May 8-9 2025, Zürich

What if Packets are not Enough ?

Rolf Leutert
Walter Hofstetter

PacketFest'25

Adding more Context to Packets

Adding more Context to Packets

What if we could enrich those packets with context?

Imagine capturing not only the raw bytes but also tagging each frame with the process ID, container name, or exact system call that emitted it. In this presentation, we'll introduce two relatively new approaches that bridge this visibility gap:

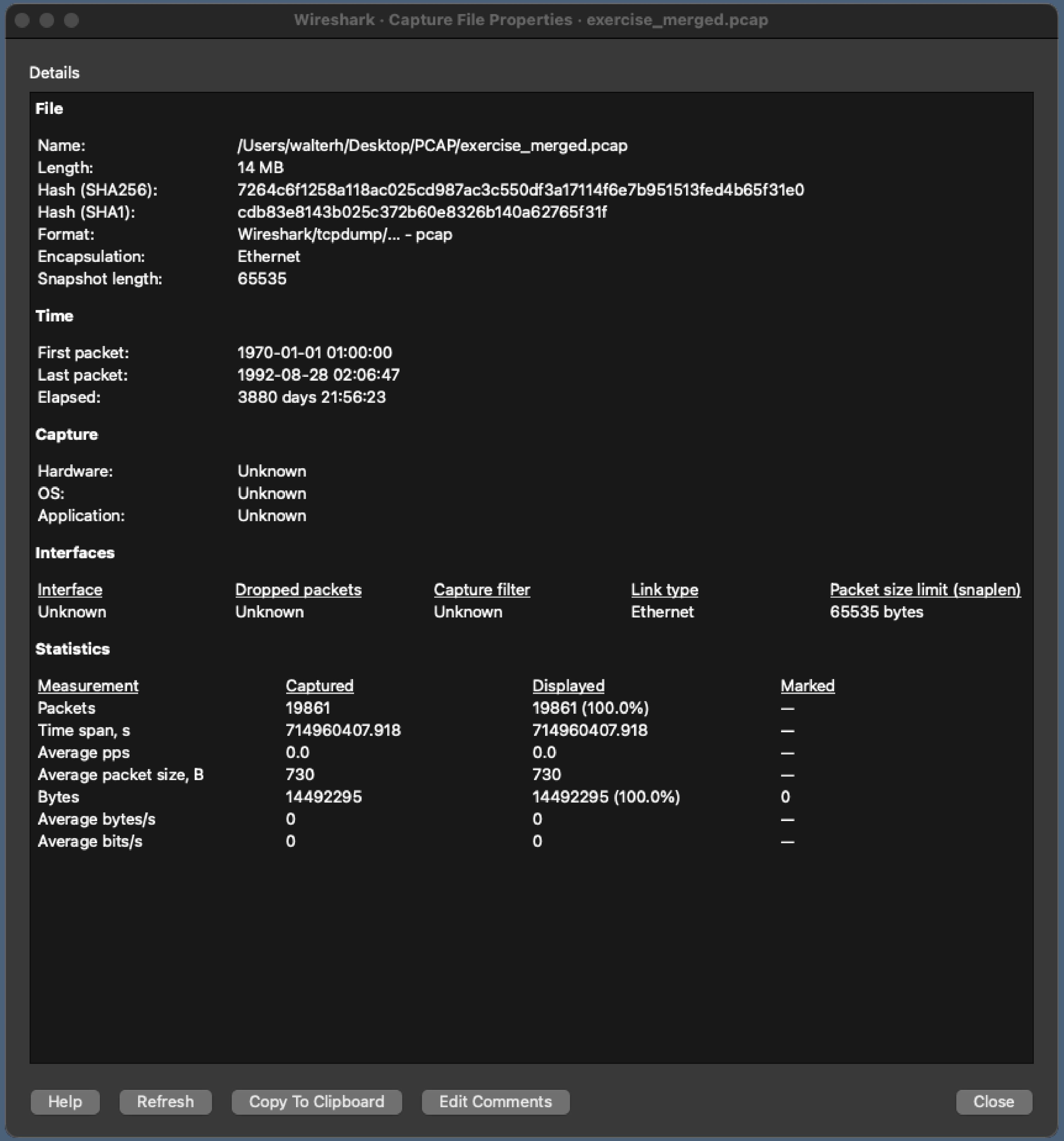
1. **ptcpdump - eBPF-Based Packet Annotation**
   Leveraging eBPF probes at the kernel level to attach process and namespace identifiers to each packet as it traverses the stack.

2. **Stratoshark - System Call and Cloud-Native Event Capture**
   Integrating with libraries behind Sysdig and Flaco (libscap and libsinsp). Stratoshark shares Wireshark's dissectors, filter syntax, and UI paradigms.

ptcpdump

ptcpdump

Remember the difference between pcap and pcapng?



Details

File

Name: /Users/walterh/Desktop/PCAP/exercise_merged.pcap
Length: 14 MB
Hash (SHA256): 7264c6f1258a118ac025cd987ac3c550df3a17114f6e7b951513fed4b65f31e0
Hash (SHA1): cdb83e8143b025c372b60e8326b140a62765f31f
Format: Wireshark/tcpdump/... - pcap
Encapsulation: Ethernet
Snapshot length: 65535

Time

First packet: 1970-01-01 01:00:00
Last packet: 1992-08-28 02:06:47
Elapsed: 3880 days 21:56:23

Capture

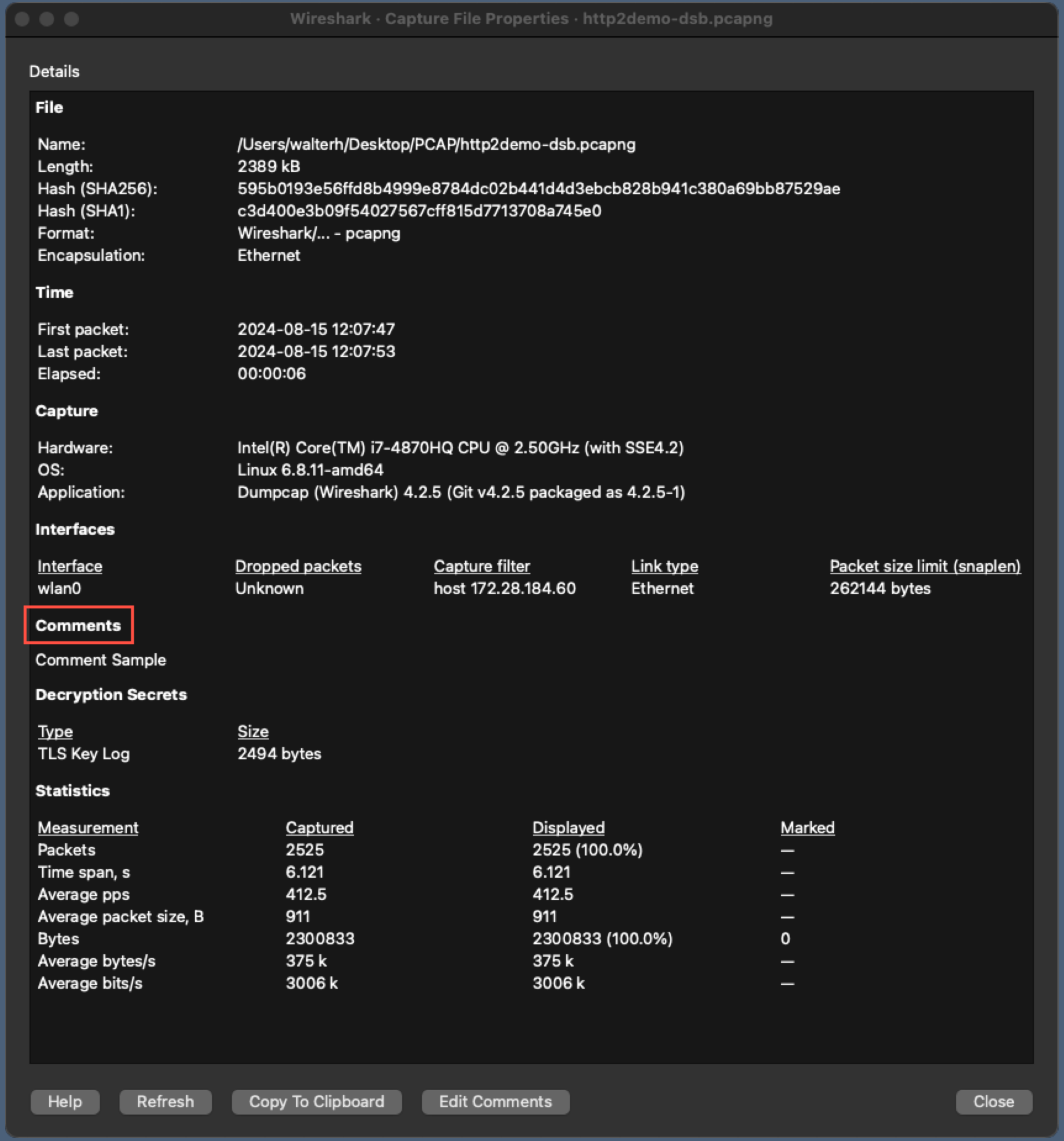
Hardware: Unknown
OS: Unknown
Application: Unknown

