Stratoshark

... a new kid on the block!

SYN-bit
deep traffic analysis

Sake Blok
Relational therapist for computer systems
sake.blok@SYN-bit.nl

$ whoami

EURONET INTERNET

ABN·AMRO

ion ip

SYN-bit
deep traffic analysis

SYN-bit

deep traffic analysis

Application and network troubleshooting

Protocol and packet analysis

Training (Wireshark, TCP, SSL)

www.SYN-bit.nl

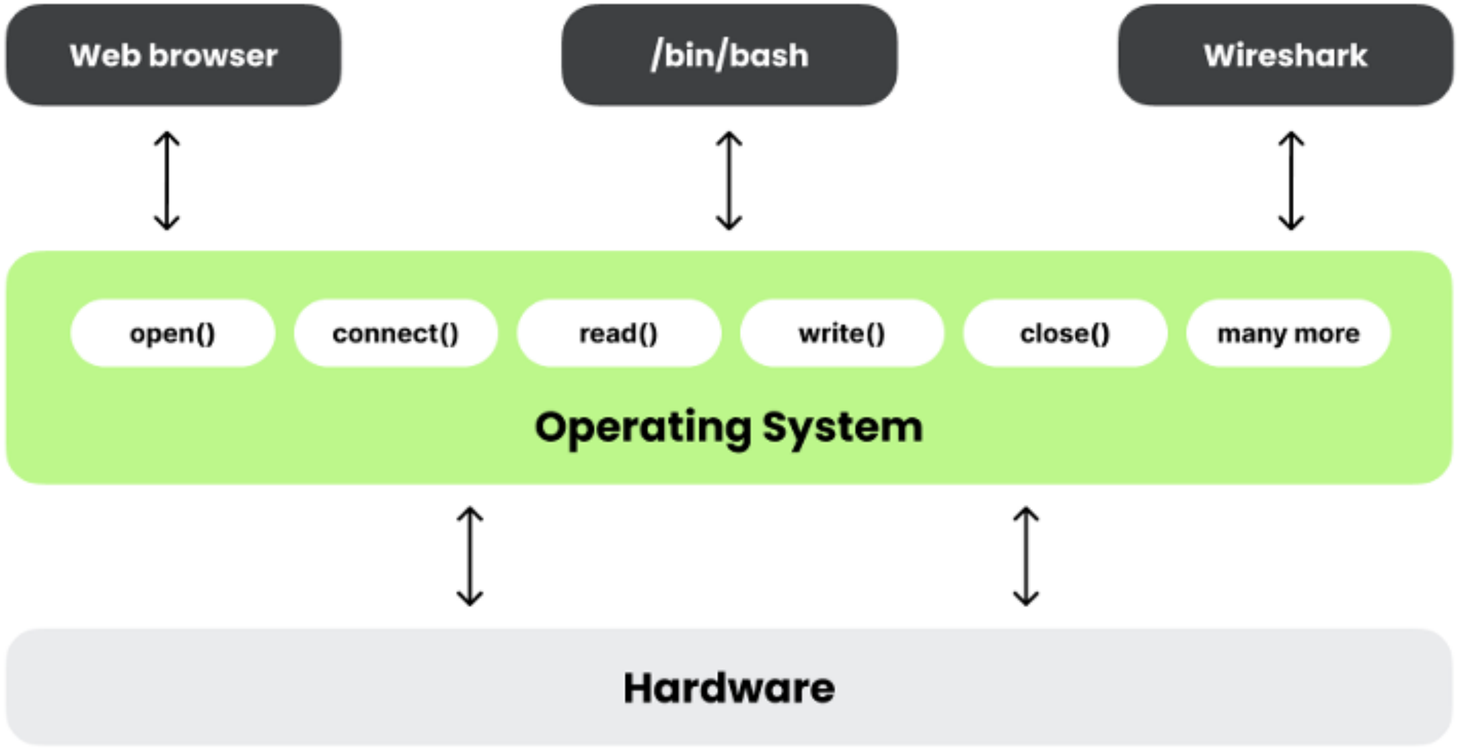
A little history of packet capturing

- 1988: tcpdump
  - Van Jacobsen, Sally Floyd, Vern Paxton, Steve McCanne
  - Lawrence Berkely Laboratory
- 1994: libpcap becomes separate library
- 1998: Ethereal
- 1999: WinPcap
  - Development stopped in 2018
- 2006: Wireshark
- 2013: Npcap
  - Replaces WinPcap in Wireshark Installer

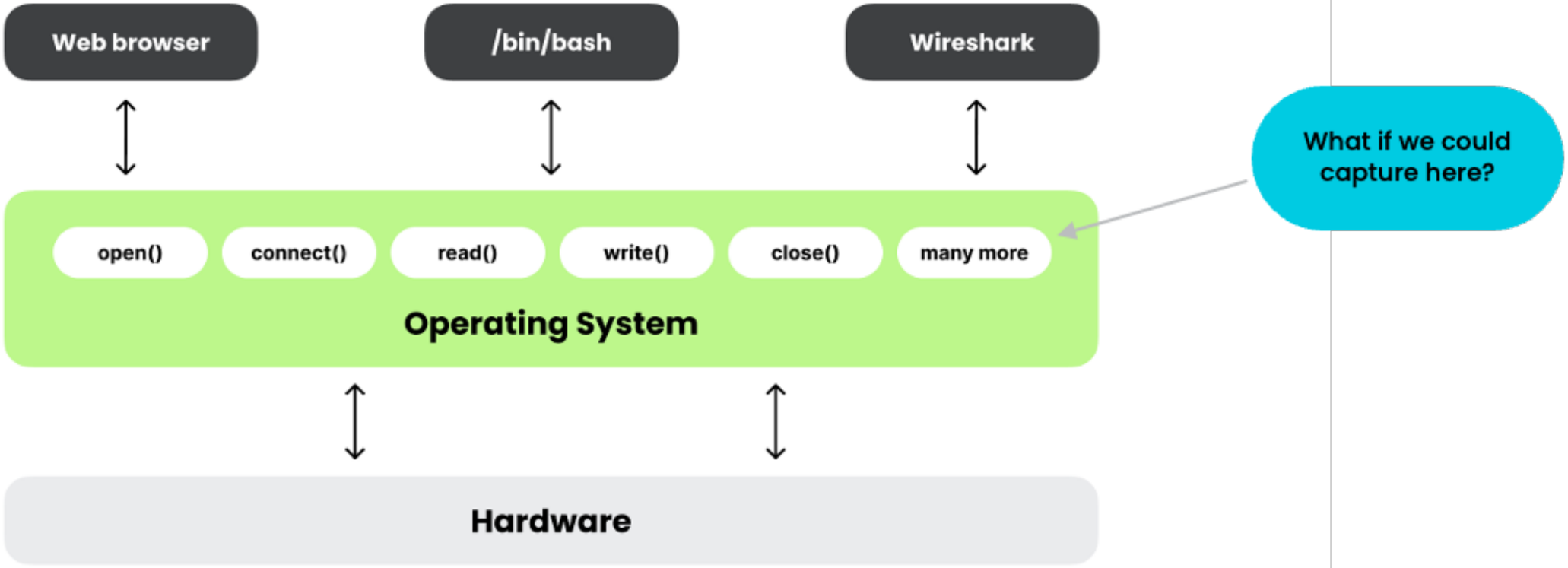
Challenges with packets?

- Access to packets more and more challenging, especially in cloud environments
- Encryption is more widespread and decryption not always possible
- Containers and kubernetes make workloads more volatile
- What if we could dig into system calls instead?

