The Lind Virtual Machine

Userspace system call emulation and auditing

Joey Pabalinas

contracted for implementation by

Professor Justin Cappos

Secure Systems Laboratory, New York University

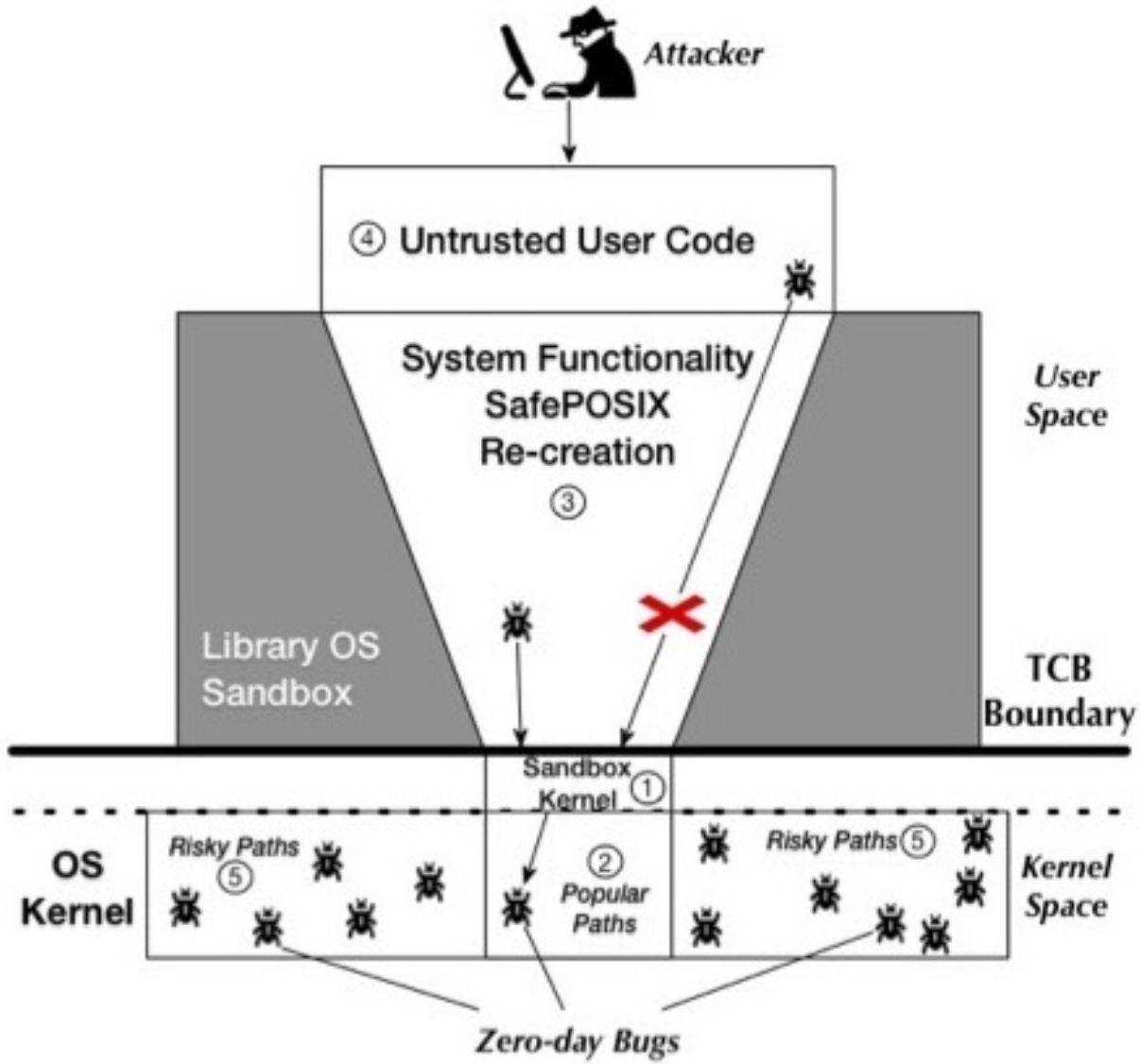
December 20, 2018

Purpose

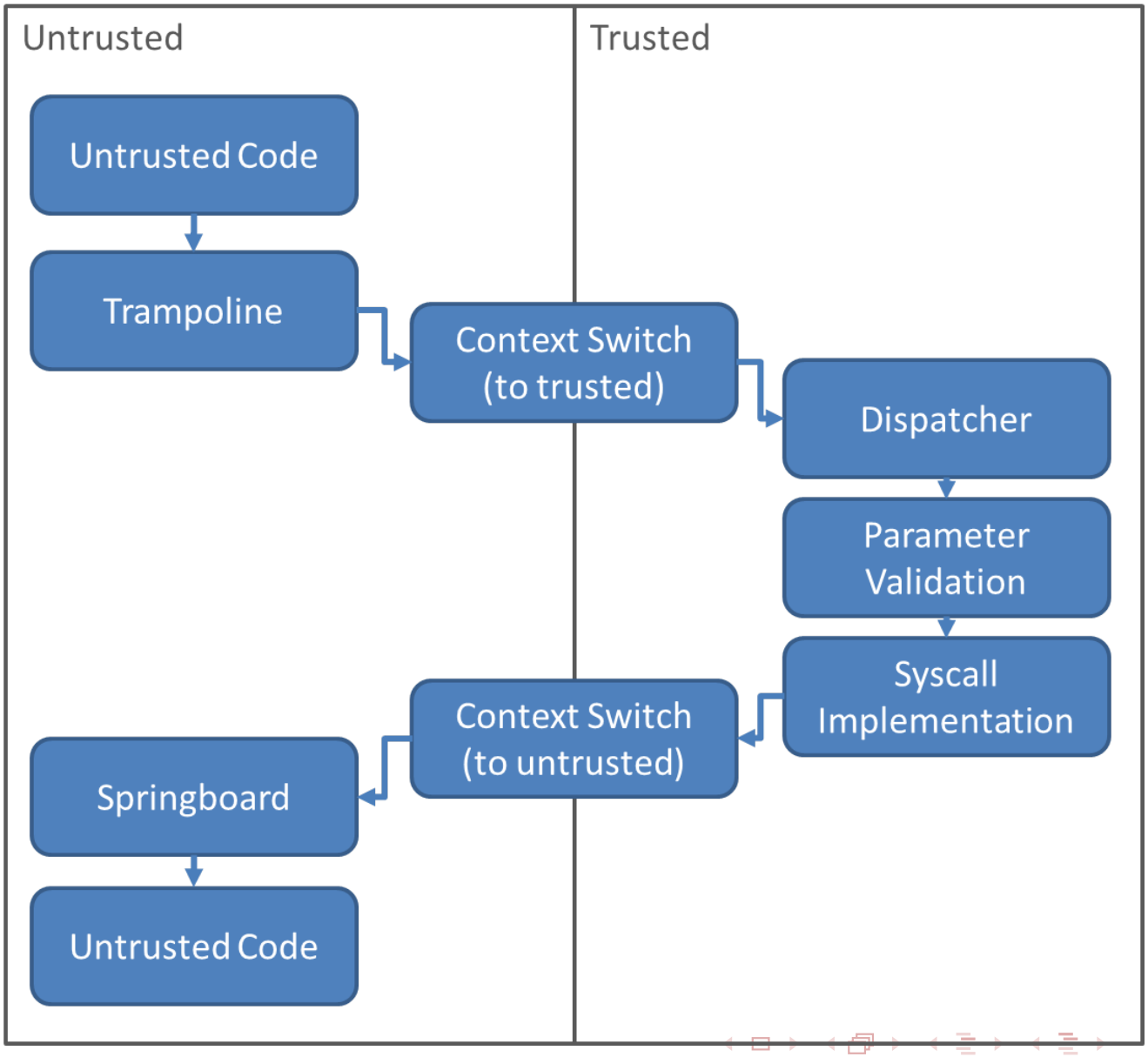
Lind is a fork of Google's NaCl.

1. Security goals fall somewhere in-between a virtual machine and a container
2. All system calls are trapped and handled by Lind
   - Safe System Calls - Python POSIX Compatibility Layer
   - Low-level System Calls - NaCl Internal Emulation
   - Unsafe System Calls - Return Unimplemented

Architecture



System Call Anatomy



Debugging Segfaults Inside Lind

```
$ lind -g /bin/bash -c '/hello; /hello'

Starting program: sel_ldr [...]
Thread 7 "sel_ldr" received signal
SIGSEGV, Segmentation fault.

0x0000690501297143 in ?? ()
```

Our Incrediby Useful Backtrace

```
@ [#7:None()] backtrace

#0  0x0006905012971 43 in ?? ()
#1  0x0000000000000000 in ?? ()
```

This ASM Doesn't Seem Right...

```
@ [#7: None()] x/10i $rip

=> 0x690501297143:   lock cmpxchg %edx,(%r15,%r13,1)
   0x690501297149:   test   %eax,%eax
   0x69050129714b:   je     0x6905012971a0
