SHARKFEST '13

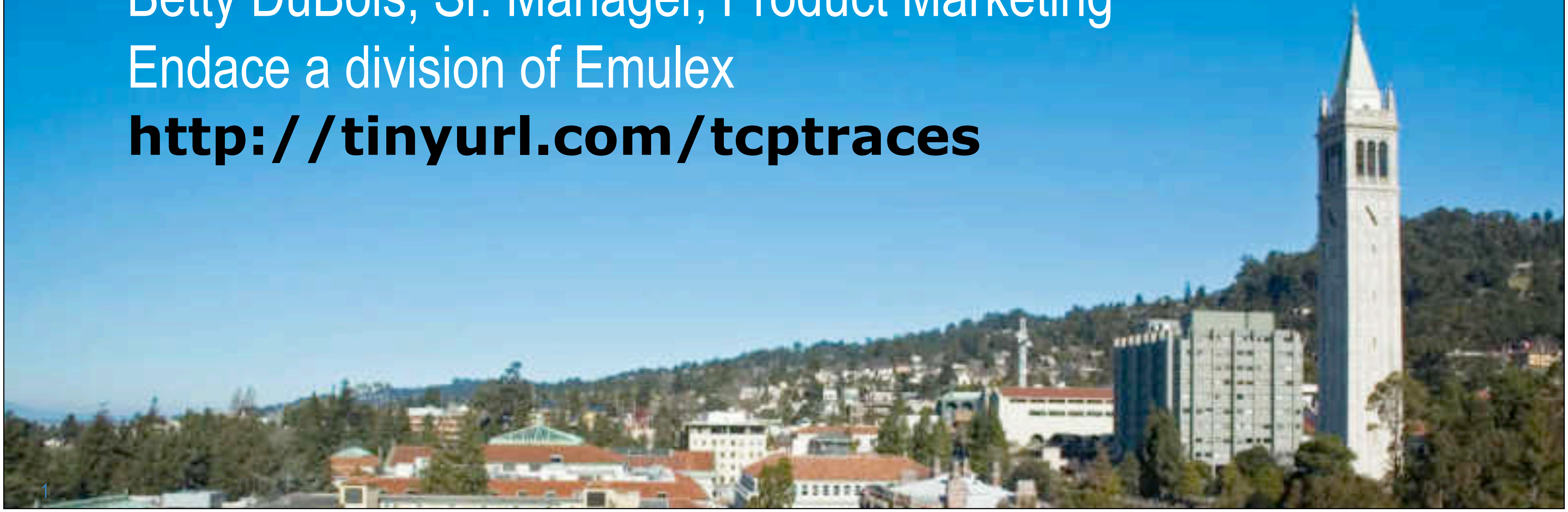
Wireshark Developer and User Conference

Inside the TCP Handshake

Betty DuBois, Sr. Manager, Product Marketing

Endace a division of Emulex

http://tinyurl.com/tcptraces



Agenda

- Goals of the TCP handshake
- Beginning sequence numbers
- Options

Let’s Go Live!

- Start a Wireshark capture
- Using your favorite FTP client:
  - ftp://ftp.FreeBSD.org/pub/FreeBSD/
  - User: anonymous
  - Password: whatever
- Click on any of the documents, let it load and then stop your capture.
- Right click on any ftp packet, and “follow the TCP stream”
- Or use “Owen – Windows7client.pcapng” as example