Interfaces

Interface	Dropped packets	Capture filter	Link type	Packet size limit (snaplen)
Unknown	Unknown	Unknown	Ethernet	65535 bytes

Statistics

Measurement	Captured	Displayed	Marked
Packets	19861	19861 (100.0%)	—
Time span, s	714960407.918	714960407.918	—
Average pps	0.0	0.0	—
Average packet size, B	730	730	—
Bytes	14492295	14492295 (100.0%)	0
Average bytes/s	0	0	—
Average bits/s	0	0	—



Details

File

Name: /Users/walterh/Desktop/PCAP/http2demo-dsb.pcapng
Length: 2389 kB
Hash (SHA256): 595b0193e56ffd8b4999e8784dc02b441d4d3ebcb828b941c380a69bb87529ae
Hash (SHA1): c3d400e3b09f54027567cff815d7713708a745e0
Format: Wireshark/... - pcapng
Encapsulation: Ethernet

Time

First packet: 2024-08-15 12:07:47
Last packet: 2024-08-15 12:07:53
Elapsed: 00:00:06

Capture

Hardware: Intel(R) Core(TM) i7-4870HQ CPU @ 2.50GHz (with SSE4.2)
OS: Linux 6.8.11-amd64
Application: Dumpcap (Wireshark) 4.2.5 (Git v4.2.5 packaged as 4.2.5-1)

Interfaces

Interface	Dropped packets	Capture filter	Link type	Packet size limit (snaplen)
wlan0	Unknown	host 172.28.184.60	Ethernet	262144 bytes

Comments

Comment Sample

Decryption Secrets

Type	Size
TLS Key Log	2494 bytes

Statistics

Measurement	Captured	Displayed	Marked
Packets	2525	2525 (100.0%)	—
Time span, s	6.121	6.121	—
Average pps	412.5	412.5	—
Average packet size, B	911	911	—
Bytes	2300833	2300833 (100.0%)	0
Average bytes/s	375 k	375 k	—
Average bits/s	3006 k	3006 k	—

ptcpdump

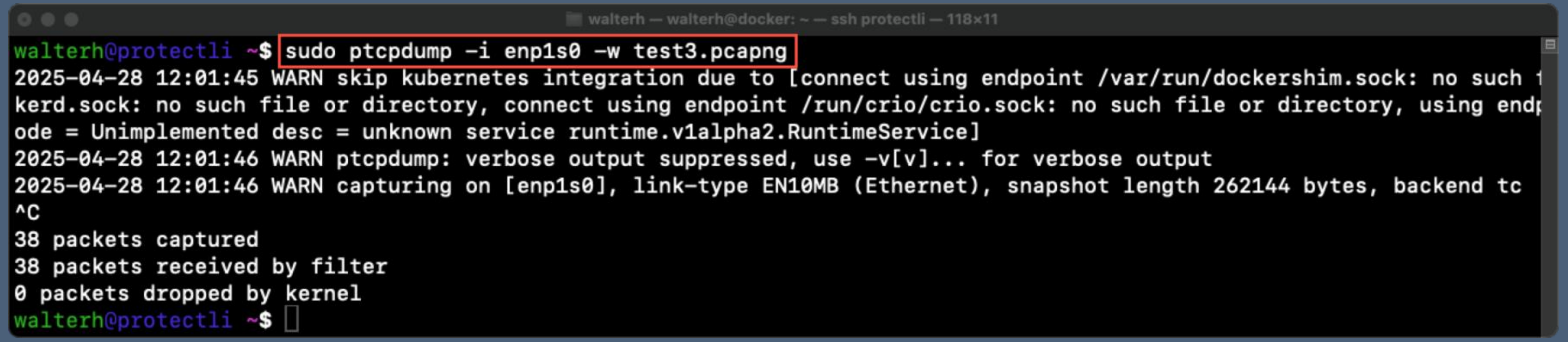
Source: https://github.com/mozillazg/ptcpdump

- Process/container/pod-aware packet capture
- Filter by: --pid (process), --pname (process name), --container-id (container), --pod-name (pod)
- tcpdump-compatible flags (-i, -w, -c, -s, -n, -C, -W, -A, and more)
- Supports pcap-filter(7) syntax like tcpdump
- tcpdump-like output + process/container/pod context
- Verbose mode shows detailed metadata for processes and containers/pods
- PcapNG with embedded metadata
- Cross-namespace capture (--netns)
- Kernel-space BPF filtering (low overhead, reduces CPU usage)
- Container runtime integration (Docker, containerd)

ptcpdump

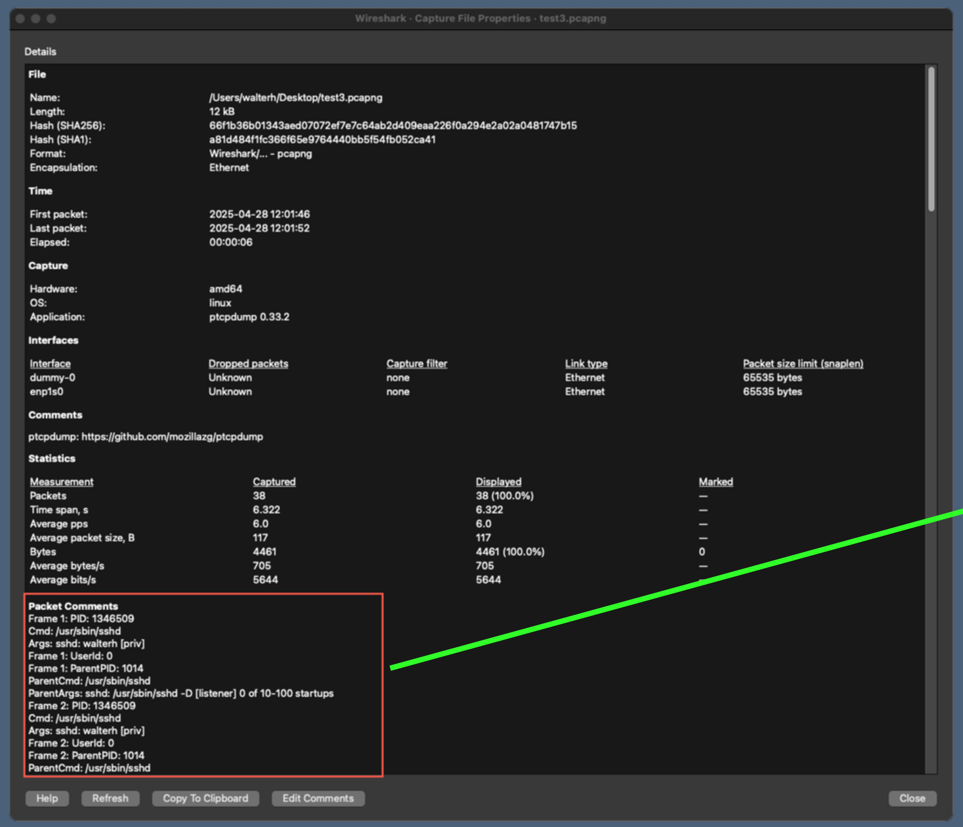
Examples:

- Use the same syntax as with *tcpdump*



```
walterh@protectli ~$ sudo ptcpdump -i enp1s0 -w test3.pcapng
2025-04-28 12:01:45 WARN skip kubernetes integration due to [connect using endpoint /var/run/dockershim.sock: no such f
kerd.sock: no such file or directory, connect using endpoint /run/crio/crio.sock: no such file or directory, using endp
ode = Unimplemented desc = unknown service runtime.v1alpha2.RuntimeService]
2025-04-28 12:01:46 WARN ptcpdump: verbose output suppressed, use -v[v]... for verbose output
2025-04-28 12:01:46 WARN capturing on [enp1s0], link-type EN10MB (Ethernet), snapshot length 262144 bytes, backend tc
^C
38 packets captured
38 packets received by filter
0 packets dropped by kernel
walterh@protectli ~$
```

- Enrichment saved to the *Comment Field*



File

Name:	/Users/walterh/Desktop/test3.pcapng
Length:	12 kB
Hash (SHA256):	66f1b36b01343aed07072ef7e7c64ab2d409eaa226f0a294e2a02a0481747b15
Hash (SHA1):	a81d484f1fc366f65e9764440bb5f54fb052ca41
Format:	Wireshark/... - pcapng
Encapsulation:	Ethernet

Time

First packet:	2025-04-28 12:01:46
Last packet:	2025-04-28 12:01:52
Elapsed:	00:00:06

Capture

Hardware:	amd64
OS:	linux
Application:	ptcpdump 0.33.2

Interfaces

Interface	Dropped packets	Capture filter	Link type	Packet size limit (snaplen)
dummy-0	Unknown	none	Ethernet	65535 bytes
enp1s0	Unknown	none	Ethernet	65535 bytes

Comments

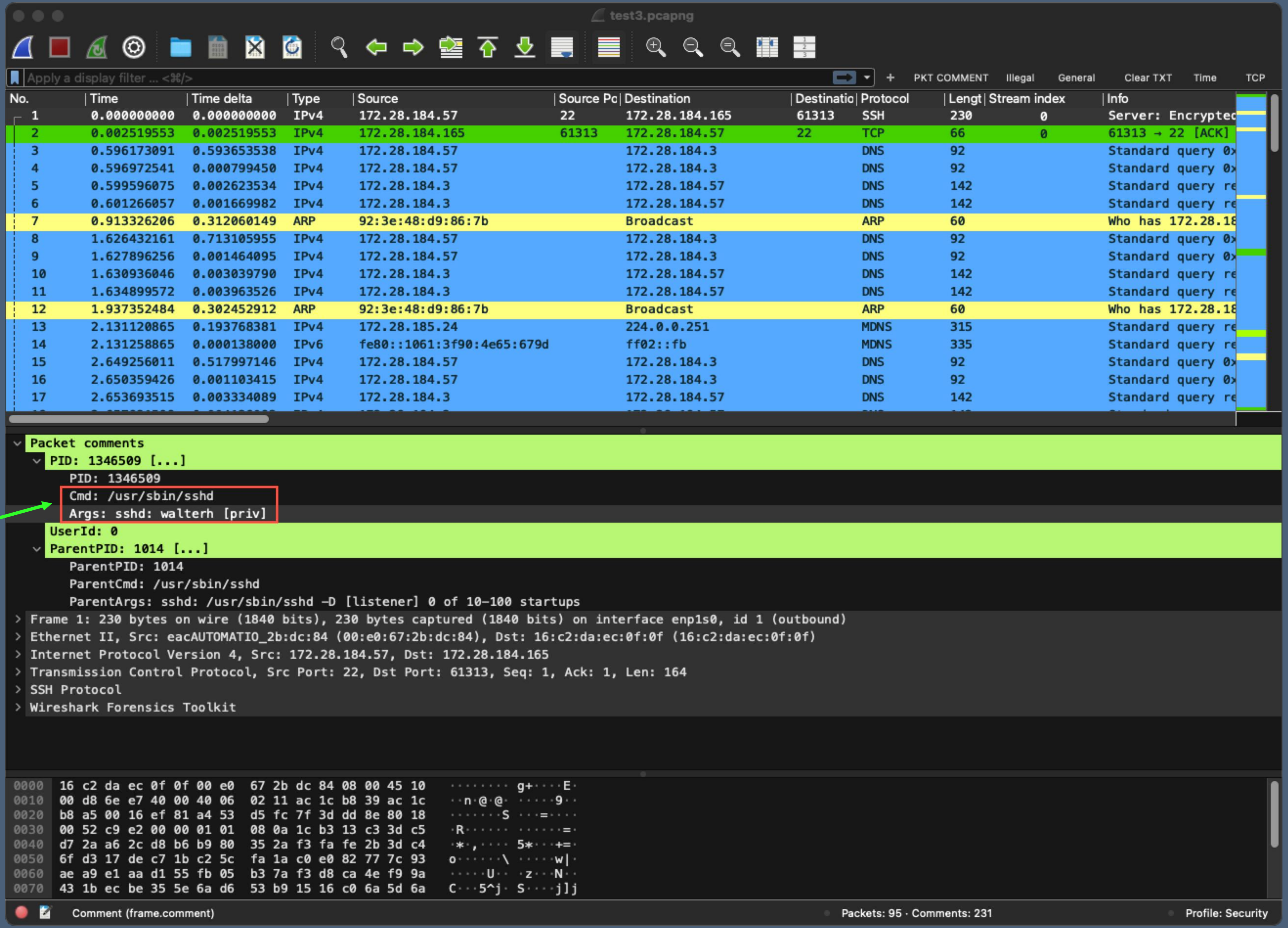
ptcpdump: https://github.com/mozillazg/ptcpdump

Statistics

Measurement	Captured	Displayed	Marked
Packets	38	38 (100.0%)	—
Time span, s	6.322	6.322	—
Average pps	6.0	6.0	—
Average packet size, B	117	117	—
Bytes	4461	4461 (100.0%)	0
Average bytes/s	705	705	—
Average bits/s	5644	5644	—

Packet Comments

Frame 1: PID: 1346509
Cmd: /usr/sbin/sshd
Args: sshd: walterh [priv]
Frame 1: UserId: 0
Frame 1: ParentPID: 1014
ParentCmd: /usr/sbin/sshd
ParentArgs: sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups
Frame 2: PID: 1346509
Cmd: /usr/sbin/sshd
Args: sshd: walterh [priv]
Frame 2: UserId: 0
Frame 2: ParentPID: 1014
ParentCmd: /usr/sbin/sshd

