Volafox

Finding treasure in your memory

Who Am I

- Kyeongsik Lee, nick: n0fate
- Digital Forensic Research Center, CIST
- research: FreeBSD, Solaris, Mac OS X

Index

- Introduction
- Memory Analysis Process
- Physical Memory Analysis
  - Physical Memory Acquisition
  - symbol acquisition
  - Virtual Address Translation
- volafox

Introduction

- DFRWS 2005 Memory Forensic Challenge
- rootkit
- many operating system

Target System

Target System

Target System

- Intel 32bit
- Intel 64bit
- PowerPC

Target System

Intel 32bit

Intel 64bit

PowerPC

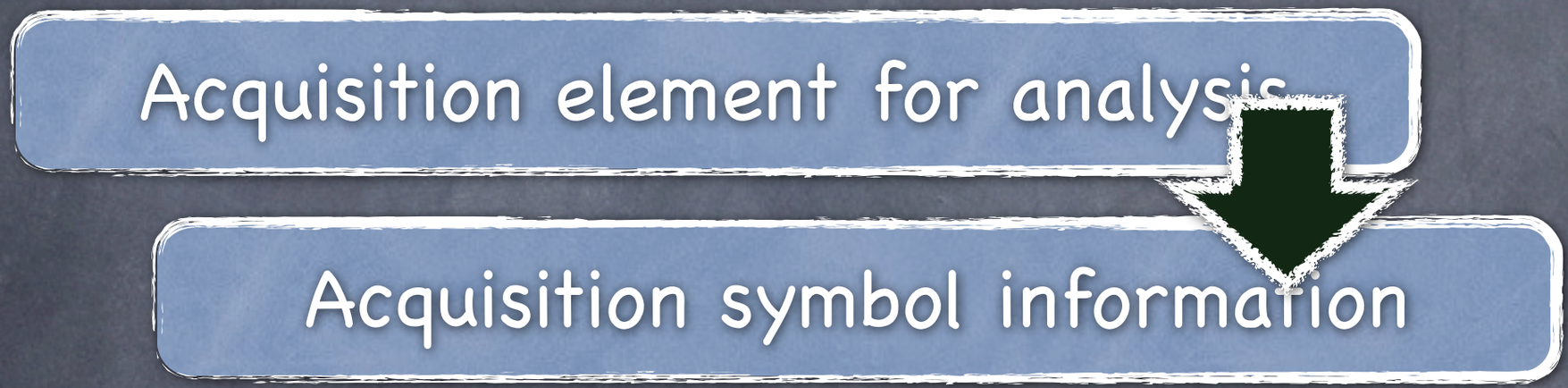
Process

Acquisition element for analysis

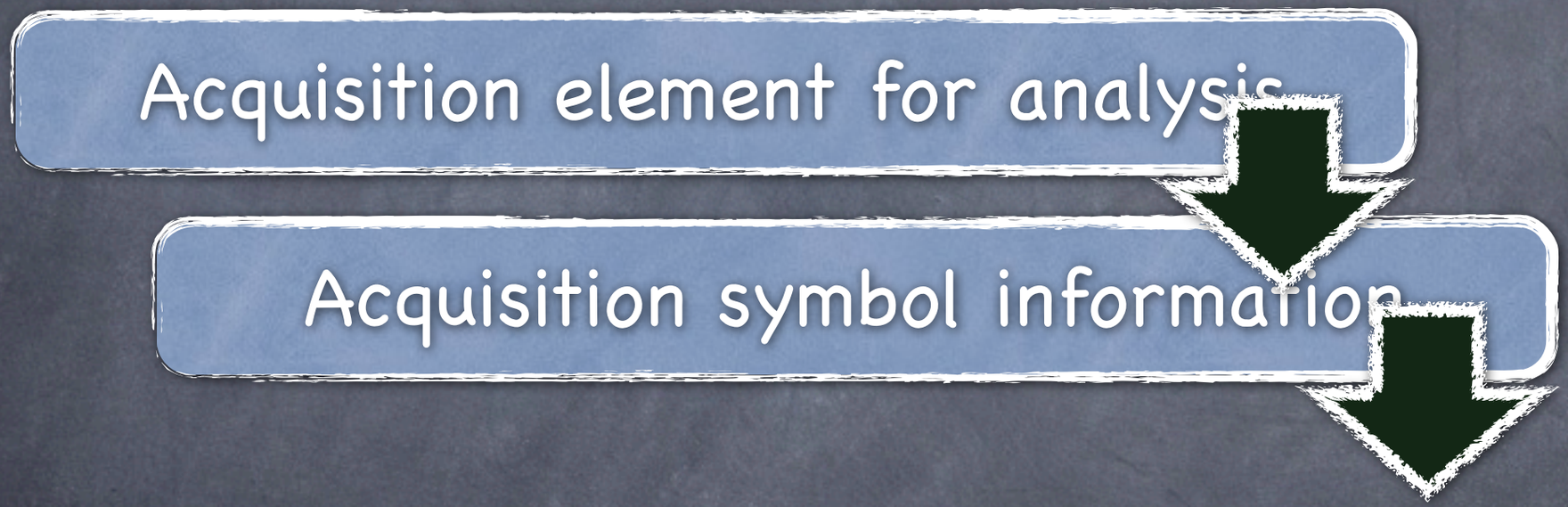
Process

Acquisition element for analysis

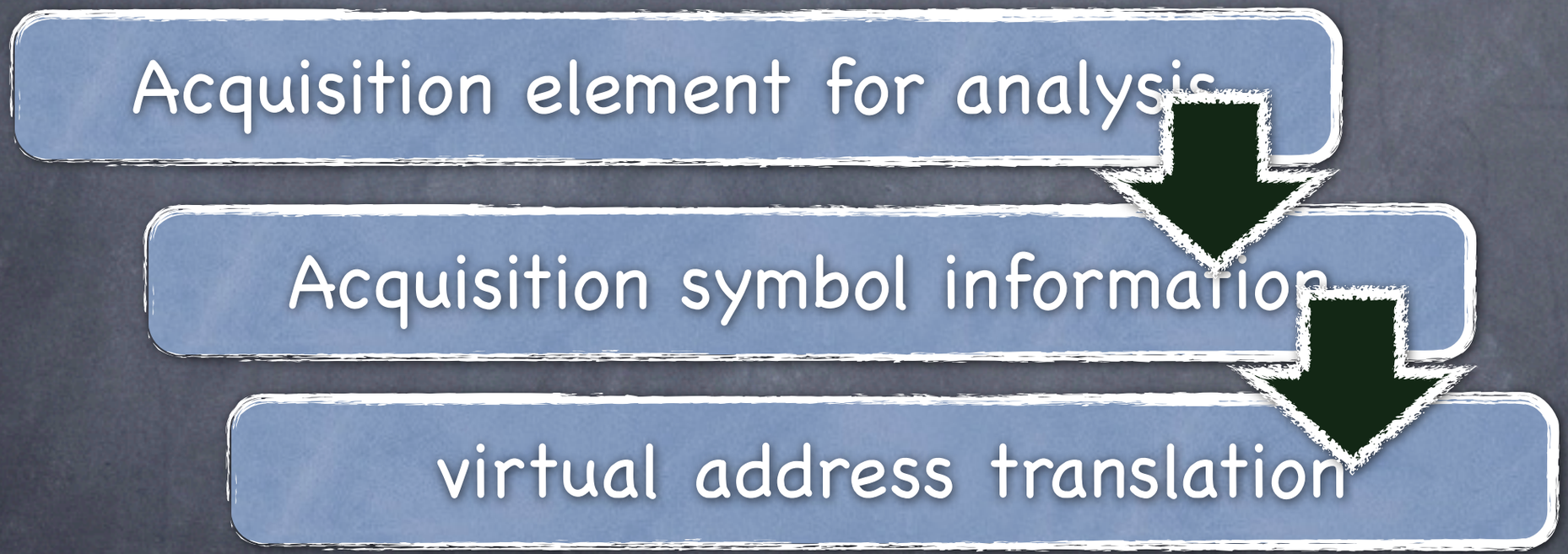
Process


Acquisition element for analysis
Acquisition symbol information

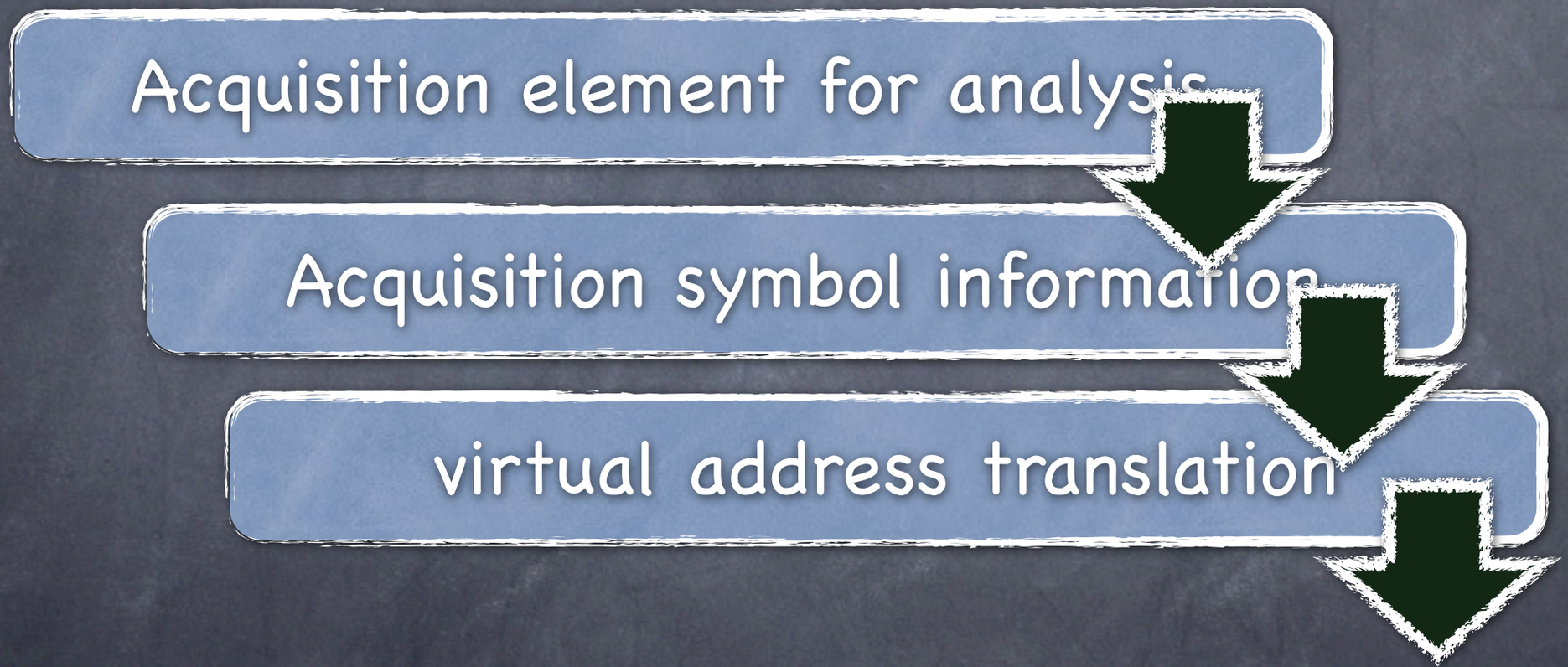
Process


Acquisition element for analysis
Acquisition symbol information

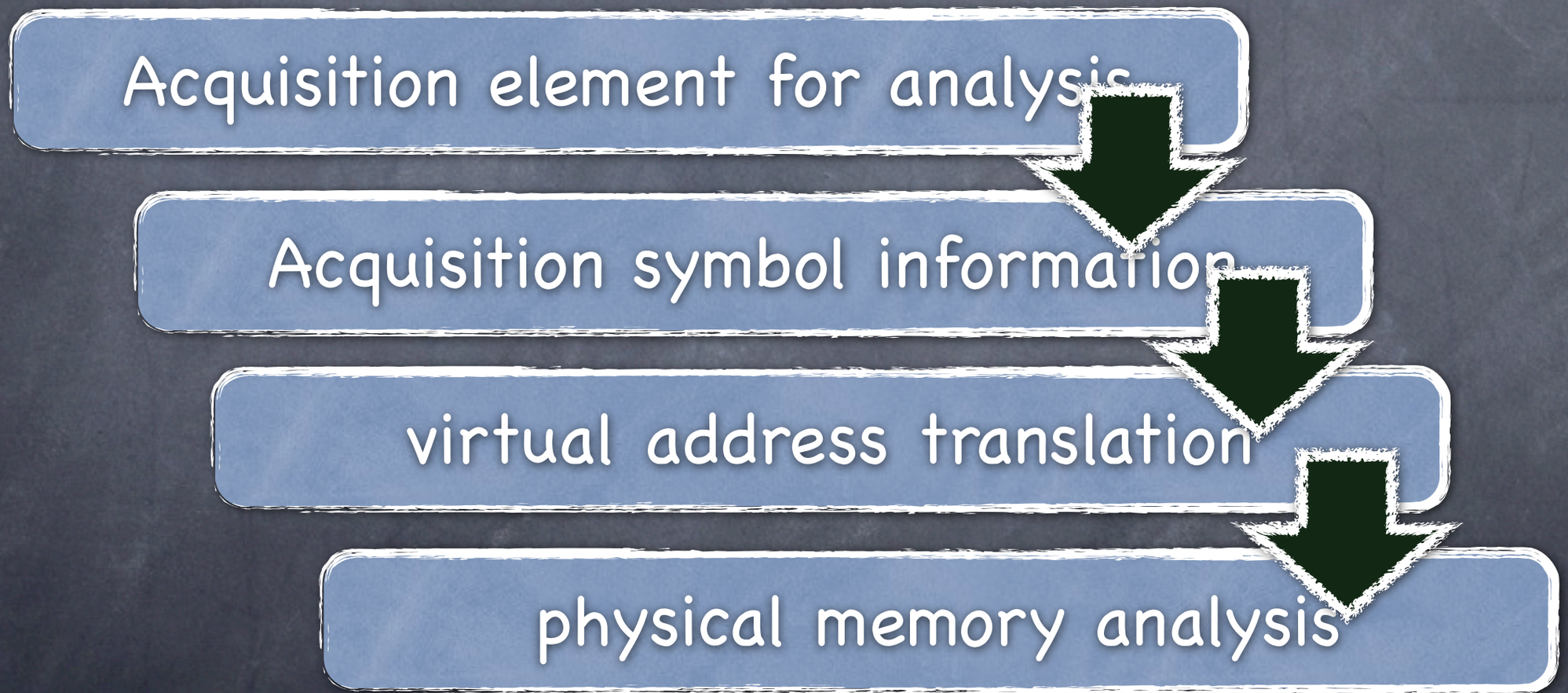
Process


Acquisition element for analysis
Acquisition symbol information
virtual address translation

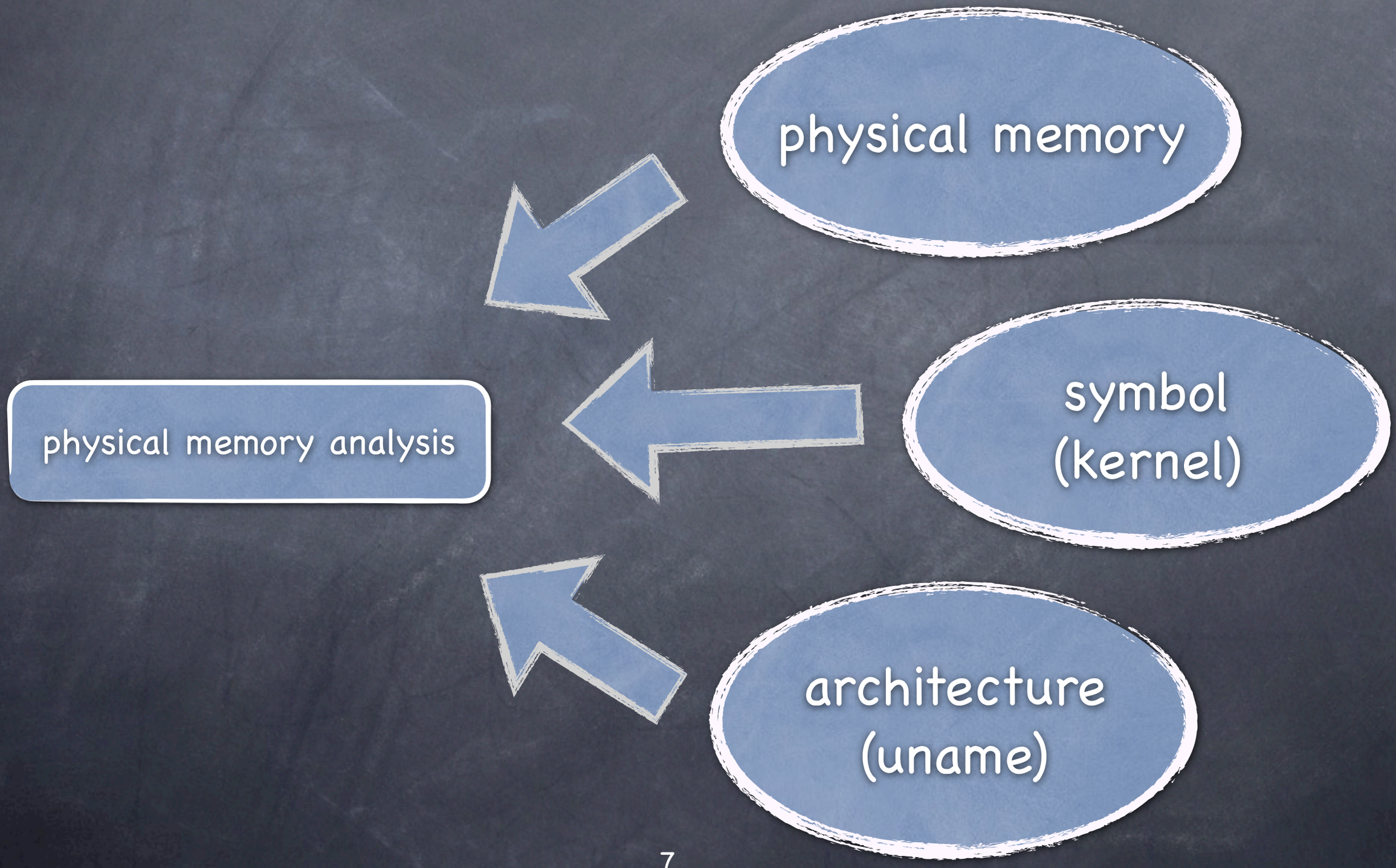
Process


Acquisition element for analysis
Acquisition symbol information
virtual address translation