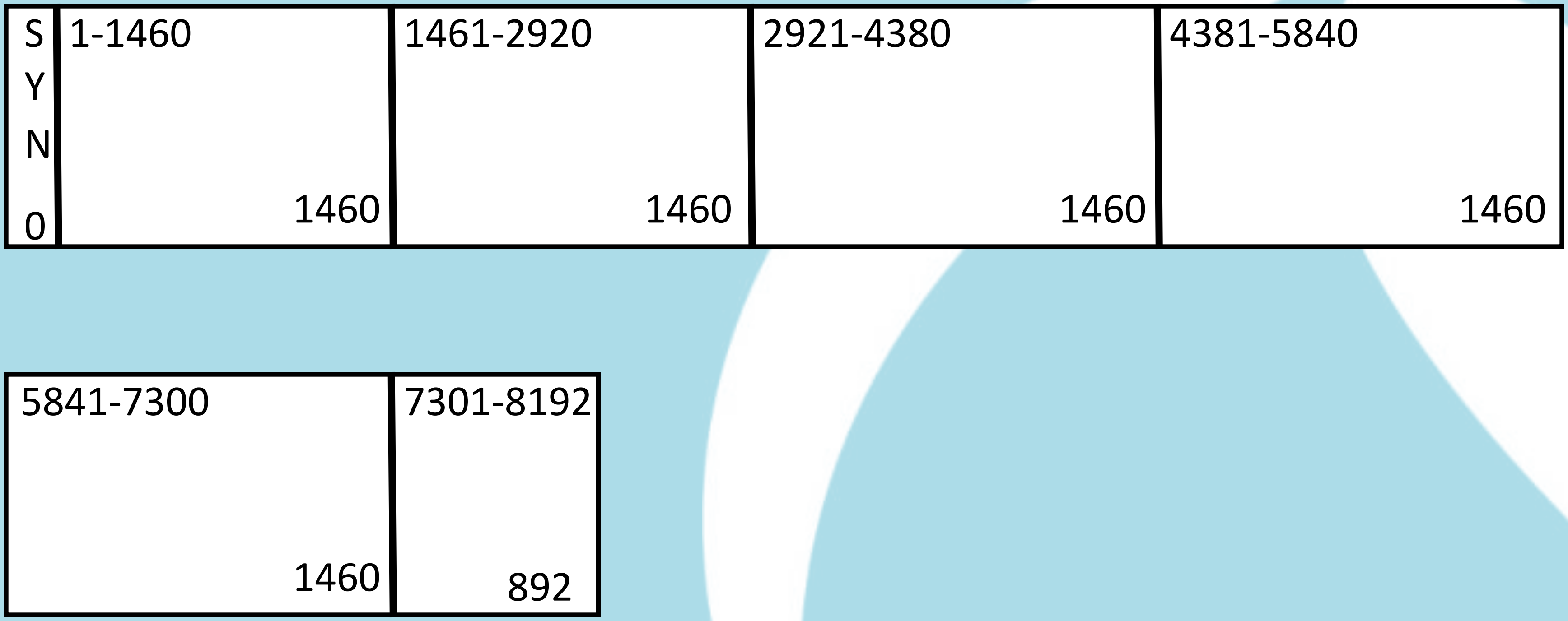
Goals of the Handshake

- Is destination port open?
- Notification of opened ephemeral port
- Notification of each sides beginning sequence #
- Notification of each sides receive window size
- Option negotiation

Beginning Sequence #'s

- Each side will give their starting sequence number
- They will be different on each side
- The TCP stack uses them for byte count
- Wireshark will show relative numbers so it looks as if both sides start at zero.
  - The numbers are relative to the source IP and source port (i.e. socket)
  - The beauty is using them to see how deep you are into the data transfer at any given point

