An Experience Report on Extracting and Viewing Memory Events Via Wireshark

Sarah Laing, Michael E. Locasto, John Aycock
University of Calgary

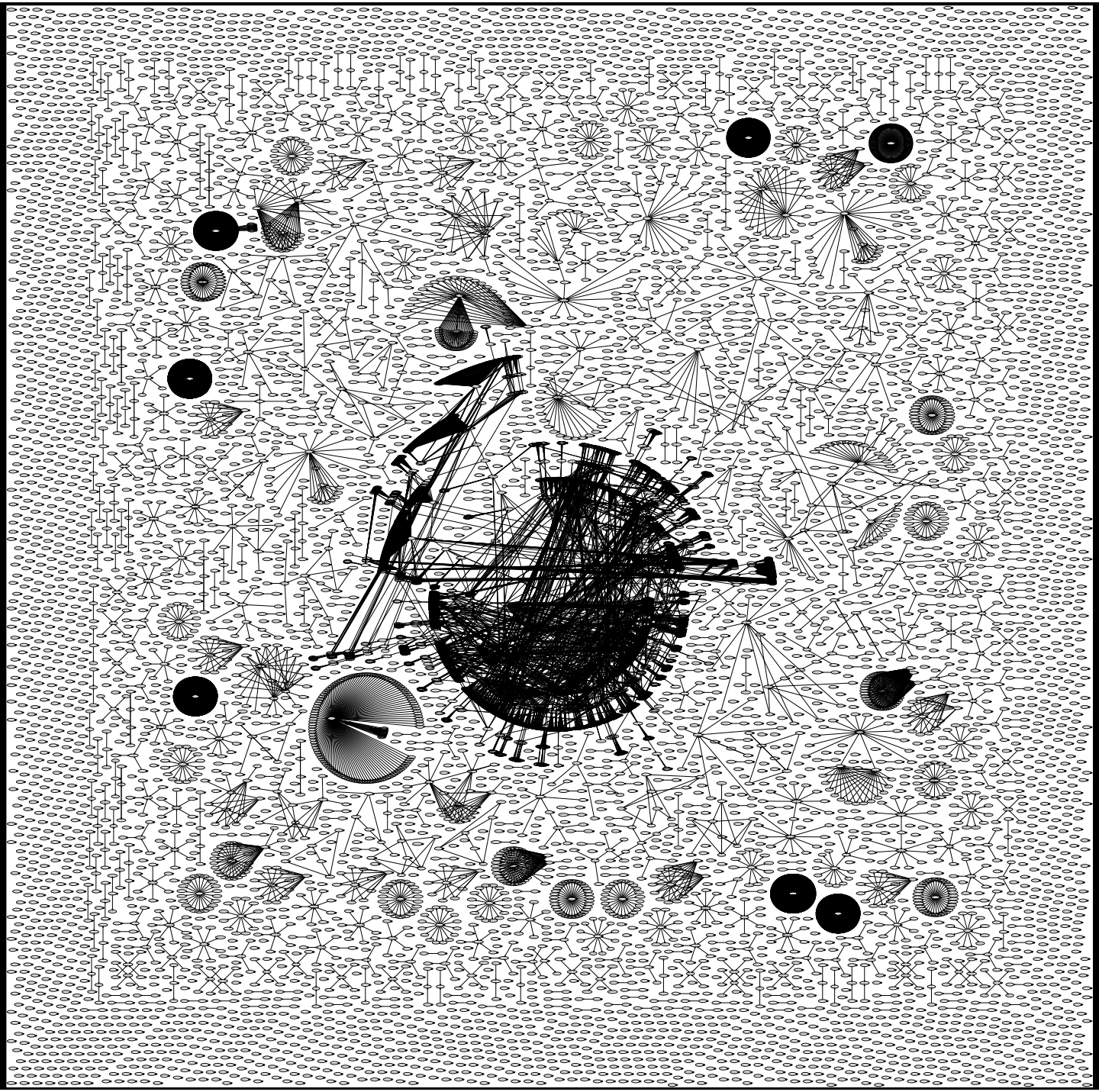
USENIX WOOT 2014

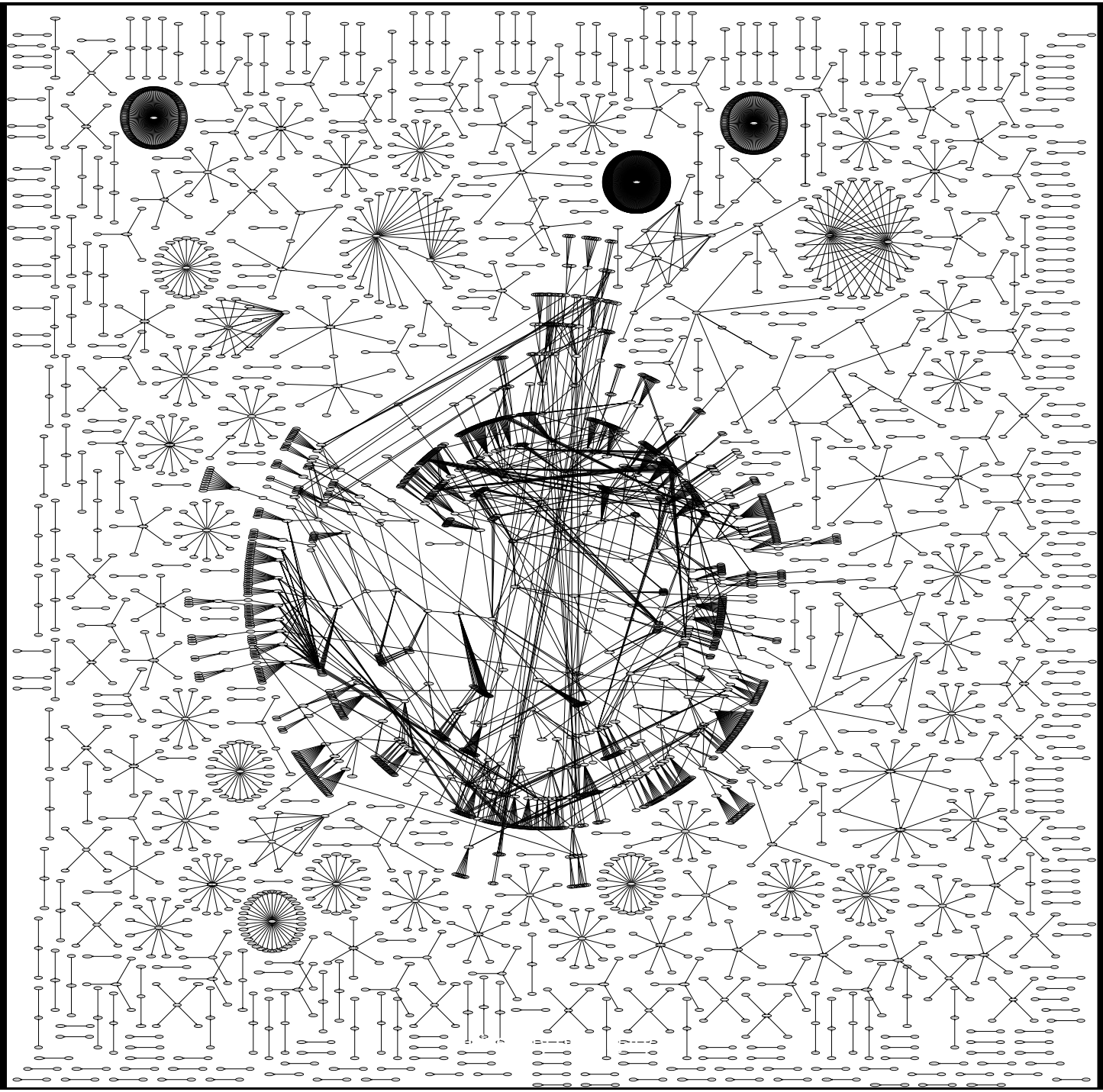
Challenge: Memory Event Analysis

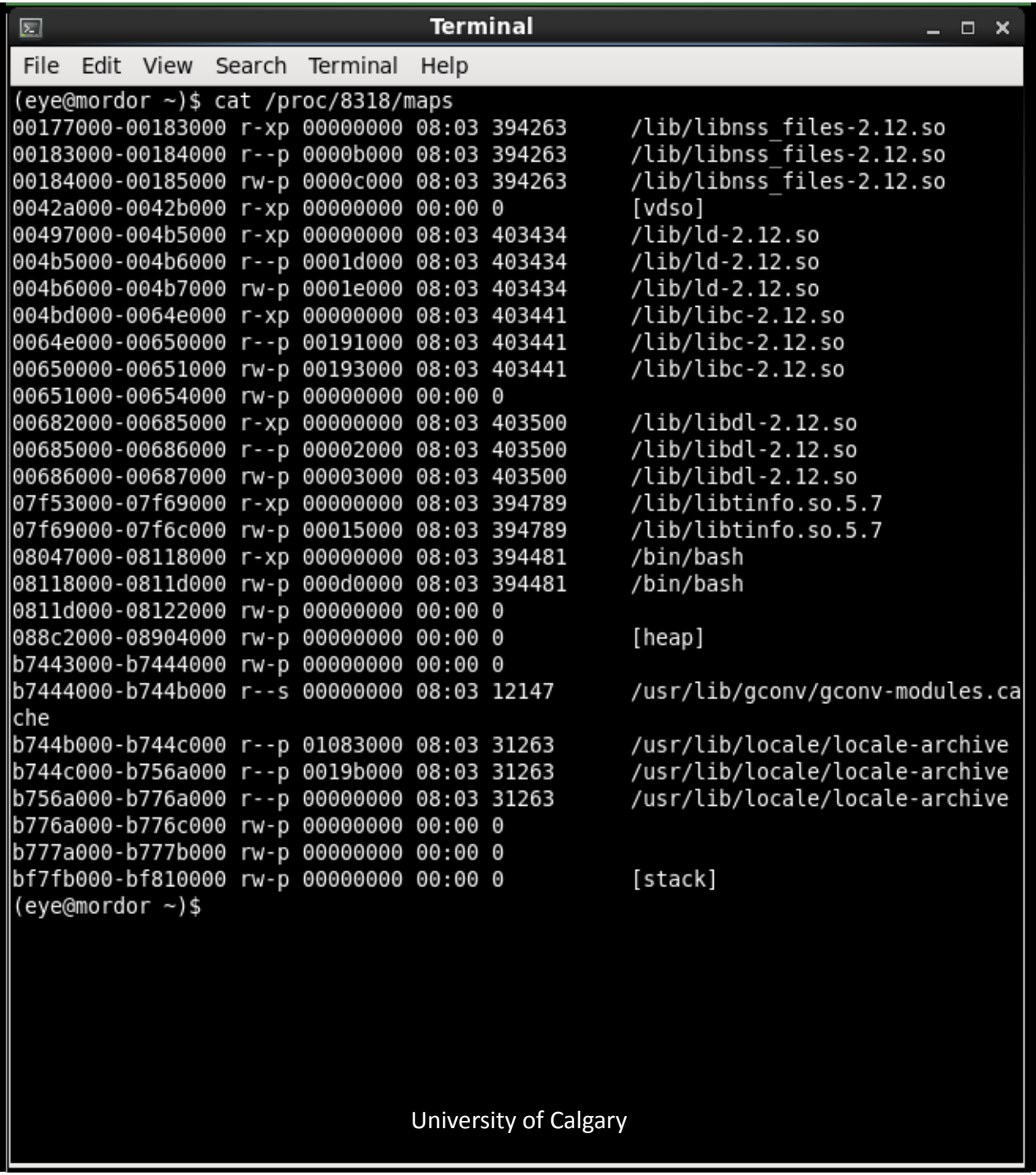
Cage: a kernel-level mechanism for monitoring process memory events and exporting them via a network interface

Neat twist: display in Wireshark

Example Uses: find private key in SSH, overwrite data, overwrite instructions, find all buffers in a program, …






Terminal
File Edit View Search Terminal Help
(eye@mordor ~)$ cat /proc/8318/maps
00177000-00183000 r-xp 00000000 08:03 394263 /lib/libnss_files-2.12.so
00183000-00184000 r--p 0000b000 08:03 394263 /lib/libnss_files-2.12.so
00184000-00185000 rw-p 0000c000 08:03 394263 /lib/libnss_files-2.12.so
0042a000-0042b000 r-xp 00000000 00:00 0 [vdso]
00497000-004b5000 r-xp 00000000 08:03 403434 /lib/ld-2.12.so
004b5000-004b6000 r--p 0001d000 08:03 403434 /lib/ld-2.12.so
004b6000-004b7000 rw-p 0001e000 08:03 403434 /lib/ld-2.12.so
004bd000-0064e000 r-xp 00000000 08:03 403441 /lib/libc-2.12.so
0064e000-00650000 r--p 00191000 08:03 403441 /lib/libc-2.12.so
00650000-00651000 rw-p 00193000 08:03 403441 /lib/libc-2.12.so
00651000-00654000 rw-p 00000000 00:00 0
00682000-00685000 r-xp 00000000 08:03 403500 /lib/libdl-2.12.so
00685000-00686000 r--p 00002000 08:03 403500 /lib/libdl-2.12.so
