SharkFest ‘16

Tackling the Haystack

Tuesday, June 14, 2016

Jasper Bongertz

Expert Analyst | Airbus Defence and Space CyberSecurity

Agenda

1. Haystack?
2. Capture
3. Methodology
4. Tools
5. Demos/Scenarios

What‘s your Haystack size?

What‘s your haystack size?

- This?

What‘s your haystack size?

- This?

What‘s your haystack size?

- This?

Haystack size

- Everybody has a different "haystack size"
  - new analysts may find 20 packets too hard to understand
  - experienced analysts can deal with gigabytes of traffic if they have to
- Capture files
  - dealing with a single file vs. dealing with file sets

Example Sets

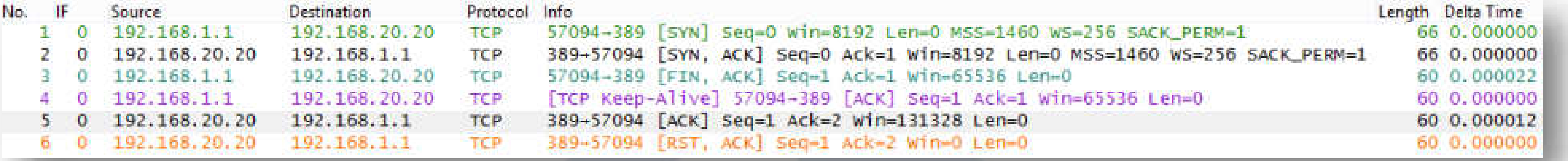
- October 2015: ~300GB
  - Trouble with latency of CAD designing in Citrix sessions
- November 2015: ~500GB
  - "see if you can find anything that we can improve/fix"
- February 2016: ~600GB (sliced to 256 bytes)
  - Web application trouble with long proxy chain
- May 2016: ~4000GB
  - Checking for Indicators of Compromise

Working with the haystack

Reducing the haystack size

- Knowledge is a basic building block:
  - protocol behavior, especially IPv4/6 and TCP/UDP
  - application behavior
  - user behavior
  - typical network & security devices, e.g firewalls, packet shapers etc.
- Experience is key
  - spot the important stuff faster
  - know what you can safely ignore & not waste time on
  - Problem: experience is usually gained **after** you needed it most

Experience vs. Knowledge



No.	IF	Source	Destination	Protocol	Info	Length	Delta Time
1	0	192.168.1.1	192.168.20.20	TCP	57094→389 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1	66	0.000000
2	0	192.168.20.20	192.168.1.1	TCP	389→57094 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1	66	0.000000
3	0	192.168.1.1	192.168.20.20	TCP	57094→389 [FIN, ACK] Seq=1 Ack=1 Win=65536 Len=0	60	0.000022
4	0	192.168.1.1	192.168.20.20	TCP	[TCP Keep-Alive] 57094→389 [ACK] Seq=1 Ack=1 Win=65536 Len=0	60	0.000000
5	0	192.168.20.20	192.168.1.1	TCP	389→57094 [ACK] Seq=1 Ack=2 Win=131328 Len=0	60	0.000012
6	0	192.168.20.20	192.168.1.1	TCP	389→57094 [RST, ACK] Seq=1 Ack=2 Win=0 Len=0	60	0.000000

- **Knowledge** allows you to understand the meaning of the TCP packets
- **Experience** tells you if this conversations is worth mentioning in a analysis report