   0x69050129714d:   nopl   0x0(%rax)
   0x690501297151:   data16 nopw %cs:0x0(%rax,%rax,1)
   0x690501297160:   mov    %r13d,%edi
   0x690501297163:   mov    %r12d,%esi
   0x690501297166:   xor    %ebx,%ebx
   0x690501297168:   nopl   0x0(%rax)
   0x69050129716c:   data16 nopw %cs:0x0(%rax,%rax,1)
```

The Fault Address

```
@ [#7:None()] print $_siginfo._sifields._sigfault.si_addr

$5 = (void *) 0x6905000368e0
```

Conclusion: The Problem Address: r15 + r13

```
@ [#7: None()] info registers

rax            0x0                 0
rbx            0x10057910          268794128
rcx            0x1297120           19493152
[...]
r13            0x368e0             223456
r14            0x0                 0
r15            0x690500000000      115470195752960
rip            0x690501297143      0x690501297143
eflags         0x10246             [ PF ZF IF RF ]
```

Conclusion: Memory Not Writable

```
$ grep 6905000 /proc/$(pidof sel_ldr)/maps

    68fb00000000-690500000000 ---p 00000000 00:00 0
    690500000000-690500010000 rw-p 00000000 00:00 0
=> 690500010000-690510020000 r-xp 00000000 00:00 0
```

Questions?

Questions?

o..

Questions?

o..

.o.

Questions?

o..

.o.

[1] 3286 segmentation fault (core dumped)

Thank you.