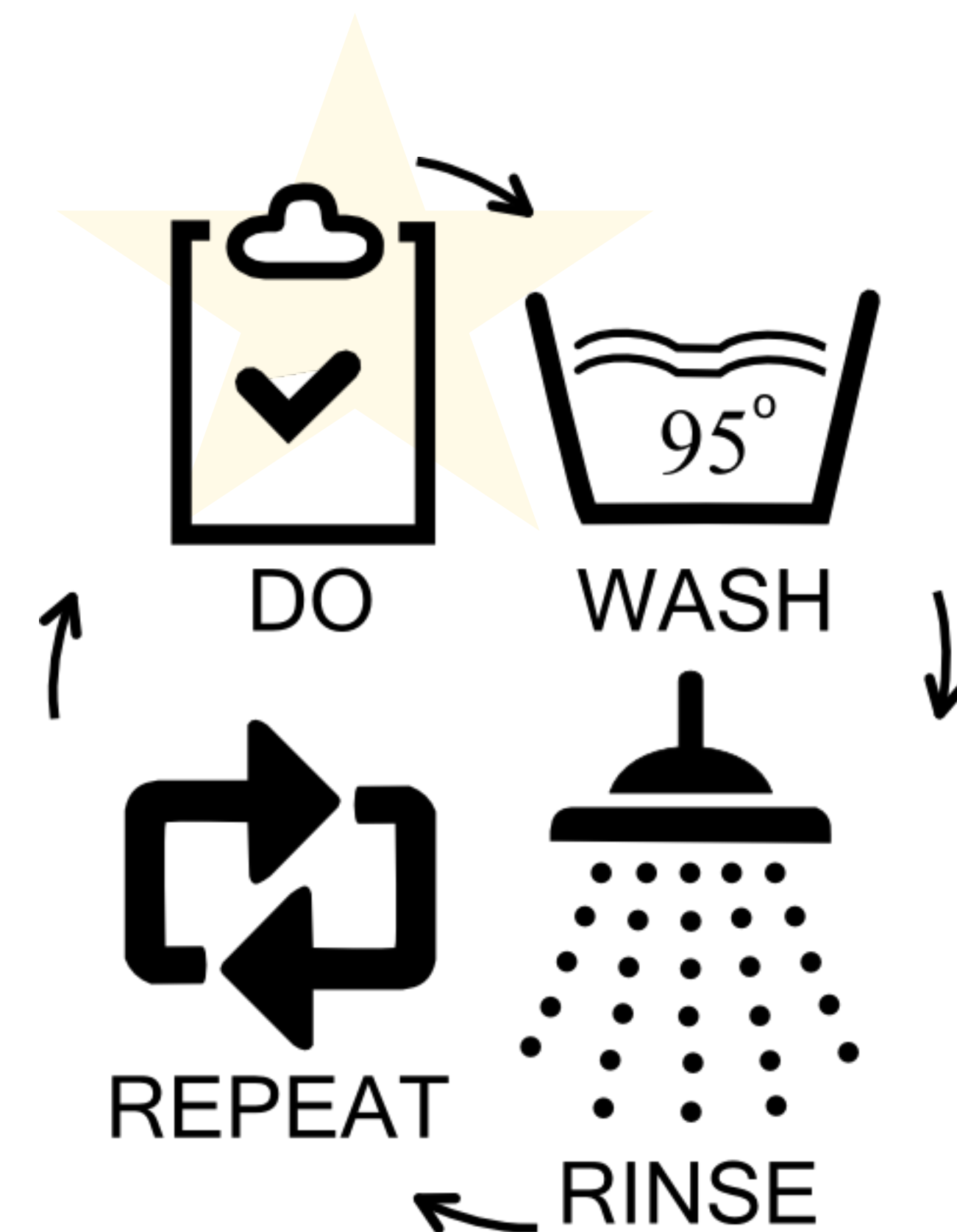




Writing: trade.lua



- Write a basic LUA dissector
- Get a protocol description
- Add some reverse engineering
 - ▶ Interpret packet data
 - ▶ Write code
 - ▶ Test
 - ▶ Rinse
 - ▶ Repeat





PDU format



- Each packet has a 5 byte header:
- 1: sender
 - ▶ List of module ID's provided
- 2-3: total size including header
- 4-5: receiver
 - ▶ In case of control packets: The receiver will be ff:ff
- The header is followed by a keyword and a null termination
- Optional data





Examples



- Keepalive packets from module (DocumentSystem) to multiplexer contain the following bytes:
 - ▶ 25:00:0a:ff:ff:50:69:6e:67:00
- The Multiplexer replies with:
 - ▶ 01:00:0a:ff:ff:50:6f:6e:67:00
- Roundtrip packets from the Multiplexer to module(DocumentSystem) contain the following bytes:
 - ▶ 01:00:17:ff:ff:52:6f:75:6e:64:74:72:69:70:00:07:1b:88:aa:db:46:b6:0c
- The module replies with:
 - ▶ 25:00:17:ff:ff:52:6f:75:6e:64:74:72:69:70:00:07:1b:88:aa:db:46:b6:0c





Example Drilldown



| | |
|----------------------------|-----------------------------------|
| 25 | Module ID |
| 00:17 | Size (Big endian) |
| ff:ff | Receiver |
| 52:6f:75:6e:64:74:72:69:70 | Keyword: "Roundtrip" (ASCII) |
| 00 | Keyword termination |
| 07 | Size of following data |
| 1b:88:aa:db:46:b6:0c | Microsecond count (Little endian) |





01.lua: Register the dissector



```
do
  local trade = Proto("trade", "TRADE");

  function trade.dissector(tvb, pinfo, tree)
  end

  local tcp_encap_table = DissectorTable.get("tcp.port")
  tcp_encap_table:add(2107, trade)
end
```





Result of 01.lua



The screenshot shows a Wireshark window titled 'anonymized.pcap'. The main pane displays a list of 13 network packets. The first packet is selected, and its details pane is expanded to show the following structure:

- Frame 1: 64 bytes on wire (512 bits), 64 bytes captured (512 bits)
- Ethernet II, Src: 02:00:00:00:00:01 (02:00:00:00:00:01), Dst: 02:00:00:00:00:02 (02:00:00:00:00:02)
- Internet Protocol Version 4, Src: 10.0.0.1, Dst: 192.168.0.1
- Transmission Control Protocol, Src Port: 1181, Dst Port: 2107, Seq: 1, Ack: 1, Len: 10

The packet bytes pane shows the following hex and ASCII representation:

```
0000 02 00 00 00 00 02 02 00 00 00 00 01 08 00 45 00 .....
0010 00 32 2c 8e 40 00 7f 06 04 8e 0a 00 00 01 c0 a8 .2,.@...
0020 00 01 04 9d 08 3b dd 42 5c 54 21 66 4f 0f 50 18 .....;B
0030 01 ff 46 76 00 00 0b 00 0a ff ff 50 69 6e 67 00 ..Fv....
```

trade

| No. | Time | Delta |
|-----|------|-------|
|-----|------|-------|

Packets: 33471 · Displayed: 33471 (100.0%) · Load time: 0:0.344 Profile: sf2017eu-dummy



02.lua: Update the protocol column



```
do
  local trade = Proto("trade", "TRADE");

  function trade.dissector(tvb, pinfo, tree)
    -- info("Entering dissector for frame " .. pinfo.number)
    pinfo.cols.protocol = "TRADE"
  end

  local tcp_encap_table = DissectorTable.get("tcp.port")
  tcp_encap_table:add(2107, trade)
end
```





Result of 02.lua



anonymized.pcap

Apply a display filter ... <#>

| No. | Time | Delta | Source | Destination | Protocol | Length | Info |
|-----|-----------------|----------|-------------|-------------|----------|--------|--|
| 1 | 09:10:04.360322 | 0.000000 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 1181 → 2107 [PSH, ACK] Seq=1 Ack=1 Win=130816 Len=10 |
| 2 | 09:10:04.360410 | 0.000088 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 1184 → 2107 [PSH, ACK] Seq=1 Ack=1 Win=129792 Len=10 |
| 3 | 09:10:04.360413 | 0.000003 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 1553 → 2107 [PSH, ACK] Seq=1 Ack=1 Win=130816 Len=10 |
| 4 | 09:10:04.562429 | 0.202016 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 → 1184 [ACK] Seq=1 Ack=11 Win=130816 Len=0 |
| 5 | 09:10:04.562433 | 0.000004 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 → 1553 [ACK] Seq=1 Ack=11 Win=130560 Len=0 |
| 6 | 09:10:04.562436 | 0.000003 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 → 1181 [ACK] Seq=1 Ack=11 Win=1335552 Len=0 |
| 7 | 09:10:04.609538 | 0.047102 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 1536 → 2107 [PSH, ACK] Seq=1 Ack=1 Win=129536 Len=10 |
| 8 | 09:10:04.609921 | 0.000383 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 1177 → 2107 [PSH, ACK] Seq=1 Ack=1 Win=130560 Len=10 |
| 9 | 09:10:04.609974 | 0.000053 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 1192 → 2107 [PSH, ACK] Seq=1 Ack=1 Win=130304 Len=10 |
| 10 | 09:10:04.610060 | 0.000086 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 1200 → 2107 [PSH, ACK] Seq=1 Ack=1 Win=130560 Len=10 |
| 11 | 09:10:04.610172 | 0.000112 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 1203 → 2107 [PSH, ACK] Seq=1 Ack=1 Win=130560 Len=10 |
| 12 | 09:10:04.610245 | 0.000073 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 1546 → 2107 [PSH, ACK] Seq=1 Ack=1 Win=129792 Len=10 |
| 13 | 09:10:04.735016 | 0.124771 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 1188 → 2107 [PSH, ACK] Seq=1 Ack=1 Win=130816 Len=10 |

▶ Frame 8: 64 bytes on wire (512 bits), 64 bytes captured (512 bits)

▶ Ethernet II, Src: 02:00:00:00:00:01 (02:00:00:00:00:01), Dst: 02:00:00:00:00:02 (02:00:00:00:00:02)

▶ Internet Protocol Version 4, Src: 10.0.0.1, Dst: 192.168.0.1

▶ Transmission Control Protocol, Src Port: 1177, Dst Port: 2107, Seq: 1, Ack: 1, Len: 10

0000 02 00 00 00 00 02 02 00 00 00 00 01 08 00 45 00
0010 00 32 2c 92 40 00 7f 06 04 8a 0a 00 00 01 c0 a8 .2,.@...
0020 00 01 04 99 08 3b 05 a8 43 07 0c 67 9b 7a 50 18;
0030 01 fe e5 f6 00 00 25 00 0a ff ff 50 69 6e 67 00%.

anonymized

Packets: 33471 · Displayed: 33471 (100.0%) · Load time: 0:0.428 Profile: sf2017eu-dummy



03.lua: Add a subtree



```
do
  local trade = Proto("trade", "TRADE");

  function trade.dissector(tvb, pinfo, tree)
    -- info("Entering dissector for frame " .. pinfo.number)
    pinfo.cols.protocol = "TRADE"

    local subtree

    subtree = tree:add(trade, tvb(0, tvb:len()), "Trade PDU : ")
  end

  local tcp_encap_table = DissectorTable.get("tcp.port")
  tcp_encap_table:add(2107, trade)
end
```





Result of 03.lua



The screenshot shows a network traffic analysis tool window titled "anonymized.pcap". The main pane displays a list of 13 packets. The first packet is highlighted in blue. Below the list, a detailed view of the first packet is shown, including its frame size, Ethernet II header, Internet Protocol Version 4 header, and Transmission Control Protocol header. A blue arrow points to the "Trade PDU" field in the detailed view.

| No. | Time | Delta | Source | Destination | Protocol | Length | Info |
|-----|-----------------|----------|-------------|-------------|----------|--------|--|
| 1 | 09:10:04,360322 | 0.000000 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 1181 → 2107 [PSH, ACK] Seq=1 Ack=1 Win=130816 Len=10 |
| 2 | 09:10:04,360410 | 0.000088 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 1184 → 2107 [PSH, ACK] Seq=1 Ack=1 Win=129792 Len=10 |
| 3 | 09:10:04,360413 | 0.000003 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 1553 → 2107 [PSH, ACK] Seq=1 Ack=1 Win=130816 Len=10 |
| 4 | 09:10:04,562429 | 0.202016 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 → 1184 [ACK] Seq=1 Ack=11 Win=130816 Len=0 |
| 5 | 09:10:04,562433 | 0.000004 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 → 1553 [ACK] Seq=1 Ack=11 Win=130560 Len=0 |
| 6 | 09:10:04,562436 | 0.000003 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 → 1181 [ACK] Seq=1 Ack=11 Win=1335552 Len=0 |
| 7 | 09:10:04,609538 | 0.047102 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 1536 → 2107 [PSH, ACK] Seq=1 Ack=1 Win=129536 Len=10 |
| 8 | 09:10:04,609921 | 0.000383 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 1177 → 2107 [PSH, ACK] Seq=1 Ack=1 Win=130560 Len=10 |
| 9 | 09:10:04,609974 | 0.000053 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 1192 → 2107 [PSH, ACK] Seq=1 Ack=1 Win=130304 Len=10 |
| 10 | 09:10:04,610060 | 0.000086 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 1200 → 2107 [PSH, ACK] Seq=1 Ack=1 Win=130560 Len=10 |
| 11 | 09:10:04,610172 | 0.000112 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 1203 → 2107 [PSH, ACK] Seq=1 Ack=1 Win=130560 Len=10 |
| 12 | 09:10:04,610245 | 0.000073 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 1546 → 2107 [PSH, ACK] Seq=1 Ack=1 Win=129792 Len=10 |
| 13 | 09:10:04,735016 | 0.124771 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 1188 → 2107 [PSH, ACK] Seq=1 Ack=1 Win=130816 Len=10 |