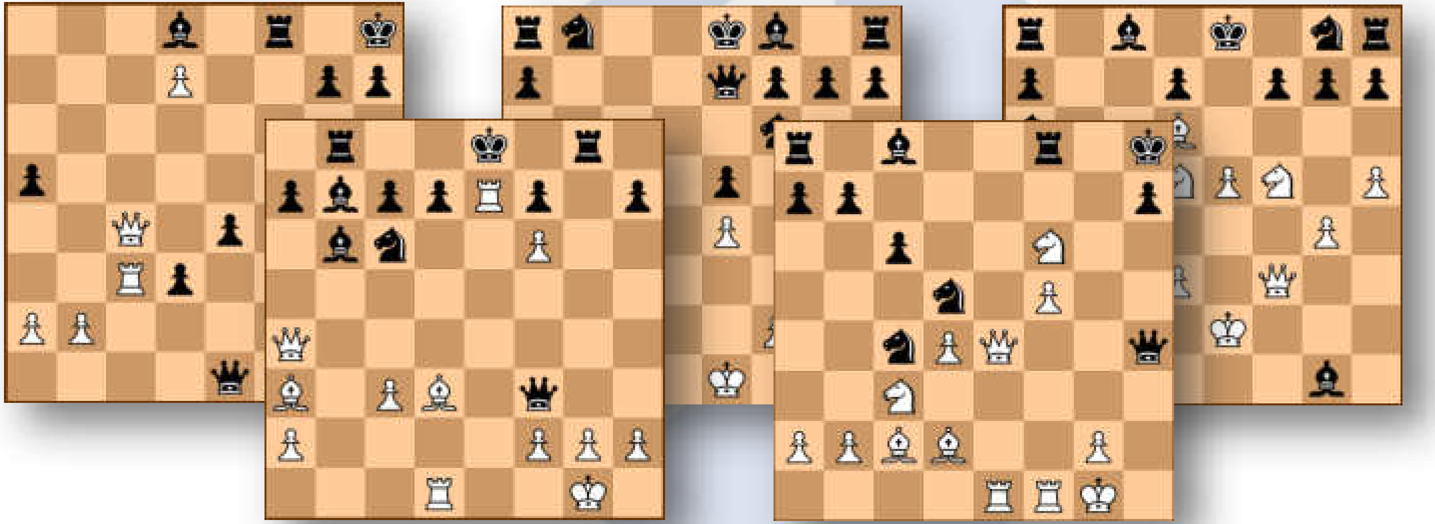
The path to experience

- When no/little experience is available, you can still reduce the haystack using knowledge
  - read documentation on protocols, applications, etc.
  - gather information about IPs/Users/Ports involved
  - get detailed problem descriptions, with exact date/time info
- Basically you'll need to "learn on the fly"
- Double check your findings whenever you're not sure
  - if possible, ask experienced analysts for a review

General Best Practises

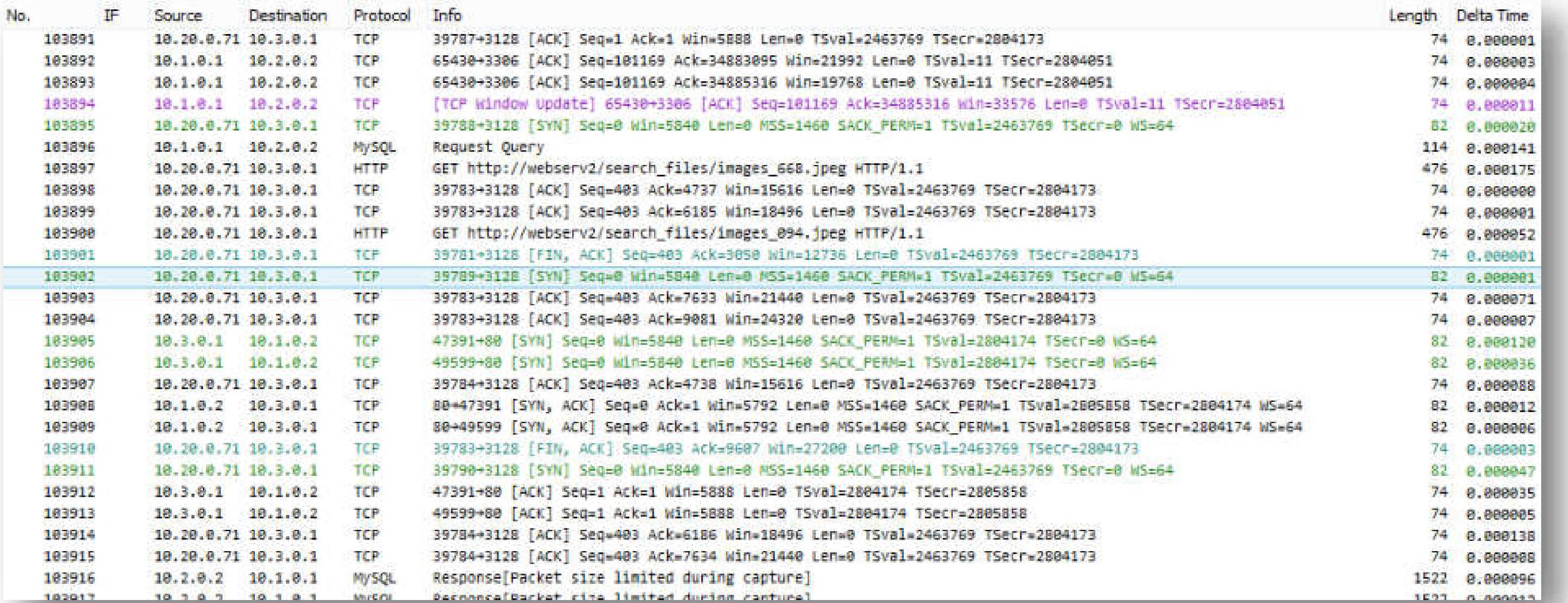
TCP Sessions vs. Chess Games

- How many chess games can you watch/play simultaneously?



