



IBM Software Group

TCP Packet Tracing – Part 2

Robert L Boretti Jr (robb@us.ibm.com)

Marvin Knight (knightm@us.ibm.com)

Advisory Software Engineers

26 May 2011



WebSphere® Support Technical Exchange



Agenda

- **Main Focus - TCP Packet Tracing**
 - ▶ ***Debugging / Problem Analysis***
 - **A Few Quick Tips** - wireshark *filter expressions, packet searches, capture save options*
 - **Packet Correlation** - with other logs *IBM® HTTP Server, WebSphere HTTP plug-in and WebSphere Application Server*
 - **Decryption** - How to *decrypt* SSL packets

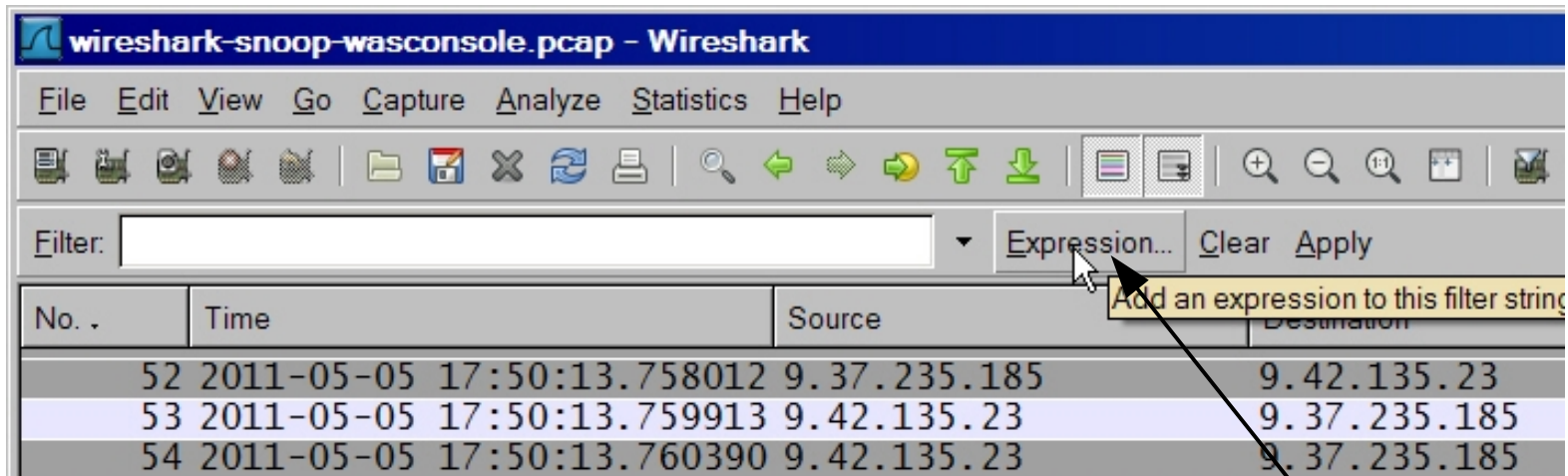


A Few Quick Tips



A Few Quick Tips

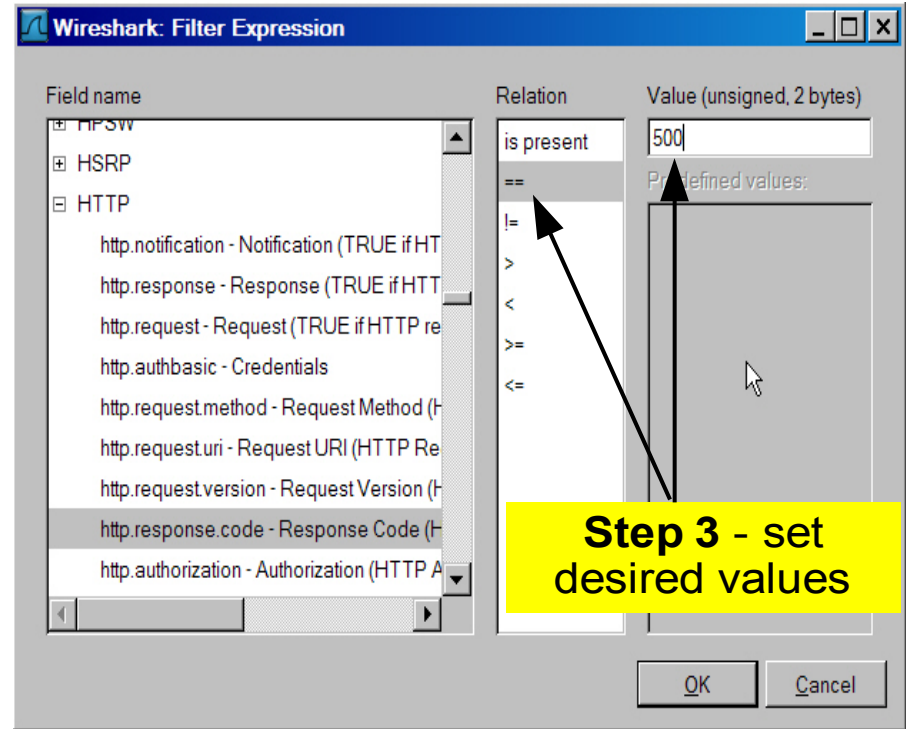
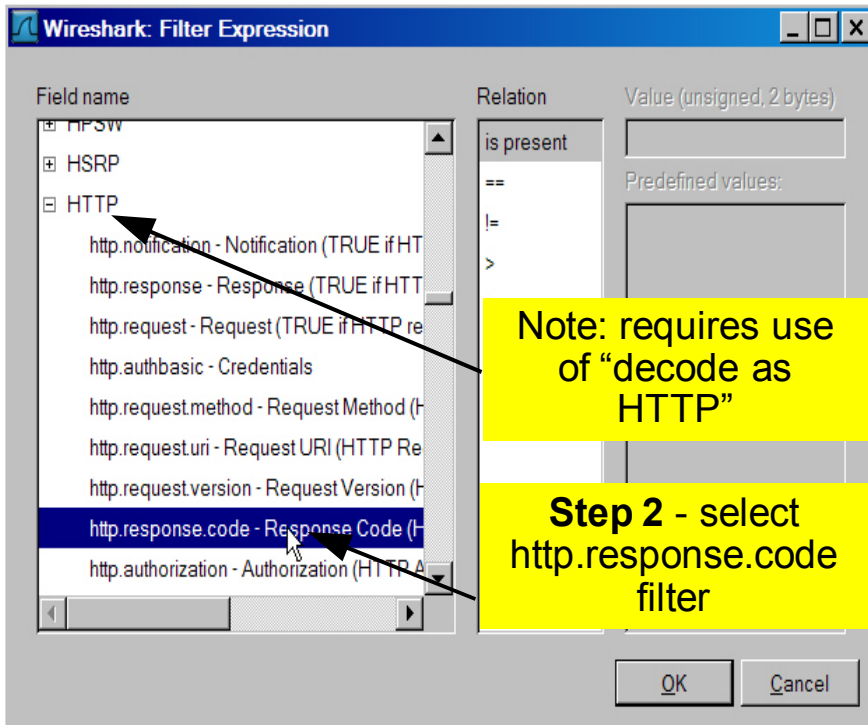
- **Tip #1** - How to use a **filter expression** to quickly find packets containing an *HTTP status code*(e.g. 200, 500, 400) on a specific *server port*(e.g. 80, 443)



Step 1 - click on expression

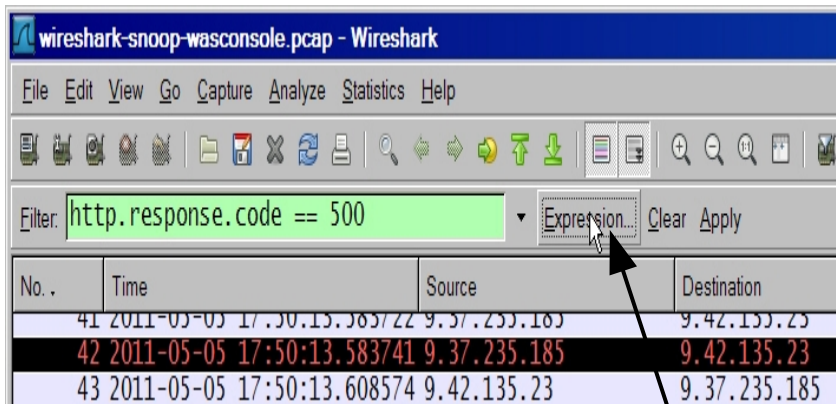
A Few Quick Tips

- **Tip #1** - How to use a **filter expression** to quickly find packets containing an *HTTP status code*(e.g. 200, 500, 400) on a specific *server port*(e.g. 80, 443)

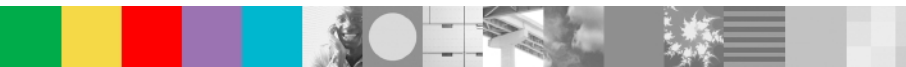
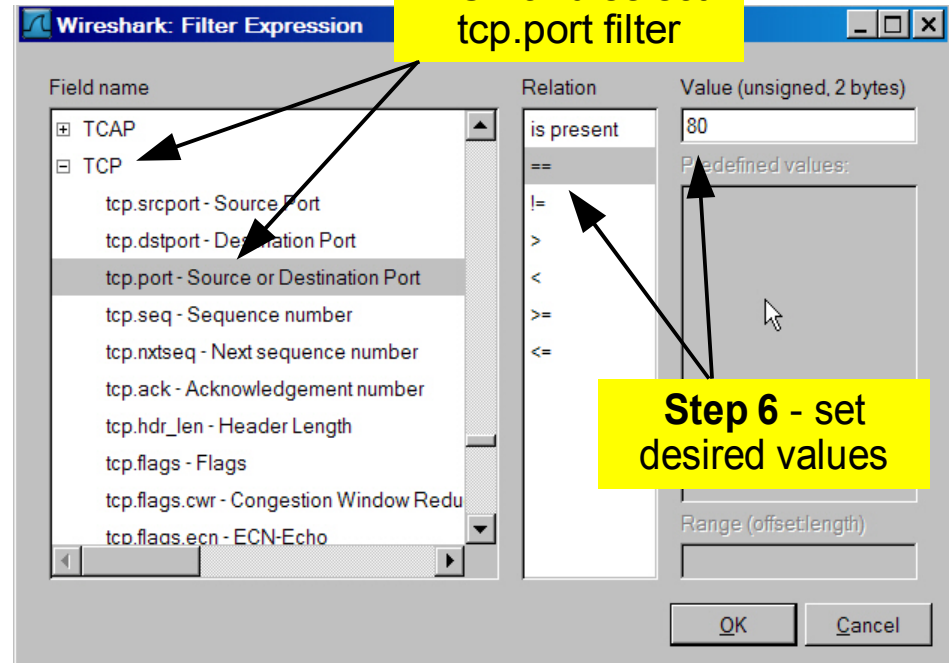


A Few Quick Tips

- **Tip #1** - How to use a **filter expression** to quickly find packets containing an *HTTP status code*(e.g. 200, 500, 400) on a specific *server port*(e.g. 80, 443)

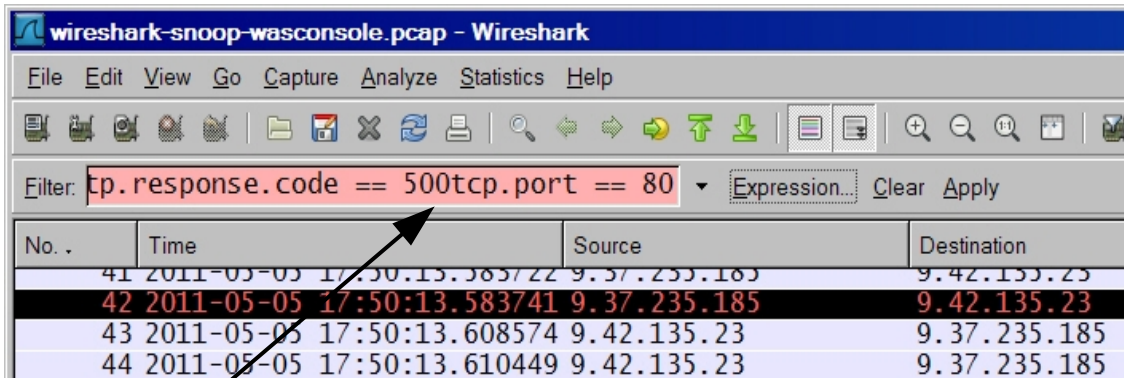


Step 4 - set a second expression



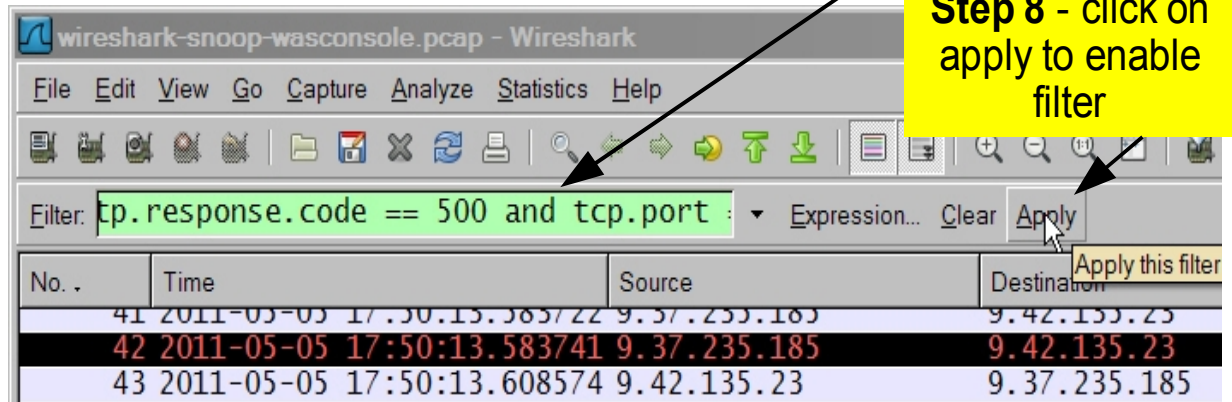
A Few Quick Tips

- **Tip #1** - How to use a **filter expression** to quickly find packets containing an *HTTP status code*(e.g. 200, 500, 400) on a specific *server port*(e.g. 80, 443)



