

Code sandboxing

Alpha Abdoulaye -Pierre Marsais

Introduction

Solutions

Elf tricker

Virtualization

Conclusion

Code sandboxing

Alpha Abdoulaye - Pierre Marsais

EPITA Systems/Security Laboratory (LSE)

July 14, 2017



Code sandboxing

Alpha Abdoulaye Pierre Marsais

Introduction

Solutions

If tricke

Virtualizatio

VIITUUIIZUIIOI

Conclusion

Introduction



What do we want?

Code sandboxina

Introduction

- Limit usage of some resources such as system calls and shared object functions
- But not from the whole program (we trust our libc.so, ld.so,...)



Code sandboxing

Code sandboxing

Alpha Abdoulaye Pierre Marsais

Introduction

EIT Trickery

Virtualizatio

Conclusion

- Needed when executing untrusted code on your machine.
- Allow or deny use of some "resources"
- Usually theses "resources" are accessed through syscalls
- We already have namespaces(7) and seccomp(2)



Are all needs fulfilled?

Code sandboxing Alpha

Alpha Abdoulaye Pierre Marsais

Introduction

Solutions

Elf tricker

Virtualization

VIITaalizatio

There is no "ready-to-use" solution for:

- Function usage
- Library usage



Technical choices

Code sandboxing Alpha

Abdoulaye Pierre Marsais

Introduction

SOlution

Elf fricker

Virtualizatio

. . .

Aim for:

- Speed
- Reliability
- Security



Code sandboxing

Alpha Abdoulaye Pierre Marsais

Introductio

Solutions

Elf tricko

Virtualization

Conclusion

Solutions



Solving the problem

Code sandboxing Alpha

Abdoulaye Pierre Marsais

Introductio

Solutions

EII IIICKEI)

Virtualization

. . .

- Trap at each function call
- Check if the call is righteous
- Continue as if nothing happened



Code sandboxing

Alpha Abdoulaye Pierre Marsais

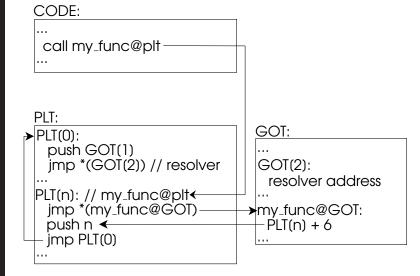
Introduction

Solutions

.

Virtualizatio

VIII TAGILLATIO





Code sandboxing

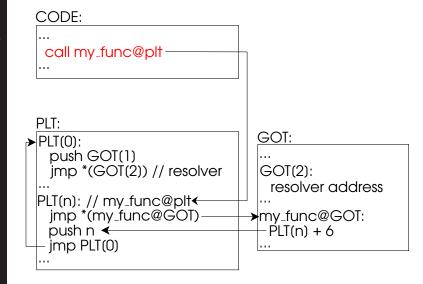
Alpha Abdoulaye Pierre Marsais

Introduction

Solutions

Virtualizatio

VII. GGIIZGIIC





Code sandboxing

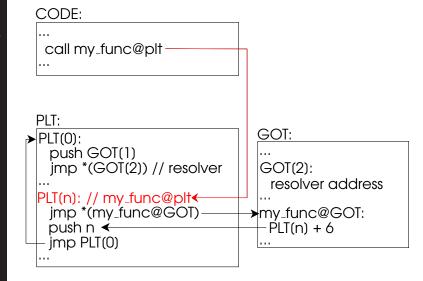
Alpha Abdoulaye Pierre Marsais

Introduction

Solutions

Virtualizatio

VII. GGIIZGIIC





Code sandboxina

Alpha Abdoulaye Pierre Marsais

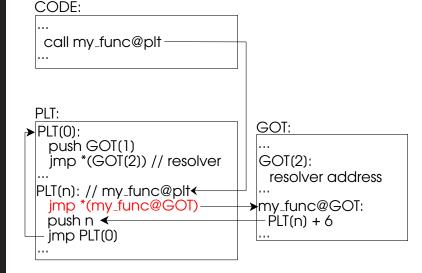
Introduction

Solutions

=....