Frame 1: 64 bytes on wire (512 bits), 64 bytes captured (512 bits)
Ethernet II, Src: 02:00:00:00:00:01 (02:00:00:00:00:01), Dst: 02:00:00:00:00:02 (02:00:00:00:00:02)
Internet Protocol Version 4, Src: 10.0.0.1, Dst: 192.168.0.1
Transmission Control Protocol, Src Port: 1181, Dst Port: 2107, Seq: 1, Ack: 1, Len: 10
Trade PDU :





04.lua: Add a field



```
sake@MacSake:~$ diff -b 03.lua 04.lua
2a3,5
>     local f = trade.fields
>
>     f.sender      = ProtoField.uint8("trade.sender", "Sender", base.DEC)
9a13,20
>         local offset = 0
>         local info = ""
>
>         subtree = tree:add(trade, tvb(offset, tvb:len()), "Trade PDU : ")
>
>         -- Add sender
>         subtree:add(f.sender, tvb(offset, 1))
>         offset = offset + 1
11c22,23
<         subtree = tree:add(trade, tvb(0, tvb:len()), "Trade PDU : ")
---
>         -- Tell the calling dissector how many bytes we dissected
>         return tvb:len()
sake@MacSake:~$
```





Result of 04.lua



anonymized.pcap

Apply a display filter ... <⌘/> Expression...

| No. | Time | Delta | Source | Destination | Protocol | Length | Info |
|-----|-----------------|----------|-------------|-------------|----------|--------|--|
| 1 | 09:10:04,360322 | 0.000000 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 1181 → 2107 [PSH, ACK] Seq=1 Ack=1 Win=130816 Len=10 |
| 2 | 09:10:04,360410 | 0.000088 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 1184 → 2107 [PSH, ACK] Seq=1 Ack=1 Win=129792 Len=10 |
| 3 | 09:10:04,360413 | 0.000003 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 1553 → 2107 [PSH, ACK] Seq=1 Ack=1 Win=130816 Len=10 |
| 4 | 09:10:04,562429 | 0.202016 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 → 1184 [ACK] Seq=1 Ack=11 Win=130816 Len=0 |
| 5 | 09:10:04,562433 | 0.000004 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 → 1553 [ACK] Seq=1 Ack=11 Win=130560 Len=0 |
| 6 | 09:10:04,562436 | 0.000003 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 → 1181 [ACK] Seq=1 Ack=11 Win=1335552 Len=0 |
| 7 | 09:10:04,609538 | 0.047102 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 1536 → 2107 [PSH, ACK] Seq=1 Ack=1 Win=129536 Len=10 |
| 8 | 09:10:04,609921 | 0.000383 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 1177 → 2107 [PSH, ACK] Seq=1 Ack=1 Win=130560 Len=10 |
| 9 | 09:10:04,609974 | 0.000053 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 1192 → 2107 [PSH, ACK] Seq=1 Ack=1 Win=130304 Len=10 |
| 10 | 09:10:04,610060 | 0.000086 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 1200 → 2107 [PSH, ACK] Seq=1 Ack=1 Win=130560 Len=10 |
| 11 | 09:10:04,610172 | 0.000112 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 1203 → 2107 [PSH, ACK] Seq=1 Ack=1 Win=130560 Len=10 |
| 12 | 09:10:04,610245 | 0.000073 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 1546 → 2107 [PSH, ACK] Seq=1 Ack=1 Win=129792 Len=10 |
| 13 | 09:10:04,735016 | 0.124771 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 1188 → 2107 [PSH, ACK] Seq=1 Ack=1 Win=130816 Len=10 |

▶ Frame 1: 64 bytes on wire (512 bits), 64 bytes captured (512 bits)

▶ Ethernet II, Src: 02:00:00:00:00:01 (02:00:00:00:00:01), Dst: 02:00:00:00:00:02 (02:00:00:00:00:02)

▶ Internet Protocol Version 4, Src: 10.0.0.1, Dst: 192.168.0.1

▶ Transmission Control Protocol, Src Port: 1181, Dst Port: 2107, Seq: 1, Ack: 1, Len: 10

▼ Trade PDU :

Sender: 11

0000 02 00 00 00 00 02 02 00 00 00 00 01 08 00 45 00
0010 00 32 2c 8e 40 00 7f 06 04 8e 0a 00 00 01 c0 a8 .2,.@...
0020 00 01 04 9d 08 3b dd 42 5c 54 21 66 4f 0f 50 18;B
0030 01 ff 46 76 00 00 0b 00 0a ff ff 50 69 6e 67 00 ..Fv...

Sender (trade.sender), 1 byte

Packets: 33471 · Displayed: 33471 (100.0%) · Load time: 0:0.617 Profile: sf2017eu-dummy



05.lua: Fill the info column



```
-- Add sender
local sender = tvb(offset,1):uint()
info = tostring(sender)

subtree:add(f.sender,tvb(offset,1))
offset = offset + 1

subtree:append_text(info)
pinfo.cols.info = info
```





Result of 05.lua



The screenshot shows a Wireshark interface with a packet list and a packet details pane. The packet list shows 13 packets, with packet 13 selected. The details pane shows the structure of the selected packet, including Ethernet II, Internet Protocol Version 4, and Transmission Control Protocol. A custom dissector, 'Trade PDU', is shown with a sender value of 13. Two blue arrows point from the 'Trade PDU : 13' entry in the details pane to the 'Info' column of packet 13 in the packet list and to the 'Info' column of packet 2 in the packet list.

| No. | Time | Delta | Source | Destination | Protocol | Length | Info |
|-----|-----------------|----------|-------------|-------------|----------|--------|--|
| 1 | 09:10:04,360322 | 0.000000 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 11 |
| 2 | 09:10:04,360410 | 0.000088 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 13 |
| 3 | 09:10:04,360413 | 0.000003 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 99 |
| 4 | 09:10:04,562429 | 0.202016 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 → 1184 [ACK] Seq=1 Ack=11 Win=130816 Len=0 |
| 5 | 09:10:04,562433 | 0.000004 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 → 1553 [ACK] Seq=1 Ack=11 Win=130560 Len=0 |
| 6 | 09:10:04,562436 | 0.000003 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 → 1181 [ACK] Seq=1 Ack=11 Win=1335552 Len=0 |
| 7 | 09:10:04,609538 | 0.047102 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 20 |
| 8 | 09:10:04,609921 | 0.000383 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 37 |
| 9 | 09:10:04,609974 | 0.000053 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 8 |
| 10 | 09:10:04,610060 | 0.000086 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 21 |
| 11 | 09:10:04,610172 | 0.000112 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 57 |
| 12 | 09:10:04,610245 | 0.000073 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 93 |
| 13 | 09:10:04,735016 | 0.124771 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | 14 |

▼ Trade PDU : 13
Sender: 13

0000 02 00 00 00 00 02 02 00 00 00 00 01 08 00 45 00
0010 00 32 2c 8f 40 00 7f 06 04 8d 0a 00 00 01 c0 a8 .2,.@...
0020 00 01 04 a0 08 3b 7c fb a4 98 96 8d e0 61 50 18|.
0030 01 fb 56 00 00 00 0d 00 0a ff ff 50 69 6e 67 00 ..V.....

TRADE (trade), 10 bytes

Packets: 33471 · Displayed: 33471 (100.0%) · Load time: 0:0.604 Profile: sf2017eu-dummy





06.lua: Interpret field values



```
local modules = {
  [1]   = 'Multiplexer',
  [6]   = 'CarpenterUpdate',
  [7]   = 'Tradesite',
  [. . . . .]
  [93]  = 'Zep',
  [99]  = 'Scripting'
}

f.sender      = ProtoField.uint8("trade.sender", "Sender", base.DEC, modules)

-- Add sender
local sender = tvb(offset, 1):uint()
local mod = modules[sender]
assert( mod ~= nil, "Unknown Sender!")
info = mod
```





Result of 06.lua



anonymized.pcap

Apply a display filter ... <#%/>

| No. | Time | Delta | Source | Destination | Protocol | Length | Info |
|-----|-----------------|----------|-------------|-------------|----------|--------|--|
| 1 | 09:10:04,360322 | 0.000000 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | BackOffice |
| 2 | 09:10:04,360410 | 0.000088 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | MarketConfiguration |
| 3 | 09:10:04,360413 | 0.000003 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Scripting |
| 4 | 09:10:04,562429 | 0.202016 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 → 1184 [ACK] Seq=1 Ack=11 Win=130816 Len=0 |
| 5 | 09:10:04,562433 | 0.000004 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 → 1553 [ACK] Seq=1 Ack=11 Win=130560 Len=0 |
| 6 | 09:10:04,562436 | 0.000003 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 → 1181 [ACK] Seq=1 Ack=11 Win=1335552 Len=0 |
| 7 | 09:10:04,609538 | 0.047102 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | TranquilizerUpdate |
| 8 | 09:10:04,609921 | 0.000383 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | DocumentSystem |
| 9 | 09:10:04,609974 | 0.000053 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Settlement |
| 10 | 09:10:04,610060 | 0.000086 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Zap |
| 11 | 09:10:04,610172 | 0.000112 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Razzamatazz |
| 12 | 09:10:04,610245 | 0.000073 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Zep |
| 13 | 09:10:04,735016 | 0.124771 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | MarginManagement |

Frame 1: 64 bytes on wire (512 bits), 64 bytes captured (512 bits)

Ethernet II, Src: 02:00:00:00:00:01 (02:00:00:00:00:01), Dst: 02:00:00:00:00:02 (02:00:00:00:00:02)

Internet Protocol Version 4, Src: 10.0.0.1, Dst: 192.168.0.1

Transmission Control Protocol, Src Port: 1181, Dst Port: 2107, Seq: 1, Ack: 1, Len: 64

Trade PDU : BackOffice

Sender: BackOffice (11)

0000 02 00 00 00 00 02 02 00 00 00 00 01 08 00 45 00
0010 00 32 2c 8e 40 00 7f 06 04 8e 0a 00 00 01 c0 a8 .2,..@...
0020 00 01 04 9d 08 3b dd 42 5c 54 21 66 4f 0f 50 18;.B
0030 01 ff 46 76 00 00 0b 00 0a ff ff 50 69 6e 67 00 ..Fv...B

Sender (trade.sender), 1 byte

Packets: 33471 · Displayed: 33471 (100.0%) · Load time: 0:0.642 Profile: sf2017eu-dummy





07.lua: Add more fields



```
sake@MacSake:~/Dropbox/sharkfest/2017eu/anonymized$ diff 06.lua 07.lua
23a24,25
>     f.len          = ProtoField.uint16("trade.length", "Length", base.DEC)
>     f.receiver     = ProtoField.uint16("trade.receiver", "Receiver", base.DEC)
44a47,57
>         -- Add length
>         subtree:add(f.len, tvb(offset, 2))
>         offset = offset + 2
>
>         -- Add receiver
>         local receiver = tvb(offset, 2):uint()
>         info = info .. " -> " .. receiver
>
>         subtree:add(f.receiver, tvb(offset, 2))
>         offset = offset + 2
>
sake@MacSake:~/Dropbox/sharkfest/2017eu/anonymized$
```





Result of 07.lua



The screenshot shows a Wireshark interface with a list of 13 packets. The selected packet (No. 1) is expanded to show its details. A blue arrow points to the 'Receiver: 65535' field in the Trade PDU details pane.

| No. | Time | Delta | Source | Destination | Protocol | Length | Info |
|-----|-----------------|----------|-------------|-------------|----------|--------|---|
| 1 | 09:10:04,360322 | 0.000000 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | BackOffice -> 65535 |
| 2 | 09:10:04,360410 | 0.000088 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | MarketConfiguration -> 65535 |
| 3 | 09:10:04,360413 | 0.000003 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Scripting -> 65535 |
| 4 | 09:10:04,562429 | 0.202016 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 -> 1184 [ACK] Seq=1 Ack=11 Win=130816 Len=0 |
| 5 | 09:10:04,562433 | 0.000004 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 -> 1553 [ACK] Seq=1 Ack=11 Win=130560 Len=0 |
| 6 | 09:10:04,562436 | 0.000003 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 -> 1181 [ACK] Seq=1 Ack=11 Win=1335552 Len=0 |
| 7 | 09:10:04,609538 | 0.047102 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | TranquilizerUpdate -> 65535 |
| 8 | 09:10:04,609921 | 0.000383 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | DocumentSystem -> 65535 |
| 9 | 09:10:04,609974 | 0.000053 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Settlement -> 65535 |
| 10 | 09:10:04,610060 | 0.000086 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Zap -> 65535 |
| 11 | 09:10:04,610172 | 0.000112 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Razzamatazz -> 65535 |
| 12 | 09:10:04,610245 | 0.000073 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Zep -> 65535 |
| 13 | 09:10:04,735016 | 0.124771 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | MarginManagement -> 65535 |

Frame 1: 64 bytes on wire (512 bits), 64 bytes captured (512 bits)
Ethernet II, Src: 02:00:00:00:00:01 (02:00:00:00:00:01), Dst: 02:00:00:00:00:02 (02:00:00:00:00:02)
Internet Protocol Version 4, Src: 10.0.0.1, Dst: 192.168.0.1
Transmission Control Protocol, Src Port: 1181, Dst Port: 2107, Seq: 1, Ack: 1, Len: 64
Trade PDU : BackOffice -> 65535
Sender: BackOffice (11)
Length: 10
Receiver: 65535

0000 02 00 00 00 00 02 02 00 00 00 00 01 08 00 45 00
0010 00 32 2c 8e 40 00 7f 06 04 8e 0a 00 00 01 c0 a8 .2,.@...
0020 00 01 04 9d 08 3b dd 42 5c 54 21 66 4f 0f 50 18;B
0030 01 ff 46 76 00 00 0b 00 0a ff ff 50 69 6e 67 00 ..Fv....

