scat

Learning from a Single Execution of a Binary

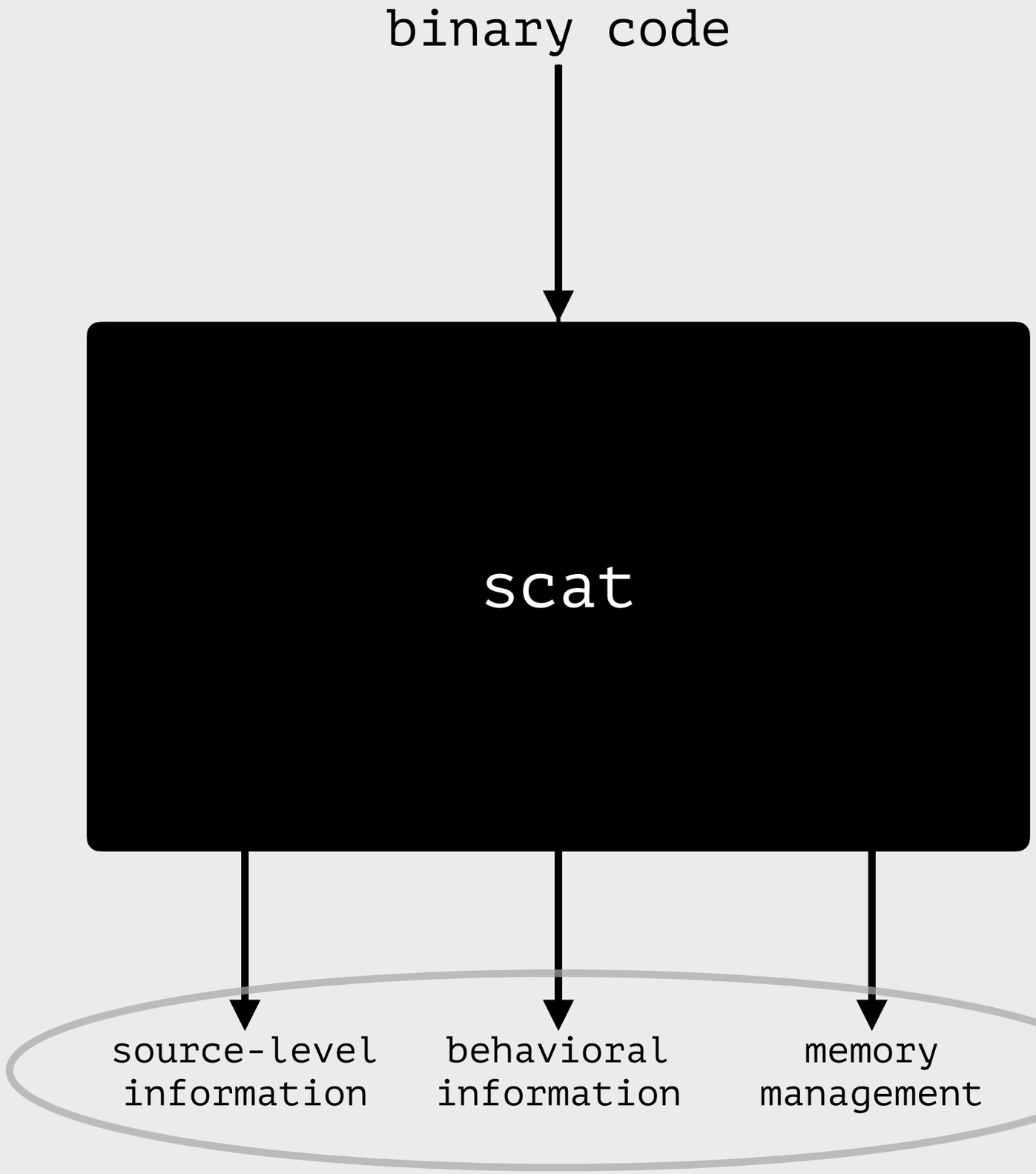
23/02/17

Franck de Goër

Christopher Ferreira

Laurent Mounier

BINSEC



at level
of **functions**

WARNING!