Notice 2nd expression runs in to first

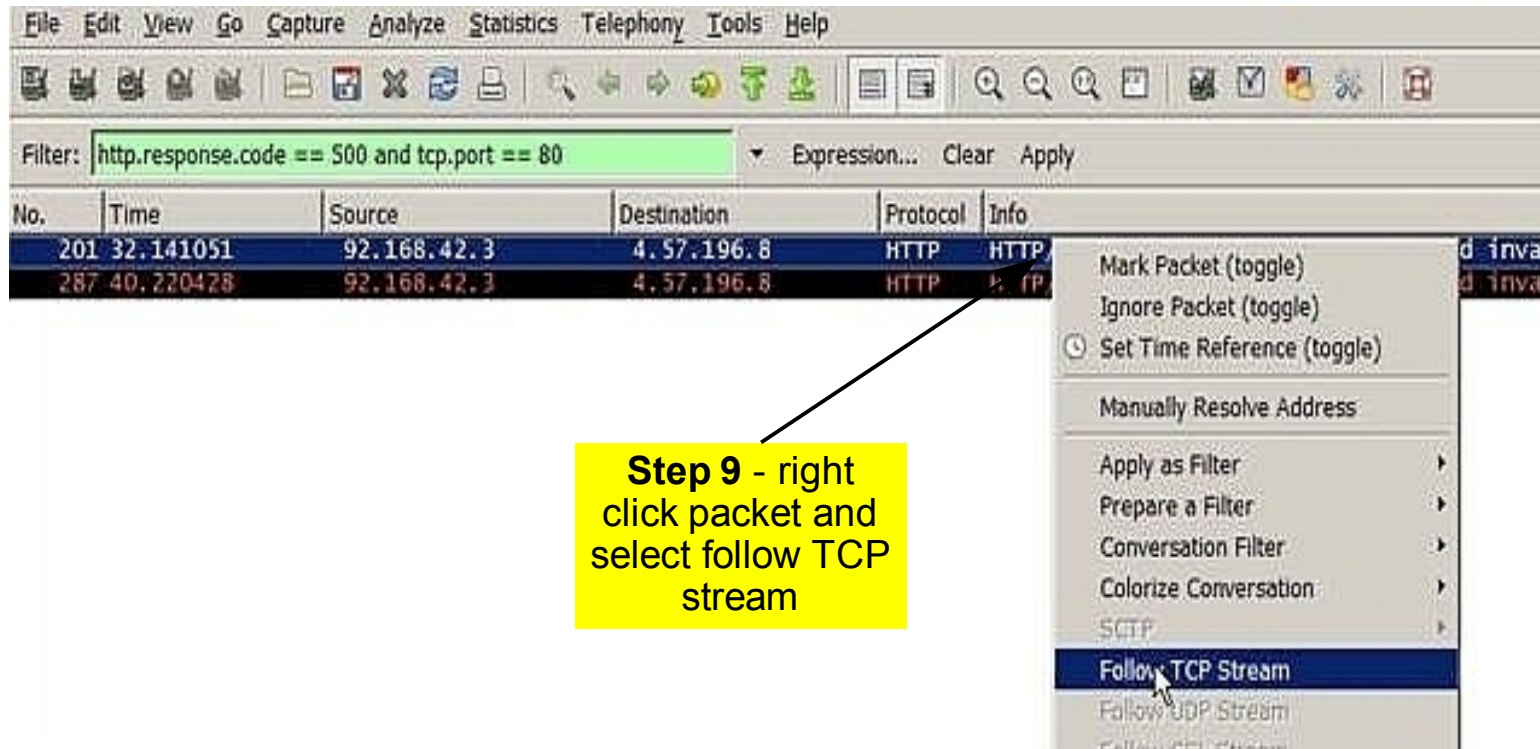
Step 7 - add boolean connector (and..or)



Step 8 - click on apply to enable filter

A Few Quick Tips

- **Tip #1** - How to use a **filter expression** to quickly find packets containing an *HTTP status code*(e.g. 200, 500, 400) on a specific *server port*(e.g. 80, 443)



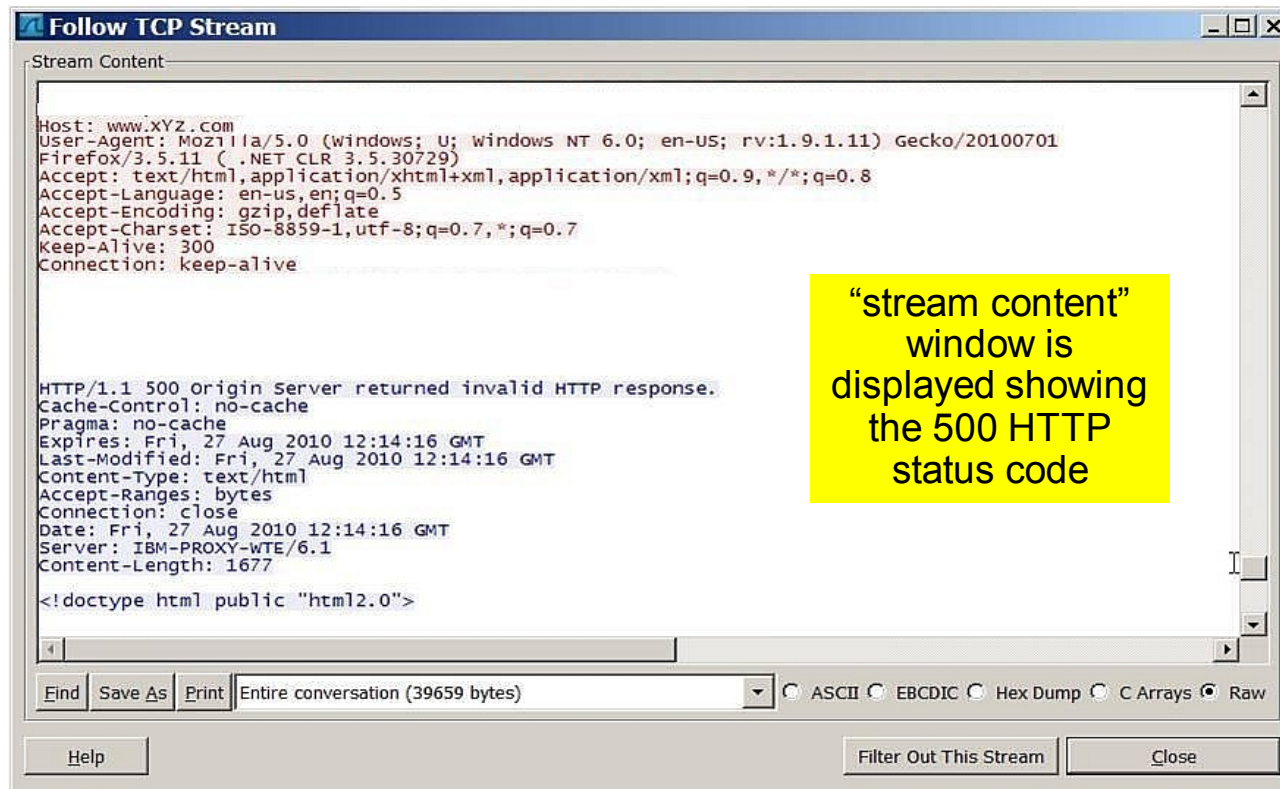
The screenshot shows the Wireshark interface with a filter expression `http.response.code == 500 and tcp.port == 80` applied. The packet list shows two HTTP packets. A context menu is open over the first packet, and the 'Follow TCP Stream' option is highlighted. A yellow callout box points to this option.

| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|------|
| 201 | 32.141051 | 92.168.42.3 | 4.57.196.8 | HTTP | HTTP |
| 287 | 40.220428 | 92.168.42.3 | 4.57.196.8 | HTTP | HTTP |

Step 9 - right click packet and select follow TCP stream

A Few Quick Tips

- **Tip #1** - How to use a **filter expression** to quickly find packets containing an *HTTP status code*(e.g. 200, 500, 400) on a specific *server port*(e.g. 80, 443)



A Few Quick Tips

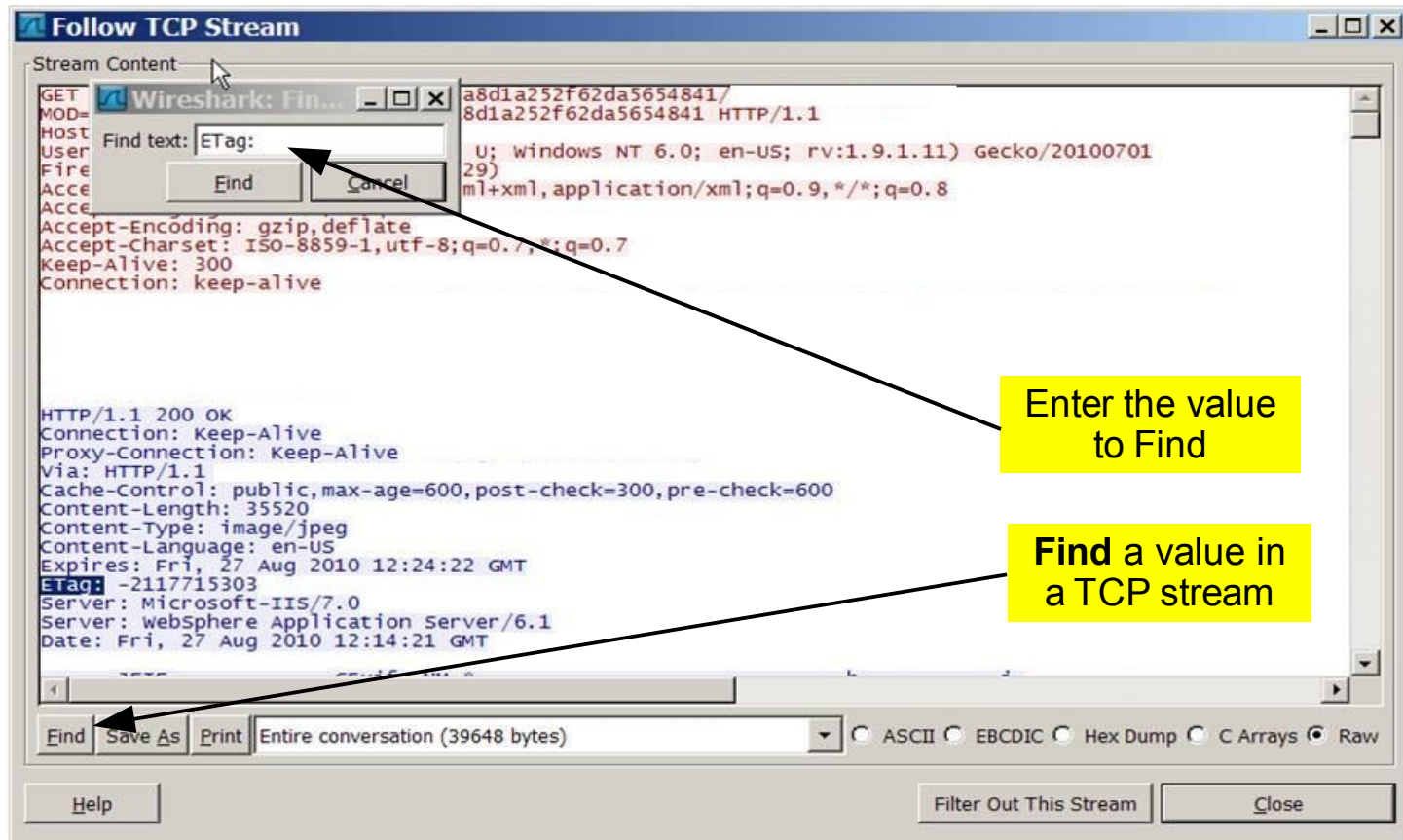
- **Tip #1** - How to use a **filter expression** to quickly find packets containing an *HTTP status code*(e.g. 200, 500, 400) on a specific *server port*(e.g. 80, 443)

| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|---|
| 138 | 28.466797 | 4.57.196.8 | 92.168.42.3 | TCP | 62367 > http [ACK] Seq=775 Ack=28781 win=64860 Len=0 |
| 139 | 28.476173 | 92.168.42.3 | 4.57.196.8 | TCP | [TCP segment of a reassembled PDU] |
| 140 | 28.479410 | 92.168.42.3 | 4.57.196.8 | TCP | [TCP segment of a reassembled PDU] |
| 142 | 28.498773 | 4.57.196.8 | 92.168.42.3 | TCP | 62367 > http [ACK] Seq=775 Ack=31185 win=64860 Len=0 |
| 143 | 28.498776 | 4.57.196.8 | 92.168.42.3 | TCP | 62367 > http [ACK] Seq=775 Ack=32877 win=64860 Len=0 |
| 144 | 28.504119 | 92.168.42.3 | 4.57.196.8 | HTTP | HTTP/1.1 200 OK (JPEG JFIF image) |
| 145 | 28.526506 | 4.57.196.8 | 92.168.42.3 | TCP | 62367 > http [ACK] Seq=775 Ack=36038 win=64860 Len=0 |
| 188 | 30.397032 | 4.57.196.8 | 92.168.42.3 | HTTP | GET /wps/wcm/connect/8fe1fb80439ea8d1a252f62da5654841/outdoor |
| 189 | 30.592425 | 92.168.42.3 | 4.57.196.8 | TCP | http > 62367 [ACK] Seq=36038 Ack=1603 win=64032 Len=0 |
| 200 | 32.132765 | 92.168.42.3 | 4.57.196.8 | TCP | [TCP segment of a reassembled PDU] |
| 201 | 32.141051 | 92.168.42.3 | 4.57.196.8 | HTTP | HTTP/1.1 500 Origin Server returned invalid HTTP response. |
| 202 | 32.164014 | 4.57.196.8 | 92.168.42.3 | TCP | 62367 > http [ACK] Seq=1603 Ack=38058 win=64860 Len=0 |
| 203 | 32.164051 | 4.57.196.8 | 92.168.42.3 | TCP | 62367 > http [FIN, ACK] Seq=1603 Ack=38058 win=64860 Len=0 |
| 204 | 32.164078 | 92.168.42.3 | 4.57.196.8 | TCP | http > 62367 [ACK] Seq=38058 Ack=1604 win=64032 Len=0 |
| 205 | 32.177389 | 92.168.42.3 | 4.57.196.8 | TCP | http > 62367 [FIN, ACK] Seq=38058 Ack=1604 win=64032 Len=0 |
| 209 | 32.197736 | 4.57.196.8 | 92.168.42.3 | TCP | 62367 > http [ACK] Seq=1604 Ack=38059 win=64860 Len=0 |

After closing the “stream content” window.. what is shown is the entire TCP stream/connection that includes the 500 response code

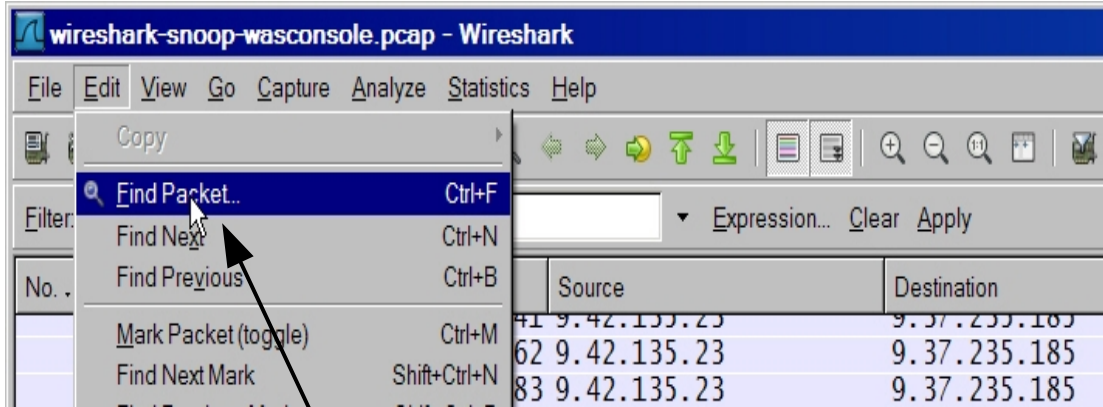
A Few Quick Tips

- **Tip #2** - How to use **Find** to search for a particular header or string in a filtered TCP stream/connection



A Few Quick Tips

- **Tip #3** - How to use a quick **search string** to *find a packet* containing specific information

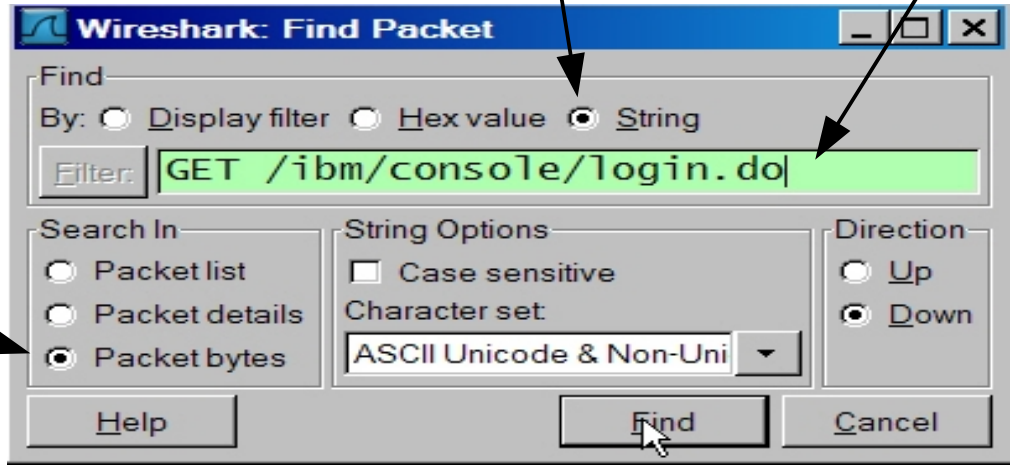


Step 1 - select Edit -> Find Packet..