Process


Acquisition element for analysis
Acquisition symbol information
virtual address translation
physical memory analysis

element for analysis



Physical Memory Acquisition

- Integrity(Physical Memory, HDD)
- Very Fast Speed
  - USB 2.0 - 480Mbit/s, FW800 - 800Mbit/s

Physical Memory Acquisition

```
import sys, time
import fw

BLOCKSIZE = 1024*1024*4
devices = fw.scanbus()

def format_guid(i):
    return ':'.join(["".join(x) for x in zip(("%016x" % i)[::2], ("%016x" % i)[1::2])])

start = 0x00000000L
end =   0x100000000L

for device in devices:
    print "Found device %s" % (format_guid(device.guid))
    fd = open("%016x-%08x-%08x.memdump" % (device.guid, start, end), "w")
    pos = start
    while pos < end:
        print "\r-> reading %08x ..." % (pos),
        fd.write(device.read(pos, BLOCKSIZE))
        print hex(device.lastResultCode*1L),
        pos += BLOCKSIZE
        sys.stdout.flush()
    fd.close()
```

Physical Memory Acquisition

- /dev/mem (dummy), /dev/kmem
- load KEXT --> disk, memory integrity (x)

Physical Memory Acquisition

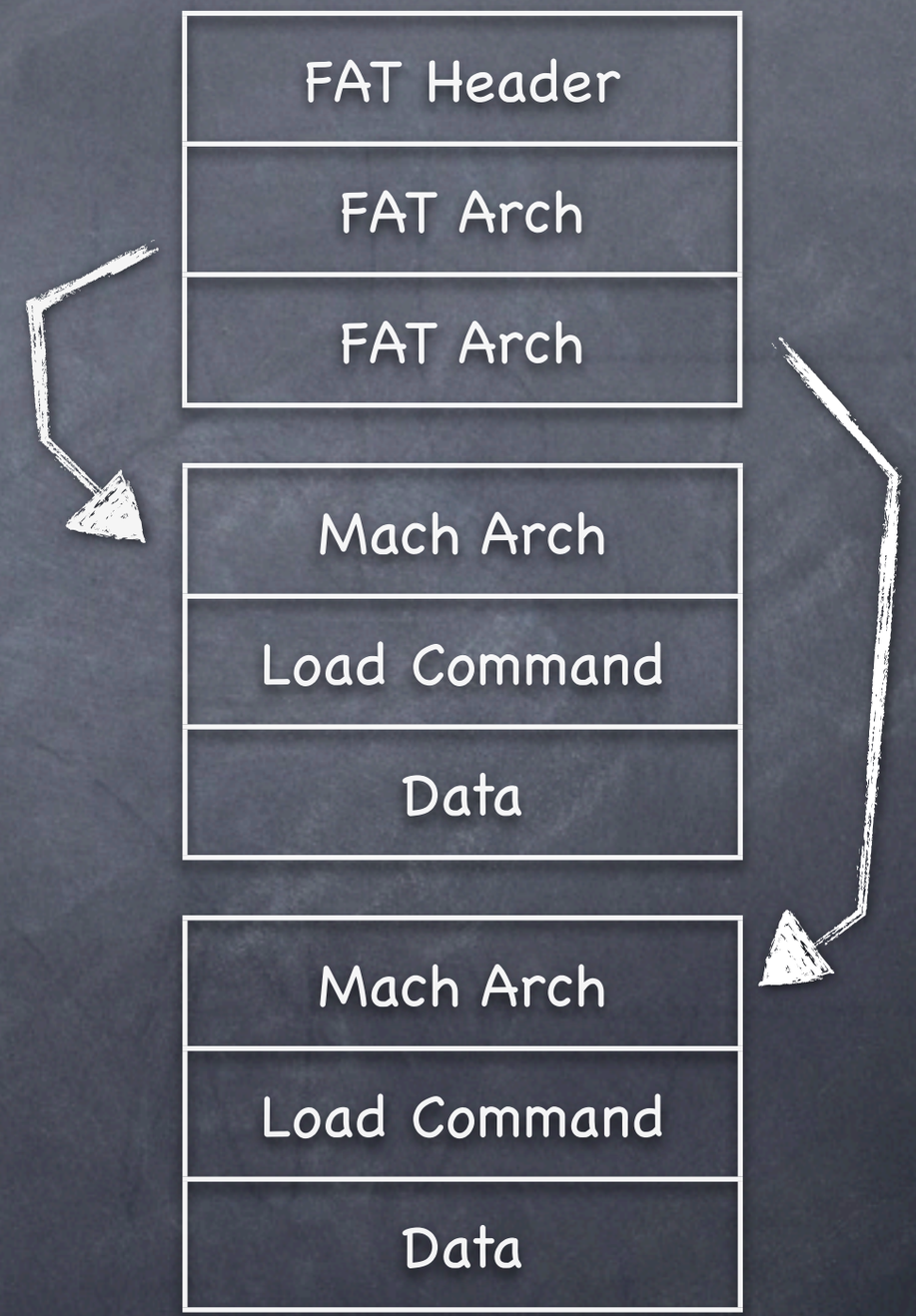
Hibernation

- sleepimage
  - path: /var/vm/
- Encryption (Unknown)

Symbol Acquisition

- kernel Image
  - Universal Binaries
  - Mach-O file format

```
In archive mach_kernel:

mach_kernel:i386:x86-64:     file format mach-o-le
mach_kernel:i386:x86-64


mach_kernel:i386:     file format mach-o-i386
mach_kernel:i386

mach_kernel:powerpc:common:     file format mach-o-be
mach_kernel:powerpc:common
```



Symbol Acquisition

Mach Header

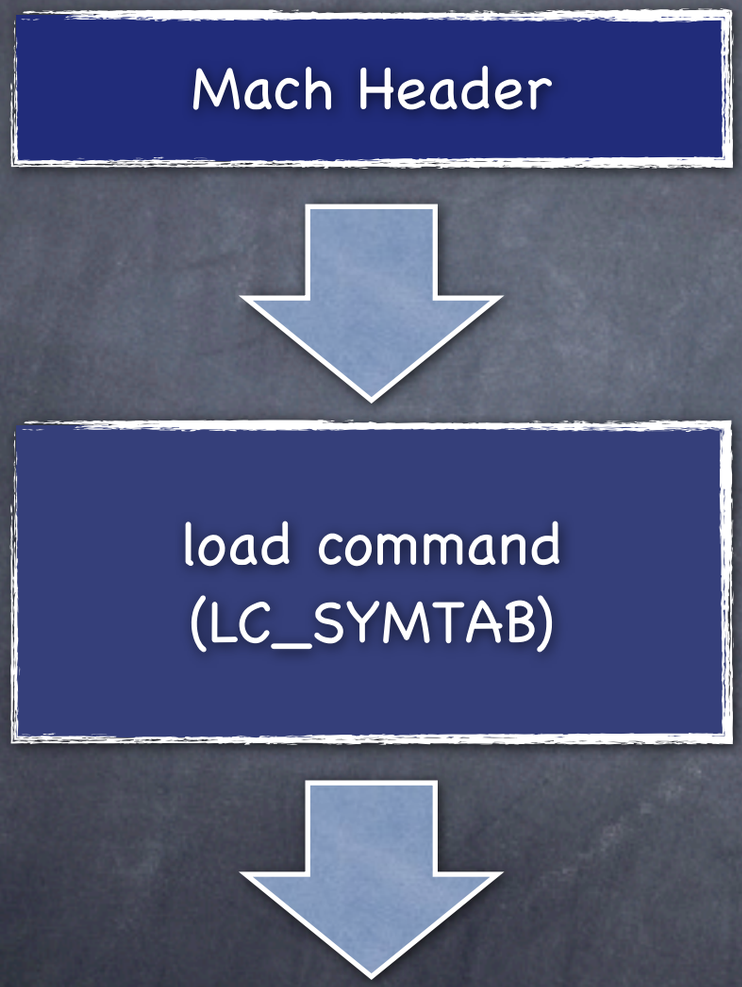
Symbol Acquisition

Symbol Acquisition

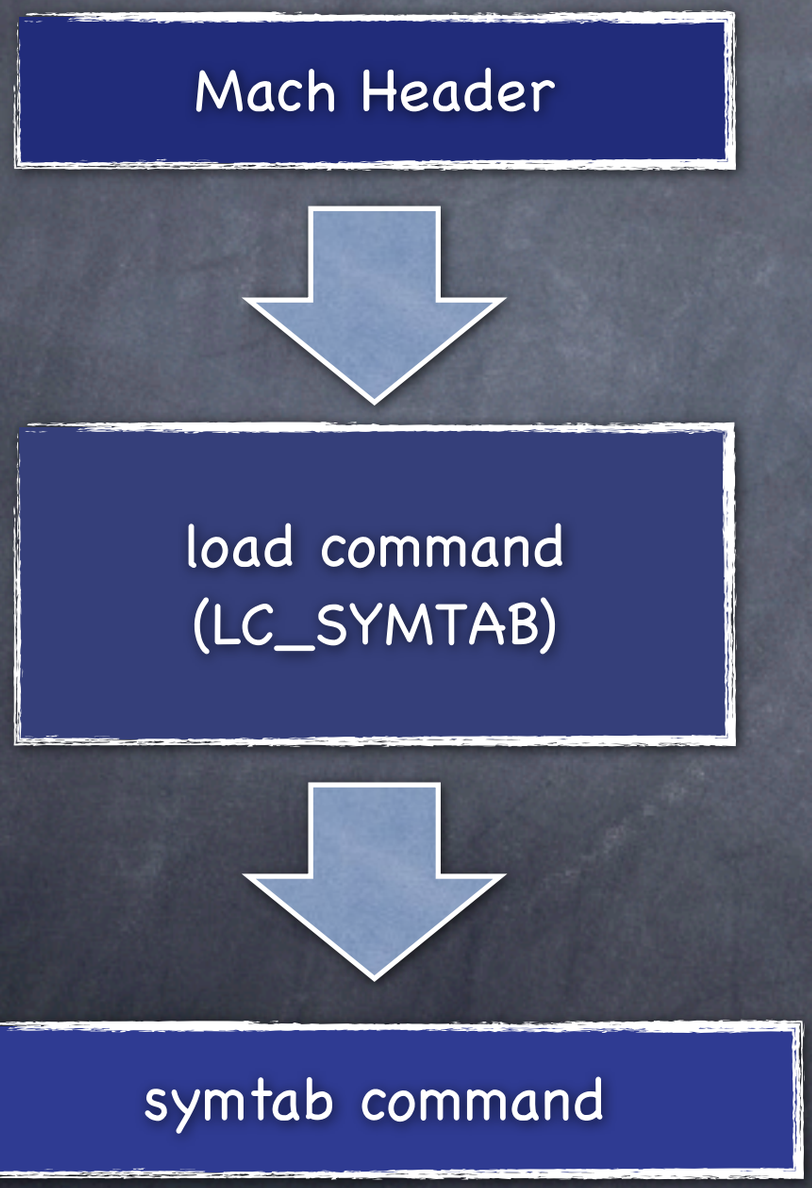
Mach Header

load command
(LC_SYMTAB)