Receiver (trade.receiver), 2 bytes Packets: 33471 · Displayed: 33471 (100.0%) · Load time: 0:0.696 Profile: sf2017eu-dummy



08.lua: Differentiate type of packets



```
sake@MacSake:~/Dropbox/sharkfest/2017eu/anonymized$ diff -b 07.lua 08.lua
36,38d35
<     subtree = tree:add(trade,tvb(offset,tvb:len()),"Trade PDU : ")
<
<     -- Add sender
43a41,49
>     local receiver = tvb(offset+3,2):uint()
>
>     if receiver == 65535 then
>         subtree = tree:add(trade,tvb(offset,tvb:len()),"Trade Control PDU : ")
>     else
>         subtree = tree:add(trade,tvb(offset,tvb:len()),"Trade Data PDU : ")
>     end
>
>     -- Add sender
52,55c58
<     local receiver = tvb(offset,2):uint()
<     info = info .. " -> " .. receiver
<
<     subtree:add(f.receiver,tvb(offset,2))
---
>     local t = subtree:add(f.receiver,tvb(offset,2))
57a61,69
>     if receiver == 65535 then
>         -- Dissect control PDU
>         t:append_text(" (Control)")
>         info = info .. " -> Control"
>     else
>         -- Dissect data PDU
>         info = info .. " -> " .. receiver
>     end
>
>
sake@MacSake:~/Dropbox/sharkfest/2017eu/anonymized$
```





Result of 08.lua



anonymized.pcap

Apply a display filter ... <⌘/> Expression...

| No. | Time | Delta | Source | Destination | Protocol | Length | Info |
|-----|-----------------|----------|-------------|-------------|----------|--------|---|
| 1 | 09:10:04,360322 | 0.000000 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | BackOffice -> Control |
| 2 | 09:10:04,360410 | 0.000088 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | MarketConfiguration -> Control |
| 3 | 09:10:04,360413 | 0.000003 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Scripting -> Control |
| 4 | 09:10:04,562429 | 0.202016 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 -> 1181 [ACK] Seq=1 Ack=11 Win=130816 Len=0 |
| 5 | 09:10:04,562433 | 0.000004 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 -> 1153 [ACK] Seq=1 Ack=11 Win=130560 Len=0 |
| 6 | 09:10:04,562436 | 0.000003 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 -> 1181 [ACK] Seq=1 Ack=11 Win=1335552 Len=0 |
| 7 | 09:10:04,609538 | 0.047102 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | TranquilizerUpdate -> Control |
| 8 | 09:10:04,609921 | 0.000383 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | DocumentSystem -> Control |
| 9 | 09:10:04,609974 | 0.000053 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Settlement -> Control |
| 10 | 09:10:04,610060 | 0.000086 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Zap -> Control |
| 11 | 09:10:04,610172 | 0.000112 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Razzamatazz -> Control |
| 12 | 09:10:04,610245 | 0.000073 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Zep -> Control |
| 13 | 09:10:04,735016 | 0.124771 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | MarginManagement -> Control |

Frame 1: 64 bytes on wire (512 bits), 64 bytes captured (512 bits)

Ethernet II, Src: 02:00:00:00:00:01 (02:00:00:00:00:01), Dst: 02:00:00:00:00:02 (02:00:00:00:00:02)

Internet Protocol Version 4, Src: 10.0.0.1, Dst: 192.168.0.1

Transmission Control Protocol, Src Port: 1181, Dst Port: 2107, Seq: 1, Ack: 1, Len: 64

Trade Control PDU : BackOffice -> Control

Sender: BackOffice (11)

Length: 10

Receiver: 65535 (Control)

0000 02 00 00 00 00 02 02 00 00 00 00 01 08 00 45 00
0010 00 32 2c 8e 40 00 7f 06 04 8e 0a 00 00 01 c0 a8 .2,.@...
0020 00 01 04 9d 08 3b dd 42 5c 54 21 66 4f 0f 50 18;.B
0030 01 ff 46 76 00 00 0b 00 0a ff ff 50 69 6e 67 00 ..Fv....

TRADE (trade), 10 bytes

Packets: 33471 · Displayed: 33471 (100.0%) · Load time: 0:0.602 Profile: sf2017eu-dummy





Multiple PDU's in one packet?



The screenshot shows a Wireshark interface with a packet capture of 'anonymized.pcap'. The packet list shows a single packet (No. 57) of length 522 bytes, identified as 'Multiplexer -> Control'. The packet details pane shows the following structure:

- Frame 57: 522 bytes on wire (4176 bits), 522 bytes captured (4176 bits) on interface 0
- Ethernet II, Src: 02:00:00:00:00:02 (02:00:00:00:00:02), Dst: 02:00:00:00:00:01 (02:00:00:00:00:01)
- Internet Protocol Version 4, Src: 192.168.0.1, Dst: 10.0.0.1
- Transmission Control Protocol, Src Port: 2107, Dst Port: 1170, Seq: 1188, Win: 0, Len: 0
- Trade Control PDU : Multiplexer -> Control
 - Sender: Multiplexer (1)
 - Length: 47
 - Receiver: 65535 (Control)

The packet bytes pane shows the raw data of the packet, with a blue arrow pointing to the first 47 bytes, which correspond to the Trade Control PDU. The remaining bytes of the packet are also visible in the pane.

???

#





09.lua: Support multiple PDU's



```
repeat
  -- Add sender
  subtree:add(f.sender, tvb(offset, 1))
  offset = offset + 1
  -- Add length
  -- Add receiver
  -- Add optional data
until offset >= tvb:len() or count >= 100

if count == 1 then
  pinfo.cols.info = info
else
  pinfo.cols.info = count .. " Trade PDUs"
end
```





09.lua



```
repeat
  len = tvb(offset+1,2):uint()
  -- Add sender
  subtree:add(f.sender,tvb(offset,1))
  offset = offset + 1
  len = len - 1
  -- Add length etc.

  if len > 0 then
    -- Add data field for unknown data
    local data = tvb(offset,len):bytes()
    subtree:add(f.data,tvb(offset,data:len()))
    offset = offset + data:len()
    len = len - data:len()
  end
  count = count + 1
until offset >= tvb:len() or count >= 100
```





Result of 09.lua



anonymized.pcap

Apply a display filter ... <#>

| No. | Time | Delta | Source | Destination | Protocol | Length | Info |
|-----|-----------------|----------|-------------|-------------|----------|--------|---|
| 51 | 09:10:05,099621 | 0.000510 | 192.168.0.1 | 10.0.0.1 | TRADE | 64 | Multiplexer -> Control, len=10 |
| 52 | 09:10:05,100100 | 0.000479 | 192.168.0.1 | 10.0.0.1 | TRADE | 64 | Multiplexer -> Control, len=10 |
| 53 | 09:10:05,100575 | 0.000475 | 192.168.0.1 | 10.0.0.1 | TRADE | 64 | Multiplexer -> Control, len=10 |
| 54 | 09:10:05,101058 | 0.000483 | 192.168.0.1 | 10.0.0.1 | TRADE | 64 | Multiplexer -> Control, len=10 |
| 55 | 09:10:05,101537 | 0.000479 | 192.168.0.1 | 10.0.0.1 | TRADE | 64 | Multiplexer -> Control, len=10 |
| 56 | 09:10:05,187740 | 0.086203 | 10.0.0.1 | 192.168.0.1 | TRADE | 65 | Tradesite -> 2642, len=11 |
| 57 | 09:10:05,187978 | 0.000238 | 192.168.0.1 | 10.0.0.1 | TRADE | 522 | 11 Trade PDUs |
| 58 | 09:10:05,189726 | 0.001748 | 10.0.0.1 | 192.168.0.1 | TRADE | 72 | Tradesite -> Control, len=18 |
| 59 | 09:10:05,192796 | 0.003070 | 192.168.0.1 | 10.0.0.1 | TRADE | 95 | Multiplexer -> Control, len=41 |
| 60 | 09:10:05,295692 | 0.102896 | 10.0.0.1 | 192.168.0.1 | TCP | 60 | 1188 -> 2107 [ACK] Seq=29 Ack=29 Win=130816 Len=0 |
| 61 | 09:10:05,299435 | 0.003743 | 10.0.0.1 | 192.168.0.1 | TCP | 60 | 1192 -> 2107 [ACK] Seq=29 Ack=29 Win=130304 Len=0 |
| 62 | 09:10:05,299495 | 0.000060 | 10.0.0.1 | 192.168.0.1 | TCP | 60 | 1184 -> 2107 [ACK] Seq=29 Ack=29 Win=129792 Len=0 |
| 63 | 09:10:05,299498 | 0.000003 | 10.0.0.1 | 192.168.0.1 | TCP | 60 | 1200 -> 2107 [ACK] Seq=29 Ack=29 Win=130560 Len=0 |

Frame 57: 522 bytes on wire (4176 bits), 522 bytes captured (4176 bits) on interface 0

- Ethernet II, Src: 02:00:00:00:00:02 (02:00:00:00:00:02), Dst: 02:00:00:00:00:02
- Internet Protocol Version 4, Src: 192.168.0.1, Dst: 10.0.0.1
- Transmission Control Protocol, Src Port: 2107, Dst Port: 1170, Seq: 1188, Len: 522
- Trade Control PDU : Multiplexer -> Control, len=47
 - Sender: Multiplexer (1)
 - Length: 47
 - Receiver: 65535 (Control)
 - Data: Roundtrip for Module
- Trade Control PDU : Multiplexer -> Control, len=42
- Trade Control PDU : Multiplexer -> Control, len=50
- Trade Control PDU : Multiplexer -> Control, len=46

0030 01 ff 12 18 00 00 01 00 2f ff ff 52 6f 75 6e 64 /..Round
0040 74 72 69 70 20 66 6f 72 20 4d 6f 64 75 6c 65 00 trip for Module.
0050 46 69 6e 61 6e 63 69 61 6c 4d 6f 64 75 6c 65 00 Financia lModule.
0060 04 ac d4 0a 00 01 00 2a ff ff 52 6f 75 6e 64 74* ..Roundt
0070 72 69 70 20 66 6f 72 20 4d 6f 64 75 6c 65 00 42 rip for Module.B
0080 61 63 6b 4f 66 66 69 63 65 00 04 84 fd 07 00 01 ackOffic e.....
0090 00 32 ff ff 52 6f 75 6e 64 74 72 69 70 20 66 6f .2..Roun dtrip fo
00a0 72 20 4d 6f 64 75 6c 65 00 54 72 61 64 65 73 69 r Module .Tradesi
00b0 74 65 53 63 72 69 70 74 69 6e 67 00 04 cb c4 07 teScript ing.....
00c0 00 01 00 2e ff ff 52 6f 75 6e 64 74 72 69 70 20Ro undtrip
00d0 66 6f 72 20 4d 6f 64 75 6c 65 00 44 6f 63 75 6d for Modu le.Docum
00e0 65 6e 74 53 79 73 74 65 6d 00 04 35 93 0b 00 01 entSyste m..5....
00f0 00 2a ff ff 52 6f 75 6e 64 74 72 69 70 20 66 6f ..*..Roun dtrip fo
0100 72 20 4d 6f 64 75 6c 65 00 53 65 74 74 6c 65 6d r Module .Settlem
0110 65 6e 74 00 04 45 21 0b 00 01 00 23 ff ff 52 6f ent..E!. ...#..Ro

Data (trade.data), 42 bytes

Packets: 33471 · Displayed: 33471 (100.0%) · Load time: 0:0.933 Profile: sf2017eu-dummy