COMMAND LINE INTERFACE TOOL

MAIN FEATURES

scat is **not** only music-relative

MAIN FEATURES

arity | type | coupling | (wip) allocator

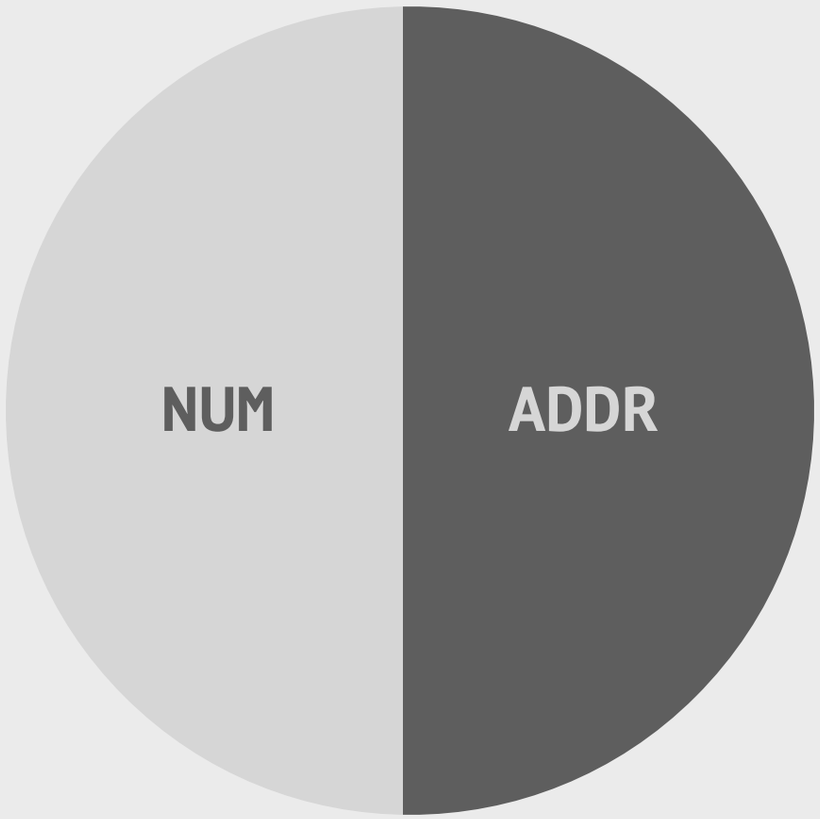
```
scat > launch arity pgm/bin/mupdf-x11 poc.pdf
[*] Launching arity inference on pgm/bin/mupdf-x11
[*] /usr/bin/pin/pin -ifeellucky -t ./bin/obj-intel64/arity.so -o
./log/mupdf-x11_arity_1487084838.log -logfile ./log/mupdf
x11_arity_1487084838.dbg  -- pgm/bin/mupdf-x11 poc.pdf
[*] Inference results logged in ./log/mupdf-
x11_arity_1487084838.log
[*] Execution time: 18.441624s

scat > display mupdf-x11 arity
mupdf-x11:4216716:wincursor: 2 -> 0
mupdf-x11:4221641:onmouse: 5 -> 0
mupdf-x11:4225337:fz_mini: 2 -> 1
…
Inference
| Date:                          2017-02-14 16:07:18
| Total functions inferred:    1128
- Program functions inferred:  775
```

MAIN FEATURES