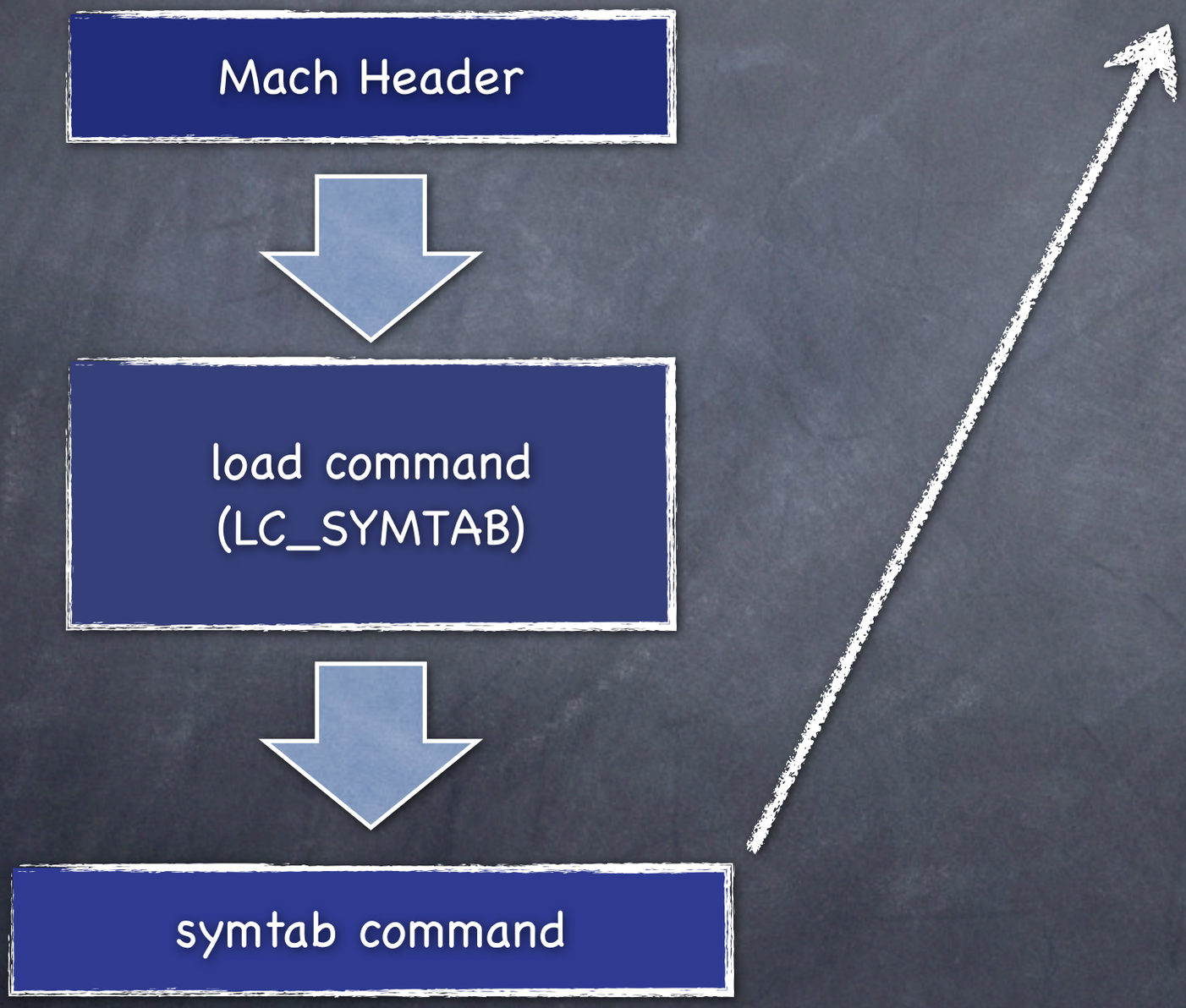
Symbol Acquisition



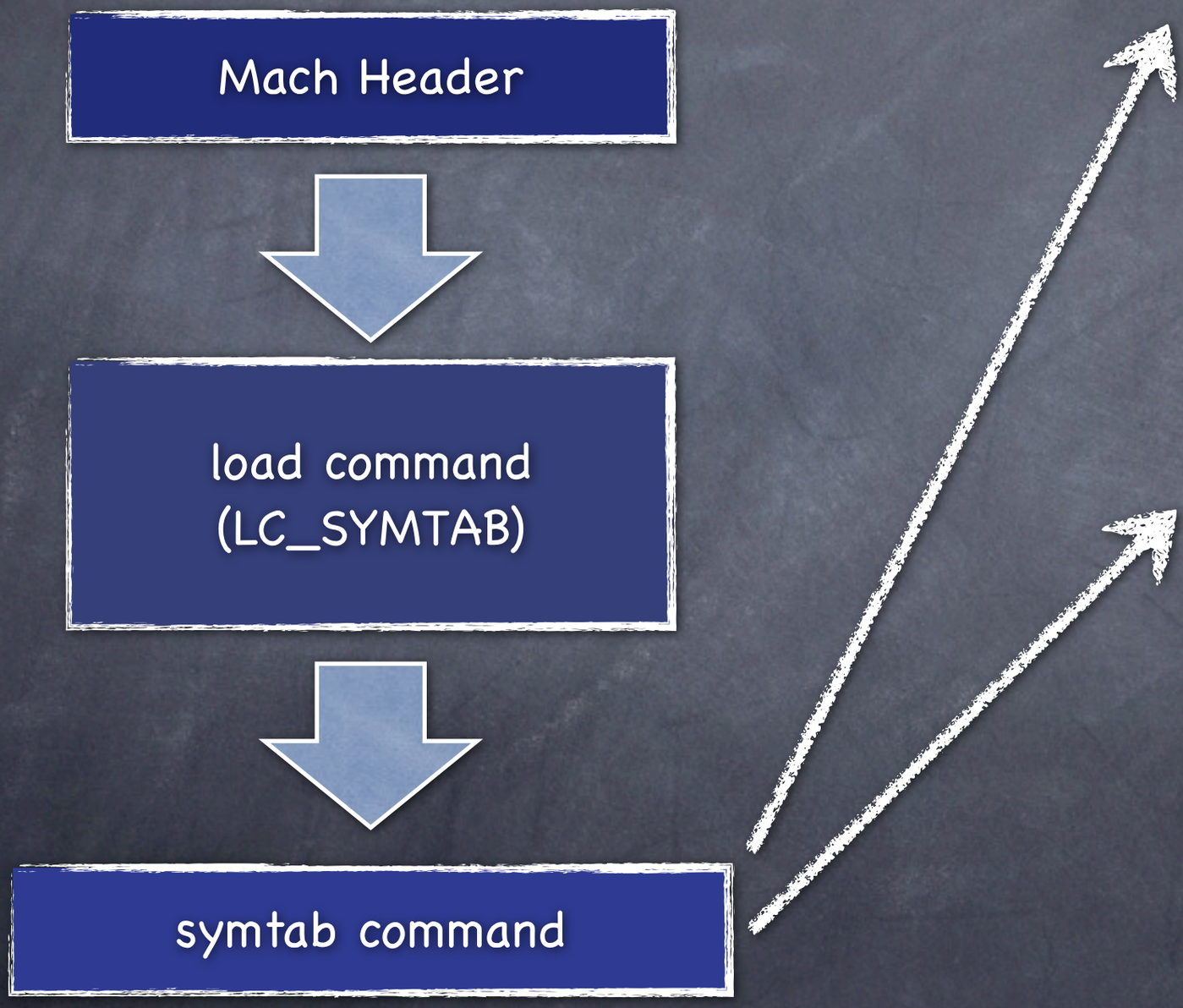
Symbol Acquisition



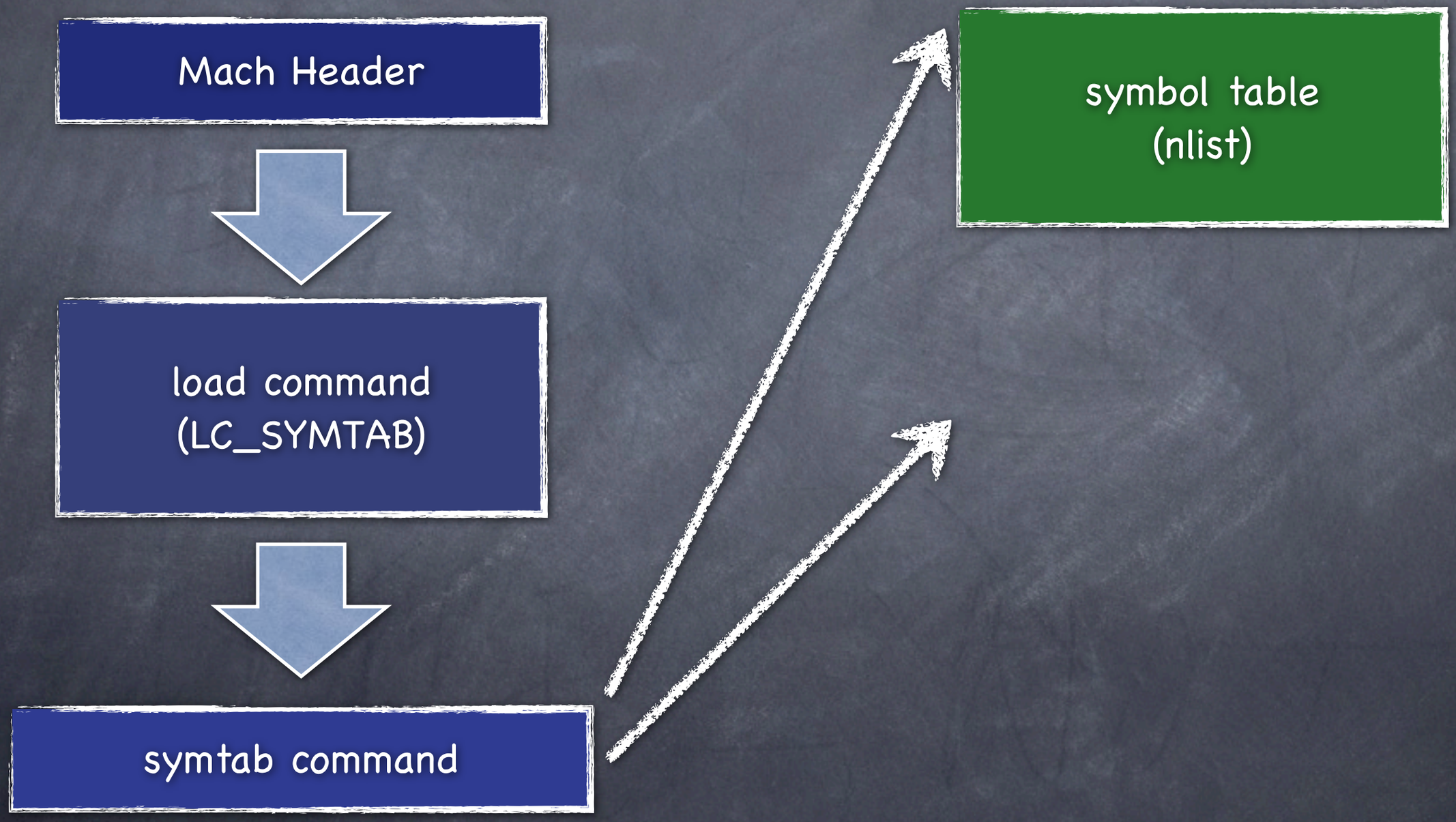
Symbol Acquisition



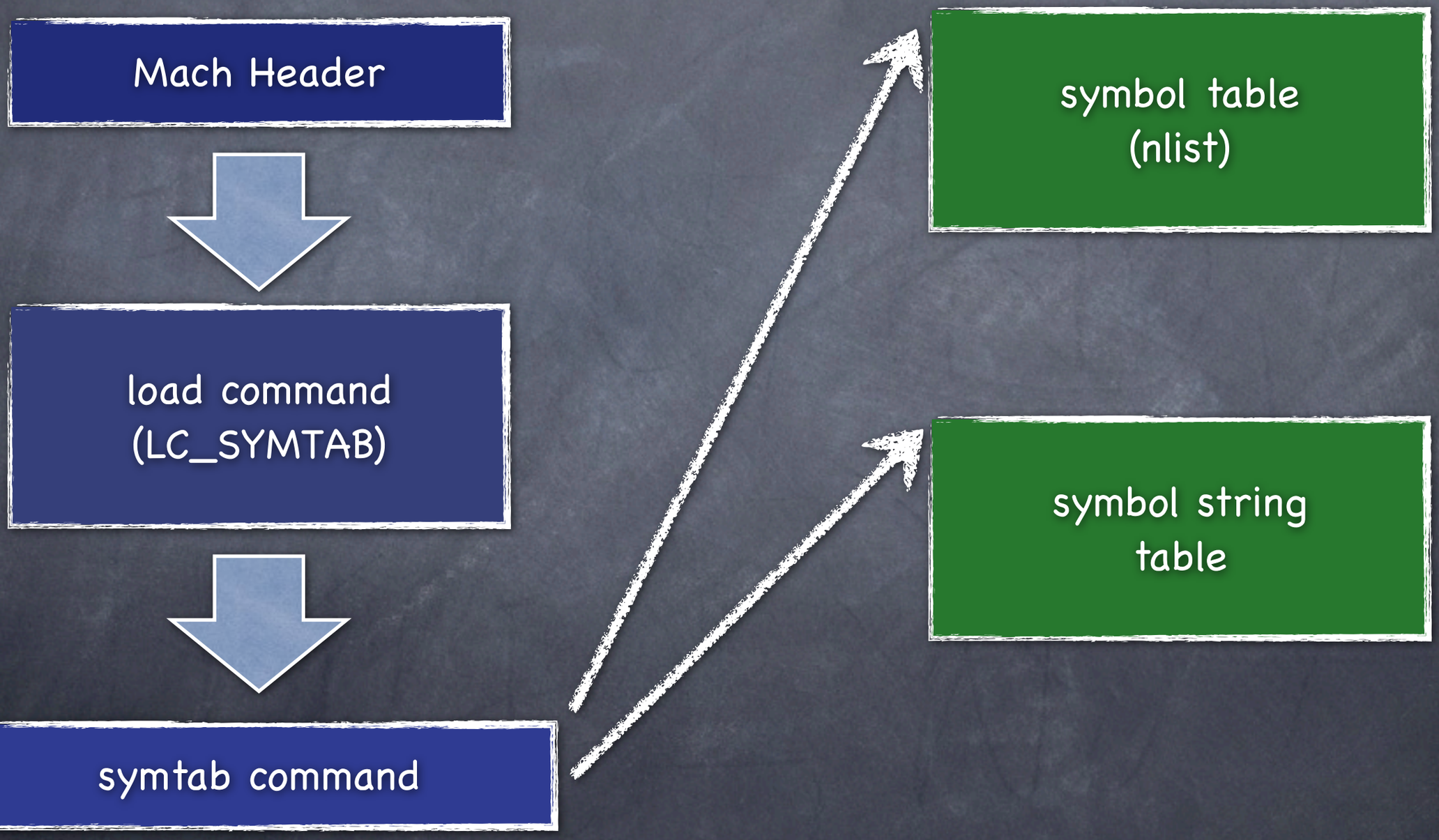
Symbol Acquisition



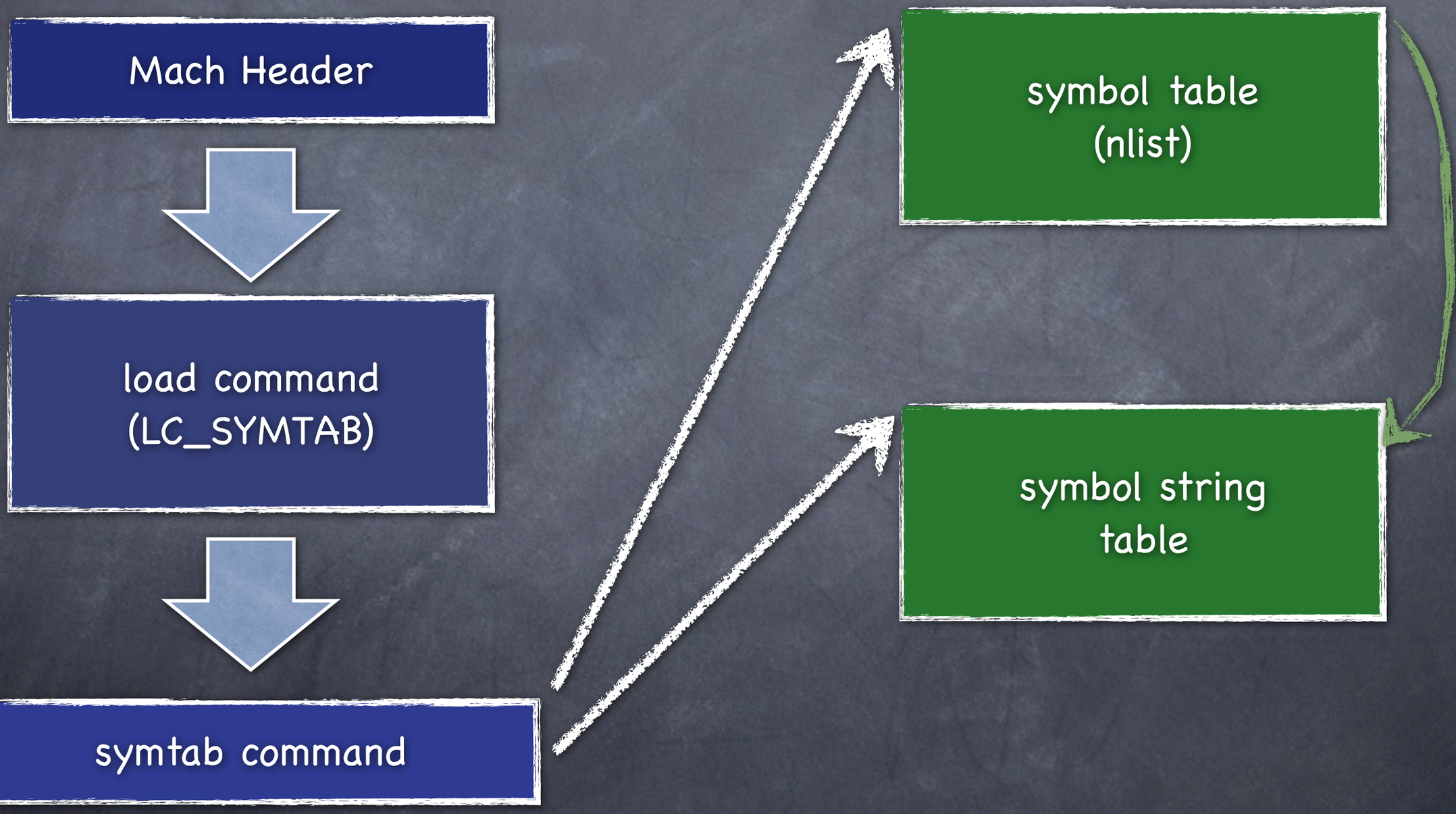
Symbol Acquisition



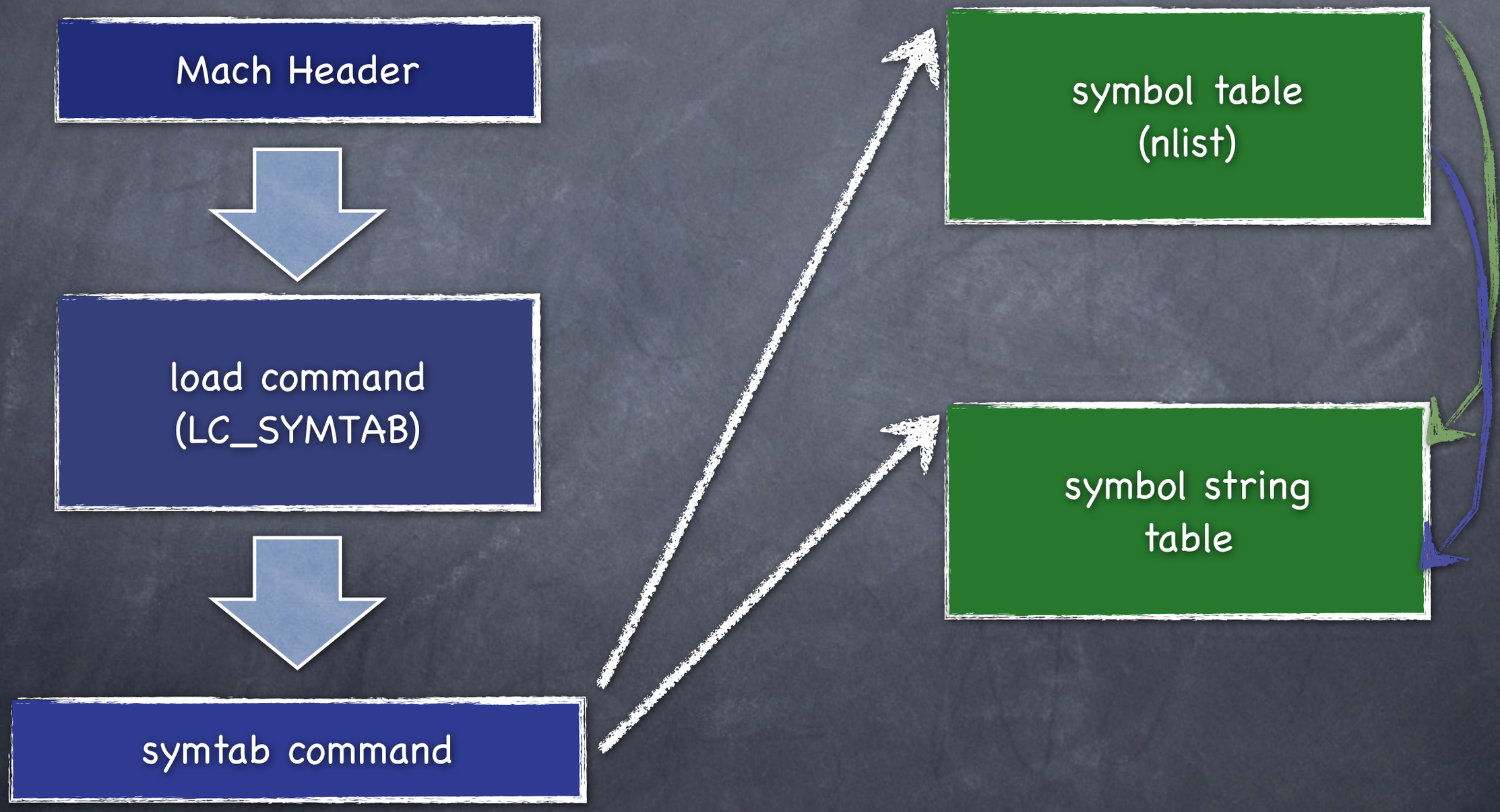
Symbol Acquisition



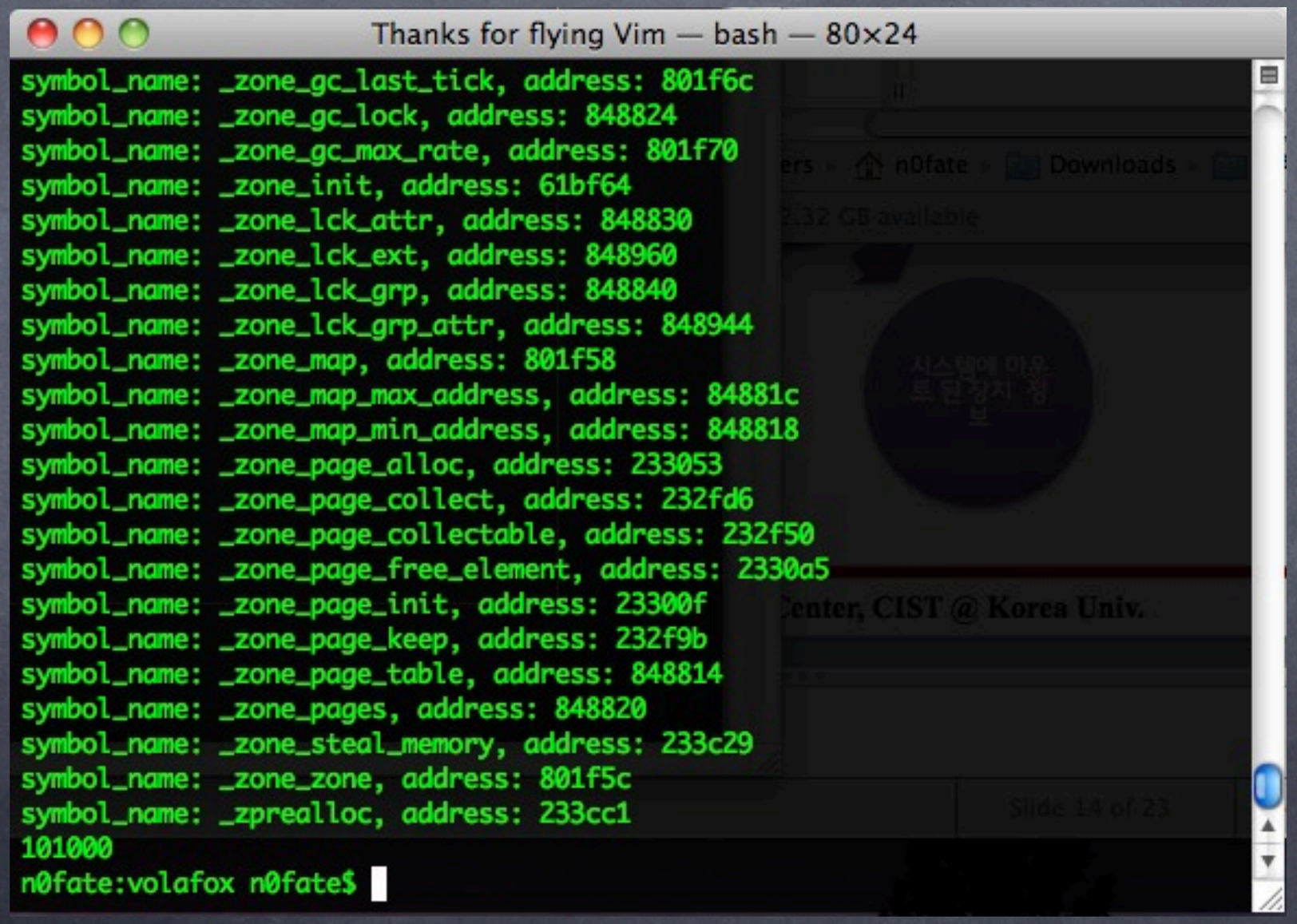
Symbol Acquisition



Symbol Acquisition



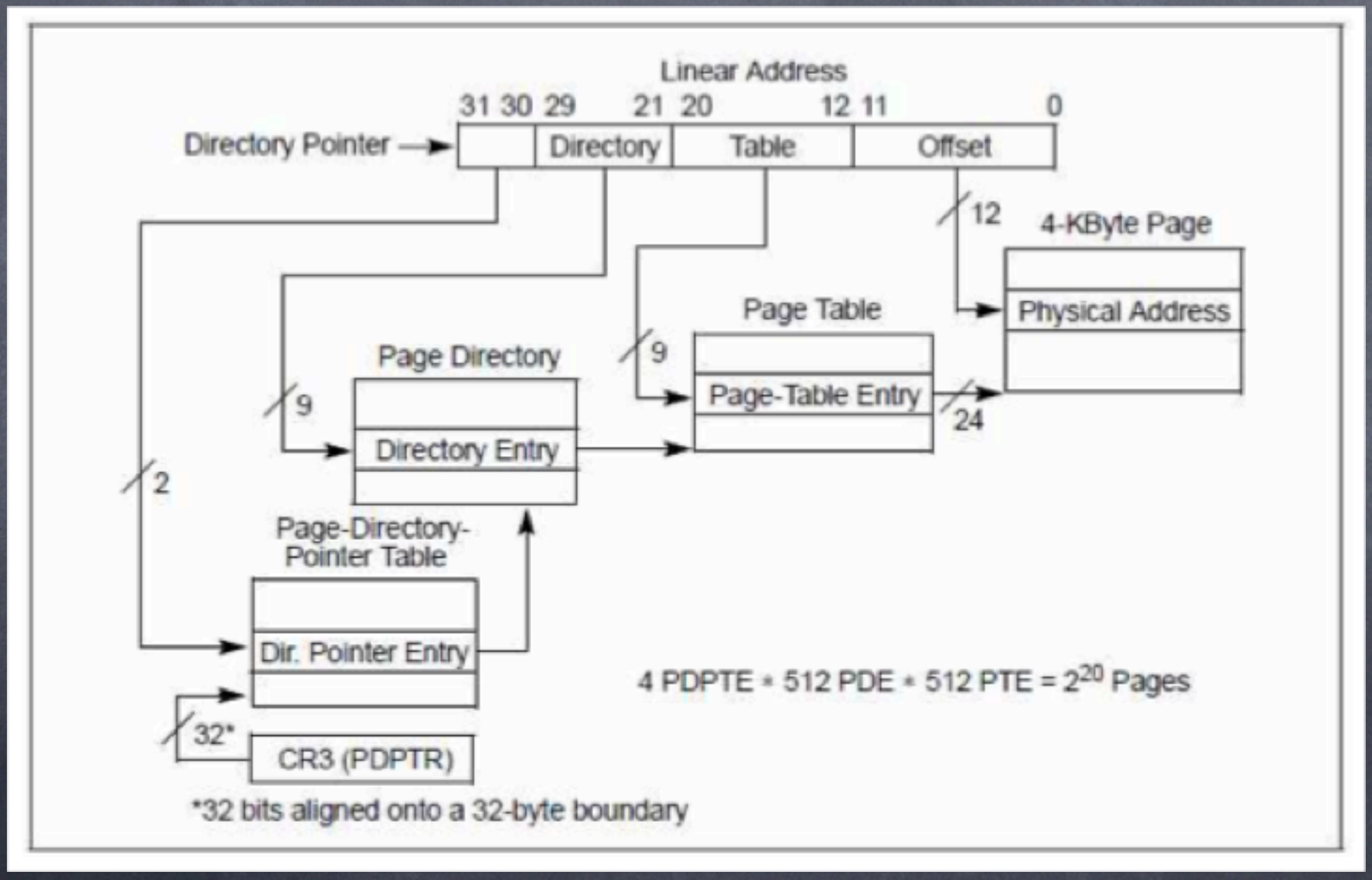
Symbol Acquisition



Virtual Address Translation

- Technique
  - PAE (Physical Address Extensions)
    - PML4 (Page Map Level 4)