PDU's spanning multiple packets



anonymized.pcap

Apply a display filter ... <⌘/> Expression... +

| No. | Time | Delta | Source | Destination | Protocol | Length | Info |
|-----|-----------------|----------|-------------|-------------|----------|--------|--|
| 39 | 09:10:05,083714 | 0.000501 | 10.0.0.1 | 192.168.0.1 | TRADE | 72 | Scripting -> Control, len=18 |
| 40 | 09:10:05,083744 | 0.000030 | 192.168.0.1 | 10.0.0.1 | TRADE | 72 | Multiplexer -> Control, len=18 |
| 41 | 09:10:05,084426 | 0.000682 | 10.0.0.1 | 192.168.0.1 | TRADE | 72 | MarketConfiguration -> Control, len=18 |
| 42 | 09:10:05,095155 | 0.010729 | 192.168.0.1 | 10.0.0.1 | TRADE | 1434 | 2107 -> 1195 [ACK] Seq=1 Ack=1 Win=1467392 Len=1380 |
| 43 | 09:10:05,095160 | 0.000005 | 192.168.0.1 | 10.0.0.1 | TRADE | 1056 | 2107 -> 1195 [PSH, ACK] Seq=1381 Ack=1 Win=1467392 Len=... |
| 44 | 09:10:05,095511 | 0.000351 | 10.0.0.1 | 192.168.0.1 | TCP | 60 | 1195 -> 2107 [ACK] Seq=1 Ack=2383 Win=8142848 Len=0 |
| 45 | 09:10:05,095756 | 0.000245 | 192.168.0.1 | 10.0.0.1 | TRADE | 64 | Multiplexer -> Control, len=10 |
| 46 | 09:10:05,097168 | 0.001412 | 192.168.0.1 | 10.0.0.1 | TRADE | 64 | Multiplexer -> Control, len=10 |
| 47 | 09:10:05,097673 | 0.000505 | 192.168.0.1 | 10.0.0.1 | TRADE | 64 | Multiplexer -> Control, len=10 |
| 48 | 09:10:05,098156 | 0.000483 | 192.168.0.1 | 10.0.0.1 | TRADE | 64 | Multiplexer -> Control, len=10 |
| 49 | 09:10:05,098636 | 0.000480 | 192.168.0.1 | 10.0.0.1 | TRADE | 64 | Multiplexer -> Control, len=10 |
| 50 | 09:10:05,099111 | 0.000475 | 192.168.0.1 | 10.0.0.1 | TRADE | 64 | Multiplexer -> Control, len=10 |
| 51 | 09:10:05,099621 | 0.000510 | 192.168.0.1 | 10.0.0.1 | TRADE | 64 | Multiplexer -> Control, len=10 |

▶ Frame 42: 1434 bytes on wire (11472 bits), 1434 bytes captured (11472 bits)

▶ Ethernet II, Src: 02:00:00:00:00:02 (02:00:00:00:00:02), Dst: 02:00:00:00:00:01 (02:00:00:00:00:01)

▶ Internet Protocol Version 4, Src: 192.168.0.1, Dst: 10.0.0.1

▶ Transmission Control Protocol, Src Port: 2107, Dst Port: 1195, Seq: 1, Ack: 1, Len: 1380

▼ **[Lua Error: [string \"/Users/sake/.config/wireshark/plugins/tmp.lua...\"]:52: Range is out of bounds]**

▼ [Expert Info (Error/Undecoded): Lua Error: [string \"/Users/sake/.config/wireshark/plugins/tmp.lua...\"]:52: Range is out of bounds]

[Lua Error: [string \"/Users/sake/.config/wireshark/plugins/tmp.lua...\"]:52: Range is out of bounds]

[Severity level: Error]

[Group: Undecoded]

Lua Error (_ws.lua.error)

Packets: 33471 · Displayed: 33471 (100.0%) · Load time: 0:0.943 Profile: sf2017eu-dummy





10.lua: Add reassembly



```
local offset = pinfo.desegment_offset or 0
repeat
  if tvb:len() >= offset + 3 then
    -- we have enough bytes for the length field of the PDU
    len = tvb(offset+1,2):uint()
    if offset + len > tvb:len() then
      -- we don't have all the data we need for the full PDU yet
      pinfo.desegment_offset = offset
      pinfo.desegment_len = len - tvb:len()
      return
    end
  else
    -- we don't have all of length field yet
    pinfo.desegment_offset = offset
    pinfo.desegment_len = DESEGMENT_ONE_MORE_SEGMENT
    return
  end
end
```





Result of 10.lua



anonymized.pcap

Apply a display filter ... <⌘/> Expression...

| No. | Time | Delta | Source | Destination | Protocol | Length | Info |
|-----|-----------------|----------|-------------|-------------|----------|--------|--|
| 31 | 09:10:05,081205 | 0.000561 | 192.168.0.1 | 10.0.0.1 | TRADE | 72 | Multiplexer -> Control, len=18 |
| 32 | 09:10:05,081275 | 0.000070 | 10.0.0.1 | 192.168.0.1 | TRADE | 72 | Razzamatazz -> Control, len=18 |
| 33 | 09:10:05,081843 | 0.000568 | 192.168.0.1 | 10.0.0.1 | TRADE | 72 | Multiplexer -> Control, len=18 |
| 34 | 09:10:05,081912 | 0.000069 | 10.0.0.1 | 192.168.0.1 | TRADE | 72 | Zep -> Control, len=18 |
| 35 | 09:10:05,082487 | 0.000575 | 192.168.0.1 | 10.0.0.1 | TRADE | 72 | Multiplexer -> Control, len=18 |
| 36 | 09:10:05,082588 | 0.000101 | 10.0.0.1 | 192.168.0.1 | TRADE | 72 | MarginManagement -> Control, len=18 |
| 37 | 09:10:05,083118 | 0.000530 | 192.168.0.1 | 10.0.0.1 | TRADE | 72 | Multiplexer -> Control, len=18 |
| 38 | 09:10:05,083213 | 0.000095 | 10.0.0.1 | 192.168.0.1 | TRADE | 72 | BackOffice -> Control, len=18 |
| 39 | 09:10:05,083714 | 0.000501 | 10.0.0.1 | 192.168.0.1 | TRADE | 72 | Scripting -> Control, len=18 |
| 40 | 09:10:05,083744 | 0.000030 | 192.168.0.1 | 10.0.0.1 | TRADE | 72 | Multiplexer -> Control, len=18 |
| 41 | 09:10:05,084426 | 0.000682 | 10.0.0.1 | 192.168.0.1 | TRADE | 72 | MarketConfiguration -> Control, len=18 |
| 42 | 09:10:05,095155 | 0.010729 | 192.168.0.1 | 10.0.0.1 | TCP | 1434 | 2107 -> 1195 [ACK] Seq=1 Ack=1 Win=1467392 Len=1380 [TC... |
| 43 | 09:10:05,095160 | 0.000005 | 192.168.0.1 | 10.0.0.1 | TRADE | 1056 | Trader -> 2405, len=2382 |

Frame 43: 1056 bytes on wire (8448 bits), 1056 bytes captured (8448 bits)

- Ethernet II, Src: 02:00:00:00:00:02 (02:00:00:00:00:02), Dst: 02:00:00:00:00:01 (02:00:00:00:00:01)
- Internet Protocol Version 4, Src: 192.168.0.1, Dst: 10.0.0.1
- Transmission Control Protocol, Src Port: 2107, Dst Port: 1195, Seq: 1381, Ack: 1, Len: 1002
- [2 Reassembled TCP Segments (2382 bytes): #42(1380), #43(1002)]
- Trade Data PDU : Trader -> 2405, len=2382
 - Sender: Trader (12)
 - Length: 2382
 - Receiver: 2405
 - Data:

Length (trade.length), 2 bytes

Packets: 33471 · Displayed: 33471 (100.0%) · Load time: 0:0.915 Profile: sf2017eu-dummy





11.lua: Add more fields



```
sake@MacSake:~/Dropbox/sharkfest/2017eu/anonymized$ diff 10.lua 11.lua
25a26
>     f.keyword      = ProtoField.stringz("trade.keyword", "Keyword")
88a90,101
>
>     -- Add keyword
>     local keyword = tvb(offset):stringz()
>     if keyword:len() > 0 then
>         subtree:add(f.keyword, tvb(offset, keyword:len()))
>         info = info .. ", " .. keyword
>         offset = offset + keyword:len() + 1
>         len = len - keyword:len() - 1
>     else
>         -- there was no keyword (first byte was 0x00)
>         subtree:add_expert_info(PI_MALFORMED, PI_ERROR, "There was no
keyword in this control frame")
>     end
sake@MacSake:~/Dropbox/sharkfest/2017eu/anonymized$
```





Result of 11.lua



The screenshot shows a Wireshark interface with a packet list table and a detailed view pane. The packet list table contains 13 entries, all of which are TRADE packets of length 64 bytes, sent from 10.0.0.1 to 192.168.0.1. The detailed view pane shows the structure of the selected packet (No. 1), which is a Trade Control PDU. The PDU structure is as follows:

- Frame 1: 64 bytes on wire (512 bits), 64 bytes captured (512 bits)
- Ethernet II, Src: 02:00:00:00:00:01 (02:00:00:00:00:01), Dst: 02:00:00:00:00:02 (02:00:00:00:00:02)
- Internet Protocol Version 4, Src: 10.0.0.1, Dst: 192.168.0.1
- Transmission Control Protocol, Src Port: 1181, Dst Port: 2107, Seq: 1, Ack: 1, Len: 10
- Trade Control PDU : BackOffice -> Control, Ping, len=10**
 - Sender: BackOffice (11)
 - Length: 10
 - Receiver: 65535 (Control)
 - Keyword: Ping

A blue arrow points to the 'Keyword: Ping' field in the detailed view pane.





12.lua: Add more fields



```
sake@MacSake:~/Dropbox/sharkfest/2017eu/anonymized$ diff -b 11.lua 12.lua
26a27
>     f.ping_clock = ProtoField.uint64("trade.ping_clock","Ping Time")
95c96
<         info = info .. ", " .. keyword
---
>         info = info .. ", "
97a99,114
>         if len > 0 then
>             if keyword == "Ping" or
>                keyword == "Pong" then
>                 -- length (1 byte)
>                 -- time in microseconds (since Jan 1, 1901, see http://en.wikipedia.org/wiki/System\_time)
>                 local clocklen = tvb(offset,1):uint()
>                 subtree:add_le(f.ping_clock,tvb(offset+1,clocklen))
>                 offset = offset + clocklen + 1
>                 len = len - clocklen - 1
>                 info = info .. keyword
>             else
>                 info = info .. keyword .. " (Unknown keyword)"
>             end
>         else
>             info = info .. keyword
>         end
sake@MacSake:~/Dropbox/sharkfest/2017eu/anonymized$
```





Result of 12.lua



The screenshot shows a network traffic analysis tool interface. The top toolbar includes various navigation and search icons. Below the toolbar is a search bar with the text "Apply a display filter ... <#/>". The main area displays a table of network packets with columns for No., Time, Delta, Source, Destination, Protocol, Length, and Info. Packet 22 is highlighted in blue. Below the table, a detailed view of packet 22 is shown, including Ethernet II, Internet Protocol Version 4, and Trade Control PDU information. A blue arrow points to the "Ping Time: 3579412204816491" field in the detailed view. The bottom status bar shows "Ping Time (trade.ping_clock), 7 bytes" and "Packets: 33471 · Displayed: 33471 (100.0%) · Load time: 0:0.895 Profile: sf2017eu-dummy".

| No. | Time | Delta | Source | Destination | Protocol | Length | Info |
|-----|-----------------|----------|-------------|-------------|----------|--------|--|
| 13 | 09:10:04,735016 | 0.124771 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | MarginManagement -> Control, Ping, len=10 |
| 14 | 09:10:04,811972 | 0.076956 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 -> 1200 [ACK] Seq=1 Ack=11 Win=130304 Len=0 |
| 15 | 09:10:04,811975 | 0.000003 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 -> 1192 [ACK] Seq=1 Ack=11 Win=130048 Len=0 |
| 16 | 09:10:04,811978 | 0.000003 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 -> 1536 [ACK] Seq=1 Ack=11 Win=129536 Len=0 |
| 17 | 09:10:04,811981 | 0.000003 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 -> 1203 [ACK] Seq=1 Ack=11 Win=130304 Len=0 |
| 18 | 09:10:04,811984 | 0.000003 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 -> 1546 [ACK] Seq=1 Ack=11 Win=131072 Len=0 |
| 19 | 09:10:04,817329 | 0.005345 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 -> 1177 [ACK] Seq=1 Ack=11 Win=130560 Len=0 |
| 20 | 09:10:04,907771 | 0.090442 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Tradesite -> Control, Ping, len=10 |
| 21 | 09:10:04,936727 | 0.028956 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 -> 1188 [ACK] Seq=1 Ack=11 Win=130816 Len=0 |
| 22 | 09:10:05,078151 | 0.141424 | 192.168.0.1 | 10.0.0.1 | TRADE | 72 | Multiplexer -> Control, Ping, len=18 |
| 23 | 09:10:05,078632 | 0.000481 | 192.168.0.1 | 10.0.0.1 | TRADE | 72 | Multiplexer -> Control, Ping, len=18 |
| 24 | 09:10:05,078839 | 0.000207 | 10.0.0.1 | 192.168.0.1 | TRADE | 72 | TranquilizerUpdate -> Control, Pong, len=18 |
| 25 | 09:10:05,079292 | 0.000453 | 192.168.0.1 | 10.0.0.1 | TRADE | 72 | Multiplexer -> Control, Ping, len=18 |

▶ Frame 22: 72 bytes on wire (576 bits), 72 bytes captured (576 bits)
▶ Ethernet II, Src: 02:00:00:00:00:02 (02:00:00:00:00:02), Dst: 02:00:00:00:00:01 (02:00:00:00:00:01)
▶ Internet Protocol Version 4, Src: 192.168.0.1, Dst: 10.0.0.1
▶ Transmission Control Protocol, Src Port: 2107, Dst Port: 1536, Seq: 1, Ack: 11, Len: 18
▼ Trade Control PDU : Multiplexer -> Control, Ping, len=18
Sender: Multiplexer (1)
Length: 18
Receiver: 65535 (Control)
Keyword: Ping
Ping Time: 3579412204816491