Sequence Numbers



SYN 0

1-1460 1460

1461-2920 1460

2921-4380 1460

4381-5840 1460

5841-7300 1460

7301-8192 892

Open Negotiation

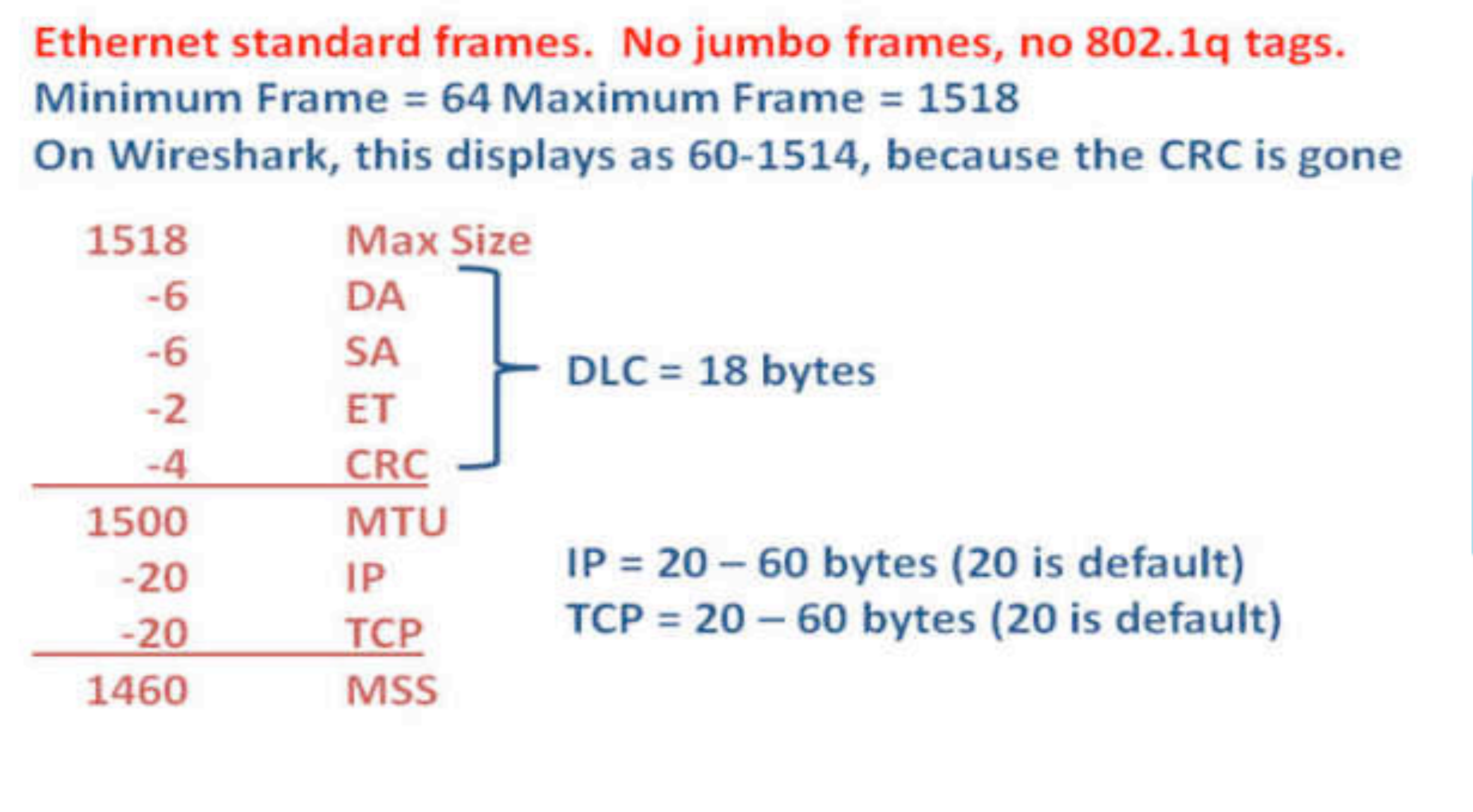
- Silence means NO
- MSS
- Window Scaling
- SACK
- Time Stamp

Silence means NO

- There is not a negative ACK/NACK
- So if a host does not support an option:
  - There is no request from the client
  - Or
  - There is no mention of the option in the server's response

Maximum Segment Size

- How much TCP Data can fit in a single packet?
- Implementation is that lowest number wins



Ethernet standard frames. No jumbo frames, no 802.1q tags.

Minimum Frame = 64 Maximum Frame = 1518

On Wireshark, this displays as 60-1514, because the CRC is gone

1518	Max Size
-6	DA
-6	SA
-2	ET
-4	CRC
1500	MTU
-20	IP
-20	TCP
1460	MSS

DLC = 18 bytes

IP = 20 – 60 bytes (20 is default)

TCP = 20 – 60 bytes (20 is default)

Window Scaling

- Both sides must support, but do not have to agree on amount
- Simply a way to take advantage of bigger buffers

Selective Ack - SACK

- Both sides must support
- ACK field is always cumulative data
- SACK field is for the data after last segments
- Room for 3 SACK sections in the options section
- Once data is sacked it can be flushed from the sender’s TCP window

Timestamp

- Both sides must support
- Goals:
  - More granular Round Trip Time (RTT) measurements
  - Tie-breaker when sequence number wraps aka Protect Against Wrapped Sequence (PAWS)
- RFCs
  - 1323
  - 3522
- Use “Betty_LionClient.pcapng” for example

Questions?