- Store kernel symbol to translate VA
  - IdlePDPT, IdlePML4

Virtual Address Translation



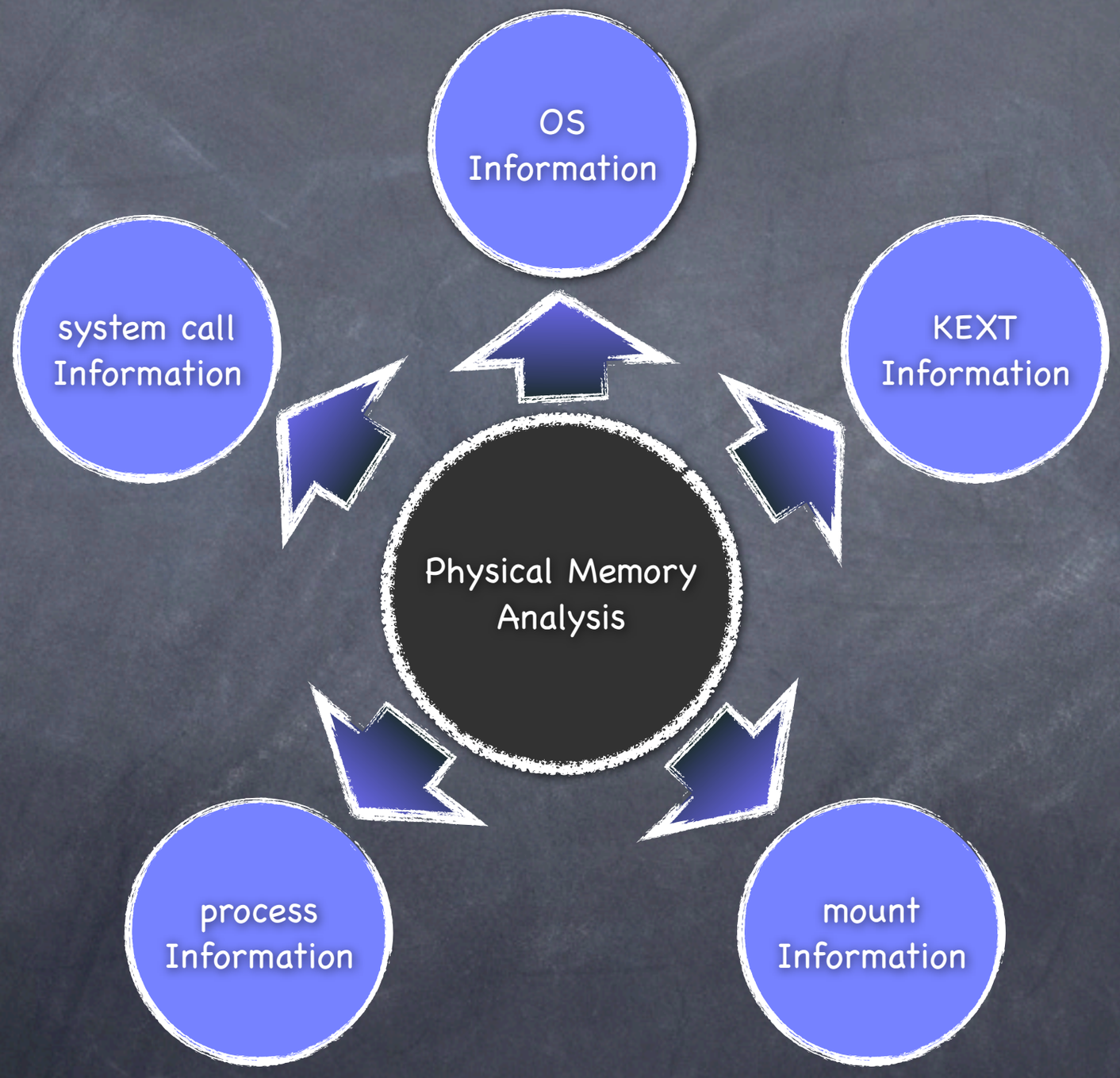
Physical Memory Analysis

- Symbol
  - select kernel symbol to analysis
- Structure
  - http://opensource.apple.com

volafox

- Memory Analyzer for OS X
  - Code: Python 2.5
  - Support OS: everything(?)
- Module
  - mach-o format analyzer
  - memory analyzer
  - Virtual Address Translation (Volatility Framework)

volafox

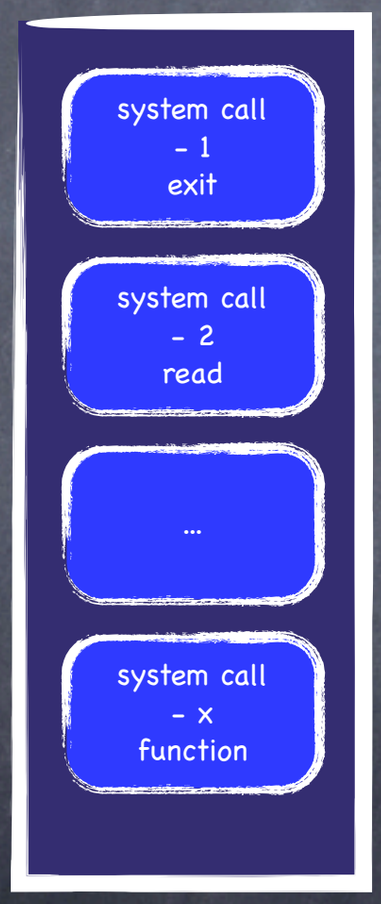

OS
Information
system call
Information
KEXT
Information
Physical Memory
Analysis
process
Information
mount
Information