13.lua: Add more fields



```
sake@MacSake:~/Dropbox/sharkfest/2017eu/anonymized$ diff -b 12.lua 13.lua
27a28,29
>     f.rtt_time    = ProtoField.double("trade.rtt_time","Roundtrip Time")
>     f.mod         = ProtoField.stringz("trade.module","Module")
108a111,131
>
>         elseif keyword == "Roundtrip for Module" then
>             -- string with module name
>             -- length (1 byte)
>             -- time in milliseconds
>             local mod = tvb(offset):stringz()
>             subtree:add(f.mod,tvb(offset,mod:len()))
>             offset = offset + mod:len() + 1
>             len = len - mod:len() - 1
>
>             local clocklen = tvb(offset,1):uint()
>             assert(clocklen<=4, "clocklength too big")
>
>             local rtt_time = tvb(offset+1,clocklen):le_uint()
>             local t = subtree:add(f.rtt_time,tvb(offset+1,clocklen),rtt_time/1000)
>             t:append_text(" ms")
>             offset = offset + clocklen + 1
>             len = len - clocklen - 1
>
>             info = info .. "Roundtrip for Module " .. mod .. " is " .. rtt_time/1000 .. " ms"
>
sake@MacSake:~/Dropbox/sharkfest/2017eu/anonymized$
```





Result of 13.lua



The screenshot shows a Wireshark interface with a filter expression `trade.keyword == "Roundtrip for Module"`. The packet list contains several entries, with packet 59 highlighted. The detailed view for packet 59 shows the following structure:

- Frame 59: 95 bytes on wire (760 bits), 95 bytes captured (760 bits)
- Ethernet II, Src: 02:00:00:00:00:02 (02:00:00:00:00:02), Dst: 02:00:00:00:00:01 (02:00:00:00:00:01)
- Internet Protocol Version 4, Src: 192.168.0.1, Dst: 10.0.0.1
- Transmission Control Protocol, Src Port: 2107, Dst Port: 1170, Seq: 479, Ack: 40, Len: 41
- Trade Control PDU : Multiplexer -> Control, Roundtrip for Module Tradesite is 68,107 ms, len=41
 - Sender: Multiplexer (1)
 - Length: 41
 - Receiver: 65535 (Control)
 - Keyword: Roundtrip for Module
 - Module: Tradesite
 - Roundtrip Time: 68,107 ms

Two blue arrows point from the highlighted text in the detailed view back to the corresponding fields in the packet list table above.





14.lua: Add more fields



```
sake@MacSake:~/Dropbox/sharkfest/2017eu/anonymized$ diff -b 13.lua 14.lua
29a30,31
>     f.conn_id      = ProtoField.uint16("trade.conn_id","Connection ID",base.DEC)
>     f.client_ip    = ProtoField.stringz("trade.client_ip","Client IP")
131a134,157
>
>         elseif keyword == "New Connection" then
>             -- 2 bytes connection ID
>             -- string with IP address
>             local conn_id = tvb(offset,2):uint()
>             subtree:add(f.conn_id,tvb(offset,2))
>             offset = offset + 2
>             len = len - 2
>
>             local ip = tvb(offset):stringz()
>             subtree:add(f.client_ip,tvb(offset,ip:len()))
>             offset = offset + ip:len() + 1
>             len = len - ip:len() - 1
>
>             info = info .. "New Connection from " .. ip .. " (id=" .. conn_id .. ")"
>
>         elseif keyword == "Close Connection" then
>             -- 2 bytes connection ID
>             local conn_id = tvb(offset,2):uint()
>             subtree:add(f.conn_id,tvb(offset,2))
>             offset = offset + 2
>             len = len - 2
>
>             info = info .. "Connection closed (id=" .. conn_id .. ")"
>
sake@MacSake:~/Dropbox/sharkfest/2017eu/anonymized$
```





Result of 14.lua



trade.keyword == "New Connection"

| No. | Time | Delta | Source | Destination | Protocol | Length | Info |
|-------|-----------------|-------------|-------------|-------------|----------|--------|---|
| 1149 | 09:13:57,190145 | 0.000000 | 192.168.0.1 | 10.0.0.1 | TRADE | 92 | Multiplexer -> Control, New Connection from 194.134.00... |
| 2222 | 09:15:48,761581 | 111.5714... | 192.168.0.1 | 10.0.0.1 | TRADE | 92 | Multiplexer -> Control, New Connection from 194.134.00... |
| 4009 | 09:17:26,760954 | 97.999373 | 192.168.0.1 | 10.0.0.1 | TRADE | 92 | Multiplexer -> Control, New Connection from 194.134.00... |
| 4257 | 09:17:49,474616 | 22.713662 | 192.168.0.1 | 10.0.0.1 | TRADE | 92 | Multiplexer -> Control, New Connection from 194.134.00... |
| 5989 | 09:19:00,158317 | 70.683701 | 192.168.0.1 | 10.0.0.1 | TRADE | 92 | Multiplexer -> Control, New Connection from 194.134.00... |
| 7869 | 09:20:43,305697 | 103.1473... | 192.168.0.1 | 10.0.0.1 | TRADE | 92 | Multiplexer -> Control, New Connection from 194.134.00... |
| 9823 | 09:24:06,433669 | 203.1279... | 192.168.0.1 | 10.0.0.1 | TRADE | 92 | Multiplexer -> Control, New Connection from 194.134.00... |
| 10846 | 09:25:26,617811 | 80.184142 | 192.168.0.1 | 10.0.0.1 | TRADE | 92 | Multiplexer -> Control, New Connection from 194.134.00... |
| 12074 | 09:26:44,430674 | 77.812863 | 192.168.0.1 | 10.0.0.1 | TRADE | 92 | Multiplexer -> Control, New Connection from 194.134.00... |
| 14962 | 09:27:34,241613 | 49.810939 | 192.168.0.1 | 10.0.0.1 | TRADE | 92 | Multiplexer -> Control, New Connection from 194.134.00... |
| 16326 | 09:28:24,723336 | 50.481723 | 192.168.0.1 | 10.0.0.1 | TRADE | 92 | Multiplexer -> Control, New Connection from 194.134.00... |
| 30388 | 09:30:25,717177 | 120.9938... | 192.168.0.1 | 10.0.0.1 | TRADE | 92 | Multiplexer -> Control, New Connection from 194.134.00... |
| 31558 | 09:32:28,317857 | 122.6006... | 192.168.0.1 | 10.0.0.1 | TRADE | 92 | Multiplexer -> Control, New Connection from 194.134.00... |

Frame 1149: 92 bytes on wire (736 bits), 92 bytes captured (736 bits)

- Ethernet II, Src: 02:00:00:00:00:02 (02:00:00:00:00:02), Dst: 02:00:00:00:00:01 (02:00:00:00:00:01)
- Internet Protocol Version 4, Src: 192.168.0.1, Dst: 10.0.0.1
- Transmission Control Protocol, Src Port: 2107, Dst Port: 1170, Seq: 4871, Ack: 3382, Len: 38
- Trade Control PDU : Multiplexer -> Control, New Connection from 194.134.005.011 (id=2645), len=38
 - Sender: Multiplexer (1)
 - Length: 38
 - Receiver: 65535 (Control)
 - Keyword: New Connection**
 - Connection ID: 2645
 - Client IP: 194.134.005.011

Keyword (trade.keyword), 15 bytes

Packets: 33471 · Displayed: 36 (0.1%) · Load time: 0:0.870 Profile: sf2017eu-dummy





Result of 14.lua



The screenshot shows a Wireshark interface with a filter applied: `trade.keyword == "Close Connection"`. The packet list pane displays several packets, all of which are Trade Control PDUs with the keyword "Close Connection". The detailed view pane shows the structure of the selected packet (No. 8705):

- Frame 8705: 78 bytes on wire (624 bits), 78 bytes captured (624 bits)
- Ethernet II, Src: 02:00:00:00:00:02 (02:00:00:00:00:02), Dst: 02:00:00:00:00:01 (02:00:00:00:00:01)
- Internet Protocol Version 4, Src: 192.168.0.1, Dst: 10.0.0.1
- Transmission Control Protocol, Src Port: 2107, Dst Port: 1181, Seq: 11139, Ack: 2425842, Len: 24
- Trade Control PDU : Multiplexer -> Control, Connection closed (id=2645), len=24
 - Sender: Multiplexer (1)
 - Length: 24
 - Receiver: 65535 (Control)
 - Keyword: Close Connection** (highlighted with a blue arrow)
 - Connection ID: 2645

The bottom status bar indicates: Keyword (trade.keyword), 17 bytes. Packets: 33471 · Displayed: 133 (0.4%) · Load time: 0:0.812 · Profile: sf2017eu-dummy





15.lua: Add more fields



```
sake@MacSake:~/Dropbox/sharkfest/2017eu/anonymized$ diff -b 14.lua 15.lua
31a32
>     f.username    = ProtoField.stringz("trade.username","Username")
157a159,195
>
>         elseif keyword == "Login" then
>             -- string with module (by sender Sylvester (7))
>             -- 2 bytes connection ID
>             -- string with username
>             -- string with IP address
>             -- "read"
>             -- 0x00
>             local mod
>
>             if sender == 7 then
>                 mod = tvb(offset):stringz()
>                 subtree:add(f.mod,tvb(offset,mod:len()))
>                 offset = offset + mod:len() + 1
>                 len = len - mod:len() - 1
>             end
>
>             local conn_id = tvb(offset,2):uint()
>             subtree:add(f.conn_id,tvb(offset,2))
>             offset = offset + 2
>             len = len - 2
>
>             local username = tvb(offset):stringz()
>             subtree:add(f.username,tvb(offset,username:len()))
>             offset = offset + username:len() + 1
>             len = len - username:len() - 1
>
>             local ip = tvb(offset):stringz()
>             subtree:add(f.client_ip,tvb(offset,ip:len()))
>             offset = offset + ip:len() + 1
>             len = len - ip:len() - 1
>
>             if sender == 7 then
>                 info = info .. " User " .. username .. " logged in from " .. ip .. " to " .. mod .. " (id=" .. conn_id .. ")"
>             else
>                 info = info .. " User " .. username .. " logged in from " .. ip .. " (id=" .. conn_id .. ")"
>             end
>         end
>
sake@MacSake:~/Dropbox/sharkfest/2017eu/anonymized$
```





Result of 15.lua



trade.keyword == "Login"

| No. | Time | Delta | Source | Destination | Protocol | Length | Info |
|------|-----------------|-------------|-------------|-------------|----------|--------|--|
| 1245 | 09:14:07,851579 | 0.000000 | 10.0.0.1 | 192.168.0.1 | TRADE | 111 | Tradesite -> Control, User XXXXXX0011 logged in from ... |
| 1246 | 09:14:07,852641 | 0.001062 | 192.168.0.1 | 10.0.0.1 | TRADE | 100 | Multiplexer -> Control, User XXXXXX0011 logged in fro... |
| 1249 | 09:14:08,063517 | 0.210876 | 10.0.0.1 | 192.168.0.1 | TRADE | 714 | 5 Trade PDUs |
| 1250 | 09:14:08,069751 | 0.006234 | 192.168.0.1 | 10.0.0.1 | TRADE | 106 | Multiplexer -> Control, User XXXXXX0011 logged in fro... |
| 1251 | 09:14:08,072339 | 0.002588 | 192.168.0.1 | 10.0.0.1 | TRADE | 100 | Multiplexer -> Control, User XXXXXX0011 logged in fro... |
| 1252 | 09:14:08,074901 | 0.002562 | 192.168.0.1 | 10.0.0.1 | TRADE | 100 | Multiplexer -> Control, User XXXXXX0011 logged in fro... |
| 2364 | 09:16:14,416221 | 126.3413... | 10.0.0.1 | 192.168.0.1 | TRADE | 111 | Tradesite -> Control, User XXXXXX0012 logged in from ... |
| 2365 | 09:16:14,418420 | 0.002199 | 192.168.0.1 | 10.0.0.1 | TRADE | 100 | Multiplexer -> Control, User XXXXXX0012 logged in fro... |
| 2368 | 09:16:14,626496 | 0.208076 | 10.0.0.1 | 192.168.0.1 | TRADE | 714 | 5 Trade PDUs |
| 2369 | 09:16:14,627826 | 0.001330 | 192.168.0.1 | 10.0.0.1 | TRADE | 106 | Multiplexer -> Control, User XXXXXX0012 logged in fro... |
| 2370 | 09:16:14,629067 | 0.001241 | 192.168.0.1 | 10.0.0.1 | TRADE | 100 | Multiplexer -> Control, User XXXXXX0012 logged in fro... |

▶ Frame 1245: 111 bytes on wire (888 bits), 111 bytes captured (888 bits)

▶ Ethernet II, Src: 02:00:00:00:00:01 (02:00:00:00:00:01), Dst: 02:00:00:00:00:02 (02:00:00:00:00:02)

▶ Internet Protocol Version 4, Src: 10.0.0.1, Dst: 192.168.0.1

▶ Transmission Control Protocol, Src Port: 1170, Dst Port: 2107, Seq: 4151, Ack: 5916, Len: 57