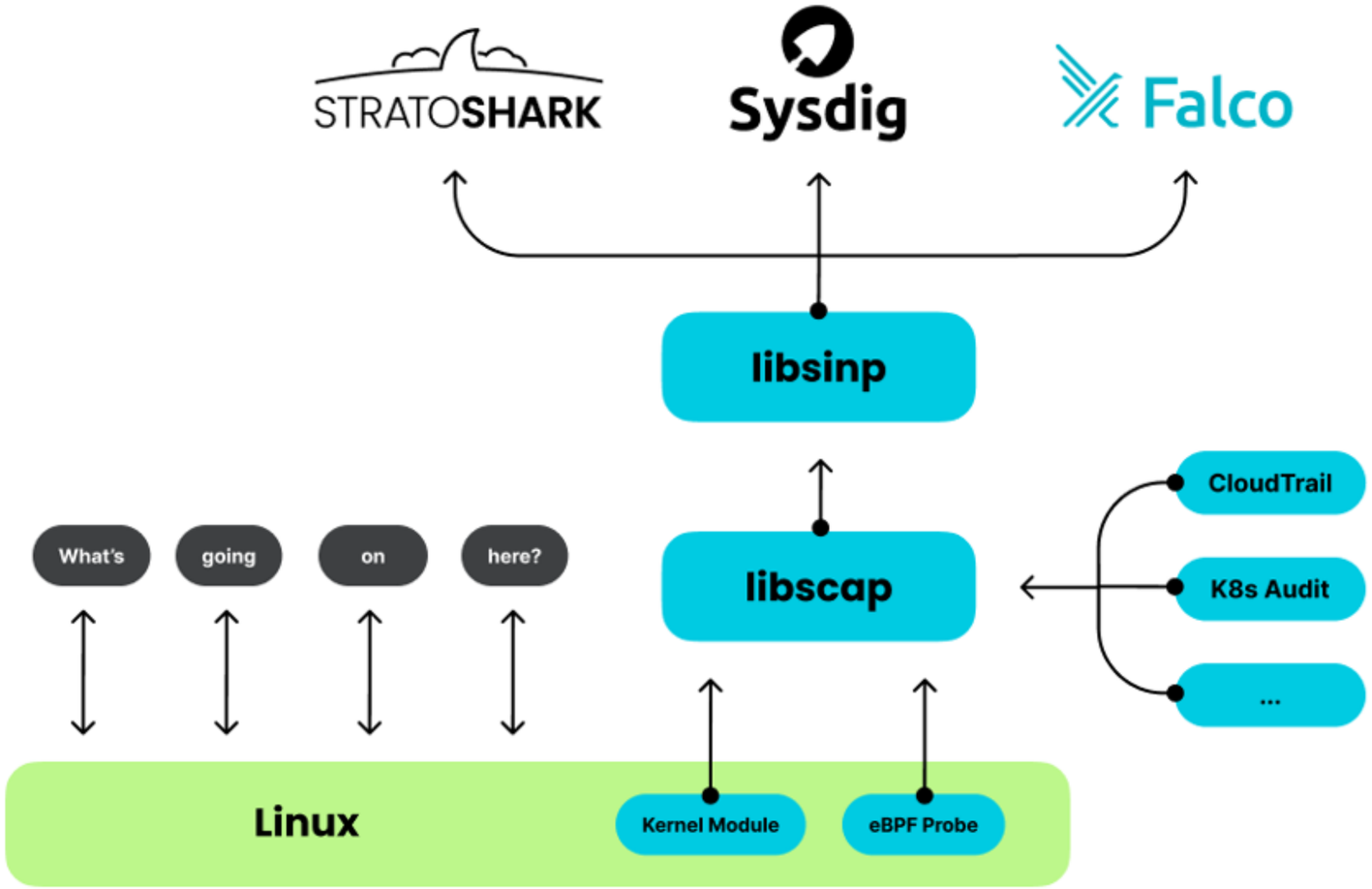
What are system calls?



Capturing system calls?

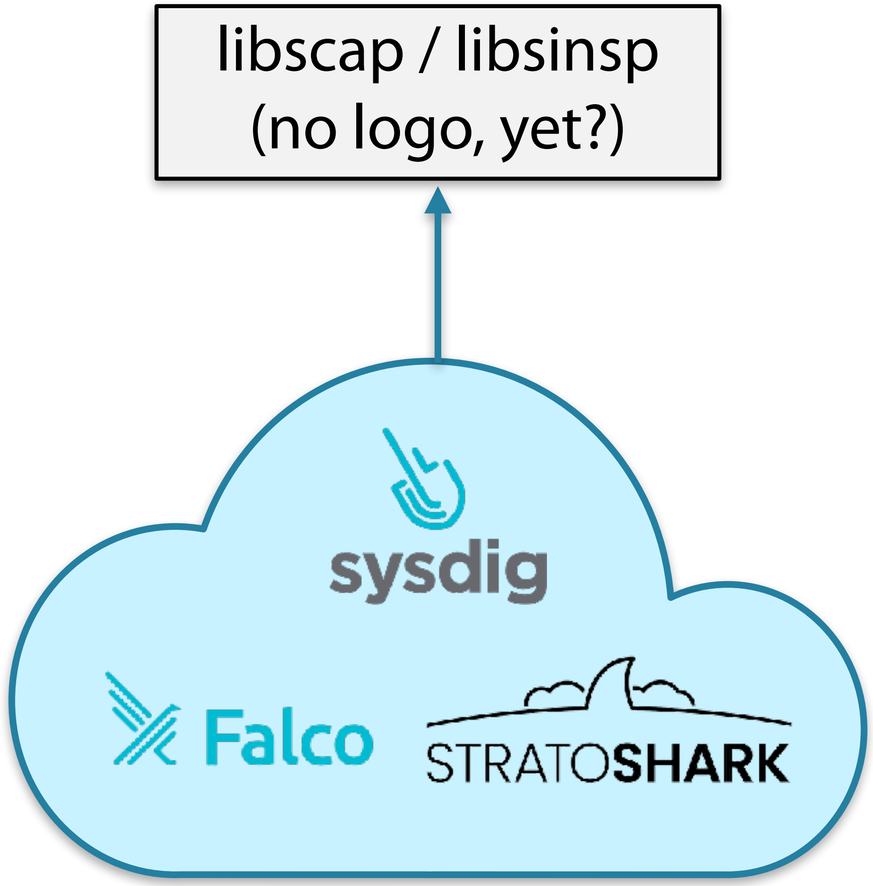


libscap/libsinsp usage very similar to using libpcap



A little history of syscall capturing

- 2014: sysdig
  - Loris Degioanni (who also started WinPcap)
- 2016: Falco
  - 2018: Falco handed over to CNCF, graduates in 2024
  - https://github.com/falcosecurity/falco
- 2021: libscap/libsinsp handed over to the CNCF
  - https://github.com/falcosecurity/libs
- *2021: Add plugin infrastructure to falco libs*
  - *Makes ingesting cloud logging possible (like AWS cloudtrail)*
- 2025: Stratoshark
  - Gerald Combs (who also started Wireshark)
  - https://gitlab.com/wireshark/wireshark/-/tree/master/ui/stratoshark



Similarities...