Demo

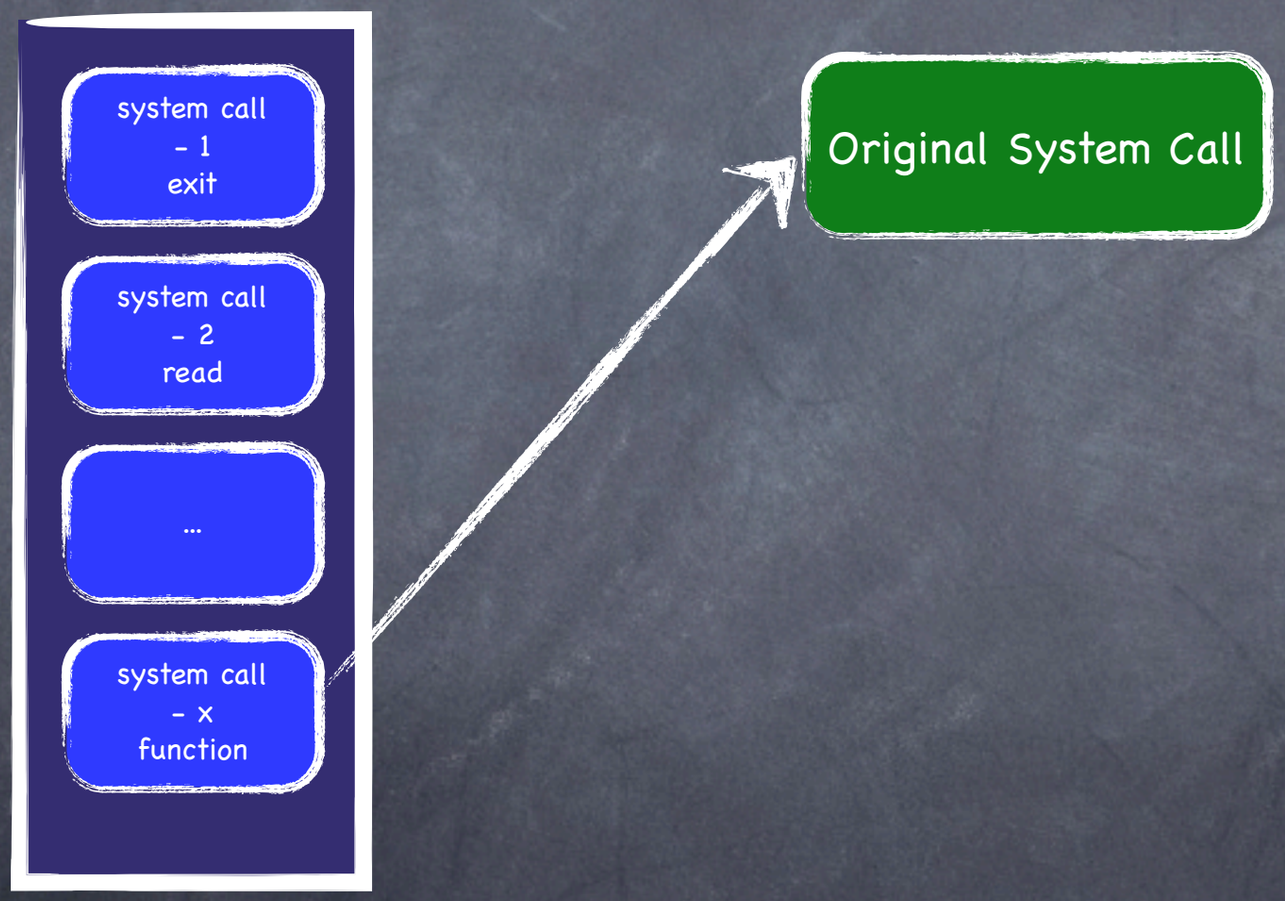
Terminal — bash — 80×24

```
n0fate:volafox n0fate$ python volafox.py
Memory analyzer for OS X 0.5 - n0fate
Contact: rapfer@gmail.com
usage: python volafox.py -i MEMORY_IMAGE -s KERNEL_IMAGE -o INFORMATION

-= CAUTION =-
this program need to physical memory image, kernel image(mach_kernel)
and it support to Intel x86 Architecture only :(

INFORMATION:
os_version          Dawin kernel detail version
machine_info        Kernel version, cpu, memory information
mount_info          Mount information
kern_kext_info      Kernel KEXT(Kernel Extensions) information
kext_info           KEXT(Kernel Extensions) information
proc_info           Process list
syscall_info        Kernel systemcall information
n0fate:volafox n0fate$
```