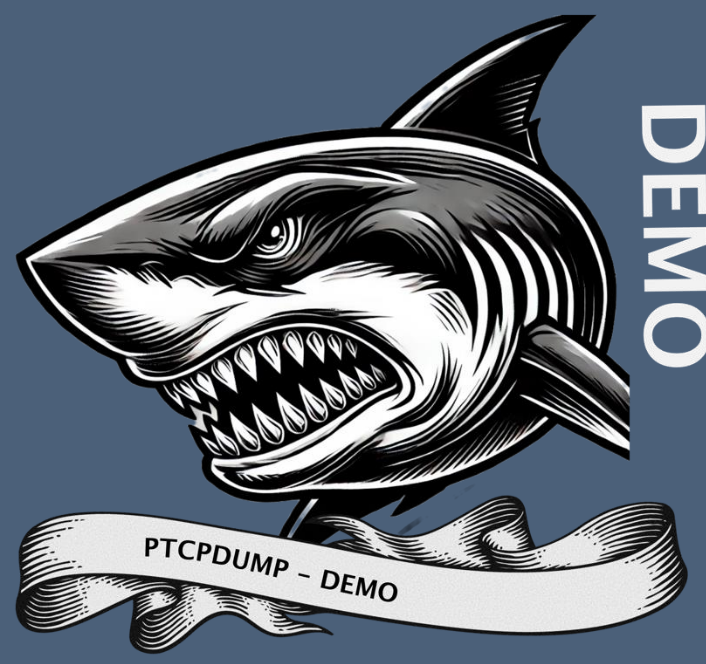


No.	Time	Time delta	Type	Source	Source Po	Destination	Destinatio	Protocol	Lengt	Stream index	Info
1	0.000000000	0.000000000	IPv4	172.28.184.57	22	172.28.184.165	61313	SSH	230	0	Server: Encrypte
2	0.002519553	0.002519553	IPv4	172.28.184.165	61313	172.28.184.57	22	TCP	66	0	61313 → 22 [ACK]
3	0.596173091	0.593653538	IPv4	172.28.184.57		172.28.184.3		DNS	92		Standard query 0x
4	0.596972541	0.000799450	IPv4	172.28.184.57		172.28.184.3		DNS	92		Standard query 0x
5	0.599596075	0.002623534	IPv4	172.28.184.3		172.28.184.57		DNS	142		Standard query re
6	0.601266057	0.001669982	IPv4	172.28.184.3		172.28.184.57		DNS	142		Standard query re
7	0.913326206	0.312060149	ARP	92:3e:48:d9:86:7b		Broadcast		ARP	60		Who has 172.28.18
8	1.626432161	0.713105955	IPv4	172.28.184.57		172.28.184.3		DNS	92		Standard query 0x
9	1.627896256	0.001464095	IPv4	172.28.184.57		172.28.184.3		DNS	92		Standard query 0x
10	1.630936046	0.003039790	IPv4	172.28.184.3		172.28.184.57		DNS	142		Standard query re
11	1.634899572	0.003963526	IPv4	172.28.184.3		172.28.184.57		DNS	142		Standard query re
12	1.937352484	0.302452912	ARP	92:3e:48:d9:86:7b		Broadcast		ARP	60		Who has 172.28.18
13	2.131120865	0.193768381	IPv4	172.28.185.24		224.0.0.251		MDNS	315		Standard query re
14	2.131258865	0.000138000	IPv6	fe80::1061:3f90:4e65:679d		ff02::fb		MDNS	335		Standard query re
15	2.649256011	0.517997146	IPv4	172.28.184.57		172.28.184.3		DNS	92		Standard query 0x
16	2.650359426	0.001103415	IPv4	172.28.184.57		172.28.184.3		DNS	92		Standard query 0x
17	2.653693515	0.003334089	IPv4	172.28.184.3		172.28.184.57		DNS	142		Standard query re

```
∨ Packet comments
  ∨ PID: 1346509 [...]
      PID: 1346509
      Cmd: /usr/sbin/sshd
      Args: sshd: walterh [priv]
    UserId: 0
  ∨ ParentPID: 1014 [...]
      ParentPID: 1014
      ParentCmd: /usr/sbin/sshd
      ParentArgs: sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups
> Frame 1: 230 bytes on wire (1840 bits), 230 bytes captured (1840 bits) on interface enp1s0, id 1 (outbound)
> Ethernet II, Src: eacAUTOMATIO_2b:dc:84 (00:e0:67:2b:dc:84), Dst: 16:c2:da:ec:0f:0f (16:c2:da:ec:0f:0f)
> Internet Protocol Version 4, Src: 172.28.184.57, Dst: 172.28.184.165
> Transmission Control Protocol, Src Port: 22, Dst Port: 61313, Seq: 1, Ack: 1, Len: 164
> SSH Protocol
> Wireshark Forensics Toolkit
```

```
0000  16 c2 da ec 0f 0f 00 e0  67 2b dc 84 08 00 45 10   ········ g+····E·
0010  00 d8 6e e7 40 00 40 06  02 11 ac 1c b8 39 ac 1c   ··n·@·@· ·····9··
0020  b8 a5 00 16 ef 81 a4 53  d5 fc 7f 3d dd 8e 80 18   ·······S ···=····
0030  00 52 c9 e2 00 00 01 01  08 0a 1c b3 13 c3 3d c5   ·R······ ······=·
0040  d7 2a a6 2c d8 b6 b9 80  35 2a f3 fa fe 2b 3d c4   ·*·,···· 5*···+=·
0050  6f d3 17 de c7 1b c2 5c  fa 1a c0 e0 82 77 7c 93   o······\ ·····w|·
0060  ae a9 e1 aa d1 55 fb 05  b3 7a f3 d8 ca 4e f9 9a   ·····U·· ·z···N··
0070  43 1b ec be 35 5e 6a d6  53 b9 15 16 c0 6a 5d 6a   C···5^j· S····j]j
```

Comment (frame.comment)   Packets: 95 · Comments: 231   Profile: Security



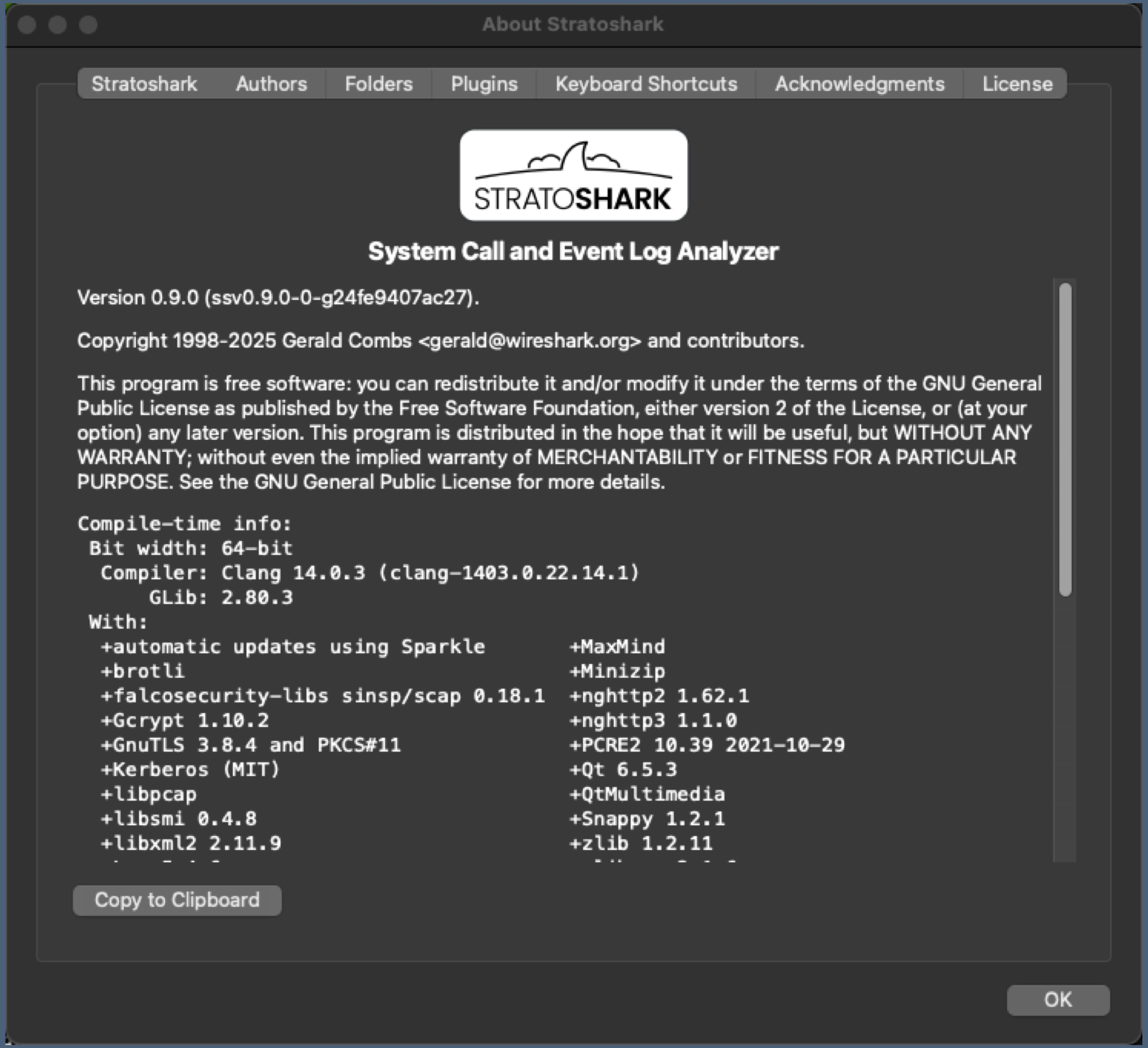
DEMO

Useing ptcpdump (DEMO)