00686000-00687000 rw-p 00003000 08:03 403500 /lib/libdl-2.12.so
07f53000-07f69000 r-xp 00000000 08:03 394789 /lib/libtinfo.so.5.7
07f69000-07f6c000 rw-p 00015000 08:03 394789 /lib/libtinfo.so.5.7
08047000-08118000 r-xp 00000000 08:03 394481 /bin/bash
08118000-0811d000 rw-p 000d0000 08:03 394481 /bin/bash
0811d000-08122000 rw-p 00000000 00:00 0
088c2000-08904000 rw-p 00000000 00:00 0 [heap]
b7443000-b7444000 rw-p 00000000 00:00 0
b7444000-b744b000 r--s 00000000 08:03 12147 /usr/lib/gconv/gconv-modules.cache
b744b000-b744c000 r--p 01083000 08:03 31263 /usr/lib/locale/locale-archive
b744c000-b756a000 r--p 0019b000 08:03 31263 /usr/lib/locale/locale-archive
b756a000-b776a000 r--p 00000000 08:03 31263 /usr/lib/locale/locale-archive
b776a000-b776c000 rw-p 00000000 00:00 0
b777a000-b777b000 rw-p 00000000 00:00 0
bf7fb000-bf810000 rw-p 00000000 00:00 0 [stack]
(eye@mordor ~)$

Challenge: Memory Event Analysis

Cage: a kernel-level mechanism for monitoring (+ modifying) process memory events and exporting them via a network interface

Neat twist: display in Wireshark

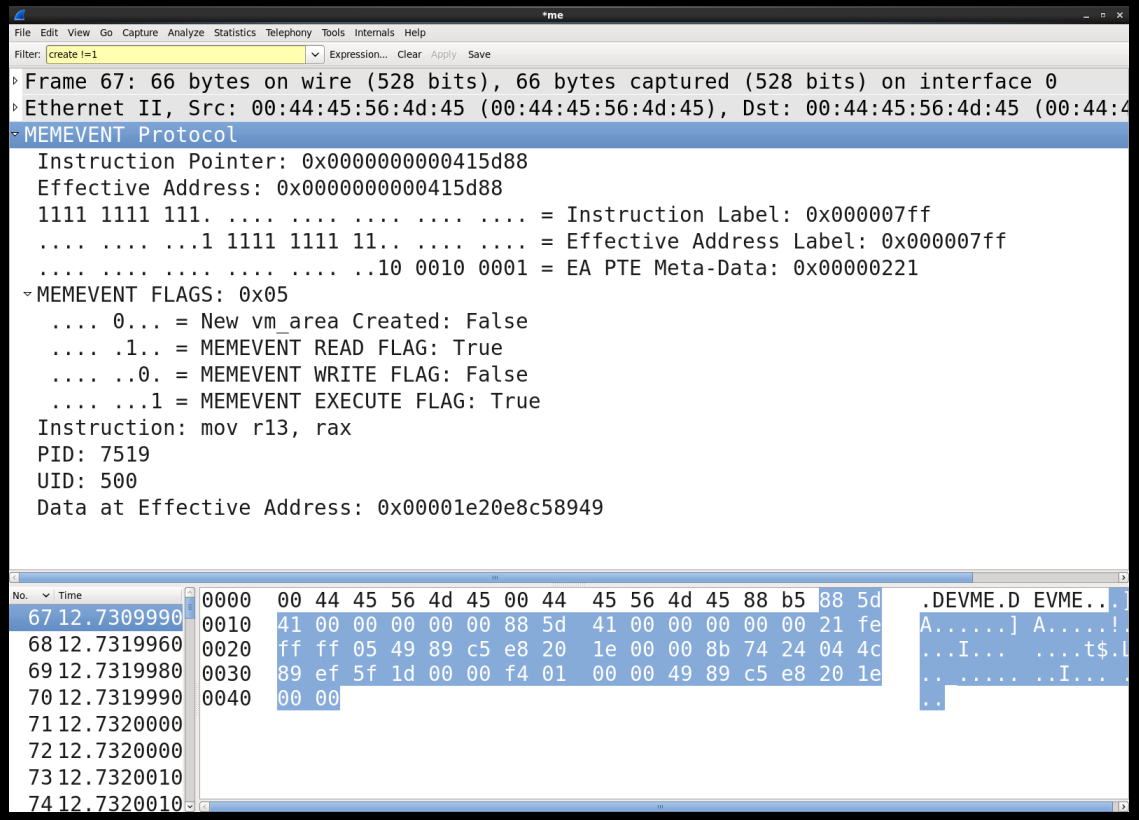
Example Uses: find private key in SSH, overwrite data, overwrite instructions, find all buffers in a program, …

Underlying Insight / Why Wireshark?

BPF: pre-filtering

Wireshark: post-filtering

Treat a stream of memory events like a packet trace, and then benefit from the types of filtering languages that exist for such streams.