network packet domain

cli capture/viewer

gui capture/viewer

intrusion detection

shared libraries

system call domain

cli capture/viewer

gui capture/viewer

intrusion detection

libscap / libsinsp

shared libraries

sysdig

```
beheer@docker-macbook:~$ timeout 1 sudo sysdig
18 17:26:38.319058248 1 sysdig (175349.175349) > switch next=0 pgft_maj=0 pgft_min=1043 vm_size=282380 vm_rss=15176 vm_swap=0
19 17:26:38.319130574 1 <NA> (<NA>.0) > switch next=175349(sysdig) pgft_maj=0 pgft_min=0 vm_size=0 vm_rss=0 vm_swap=0
40 17:26:38.319211982 1 sysdig (175349.175349) > switch next=0 pgft_maj=0 pgft_min=1047 vm_size=282380 vm_rss=15176 vm_swap=0
41 17:26:38.319214149 0 <NA> (<NA>.0) > switch next=174903 pgft_maj=0 pgft_min=0 vm_size=0 vm_rss=0 vm_swap=0
42 17:26:38.319221690 0 <NA> (<NA>.174903) > switch next=0 pgft_maj=0 pgft_min=0 vm_size=0 vm_rss=0 vm_swap=0
43 17:26:38.319233397 1 <NA> (<NA>.0) > switch next=175347(sudo) pgft_maj=0 pgft_min=0 vm_size=0 vm_rss=0 vm_swap=0
44 17:26:38.319239230 1 sudo (175347.175347) < ppoll res=1 fds=11:u0 3:p0 9:f1 8:f0
45 17:26:38.319247187 1 sudo (175347.175347) > rt_sigaction
46 17:26:38.319247520 1 sudo (175347.175347) < rt_sigaction
47 17:26:38.319248104 1 sudo (175347.175347) > read fd=9(<f>/dev/ptmx) size=65536
48 17:26:38.319252937 1 sudo (175347.175347) < read res=324 data=18 17:26:38.319058248 1 .[01;32msysdig.[00m (.[01;36m175349.[00m.175349) > .[01; fd=9(<f>/dev/ptmx) size=65536
[...]
259719 17:26:38.500757839 0 sshd (145515.145515) > rt_sigprocmask
259720 17:26:38.500757922 0 sshd (145515.145515) < rt_sigprocmask
259722 17:26:38.500758256 0 sshd (145515.145515) > read fd=10(<f>/dev/ptmx) size=32768
259723 17:26:38.500759172 0 sshd (145515.145515) < read res=2048 data=68681 17:26:38.365054879 0 .[01;32msudo.[00m (.[01;36m175347.[00m.175347) < .[01 fd=10(<f>/dev/ptmx) size=32768
259725 17:26:38.500760922 0 sshd (145515.145515) > switch next=174904 pgft_maj=7 pgft_min=1414 vm_size=20160 vm_rss=6352 vm_swap=408
259729 17:26:38.500763380 0 <NA> (<NA>.174904) > switch next=145515(sshd) pgft_maj=0 pgft_min=0 vm_size=0 vm_rss=0 vm_swap=0
259730 17:26:38.500764172 0 sshd (145515.145515) > getrandom
beheer@docker-macbook:~$
```

```
beheer@docker-macbook:~$ timeout 1 sudo sysdig -w 1sec.scap
beheer@docker-macbook:~$ sysdig -r 1sec.scap | wc -l
1022
beheer@docker-macbook:~$
beheer@docker-macbook:~$
```

DEMO SYSDIG

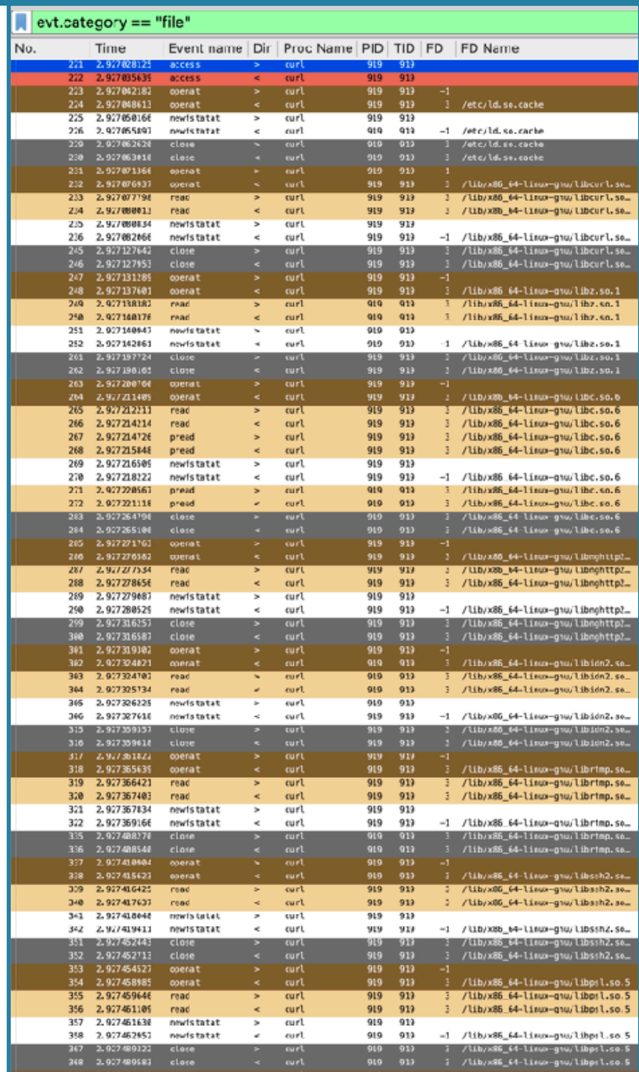
Stratoshark

- All the features of Wireshark... but for system calls
  - extensive filtering
  - filter buttons
  - expanding details
  - configuration profiles for easy switching
  - io graphs
  - etc
- Runs on Windows/MacOS/Linux
- Maintained by the Wireshark Foundation

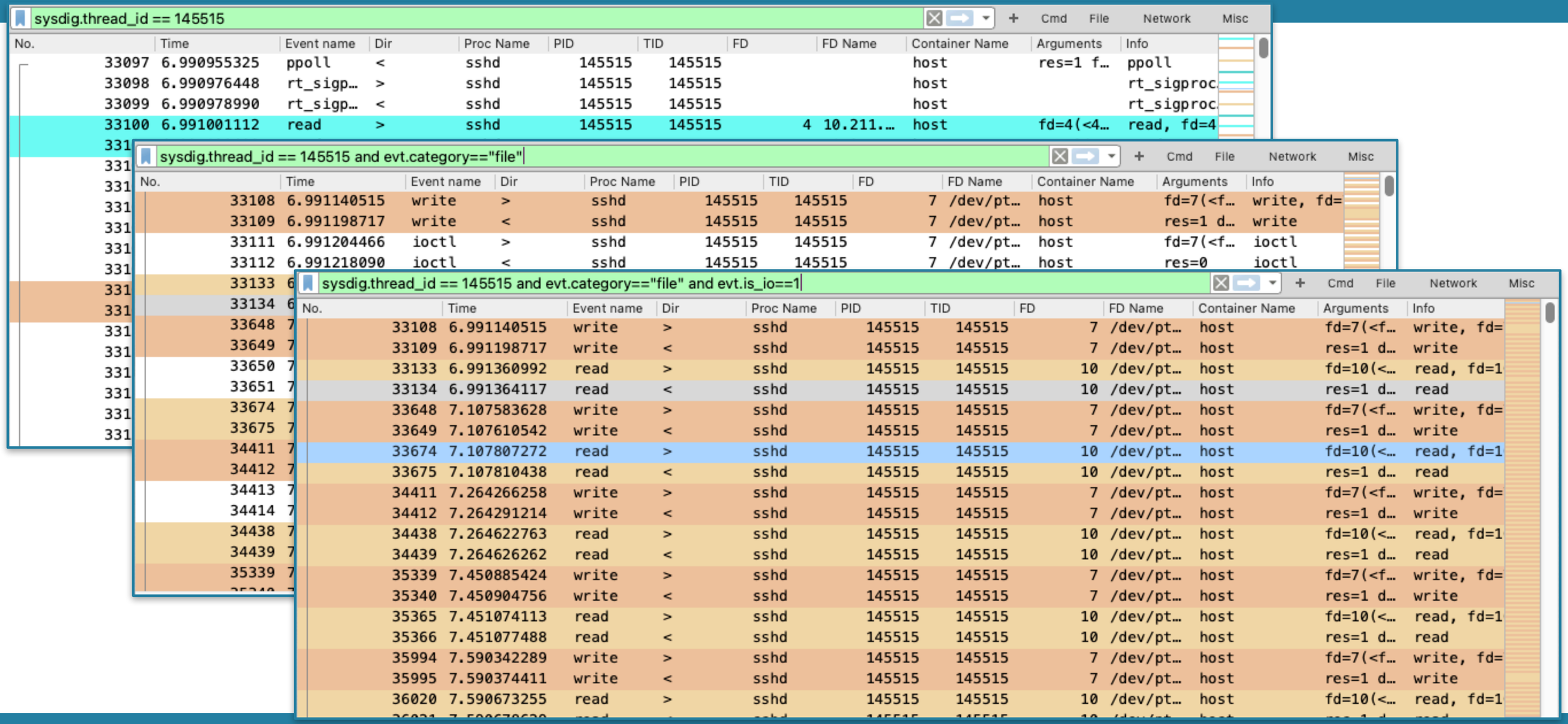
https://www.pexels.com/photo/purple-neon-light-signage-3960381/