Step 2 - click on "packet bytes" radio button

Step 3 - click on "String" radio button

Step 4 - enter the desired search information



A Few Quick Tips

- **Tip #3** - How to use a quick **search string** to *find a packet* containing specific information

The screenshot shows the Wireshark interface with a list of network packets. Packet #40 is highlighted in blue. Below the list, the packet bytes pane is visible, showing the raw data of the selected packet. A yellow callout box points to packet #40 in the list and the corresponding data in the bytes pane.

| No. | Time | Source | Destination | Protocol | Info |
|-----|----------------------------|--------------|--------------|----------|---------------------------|
| 30 | 2011-05-05 17:50:13.581547 | 9.42.135.23 | 9.37.235.185 | TCP | 9060 > els [ACK] Seq=1 |
| 31 | 2011-05-05 17:50:13.581588 | 9.42.135.23 | 9.37.235.185 | TCP | 9060 > vrts-ipcserver [|
| 32 | 2011-05-05 17:50:13.581610 | 9.42.135.23 | 9.37.235.185 | TCP | 9060 > amx-icsp [ACK] S |
| 33 | 2011-05-05 17:50:13.581630 | 9.42.135.23 | 9.37.235.185 | TCP | 9060 > exhibit-escp [ACK] |
| 34 | 2011-05-05 17:50:13.581651 | 9.42.135.23 | 9.37.235.185 | TCP | 9060 > amx-axbnet [ACK] |
| 35 | 2011-05-05 17:50:13.581987 | 9.42.135.23 | 9.37.235.185 | TCP | 9060 > krb5gatekeeper [|
| 36 | 2011-05-05 17:50:13.583304 | 9.42.135.23 | 9.37.235.185 | TCP | 9060 > brcd [SYN, ACK] |
| 37 | 2011-05-05 17:50:13.583364 | 9.37.235.185 | 9.42.135.23 | TCP | brcd > 9060 [ACK] Seq=1 |
| 38 | 2011-05-05 17:50:13.583388 | 9.37.235.185 | 9.42.135.23 | TCP | [TCP Dup ACK 37#1] brcd |
| 39 | 2011-05-05 17:50:13.583661 | 9.37.235.185 | 9.42.135.23 | TCP | brcd > 9060 [ACK] Seq=1 |
| 40 | 2011-05-05 17:50:13.583689 | 9.37.235.185 | 9.42.135.23 | TCP | [TCP out-of-order] brcd |
| 41 | 2011-05-05 17:50:13.583722 | 9.37.235.185 | 9.42.135.23 | TCP | brcd > 9060 [PSH, ACK] |
| 42 | 2011-05-05 17:50:13.583741 | 9.37.235.185 | 9.42.135.23 | TCP | 9.42.135.23 |
| 43 | 2011-05-05 17:50:13.608574 | 9.42.135.23 | 9.37.235.185 | TCP | 9.42.135.23 |
| 44 | 2011-05-05 17:50:13.610449 | 9.42.135.23 | 9.37.235.185 | TCP | 9.42.135.23 |
| 45 | 2011-05-05 17:50:13.610503 | 9.37.235.185 | 9.42.135.23 | TCP | 9.42.135.23 |

Frame 40 (1514 bytes on wire, 1514 bytes captured)
 Ethernet II, Src: Usi_ce:08:8c (00:1e:37:ce:08:8c), Dst: All-HSRP-
 Internet Protocol, Src: 9.37.235.185 (9.37.235.185), Dst: 9.42.135.
 Transmission Control Protocol, Src Port: brcd (1323), Dst Port: 9060
 Data (1460 bytes)
 Data: 474554202F69626D2F636F6E736F6C652F6C6F67696E2E64...

Result – Packet #40 is first packet found that contains the GET request and is displayed in the packet bytes pane

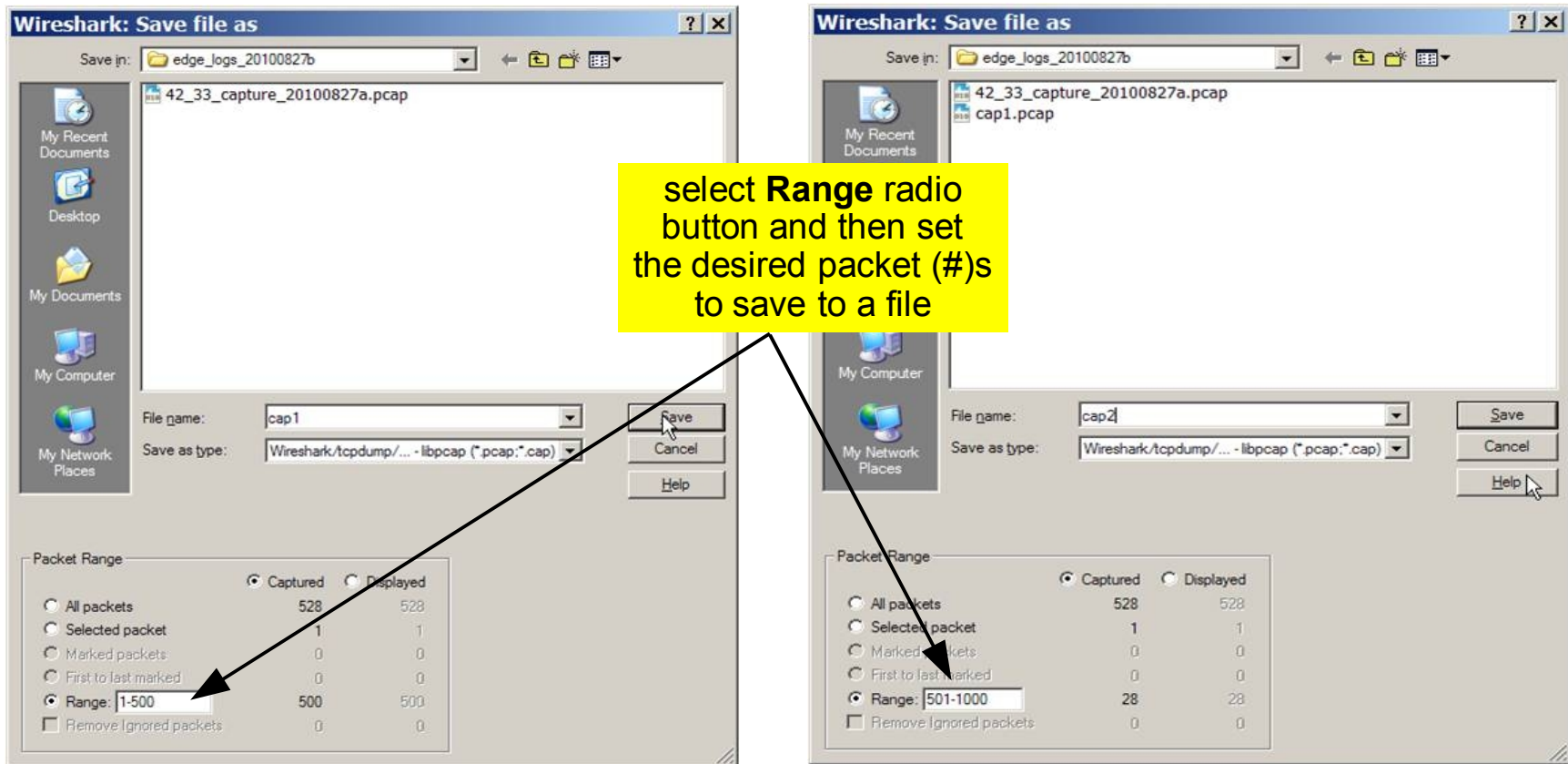
The screenshot shows the 'Edit' menu in Wireshark. The 'Find Next' option is highlighted in blue. A yellow callout box points to this option.

- Copy
- Find Packet... (Ctrl+F)
- Find Next (Ctrl+N)**
- Find Previous (Ctrl+B)
- Mark Packet (toggle) (Ctrl+M)
- Find Next Mark (Shift+Ctrl+N)
- Find Previous Mark (Shift+Ctrl+B)
- Mark All Packets
- Unmark All Packets

Select Edit -> Find Next to find additional packets that contain the search information

A Few Quick Tips

- **Tip #4 - How to Save** a TCP packet capture into *smaller more manageable files*



PACKET CORRELATION with other logs



Packet Correlation - with other logs

- As already mentioned in the previous WSTE presentation titled **TCP Packet Tracing – Part1**, packet tracing can be helpful in debugging many different types of *technical problems*. for example..
 - HTTP Header problems
 - Large file downloads or POST upload problems
 - General TCP connect failures, premature connection closures and packet delays
 - SSL handshake problems
- The purpose of this section will NOT be about *solving* any particular problem. But Instead, the main focus will be to demonstrate *how to find* the **correct request** and **TCP connection** in a **packet trace** using all the information available to you, including information discovered from other WebSphere related logs. Finding the correct request and TCP connection in a packet trace is vital to debugging any of the above mentioned technical problems

Packet Correlation - with other logs

- Needed background **system** information
 - ▶ **IP-addresses**
 - *client* (if known), *server* and *middle device* (if present)
 - ▶ **Ports**
 - *server* and *middle device* (if present)
 - ▶ **Mac-addresses** (*if known*)
 - *all systems*
 - ▶ From *system(s)* where TCP packet trace was collected
 - **Time Zone** (e.g. EDT, CDT)
 - **System Date/Clock Time**
 - Full **URL** of request (if known)
 - **Date/Time of failure** (if known)



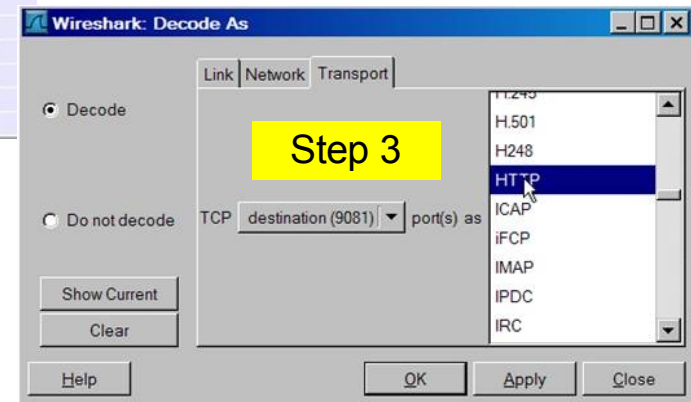
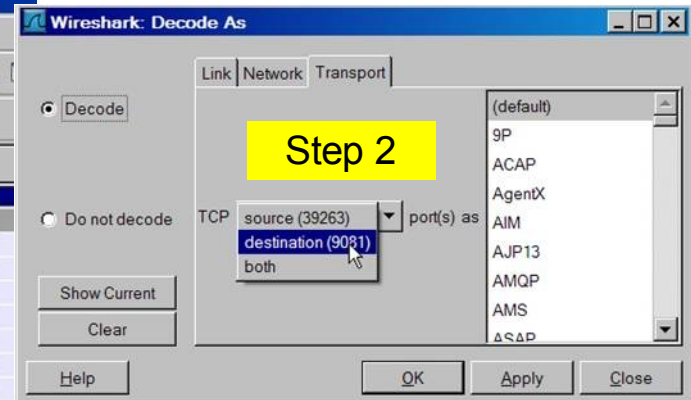
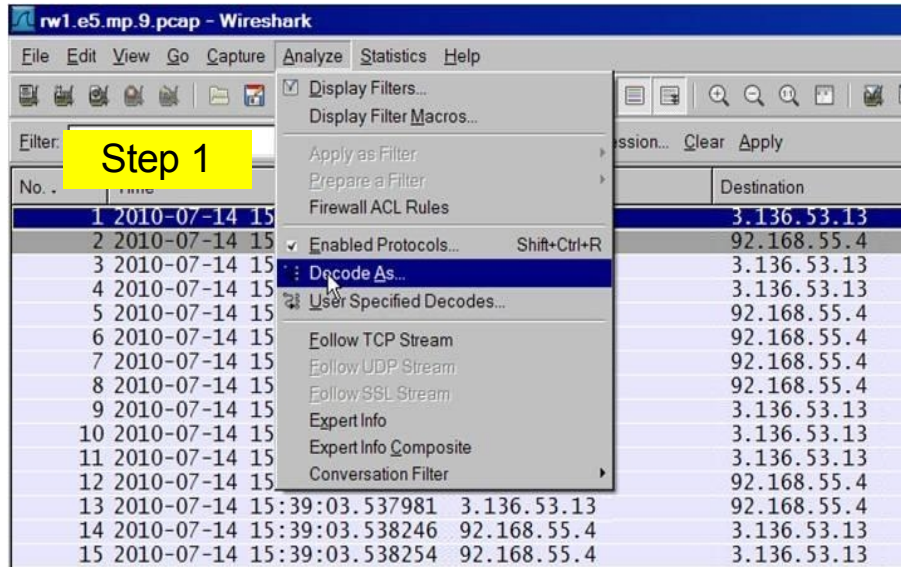
Packet Correlation - with other logs