arity | type | coupling | (wip) allocator

Undertyping



MAIN FEATURES

arity | **type** | coupling | (wip) allocator

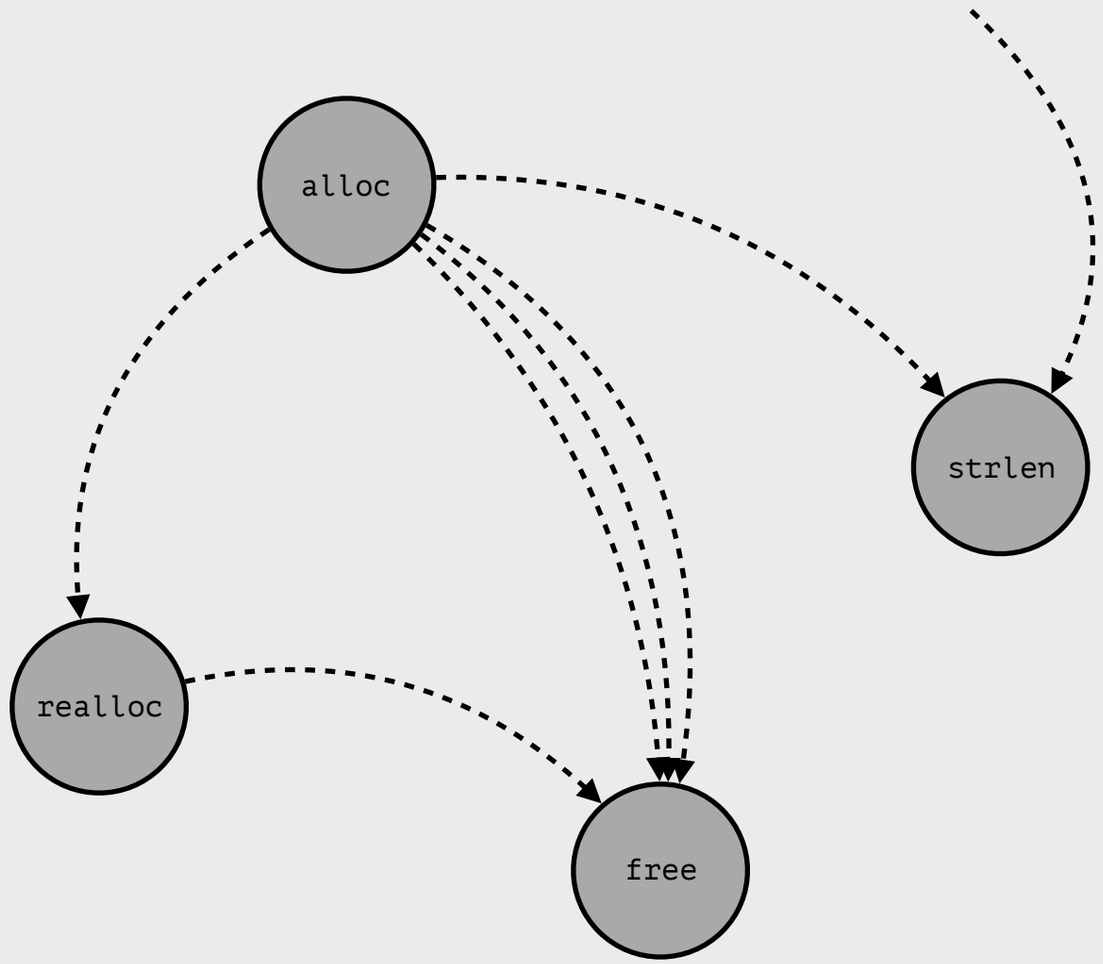
```
scat > launch type pgm/bin/mupdf-x11 poc.pdf
[*] Launching type inference on pgm/bin/mupdf-x11
[*] /usr/bin/pin/pin -ifeellucky -t ./bin/obj-intel64/type.so -o
./log/mupdf-x11_type_1487085653.log -logfile
./log/mupdf-x11_type_1487085653.dbg -i
./log/mupdf-x11_arity_1487084838.log -- pgm/bin/mupdf-x11 poc.pdf
[*] Inference results logged in ./log/mupdf-
x11_type_1487085653.log
[*] Execution time: 16.957328s

scat > display mupdf-x11 type
void onmouse(int, int, int, int, undef);
int fz_mini(int, int);
addr next_pool_image(void);
…
Inference
| Date:                          2017-02-14 16:20:53
| Total functions inferred:    1127
- Program functions inferred:  774
```