Building a memory trapping system seems conceptually easy, but is non-trivial and difficult if you want to do seamlessly; see Figures 5 and 6 in the paper for validation across multiple architectures, OS distros, and VMs

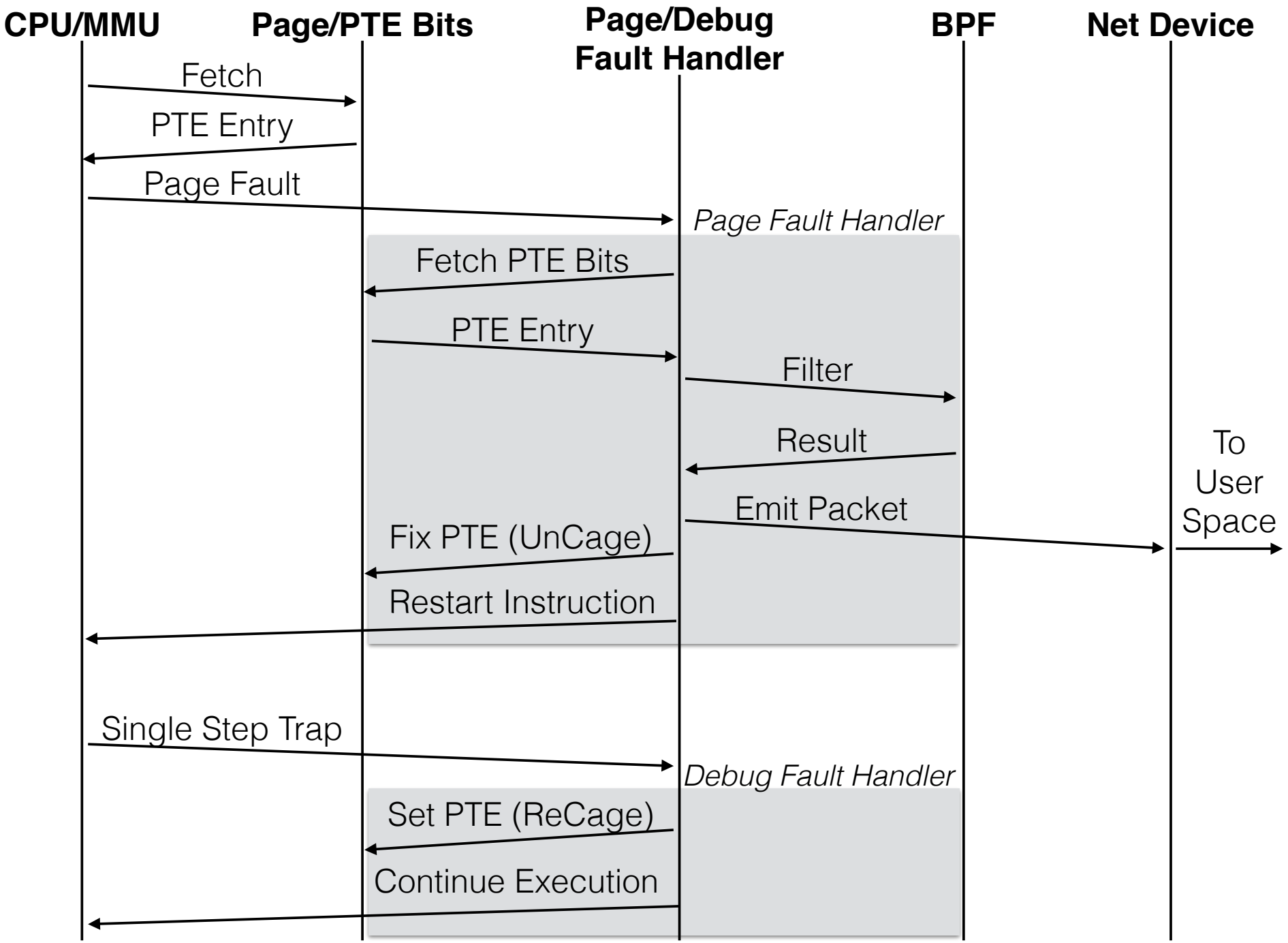
Only raises our level of respect for prior work on memory interception techniques

Cage Implementation

Modifications to the Linux 3.9.4 kernel

```
23 files changed, 1002 insertions(+), 23
  deletions(-)

  create mode 100644 linux-3.9.4/arch/x86/mm/cage.c
  create mode 100644 linux-3.9.4/chmem/Makefile
  create mode 100644 linux-3.9.4/chmem/chmem.c
  create mode 100644 linux-3.9.4/include/linux/cage.h
```


CPU/MMU
Page/PTE Bits
Page/Debug Fault Handler
BPF
Net Device
Fetch
PTE Entry
Page Fault
Page Fault Handler
Fetch PTE Bits
PTE Entry
Filter
Result
Emit Packet
To User Space
Fix PTE (UnCage)
Restart Instruction
Single Step Trap
Debug Fault Handler
Set PTE (ReCage)
Continue Execution

BPF Filters (In-kernel, pre-event)

Temporal Filter: Watch for a specific address range. Emit a packet every n-th event. Rate limiting.

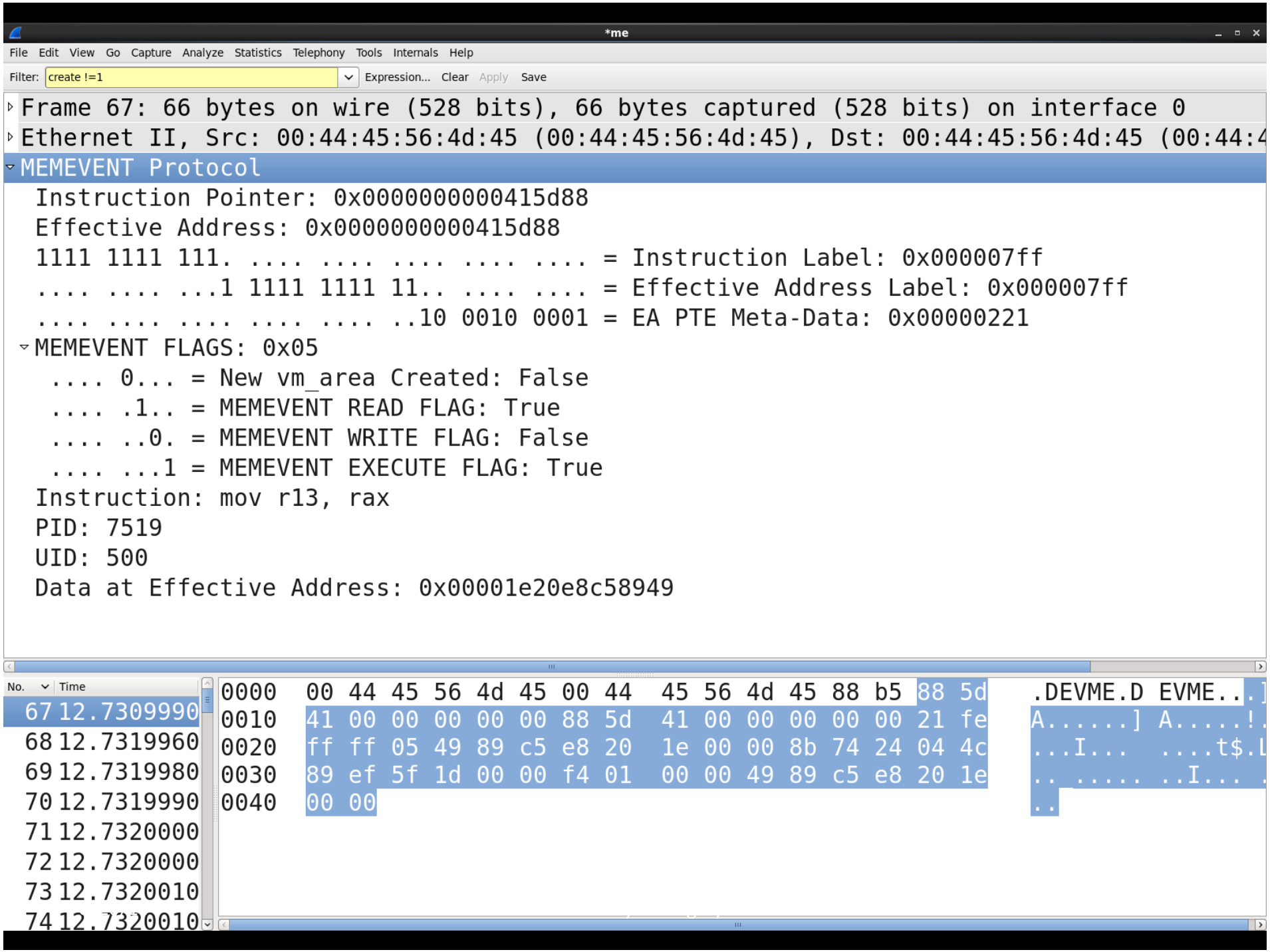
Data Overwriting Filter: Watch for a specific address and replace the data at that address with a user specified value.