Virtualizatio

VII. GGIIZGIIC





Code sandboxina

Alpha Abdoulaye Pierre Marsais

Introduction

Solutions

COlditorio

Virtualizatio

VIITUGIIZGIIC

```
CODF:
 call my_func@plt
PLT:
                                GOT:
PLT(0):
  push GOT(1)
  imp *(GOT(2)) // resolver
                                GOT(2):
                                 resolver address
PLT(n): // my_func@plt←
  jmp *(my_func@GOT)
                              >my_func@GOT:
  push n ∢
                                 PLT(n) + 6
  imp PLT(0)
```



Code sandboxing

Alpha Abdoulaye Pierre Marsais

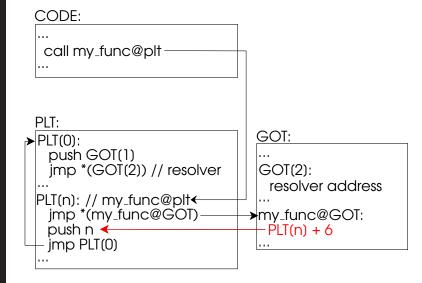
Introduction

Solutions

=....

Virtualizatio

VII TGGILLGITE





Code sandboxina

Solutions

```
CODF:
 call my_func@plt
PLT:
                                GOT:
PLT(0):
  push GOT(1)
  imp *(GOT(2)) // resolver
                                GOT(2):
                                 resolver address
PLT(n): // my_func@plt←
  jmp *(my_func@GOT)
                              >my_func@GOT:
  push n ◀
                                 PLT(n) + 6
  imp PLT(0)
```



Code sandboxing

Alpha Abdoulaye Pierre Marsais

Introduction

Solutions

Elf tricke

Virtualizatio

VIII GGILLGIII

```
CODF:
 call my_func@plt
PLT:
                               GOT:
PLT(0):
  push GOT(1)
  imp *(GOT(2)) // resolver
                                GOT(2):
                                 resolver address
PLT(n): // my_func@plt←
  jmp *(my_func@GOT)
                              >my_func@GOT:
  push n ∢
                                 PLT(n) + 6
  imp PLT(0)
```



Code sandboxing

Alpha Abdoulaye Pierre Marsais

Introduction

Solutions

Flf tricke

Virtualizatio

VIII GGIILGIII

```
CODF:
 call my_func@plt
PLT:
                               GOT:
PLT(0):
  push GOT(1)
  imp *(GOT(2)) // resolver
                                GOT(2):
                                 resolver address
PLT(n): // my_func@plt←
  jmp *(my_func@GOT)
                              >my_func@GOT:
  push n ∢
                                 PLT(n) + 6
  imp PLT(0)
```



Code sandboxing

Alpha Abdoulaye Pierre Marsais

Introduction

Solutions

Elf tricke

Virtualizatio

VIII GGIILGIII

```
CODF:
 call my_func@plt
PLT:
                               GOT:
PLT(0):
  push GOT(1)
  imp *(GOT(2)) // resolver
                                GOT(2):
                                 resolver address
PLT(n): // my_func@plt←
  jmp *(my_func@GOT)
                              >my_func@GOT:
  push n ∢
                                -my_func
  imp PLT(0)
```



Code sandboxing

Alpha Abdoulaye Pierre Marsais

Introduction

Solutions

Elf triol(o

Virtualizatio

7....aa...za...c

```
CODF:
 call my_func@plt
PLT:
                               GOT:
PLT(0):
  push GOT(1)
  imp *(GOT(2)) // resolver
                               GOT(2):
                                 resolver address
PLT(n): // my_func@plt←
  jmp *(my_func@GOT)
                              >my_func@GOT:
  push n ∢
                                my_func
  imp PLT(0)
```



Code sandboxing

Alpha Abdoulaye Pierre Marsais

Introduction

Solutions

Virtualizatio

VII. GGIIZGIIC

CODE: call my_func@plt . . . PLT: GOT: PLT(n): // my_func@plt← jmp *(my_func@GOT) →my_func@GOT: my_func . . .