How can we use Stratoshark?

- Linux
  - Capture local system calls
  - Analyze scap files
  - Capture system calls from remote Linux system over SSH
  - *Use falco plugins to ingest (cloud) logging*
- Windows / MacOS
  - Analyze scap files made remotely with sysdig/Stratoshark
  - Capture system calls from remote Linux system over SSH
  - **Not possible yet(!) to capture local system calls**
  - *Use falco plugins to ingest (cloud) logging into Stratoshark*



Drilling down...



Mixing categories...

sysdig.thread_id == 145515 and evt.category in {"net", "file"} and not sysdig.event_name=="ioctl"

+ Cmd File Network Misc

No.	Time	Event name	Dir	Proc Name	PID	TID	FD	FD Name	Container Name	Arguments	Info
33100	6.991001112	read	>	sshd	145515	145515	4	10.211.…	host	fd=4(<4…	read, fd=4
33101	6.991032692	read	<	sshd	145515	145515	4	10.211.…	host	res=36 …	read
33108	6.991140515	write	>	sshd	145515	145515	7	/dev/pt…	host	fd=7(<f…	write, fd=
33109	6.991198717	write	<	sshd	145515	145515	7	/dev/pt…	host	res=1 d…	write
33133	6.991360992	read	>	sshd	145515	145515	10	/dev/pt…	host	fd=10(<…	read, fd=1
33134	6.991364117	read	<	sshd	145515	145515	10	/dev/pt…	host	res=1 d…	read
33143	6.991390447	write	>	sshd	145515	145515	4	10.211.…	host	fd=4(<4…	write, fd=
33144	6.991494187	write	<	sshd	145515	145515	4	10.211.…	host	res=36 …	write
33640	7.107499512	read	>	sshd	145515	145515	4	10.211.…	host	fd=4(<4…	read, fd=4
33641	7.107518510	read	<	sshd	145515	145515	4	10.211.…	host	res=36 …	read
33648	7.107583628	write	>	sshd	145515	145515	7	/dev/pt…	host	fd=7(<f…	write, fd=
33649	7.107610542	write	<	sshd	145515	145515	7	/dev/pt…	host	res=1 d…	write
33674	7.107807272	read	>	sshd	145515	145515	10	/dev/pt…	host	fd=10(<…	read, fd=1
33675	7.107810438	read	<	sshd	145515	145515	10	/dev/pt…	host	res=1 d…	read
33684	7.107833936	write	>	sshd	145515	145515	4	10.211.…	host	fd=4(<4…	write, fd=
33685	7.107897429	write	<	sshd	145515	145515	4	10.211.…	host	res=36 …	write
34403	7.264201723	read	>	sshd	145515	145515	4	10.211.…	host	fd=4(<4…	read, fd=4
34404	7.264221221	read	<	sshd	145515	145515	4	10.211.…	host	res=36 …	read
34411	7.264266258	write	>	sshd	145515	145515	7	/dev/pt…	host	fd=7(<f…	write, fd=

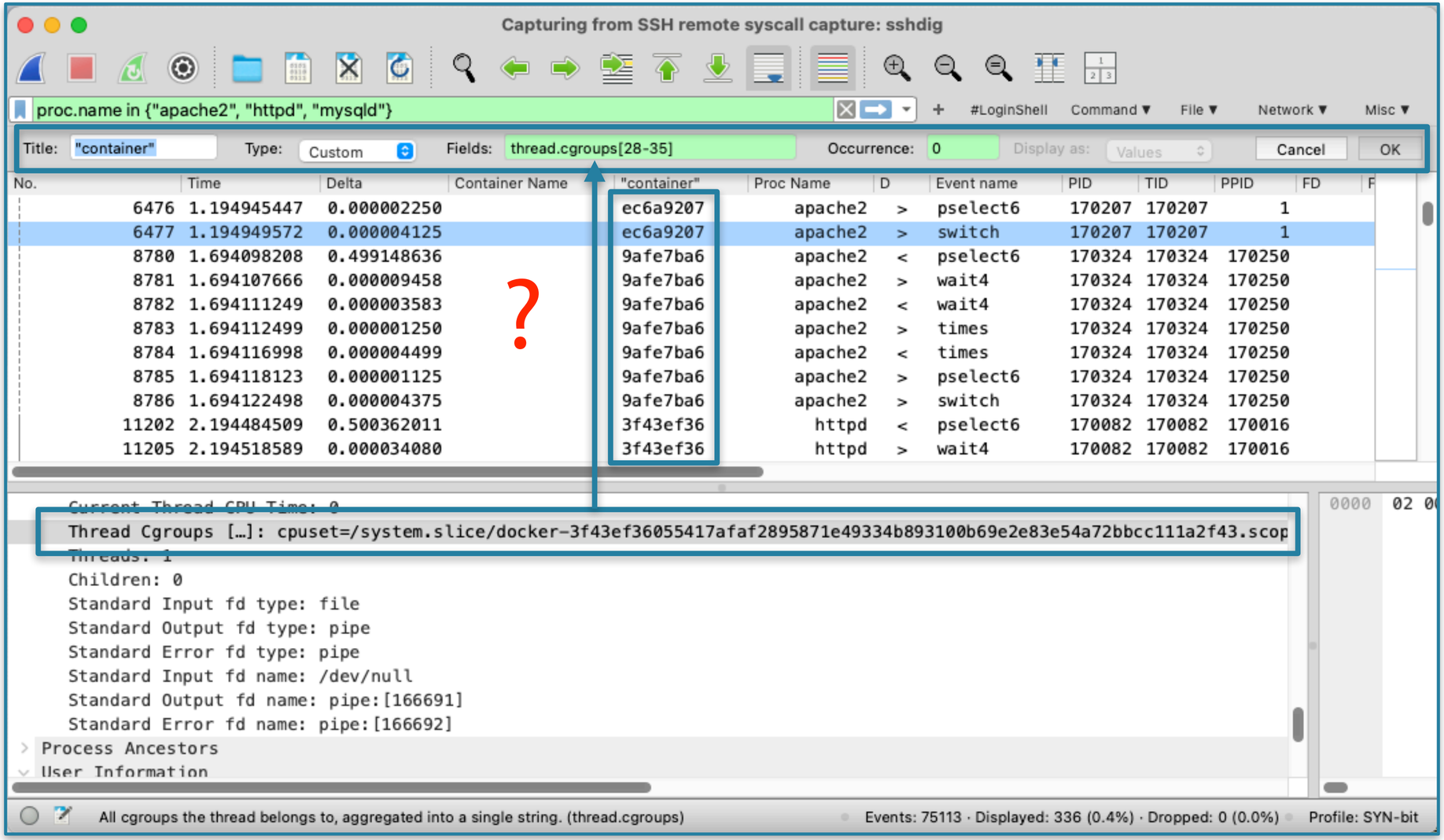
Combining processes...