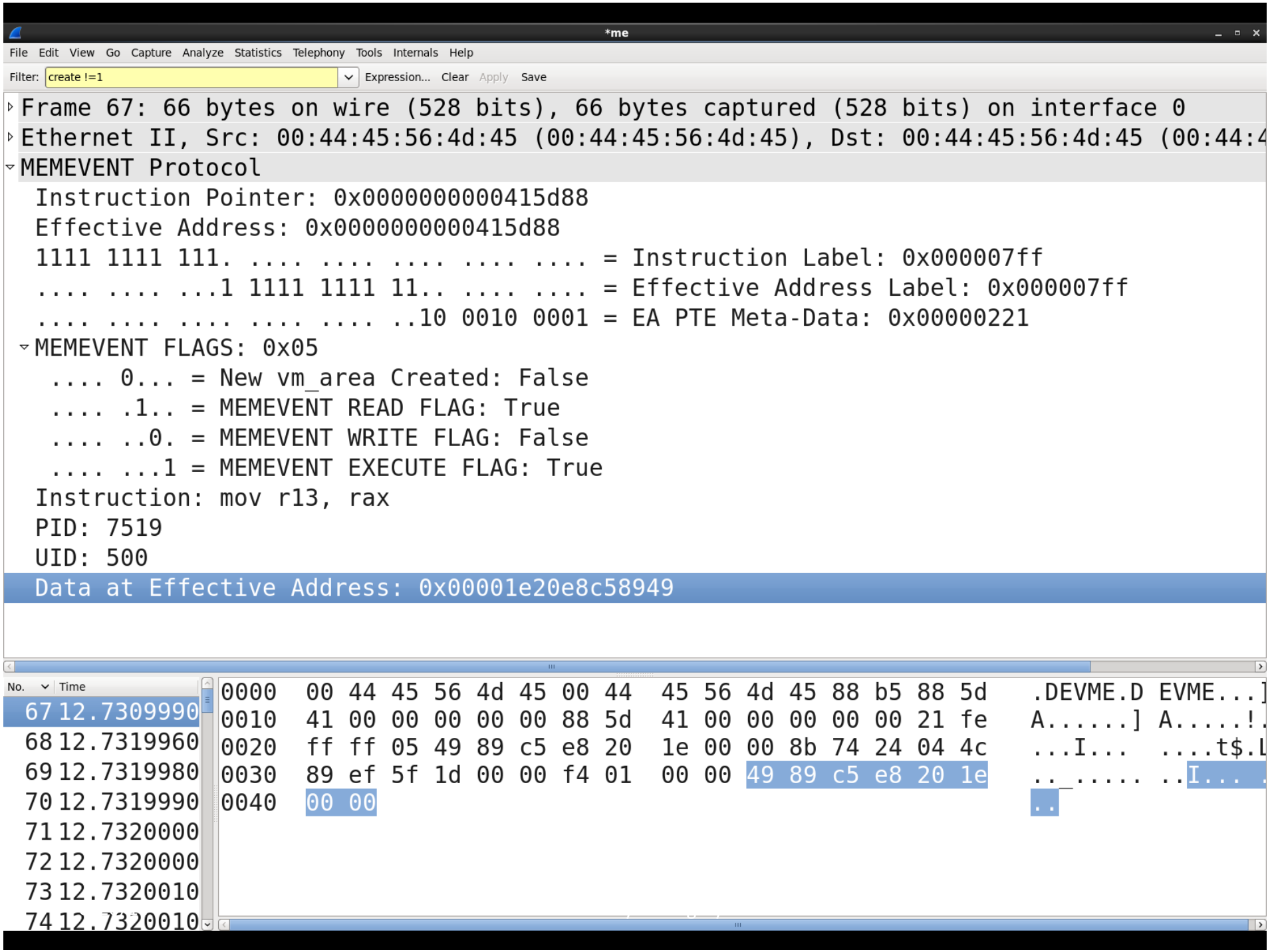
Instruction Overwriting Filter: Watch for a specific RIP/EIP and replace the instruction at that address with a user specified instruction.

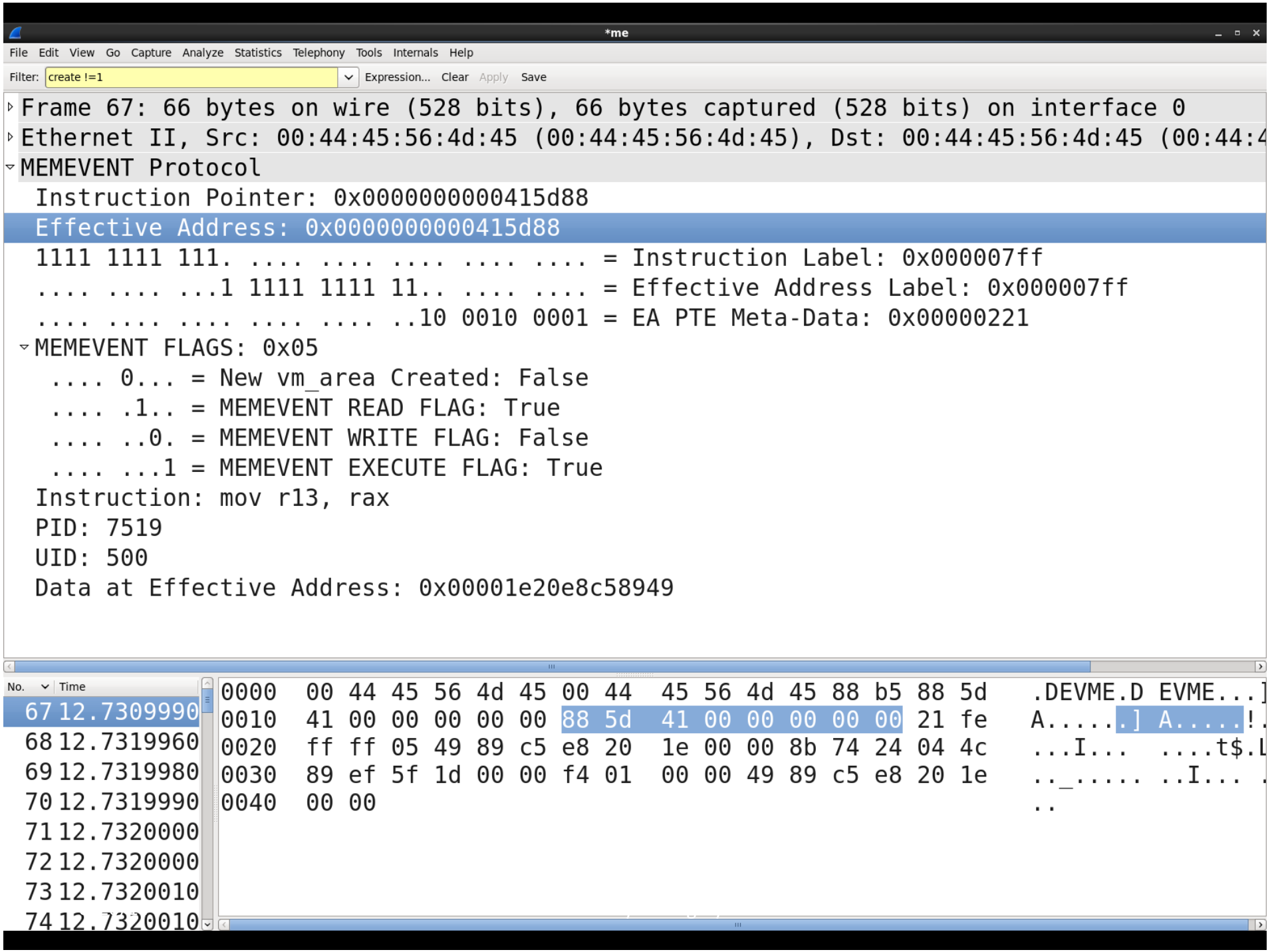
Buffer Viewing Filter: Watch for the creation of a specific buffer and emit all packets that touch that buffer. (SSH example)