Code sandboxing

Alpha Abdoulaye Pierre Marsais

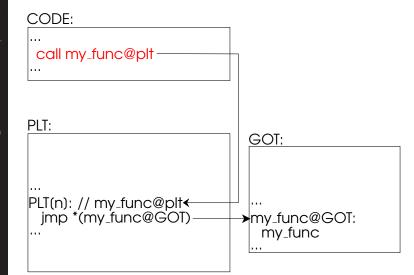
Introduction

Solutions

Elf triol(o)

Virtualizatio

7................





Code sandboxing

Alpha Abdoulaye Pierre Marsais

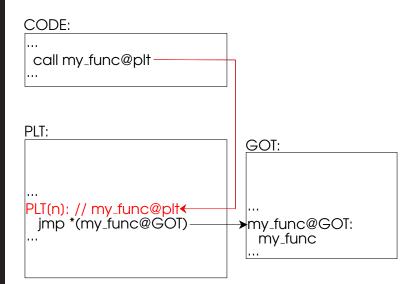
Introduction

Solutions

Elf tricke

Virtualizatio

. . .





Code sandboxing

Alpha Abdoulaye Pierre Marsais

Introduction

Solutions

Virtualizatio

VII. GGIIZGIIC

CODE: call my_func@plt . . . PLT: GOT: PLT(n): // my_func@plt← jmp *(my_func@GOT) →my_func@GOT: my_func . . .



Code sandboxina

Solutions

```
CODE:
 call my_func@plt
. . .
PLT:
                                GOT:
PLT(n): // my_func@plt←
 jmp *(my_func@GOT)
                               >my_func@GOT:
                                 my_func
. . .
```



Code sandboxing

Alpha Abdoulaye Pierre Marsais

Introduction

Solutions

Ell IIICKely

Virtualizatio

. . .

We have two solutions:

- Disallow GOT reads of the sandboxed ELF
- Disallow code execution of executable mapping

Then handle the rights violation and check if the ressource access is allowed or not.



Solution analysis

Code sandboxing

Alpha Abdoulaye Pierre Marsais

Introductio

Solutions

Elf tricke

Virtualizatio

Conclusion

GOT protection can be bypassed.

The correct solution would be unallowing execution of executable mappings.



Code sandboxing

Elf trickery

Elf trickery



Goal

Code sandboxing

Elf trickery

Can we solve our problem without privileged code?



What do we need?

Code sandboxing

Elf trickery

- Change mapping rights
- Handle mapping violation



What can we use?

Code sandboxing

Elf trickery

- ptrace(2)
- procfs(5)



Is this enough?

Code sandboxing

Alpha Abdoulaye Pierre Marsais

Introductio

SOlutions

Elf trickery

Virtualizatio

No:

- How to change mapping permissions from the tracer?
- What about non-GOT data on GOT pages?
- What about multithreaded programs?



The first hack

Code sandboxing

Alpha Abdoulaye Pierre Marsais

Introduction

SOlutions

Elf trickery

Virtualization

How to change mappings from the tracer?

- We can link an ELF to the sandboxed binary.
- We can use signal handlers in order to protect and unprotect the GOT.
- Use ELF constructors to setup everything.



The second hack

Code sandboxing

Alpha Abdoulaye Pierre Marsais

Introduction

SOlutions

Elf trickery

Virtualization

How to handle non-GOT data on GOT pages?