▼ Trade Control PDU : Tradesite -> Control, User XXXXXX0011 logged in from 194.134.005.011 to BackOffice (id=2645), len=57

- Sender: Tradesite (7)
- Length: 57
- Receiver: 65535 (Control)
- Keyword: Login
- Module: BackOffice
- Connection ID: 2645
- Username: XXXXXX0011
- Client IP: 194.134.005.011
- Data: Read

anonymized

Packets: 33471 · Displayed: 171 (0.5%) · Load time: 0:0.864 Profile: sf2017eu-dummy





16.lua: Add more fields



```
sake@MacSake:~/Dropbox/sharkfest/2017eu/anonymized$ diff -b 15.lua 16.lua
32a33,35
>     f.module_count  = ProtoField.uint16("trade.module_count","Module Count",base.DEC)
>     f.client_count  = ProtoField.uint16("trade.client_count","Client Count",base.DEC)
>     f.status        = ProtoField.stringz("trade.status","Status")
41c44
<         local offset = pinfo.desegment_offset or 0
---
>         local offset = 0
195a199,219
>                 elseif keyword == "ModuleAndClientCount" then
>                     local module_count = tvb(offset,2):uint()
>                     subtree:add(f.module_count,tvb(offset,2))
>                     offset = offset + 2
>                     len = len - 2
>
>                     local client_count = tvb(offset,2):uint()
>                     subtree:add(f.client_count,tvb(offset,2))
>                     offset = offset + 2
>                     len = len - 2
>
>                     info = info .. keyword .. ", Modules: " .. module_count .. ", Clients: " .. client_count
>
>                 elseif keyword == "Status" then
>                     local status = tvb(offset):stringz()
>                     subtree:add(f.status,tvb(offset,status:len()))
>                     offset = offset + status:len() + 1
>                     len = len - status:len() - 1
>
>                     info = info .. keyword .. ": " .. status
>
sake@MacSake:~/Dropbox/sharkfest/2017eu/anonymized$
```





Result of 16.lua



The screenshot shows a Wireshark interface with a filter expression `trade.keyword == "ModuleAndClientCount"`. The packet list pane displays several packets, with packet 1145 selected. The packet details pane shows the following information:

- Frame 1145: 84 bytes on wire (672 bits), 84 bytes captured (672 bits)
- Ethernet II, Src: 02:00:00:00:00:02 (02:00:00:00:00:02), Dst: 02:00:00:00:00:01 (02:00:00:00:00:01)
- Internet Protocol Version 4, Src: 192.168.0.1, Dst: 10.0.0.1
- Transmission Control Protocol, Src Port: 2107, Dst Port: 1170, Seq: 4841, Ack: 3382, Len: 30
- Trade Control PDU : Multiplexer -> Control, ModuleAndClientCount, Modules: 14, Clients: 73, len=30
 - Sender: Multiplexer (1)
 - Length: 30
 - Receiver: 65535 (Control)
 - Keyword: ModuleAndClientCount**
 - Module Count: 14
 - Client Count: 73

A blue arrow points to the 'Module Count: 14' and 'Client Count: 73' fields in the details pane.

At the bottom of the interface, the status bar shows: Keyword (trade.keyword), 21 bytes | Packets: 33471 · Displayed: 73 (0.2%) · Load time: 0:0.905 | Profile: sf2017eu-dummy





17.lua: Use fields from other layers



```
sake@MacSake:~/Dropbox/sharkfest/2017eu/anonymized$ diff -b 16.lua 17.lua
4a5,6
>     local tcp_seq_f = Field.new("tcp.seq")
>
29a32
>     f.greeting      = ProtoField.stringz("trade.greeting","Server Greeting")
44c47
<         local offset = 0
---
>         local offset = pinfo.desegment_offset or 0
54a58,71
>         -- Handle the server greeting message
>         local tcp_seq = tostring(tcp_seq_f())
>         len = tvb(1,1):uint()
>         if tcp_seq == "1" and (tvb:len() == len + 3) then
>             subtree = tree:add(trade,tvb(0,len),"Trade Server Greeting : ")
>             subtree:add(f.sender,tvb(0,1))
>             subtree:add(f.len,tvb(1,1))
>             subtree:add(f.greeting,tvb(2,len - 2))
>             local mod = tvb(2,len - 2):stringz()
>             subtree:append_text(mod)
>             pinfo.cols.info = "Setting up connection to " .. mod
>             return len
>         end
>
sake@MacSake:~/Dropbox/sharkfest/2017eu/anonymized$
```





Result of 17.lua



| No. | Time | Delta | Source | Destination | Protocol | Length | Info |
|-------|-----------------|-------------|----------|-------------|----------|--------|--|
| 31380 | 09:32:03,344877 | 0.000000 | 10.0.0.1 | 192.168.0.1 | TRADE | 63 | Setting up connection to Trader |
| 32779 | 09:35:32,260063 | 208.9151... | 10.0.0.1 | 192.168.0.1 | TRADE | 73 | Setting up connection to JackhammerUpdate |
| 32781 | 09:35:32,263351 | 0.003288 | 10.0.0.1 | 192.168.0.1 | TRADE | 75 | Setting up connection to TranquilizerUpdate |
| 32788 | 09:35:32,276936 | 0.013585 | 10.0.0.1 | 192.168.0.1 | TRADE | 75 | Setting up connection to TradesiteScripting |
| 32789 | 09:35:32,279921 | 0.002985 | 10.0.0.1 | 192.168.0.1 | TRADE | 71 | Setting up connection to DocumentSystem |
| 32790 | 09:35:32,281271 | 0.001350 | 10.0.0.1 | 192.168.0.1 | TRADE | 67 | Setting up connection to BackOffice |
| 32802 | 09:35:32,307129 | 0.025858 | 10.0.0.1 | 192.168.0.1 | TRADE | 76 | Setting up connection to MarketConfiguration |
| 32804 | 09:35:32,309232 | 0.002103 | 10.0.0.1 | 192.168.0.1 | TRADE | 72 | Setting up connection to FinancialModule |
| 32805 | 09:35:32,311170 | 0.001938 | 10.0.0.1 | 192.168.0.1 | TRADE | 67 | Setting up connection to Settlement |
| 32806 | 09:35:32,313115 | 0.001945 | 10.0.0.1 | 192.168.0.1 | TRADE | 60 | Setting up connection to Zap |
| 32812 | 09:35:32,337094 | 0.023979 | 10.0.0.1 | 192.168.0.1 | TRADE | 69 | Setting up connection to TradesiteZep |

▶ Frame 32790: 67 bytes on wire (536 bits), 67 bytes captured (536 bits) on interface 0
▶ Ethernet II, Src: 02:00:00:00:00:01 (02:00:00:00:00:01), Dst: 02:00:00:00:00:01
▶ Internet Protocol Version 4, Src: 10.0.0.1, Dst: 192.168.0.1
▶ Transmission Control Protocol, Src Port: 1617, Dst Port: 2107, Seq: 1, Len: 67
▼ Trade Server Greeting : BackOffice
Sender: BackOffice (11)
Length: 10
Server Greeting: BackOffice

0000 02 00 00 00 00 02 02 00 00 00 00 01 08 00 45 00E.
0010 00 35 6f 76 40 00 7f 06 c1 a2 0a 00 00 01 c0 a8 .5ov@...
0020 00 01 06 51 08 3b ab da 54 77 33 7e 2d 50 50 18 ...Q.;... Tw3~PP.
0030 02 00 a9 5d 00 00 0b 0a 42 61 63 6b 4f 66 66 69 ...].... BackOffi
0040 63 65 00 ce.





Road taken



- Create a LUA dissector skeleton
- Add fields, fill info column and subtree for the basic PDU header
- Add multi-PDU support
- Add reassembly
- Dissect more parts of the PDU's
- Now back to troubleshooting!!!





Ping-Pong?



anonymized.pcap

(tcp.stream == 11 && (tcp.flags&7 or (trade.keyword in {"Ping" "Pong"})))

| No. | Time | Delta | Source | Destination | Protocol | Length | Roundtrip Time | Info |
|-------|-----------------|-------------|-------------|-------------|----------|--------|----------------|--|
| 12117 | 09:26:56.806849 | 0.001147 | 192.168.0.1 | 10.0.0.1 | TRADE | 64 | | Multiplexer -> Control, Pong, len=10 |
| 12118 | 09:26:56.807115 | 0.000266 | 10.0.0.1 | 192.168.0.1 | TRADE | 72 | | Trader -> Control, Pong, len=18 |
| 14858 | 09:27:26.815959 | 30.008844 | 192.168.0.1 | 10.0.0.1 | TRADE | 72 | | Multiplexer -> Control, Ping, len=18 |
| 14859 | 09:27:26.816078 | 0.000119 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | | Trader -> Control, Ping, len=10 |
| 14861 | 09:27:26.864085 | 0.048007 | 10.0.0.1 | 192.168.0.1 | TRADE | 666 | | 23 Trade PDUs |
| 14869 | 09:27:26.864242 | 0.000157 | 192.168.0.1 | 10.0.0.1 | TRADE | 64 | | Multiplexer -> Control, Pong, len=10 |
| 15890 | 09:27:56.877668 | 30.013426 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | | Trader -> Control, Ping, len=10 |
| 15891 | 09:27:56.877671 | 0.000003 | 192.168.0.1 | 10.0.0.1 | TRADE | 72 | | Multiplexer -> Control, Ping, len=18 |
| 15894 | 09:27:57.080057 | 0.202386 | 10.0.0.1 | 192.168.0.1 | TRADE | 72 | | Trader -> Control, Pong, len=18 |
| 15896 | 09:27:57.921887 | 0.841830 | 192.168.0.1 | 10.0.0.1 | TRADE | 64 | | Multiplexer -> Control, Pong, len=10 |
| 16345 | 09:28:27.937058 | 30.015171 | 192.168.0.1 | 10.0.0.1 | TRADE | 72 | | Multiplexer -> Control, Ping, len=18 |
| 16346 | 09:28:27.937194 | 0.000136 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | | Trader -> Control, Ping, len=10 |
| 16348 | 09:28:28.139867 | 0.202673 | 192.168.0.1 | 10.0.0.1 | TRADE | 64 | | Multiplexer -> Control, Pong, len=10 |
| 16349 | 09:28:28.140156 | 0.000289 | 10.0.0.1 | 192.168.0.1 | TRADE | 72 | | Trader -> Control, Pong, len=18 |
| 17114 | 09:28:58.154115 | 30.013959 | 192.168.0.1 | 10.0.0.1 | TRADE | 72 | | Multiplexer -> Control, Ping, len=18 |
| 17115 | 09:28:58.154966 | 0.000851 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | | Trader -> Control, Ping, len=10 |
| 17116 | 09:28:58.156050 | 0.001084 | 192.168.0.1 | 10.0.0.1 | TRADE | 64 | | Multiplexer -> Control, Pong, len=10 |
| 17117 | 09:28:58.156279 | 0.000229 | 10.0.0.1 | 192.168.0.1 | TRADE | 72 | | Trader -> Control, Pong, len=18 |
| 20997 | 09:29:28.168604 | 30.012325 | 192.168.0.1 | 10.0.0.1 | TRADE | 72 | | Multiplexer -> Control, Ping, len=18 |
| 20998 | 09:29:28.171420 | 0.002816 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | | Trader -> Control, Ping, len=10 |
| 20999 | 09:29:28.172611 | 0.001191 | 192.168.0.1 | 10.0.0.1 | TRADE | 64 | | Multiplexer -> Control, Pong, len=10 |
| 21000 | 09:29:28.172847 | 0.000236 | 10.0.0.1 | 192.168.0.1 | TRADE | 72 | | Trader -> Control, Pong, len=18 |
| 21165 | 09:29:58.167667 | 29.994820 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | | Trader -> Control, Ping, len=10 |
| 21166 | 09:29:58.169923 | 0.002256 | 192.168.0.1 | 10.0.0.1 | TRADE | 64 | | Multiplexer -> Control, Pong, len=10 |
| 21168 | 09:29:58.370326 | 0.200403 | 192.168.0.1 | 10.0.0.1 | TRADE | 72 | | Multiplexer -> Control, Ping, len=18 |
| 21169 | 09:29:58.370998 | 0.000672 | 10.0.0.1 | 192.168.0.1 | TRADE | 72 | | Trader -> Control, Pong, len=18 |
| 31491 | 09:32:12.351912 | 133.9809... | 10.0.0.1 | 192.168.0.1 | TCP | 60 | | 1195 -> 2107 [RST, ACK] Seq=21251242 ... |

anonymized

Packets: 33471 · Displayed: 161 (0.5%) · Load time: 0:1.63 Profile: sf2017eu-dummy





The case of the missing ping!