- Problem#1 - In the WebSphere plugin trace log (http_plugin.log) – 5 second delay

```
[Wed Jul 14 15:47:09 2010] 00007606 b7fcd710 - DEBUG: ws_common: websphereGetStream: socket 14 connected to websphere.host.com:9081
..
[Wed Jul 14 15:47:09 2010] 00007606 b7fcd710 - DETAIL: GET /snoop/ HTTP/1.1
[Wed Jul 14 15:47:09 2010] 00007606 b7fcd710 - DETAIL: Accept */*
[Wed Jul 14 15:47:09 2010] 00007606 b7fcd710 - DETAIL: Referer: https://www.xyz.com/
[Wed Jul 14 15:47:09 2010] 00007606 b7fcd710 - DETAIL: Accept-Language: en-us
[Wed Jul 14 15:47:09 2010] 00007606 b7fcd710 - DETAIL: UA-CPU: x86
[Wed Jul 14 15:47:09 2010] 00007606 b7fcd710 - DETAIL: Accept-Encoding: gzip, deflate
[Wed Jul 14 15:47:09 2010] 00007606 b7fcd710 - DETAIL: User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 5.1; .NET CLR 2.0.50727; .NET CLR 3.0.4506.2152; .NET CLR 3.5.30729; InfoPath.1; MS-RTC LM 8)
[Wed Jul 14 15:47:09 2010] 00007606 b7fcd710 - DETAIL: Host: www.xyz.com
[Wed Jul 14 15:47:09 2010] 00007606 b7fcd710 - DETAIL: Cookie: JSESSIONID=0000vU-Ci1V-ZIXw4HkmlavM2bt
[Wed Jul 14 15:47:09 2010] 00007606 b7fcd710 - DETAIL: $WSIS: true
[Wed Jul 14 15:47:09 2010] 00007606 b7fcd710 - DETAIL: $WSSC: https
[Wed Jul 14 15:47:09 2010] 00007606 b7fcd710 - DETAIL: $WSPR: HTTP/1.1
[Wed Jul 14 15:47:09 2010] 00007606 b7fcd710 - DETAIL: $WSRA: 7.211.65.8
[Wed Jul 14 15:47:09 2010] 00007606 b7fcd710 - DETAIL: $WSRH: 7.211.65.8
[Wed Jul 14 15:47:09 2010] 00007606 b7fcd710 - DETAIL: $WSSN: www.xyz.com
[Wed Jul 14 15:47:09 2010] 00007606 b7fcd710 - DETAIL: $WSSP: 443
[Wed Jul 14 15:47:09 2010] 00007606 b7fcd710 - DETAIL: Surrogate-Capability: WS-ESI="ESI/1.0+"
[Wed Jul 14 15:47:09 2010] 00007606 b7fcd710 - DETAIL: _WS_HAPRT_WLMVERSION: -1
[Wed Jul 14 15:47:09 2010] 00007606 b7fcd710 - TRACE: ws_common: websphereExecute: Wrote the request; reading the response
[Wed Jul 14 15:47:09 2010] 00007606 b7fcd710 - DETAIL: lib_htresponse: htresponseRead: Reading the response: 9f5ebbc
[Wed Jul 14 15:47:14 2010] 00007606 b7fcd710 - DETAIL: HTTP/1.1 200 OK
[Wed Jul 14 15:47:14 2010] 00007606 b7fcd710 - DETAIL: Pragma: no-cache
[Wed Jul 14 15:47:14 2010] 00007606 b7fcd710 - DETAIL: Expires: Thu, 01 Jan 1970 00:00:00 GMT
[Wed Jul 14 15:47:14 2010] 00007606 b7fcd710 - DETAIL: Cache-Control: no-cache
[Wed Jul 14 15:47:14 2010] 00007606 b7fcd710 - DETAIL: Cache-Control: no-store
[Wed Jul 14 15:47:14 2010] 00007606 b7fcd710 - DETAIL: Content-Type: text/html;charset=ISO-8859-1
[Wed Jul 14 15:47:14 2010] 00007606 b7fcd710 - DETAIL: Content-Language: en-US
[Wed Jul 14 15:47:14 2010] 00007606 b7fcd710 - DETAIL: Transfer-Encoding: chunked
[Wed Jul 14 15:47:14 2010] 00007606 b7fcd710 - DETAIL: Date: Wed, 14 Jul 2010 19:47:14 GMT
[Wed Jul 14 15:47:14 2010] 00007606 b7fcd710 - DETAIL: Server: WebSphere Application Server/6.1
```

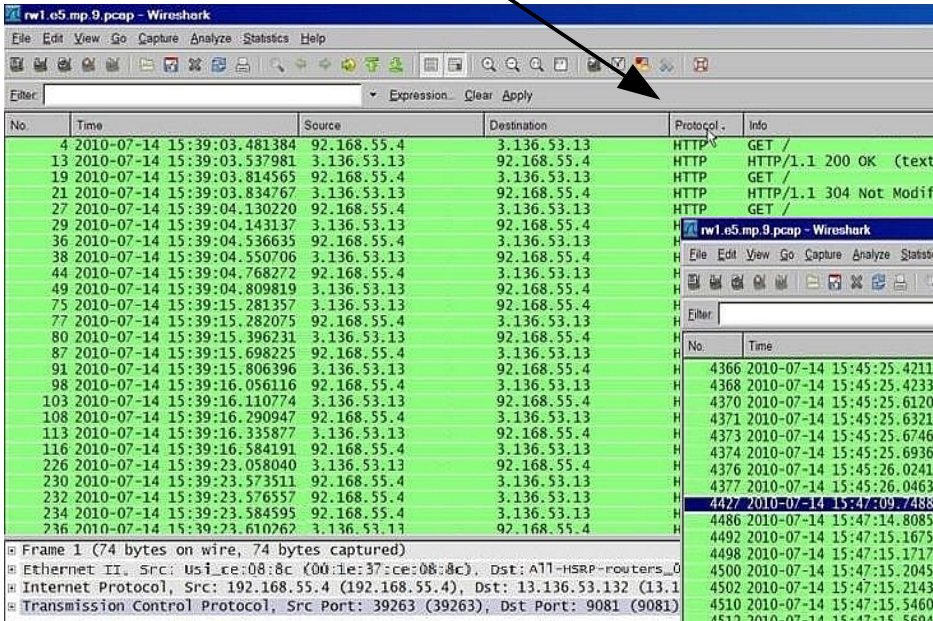
Packet Correlation - with other logs

- Since the delay is between **plugin<-->WebSphere port 9081**, start by **decoding port 9081** traffic as **HTTP**. Decoding as HTTP *allows you to see requests (e.g GET) in info column.*

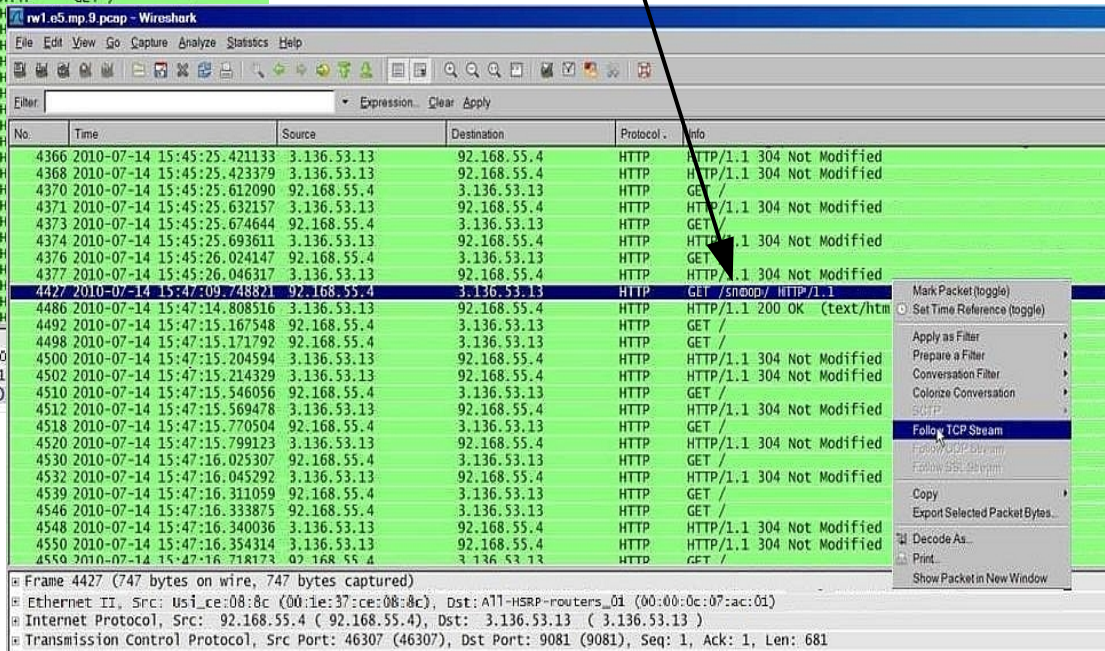


Packet Correlation - with other logs

- Next – click on the **protocol** column so all the HTTP decoded packets show at the top of the trace as shown below.



Then scroll down until you see the packet that contains the *GET* request and *timestamp (15:47:09)* and then *right click* on the packet and select “*follow TCP stream*”



Packet Correlation - with other logs

- The **Stream Content** window appears showing the request(s) handled over this connection. Notice the first request matches the request shown in plugin trace log. Close the window to proceed with further review of the tcp stream/connection.

The screenshot shows the Wireshark interface with a packet list and a 'Follow TCP Stream' window. The packet list shows the following entries:

| No. | Time | Source | Destination | Protocol | Info |
|------|----------------------------|-------------|-------------|----------|-----------------------------|
| 4427 | 2010-07-14 15:47:09.748821 | 92.168.55.4 | 3.136.53.13 | HTTP | GET / |
| 4486 | 2010-07-14 15:47:14.808516 | 3.136.53.13 | 92.168.55.4 | HTTP | HTTP/1.1 200 OK (text/html) |
| 4565 | 2010-07-14 15:47:16.979464 | 92.168.55.4 | 3.136.53.13 | HTTP | GET / |
| 4569 | 2010-07-14 15:47:16.997520 | 3.136.53.13 | 92.168.55.4 | HTTP | HTTP/1.1 304 Not Modified |
| 4703 | 2010-07-14 15:47:23.967362 | 92.168.55.4 | 3.136.53.13 | HTTP | GET / |
| 4705 | 2010-07-14 15:47:23.996427 | 3.136.53.13 | 92.168.55.4 | HTTP | HTTP/1.1 304 Not Modified |

The 'Follow TCP Stream' window displays the following stream content:

```

GET /snoop/ HTTP/1.1
Accept: */*
Referer: https://www.xyz.com/

Accept-Language: en-us
UA-CPU: x86
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; windows NT 5.1; .NET CLR
2.0.50727; .NET CLR 3.0.4506.2152; .NET CLR 3.5.30729; InfoPath.1; MS-RTC LM 8)
Host: www.xyz.com
Cookie: JSESSIONID=0000vU-Ci1V-ZIXw4HkmIavM2bt
$WSIS: true
$WSSC: https
$WSPR: HTTP/1.1
$WSRA: 7.211.65.8
$WSRH: 7.211.65.8
$WSSN: www.xyz.com
$WSSP: 443
Surrogate-Capability: WS-ESI="ESI/1.0+"
_WS_HAPRT_WLMVERSION: -1

HTTP/1.1 200 OK
Pragma: no-cache
Expires: Thu, 01 Jan 1970 00:00:00 GMT
  
```


Packet Correlation - with other logs

- Now, click the **No.** column to sort the connection by packet number. This will ensure the packets are listed in *sequential order*

| No. | Time | Source | Destination |
|------|----------------------------|-------------|-------------|
| 4427 | 2010-07-14 15:47:09.748821 | 92.168.55.4 | 3.136.53.13 |
| 4486 | 2010-07-14 15:47:14.808516 | 3.136.53.13 | 92.168.55.4 |
| 4565 | 2010-07-14 15:47:16.979464 | 92.168.55.4 | 3.136.53.13 |
| 4569 | 2010-07-14 15:47:16.997520 | 3.136.53.13 | 92.168.55.4 |
| 4703 | 2010-07-14 15:47:23.967362 | 92.168.55.4 | 3.136.53.13 |
| 4705 | 2010-07-14 15:47:23.996427 | 3.136.53.13 | 92.168.55.4 |
| 4729 | 2010-07-14 15:47:24.454501 | 92.168.55.4 | 3.136.53.13 |
| 4732 | 2010-07-14 15:47:24.474789 | 3.136.53.13 | 92.168.55.4 |
| 4752 | 2010-07-14 15:47:25.002348 | 92.168.55.4 | 3.136.53.13 |
| 4758 | 2010-07-14 15:47:25.022916 | 3.136.53.13 | 92.168.55.4 |
| 4785 | 2010-07-14 15:47:26.046916 | 92.168.55.4 | 3.136.53.13 |
| 4789 | 2010-07-14 15:47:26.077436 | 3.136.53.13 | 92.168.55.4 |
| 5017 | 2010-07-14 15:47:52.899634 | 92.168.55.4 | 3.136.53.13 |
| 5020 | 2010-07-14 15:47:52.938899 | 3.136.53.13 | 92.168.55.4 |

Results:

- packet no. 4427 shows the GET /snoop/ sent at **15:47:09** by plugin side (ip **92.168.55.4**)
- packet no. 4428 is an immediate ACK from websphere side (ip **3.136.53.13** and port **9081**)
- Packet no. 4433 is the 200 HTTP Response headers from websphere. ****This packet is sent by websphere side 5 seconds after the previous ACK at 15:47:14**

| No. | Time | Source | Destination | Protocol | Info |
|------|----------------------------|-------------|-------------|----------|---|
| 4424 | 2010-07-14 15:47:09.747238 | 92.168.55.4 | 3.136.53.13 | TCP | 46307 > 9081 [SYN] Seq=0 Win=5840 Len=... |
| 4425 | 2010-07-14 15:47:09.748040 | 3.136.53.13 | 92.168.55.4 | TCP | 9081 > 46307 [SYN, ACK] Seq=0 Ack=1 Win=... |
| 4426 | 2010-07-14 15:47:09.748272 | 92.168.55.4 | 3.136.53.13 | TCP | 46307 > 9081 [ACK] Seq=1 Ack=1 Win=5840... |
| 4427 | 2010-07-14 15:47:09.748821 | 92.168.55.4 | 3.136.53.13 | HTTP | GET /snoop/ |
| 4428 | 2010-07-14 15:47:09.749331 | 3.136.53.13 | 92.168.55.4 | TCP | 9081 > 46307 [ACK] Seq=1 Ack=682 Win=... |
| 4433 | 2010-07-14 15:47:14.774372 | 3.136.53.13 | 92.168.55.4 | TCP | [TCP segment of a reassembled PDU] |
| 4434 | 2010-07-14 15:47:14.774009 | 3.136.53.13 | 92.168.55.4 | TCP | [TCP segment of a reassembled PDU] |
| 4435 | 2010-07-14 15:47:14.774019 | 3.136.53.13 | 92.168.55.4 | TCP | [TCP segment of a reassembled PDU] |
| 4436 | 2010-07-14 15:47:14.774363 | 92.168.55.4 | 3.136.53.13 | TCP | 46307 > 9081 [ACK] Seq=682 Ack=1449 W... |
| 4437 | 2010-07-14 15:47:14.774372 | 92.168.55.4 | 3.136.53.13 | TCP | 46307 > 9081 [ACK] Seq=682 Ack=2897 W... |
| 4438 | 2010-07-14 15:47:14.774378 | 92.168.55.4 | 3.136.53.13 | TCP | 46307 > 9081 [ACK] Seq=682 Ack=4345 W... |
| 4439 | 2010-07-14 15:47:14.774957 | 3.136.53.13 | 92.168.55.4 | TCP | [TCP segment of a reassembled PDU] |
| 4440 | 2010-07-14 15:47:14.775007 | 3.136.53.13 | 92.168.55.4 | TCP | [TCP segment of a reassembled PDU] |
| 4441 | 2010-07-14 15:47:14.775058 | 3.136.53.13 | 92.168.55.4 | TCP | [TCP segment of a reassembled PDU] |
| 4442 | 2010-07-14 15:47:14.775067 | 3.136.53.13 | 92.168.55.4 | TCP | [TCP segment of a reassembled PDU] |
| 4443 | 2010-07-14 15:47:14.775108 | 3.136.53.13 | 92.168.55.4 | TCP | [TCP segment of a reassembled PDU] |
| 4444 | 2010-07-14 15:47:14.775117 | 3.136.53.13 | 92.168.55.4 | TCP | [TCP segment of a reassembled PDU] |