STRATOSHARK

What is Stratoshark

Stratoshark gives you deep visibility into your systems' application-level activity. You can record system calls and log events, then leverage a suite of powerful troubleshooting and analysis tools to inspect that data. If you're familiar with Wireshark, you'll feel right at home—Stratoshark uses the same dissection and filtering engine and shares much of its interface. It also reads the same file format as Falco and the Sysdig CLI, making it easy to switch back and forth between tools. Plus, like Wireshark and Falco, Stratoshark is completely open source.



System Call and Event Log Analyzer

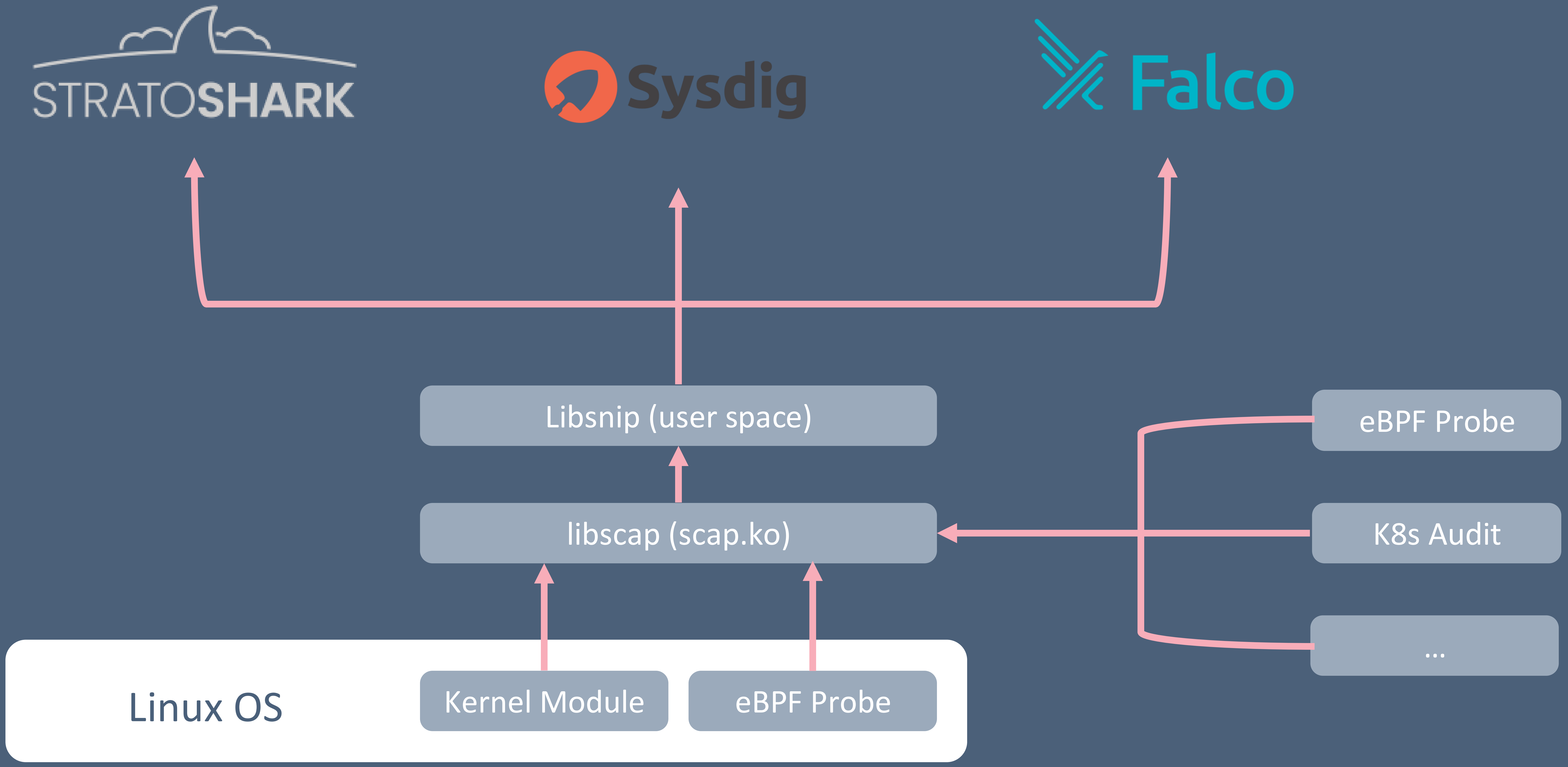
Version 0.9.0 (ssv0.9.0-0-g24fe9407ac27).

Copyright 1998-2025 Gerald Combs <gerald@wireshark.org> and contributors.

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 2 of the License, or (at your option) any later version. This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

```
Compile-time info:
 Bit width: 64-bit
  Compiler: Clang 14.0.3 (clang-1403.0.22.14.1)
      GLib: 2.80.3
 With:
  +automatic updates using Sparkle       +MaxMind
  +brotli                                +Minizip
  +falcosecurity-libs sinsp/scap 0.18.1  +nghttp2 1.62.1
  +Gcrypt 1.10.2                         +nghttp3 1.1.0
  +GnuTLS 3.8.4 and PKCS#11              +PCRE2 10.39 2021-10-29
  +Kerberos (MIT)                        +Qt 6.5.3
  +libpcap                               +QtMultimedia
  +libsmi 0.4.8                          +Snappy 1.2.1
  +libxml2 2.11.9                        +zlib 1.2.11
```