Same problem with TCP Sessions

- Can you keep track of more than one?



No.	IF	Source	Destination	Protocol	Info	Length	Delta Time
103891		10.20.0.71	10.3.0.1	TCP	39787→3128 [ACK] Seq=1 Ack=1 Win=5888 Len=0 TSval=2463769 TSecr=2804173	74	0.000001
103892		10.1.0.1	10.2.0.2	TCP	65430→3306 [ACK] Seq=101169 Ack=34883095 Win=21992 Len=0 TSval=11 TSecr=2804051	74	0.000003
103893		10.1.0.1	10.2.0.2	TCP	65430→3306 [ACK] Seq=101169 Ack=34885316 Win=19768 Len=0 TSval=11 TSecr=2804051	74	0.000004
103894		10.1.0.1	10.2.0.2	TCP	[TCP Window Update] 65430→3306 [ACK] Seq=101169 Ack=34885316 Win=33576 Len=0 TSval=11 TSecr=2804051	74	0.000011
103895		10.20.0.71	10.3.0.1	TCP	39788→3128 [SYN] Seq=0 Win=5840 Len=0 MSS=1460 SACK_PERM=1 TSval=2463769 TSecr=0 WS=64	82	0.000020
103896		10.1.0.1	10.2.0.2	MySQL	Request Query	114	0.000141
103897		10.20.0.71	10.3.0.1	HTTP	GET http://webserv2/search_files/images_668.jpeg HTTP/1.1	476	0.000175
103898		10.20.0.71	10.3.0.1	TCP	39783→3128 [ACK] Seq=403 Ack=4737 Win=15616 Len=0 TSval=2463769 TSecr=2804173	74	0.000000
103899		10.20.0.71	10.3.0.1	TCP	39783→3128 [ACK] Seq=403 Ack=6185 Win=18496 Len=0 TSval=2463769 TSecr=2804173	74	0.000001
103900		10.20.0.71	10.3.0.1	HTTP	GET http://webserv2/search_files/images_094.jpeg HTTP/1.1	476	0.000052
103901		10.20.0.71	10.3.0.1	TCP	39781→3128 [FIN, ACK] Seq=403 Ack=3050 Win=12736 Len=0 TSval=2463769 TSecr=2804173	74	0.000001
103902		10.20.0.71	10.3.0.1	TCP	39789→3128 [SYN] Seq=0 Win=5840 Len=0 MSS=1460 SACK_PERM=1 TSval=2463769 TSecr=0 WS=64	82	0.000001
103903		10.20.0.71	10.3.0.1	TCP	39783→3128 [ACK] Seq=403 Ack=7633 Win=21440 Len=0 TSval=2463769 TSecr=2804173	74	0.000071
103904		10.20.0.71	10.3.0.1	TCP	39783→3128 [ACK] Seq=403 Ack=9081 Win=24320 Len=0 TSval=2463769 TSecr=2804173	74	0.000007
103905		10.3.0.1	10.1.0.2	TCP	47391→80 [SYN] Seq=0 Win=5840 Len=0 MSS=1460 SACK_PERM=1 TSval=2804174 TSecr=0 WS=64	82	0.000120
103906		10.3.0.1	10.1.0.2	TCP	49599→80 [SYN] Seq=0 Win=5840 Len=0 MSS=1460 SACK_PERM=1 TSval=2804174 TSecr=0 WS=64	82	0.000036
103907		10.20.0.71	10.3.0.1	TCP	39784→3128 [ACK] Seq=403 Ack=4738 Win=15616 Len=0 TSval=2463769 TSecr=2804173	74	0.000088
103908		10.1.0.2	10.3.0.1	TCP	80→47391 [SYN, ACK] Seq=0 Ack=1 Win=5792 Len=0 MSS=1460 SACK_PERM=1 TSval=2805858 TSecr=2804174 WS=64	82	0.000012
103909		10.1.0.2	10.3.0.1	TCP	80→49599 [SYN, ACK] Seq=0 Ack=1 Win=5792 Len=0 MSS=1460 SACK_PERM=1 TSval=2805858 TSecr=2804174 WS=64	82	0.000006
103910		10.20.0.71	10.3.0.1	TCP	39783→3128 [FIN, ACK] Seq=403 Ack=9607 Win=27200 Len=0 TSval=2463769 TSecr=2804173	74	0.000003
103911		10.20.0.71	10.3.0.1	TCP	39790→3128 [SYN] Seq=0 Win=5840 Len=0 MSS=1460 SACK_PERM=1 TSval=2463769 TSecr=0 WS=64	82	0.000047
103912		10.3.0.1	10.1.0.2	TCP	47391→80 [ACK] Seq=1 Ack=1 Win=5888 Len=0 TSval=2804174 TSecr=2805858	74	0.000035
103913		10.3.0.1	10.1.0.2	TCP	49599→80 [ACK] Seq=1 Ack=1 Win=5888 Len=0 TSval=2804174 TSecr=2805858	74	0.000005
103914		10.20.0.71	10.3.0.1	TCP	39784→3128 [ACK] Seq=403 Ack=6186 Win=18496 Len=0 TSval=2463769 TSecr=2804173	74	0.000138
103915		10.20.0.71	10.3.0.1	TCP	39784→3128 [ACK] Seq=403 Ack=7634 Win=21440 Len=0 TSval=2463769 TSecr=2804173	74	0.000088
103916		10.2.0.2	10.1.0.1	MySQL	Response[Packet size limited during capture]	1522	0.000096
103917		10.2.0.2	10.1.0.1	MySQL	Response[Packet size limited during capture]	1522	[illegible]