Packet Correlation - with other logs

- **Problem #2 – WebSphere Application Server's webcontainer tracing (trace.log) indicates a failed POST due to *invalid* content length**

```
[5/9/11 12:43:50:624 EDT] 00000030 HttpRequestMe 1  setMethod(v): POST
..
[5/9/11 12:43:50:624 EDT] 00000030 HttpRequestMe 3  setRequestURL: set URI to /WSsamples/
..
[5/9/11 12:43:50:626 EDT] 00000030 BNFHeadersImp 1  Adding header [Host] with value [robo.raleigh.ibm.com]
..
[5/9/11 12:43:50:626 EDT] 00000030 BNFHeadersImp 3  Saved token [4875000]
[5/9/11 12:43:50:626 EDT] 00000030 BNFHeadersImp 1  Adding header [Content-Length] with value [4875000]
[5/9/11 12:43:50:626 EDT] 00000030 HttpBaseMessa 1  Adding: Content-Length:4875000
..
[5/9/11 12:43:52:038 EDT] 00000030 srt          1 com.ibm.ws.webcontainer.srt.SRTServletRequest finish
SRVE0189E: Error occurred while finishing request
java.io.IOException: SRVE0080E: Invalid content length
at com.ibm.ws.webcontainer.srt.http.HttpInputStream.finish(HttpInputStream.java:184)
at com.ibm.ws.webcontainer.srt.http.HttpInputStream.close(HttpInputStream.java:532)
at com.ibm.ws.webcontainer.srt.SRTServletRequest.finish(SRTServletRequest.java:2103)
at com.ibm.ws.webcontainer.srt.SRTConnectionContext.finishConnection(SRTConnectionContext.java:80)
at com.ibm.ws.webcontainer.WebContainer.handleRequest(WebContainer.java:1034)
..
[5/9/11 12:43:52:052 EDT] 00000030 AioSocketIOCh 1  AsyncSocketChannel close, local:
aquarius/9.42.135.23:9080 remote: robo.raleigh.ibm.com/9.37.235.185:1200
```

Packet Correlation - with other logs

- display filters **ip.addr eq** and **tcp.port eq** can be used as shown below to quickly find the correct connection in the TCP packet trace
 - (**ip.addr eq 9.37.235.185** and **ip.addr eq 9.42.135.23**) and (**tcp.port eq 1200** and **tcp.port eq 9080**)

The screenshot shows the Wireshark interface with a packet capture filter applied: `ip.port eq 1200 and tcp.port eq 9080`. The main packet list shows several packets, with packet 339 selected. The detailed view of packet 339 shows the following content:

```
POST /wssamples/ HTTP/1.1
User-Agent: Fiddler
Host: robo.raleigh.ibm.com
Content-Length: 4875000
Content-Type: text/xml
$WSIS: false
$WSSC: http
$WSPR: HTTP/1.1
$WSRA: 9.37.235.185
$WSRH: 9.37.235.185
$WSSN: robo.raleigh.ibm.com
$WSSP: 80
Surrogate-Capability: WS-ESI="ESI/1.0+"
_WS_HAPRT_WLMVERSION: -1
Expect: 100-Continue

HTTP/1.1 100 Continue
Content-Length: 0
Date: Mon, 09 May 2011 16:43:47 GMT
Server: WebSphere Application Server/7.0

<sim:GoToLoopback>
<test>blahb!ahb!ah</test>
</sim:GoToLoopback>
</soapenv:Body>
```

The right pane shows the TCP stream content, including sequence and acknowledgment numbers for various packets.

Packet Correlation - with other logs

- On the *web server side* in the **WebSphere plugin trace log** (`http_plugin.log`) we see the following error and reason why WebSphere side reported **SRVE0080E: Invalid content length**

```
[Mon May 09 12:43:50 2011] 00001c50 00000c04 - DETAIL: POST /WSsamples/ HTTP/1.1
[Mon May 09 12:43:50 2011] 00001c50 00000c04 - DETAIL: User-Agent: Fiddler
[Mon May 09 12:43:50 2011] 00001c50 00000c04 - DETAIL: Host: robo.raleigh.ibm.com
[Mon May 09 12:43:50 2011] 00001c50 00000c04 - DETAIL: Content-Length: 4875000
[Mon May 09 12:43:50 2011] 00001c50 00000c04 - DETAIL: Content-Type: text/xml
[Mon May 09 12:43:50 2011] 00001c50 00000c04 - DETAIL: $WSIS: false
[Mon May 09 12:43:50 2011] 00001c50 00000c04 - DETAIL: $WSSC: http
[Mon May 09 12:43:50 2011] 00001c50 00000c04 - DETAIL: $WSPR: HTTP/1.1
[Mon May 09 12:43:50 2011] 00001c50 00000c04 - DETAIL: $WSRA: 9.37.235.1
[Mon May 09 12:43:50 2011] 00001c50 00000c04 - DETAIL: $WSRH: 9.37.235.1
[Mon May 09 12:43:50 2011] 00001c50 00000c04 - DETAIL: $WSSN: robo.raleigh.ibm.com
[Mon May 09 12:43:50 2011] 00001c50 00000c04 - DETAIL: $WSSP: 80
[Mon May 09 12:43:50 2011] 00001c50 00000c04 - DETAIL: Surrogate-Capability: WS-ESI="ESI/1.0+"
[Mon May 09 12:43:50 2011] 00001c50 00000c04 - DETAIL: _WS_HAPRT_WLMVERSION: -1
[Mon May 09 12:43:50 2011] 00001c50 00000c04 - DETAIL: Expect: 100-Continue
[Mon May 09 12:43:50 2011] 00001c50 00000c04 - DEBUG: lib_htrequest: htrequestWrite: Waiting for the continue response
[Mon May 09 12:43:50 2011] 00001c50 00000c04 - DETAIL: HTTP/1.1 100 Continue
[Mon May 09 12:43:50 2011] 00001c50 00000c04 - DETAIL: Content-Length: 0
[Mon May 09 12:43:50 2011] 00001c50 00000c04 - DETAIL: Date: Mon, 09 May 2011 16:43:47 GMT
[Mon May 09 12:43:50 2011] 00001c50 00000c04 - DETAIL: Server: WebSphere Application Server/7.0
[Mon May 09 12:43:50 2011] 00001c50 00000c04 - DEBUG: lib_htrequest: htrequestWrite: Writing the request content
[Mon May 09 12:43:50 2011] 00001c50 00000c04 - TRACE: lib_htrequest: htrequestWrite: content length is 4875000
..
[Mon May 09 12:43:52 2011] 00001c50 00000c04 - TRACE: lib_htrequest: htreqesWrite: Read 302200 of the expected 4875000 bytes so far
[Mon May 09 12:43:52 2011] 00001c50 00000c04 - TRACE: mod_was_ap20_http: cb_read_body: In the read body callback
[Mon May 09 12:43:52 2011] 00001c50 00000c04 - TRACE: mod_was_ap20_http: cb_read_body: Failed to read the full body from the browser. just_read
= 0 of the expected 65536
[Mon May 09 12:43:52 2011] 00001c50 00000c04 - TRACE: lib_htrequest: htrequestSetError: Setting the error to: |READ_FAILED|(1, Line: 1625)
[Mon May 09 12:43:52 2011] 00001c50 00000c04 - WARNING: ws_common: websphereExecute: Error reading post data from client
```

Packet Correlation - with other logs

- **Problem #3 – WebSphere plugin trace log (http_plugin.log) on web server 192.168.185.2 reports POST body read failure from client side @ 07:45:31 CDT**

```
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - DEBUG: lib_htrequest: htrequestWrite: Waiting for the continue response
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - DETAIL: HTTP/1.1 100 Continue
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - DETAIL: Content-Length: 0
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - DETAIL: Date: Thu, 18 Mar 2010 12:45:31 GMT
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - DETAIL: Server: WebSphere Application Server/6.1
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - DEBUG: lib_htrequest: htrequestWrite: Writing the request content
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - TRACE: lib_htrequest: htrequestWrite: content length is 50266
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - TRACE: lib_htrequest: htrequestWrite: Allocating buffer of 50266 for POST content
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - TRACE: mod_was_ap20_http: cb_read_body: In the read body callback
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - TRACE: mod_was_ap20_http: cb_read_body: Read from IHS client 50266 - available 50266
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - TRACE: lib_htrequest: htrequestWrite: Read 1460 of the expected 50266 bytes so far
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - TRACE: mod_was_ap20_http: cb_read_body: In the read body callback
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - TRACE: lib_htrequest: htrequestWrite: Read 2920 of the expected 50266 bytes so far
..
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - TRACE: mod_was_ap20_http: cb_read_body: In the read body callback
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - TRACE: lib_htrequest: htrequestWrite: Read 49640 of the expected 50266 bytes so far
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - TRACE: mod_was_ap20_http: cb_read_body: In the read body callback
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - WARNING: mod_was_ap20_http: cb_read_body: Failed to read the full
body from the browser. just_read = -1 of the expected 626
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - TRACE: lib_htrequest: htrequestSetError: Setting the error to: [READ_FAILED](1, Line: 1625)
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - WARNING: ws_common: websphereExecute: Error reading post data from client
```


Packet Correlation - with other logs

- By following the thread *backwards* the **request** is revealed which provides useful information for parsing the TCP packet trace

```
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - DETAIL: POST /snoop/ HTTP/1.1
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - DETAIL: Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, application/x-shockwave-flash, application/vnd.ms-excel,
application/vnd.ms-powerpoint, application/msword, application/xhtml+xml, application/vnd.ms-xpsdocument, application/x-ms-xbap, application/x-ms-application, */*
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - DETAIL: Accept-Language: en-us
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - DETAIL: Content-Type: application/x-www-form-urlencoded
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - DETAIL: Accept-Encoding: gzip, deflate
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - DETAIL: User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; AAFES IE6 SP1 Build 1.2; .NET CLR 1.1.4322; .NET CLR
2.0.50727; .NET CLR 3.0.04506.30)
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - DETAIL: Host: www.xyz.com
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - DETAIL: Content-Length: 50266
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - DETAIL: Cache-Control: no-cache
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - DETAIL: Cookie: JSESSIONID=0000ZaGcCpWO279w0-n9l5Wft0U
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - DETAIL: $WSIS: false
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - DETAIL: $WSSC: http
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - DETAIL: $WSPPR: HTTP/1.1
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - DETAIL: $WSRA: 10.12.5.6
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - DETAIL: $WSRH: 10.12.5.6
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - DETAIL: $WSSN: www.xyz.com
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - DETAIL: $WSSP: 8080
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - DETAIL: Surrogate-Capability: WS-ESI="ESI/1.0+"
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - DETAIL: _WS_HAPRT_WLMVERSION: -1
[Thu Mar 18 07:45:31 2010] 0008406e 00000506 - DETAIL: Expect: 100-Continue
```

I

Packet Correlation - with other logs

- For starters, use a display filter for the (client ip and web server port) and the web server ip
 - (ip.addr eq 10.12.5.6 and ip.addr eq 192.168.185.2) and (tcp.port eq 8080)

Filter: (ip.addr eq 10.12.5.6) and (ip.addr eq 192.168.185.2)

| No. | Time | Source | Destination | Protocol | Info |
|-------|-------------------------------|---------------|---------------|----------|---|
| 40162 | 2010-03-18 08:43:47.456258884 | 10.12.5.6 | 192.168.185.2 | TCP | dvl-activemail > http-alt [SYN] Seq=0 Win=65535 Len=0 MSS=1460 |
| 40163 | 2010-03-18 08:43:47.456261767 | 192.168.185.2 | 10.12.5.6 | TCP | http-alt > dvl-activemail [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 |
| 40164 | 2010-03-18 08:43:47.47669269 | 10.12.5.6 | 192.168.185.2 | TCP | dvl-activemail > http-alt [ACK] Seq=1 Ack=1 Win=65535 Len=0 |
| 40165 | 2010-03-18 08:43:47.481952093 | 10.12.5.6 | 192.168.185.2 | HTTP | GET / |
| 40170 | 2010-03-18 08:43:47.527321267 | 192.168.185.2 | 10.12.5.6 | TCP | [TCP segment of a reassembled PDU] |
| 40171 | 2010-03-18 08:43:47.527321234 | 192.168.185.2 | 10.12.5.6 | HTTP | HTTP/1.1 302 Found (text/html) |
| 40172 | 2010-03-18 08:43:47.551338898 | 10.12.5.6 | 192.168.185.2 | TCP | dvl-activemail > http-alt [ACK] Seq=930 Ack=606 Win=64930 Len=0 |
| 40173 | 2010-03-18 08:43:47.557227125 | 10.12.5.6 | 192.168.185.2 | HTTP | GET / |
| 40177 | 2010-03-18 08:43:47.605148119 | 192.168.185.2 | 10.12.5.6 | TCP | [TCP segment of a reassembled PDU] |
| 40178 | 2010-03-18 08:43:47.605337966 | 192.168.185.2 | 10.12.5.6 | HTTP | HTTP/1.1 302 Found (text/html) |
| 40179 | 2010-03-18 08:43:47.628821515 | 10.12.5.6 | 192.168.185.2 | TCP | dvl-activemail > http-alt [ACK] Seq=1712 Ack=1162 Win=64374 Len=0 |
| 40180 | 2010-03-18 08:43:47.634100609 | 10.12.5.6 | 192.168.185.2 | HTTP | GET / |
| 40181 | 2010-03-18 08:43:47.644832582 | 192.168.185.2 | 10.12.5.6 | HTTP | HTTP/1.1 302 Found (text/html) |
| 40188 | 2010-03-18 08:43:47.644832582 | 192.168.185.2 | 10.12.5.6 | HTTP | Continuation of non-HTTP traffic |
| 40189 | 2010-03-18 08:43:47.681042478 | 10.12.5.6 | 192.168.185.2 | TCP | dvl-activemail > http-alt [ACK] Seq=2410 Ack=4082 Win=65535 Len=0 |
| 40190 | 2010-03-18 08:43:47.688817330 | 10.12.5.6 | 192.168.185.2 | TCP | dvl-activemail > http-alt [ACK] Seq=2410 Ack=5505 Win=64112 Len=0 |
| 40191 | 2010-03-18 08:43:47.694299333 | 10.12.5.6 | 192.168.185.2 | HTTP | GET / |
| 40194 | 2010-03-18 08:43:47.885538566 | 192.168.185.2 | 10.12.5.6 | TCP | http-alt > dvl-activemail [ACK] Seq=5505 Ack=3077 Win=65535 Len=0 |
| 40225 | 2010-03-18 08:43:48.017169201 | 192.168.185.2 | 10.12.5.6 | HTTP | HTTP/1.1 200 OK [unreassembled packet [incorrect TCP checksum]] |
| 40291 | 2010-03-18 08:43:48.653319716 | 10.12.5.6 | 192.168.185.2 | TCP | dvl-activemail > http-alt [ACK] Seq=3077 Ack=8425 Win=65535 Len=0 |
| 40292 | 2010-03-18 08:43:48.668991607 | 10.12.5.6 | 192.168.185.2 | TCP | dvl-activemail > http-alt [ACK] Seq=3077 Ack=11345 Win=65535 Len=0 |
| 40293 | 2010-03-18 08:43:48.684766183 | 10.12.5.6 | 192.168.185.2 | TCP | dvl-activemail > http-alt [ACK] Seq=3077 Ack=14265 Win=65535 Len=0 |
| 40294 | 2010-03-18 08:43:48.701060380 | 10.12.5.6 | 192.168.185.2 | TCP | dvl-activemail > http-alt [ACK] Seq=3077 Ack=17185 Win=65535 Len=0 |
| 40299 | 2010-03-18 08:43:48.701060380 | 192.168.185.2 | 10.12.5.6 | HTTP | Continuation of non-HTTP traffic |
| 40296 | 2010-03-18 08:43:48.706275941 | 10.12.5.6 | 192.168.185.2 | TCP | audio-activemail > http-alt [SYN] Seq=0 Win=65535 Len=0 MSS=1460 |
| 40297 | 2010-03-18 08:43:48.706278580 | 192.168.185.2 | 10.12.5.6 | TCP | http-alt > audio-activemail [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 |
| 40299 | 2010-03-18 08:43:48.73486205 | 10.12.5.6 | 192.168.185.2 | TCP | dvl-activemail > http-alt [ACK] Seq=3077 Ack=20105 Win=65535 Len=0 |
| 40300 | 2010-03-18 08:43:48.737682794 | 10.12.5.6 | 192.168.185.2 | TCP | audio-activemail > http-alt [ACK] Seq=1 Ack=1 Win=65535 Len=0 |
| 40301 | 2010-03-18 08:43:48.740769740 | 10.12.5.6 | 192.168.185.2 | HTTP | GET / |