MAIN FEATURES

arity | type | coupling | (wip) allocator

data flow

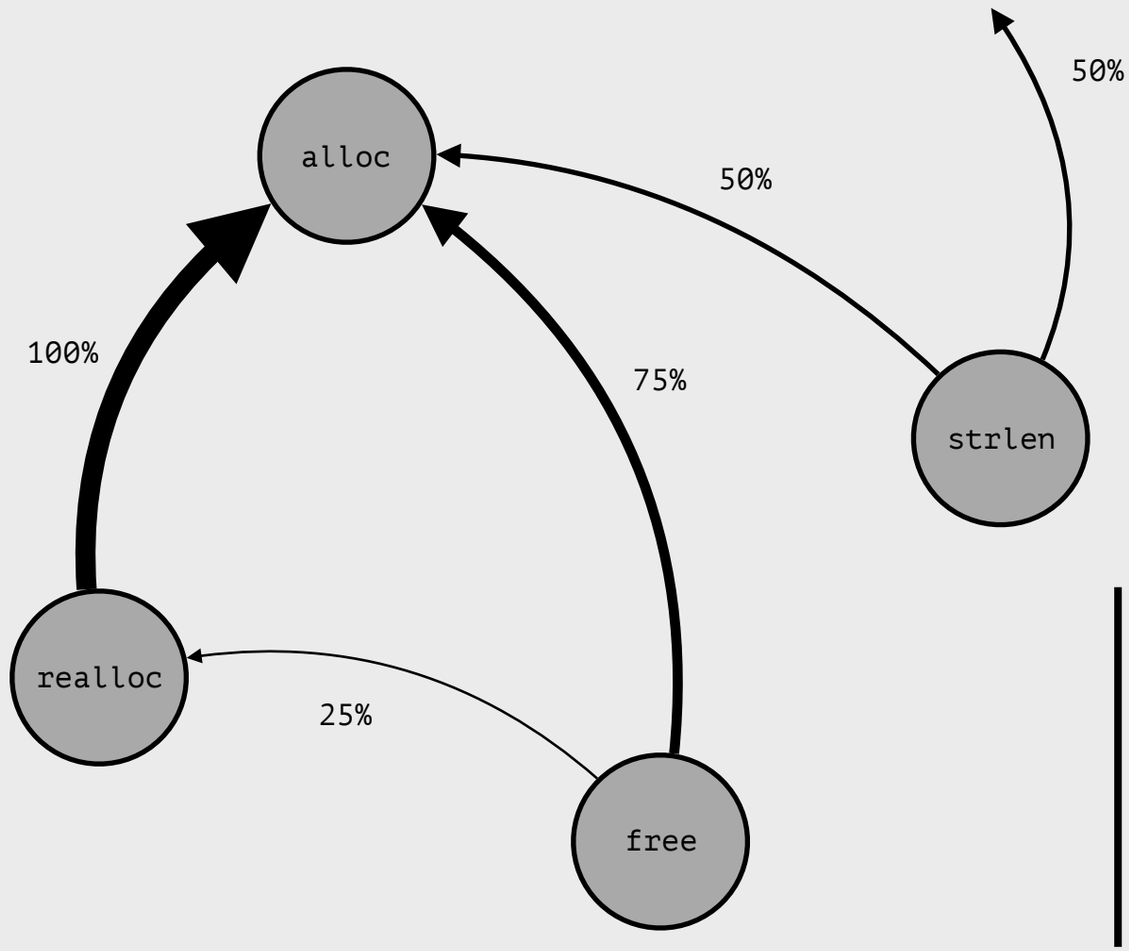


MAIN FEATURES

arity | type | coupling | (wip) allocator

data flow

coupling



malloc -> free

fopen -> fclose

…

MAIN FEATURES

arity | type | coupling | (wip) allocator

```
scat > launch couple pgm/bin/mupdf-x11 poc.pdf
[*] Launching couple inference on pgm/bin/mupdf-x11
[*] /usr/bin/pin/pin -ifeellucky -t ./bin/obj-intel64/couple.so -o
./log/mupdf-x11_couple_1487574029.log -logfile ./log/mupdf-
x11_couple_1487574029.dbg -i ./log/mupdf-x11_type_1487085653.log --
pgm/bin/mupdf-x11 poc.pdf
[*] Inference results logged in ./log/mupdf-
x11_couple_1487574029.log
[*] Execution time: 12.592736s

scat > couple mupdf-x11
[*] parsing memory blocks...
[*] #f: 982 | #g: 380 | #in: 131481 | #out: 770702
[*] computing couples...
libc.so.6:583696: -- (1.00) --> mupdf-x11:4266681:fz_device_gray
libc.so.6:583696: -- (1.00) --> mupdf-x11:4640098:pdf_keep_obj
…
```

MAIN FEATURES

arity | type | coupling | (wip) allocator

```
scat > launch memalloc pgm/bin/mupdf-x11 poc.pdf
[*] Launching memalloc inference on pgm/bin/mupdf-x11
[*] /usr/bin/pin/pin -ifeellucky -t ./bin/obj-intel64/memalloc.so -
o ./log/mupdf-x11_memalloc_1487170268.log -logfile ./log/mupdf-
x11_memalloc_1487170268.dbg -i ./log/mupdf-x11_type_1487085653.log
-- pgm/bin/mupdf-x11 poc.pdf
[*] Inference results logged in ./log/mupdf-
x11_memalloc_1487170268.log
[*] Execution time: 11.84821s

scat > memcomb mupdf-x11 --lib
[*] allocator found - libc.so.6:537936:malloc
[*] liberator found - libc.so.6:539248:__libc_free
```

PHILOSOPHY

aka design choices, but « philosophy » sounds better

PHILOSOPHY

open-source	******* analysis	scalable	modulable
one single execution		(stripped) binaries	