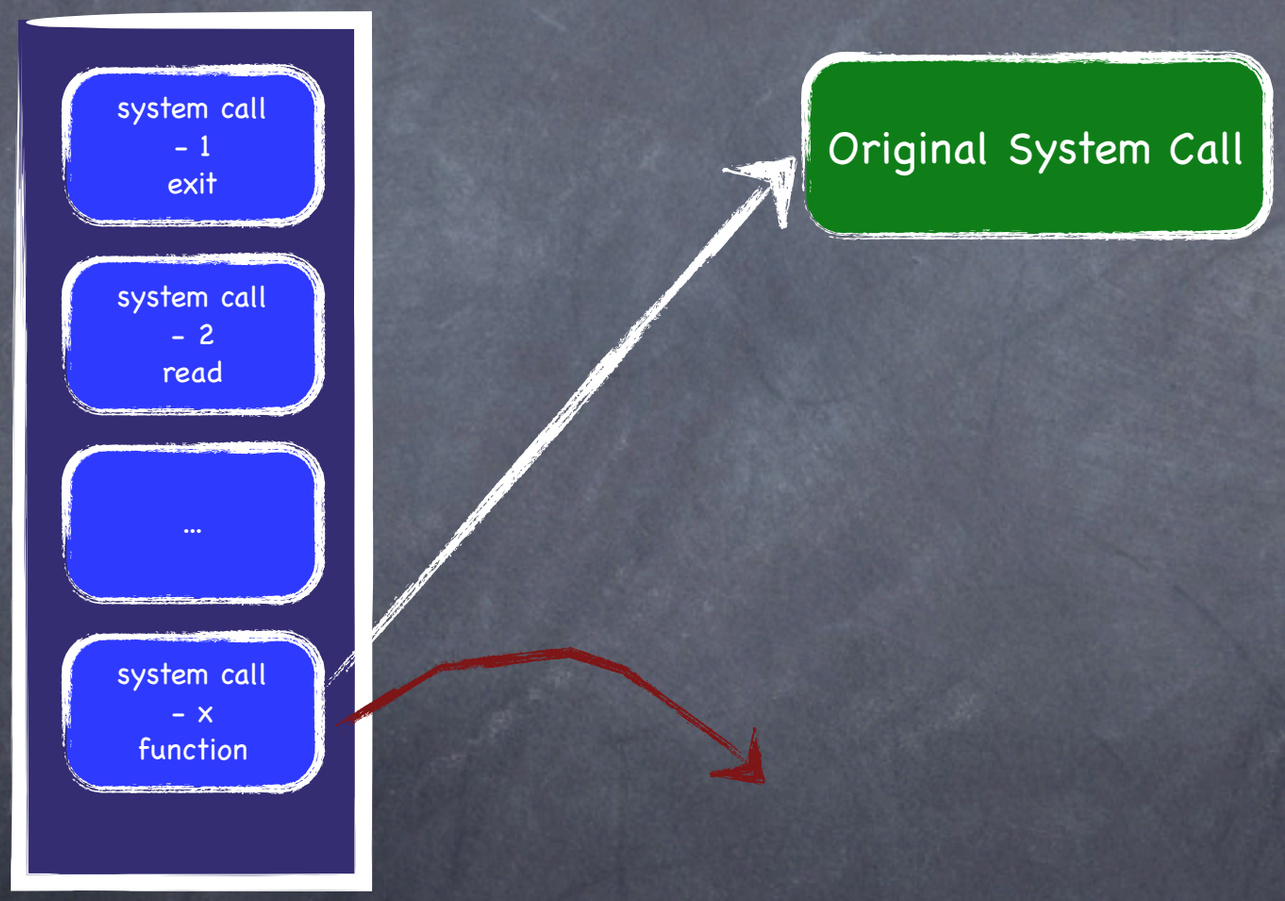
System Call Hooking



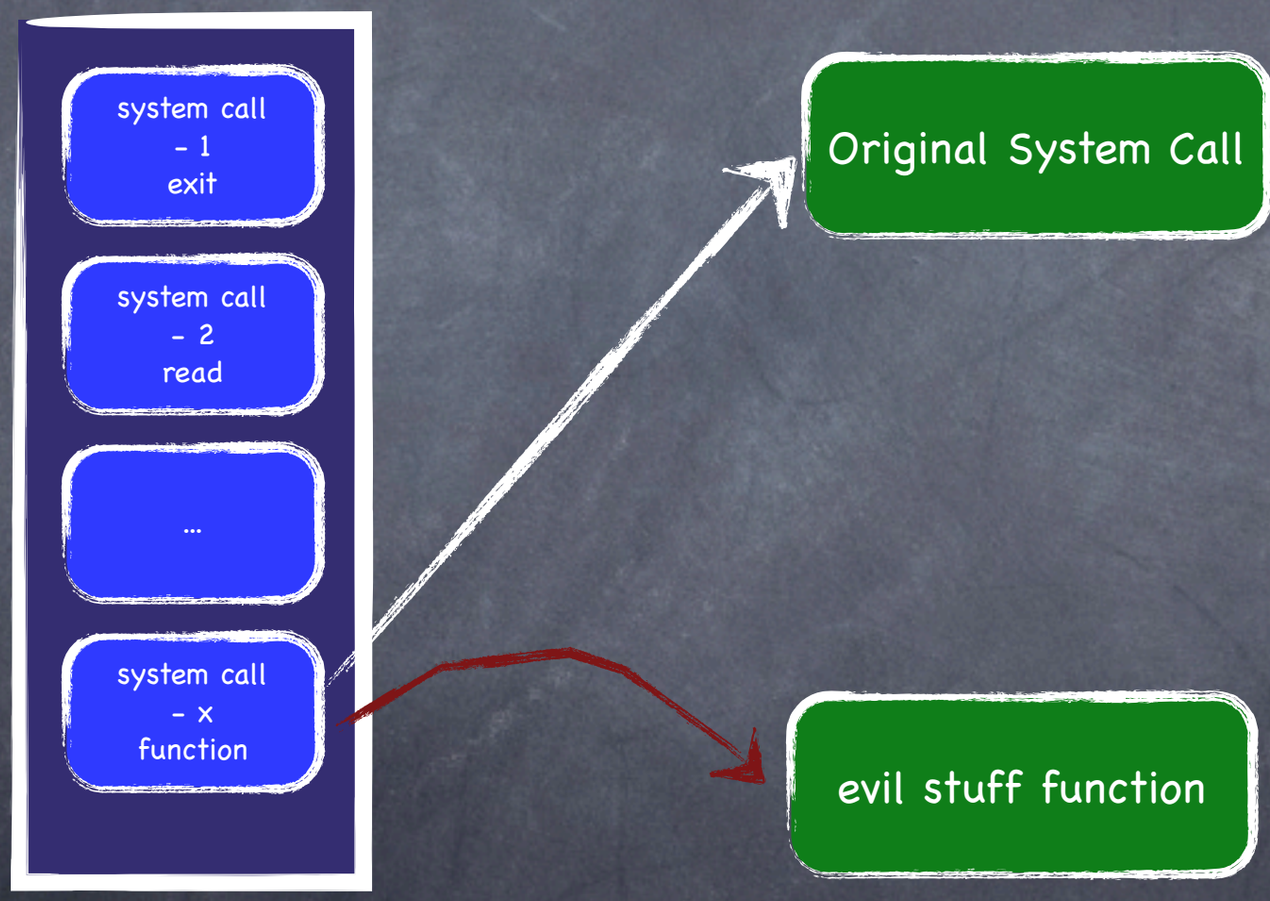
System Call Hooking



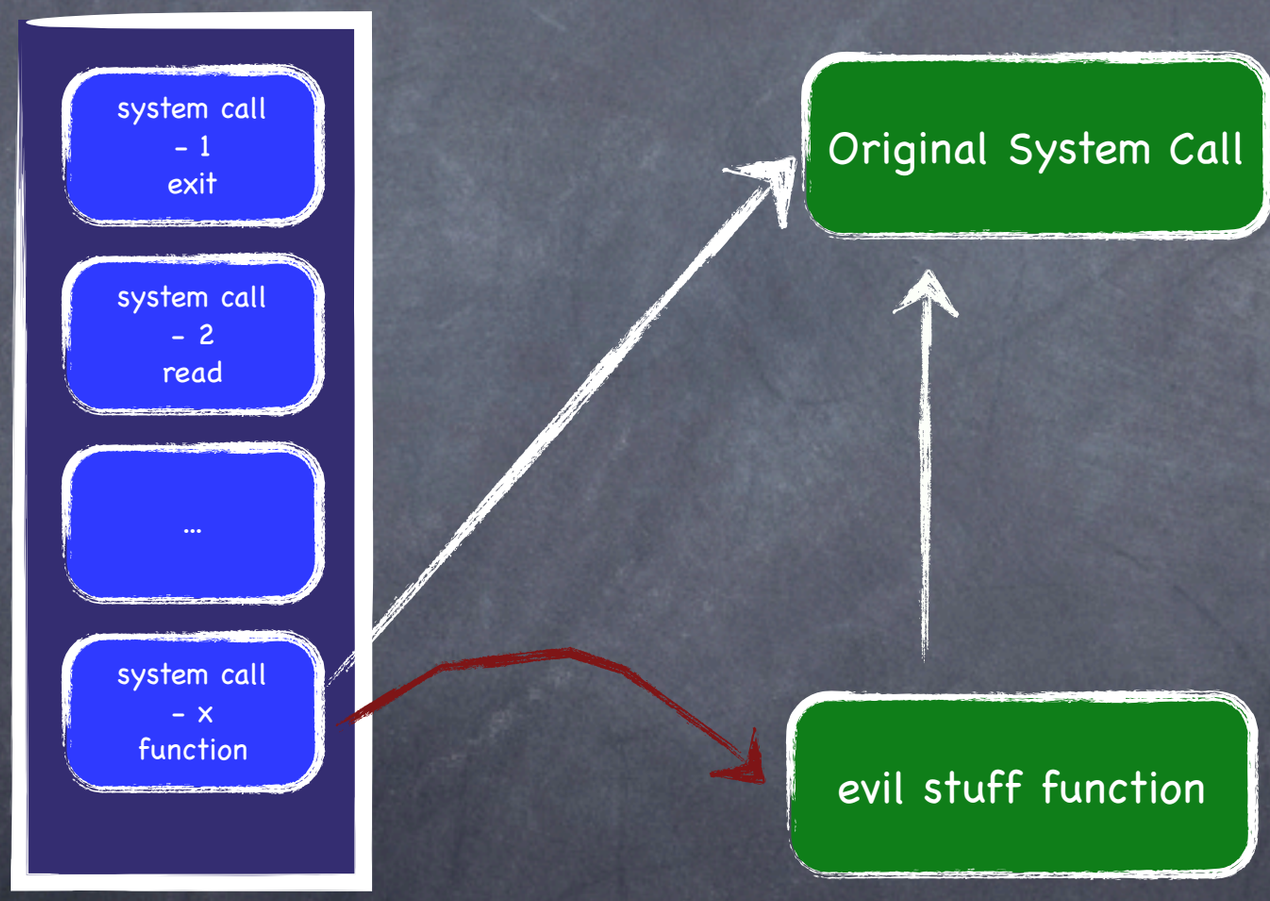
System Call Hooking



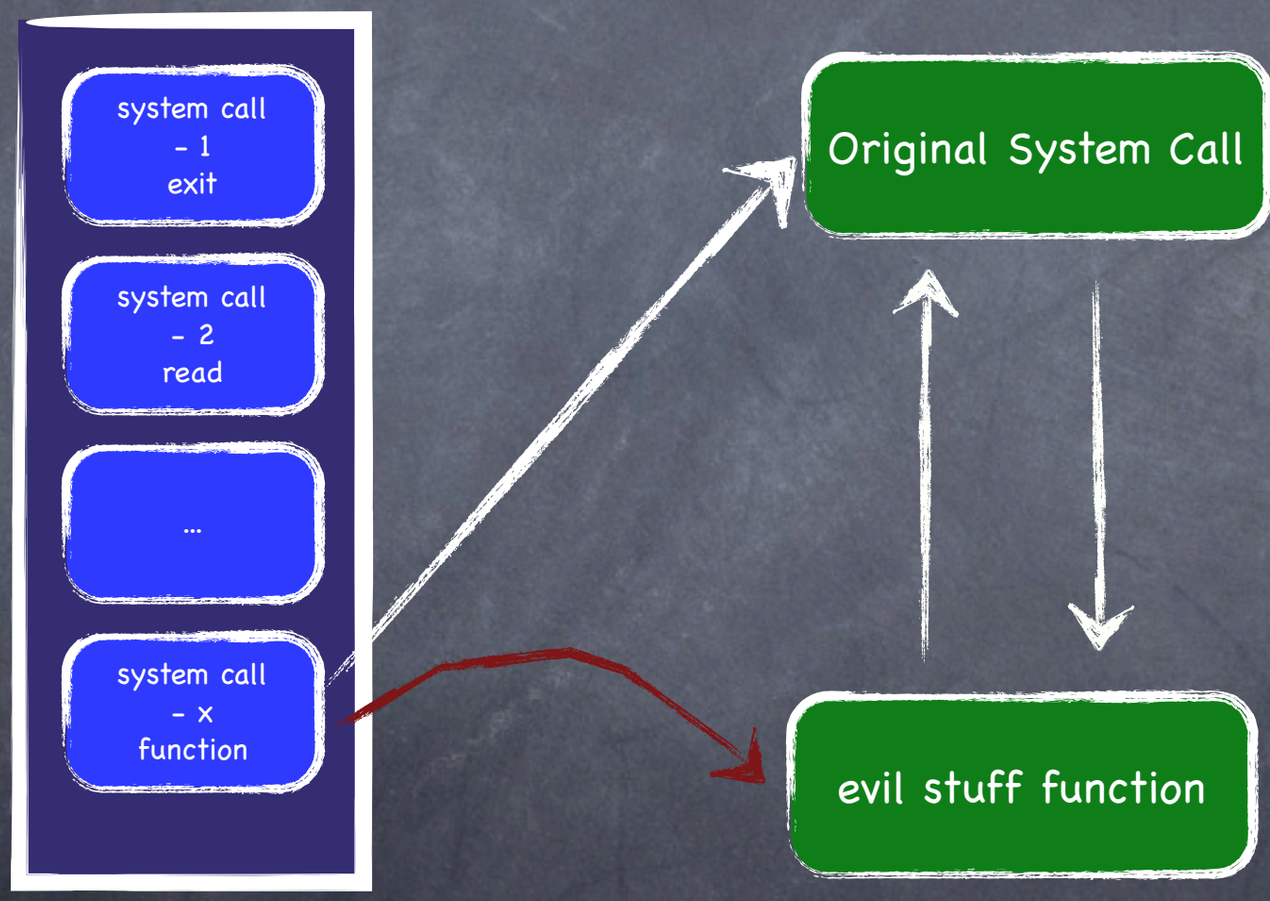
System Call Hooking



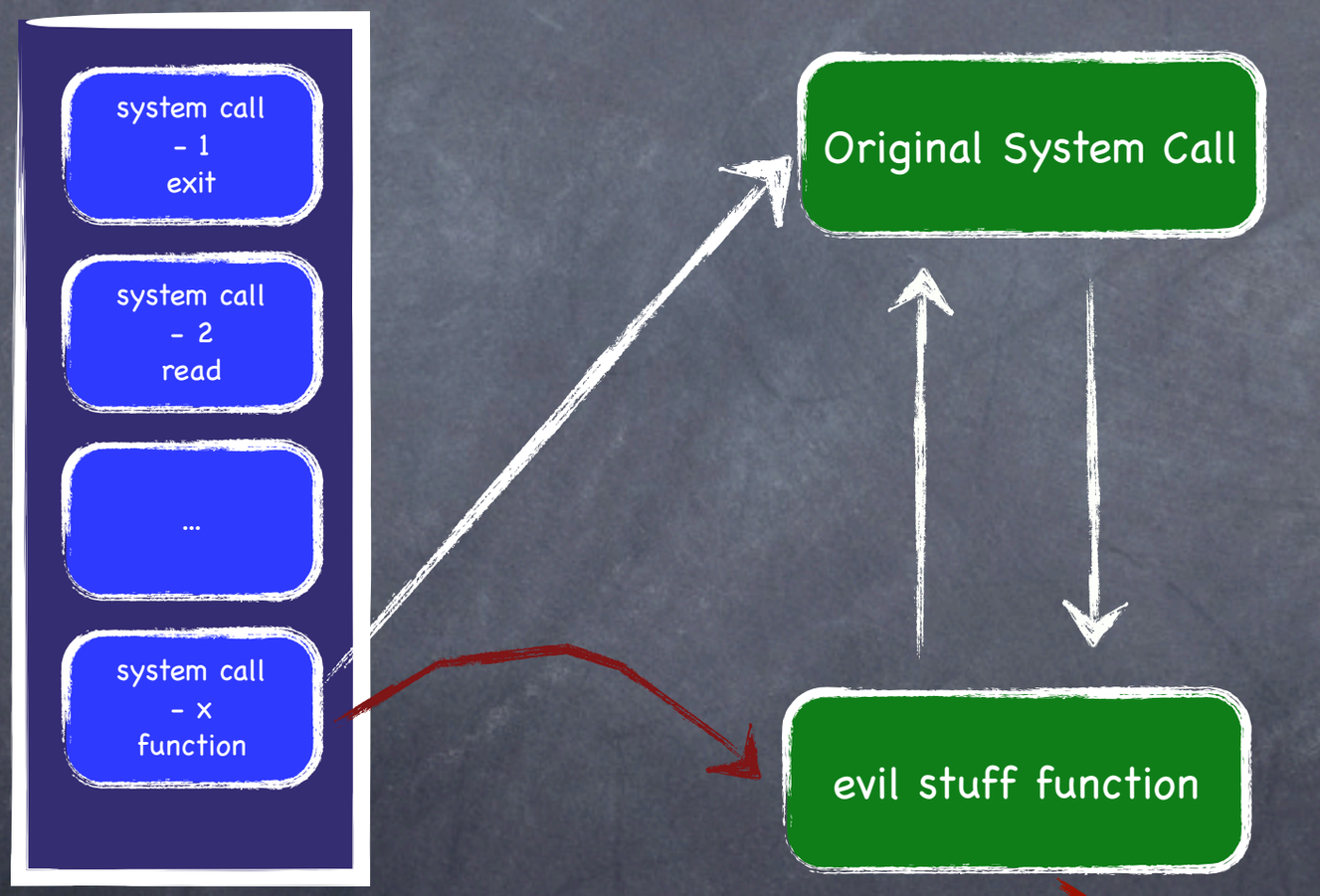
System Call Hooking



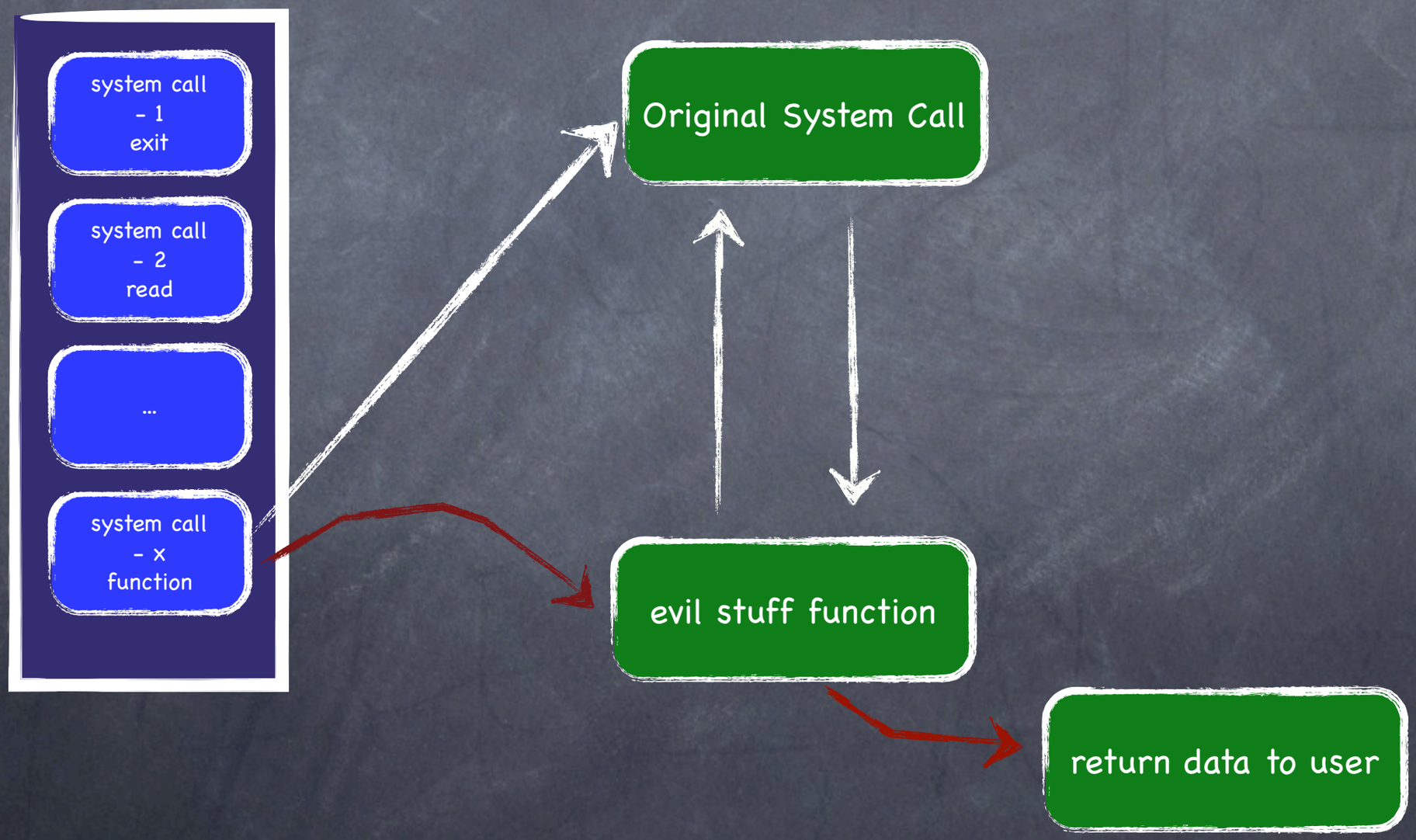
System Call Hooking



System Call Hooking



System Call Hooking



System Call Hooking

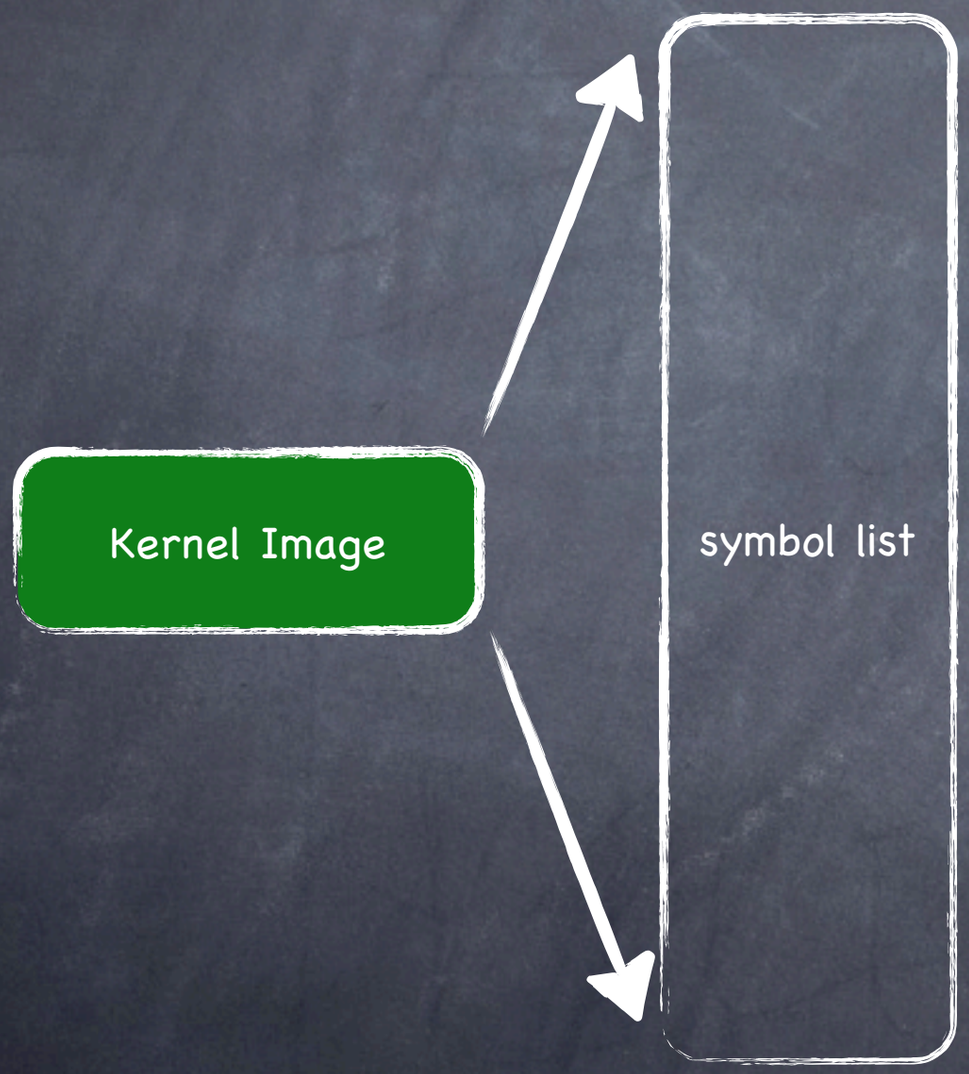
- Step
  - find `nsysent' symbol
  - Snow Leopard -> sysent = &nsysent - (nsysent * sizeof(sysent))
  - modifying system call function = sysent->sy_call = evil_stuff

```
struct sysent {
    int16_t sy_narg;
    int8_t reserved;
    int8_t sy_flags;
    sy_call_t *sy_call;
    sy_munge_t *sy_arg_munge32;
    sy_munge_t *sy_arg_munge64;
    int32_t sy_return_type;
    uint16_t sy_arg_bytes;
};
```

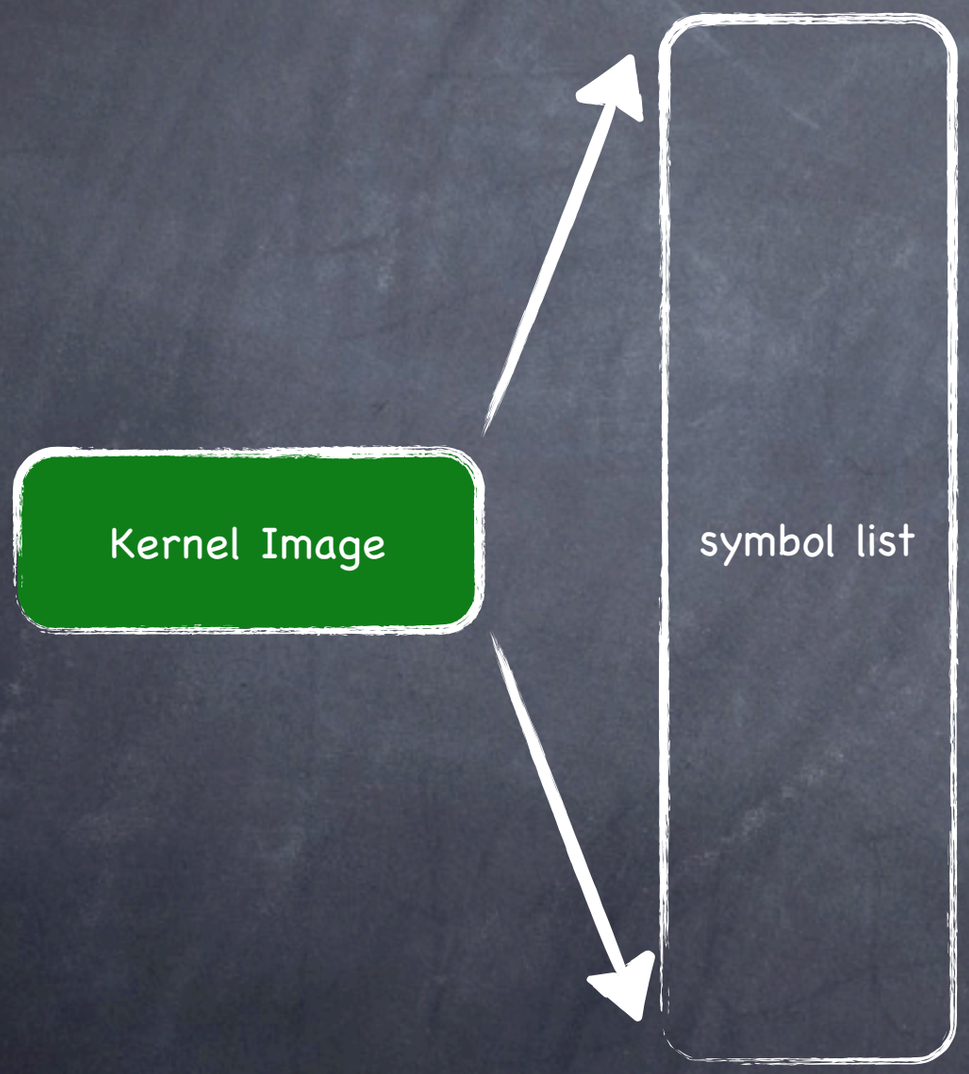
System Call Hooking

- Example) Hiding Files
- Hooking function
  - getdirentriesattr() - get file system attributes for multiple directory
  - getdirentries64() - read directory entries in a filesystem independent format