* Frame 49512 (60 bytes on wire, 60 bytes captured)
 * Ethernet II, Src: Us1_c0:08:8c (00:1e:37:ce:08:8c), Dst: All-MSRP-routers_01 (00:00:0c:07:ac:01)
 * Internet Protocol, Src: 10.12.5.6 (10.12.5.6), Dst: 192.168.185.2 (192.168.185.2)
 * Transmission Control Protocol, Src Port: prm-sm-np (1402), Dst Port: http-alt (8080), Seq: 50809, Ack: 1, Len: 0

```

0000  00 1a 64 a8 85 a6 00 01 64 f9 1a 01 08 00 45 00  ..d....d....E.
0010  00 28 46 54 40 00 79 06 32 6f 0a 0c 05 43 c0 a8  ..(FT0.y. 2o...C.
0020  b9 15 05 7a 1f 90 9c 10 db c2 5e 54 6f 48 50 14  ...z....AToHP.
0030  00 00 bc 49 00 00 00 00 00 00 00 00 00 00 00 00  ...I.....
    
```


Packet Correlation - with other logs

Now, scroll down to the time that matches the POST read failure seen in plugin trace which is **7:45:31CDT** or in this case, it would be **8:45:31 EDT**. Notice the **RST from the client** at the time of the failure. *Note: RST is a forced reset of a connection at the TCP layer*

| No. | Time | Source | Destination | Protocol | Info |
|-------|-------------------------------|---------------|---------------|----------|--------------------------------------|
| 49491 | 2010-03-18 08:45:31.740908861 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49492 | 2010-03-18 08:45:31.748895824 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49493 | 2010-03-18 08:45:31.748913314 | 192.168.185.2 | 10.12.5.6 | TCP | http-alt > prm-sm-np [ACK] Seq=1 Ack |
| 49496 | 2010-03-18 08:45:31.756682980 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49497 | 2010-03-18 08:45:31.764669121 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49498 | 2010-03-18 08:45:31.764686396 | 192.168.185.2 | 10.12.5.6 | TCP | http-alt > prm-sm-np [ACK] Seq=1 Ack |
| 49501 | 2010-03-18 08:45:31.772556673 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49502 | 2010-03-18 08:45:31.780443794 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49503 | 2010-03-18 08:45:31.780462070 | 192.168.185.2 | 10.12.5.6 | TCP | http-alt > prm-sm-np [ACK] Seq=1 Ack |
| 49506 | 2010-03-18 08:45:31.788330000 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49507 | 2010-03-18 08:45:31.796216455 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49508 | 2010-03-18 08:45:31.796233859 | 192.168.185.2 | 10.12.5.6 | TCP | http-alt > prm-sm-np [ACK] Seq=1 Ack |
| 49511 | 2010-03-18 08:45:31.804110513 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49512 | 2010-03-18 08:45:31.804112111 | 10.12.5.6 | 192.168.185.2 | TCP | prm-sm-np > http-alt [RST, ACK] Seq= |

Right click on the RST packet and select **follow TCP stream** to see the entire conversation between this client and the web server

| No. | Time | Source | Destination | Protocol | Info |
|-------|-------------------------------|---------------|---------------|----------|--------------------------------------|
| 49491 | 2010-03-18 08:45:31.740908861 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49492 | 2010-03-18 08:45:31.748895824 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49493 | 2010-03-18 08:45:31.748913314 | 192.168.185.2 | 10.12.5.6 | TCP | http-alt > prm-sm-np [ACK] Seq=1 Ack |
| 49496 | 2010-03-18 08:45:31.756682980 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49497 | 2010-03-18 08:45:31.764669121 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49498 | 2010-03-18 08:45:31.764686396 | 192.168.185.2 | 10.12.5.6 | TCP | http-alt > prm-sm-np [ACK] Seq=1 Ack |
| 49501 | 2010-03-18 08:45:31.772556673 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49502 | 2010-03-18 08:45:31.780443794 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49503 | 2010-03-18 08:45:31.780462070 | 192.168.185.2 | 10.12.5.6 | TCP | http-alt > prm-sm-np [ACK] Seq=1 Ack |
| 49506 | 2010-03-18 08:45:31.788330000 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49507 | 2010-03-18 08:45:31.796216455 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49508 | 2010-03-18 08:45:31.796233859 | 192.168.185.2 | 10.12.5.6 | TCP | http-alt > prm-sm-np [ACK] Seq=1 Ack |
| 49511 | 2010-03-18 08:45:31.804110513 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49512 | 2010-03-18 08:45:31.804112111 | 10.12.5.6 | 192.168.185.2 | TCP | prm-sm-np > http-alt [RST, ACK] Seq= |
| 49516 | 2010-03-18 08:45:31.806119275 | 10.12.5.6 | 192.168.185.2 | TCP | prm-nm-np > http-alt [SYN] Seq=0 win |
| 49517 | 2010-03-18 08:45:31.806120941 | 192.168.185.2 | 10.12.5.6 | TCP | http-alt > prm-nm-np [SYN, ACK] Seq= |
| 49518 | 2010-03-18 08:45:31.826666556 | 10.12.5.6 | 192.168.185.2 | TCP | prm-nm-np > http-alt [ACK] Seq=1 Ack |
| 49519 | 2010-03-18 08:45:31.833256496 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |

Packet Correlation - with other logs

- Now that the connection/stream is filtered, we see the request(s) handled over this connection

The image shows a Wireshark interface with a packet capture filtered by '(ip.addr eq 10.12.5.6 and ip.addr eq 192.168.185.2)'. The packet list shows several TCP segments. A 'Follow TCP Stream' window is open, displaying the stream content of a POST request to /snoop/IHTTP/1.1. The request includes various headers such as Accept, Accept-Language, Content-Type, User-Agent, Host, Content-Length, Connection, Cache-Control, and Cookie.

| No. | Time | Source | Destination | Protocol | Info |
|-------|-------------------------------|---------------|---------------|----------|--|
| 49466 | 2010-03-18 08:45:31.662043050 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49467 | 2010-03-18 08:45:31.669927552 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49468 | 2010-03-18 08:45:31.669946292 | 192.168.185.2 | 10.12.5.6 | TCP | http-alt > prm-sm-np [ACK] Seq=1 Ack=25989 |
| 49471 | 2010-03-18 08:45:31.677817612 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49472 | 2010-03-18 08:45:31.677817612 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49473 | 2010-03-18 08:45:31.677817612 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49476 | 2010-03-18 08:45:31.677817612 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49477 | 2010-03-18 08:45:31.677817612 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49478 | 2010-03-18 08:45:31.677817612 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49481 | 2010-03-18 08:45:31.677817612 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49482 | 2010-03-18 08:45:31.677817612 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49483 | 2010-03-18 08:45:31.677817612 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49486 | 2010-03-18 08:45:31.677817612 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49487 | 2010-03-18 08:45:31.677817612 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49488 | 2010-03-18 08:45:31.677817612 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49491 | 2010-03-18 08:45:31.677817612 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49492 | 2010-03-18 08:45:31.677817612 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49493 | 2010-03-18 08:45:31.677817612 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49496 | 2010-03-18 08:45:31.677817612 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49497 | 2010-03-18 08:45:31.677817612 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49498 | 2010-03-18 08:45:31.677817612 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49501 | 2010-03-18 08:45:31.677817612 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49502 | 2010-03-18 08:45:31.677817612 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49503 | 2010-03-18 08:45:31.677817612 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49506 | 2010-03-18 08:45:31.677817612 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49507 | 2010-03-18 08:45:31.677817612 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |

Stream Content

```

POST /snoop/IHTTP/1.1
Accept: image/gif, image/x-bitmap, image/jpeg, image/pjpeg, application/x-shockwave-flash,
application/vnd.ms-excel, application/vnd.ms-powerpoint, application/msword, application/xaml+xml,
application/vnd.ms-xpsdocument, application/x-ms-xbap, application/x-ms-application, */*
Accept-Language: en-us
Content-Type: application/x-www-form-urlencoded
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; AAFES IE6 SP1 Build 2600.5512; .NET CLR 1.1.4322; .NET CLR 2.0.50727; .NET CLR 3.0.04506.30)
Host: www.xyz.com
Content-Length: 50266
Connection: Keep-Alive
Cache-Control: no-cache
Cookie: JSESSIONID=0000ZaGcCpW0279w0-n9I5Wft0U
    
```

Packet Correlation - with other logs

- the packet details pane shows number of **data bytes (1460 bytes)** contained in each client packet

| | | | | | | |
|--|------------|--------------------|---------------|---------------|-----|--|
| 49463 | 2010-03-18 | 08:45:31.654168736 | 192.168.185.2 | 10.12.5.6 | TCP | http-alt > prm-sm-np [ACK] Seq=1 Ack=23069 |
| 49466 | 2010-03-18 | 08:45:31.662043050 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49467 | 2010-03-18 | 08:45:31.669927552 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| # Frame 49427 (1514 bytes on wire, 1514 bytes captured) | | | | | | |
| # Ethernet II, Src: us1.ce:08:8c (00:1e:37:ce:08:8c), Dst: A11-HSRP-routers_01 (00:00:0c:07:ac:01) | | | | | | |
| # Internet Protocol, Src: 10.12.5.6 (10.12.5.6), Dst: 192.168.185.2 (192.168.185.2) | | | | | | |
| # Transmission Control Protocol, Src Port: prm-sm-np (1402), Dst Port: http-alt (8080), Seq: 1169, Ack: 1, Len: 1460 | | | | | | |
| Source port: prm-sm-np (1402) | | | | | | |
| Destination port: http-alt (8080) | | | | | | |
| Sequence number: 1169 (relative sequence number) | | | | | | |
| [Next sequence number: 2629 (relative sequence number)] | | | | | | |
| Acknowledgement number: 1 (relative ack number) | | | | | | |
| Header length: 20 bytes | | | | | | |
| # Flags: 0x10 (ACK) | | | | | | |
| Window size: 65535 | | | | | | |
| # Checksum: 0x7f37 [correct] | | | | | | |
| TCP segment data (1460 bytes) | | | | | | |

Packet no. 49420 contains the **POST headers** for this failed request.

Finally, Count the number of **body data** packets from the client starting with the first packet no. 49427 to the last packet no. 49511

The total is 34 packets (each containing 1460 bytes of body data) **34 x 1460 = 49640 bytes** of data received before the. RST happened at packet no. 49512. This matches what was recorded in the plugin trace log just before the error.

[Thu Mar 18 07:45:31 2010] 00084066
 00000506 - TRACE: lib_htrequest:
 htrequestWrite:
 Read 49640 of the expected 50266 bytes
 so far