Discipline is key

- **Never** delete original capture files
  - you must always be able to check findings in filtered/carved results against the "true" capture
- Document **everything!**
  - this is actually very hard to do consistently (because Lazyness)
- Try to keep filter chains intact
  - it should be possible to retrace the steps from the original down to the final filtered result

Teamwork

- If possible, add as many analysts to the task as you can afford
  - biggest team I had was 3 experts analysts working on complex projects
- Not all analysts are equal
  - Basic skills are the same (e.g. TCP), but everybody specializes
  - WiFi, VoIP, Virtualization, SMB/CIFS are special topics
- Challenges are an important instrument
  - "Found root cause!" (me) - "What? Can't be!" (Chris)

Focus! Focus! Focus!

- It's easy to get lost in all the packets
  - Interesting/weird/unusual stuff found everywhere
- For really big tasks, a team leader is required
  - assigns tasks to members
  - keeps track of time spent
  - calls/leads status update meetings
- Add some "candyland time" if you can
  - e.g. "everybody has until lunch to do whatever he wants with the packets"

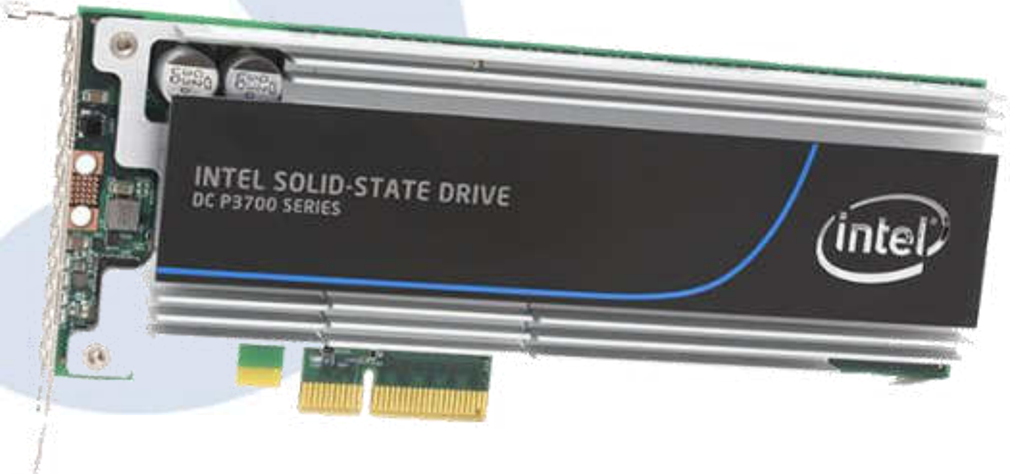
Mission Parameters

- What are you supposed to do?
- What information do you have to limit the scope?
  - IP addresses
  - Protocol ports
  - User names
  - Date/Time
  - Markers

Capture Setup

Capture Setup

- Obvious things to consider:
  - time stamp accuracy
  - lost packet ratio
- Not that obvious, but important for large captures:
  - enough free storage?
  - fast enough, too?
  - which file format?