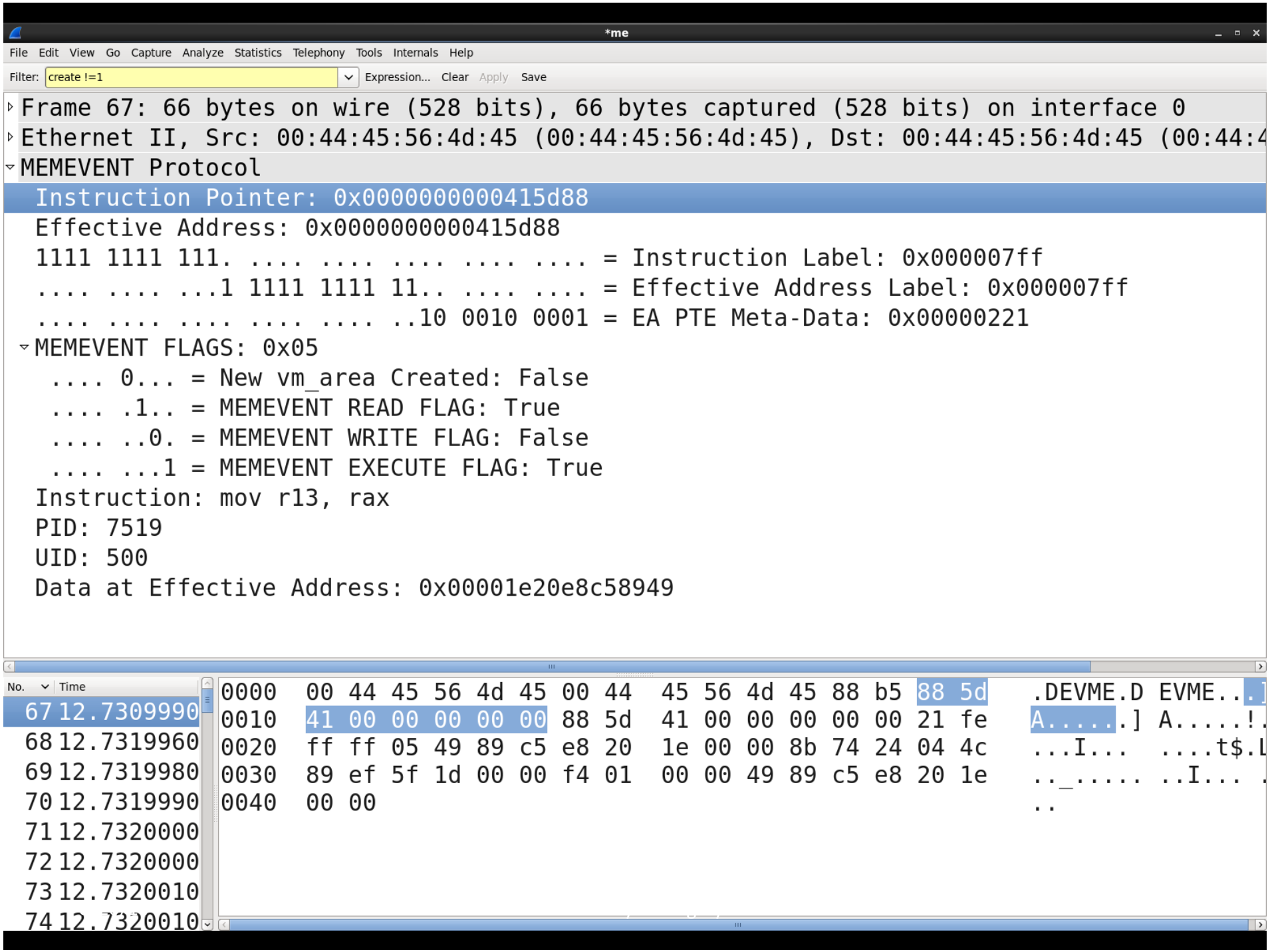
Buffer Finding Filter: Find all buffers in a program by watching for sequential accesses to memory locations.