| | | | | | | |
|-------|------------|--------------------|---------------|---------------|-----|--|
| 49420 | 2010-03-18 | 08:45:31.469259335 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49427 | 2010-03-18 | 08:45:31.477244007 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49430 | 2010-03-18 | 08:45:31.505856148 | 192.168.185.2 | 10.12.5.6 | TCP | http-alt > prm-sm-np [ACK] Seq=1 Ack=2629 |
| 49432 | 2010-03-18 | 08:45:31.542336865 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49433 | 2010-03-18 | 08:45:31.542356539 | 192.168.185.2 | 10.12.5.6 | TCP | http-alt > prm-sm-np [ACK] Seq=1 Ack=5549 |
| 49436 | 2010-03-18 | 08:45:31.550224318 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49437 | 2010-03-18 | 08:45:31.570392029 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49438 | 2010-03-18 | 08:45:31.570411535 | 192.168.185.2 | 10.12.5.6 | TCP | http-alt > prm-sm-np [ACK] Seq=1 Ack=8469 |
| 49441 | 2010-03-18 | 08:45:31.578277212 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49442 | 2010-03-18 | 08:45:31.586164048 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49443 | 2010-03-18 | 08:45:31.586182256 | 192.168.185.2 | 10.12.5.6 | TCP | http-alt > prm-sm-np [ACK] Seq=1 Ack=11389 |
| 49446 | 2010-03-18 | 08:45:31.598945660 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49447 | 2010-03-18 | 08:45:31.606846115 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49448 | 2010-03-18 | 08:45:31.606870621 | 192.168.185.2 | 10.12.5.6 | TCP | http-alt > prm-sm-np [ACK] Seq=1 Ack=14309 |
| 49451 | 2010-03-18 | 08:45:31.614724916 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49452 | 2010-03-18 | 08:45:31.622610640 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49453 | 2010-03-18 | 08:45:31.622634673 | 192.168.185.2 | 10.12.5.6 | TCP | http-alt > prm-sm-np [ACK] Seq=1 Ack=17229 |
| 49456 | 2010-03-18 | 08:45:31.630495283 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49457 | 2010-03-18 | 08:45:31.638379074 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49458 | 2010-03-18 | 08:45:31.638398128 | 192.168.185.2 | 10.12.5.6 | TCP | http-alt > prm-sm-np [ACK] Seq=1 Ack=20149 |
| 49461 | 2010-03-18 | 08:45:31.646264884 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49462 | 2010-03-18 | 08:45:31.654151550 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49463 | 2010-03-18 | 08:45:31.654168736 | 192.168.185.2 | 10.12.5.6 | TCP | http-alt > prm-sm-np [ACK] Seq=1 Ack=23069 |
| 49466 | 2010-03-18 | 08:45:31.662043050 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49467 | 2010-03-18 | 08:45:31.669927552 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49468 | 2010-03-18 | 08:45:31.669946292 | 192.168.185.2 | 10.12.5.6 | TCP | http-alt > prm-sm-np [ACK] Seq=1 Ack=25989 |
| 49471 | 2010-03-18 | 08:45:31.677817642 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49472 | 2010-03-18 | 08:45:31.685701710 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49473 | 2010-03-18 | 08:45:31.685720894 | 192.168.185.2 | 10.12.5.6 | TCP | http-alt > prm-sm-np [ACK] Seq=1 Ack=28909 |
| 49476 | 2010-03-18 | 08:45:31.693590382 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49477 | 2010-03-18 | 08:45:31.701474717 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49478 | 2010-03-18 | 08:45:31.701494724 | 192.168.185.2 | 10.12.5.6 | TCP | http-alt > prm-sm-np [ACK] Seq=1 Ack=31829 |
| 49481 | 2010-03-18 | 08:45:31.709360861 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49482 | 2010-03-18 | 08:45:31.717248548 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49483 | 2010-03-18 | 08:45:31.727265953 | 192.168.185.2 | 10.12.5.6 | TCP | http-alt > prm-sm-np [ACK] Seq=1 Ack=34749 |
| 49486 | 2010-03-18 | 08:45:31.725133193 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49487 | 2010-03-18 | 08:45:31.733020832 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49488 | 2010-03-18 | 08:45:31.733040916 | 192.168.185.2 | 10.12.5.6 | TCP | http-alt > prm-sm-np [ACK] Seq=1 Ack=37669 |
| 49491 | 2010-03-18 | 08:45:31.740908861 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49492 | 2010-03-18 | 08:45:31.748895824 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49493 | 2010-03-18 | 08:45:31.780443794 | 192.168.185.2 | 10.12.5.6 | TCP | http-alt > prm-sm-np [ACK] Seq=1 Ack=40589 |
| 49496 | 2010-03-18 | 08:45:31.756682980 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49497 | 2010-03-18 | 08:45:31.764669121 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49498 | 2010-03-18 | 08:45:31.764686396 | 192.168.185.2 | 10.12.5.6 | TCP | http-alt > prm-sm-np [ACK] Seq=1 Ack=43509 |
| 49501 | 2010-03-18 | 08:45:31.772556673 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49502 | 2010-03-18 | 08:45:31.780443794 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49503 | 2010-03-18 | 08:45:31.780462070 | 192.168.185.2 | 10.12.5.6 | TCP | http-alt > prm-sm-np [ACK] Seq=1 Ack=46429 |
| 49506 | 2010-03-18 | 08:45:31.788330000 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49507 | 2010-03-18 | 08:45:31.796216455 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49508 | 2010-03-18 | 08:45:31.796233859 | 192.168.185.2 | 10.12.5.6 | TCP | http-alt > prm-sm-np [ACK] Seq=1 Ack=49349 |
| 49511 | 2010-03-18 | 08:45:31.804110513 | 10.12.5.6 | 192.168.185.2 | TCP | [TCP segment of a reassembled PDU] |
| 49512 | 2010-03-18 | 08:45:31.804112111 | 10.12.5.6 | 192.168.185.2 | TCP | prm-sm-np > http-alt [RST, ACK] Seq=50800 |



DECRYPTION

How to decrypt SSL packets

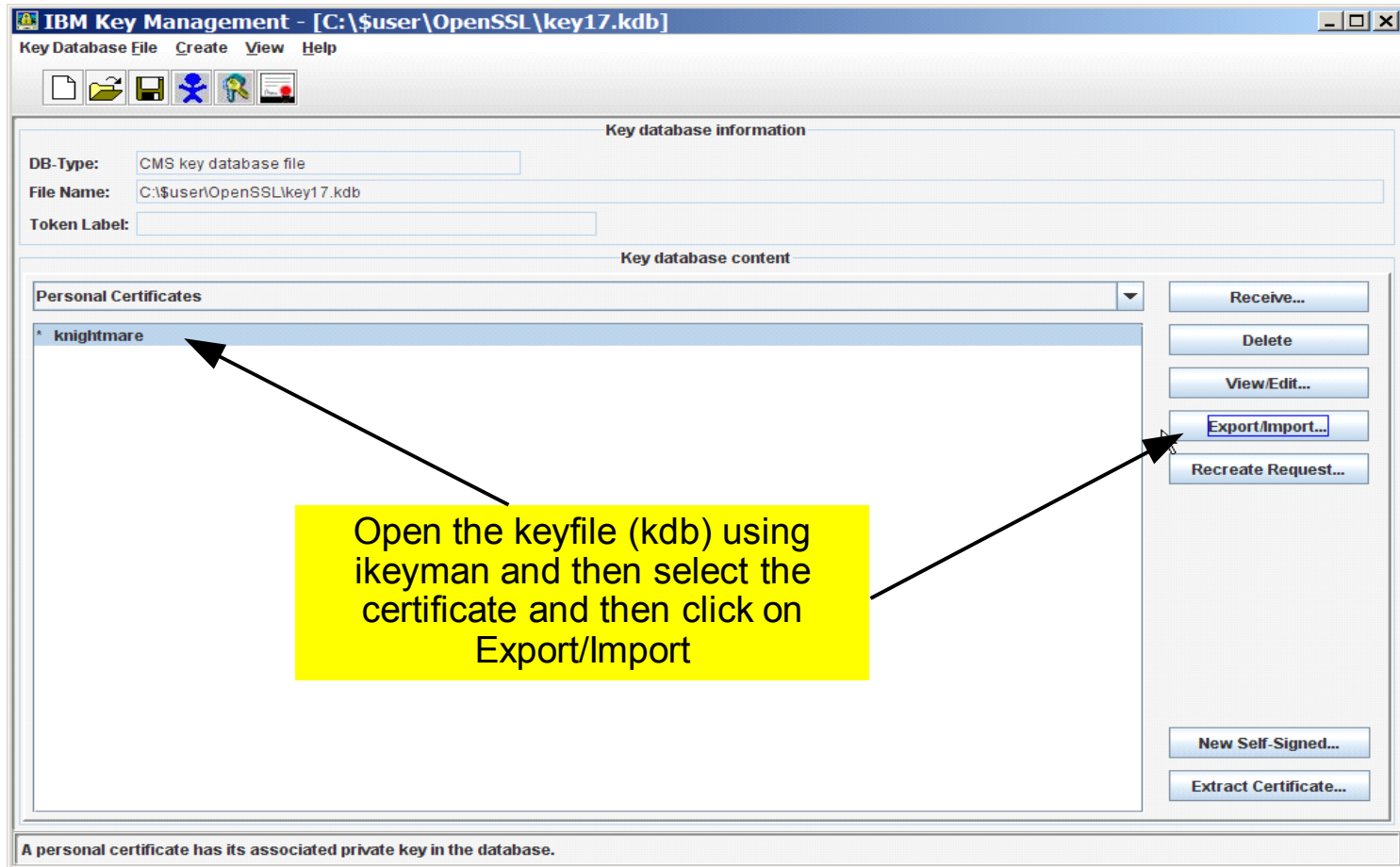


Decryption - How to *decrypt* SSL packets

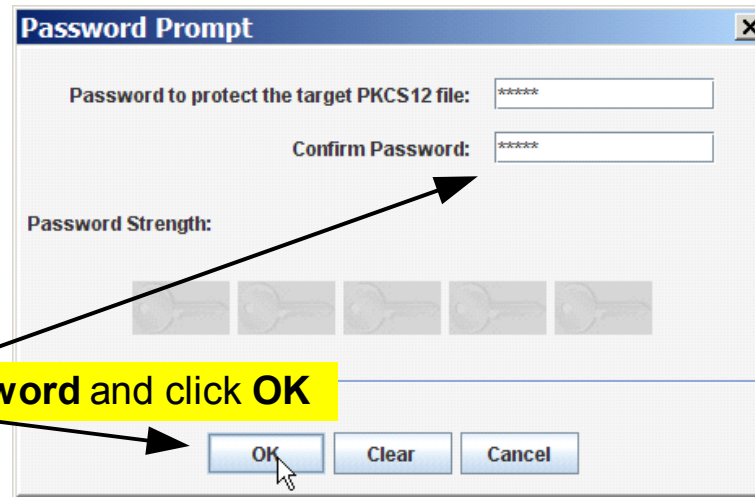
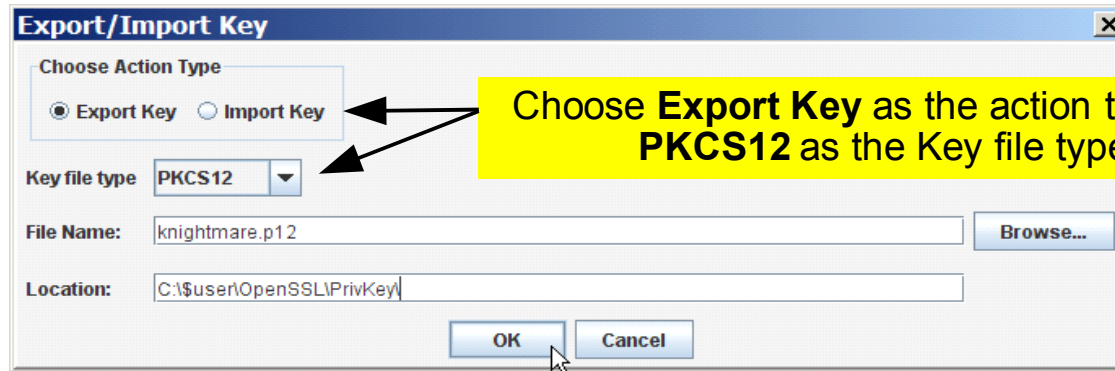
- WireShark requires the **private RSA key** in *PEM format* in order to decrypt SSL packets
 - ▶ The **Key Management Utility (ikeyman)** provided with the IBM HTTP Server can be used along with **OpenSSL** to obtain the private RSA key
 - **ikeyman** can be used to *export* the certificate into a *PKCS12* formatted file from the IBM HTTP Server's keyfile (e.g key.kdb)
 - **OpenSSL** can then be used to extract the private RSA key
 - In **wireshark**, the private RSA key can then be *added* and applied under *preferences* to decrypt the SSL packets



Decryption - How to *decrypt* SSL packets



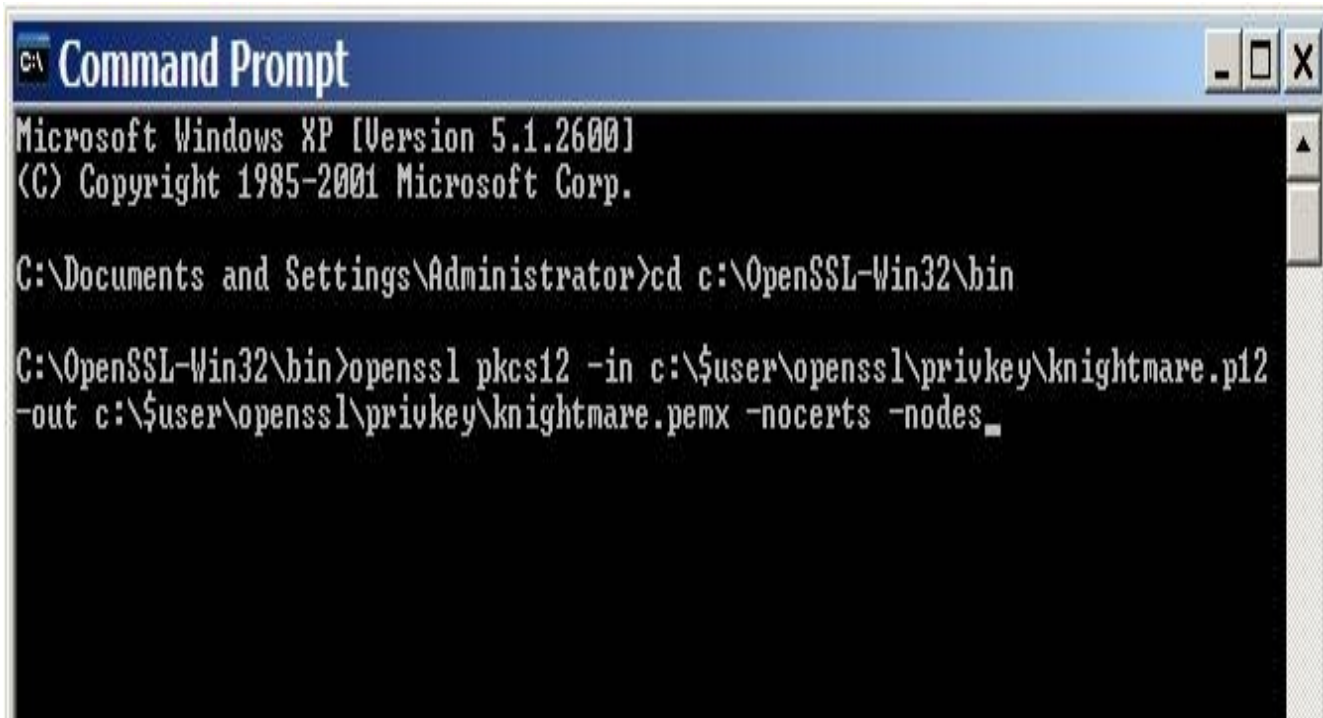
Decryption - How to *decrypt* SSL packets



Decryption - How to *decrypt* SSL packets

Obtain **OpenSSL** from www.openssl.org

Use the following openssl command to create a pemx file



```
Command Prompt
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\Administrator>cd c:\OpenSSL-Win32\bin

C:\OpenSSL-Win32\bin>openssl pkcs12 -in c:\$user\openssl\privkey\knightmare.p12
-out c:\$user\openssl\privkey\knightmare.pemx -nocerts -nodes
```


Decryption - How to *decrypt* SSL packets

```
Command Prompt - openssl pkcs12 -in c:\$user\openssl\privkey\kni...
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\Administrator>cd c:\OpenSSL-Win32\bin

C:\OpenSSL-Win32\bin>openssl pkcs12 -in c:\$user\openssl\privkey\knightmare.p12
-out c:\$user\openssl\privkey\knightmare.pemx -nocerts -nodes
Enter Import Password: _
```