File access

- How can the captured data be accessed?
  - during capture?
  - after capture?
- Multiple strategies:
  - via USB1/2/3 port (ouch, meh, yay)
  - via Gigabit (or faster) NIC
  - pulling HDD from capture device

Capture file parameters

- Single file or file set?
- Single file
  - shows all the details in Wireshark at once
  - may be to big to load though
  - can be cut into file sets using **editcap –c**
- File Set
  - Size range from 64MB to 512MB are common
  - conversations may span multipe files

Slicing

- Advantages
  - available disk space (well, not really, but it doesn't write as much)
  - can help avoiding dropped frames
  - privacy concerns can be dealt with (bluntly)
- Disadvantages
  - you're not storing everything on the wire to disk
  - if you realize you needed more bytes of a frame you have a problem
  - Reassembly/content reconstruction is not possible

Analysis setup

Analysis setup

- Number of analysts
  - if more than one, new challenges appear, e.g. how to share captures
- Number of workstations
  - more is better, helping with carve jobs
- Number of harddrives
  - reading from one, writing to another beats working on a single disk
  - SSDs prefered, but usually smaller than traditional HDDs
- Number of monitors

Typical analysis tasks

- Carve/Extraction Jobs
  - automated packet extraction from large files / set of files
  - often run for hours/days, depending on files/tools
- Filtering
  - manual filtering in Wireshark or other tools
  - only feasable for single files & small numbers of packets
- Merging
  - merge carve/extraction results

A few useful tools (1/2)

- Filtering/carving files
  - Wireshark
  - tshark
  - tcpdump/windump
  - TraceWrangler
- Convert/edit files
  - editcap
  - reordercap
  - TraceWrangler

A few useful tools (2/2)

- Merging files
  - Wireshark
  - mergecap
  - TraceWrangler
- Others
  - pcaptouch
  - ngconvert
  - Network Miner
  - tcpflow

Demo 1 - Carving "Essentials"

Hints for "Essentials" carving

- "Essentials" may vary based on the task at hand
  - usually always involves TCP handshake/teardown, so filter for "tcp.flags.syn==1 or tcp.flags.fin==1 or tcp.flags.reset==1"
  - DNS and ICMP are safe bets, too
- Distribute carve tasks across workstations if necessary/possible
  - requires distributing traces and planning carve jobs first

Demo 2 - 5 Tuple VLAN Carve

Hints for VLAN carving

- Running tshark once per VLAN may take a long time
  - each time tshark has to read all the original files
- Methods to improve performance:
  - disable irrelevant dissectors (double check!)
  - Divide & Conquer
  - e.g. carve VLANs 10,11,12 in one run, 13, 14, 15 in another, then run again on partial files for 10, then 11, then 12, etc.
  - use tcpdump/windump with BPF

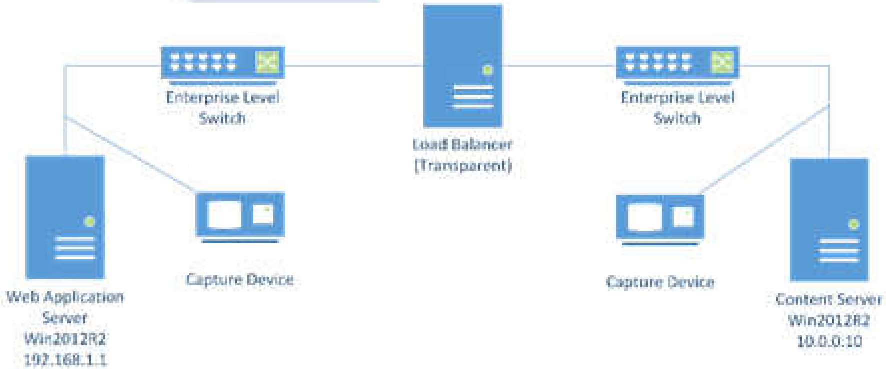
Demo 3 - Extracting Frames

Hints for extracting frames

- Adding filters for tons of frames in TraceWrangler is going to be slow
  - that's because the code isn't optimized at all
  - it's on the ToDo list ☺
- The output settings define to what file frames will be written

Demo 4: Conversation Statistics

Demo 5 - Megalodon


Enterprise Level Switch
Load Balancer (Transparent)
Enterprise Level Switch
Web Application Server Win2012R2 192.168.1.1
Capture Device
Capture Device
Content Server Win2012R2 10.0.0.10

Q&A

Mail:	jasper@packet-foo.com
Web:	blog.packet-foo.com
Twitter:	@packetjay