Detection

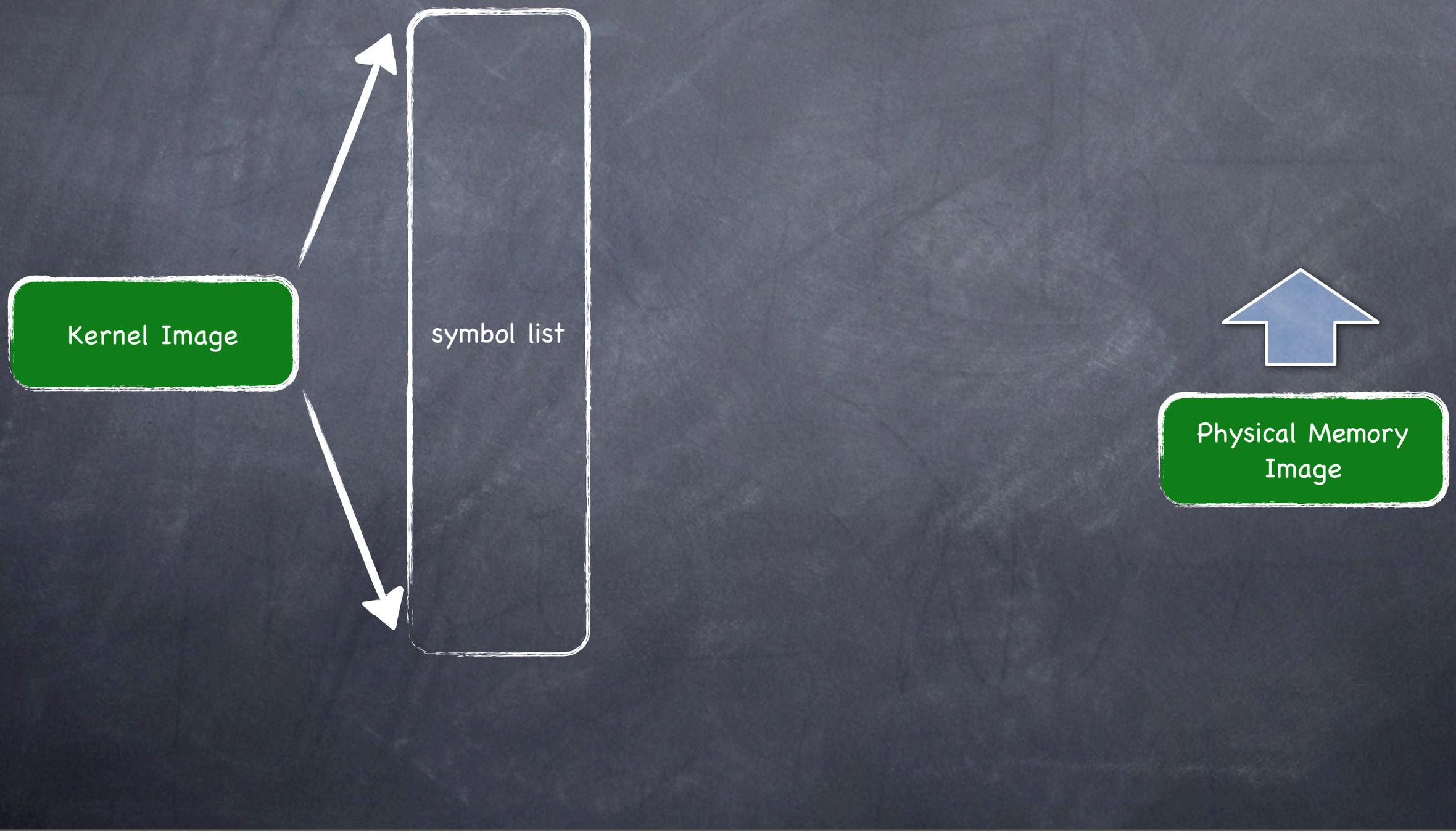


Detection

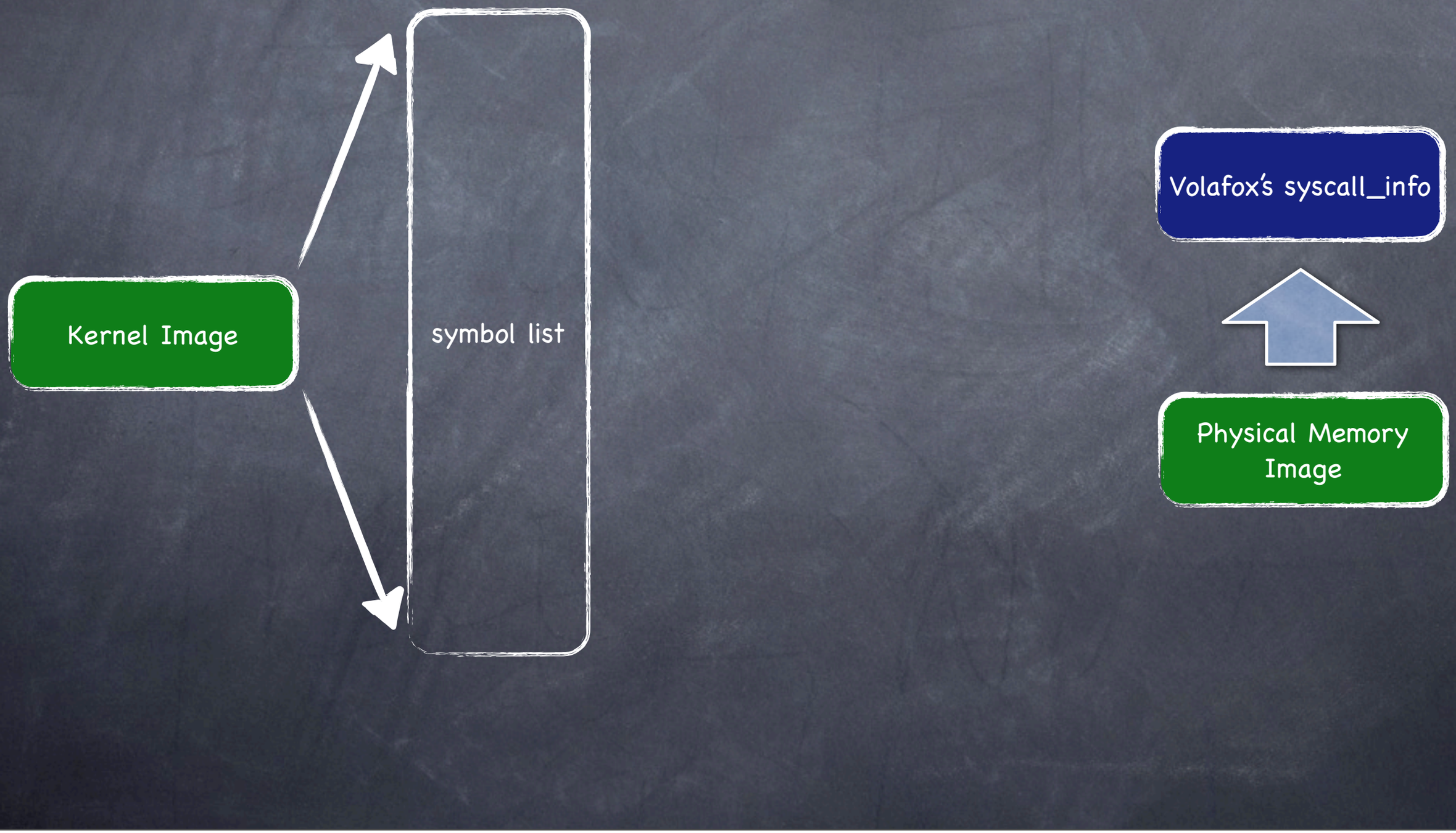


Physical Memory Image

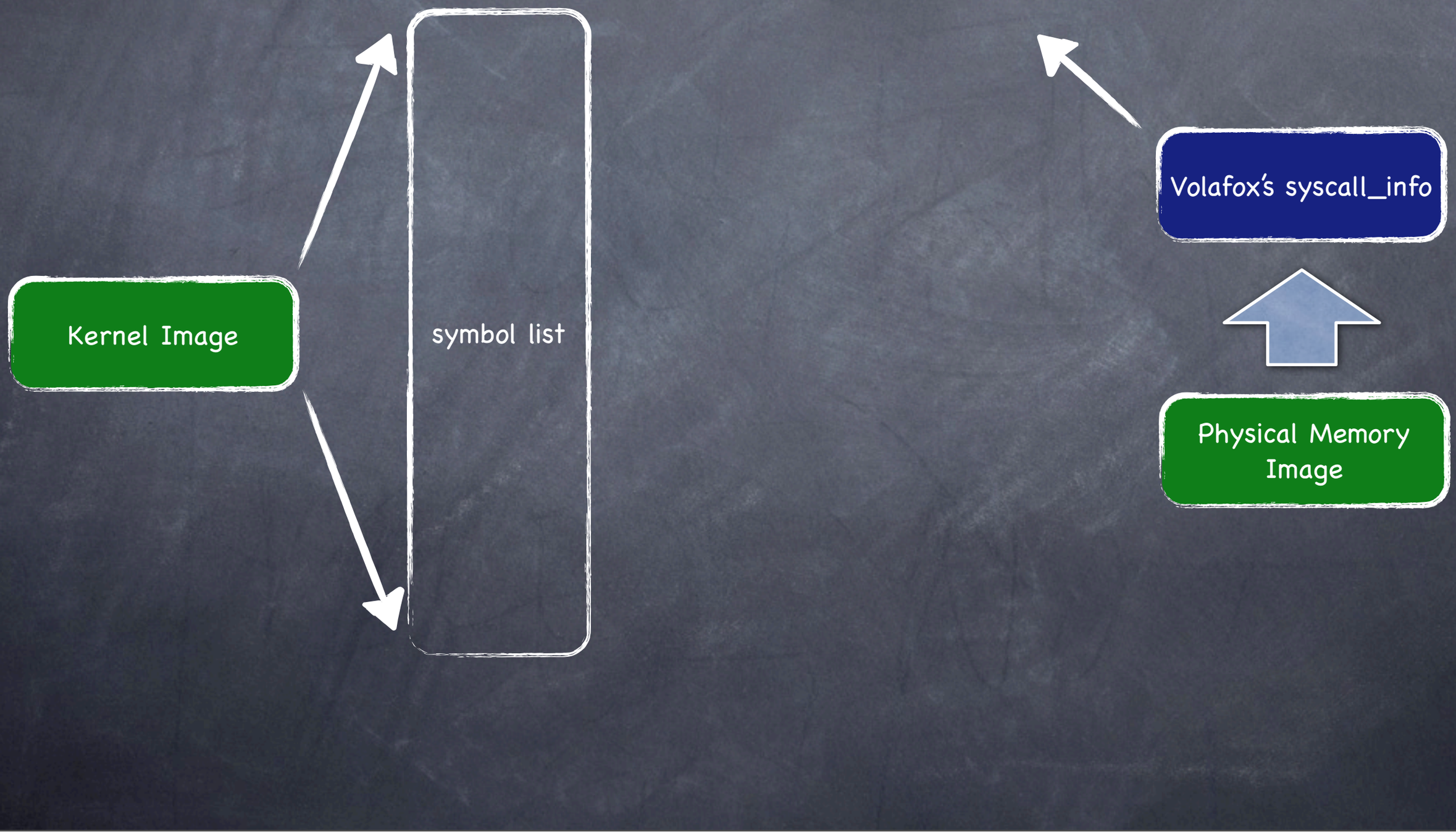
Detection



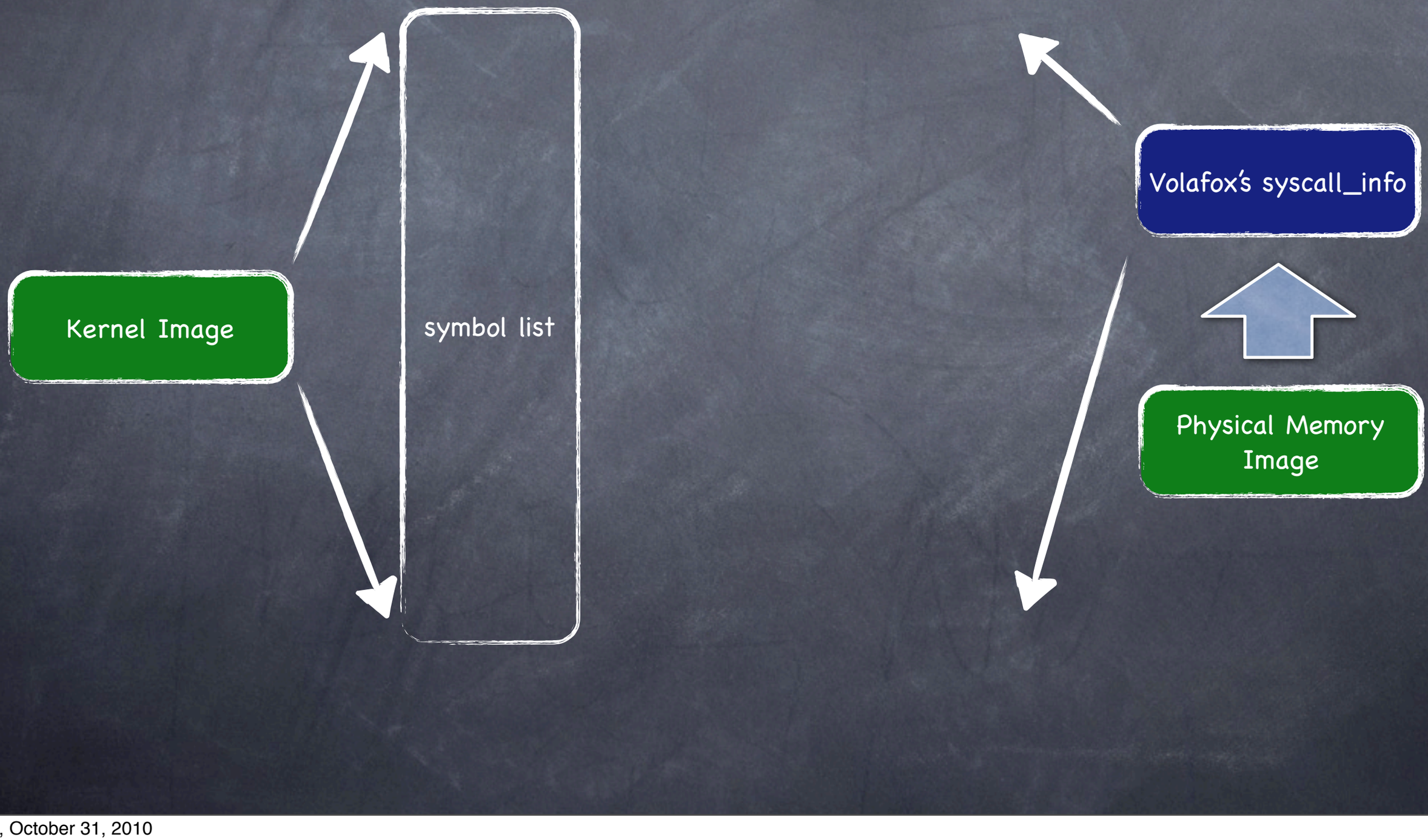
Detection


Kernel Image
symbol list
Volafox's syscall_info
Physical Memory Image

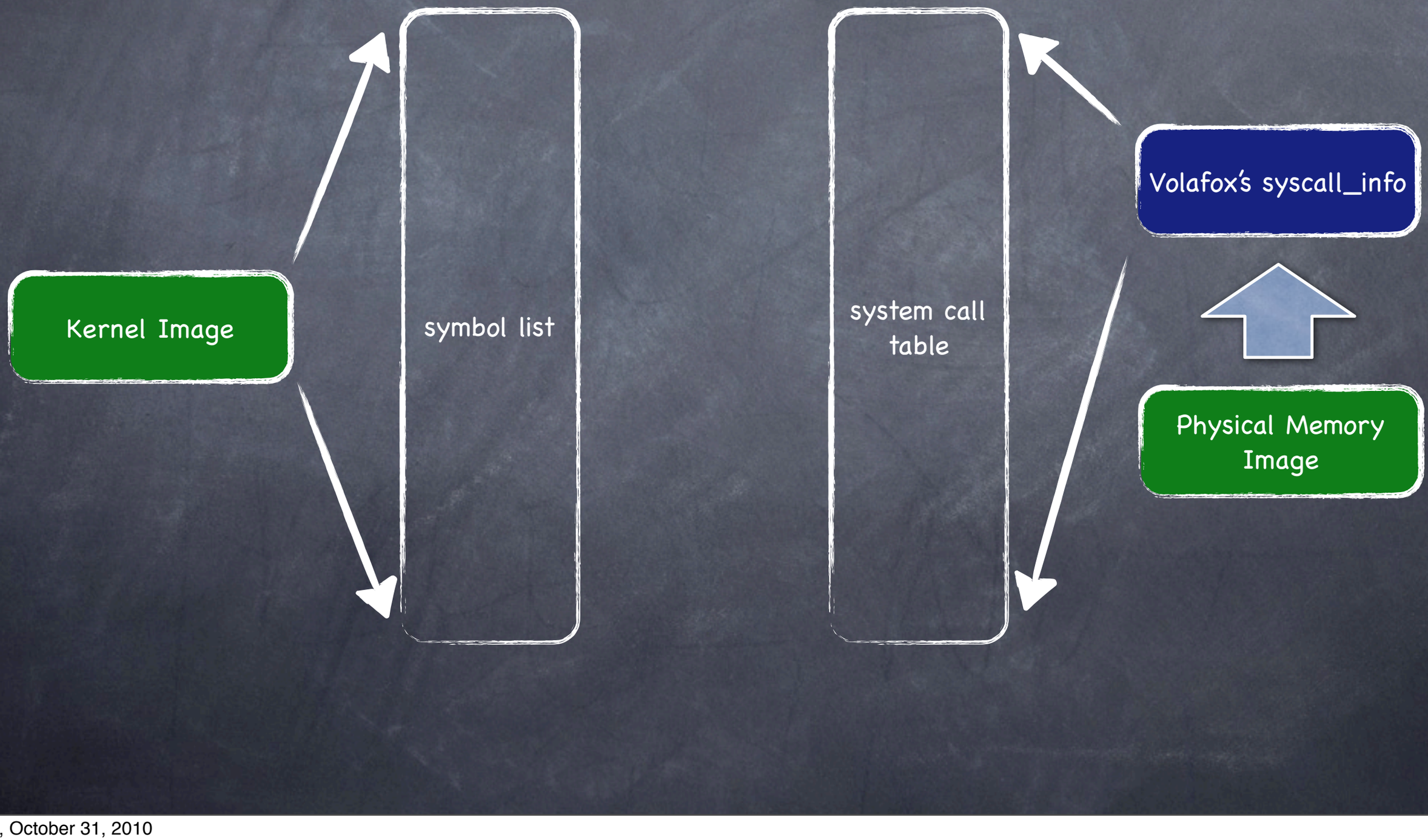
Detection


Kernel Image
symbol list
Volafox's syscall_info
Physical Memory Image

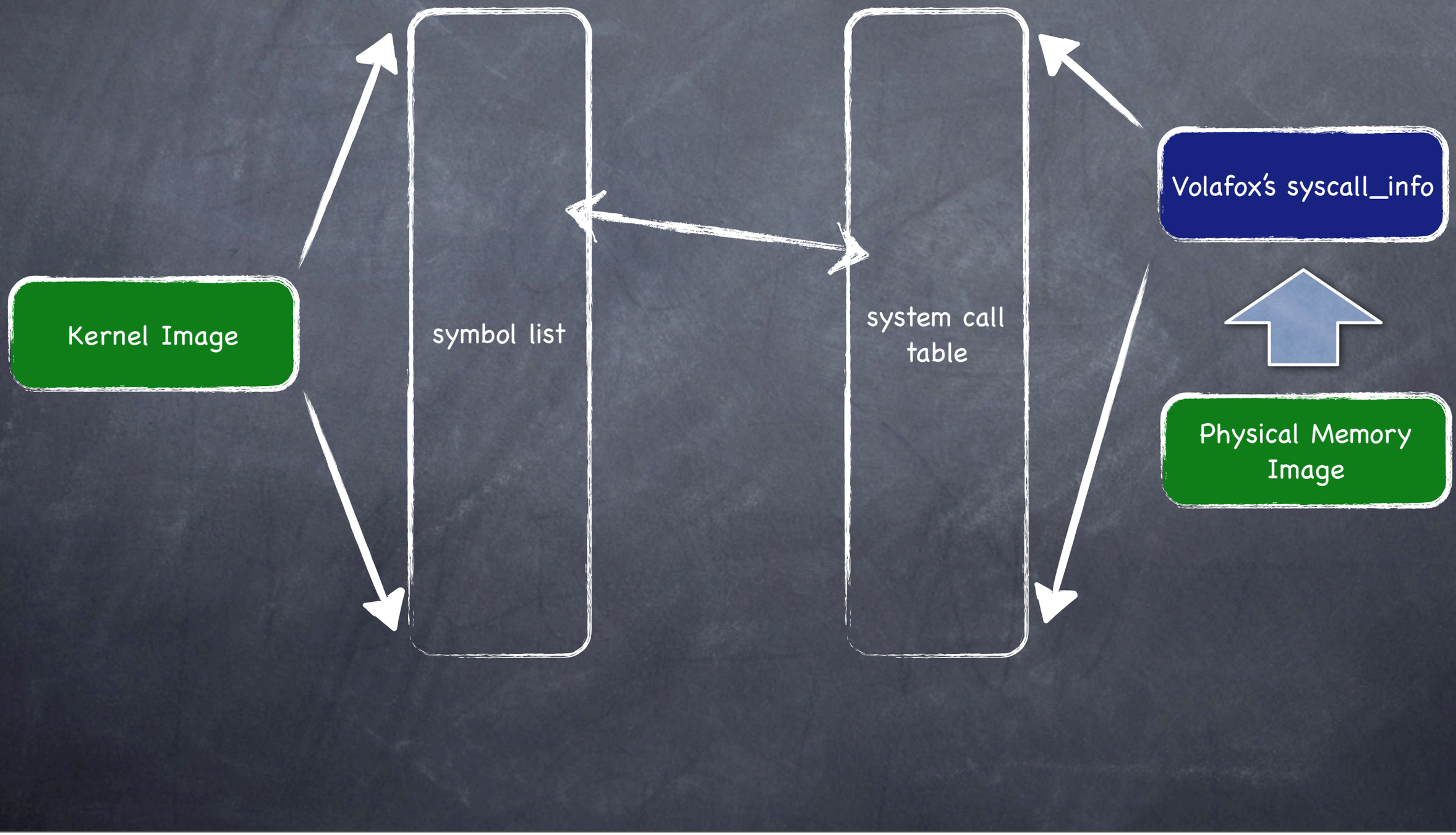
Detection


Kernel Image