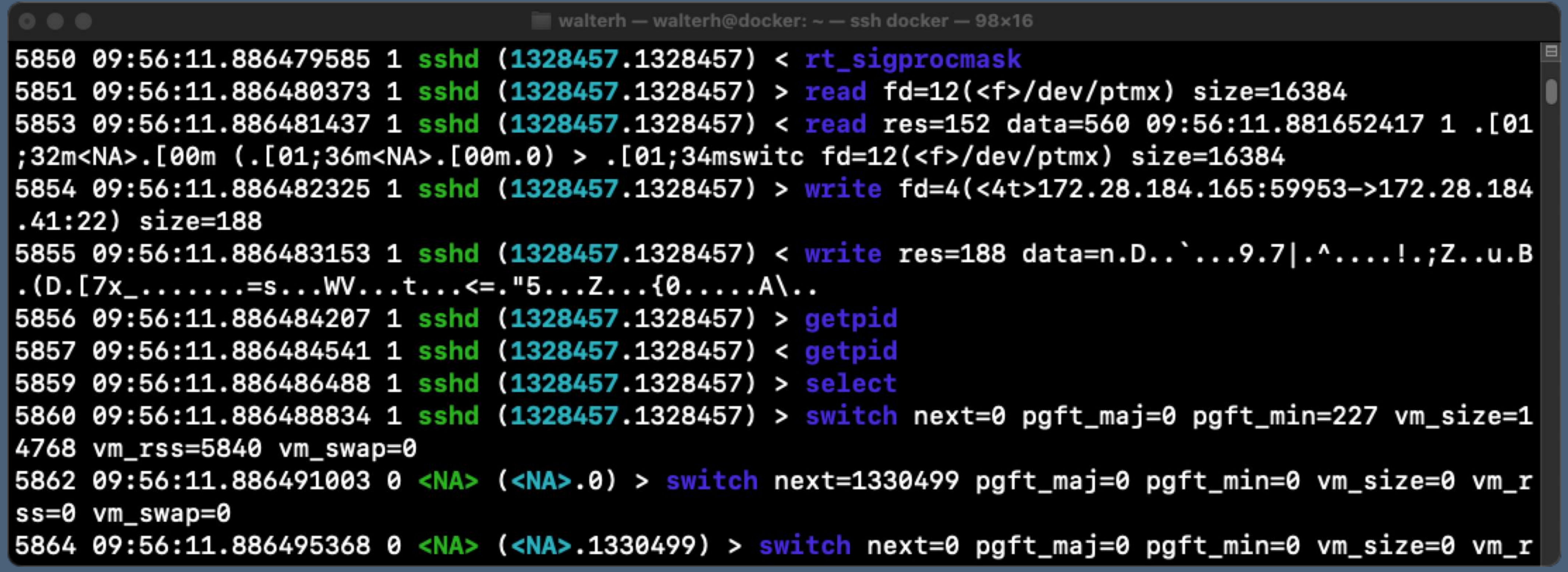
Stratoshark Architecture



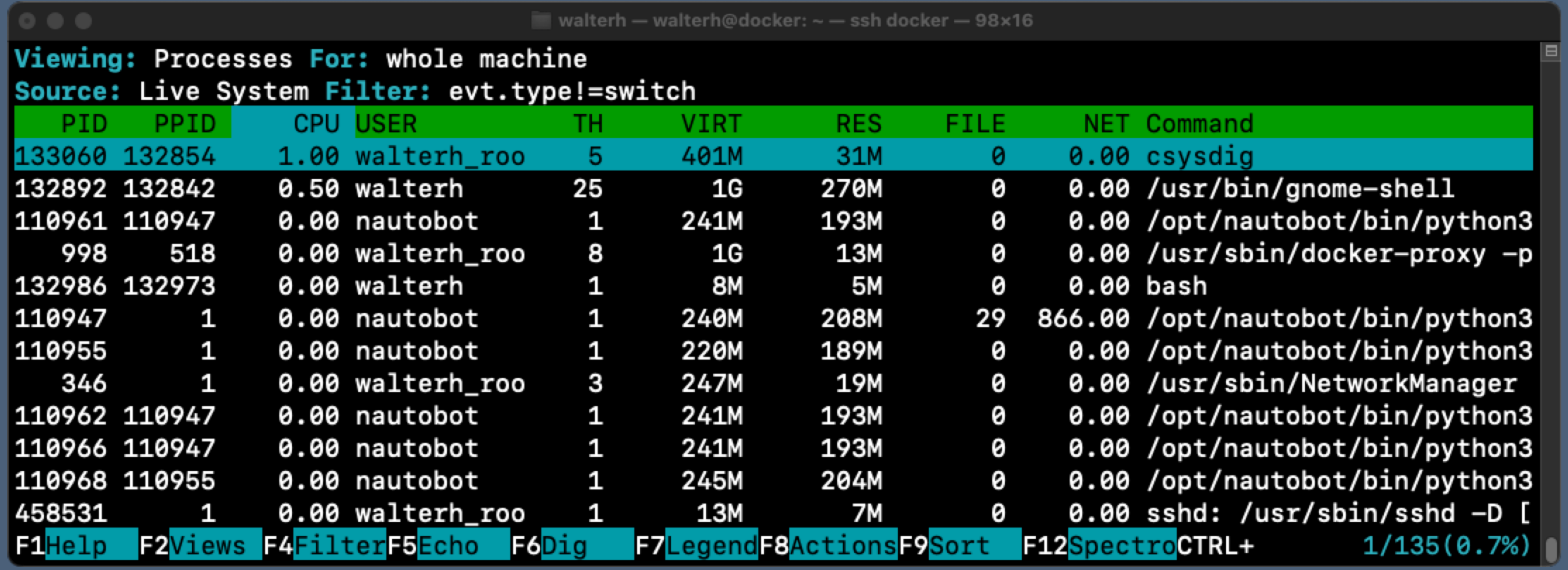
Sysdig

Sysdig instruments your physical and virtual machines at the OS level by installing into the Linux kernel and capturing system calls and other OS events. Sysdig also makes it possible to create trace files for system activity, similarly to what you can do for networks with tools like tcpdump and Wireshark. This way, problems can be analyzed at a later time, without losing important information. Rich system state is stored in the trace files, so that the captured activity can be put into full context.