(sysdig.thread_id in { 145515,145516} and evt.category in {"file"}) && evt.is_io==1

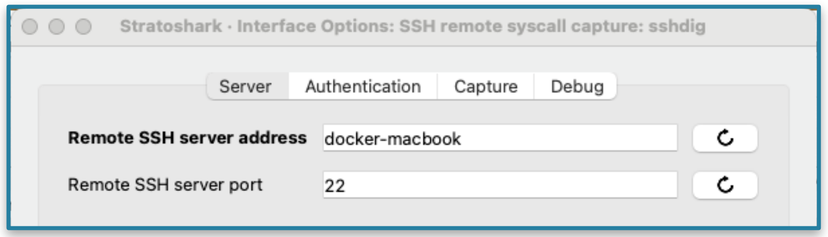
Cmd | File | Network | Misc

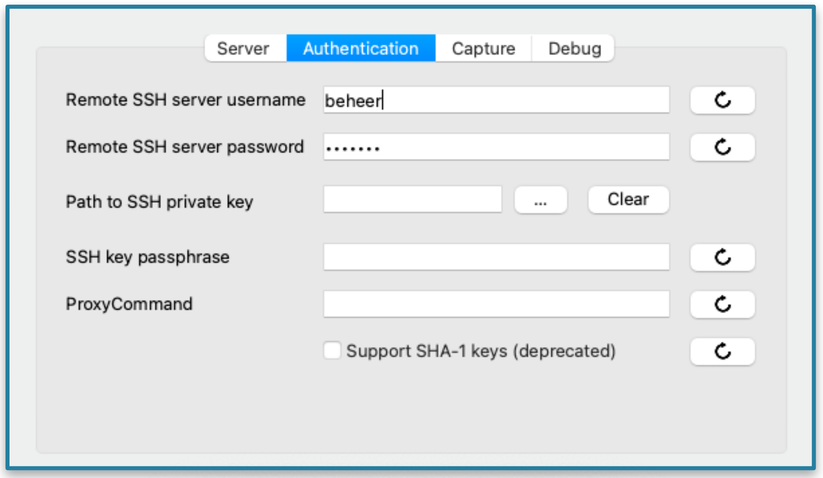
No.	Time	Event name	Dir	Proc Name	PID	TID	FD	FD Name	Container Name	Arguments	Info
33108	6.991140515	write	>	sshd	145515	145515	7	/dev/ptmx	host	fd=7(<f…	write,
33109	6.991198717	write	<	sshd	145515	145515	7	/dev/ptmx	host	res=1 d…	write
33119	6.991254420	read	>	bash	145516	145516	0	/dev/pts/0	host	fd=0(<f…	read,
33120	6.991263210	read	<	bash	145516	145516	0	/dev/pts/0	host	res=1 d…	read
33123	6.991312705	write	>	bash	145516	145516	2	/dev/pts/0	host	fd=2(<f…	write,
33124	6.991321871	write	<	bash	145516	145516	2	/dev/pts/0	host	res=1 d…	write
33133	6.991360992	read	>	sshd	145515	145515	10	/dev/ptmx	host	fd=10(<…	read,
33134	6.991364117	read	<	sshd	145515	145515	10	/dev/ptmx	host	res=1 d…	read
33648	7.107583628	write	>	sshd	145515	145515	7	/dev/ptmx	host	fd=7(<f…	write,
33649	7.107610542	write	<	sshd	145515	145515	7	/dev/ptmx	host	res=1 d…	write
33660	7.107696325	read	>	bash	145516	145516	0	/dev/pts/0	host	fd=0(<f…	read,
33661	7.107706074	read	<	bash	145516	145516	0	/dev/pts/0	host	res=1 d…	read
33664	7.107755902	write	>	bash	145516	145516	2	/dev/pts/0	host	fd=2(<f…	write,
33665	7.107766734	write	<	bash	145516	145516	2	/dev/pts/0	host	res=1 d…	write
33674	7.107807272	read	>	sshd	145515	145515	10	/dev/ptmx	host	fd=10(<…	read,
33675	7.107810438	read	<	sshd	145515	145515	10	/dev/ptmx	host	res=1 d…	read
34411	7.264266258	write	>	sshd	145515	145515	7	/dev/ptmx	host	fd=7(<f…	write,
34412	7.264291214	write	<	sshd	145515	145515	7	/dev/ptmx	host	res=1 d…	write
34423	7.264425700	read	>	bash	145516	145516	0	/dev/pts/0	host	fd=0(<f…	read,

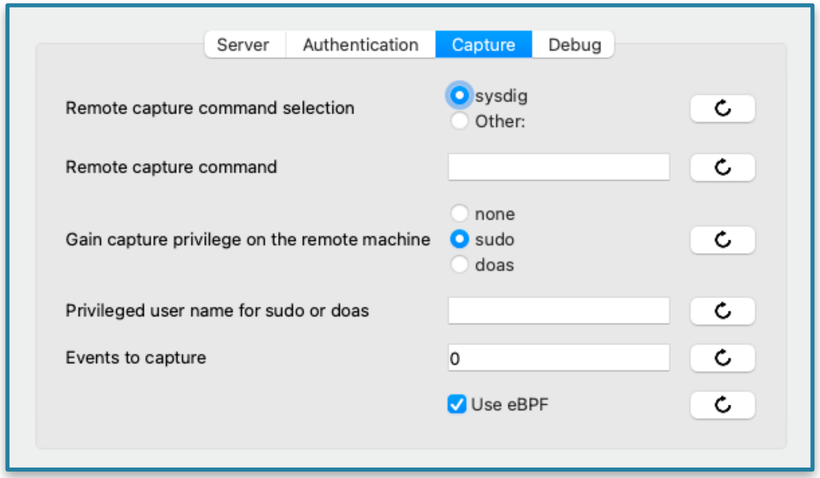
Workaround for the container name

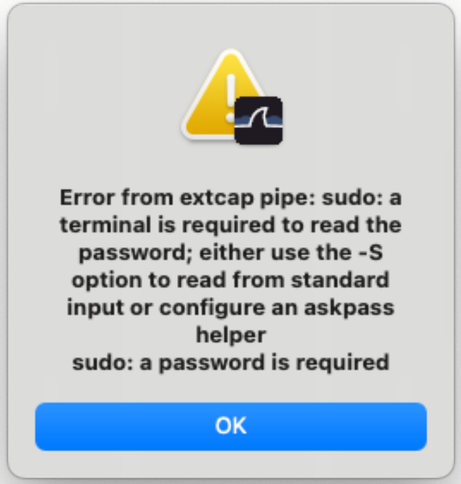


Setting up sshdig...