anonymized.pcap

tcp.stream == 11 && (tcp.flags&7 || trade.keyword in {"Ping" }) && ip.src==10.0.0.1

| No. | Time | Delta | Source | Destination | Protocol | Length | Info |
|-------|-----------------|------------|----------|-------------|----------|--------|---|
| 4283 | 09:17:55.544826 | 30.015988 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Trader -> Control, Ping, len=10 |
| 7782 | 09:20:26.054794 | 30.203947 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Trader -> Control, Ping, len=10 |
| 7955 | 09:20:56.066886 | 30.012092 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Trader -> Control, Ping, len=10 |
| 8764 | 09:21:26.081451 | 30.014565 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Trader -> Control, Ping, len=10 |
| 9197 | 09:21:56.095734 | 30.014283 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Trader -> Control, Ping, len=10 |
| 9334 | 09:22:26.110077 | 30.014343 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Trader -> Control, Ping, len=10 |
| 9537 | 09:22:56.124634 | 30.014557 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Trader -> Control, Ping, len=10 |
| 9681 | 09:23:26.139578 | 30.014744 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Trader -> Control, Ping, len=10 |
| 9804 | 09:23:56.154066 | 30.014688 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Trader -> Control, Ping, len=10 |
| 10543 | 09:24:26.168663 | 30.014597 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Trader -> Control, Ping, len=10 |
| 10666 | 09:24:56.185118 | 30.014455 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Trader -> Control, Ping, len=10 |
| 10841 | 09:25:26.197440 | 30.014222 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Trader -> Control, Ping, len=10 |
| 11554 | 09:25:27.937194 | 31.059526 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Trader -> Control, Ping, len=10 |
| 11759 | 09:26:17.154966 | 30.217772 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Trader -> Control, Ping, len=10 |
| 12116 | 09:26:58.154966 | 30.016454 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Trader -> Control, Ping, len=10 |
| 14859 | 09:27:28.171420 | 30.016454 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Trader -> Control, Ping, len=10 |
| 15890 | 09:27:58.167667 | 29.996247 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Trader -> Control, Ping, len=10 |
| 16346 | 09:28:27.937194 | 29.996247 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Trader -> Control, Ping, len=10 |
| 17115 | 09:28:58.154966 | 30.217772 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Trader -> Control, Ping, len=10 |
| 20998 | 09:29:28.171420 | 30.016454 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Trader -> Control, Ping, len=10 |
| 21165 | 09:29:58.167667 | 29.996247 | 10.0.0.1 | 192.168.0.1 | TRADE | 64 | Trader -> Control, Ping, len=10 |
| 31491 | 09:32:12.351912 | 134.184245 | 10.0.0.1 | 192.168.0.1 | TCP | 60 | 1195 -> 2107 [RST, ACK] Seq=21251242 Ack=405179 Win=0 Len=0 |

2014-06-05 09:31:58,978 ERROR - No response from multiplexer: 192.168.0.1:2107 within 1 minutes 30 seconds. Terminating connection.
2014-06-05 09:31:58,978 WARN - ModuleServer Trader lost connection to multiplexer on 192.168.0.1:2107

09:31:58 - 0:01:30 = ~09:30:28 and not 09:29:58 !!!

anonymized

Packets: 33471 · Displayed: 41 (0.1%) · Load time: 0:1.62 Profile: sf2017eu-dummy





RTT=188.409 ms???



anonymized.pcap

Expression... + RTT

| No. | Time | Delta | Source | Destination | Protocol | Length | Roundtrip Time | Info |
|-------|-----------------|------------|-------------|-------------|----------|--------|----------------|--|
| 17114 | 09:28:58.154115 | 30.012684 | 192.168.... | 10.0.0.1 | TRADE | 72 | | Multiplexer -> Control, Ping, len=18 |
| 17117 | 09:28:58.156279 | 0.002164 | 10.0.0.1 | 192.168.0.1 | TRADE | 72 | | Trader -> Control, Pong, len=18 |
| 17118 | 09:28:58.157637 | 0.001358 | 192.168.... | 10.0.0.1 | TRADE | 92 | 3.234 | Multiplexer -> Control, Roundtrip for Modu... |
| 20997 | 09:29:28.168604 | 30.010967 | 192.168.... | 10.0.0.1 | TRADE | 72 | | Multiplexer -> Control, Ping, len=18 |
| 21000 | 09:29:28.172847 | 0.004243 | 10.0.0.1 | 192.168.0.1 | TRADE | 72 | | Trader -> Control, Pong, len=18 |
| 21001 | 09:29:28.174261 | 0.001414 | 192.168.... | 10.0.0.1 | TRADE | 92 | 5.386 | Multiplexer -> Control, Roundtrip for Modu... |
| 21168 | 09:29:58.370326 | 30.196065 | 192.168.... | 10.0.0.1 | TRADE | 72 | | Multiplexer -> Control, Ping, len=18 |
| 21169 | 09:29:58.370998 | 0.000672 | 10.0.0.1 | 192.168.0.1 | TRADE | 72 | | Trader -> Control, Pong, len=18 |
| 21170 | 09:29:58.372497 | 0.001499 | 192.168.... | 10.0.0.1 | TRADE | 92 | 188.409 | Multiplexer -> Control, Roundtrip for Modu... |
| 31491 | 09:32:12.351912 | 133.979415 | 10.0.0.1 | 192.168.0.1 | TCP | 60 | | 1195 -> 2107 [RST, ACK] Seq=21251242 Ack=40... |
| 31590 | 09:32:33.358029 | 21.006117 | 192.168.... | 10.0.0.1 | TRADE | 92 | 1.492 | Multiplexer -> Control, Roundtrip for Modu... |
| 31831 | 09:33:03.372402 | 30.014373 | 192.168.... | 10.0.0.1 | TRADE | 92 | 1.408 | Multiplexer -> Control, Roundtrip for Modu... |
| 32111 | 09:33:33.388628 | 30.016226 | 192.168.... | 10.0.0.1 | TRADE | 92 | 3.129 | Multiplexer -> Control, Roundtrip for Modu... |

Frame 21170: 92 bytes on wire (736 bits), 92 bytes captured (736 bits)

- Ethernet II, Src: 02:00:00:00:00:02 (02:00:00:00:00:02), Dst: 02:00:00:00:00:01 (02:00:00:00:00:01)
- Internet Protocol Version 4, Src: 192.168.0.1, Dst: 10.0.0.1
- Transmission Control Protocol, Src Port: 2107, Dst Port: 1170, Seq: 61425, Ack: 137213, Len: 38
- Trade Control PDU : Multiplexer -> Control, Roundtrip for Module Trader is 188.409 ms, len=38
 - Sender: Multiplexer (1)
 - Length: 38
 - Receiver: 65535 (Control)
 - Keyword: Roundtrip for Module
 - Module: Trader
 - Roundtrip Time: 188.409 ms

Frame (frame), 92 bytes

Packets: 33471 · Displayed: 124 (0.4%) · Load time: 0:1.139 Profile: sf2017eu-dummy





Stop talking... PLEASE!!!



anonymized.pcap

tcp.stream == 11

| No. | Time | Delta | Source | Destination | Protocol | Length | Info |
|-------|-----------------|-----------|-------------|-------------|----------|--------|--|
| 30330 | 09:30:11.830184 | 0.000005 | 10.0.0.1 | 192.168.0.1 | TCP | 1434 | 1195 → 2107 [ACK] Seq=21245508 Ack=277140 Win=7868160 L... |
| 30331 | 09:30:11.830187 | 0.000003 | 10.0.0.1 | 192.168.0.1 | TRADE | 1389 | Trader → 2643, len=65535 |
| 30332 | 09:30:11.830267 | 0.000080 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 → 1195 [ACK] Seq=277140 Ack=21248223 Win=2816 Len=0 |
| 30333 | 09:30:11.836826 | 0.006559 | 10.0.0.1 | 192.168.0.1 | TCP | 1434 | 1195 → 2107 [ACK] Seq=21248223 Ack=277140 Win=7868160 L... |
| 30334 | 09:30:11.836830 | 0.000004 | 10.0.0.1 | 192.168.0.1 | TCP | 1434 | 1195 → 2107 [ACK] Seq=21249603 Ack=277140 Win=7868160 L... |
| 30335 | 09:30:11.836914 | 0.000084 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 → 1195 [ACK] Seq=277140 Ack=21250983 Win=256 Len=0 |
| 30339 | 09:30:11.963966 | 0.127052 | 192.168.0.1 | 10.0.0.1 | TRADE | 78 | 2107 → 1195 [PSH, ACK] Seq=277140 Ack=21250983 Win=256 ... |
| 30343 | 09:30:12.176270 | 0.212304 | 10.0.0.1 | 192.168.0.1 | TCP | 60 | 1195 → 2107 [ACK] Seq=21250983 Ack=277164 Win=7868160 L... |
| 30354 | 09:30:16.856398 | 4.680128 | 10.0.0.1 | 192.168.0.1 | TCP | 310 | 1195 → 2107 [ACK] Seq=21250983 Ack=277164 Win=7868160 L... |
| 30356 | 09:30:17.074505 | 0.218107 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | [TCP ZeroWindow] 2107 → 1195 [ACK] Seq=277164 Ack=21251... |
| 30363 | 09:30:17.386582 | 0.312077 | 10.0.0.1 | 192.168.0.1 | TCP | 60 | [TCP ZeroWindowProbe] 1195 → 2107 [ACK] Seq=21251239 Ac... |
| 30365 | 09:30:17.589253 | 0.202671 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | [TCP ZeroWindow] [TCP ACKed unseen segment] 2107 → 1195... |
| 30369 | 09:30:18.416293 | 0.827040 | 10.0.0.1 | 192.168.0.1 | TCP | 60 | [TCP Previous segment not captured] 1195 → 2107 [ACK] S... |
| 30370 | 09:30:18.634502 | 0.218209 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | [TCP ZeroWindow] [TCP ACKed unseen segment] 2107 → 1195... |
| 30375 | 09:30:20.693836 | 2.059334 | 10.0.0.1 | 192.168.0.1 | TCP | 60 | [TCP ZeroWindowProbe] 1195 → 2107 [ACK] Seq=21251241 Ac... |
| 30377 | 09:30:20.912079 | 0.218243 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | [TCP ZeroWindow] [TCP ACKed unseen segment] 2107 → 1195... |
| 30383 | 09:30:25.285042 | 4.372963 | 192.168.0.1 | 10.0.0.1 | TRADE | 1434 | [TCP ZeroWindow] [TCP ACKed unseen segment] 2107 → 1195... |
| 30384 | 09:30:25.285046 | 0.000004 | 192.168.0.1 | 10.0.0.1 | TRADE | 1044 | [TCP ZeroWindow] [TCP ACKed unseen segment] 2107 → 1195... |
| 30385 | 09:30:25.285824 | 0.000778 | 10.0.0.1 | 192.168.0.1 | TCP | 60 | [TCP Previous segment not captured] 1195 → 2107 [ACK] S... |
| 30407 | 09:30:28.369182 | 3.083358 | 192.168.0.1 | 10.0.0.1 | TRADE | 72 | [TCP ZeroWindow] [TCP ACKed unseen segment] 2107 → 1195... |
| 30409 | 09:30:28.571787 | 0.202605 | 10.0.0.1 | 192.168.0.1 | TCP | 60 | 1195 → 2107 [ACK] Seq=21251242 Ack=279552 Win=7865600 L... |
| 30413 | 09:30:34.471800 | 5.900013 | 192.168.0.1 | 10.0.0.1 | TRADE | 1434 | [TCP ZeroWindow] 2107 → 1195 [ACK] Seq=279552 Ack=21251... |
| 30414 | 09:30:34.471804 | 0.000004 | 192.168.0.1 | 10.0.0.1 | TRADE | 992 | [TCP ZeroWindow] 2107 → 1195 [PSH, ACK] Seq=280932 Ack=... |
| 30415 | 09:30:34.472218 | 0.000414 | 10.0.0.1 | 192.168.0.1 | TCP | 60 | 1195 → 2107 [ACK] Seq=21251242 Ack=281870 Win=7863296 L... |
| 30520 | 09:30:47.263035 | 12.790817 | 192.168.0.1 | 10.0.0.1 | TRADE | 106 | [TCP ZeroWindow] 2107 → 1195 [PSH, ACK] Seq=281870 Ack=... |
| 30527 | 09:30:47.463450 | 0.200415 | 10.0.0.1 | 192.168.0.1 | TCP | 60 | 1195 → 2107 [ACK] Seq=21251242 Ack=281922 Win=7863296 L... |
| 31012 | 09:30:49.499261 | 2.035811 | 192.168.0.1 | 10.0.0.1 | TRADE | 161 | [TCP ZeroWindow] 2107 → 1195 [PSH, ACK] Seq=281922 Ack=... |
| 31015 | 09:30:49.709994 | 0.210733 | 10.0.0.1 | 192.168.0.1 | TCP | 60 | 1195 → 2107 [ACK] Seq=21251242 Ack=282029 Win=7863296 L... |

Frame (frame), 60 bytes

Packets: 33471 · Displayed: 20924 (62.5%) · Load time: 0:1.51 Profile: sf2017eu-dummy





But I have so much to say...