https://github.com/draios/sysdig/

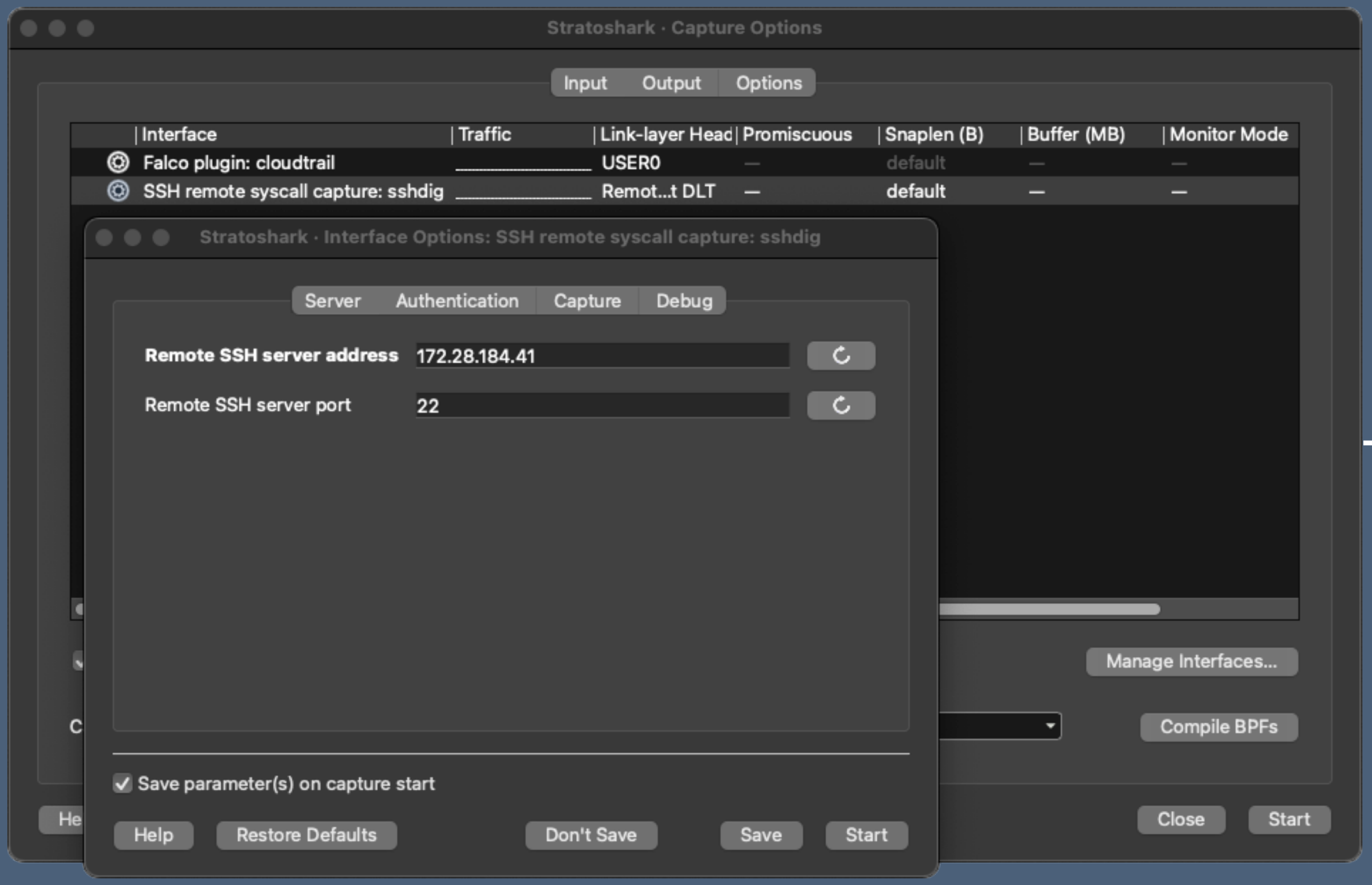


```
5850 09:56:11.886479585 1 sshd (1328457.1328457) < rt_sigprocmask
5851 09:56:11.886480373 1 sshd (1328457.1328457) > read fd=12(<f>/dev/ptmx) size=16384
5853 09:56:11.886481437 1 sshd (1328457.1328457) < read res=152 data=560 09:56:11.881652417 1 .[01
;32m<NA>.[00m (.[01;36m<NA>.[00m.0) > .[01;34mswitc fd=12(<f>/dev/ptmx) size=16384
5854 09:56:11.886482325 1 sshd (1328457.1328457) > write fd=4(<4t>172.28.184.165:59953->172.28.184
.41:22) size=188
5855 09:56:11.886483153 1 sshd (1328457.1328457) < write res=188 data=n.D..`...9.7|.^....!.;Z..u.B
.(D.[7x_.......=s...WV...t...<=."5...Z...{0.....A\..
5856 09:56:11.886484207 1 sshd (1328457.1328457) > getpid
5857 09:56:11.886484541 1 sshd (1328457.1328457) < getpid
5859 09:56:11.886486488 1 sshd (1328457.1328457) > select
5860 09:56:11.886488834 1 sshd (1328457.1328457) > switch next=0 pgft_maj=0 pgft_min=227 vm_size=1
4768 vm_rss=5840 vm_swap=0
5862 09:56:11.886491003 0 <NA> (<NA>.0) > switch next=1330499 pgft_maj=0 pgft_min=0 vm_size=0 vm_r
ss=0 vm_swap=0
5864 09:56:11.886495368 0 <NA> (<NA>.1330499) > switch next=0 pgft_maj=0 pgft_min=0 vm_size=0 vm_r
```



```
Viewing: Processes For: whole machine
Source: Live System Filter: evt.type!=switch
   PID   PPID     CPU USER          TH      VIRT      RES    FILE     NET Command
133060 132854    1.00 walterh_roo    5      401M      31M       0    0.00 csysdig
132892 132842    0.50 walterh       25        1G     270M       0    0.00 /usr/bin/gnome-shell
110961 110947    0.00 nautobot       1      241M     193M       0    0.00 /opt/nautobot/bin/python3
   998    518    0.00 walterh_roo    8        1G      13M       0    0.00 /usr/sbin/docker-proxy -p
132986 132973    0.00 walterh        1        8M       5M       0    0.00 bash
110947      1    0.00 nautobot       1      240M     208M      29  866.00 /opt/nautobot/bin/python3
110955      1    0.00 nautobot       1      220M     189M       0    0.00 /opt/nautobot/bin/python3
   346      1    0.00 walterh_roo    3      247M      19M       0    0.00 /usr/sbin/NetworkManager
110962 110947    0.00 nautobot       1      241M     193M       0    0.00 /opt/nautobot/bin/python3
110966 110947    0.00 nautobot       1      241M     193M       0    0.00 /opt/nautobot/bin/python3
110968 110955    0.00 nautobot       1      245M     204M       0    0.00 /opt/nautobot/bin/python3
458531      1    0.00 walterh_roo    1       13M       7M       0    0.00 sshd: /usr/sbin/sshd -D [
F1Help  F2Views F4FilterF5Echo  F6Dig   F7LegendF8ActionsF9Sort  F12SpectroCTRL+      1/135(0.7%)
```

Sysdig & Stratoshark



Stratoshark · Capture Options

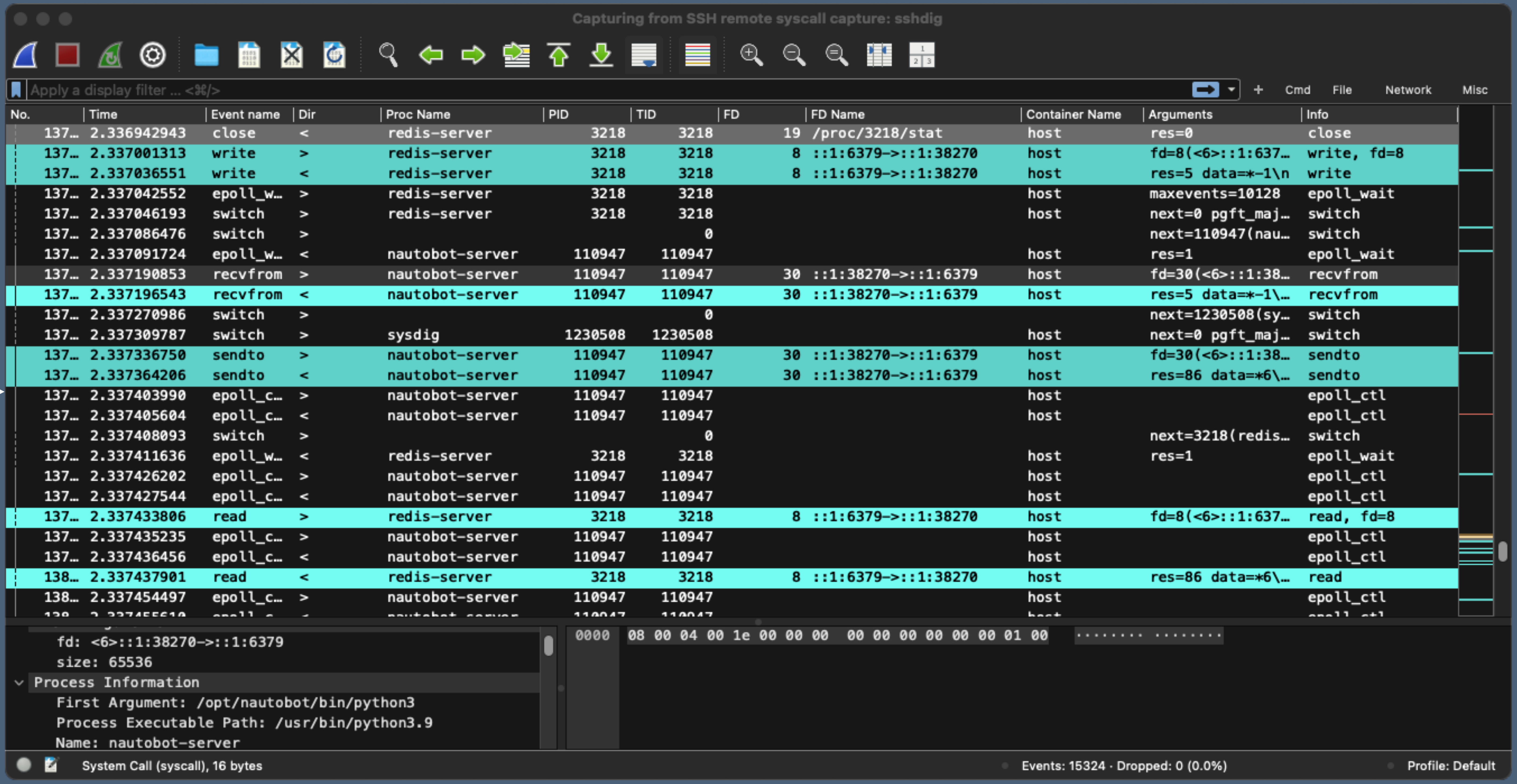
	Interface	Traffic	Link-layer Head	Promiscuous	Snaplen (B)	Buffer (MB)	Monitor Mode
	Falco plugin: cloudtrail		USER0	—	default	—	—
	SSH remote syscall capture: sshdig		Remot...t DLT	—	default	—	—