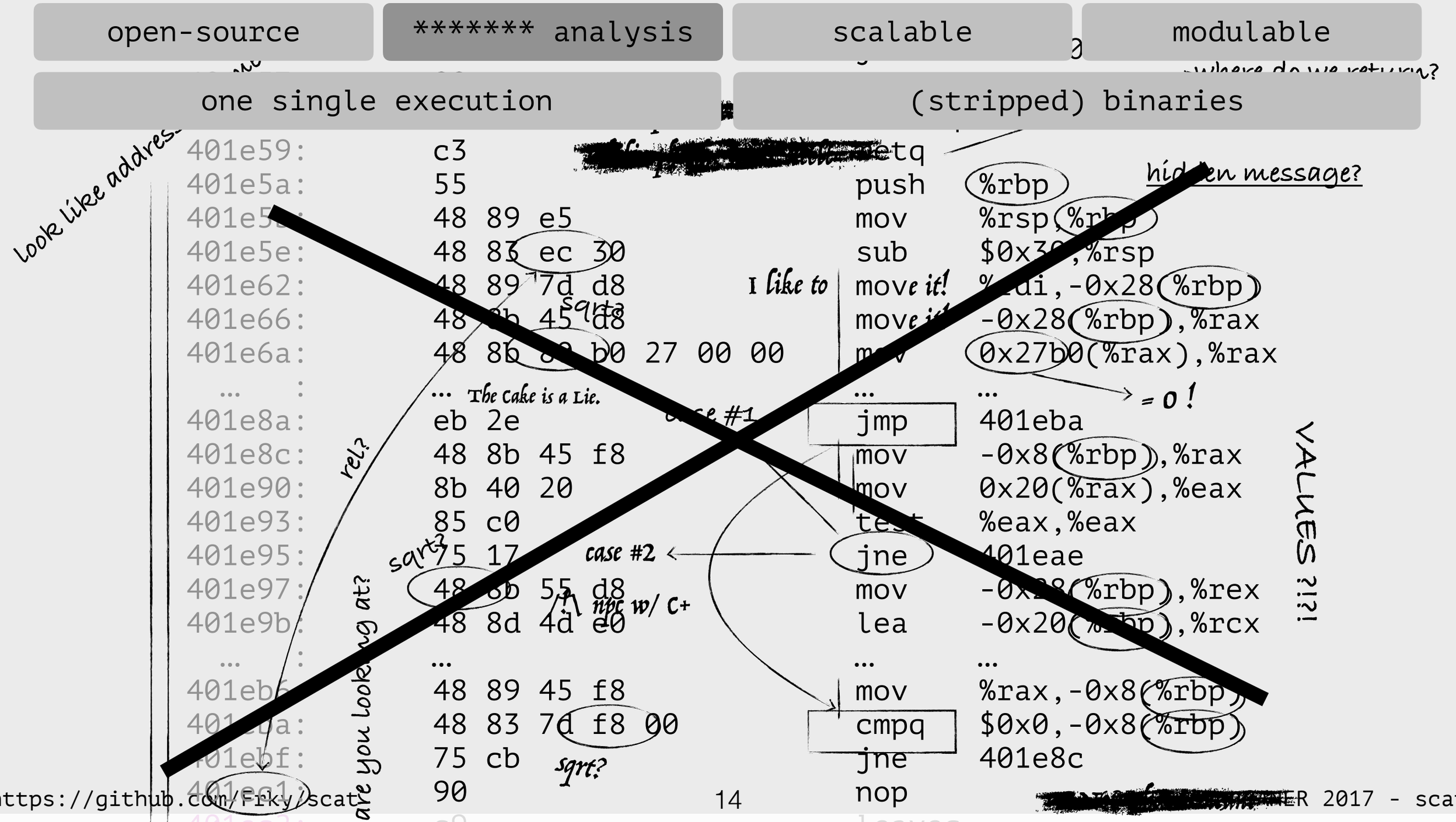
https://github.com/Frky/scat

Feel free to contribute

(plus there is a README)