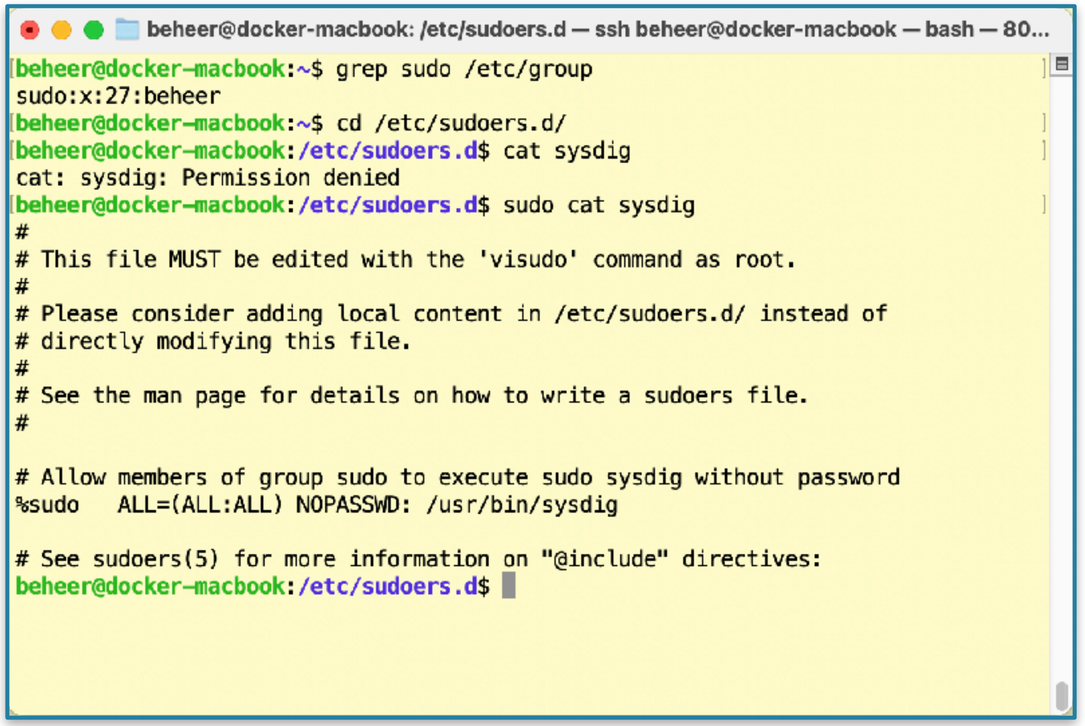


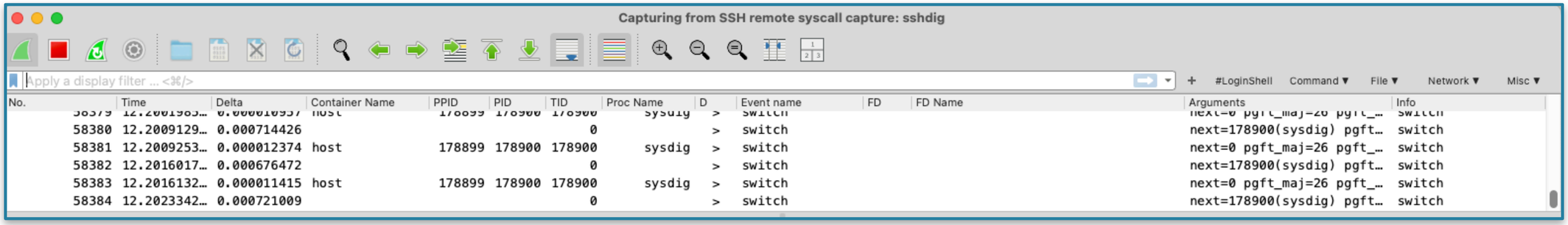






... with passwordless sudo!





System calls are boring (by themselves)

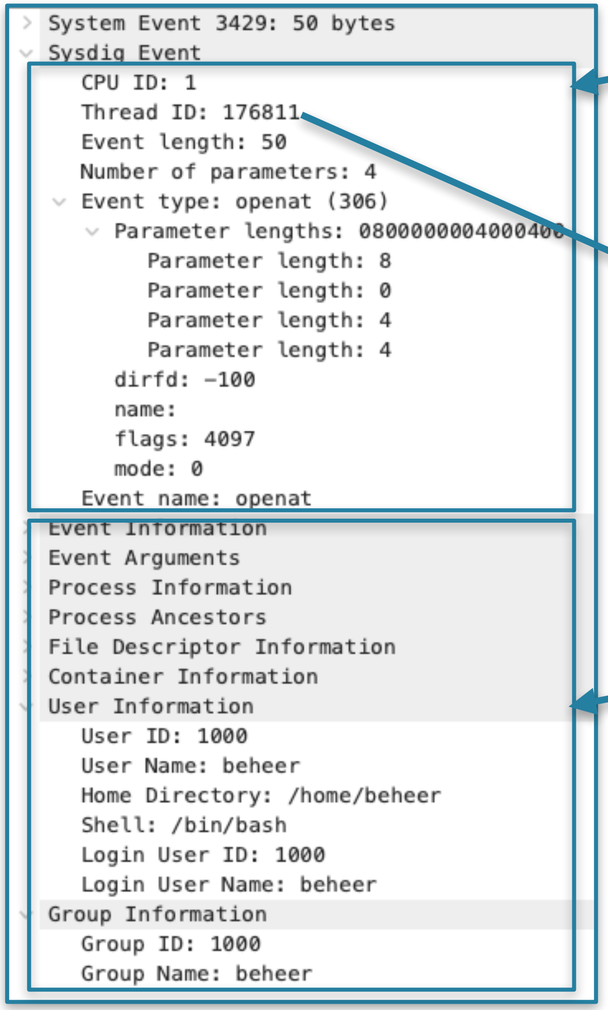
```
int openat(int dirfd, const char *pathname, int flags, mode_t mode);
```

But a lot more fun when enriched with metadata

- On start, libscap collects system state
  - containers, users, groups, processes, file descriptors, etc
- During capturing, libsinsp updates tables
  - So always a mirror of the system state available
  - Makes filtering on all kinds of information possible
  - Enables the inclusion of metadata in output (falco)
- Stratoshark shows system call data and all metadata

https://reef-aquarium-store.com/dardanus-pedunculatus-anemone-hermit-crab

Enriched data in stratoshark



system call, captured by libscap

use Thread ID to look up metadata

Add metadata to event

Context	Metadata	Field classes
Operating system	Processes and threads File descriptors Users and groups Network interfaces	`proc`, `thread`, `fd`, `fdlist`, `user`, `group`
Container	ID and name Type Image name Privileged Mount points Health checks	`container`
Kubernetes	Namespace Pod ReplicationController Service ReplicaSet Deployment	`k8s`

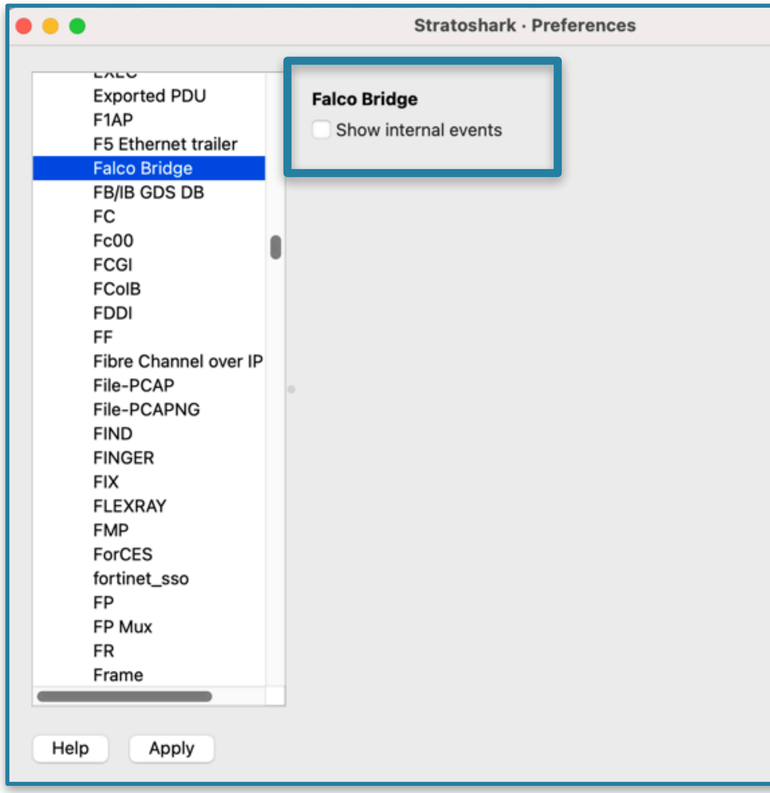
Missing events...