Stratoshark · Interface Options: SSH remote syscall capture: sshdig

Remote SSH server address 172.28.184.41

Remote SSH server port 22

☑ Save parameter(s) on capture start



Capturing from SSH remote syscall capture: sshdig

No.	Time	Event name	Dir	Proc Name	PID	TID	FD	FD Name	Container Name	Arguments	Info
137…	2.336942943	close	<	redis-server	3218	3218	19	/proc/3218/stat	host	res=0	close
137…	2.337001313	write	>	redis-server	3218	3218	8	::1:6379->::1:38270	host	fd=8(<6>::1:637…	write, fd=8
137…	2.337036551	write	<	redis-server	3218	3218	8	::1:6379->::1:38270	host	res=5 data=*-1\n	write
137…	2.337042552	epoll_w…	>	redis-server	3218	3218			host	maxevents=10128	epoll_wait
137…	2.337046193	switch	>	redis-server	3218	3218			host	next=0 pgft_maj…	switch
137…	2.337086476	switch	>			0				next=110947(nau…	switch
137…	2.337091724	epoll_w…	<	nautobot-server	110947	110947			host	res=1	epoll_wait
137…	2.337190853	recvfrom	>	nautobot-server	110947	110947	30	::1:38270->::1:6379	host	fd=30(<6>::1:38…	recvfrom
137…	2.337196543	recvfrom	<	nautobot-server	110947	110947	30	::1:38270->::1:6379	host	res=5 data=*-1\…	recvfrom
137…	2.337270986	switch	>			0				next=1230508(sy…	switch
137…	2.337309787	switch	>	sysdig	1230508	1230508			host	next=0 pgft_maj…	switch
137…	2.337336750	sendto	>	nautobot-server	110947	110947	30	::1:38270->::1:6379	host	fd=30(<6>::1:38…	sendto
137…	2.337364206	sendto	<	nautobot-server	110947	110947	30	::1:38270->::1:6379	host	res=86 data=*6\…	sendto
137…	2.337403990	epoll_c…	>	nautobot-server	110947	110947			host		epoll_ctl
137…	2.337405604	epoll_c…	<	nautobot-server	110947	110947			host		epoll_ctl
137…	2.337408093	switch	>			0				next=3218(redis…	switch
137…	2.337411636	epoll_w…	<	redis-server	3218	3218			host	res=1	epoll_wait
137…	2.337426202	epoll_c…	>	nautobot-server	110947	110947			host		epoll_ctl
137…	2.337427544	epoll_c…	<	nautobot-server	110947	110947			host		epoll_ctl
137…	2.337433806	read	>	redis-server	3218	3218	8	::1:6379->::1:38270	host	fd=8(<6>::1:637…	read, fd=8
137…	2.337435235	epoll_c…	>	nautobot-server	110947	110947			host		epoll_ctl
137…	2.337436456	epoll_c…	<	nautobot-server	110947	110947			host		epoll_ctl
138…	2.337437901	read	<	redis-server	3218	3218	8	::1:6379->::1:38270	host	res=86 data=*6\…	read
138…	2.337454497	epoll_c…	>	nautobot-server	110947	110947			host		epoll_ctl

```
    fd: <6>::1:38270->::1:6379
    size: 65536
⌄ Process Information
    First Argument: /opt/nautobot/bin/python3
    Process Executable Path: /usr/bin/python3.9
    Name: nautobot-server
```

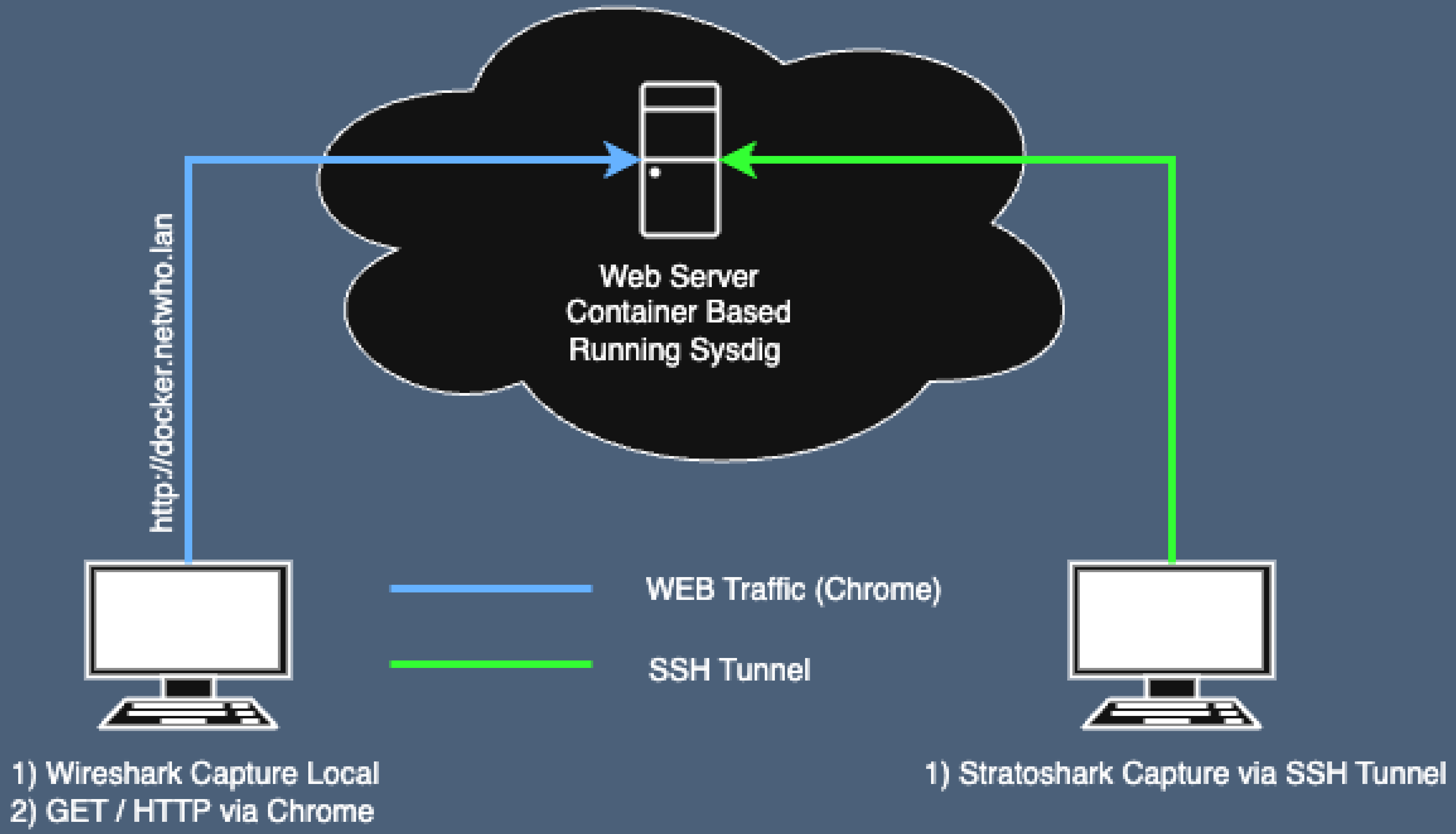
```
0000   08 00 04 00 1e 00 00 00  00 00 00 00 00 00 01 00   ········ ········
```

System Call (syscall), 16 bytes

Events: 15324 · Dropped: 0 (0.0%)

Profile: Default

Demo Setup



Web Server
Container Based
Running Sysdig

http://docker.netwho.lan

WEB Traffic (Chrome)

SSH Tunnel

1) Wireshark Capture Local
2) GET / HTTP via Chrome

1) Stratoshark Capture via SSH Tunnel

DEMO

Stratoshark using Sysdig (DEMO)

THANK YOU!
WIRESHARK