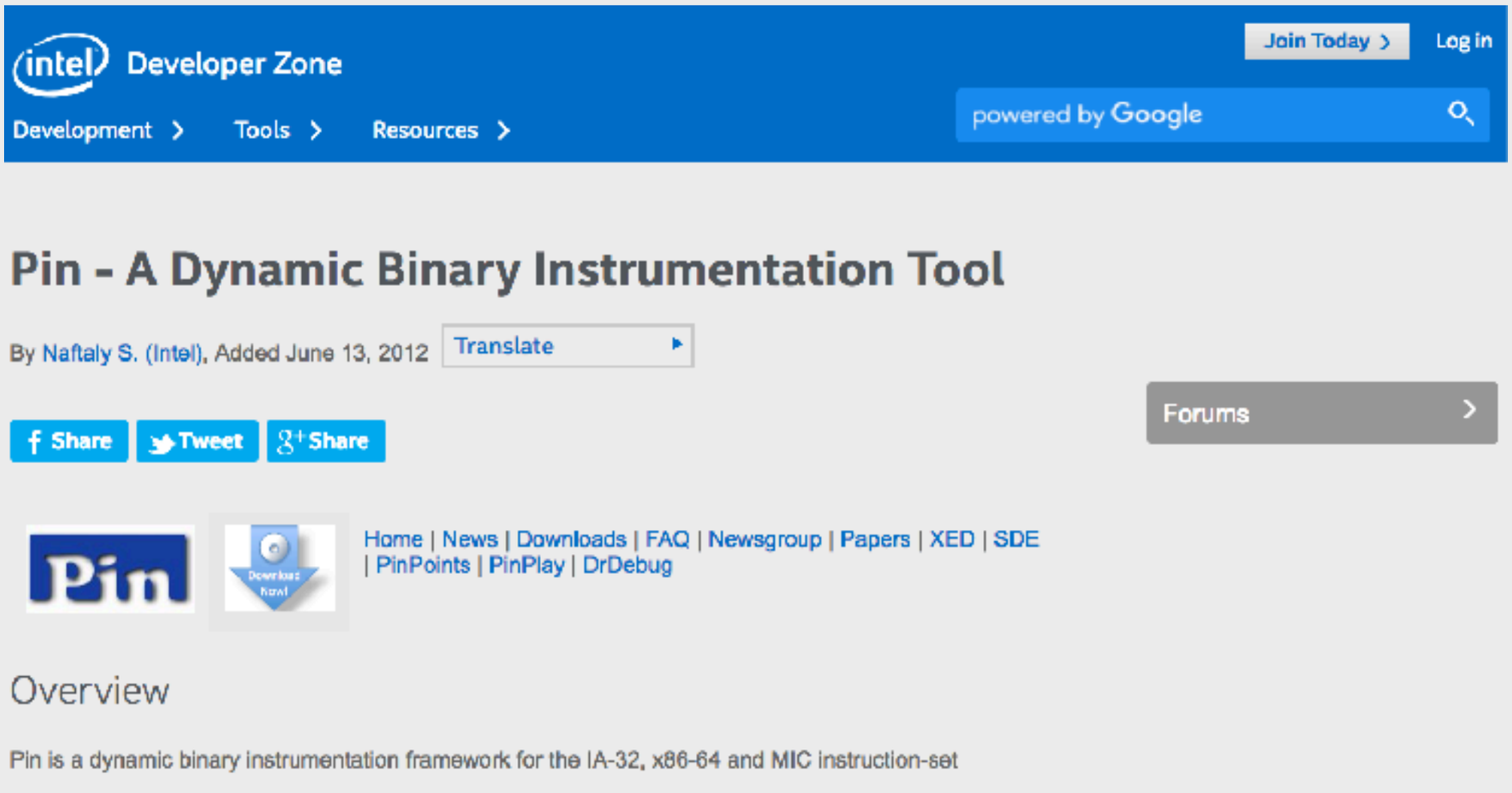
PHILOSOPHY

open-source	******* analysis	scalable	modulable
one single execution		(stripped) binaries	



PHILOSOPHY

open-source	dynamic analysis	scalable	modulable
one single execution		(stripped) binaries	



PHILOSOPHY

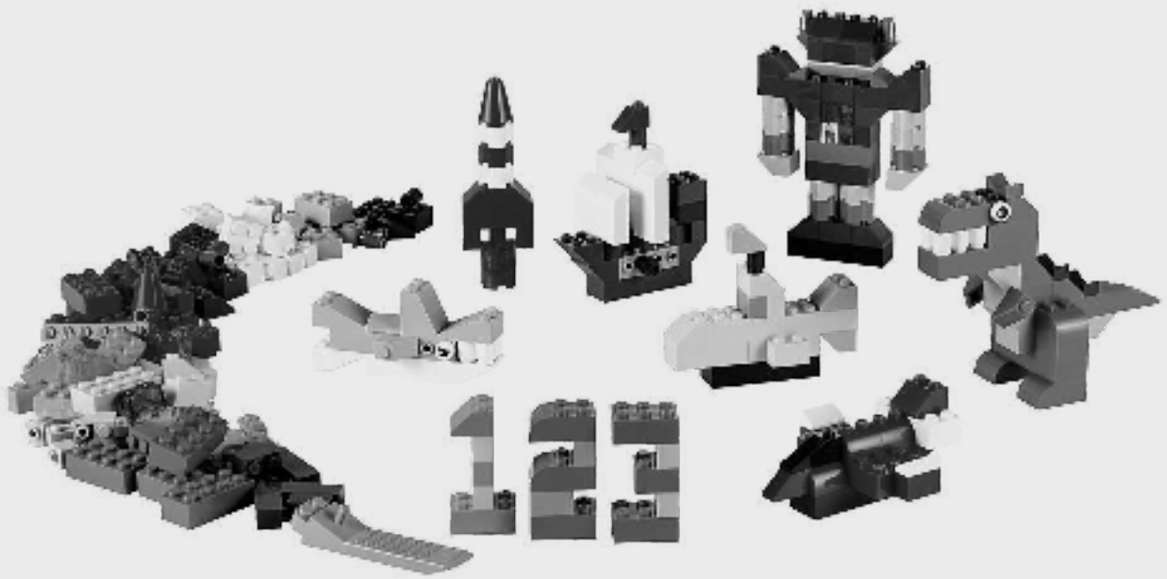
open-source	dynamic analysis	scalable	modulable
one single execution		(stripped) binaries	

Execution of mupdf: 15s

(on a 10-page PDF document)

PHILOSOPHY

open-source	dynamic analysis	scalable	modulable
one single execution		(stripped) binaries	