Apply a display filter ... <⌘/> + #LoginShell Command ▼ File ▼ Network ▼ Misc ▼

No.	Time	Delta	Container Name	"container"	Proc Name	D	Event name	PID	TID	PPID	FD
298	0.000444954	0.000000000		a8d1f262	ib_log_fi…	>	switch	170088	0	170020	
299	0.000445413	0.000000459	host	ice/sess	sysdig	>	switch	178900	178900	178899	
300	0.000450662	0.000005249	host	ice/sess	sshd	<	ppoll	178898	178898	178892	
301	0.000455786	0.000005124	host	ice/sess	sshd	>	rt_sigproc…	178898	178898	178892	
302	0.000456161	0.000000375	host	ice/sess	sshd	<	rt_sigproc…	178898	178898	178892	
303	0.000469743	0.000013582	host	ice/sess	sshd	>	brk	178898	178898	178892	
304	0.000472326	0.000002583	host	ice/sess	sshd	<	brk	178898	178898	178892	
305	0.000499615	0.000027289	host	ice/sess	sshd	>	read	178898	178898	178892	10
306	0.000507489	0.000007874	host	ice/sess	sshd	<	read	178898	178898	178892	10
307	0.000522405	0.000014916				>	switch		0		
308	0.000528362	0.000005957	host	ice/sess	sysdig	>	switch	178900	178900	178899	
309	0.000567775	0.000039413	host	ice/sess	sshd	>	getrandom	178898	178898	178892	

Apply a display filter ... <⌘/> + #LoginShell Command ▼ File ▼ Network ▼ Misc ▼

No.	Time	Delta	Container Name	"container"	Proc Name	D	Event name	PID	TID	PPID	FD
33436	6.902383004	0.000671598				>	switch		0		
33437	6.902410209	0.000027205	host	ice/sess	sysdig	>	switch	178900	178900	178899	
33438	6.903111720	0.000701511				>	switch		0		
33439	6.903120761	0.000009041	host	ice/sess	sysdig	>	switch	178900	178900	178899	
33440	6.903826688	0.000705927				>	switch		0		
33572	6.908457794	0.004631106				>	switch		0		
33573	6.908473209	0.000015415				>	switch		14		
33574	6.912152246	0.003679037				>	switch		0		
33575	6.912160495	0.000008249		a8d1f262	ib_log_fi…	<	futex	170088	170614	170020	
33576	6.912169952	0.000009457		a8d1f262	ib_log_fi…	>	futex	170088	170614	170020	
33577	6.912171786	0.000001834		a8d1f262	ib_log_fi…	<	futex	170088	170614	170020	
33578	6.912185867	0.000014081		a8d1f262	ib_log_fi…	>	futex	170088	170614	170020	



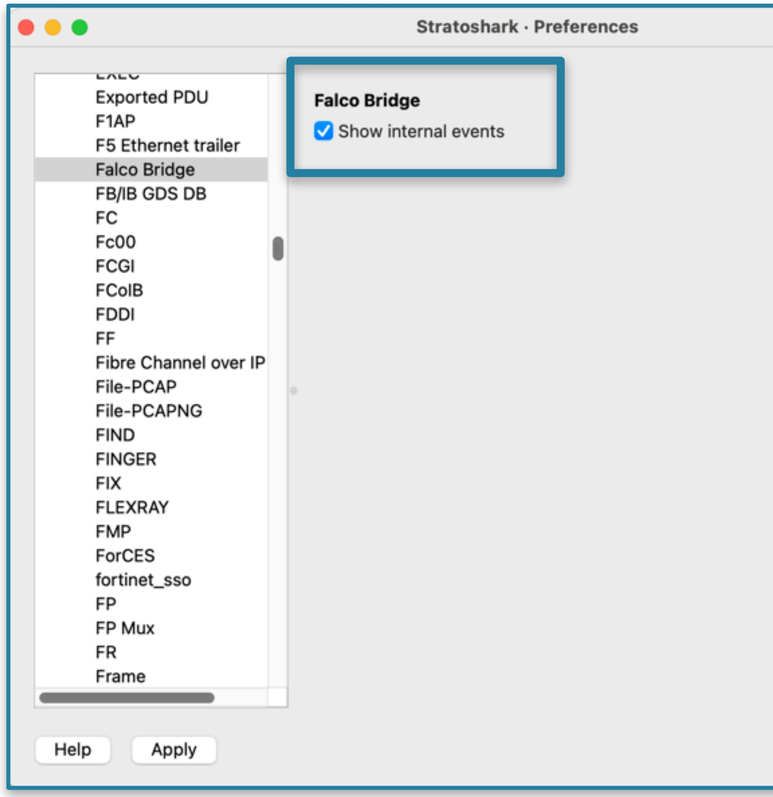
… are actually the metadata!

Apply a display filter … <⌘/> + #LoginShell Command ▼ File ▼ Network ▼ Misc ▼

No.	Time	Delta	Con	"cont	Pro	D	Event name	PID	TID	PPID	FD	FD Nar	Argument	Info
1	0.000000000	0.000000000					container							container [Internal event]
2	0.000000000	0.000000000					container							container [Internal event]
3	0.000000000	0.000000000					container							container [Internal event]
4	0.000000000	0.000000000					container							container [Internal event]
5	0.000000000	0.000000000					useradded							useradded [Internal event]
6	0.000000000	0.000000000					useradded							useradded [Internal event]
7	0.000000000	0.000000000					useradded							useradded [Internal event]
8	0.000000000	0.000000000					useradded							useradded [Internal event]
9	0.000000000	0.000000000					useradded							useradded [Internal event]
10	0.000000000	0.000000000					useradded							useradded [Internal event]
11	0.000000000	0.000000000					useradded							useradded [Internal event]
12	0.000000000	0.000000000					useradded							useradded [Internal event]
13	0.000000000	0.000000000					useradded							useradded [Internal event]

Apply a display filter … <⌘/> + #LoginShell Command ▼ File ▼ Network ▼ Misc ▼

No.	Time	Delta	Con	"cont	Pro	D	Event name	PID	TID	PPID	FD	FD Nar	Argument	Info
33438	6.903111720	0.000701511				>	switch		0				next=…	switch
33439	6.903120761	0.000009041	h…	ic…	s…	>	switch	1…	17…	17…			next=…	switch
33440	6.903826688	0.000705927				>	switch		0				next=…	switch
33441	6.903826688	0.000000000					procinfo							procinfo [Internal event]
33442	6.903826688	0.000000000					procinfo							procinfo [Internal event]
33443	6.903826688	0.000000000					procinfo							procinfo [Internal event]
33444	6.903826688	0.000000000					procinfo							procinfo [Internal event]
33445	6.903826688	0.000000000					procinfo							procinfo [Internal event]
33446	6.903826688	0.000000000					procinfo							procinfo [Internal event]
33447	6.903826688	0.000000000					procinfo							procinfo [Internal event]
33448	6.903826688	0.000000000					procinfo							procinfo [Internal event]
33449	6.903826688	0.000000000					procinfo							procinfo [Internal event]



What is this 'scap' file format?