- GOT doesn't necessarily start and end at pages boundaries
- We can force this, with a custom linker script
- All we need is to customize the default linker script to align the GOT and export its size



We want reasonable performances

Code sandboxing

Alpha Abdoulaye Pierre Marsais

Introductio

Solutions

Elf trickery

Virtualizatio

Canalusian

- LD_BIND_NOW=1
- Cache authorized GOT access



Limitations

Code sandboxing

Alpha Abdoulaye Pierre Marsais

Introduction

JOIGHOI I

Elf trickery

VIITUGIIZGIIG

We can't allow a lot of stuff for the sandboxed application:

- We currently need to link an object to the sandboxed application
- mprotect can't be used to PROT_READ the GOT
- SIGSEGV can't be handled
- Libraries addresses can be leaked



Problems

Code sandboxing

Alpha Abdoulaye Pierre Marsais

Introduction

Joiumon

Elf trickery

Virtualizatio

- Address space leaks
 - /proc/self/*
 - auxv
 - some syscalls
 - addresses on stack and structures
- Functions pointers in structures
- dlopen(3), dlsym(3)...



Shared object randomization

Code sandboxing

Alpha Abdoulaye Pierre Marsais

Introduction

301U110118

Elf trickery

- Idea taken from OpenBSD
- If the user gets a libc address, and knows what libc is used, it can easily call any function
- The problem arise for any libs, but the libc is the more annoying for us
- We currently have a script to randomize the glibc
- Additional work needed for other libraries



Code sandboxing

Alpha Abdoulaye Pierre Marsais

Introductio

Solutions

Elf tricke

Virtualization



We need to go deeper

Code sandboxing

Alpha Abdoulaye Pierre Marsais

Introduction

Solutions

2.1

- Extended Page Table
- Additional translation level
- Hardware assisted
- Solve multi-threading problem



Bareflank

Code sandboxing

Alpha Abdoulaye Pierre Marsais

Introductio

30IUIIOI I

EII Tricker

- Lightweight
- Extendable
- "C++ in Kernel"
- Multi-platform



Track program behaviour

Code sandboxing

Alpha Abdoulaye Pierre Marsais

Introductio

Solution

- ptrace(2)
- /proc/[pid]/maps
- /proc/[pid]/pagemap
- linkmap, symbols, etc.



Report and Handle

Code sandboxing

Alpha Abdoulaye Pierre Marsais

Introduction

JOIGHOIR

EITTRICKERY

Virtualization

_ . . .

- vmcall to report to hypervisor
- Virtual Machine Control Structure
 - VM State
 - Global Configuration
- VM Exits
- Enable EPT violation
- Convert to Virtualization Exception



Pull the strings

Code sandboxing

Alpha Abdoulaye Pierre Marsais

Introductio

Solutions

,

Virtualization

- Handle #VE
- Trap on protected code
- Protect executable and check
- Decide!!!!
- And so on...



Recap

Code sandboxing

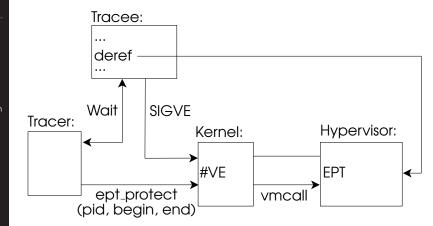
Alpha Abdoulaye Pierre Marsais

Introduction

Solutions

Elf tricker

Virtualization





Code sandboxing

Alpha Abdoulaye Pierre Marsais

Introduction

Solutions

-If tricker

Virtualization

Conclusion



Further work

Code sandboxing

Alpha Abdoulaye Pierre Marsais

Introductio

Solutions

Virtualization

- Be sure that our solution is foolproof
- handle multithreaded programs
- Work on performance
- What about statically linked ELFs?
- ROP?



Questions

Code sandboxing

Alpha Abdoulaye Pierre Marsais

Introduction

Solutions

Elf tricker

Virtualization

Conclusion

Questions?



Contact

Code sandboxing Alpha

Alpha Abdoulaye Pierre Marsais

Introductio

Solutions

Flf trickery

Virtualizatio

- Alpha Abdoulaye alpha@lse.epita.fr
- Pierre Marsais pierre.marsais@lse.epita.fr