Supply the password used to export the PKCS12 (.p12) formatted file and then hit enter and the **pemx** file will now be created

```
Command Prompt
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

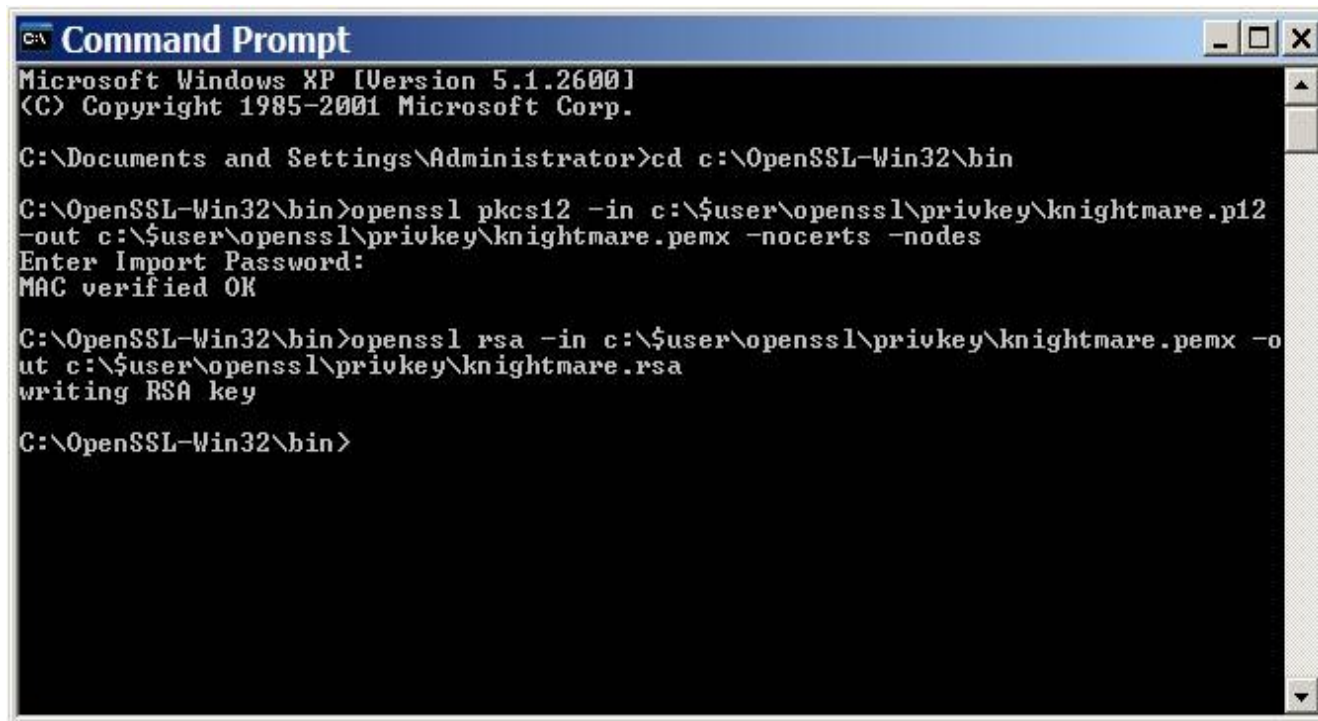
C:\Documents and Settings\Administrator>cd c:\OpenSSL-Win32\bin

C:\OpenSSL-Win32\bin>openssl pkcs12 -in c:\$user\openssl\privkey\knightmare.p12
-out c:\$user\openssl\privkey\knightmare.pemx -nocerts -nodes
Enter Import Password:
MAC verified OK

C:\OpenSSL-Win32\bin>_
```

Decryption - How to *decrypt* SSL packets

Use the following OpenSSL command to create the RSA file from pemx file



```
CA Command Prompt
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\Administrator>cd c:\OpenSSL-Win32\bin

C:\OpenSSL-Win32\bin>openssl pkcs12 -in c:\$user\openssl\privkey\knightmare.p12
-out c:\$user\openssl\privkey\knightmare.pem -nocerts -nodes
Enter Import Password:
MAC verified OK

C:\OpenSSL-Win32\bin>openssl rsa -in c:\$user\openssl\privkey\knightmare.pem -o
ut c:\$user\openssl\privkey\knightmare.rsa
writing RSA key

C:\OpenSSL-Win32\bin>
```

Decryption - How to *decrypt* SSL packets

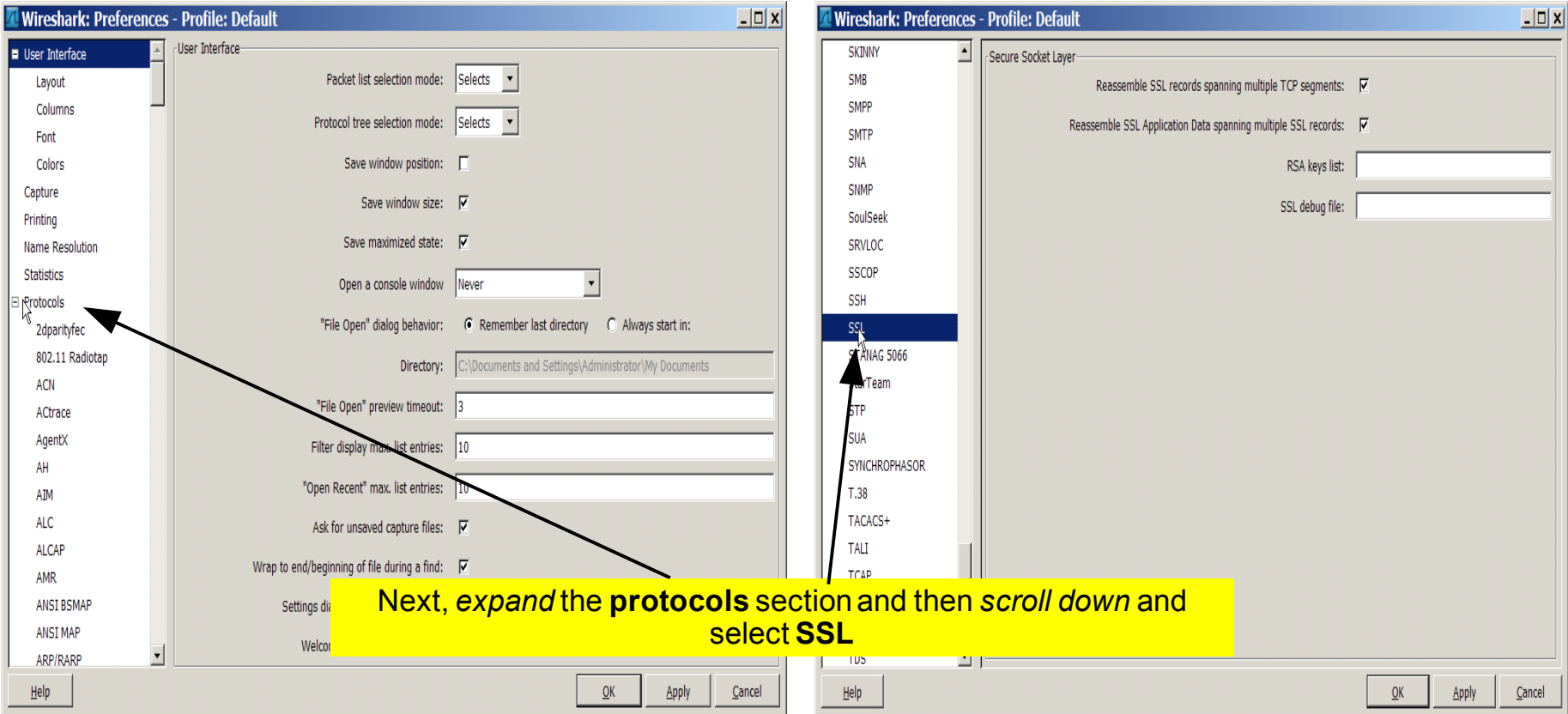
The image shows a Wireshark window titled 'knightmarv.pcap - Wireshark'. The main pane displays a list of network packets. Packet 519 is selected, showing details for the TLSv1 layer, including 'Encrypted Handshake Message'. A context menu is open over this packet, with 'Follow SSL Stream' highlighted. To the right, a 'Follow SSL Stream' window is open, but its 'Stream Content' area is empty. At the bottom of the Wireshark window, a yellow text box contains the text: **Before the SSL packets are decrypted the stream content window will be empty**.

Decryption - How to *decrypt* SSL packets

The screenshot shows the Wireshark interface with a list of captured packets. The selected packet is a TLSv1 record containing encrypted application data. A yellow callout box with a black arrow pointing to the 'Edit -> Preferences' menu item contains the following text:

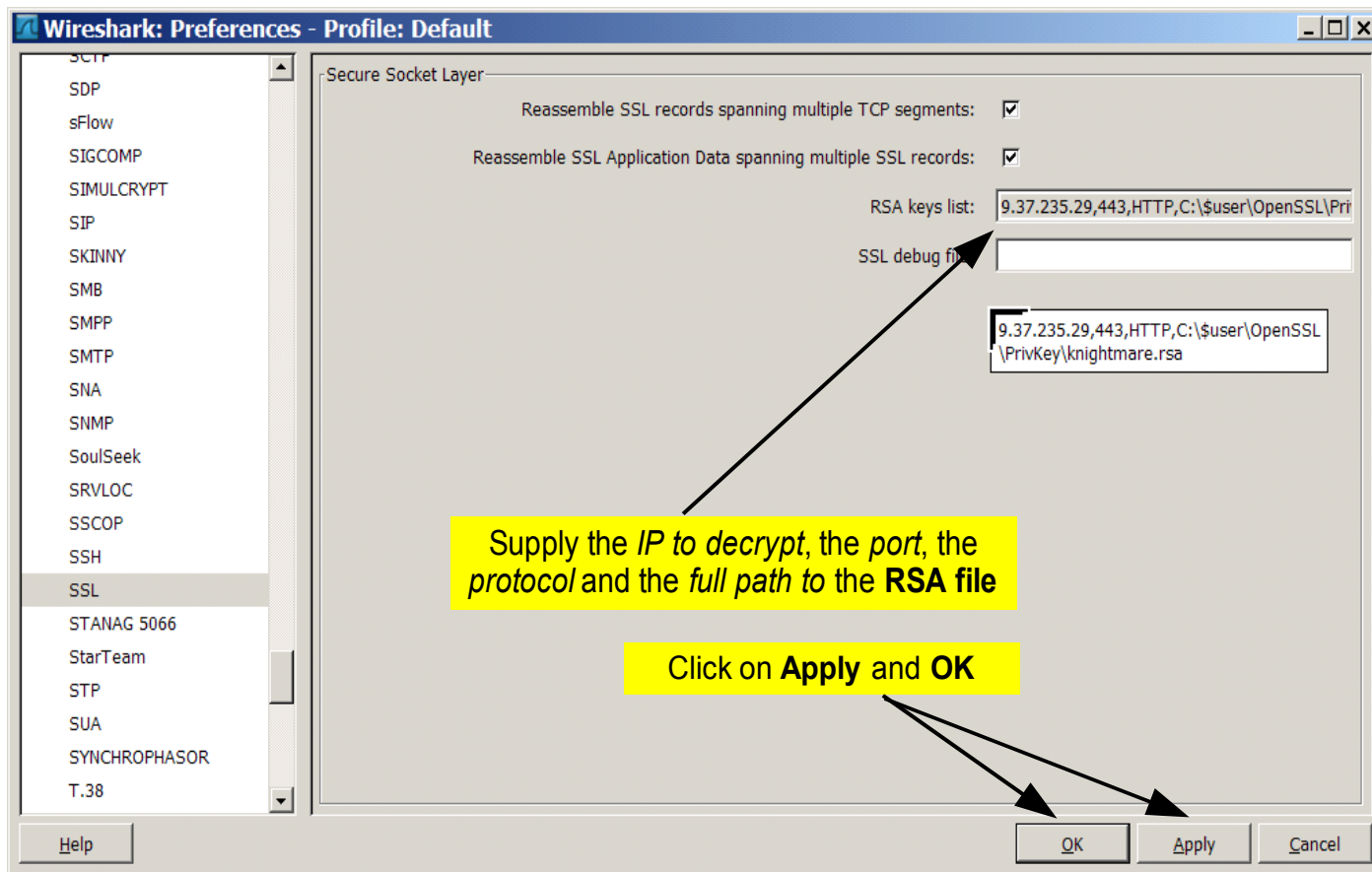
To add the RSA file containing the private key to *wireshark*..
Click on **Edit -> Preferences**

Decryption - How to *decrypt* SSL packets



Next, expand the **protocols** section and then *scroll down* and select **SSL**

Decryption - How to *decrypt* SSL packets



Decryption - How to *decrypt* SSL packets

The screenshot shows the Wireshark interface with a packet list on the left and a packet details pane on the right. The packet list shows a series of packets, with packet 555 selected. The details pane shows the decrypted SSL data (281 bytes) and the stream content window, which displays the decrypted HTTP response body. The stream content window shows the following HTML code:

```

GET / HTTP/1.1
Accept: image/gif, image/jpeg, image/pjpeg, image/pjpeg, application/x-shockwave-flash, application/vnd.ms-excel, application/vnd.ms-powerpoint, application/msword, application/x-ms-application, application/x-ms-xbap, application/vnd.ms-xpsdocument, application/xaml+xml, */*
Accept-Language: en-us
User-Agent: Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 5.1; Trident/4.0; .NET CLR 2.0.50727; .NET CLR 1.1.4322; InfoPath.2; .NET CLR 3.0.04506.648; .NET CLR 3.0.4506.2152; .NET CLR 3.5.30729; MS-RTC LM 8)
Accept-Encoding: gzip, deflate
Host: 9.37.235.29
Connection: Keep-Alive

HTTP/1.1 200 OK
Date: Wed, 01 Sep 2010 21:04:53 GMT
Server: IBM_HTTP_Server
Last-Modified: Fri, 13 Aug 2010 13:47:04 GMT
ETag: "55ab6-c6f-b7d0ecb"
Accept-Ranges: bytes
Content-Length: 3183
Keep-Alive: timeout=10, max=100
Connection: Keep-Alive
Content-Type: text/html

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4//EN">
<!-- (C) COPYRIGHT International Business Machines Corporation 1999 -->
<!-- All Rights Reserved -->
<!-- Licensed Materials - Property of IBM -->
<!-- -->
<!-- US Government Users Restricted Rights - Use, duplication or -->
<!-- disclosure restricted by GSA ADP Schedule Contract with IBM Corp.-->
<!-- -->
<html>
<HEAD>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<title>IBM HTTP Server</title>
<script language="JavaScript">

document.write('<link rel="stylesheet" href="http server styles.css">');
    
```

Below the screenshot, a yellow box contains the following text:

After the SSL packets are decrypted the stream content window will now show the *data unencrypted*

Summary

- Provided several **tips** on how to search and find information in a TCP packet capture using wireshark
- Demonstrated how to **save** a TCP packet capture into *smaller more manageable files*
- Stepped through the **debugging process** of *how to successfully correlate a TCP packet trace with other WebSphere related trace logs*
- Showed how to **export a certificate** from the IBM HTTP Server's keyfile using the *Key Management Utility*
- Walked through using *OpenSSL* to **extract the private RSA Key**
- Finally, demonstrated *how to utilize the RSA Key in wireshark* to **decrypt SSL packets**

Additional WebSphere Product Resources

- Learn about upcoming WebSphere Support Technical Exchange webcasts, and access previously recorded presentations at:
http://www.ibm.com/software/websphere/support/supp_tech.html
- Discover the latest trends in WebSphere Technology and implementation, participate in technically-focused briefings, webcasts and podcasts at:
<http://www.ibm.com/developerworks/websphere/community/>
- Join the Global WebSphere Community:
<http://www.websphereusergroup.org>
- Access key product show-me demos and tutorials by visiting IBM Education Assistant:
<http://www.ibm.com/software/info/education/assistant>
- View a webcast replay with step-by-step instructions for using the Service Request (SR) tool for submitting problems electronically:
<http://www.ibm.com/software/websphere/support/d2w.html>
- Sign up to receive weekly technical My Notifications emails:
<http://www.ibm.com/software/support/einfo.html>

We Want to Hear From You!

Tell us about what you want to learn

Suggestions for future topics
Improvements and comments about our webcasts
We want to hear everything you have to say!

Please send your suggestions and comments to:
wsehelp@us.ibm.com



Questions and Answers