- It's actually just pcapng...
- ...but with different block types

- packet data and event data can be in the same file (mergecap works), but Stratoshark crashes on the file (bug?)

- Difficult to make a profile that shows both packets and events in a clean way though...

0x00000201	Sysdig Machine Info Block
0x00000202	Sysdig Process Info Block, version 1
0x00000203	Sysdig FD List Block
0x00000204	Sysdig Event Block
0x00000205	Sysdig Interface List Block
0x00000206	Sysdig User List Block
0x00000207	Sysdig Process Info Block, version 2
0x00000208	Sysdig Event Block with flags
0x00000209	Sysdig Process Info Block, version 3
0x00000210	Sysdig Process Info Block, version 4
0x00000211	Sysdig Process Info Block, version 5
0x00000212	Sysdig Process Info Block, version 6
0x00000213	Sysdig Process Info Block, version 7

DEMO STRATOSHARK

Managing expectations...

- Stratoshark has just been born^W released...
  ... so still in its early stage!

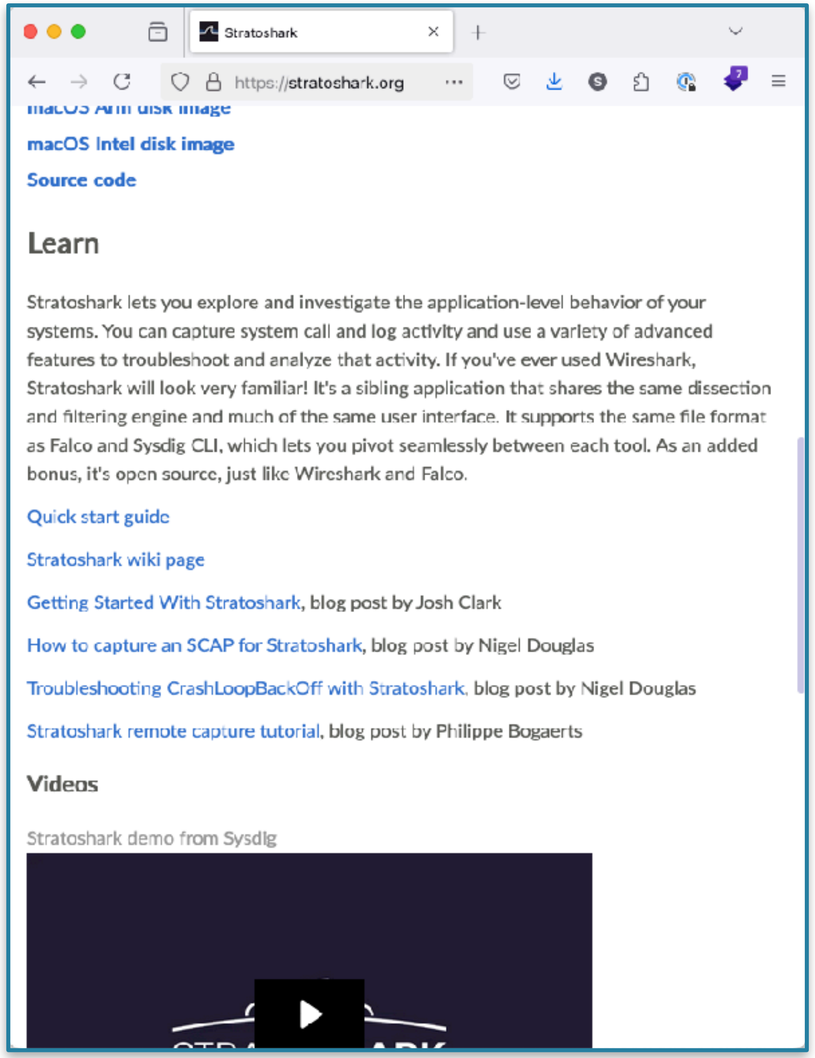
https://www.flickr.com/photos/verbeeldingskr8/28895969942

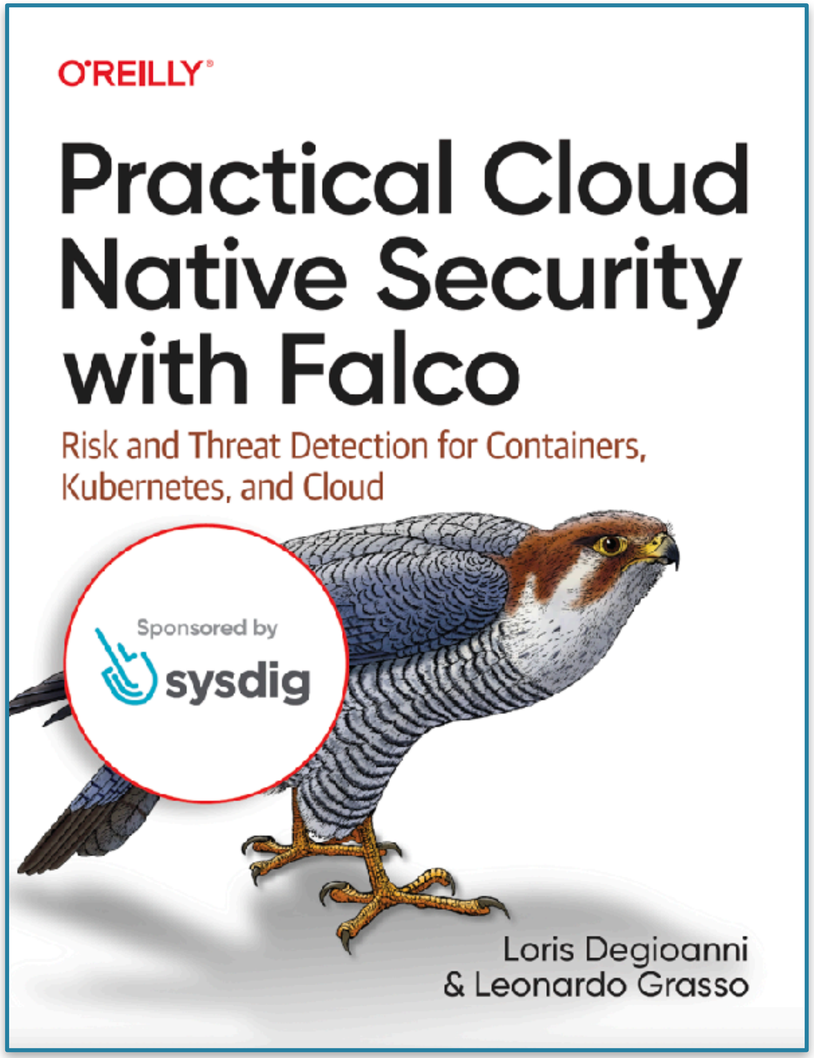
- Thus a bit rough on the edges (bugs...)
  - mergecap of pcapng and scap for instance
  - no container.name in Stratoshark (apparently a falco libs bug)
  - ~~File -> Export Specified Events... does not work correctly (use sysdig instead!)~~

- New (community) development will bring new features
  - Just like how Wireshark developed over time (25+ years by now!)


Q&A
You have
Questions
We have
Answers

Further reading...





FIN/ACK/FIN/ACK

Still questions?

sake.blok@SYN-bit.nl