https://github.com/Frky/scat#add-your-own-pintool

PHILOSOPHY

open-source | dynamic analysis | scalable | modulable

one single execution | (stripped) binaries

~~while True:~~
~~execute()~~

PHILOSOPHY

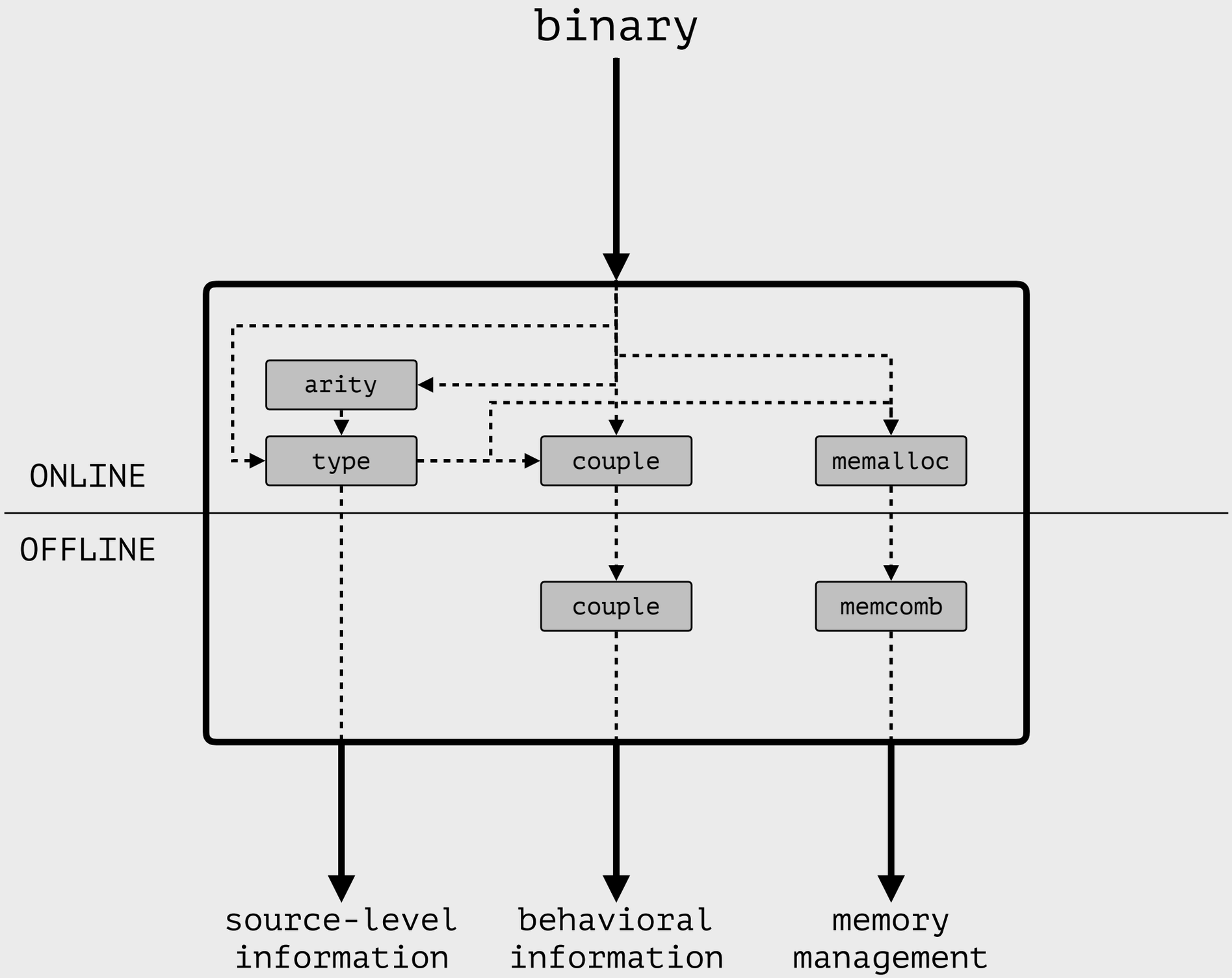
open-source | dynamic analysis | scalable | modulable

one single execution | (stripped) binaries

- binary program
- no source code
- possibly stripped
- can be obfuscated (dynamic!)