anonymized.pcap

tcp.stream==11 && ip.src==10.0.0.1 && tcp.len>0

| No. | Time | Delta | Source | Destination | Protocol | Length | Info |
|-------|-----------------|----------|----------|-------------|----------|--------|--|
| 30307 | 09:30:11.828855 | 0.000004 | 10.0.0.1 | 192.168.0.1 | TCP | 1434 | 1195 → 2107 [ACK] Seq=21219728 Ack=277140 Win=7868160 L... |
| 30308 | 09:30:11.828859 | 0.000004 | 10.0.0.1 | 192.168.0.1 | TCP | 1434 | 1195 → 2107 [ACK] Seq=21221108 Ack=277140 Win=7868160 L... |
| 30309 | 09:30:11.828863 | 0.000004 | 10.0.0.1 | 192.168.0.1 | TCP | 1214 | 1195 → 2107 [PSH, ACK] Seq=21222488 Ack=277140 Win=7868... |
| 30312 | 09:30:11.829733 | 0.000870 | 10.0.0.1 | 192.168.0.1 | TCP | 1434 | 1195 → 2107 [ACK] Seq=21223648 Ack=277140 Win=7868160 L... |
| 30313 | 09:30:11.829737 | 0.000004 | 10.0.0.1 | 192.168.0.1 | TCP | 1434 | 1195 → 2107 [ACK] Seq=21225028 Ack=277140 Win=7868160 L... |
| 30314 | 09:30:11.829741 | 0.000004 | 10.0.0.1 | 192.168.0.1 | TCP | 1434 | 1195 → 2107 [ACK] Seq=21226408 Ack=277140 Win=7868160 L... |
| 30315 | 09:30:11.829745 | 0.000004 | 10.0.0.1 | 192.168.0.1 | TCP | 1434 | 1195 → 2107 [ACK] Seq=21227788 Ack=277140 Win=7868160 L... |
| 30316 | 09:30:11.829749 | 0.000004 | 10.0.0.1 | 192.168.0.1 | TCP | 1434 | 1195 → 2107 [ACK] Seq=21229168 Ack=277140 Win=7868160 L... |
| 30317 | 09:30:11.829791 | 0.000042 | 10.0.0.1 | 192.168.0.1 | TCP | 1434 | 1195 → 2107 [ACK] Seq=21230548 Ack=277140 Win=7868160 L... |
| 30318 | 09:30:11.829795 | 0.000004 | 10.0.0.1 | 192.168.0.1 | TCP | 1434 | 1195 → 2107 [ACK] Seq=21231928 Ack=277140 Win=7868160 L... |
| 30319 | 09:30:11.829799 | 0.000004 | 10.0.0.1 | 192.168.0.1 | TCP | 1434 | 1195 → 2107 [ACK] Seq=21233308 Ack=277140 Win=7868160 L... |
| 30320 | 09:30:11.829803 | 0.000004 | 10.0.0.1 | 192.168.0.1 | TCP | 1434 | 1195 → 2107 [ACK] Seq=21234688 Ack=277140 Win=7868160 L... |
| 30321 | 09:30:11.829892 | 0.000089 | 10.0.0.1 | 192.168.0.1 | TCP | 1434 | 1195 → 2107 [ACK] Seq=21236068 Ack=277140 Win=7868160 L... |
| 30322 | 09:30:11.829896 | 0.000004 | 10.0.0.1 | 192.168.0.1 | TCP | 1434 | 1195 → 2107 [ACK] Seq=21237448 Ack=277140 Win=7868160 L... |
| 30323 | 09:30:11.829901 | 0.000005 | 10.0.0.1 | 192.168.0.1 | TCP | 1434 | 1195 → 2107 [ACK] Seq=21238828 Ack=277140 Win=7868160 L... |
| 30324 | 09:30:11.829905 | 0.000004 | 10.0.0.1 | 192.168.0.1 | TCP | 1434 | 1195 → 2107 [ACK] Seq=21240208 Ack=277140 Win=7868160 L... |
| 30325 | 09:30:11.829909 | 0.000004 | 10.0.0.1 | 192.168.0.1 | TCP | 1434 | 1195 → 2107 [ACK] Seq=21241588 Ack=277140 Win=7868160 L... |
| 30326 | 09:30:11.829912 | 0.000003 | 10.0.0.1 | 192.168.0.1 | TCP | 1214 | 1195 → 2107 [PSH, ACK] Seq=21242968 Ack=277140 Win=7868... |
| 30329 | 09:30:11.830179 | 0.000267 | 10.0.0.1 | 192.168.0.1 | TCP | 1434 | 1195 → 2107 [ACK] Seq=21244128 Ack=277140 Win=7868160 L... |
| 30330 | 09:30:11.830184 | 0.000005 | 10.0.0.1 | 192.168.0.1 | TCP | 1434 | 1195 → 2107 [ACK] Seq=21245508 Ack=277140 Win=7868160 L... |
| 30331 | 09:30:11.830187 | 0.000003 | 10.0.0.1 | 192.168.0.1 | TRADE | 1389 | Trader -> 2643, len=65535 |
| 30333 | 09:30:11.836826 | 0.006639 | 10.0.0.1 | 192.168.0.1 | TCP | 1434 | 1195 → 2107 [ACK] Seq=21248223 Ack=277140 Win=7868160 L... |
| 30334 | 09:30:11.836830 | 0.000004 | 10.0.0.1 | 192.168.0.1 | TCP | 1434 | 1195 → 2107 [ACK] Seq=21249603 Ack=277140 Win=7868160 L... |
| 30354 | 09:30:16.856398 | 5.019568 | 10.0.0.1 | 192.168.0.1 | TCP | 310 | 1195 → 2107 [ACK] Seq=21250983 Ack=277164 Win=7868160 L... |
| 30363 | 09:30:17.386582 | 0.530184 | 10.0.0.1 | 192.168.0.1 | TCP | 60 | [TCP ZeroWindowProbe] 1195 → 2107 [ACK] Seq=21251239 Ac... |
| 30369 | 09:30:18.416293 | 1.029711 | 10.0.0.1 | 192.168.0.1 | TCP | 60 | [TCP Previous segment not captured] 1195 → 2107 [ACK] S... |
| 30375 | 09:30:20.693836 | 2.277543 | 10.0.0.1 | 192.168.0.1 | TCP | 60 | [TCP ZeroWindowProbe] 1195 → 2107 [ACK] Seq=21251241 Ac... |

Frame (frame), 64 bytes

Packets: 33471 · Displayed: 17287 (51.6%) · Load time: 0:1.112 Profile: sf2017eu-dummy





Sorry, busy!



anonymized.pcap

tcp.stream==11 && ip.dst==10.0.0.1

| No. | Time | Delta | Source | Destination | Protocol | Length | Info |
|-------|-----------------|-----------|-------------|-------------|----------|--------|--|
| 30311 | 09:30:11.828945 | 0.000052 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 → 1195 [ACK] Seq=277140 Ack=21223648 Win=27392 Len... |
| 30327 | 09:30:11.829916 | 0.000971 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 → 1195 [ACK] Seq=277140 Ack=21230548 Win=20480 Len... |
| 30328 | 09:30:11.829963 | 0.000047 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 → 1195 [ACK] Seq=277140 Ack=21244128 Win=6912 Len=0 |
| 30332 | 09:30:11.830267 | 0.000304 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 → 1195 [ACK] Seq=277140 Ack=21248223 Win=2816 Len=0 |
| 30335 | 09:30:11.836914 | 0.006647 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | 2107 → 1195 [ACK] Seq=277140 Ack=21250983 Win=256 Len=0 |
| 30339 | 09:30:11.963966 | 0.127052 | 192.168.0.1 | 10.0.0.1 | TRADE | 78 | 2107 → 1195 [PSH, ACK] Seq=277140 Ack=21250983 Win=256 ... |
| 30356 | 09:30:17.074505 | 5.110539 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | [TCP ZeroWindow] 2107 → 1195 [ACK] Seq=277164 Ack=21251... |
| 30365 | 09:30:17.589253 | 0.514748 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | [TCP ZeroWindow] [TCP ACKed unseen segment] 2107 → 1195... |
| 30370 | 09:30:18.634502 | 1.045249 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | [TCP ZeroWindow] [TCP ACKed unseen segment] 2107 → 1195... |
| 30377 | 09:30:20.912079 | 2.277577 | 192.168.0.1 | 10.0.0.1 | TCP | 60 | [TCP ZeroWindow] [TCP ACKed unseen segment] 2107 → 1195... |
| 30383 | 09:30:25.285042 | 4.372963 | 192.168.0.1 | 10.0.0.1 | TRADE | 1434 | [TCP ZeroWindow] [TCP ACKed unseen segment] 2107 → 1195... |
| 30384 | 09:30:25.285046 | 0.000004 | 192.168.0.1 | 10.0.0.1 | TRADE | 1044 | [TCP ZeroWindow] [TCP ACKed unseen segment] 2107 → 1195... |
| 30407 | 09:30:28.369182 | 3.084136 | 192.168.0.1 | 10.0.0.1 | TRADE | 72 | [TCP ZeroWindow] [TCP ACKed unseen segment] 2107 → 1195... |
| 30413 | 09:30:34.471800 | 6.102618 | 192.168.0.1 | 10.0.0.1 | TRADE | 1434 | [TCP ZeroWindow] 2107 → 1195 [ACK] Seq=279552 Ack=21251... |
| 30414 | 09:30:34.471804 | 0.000004 | 192.168.0.1 | 10.0.0.1 | TRADE | 992 | [TCP ZeroWindow] 2107 → 1195 [PSH, ACK] Seq=280932 Ack=... |
| 30520 | 09:30:47.263035 | 12.791231 | 192.168.0.1 | 10.0.0.1 | TRADE | 106 | [TCP ZeroWindow] 2107 → 1195 [PSH, ACK] Seq=281870 Ack=... |
| 31012 | 09:30:49.499261 | 2.236226 | 192.168.0.1 | 10.0.0.1 | TRADE | 161 | [TCP ZeroWindow] 2107 → 1195 [PSH, ACK] Seq=281922 Ack=... |
| 31249 | 09:31:47.489355 | 57.990094 | 192.168.0.1 | 10.0.0.1 | TRADE | 1434 | [TCP ZeroWindow] 2107 → 1195 [ACK] Seq=282029 Ack=21251... |
| 31250 | 09:31:47.489360 | 0.000005 | 192.168.0.1 | 10.0.0.1 | TRADE | 1044 | [TCP ZeroWindow] 2107 → 1195 [PSH, ACK] Seq=283409 Ack=... |
| 31270 | 09:31:49.864347 | 2.374987 | 192.168.0.1 | 10.0.0.1 | TRADE | 1434 | [TCP ZeroWindow] 2107 → 1195 [ACK] Seq=284399 Ack=21251... |
| 31271 | 09:31:49.864351 | 0.000004 | 192.168.0.1 | 10.0.0.1 | TRADE | 1434 | [TCP ZeroWindow] 2107 → 1195 [ACK] Seq=285779 Ack=21251... |
| 31272 | 09:31:49.864355 | 0.000004 | 192.168.0.1 | 10.0.0.1 | TRADE | 1434 | [TCP ZeroWindow] 2107 → 1195 [ACK] Seq=287159 Ack=21251... |
| 31273 | 09:31:49.864360 | 0.000005 | 192.168.0.1 | 10.0.0.1 | TRADE | 1434 | [TCP ZeroWindow] 2107 → 1195 [ACK] Seq=288539 Ack=21251... |
| 31274 | 09:31:49.864364 | 0.000004 | 192.168.0.1 | 10.0.0.1 | TRADE | 1434 | [TCP ZeroWindow] 2107 → 1195 [ACK] Seq=289919 Ack=21251... |
| 31275 | 09:31:49.864368 | 0.000004 | 192.168.0.1 | 10.0.0.1 | TRADE | 1434 | [TCP ZeroWindow] 2107 → 1195 [ACK] Seq=291299 Ack=21251... |
| 31276 | 09:31:49.864372 | 0.000004 | 192.168.0.1 | 10.0.0.1 | TCP | 1434 | [TCP ZeroWindow] 2107 → 1195 [ACK] Seq=292679 Ack=21251... |
| 31277 | 09:31:49.864376 | 0.000004 | 192.168.0.1 | 10.0.0.1 | TRADE | 1434 | [TCP ZeroWindow] Multiplexer → 717, len=2014 |
| 31278 | 09:31:49.864380 | 0.000004 | 192.168.0.1 | 10.0.0.1 | TRADE | 1434 | [TCP ZeroWindow] 2107 → 1195 [ACK] Seq=295439 Ack=21251... |

Frame (frame), 60 bytes

Packets: 33471 · Displayed: 3487 (10.4%) · Load time: 0:1.131 Profile: sf2017eu-dummy





Root Cause Analysis



- APPSERVER1 sends a lot of data and fills receive buffer of Multiplexer
- APPSERVER1 thinks it has sent a “Ping” packet, but packet never left the TCP buffer because of RWIN=0
- After 90 seconds without “Pong”, the APPSERVER1 closes the connection
- RCA: The processes on Multiplexer somehow don't empty the receive buffer and block communication





Did LUA-dissector help?



- Understand the Ping-Pong mechanism
- Understand the RTT measurement
- Made filtering easier (or even possible)
- Gives the customer insight in their traffic in future troubleshooting events





Summary...



- Proprietary protocols are hard to analyze from hex
- LUA dissector not extremely difficult to write
 - ▶ OK, after a lot of googling, so hopefully this presentation helps you from having to do the same ;-)
- Time to find Root Cause greatly reduced
 - ▶ Well, in this case, following up on the Zero-Window problem did point in the right direction. But knowing what actually happened helped in communication with the Software vendor so they could solve the issue more quickly





Q&A





FIN/ACK, ACK, FIN/ACK, ACK



Thank You!

sake.blok@SYN-bit.nl



SYN-bit
deep traffic analysis