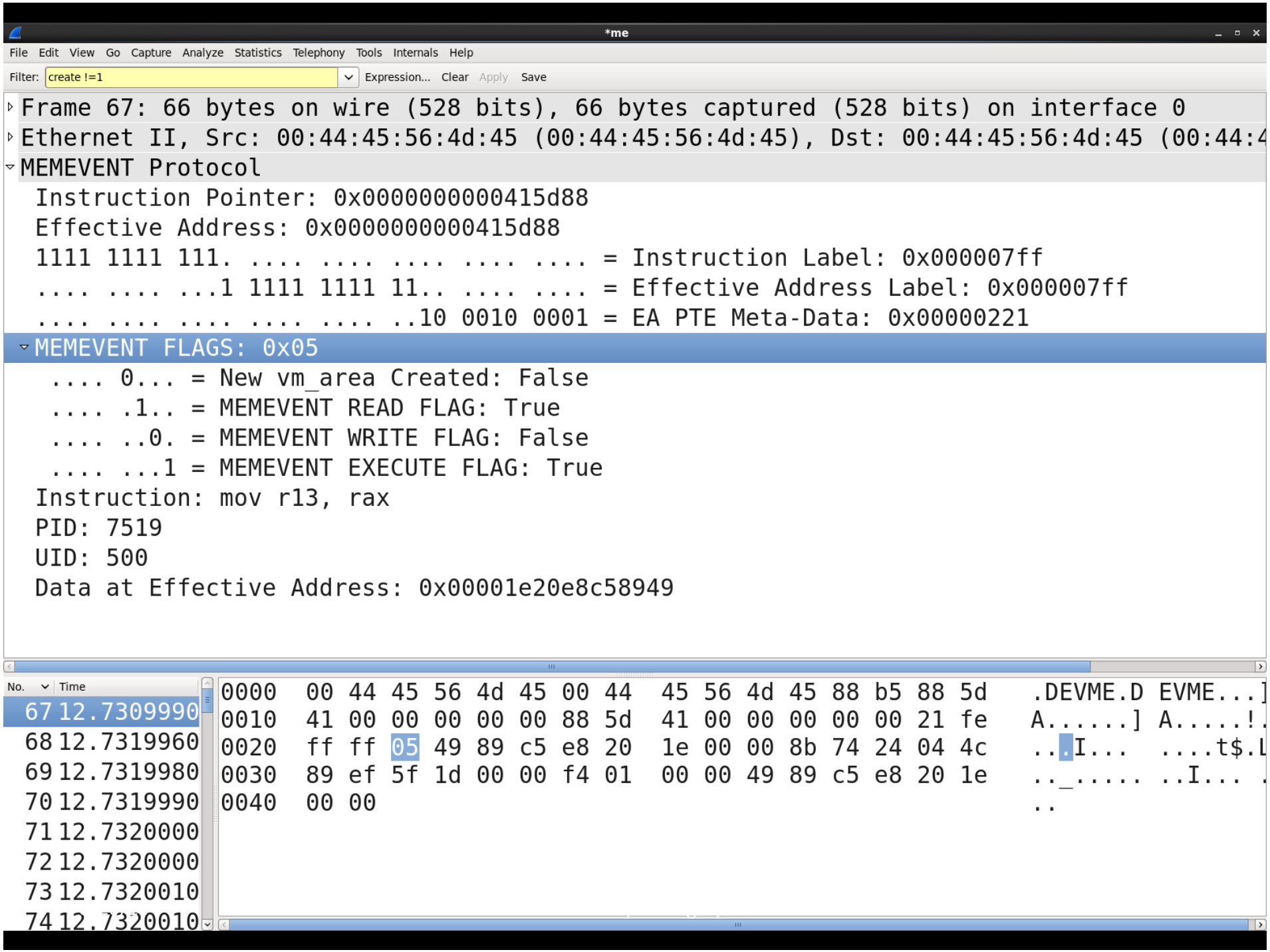
FETCH/EXECUTE

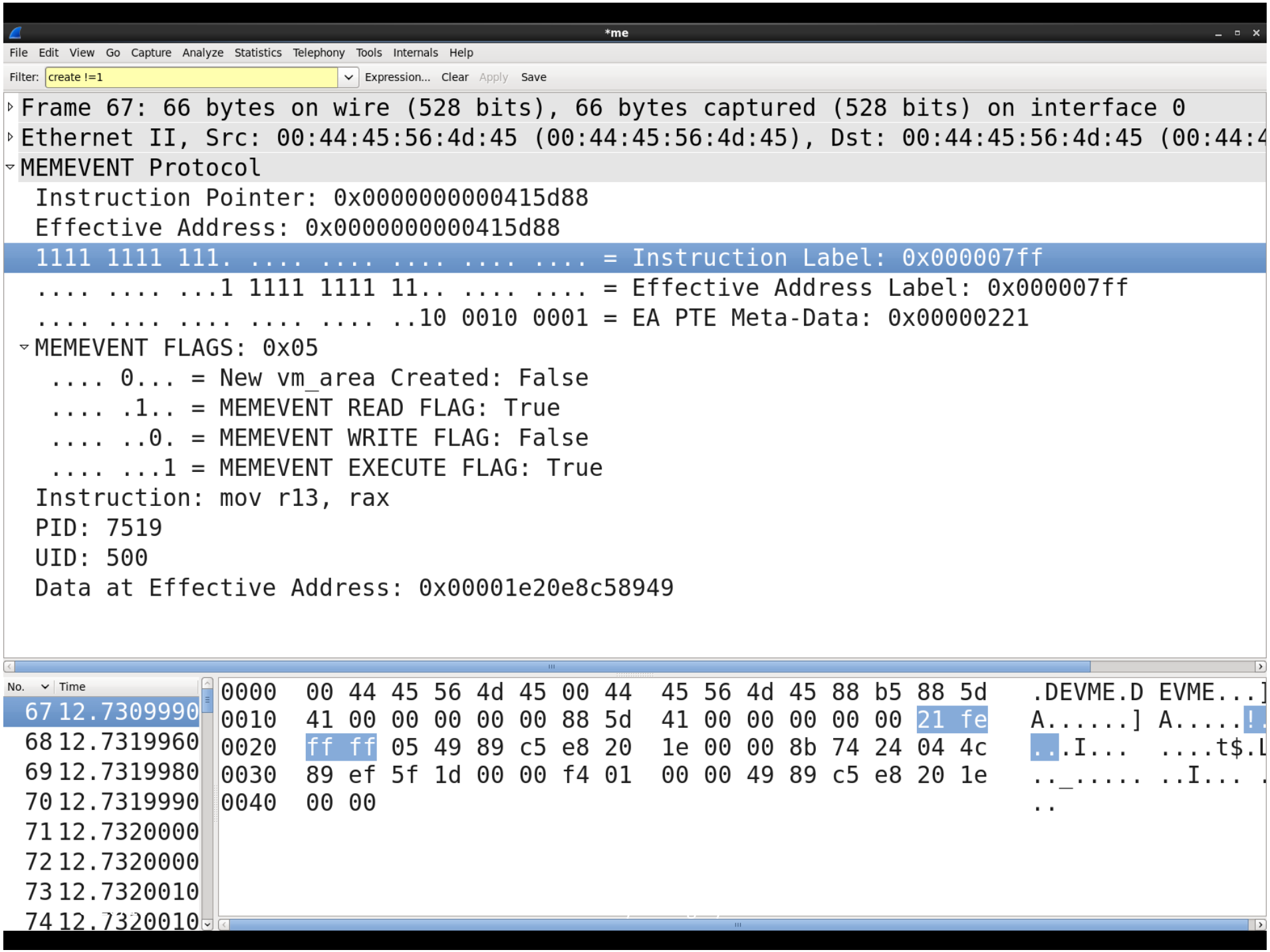

*me
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Filter: create !=1
Expression... Clear Apply Save
Frame 67: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface 0
Ethernet II, Src: 00:44:45:56:4d:45 (00:44:45:56:4d:45), Dst: 00:44:45:56:4d:45 (00:44:4
MEMEVENT Protocol
Instruction Pointer: 0x0000000000415d88
Effective Address: 0x0000000000415d88
1111 1111 111. .... = Instruction Label: 0x000007ff
....1 1111 1111 11.. .... = Effective Address Label: 0x000007ff
....10 0010 0001 = EA PTE Meta-Data: 0x00000221
MEMEVENT FLAGS: 0x05
.... 0... = New vm_area Created: False
.... .1.. = MEMEVENT READ FLAG: True
.... ..0. = MEMEVENT WRITE FLAG: False
....1 = MEMEVENT EXECUTE FLAG: True
Instruction: mov r13, rax
PID: 7519
UID: 500
Data at Effective Address: 0x00001e20e8c58949
No. Time
67 12.7309990
68 12.7319960
69 12.7319980
70 12.7319990
71 12.7320000
72 12.7320000
73 12.7320010
74 12.7320010
0000 00 44 45 56 4d 45 00 44 45 56 4d 45 88 b5 88 5d .DEVME.D EVME...]
0010 41 00 00 00 00 00 88 5d 41 00 00 00 00 00 21 fe A.....] A.....!.
0020 ff ff 05 49 89 c5 e8 20 1e 00 00 8b 74 24 04 4c ...I... ....t$.L
0030 89 ef 5f 1d 00 00 f4 01 00 00 49 89 c5 e8 20 1e .. ..I... .
0040 00 00 ..

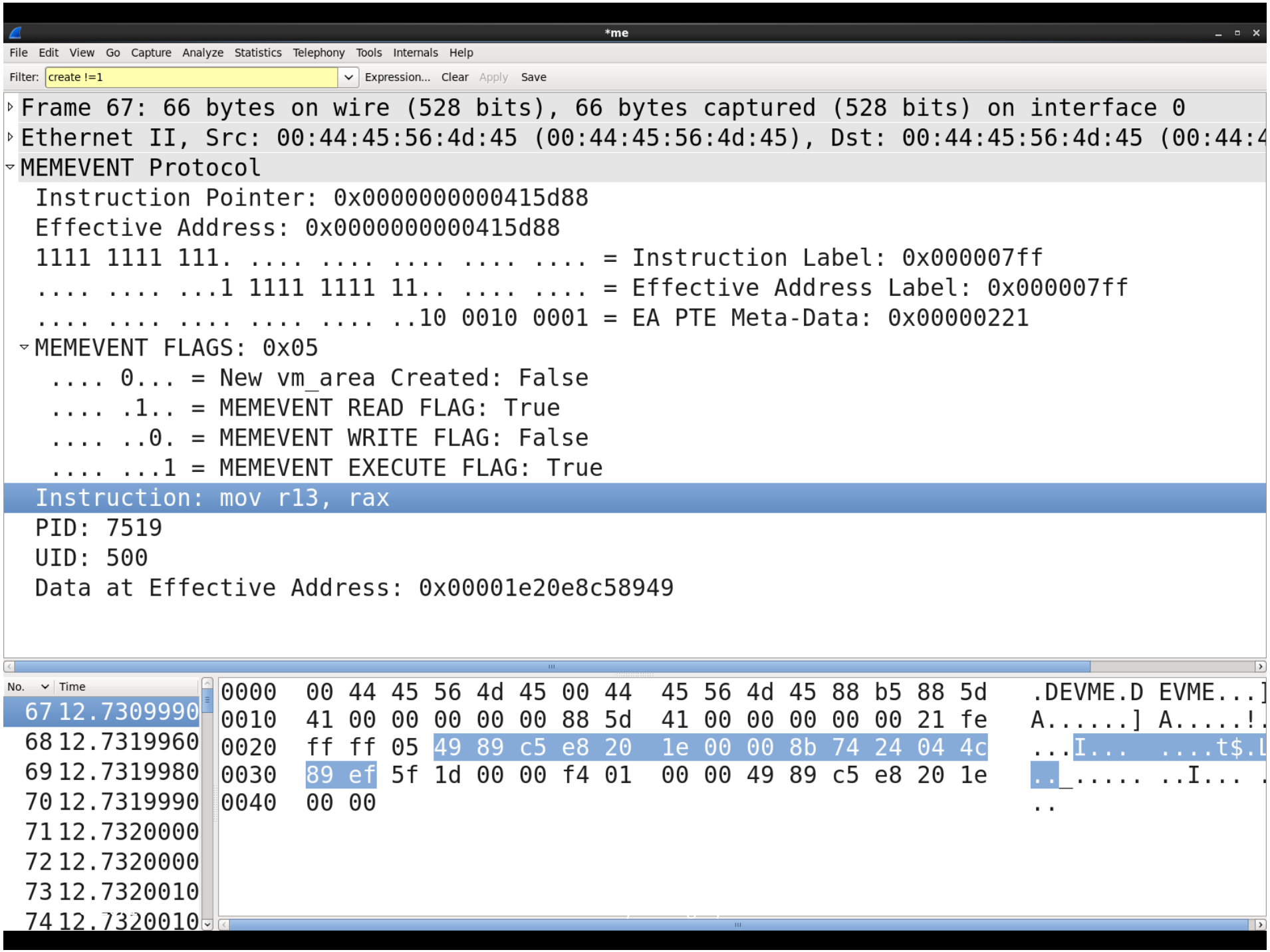

*me
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Filter: create !=1
Expression... Clear Apply Save
Frame 67: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface 0
Ethernet II, Src: 00:44:45:56:4d:45 (00:44:45:56:4d:45), Dst: 00:44:45:56:4d:45 (00:44:4
MEMEVENT Protocol
Instruction Pointer: 0x0000000000415d88
Effective Address: 0x0000000000415d88
1111 1111 111. .... = Instruction Label: 0x000007ff
....1 1111 1111 11.. .... = Effective Address Label: 0x000007ff
....10 0010 0001 = EA PTE Meta-Data: 0x00000221
MEMEVENT FLAGS: 0x05
.... 0... = New vm_area Created: False
.... .1.. = MEMEVENT READ FLAG: True
.... ..0. = MEMEVENT WRITE FLAG: False
....1 = MEMEVENT EXECUTE FLAG: True
Instruction: mov r13, rax
PID: 7519
UID: 500
Data at Effective Address: 0x00001e20e8c58949
No. Time
67 12.7309990
68 12.7319960
69 12.7319980
70 12.7319990
71 12.7320000
72 12.7320000
73 12.7320010
74 12.7320010
0000 00 44 45 56 4d 45 00 44 45 56 4d 45 88 b5 88 5d .DEVME.D EVME...]
0010 41 00 00 00 00 00 88 5d 41 00 00 00 00 00 21 fe A.....] A.....!.
0020 ff ff 05 49 89 c5 e8 20 1e 00 00 8b 74 24 04 4c ...I... ....t$.L
0030 89 ef 5f 1d 00 00 f4 01 00 00 49 89 c5 e8 20 1e .._..... ..I... .
0040 00 00 ..


*me
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Filter: create !=1
Expression... Clear Apply Save
Frame 67: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface 0
Ethernet II, Src: 00:44:45:56:4d:45 (00:44:45:56:4d:45), Dst: 00:44:45:56:4d:45 (00:44:4
MEMEVENT Protocol
Instruction Pointer: 0x0000000000415d88
Effective Address: 0x0000000000415d88
1111 1111 111. .... = Instruction Label: 0x000007ff
....1 1111 1111 11.. .... = Effective Address Label: 0x000007ff
....10 0010 0001 = EA PTE Meta-Data: 0x00000221
MEMEVENT FLAGS: 0x05
.... 0... = New vm_area Created: False
.... .1.. = MEMEVENT READ FLAG: True
.... ..0. = MEMEVENT WRITE FLAG: False
....1 = MEMEVENT EXECUTE FLAG: True
Instruction: mov r13, rax
PID: 7519
UID: 500
Data at Effective Address: 0x00001e20e8c58949
No. Time
67 12.7309990
68 12.7319960
69 12.7319980
70 12.7319990
71 12.7320000
72 12.7320000
73 12.7320010
74 12.7320010
0000 00 44 45 56 4d 45 00 44 45 56 4d 45 88 b5 88 5d .DEVME.D EVME...]
0010 41 00 00 00 00 00 88 5d 41 00 00 00 00 00 21 fe A.....] A.....!.
0020 ff ff 05 49 89 c5 e8 20 1e 00 00 8b 74 24 04 4c ...I... ....t$.L
0030 89 ef 5f 1d 00 00 f4 01 00 00 49 89 c5 e8 20 1e ..._..... ..I... .
0040 00 00 ..