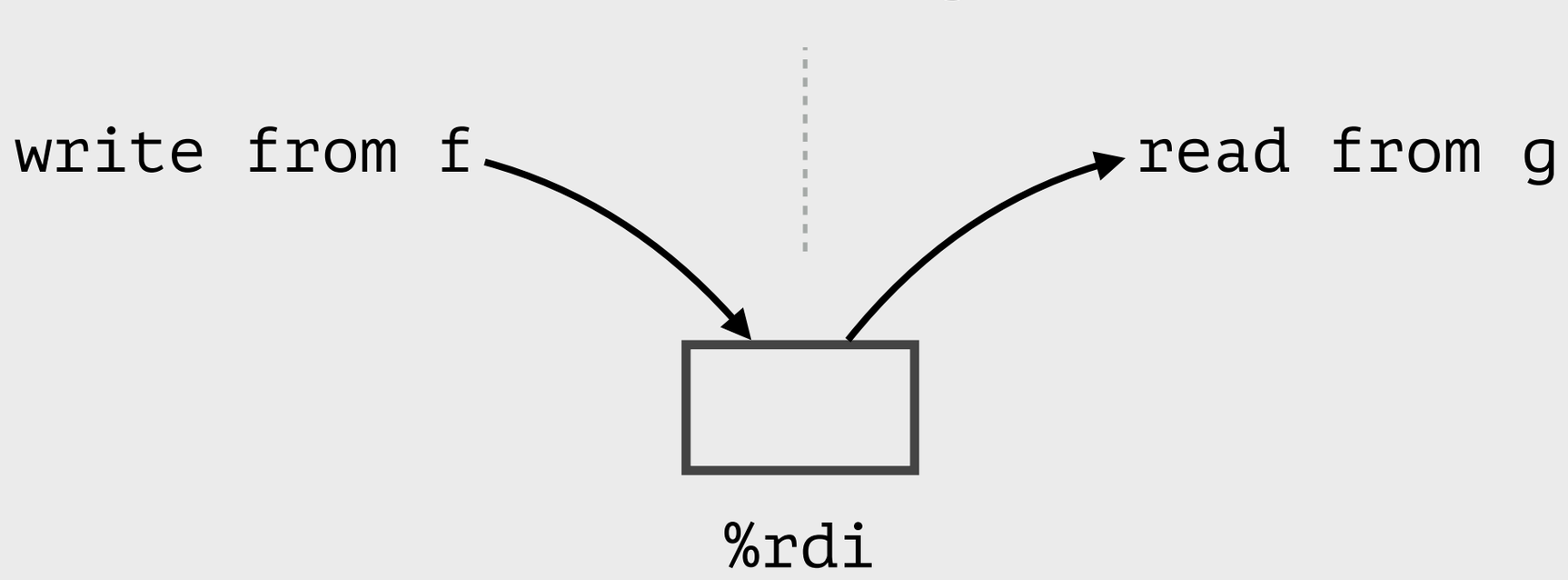
INSIDE scat

What if we analyze **scat** with **scat**? (Just kiddin’)


binary
arity
type
couple
memalloc
ONLINE
OFFLINE
couple
memcomb
source-level information
behavioral information
memory management

INSIDE scat

parameters | addresses | data flow | (wip) ALLOC



INSIDE scat

parameters | **addresses** | data flow | (wip) ALLOC

Inference of address space

- load/store operands are addresses
- a value between two addresses is an address

```
401e5a:       55                      push   %rbp
401e5b:       48 89 e5                mov    %rsp,%rbp
401e5e:       48 83 ec 30             sub    $0x30,%rsp
401e62:       48 89 7d d8             mov    %rdi,-0x28(%rbp)
401e66:       48 8b 45 d8             mov    -0x28(%rbp),%rax
401e6a:       48 8b 80 b0 27 00 00    mov    0x27b0(%rax),%rax  ---------▶   0x7fe047407b00 is an address
  …   :       …                       …      …
```

INSIDE scat

parameters | addresses | data flow | (wip) ALLOC

Assumption

coincidence of **address** values => data flow

```
4006ed:       48 83 ec 28             sub    $0x28,%rsp
4006f1:       48 89 7d d8             mov    %rdi,-0x28(%rbp)
...    :       ...                    ...
4008d9:       c9                      leaveq
4008da:       c3                      retq   ------------------------->  returns 0x7fe047419010
...    :       ...                    ...
4009f6:       55                      push   %rbp
4009f7:       48 89 e5                mov    %rsp,%rbp
4009fa:       48 83 ec 30             sub    $0x30,%rsp
4009fe:       48 89 7d d8             mov    %rdi,-0x28(%rbp) --------->   takes 0x7fe047419010
400a02:       48 89 75 d0             mov    %rsi,-0x30(%rbp)
```

INSIDE scat

parameters | addresses | data flow | (wip) ALLOC

Main heuristic

ALLOC is the function that outputs the **greatest number** of new addresses.

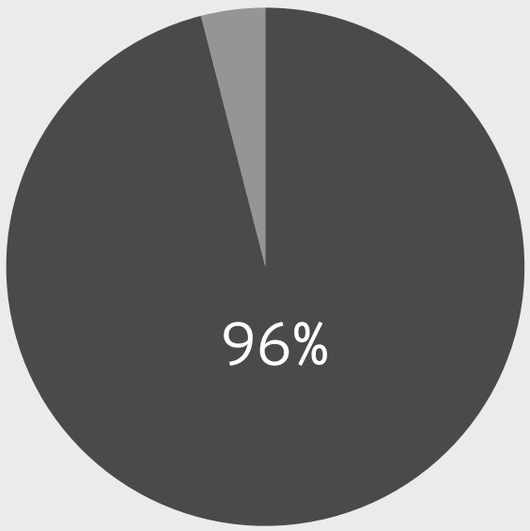
EVALUATION

It’s time to know if it really works…

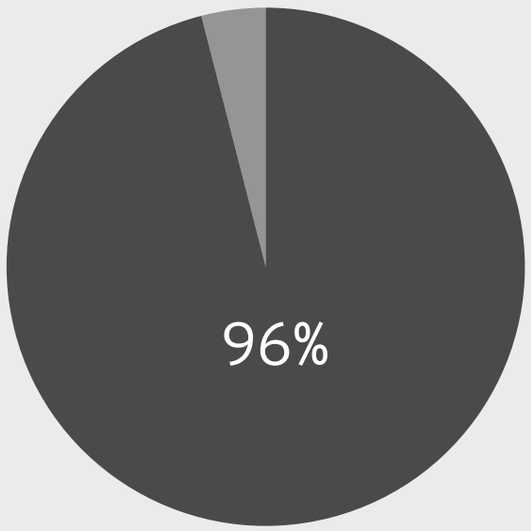
EVALUATION

accuracy | limitations

scat > accuracy * arity



scat > accuracy * type



bash | grep | mupdf | git | tar | xterm | vim

EVALUATION

accuracy | limitations

- Only x86-64 binaries
- Calling-convention dependent
- No support of OOP
- Binary coverage

scat

« Tell me! Everyone is picking up on that feline beat / 'Cause everything else is obsolete.

- Strictly high-buttoned shoes. »