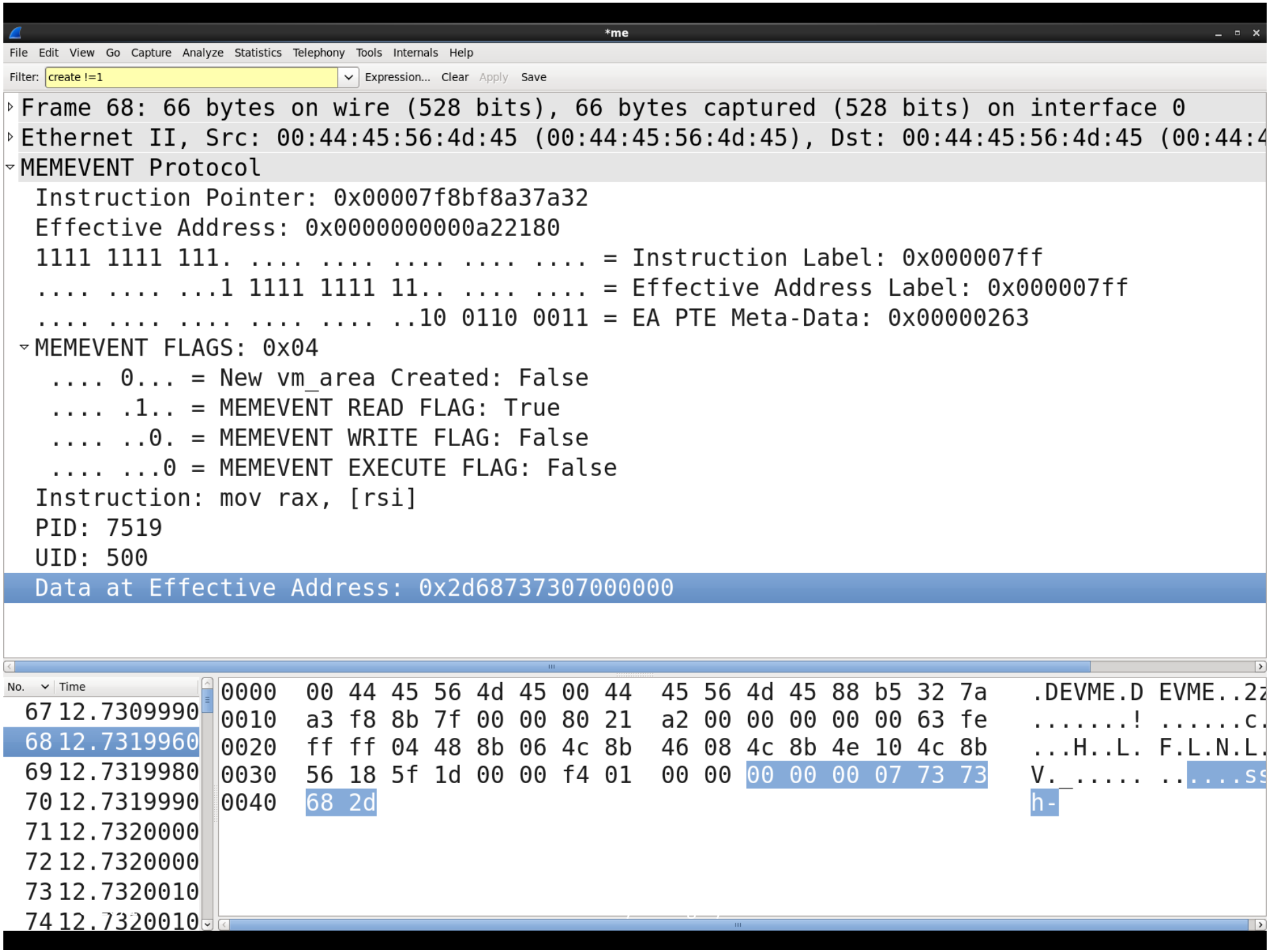

*me
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Filter: create !=1
Expression... Clear Apply Save
Frame 67: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface 0
Ethernet II, Src: 00:44:45:56:4d:45 (00:44:45:56:4d:45), Dst: 00:44:45:56:4d:45 (00:44:4
MEMEVENT Protocol
Instruction Pointer: 0x0000000000415d88
Effective Address: 0x0000000000415d88
1111 1111 111. .... = Instruction Label: 0x000007ff
....1 1111 1111 11.. .... = Effective Address Label: 0x000007ff
....10 0010 0001 = EA PTE Meta-Data: 0x00000221
MEMEVENT FLAGS: 0x05
.... 0... = New vm_area Created: False
.... .1.. = MEMEVENT READ FLAG: True
.... ..0. = MEMEVENT WRITE FLAG: False
....1 = MEMEVENT EXECUTE FLAG: True
Instruction: mov r13, rax
PID: 7519
UID: 500
Data at Effective Address: 0x00001e20e8c58949
No. Time
67 12.7309990
68 12.7319960
69 12.7319980
70 12.7319990
71 12.7320000
72 12.7320000
73 12.7320010
74 12.7320010
0000 00 44 45 56 4d 45 00 44 45 56 4d 45 88 b5 88 5d .DEVME.D EVME..]
0010 41 00 00 00 00 00 88 5d 41 00 00 00 00 00 21 fe A.....] A.....!.
0020 ff ff 05 49 89 c5 e8 20 1e 00 00 8b 74 24 04 4c ...I... ....t$.L
0030 89 ef 5f 1d 00 00 f4 01 00 00 49 89 c5 e8 20 1e ..._..... ..I... .
0040 00 00 ..

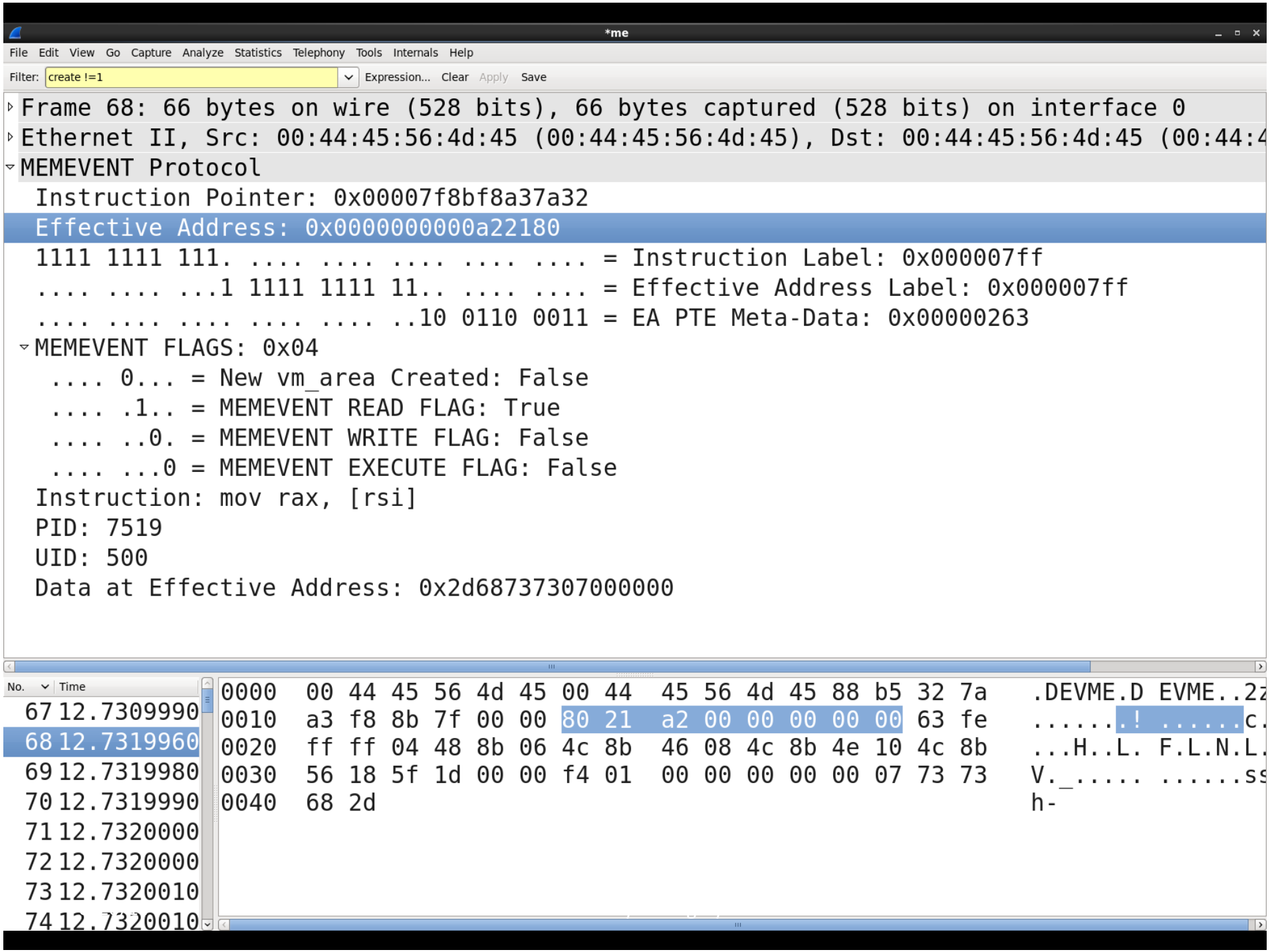

*me
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Filter: create !=1
Expression... Clear Apply Save
Frame 67: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface 0
Ethernet II, Src: 00:44:45:56:4d:45 (00:44:45:56:4d:45), Dst: 00:44:45:56:4d:45 (00:44:4
MEMEVENT Protocol
Instruction Pointer: 0x0000000000415d88
Effective Address: 0x0000000000415d88
1111 1111 111. .... = Instruction Label: 0x000007ff
....1 1111 1111 11.. .... = Effective Address Label: 0x000007ff
....10 0010 0001 = EA PTE Meta-Data: 0x00000221
MEMEVENT FLAGS: 0x05
.... 0... = New vm_area Created: False
.... .1.. = MEMEVENT READ FLAG: True
.... ..0. = MEMEVENT WRITE FLAG: False
....1 = MEMEVENT EXECUTE FLAG: True
Instruction: mov r13, rax
PID: 7519
UID: 500
Data at Effective Address: 0x00001e20e8c58949
No. Time
67 12.7309990
68 12.7319960
69 12.7319980
70 12.7319990
71 12.7320000
72 12.7320000
73 12.7320010
74 12.7320010
0000 00 44 45 56 4d 45 00 44 45 56 4d 45 88 b5 88 5d .DEVME.D EVME...]
0010 41 00 00 00 00 00 88 5d 41 00 00 00 00 00 21 fe A.....] A.....!.
0020 ff ff 05 49 89 c5 e8 20 1e 00 00 8b 74 24 04 4c ...I... ....t$.L
0030 89 ef 5f 1d 00 00 f4 01 00 00 49 89 c5 e8 20 1e .._..... ..I... .
0040 00 00 ..


*me
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Filter: create !=1
Expression... Clear Apply Save
Frame 67: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface 0
Ethernet II, Src: 00:44:45:56:4d:45 (00:44:45:56:4d:45), Dst: 00:44:45:56:4d:45 (00:44:4
MEMEVENT Protocol
Instruction Pointer: 0x0000000000415d88
Effective Address: 0x0000000000415d88
1111 1111 111. .... = Instruction Label: 0x000007ff
....1 1111 1111 11.. .... = Effective Address Label: 0x000007ff
....10 0010 0001 = EA PTE Meta-Data: 0x00000221
MEMEVENT FLAGS: 0x05
.... 0... = New vm_area Created: False
.... .1.. = MEMEVENT READ FLAG: True
.... ..0. = MEMEVENT WRITE FLAG: False
....1 = MEMEVENT EXECUTE FLAG: True
Instruction: mov r13, rax
PID: 7519
UID: 500
Data at Effective Address: 0x00001e20e8c58949
No. Time
67 12.7309990
68 12.7319960
69 12.7319980
70 12.7319990
71 12.7320000
72 12.7320000
73 12.7320010
74 12.7320010
0000 00 44 45 56 4d 45 00 44 45 56 4d 45 88 b5 88 5d .DEVME.D EVME...]
0010 41 00 00 00 00 00 88 5d 41 00 00 00 00 00 21 fe A.....] A.....!.
0020 ff ff 05 49 89 c5 e8 20 1e 00 00 8b 74 24 04 4c ...I... ....t$.L
0030 89 ef 5f 1d 00 00 f4 01 00 00 49 89 c5 e8 20 1e .._..... ..I... .
0040 00 00 ..


*me
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Filter: create !=1
Expression... Clear Apply Save
Frame 67: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface 0
Ethernet II, Src: 00:44:45:56:4d:45 (00:44:45:56:4d:45), Dst: 00:44:45:56:4d:45 (00:44:4
MEMEVENT Protocol
Instruction Pointer: 0x0000000000415d88
Effective Address: 0x0000000000415d88
1111 1111 111. .... = Instruction Label: 0x000007ff
....1 1111 1111 11.. .... = Effective Address Label: 0x000007ff
....10 0010 0001 = EA PTE Meta-Data: 0x00000221
MEMEVENT FLAGS: 0x05
.... 0... = New vm_area Created: False
.... .1.. = MEMEVENT READ FLAG: True
.... ..0. = MEMEVENT WRITE FLAG: False
....1 = MEMEVENT EXECUTE FLAG: True
Instruction: mov r13, rax
PID: 7519
UID: 500
Data at Effective Address: 0x00001e20e8c58949
No. Time
67 12.7309990
68 12.7319960
69 12.7319980
70 12.7319990
71 12.7320000
72 12.7320000
73 12.7320010
74 12.7320010
0000 00 44 45 56 4d 45 00 44 45 56 4d 45 88 b5 88 5d .DEVME.D EVME...]
0010 41 00 00 00 00 00 88 5d 41 00 00 00 00 00 21 fe A.....] A.....!.
0020 ff ff 05 49 89 c5 e8 20 1e 00 00 8b 74 24 04 4c ...I... ....t$.L
0030 89 ef 5f 1d 00 00 f4 01 00 00 49 89 c5 e8 20 1e .._..... ..I... .
0040 00 00 ..

EXECUTE/READ


*me
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Filter: create !=1
Expression... Clear Apply Save
Frame 68: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface 0
Ethernet II, Src: 00:44:45:56:4d:45 (00:44:45:56:4d:45), Dst: 00:44:45:56:4d:45 (00:44:4
MEMEVENT Protocol
Instruction Pointer: 0x00007f8bf8a37a32
Effective Address: 0x0000000000a22180
1111 1111 111. .... = Instruction Label: 0x000007ff
....1 1111 1111 11.. .... = Effective Address Label: 0x000007ff
....10 0110 0011 = EA PTE Meta-Data: 0x00000263
MEMEVENT FLAGS: 0x04
.... 0... = New vm_area Created: False
.... .1.. = MEMEVENT READ FLAG: True
.... ..0. = MEMEVENT WRITE FLAG: False
....0 = MEMEVENT EXECUTE FLAG: False
Instruction: mov rax, [rsi]
PID: 7519
UID: 500
Data at Effective Address: 0x2d68737307000000
No. Time
67 12.7309990
68 12.7319960
69 12.7319980
70 12.7319990
71 12.7320000
72 12.7320000
73 12.7320010
74 12.7320010
0000 00 44 45 56 4d 45 00 44 45 56 4d 45 88 b5 32 7a .DEVME.D EVME..2z
0010 a3 f8 8b 7f 00 00 80 21 a2 00 00 00 00 00 63 fe ....! ......c.
0020 ff ff 04 48 8b 06 4c 8b 46 08 4c 8b 4e 10 4c 8b ...H..L. F.L.N.L.
0030 56 18 5f 1d 00 00 f4 01 00 00 00 00 00 07 73 73 V._..... ......ss
0040 68 2d h-


*me
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Filter: create !=1
Expression... Clear Apply Save
Frame 68: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface 0
Ethernet II, Src: 00:44:45:56:4d:45 (00:44:45:56:4d:45), Dst: 00:44:45:56:4d:45 (00:44:4
MEMEVENT Protocol
Instruction Pointer: 0x00007f8bf8a37a32
Effective Address: 0x0000000000a22180
1111 1111 111. .... = Instruction Label: 0x000007ff
....1 1111 1111 11.. .... = Effective Address Label: 0x000007ff
....10 0110 0011 = EA PTE Meta-Data: 0x00000263
MEMEVENT FLAGS: 0x04
.... 0... = New vm_area Created: False
.... .1.. = MEMEVENT READ FLAG: True
.... ..0. = MEMEVENT WRITE FLAG: False
....0 = MEMEVENT EXECUTE FLAG: False
Instruction: mov rax, [rsi]
PID: 7519
UID: 500
Data at Effective Address: 0x2d68737307000000
No. Time
67 12.7309990
68 12.7319960
69 12.7319980
70 12.7319990
71 12.7320000
72 12.7320000
73 12.7320010
74 12.7320010
0000 00 44 45 56 4d 45 00 44 45 56 4d 45 88 b5 32 7a .DEVME.D EVME..2z
0010 a3 f8 8b 7f 00 00 80 21 a2 00 00 00 00 00 63 fe ......! ......c.
0020 ff ff 04 48 8b 06 4c 8b 46 08 4c 8b 4e 10 4c 8b ...H..L. F.L.N.L.
0030 56 18 5f 1d 00 00 f4 01 00 00 00 00 00 07 73 73 V._..... ......ss
0040 68 2d h-

Memory Analysis Based on a Simple Language

Friend asks me: “What do I go ‘WOOT’ about?”

Answer: viewing non-network stuff in Wireshark is kind of cool, but our efforts to design, build, and use Cage are *“…part of a broader argument [6] that “offensive” does not mean unprincipled or ad hoc.”*

Takeaway Message

"…it is precisely the variety of creative abuses of existing memory management circuitry that argue for a more sane and powerful hardware support for memory introspection on commodity architectures."

Thanks!

The reviewers

Our shepherd, Julien

Research is supported by Canada's NSERC Discovery Grant program

https://github.com/selaing/Cage