symbol list
Volafox's syscall_info
Physical Memory Image

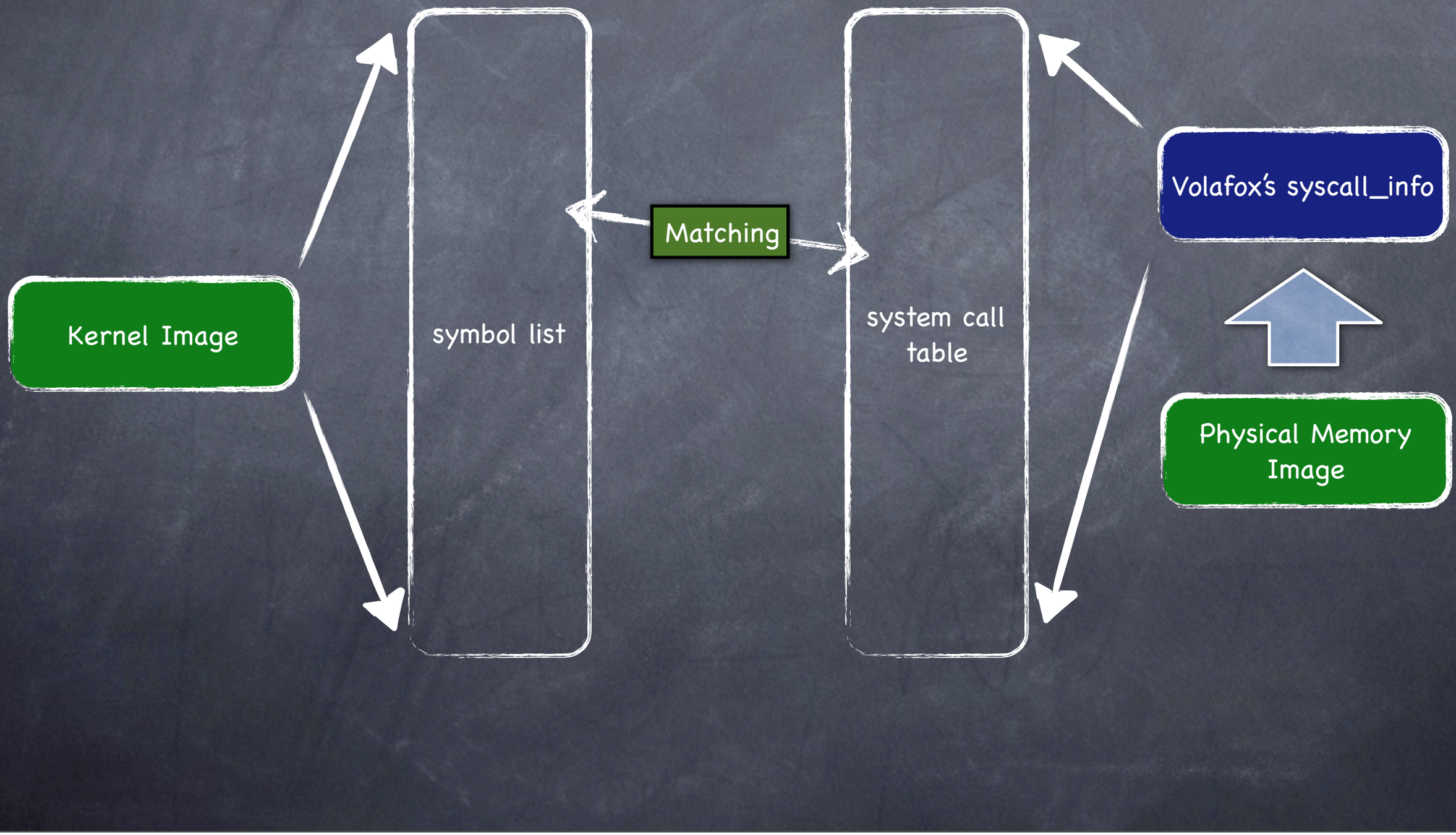
Detection


Kernel Image
symbol list
system call table
Volafox's syscall_info
Physical Memory Image

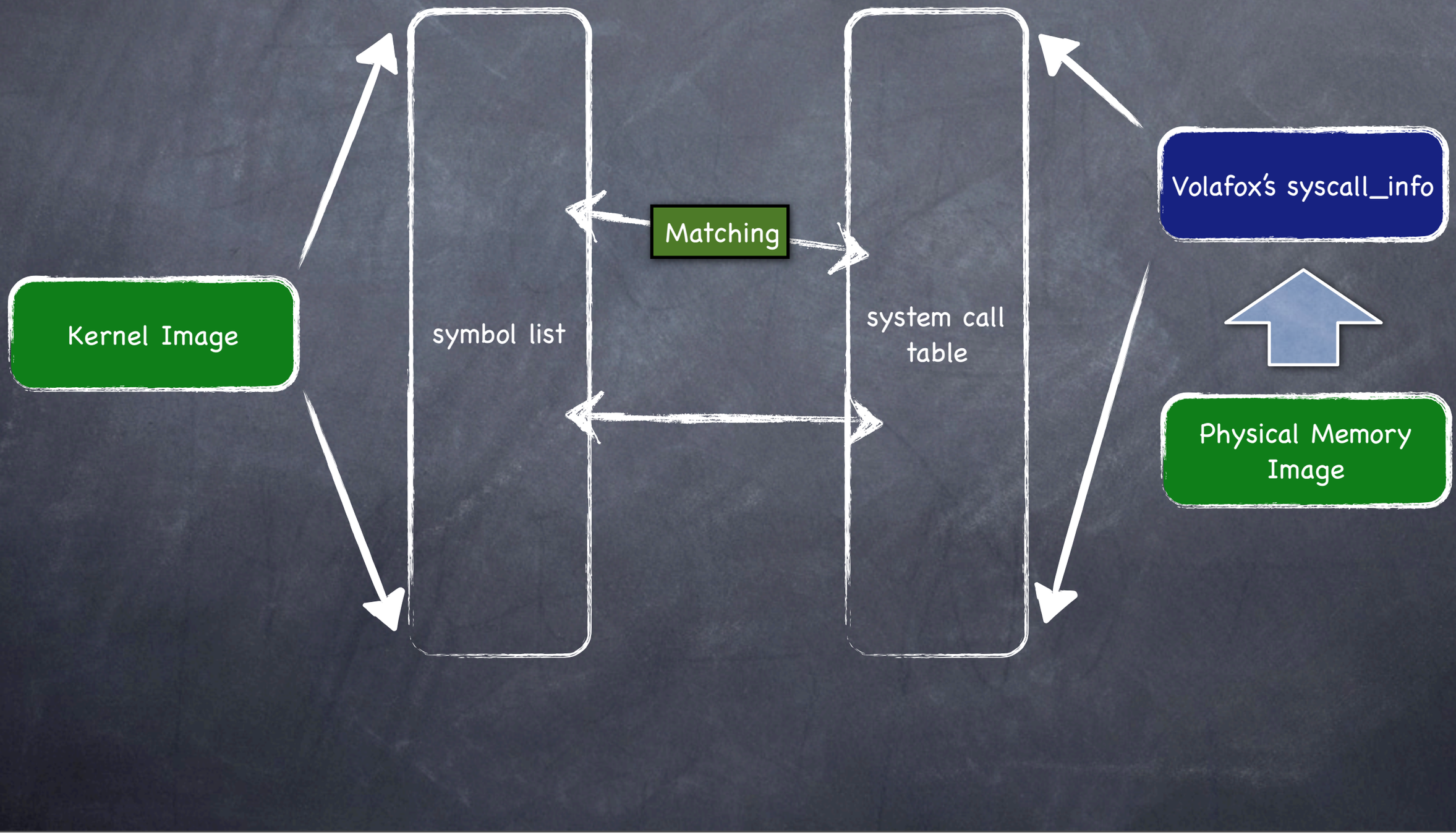
Detection


Kernel Image
symbol list
system call table
Volafox's syscall_info
Physical Memory Image

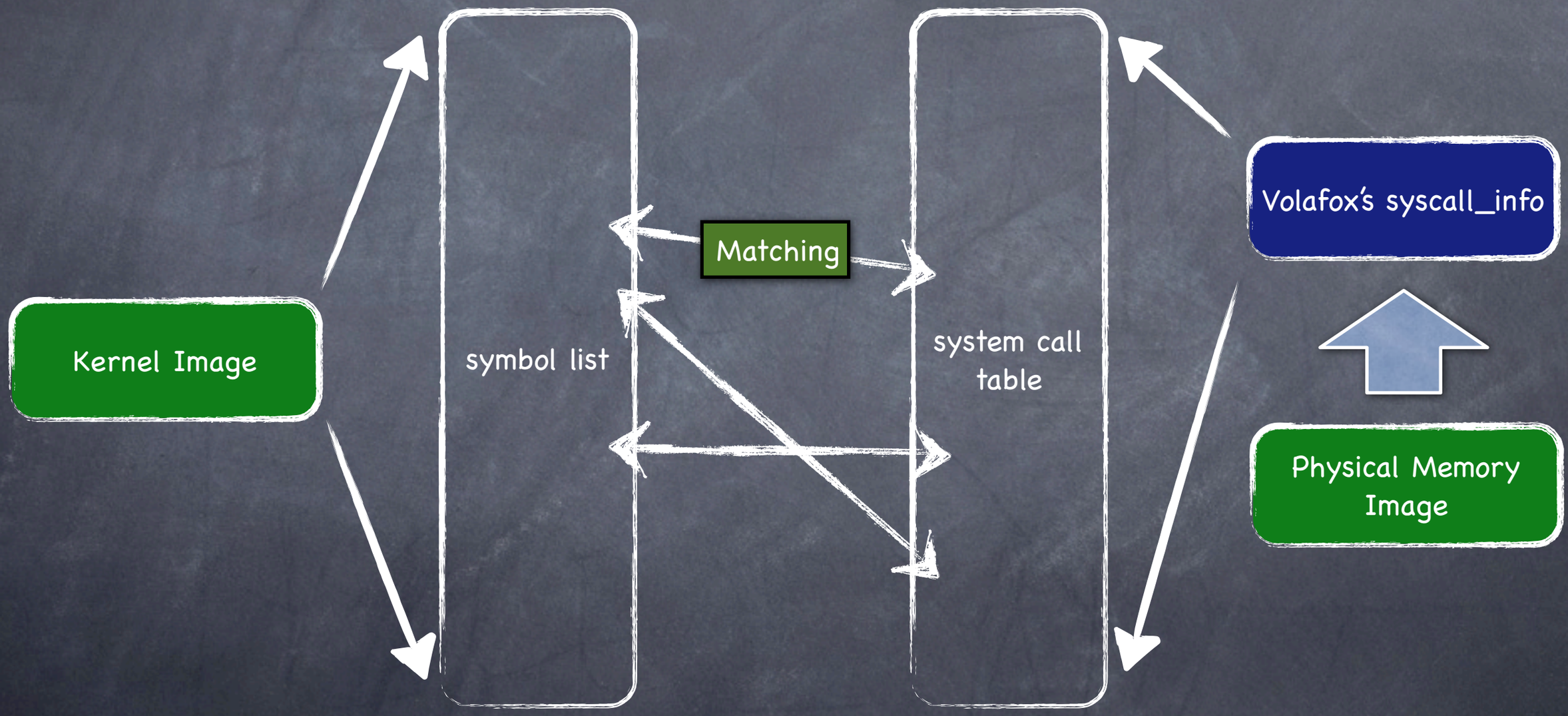
Detection


Kernel Image
symbol list
Matching
system call table
Volafox's syscall_info
Physical Memory Image

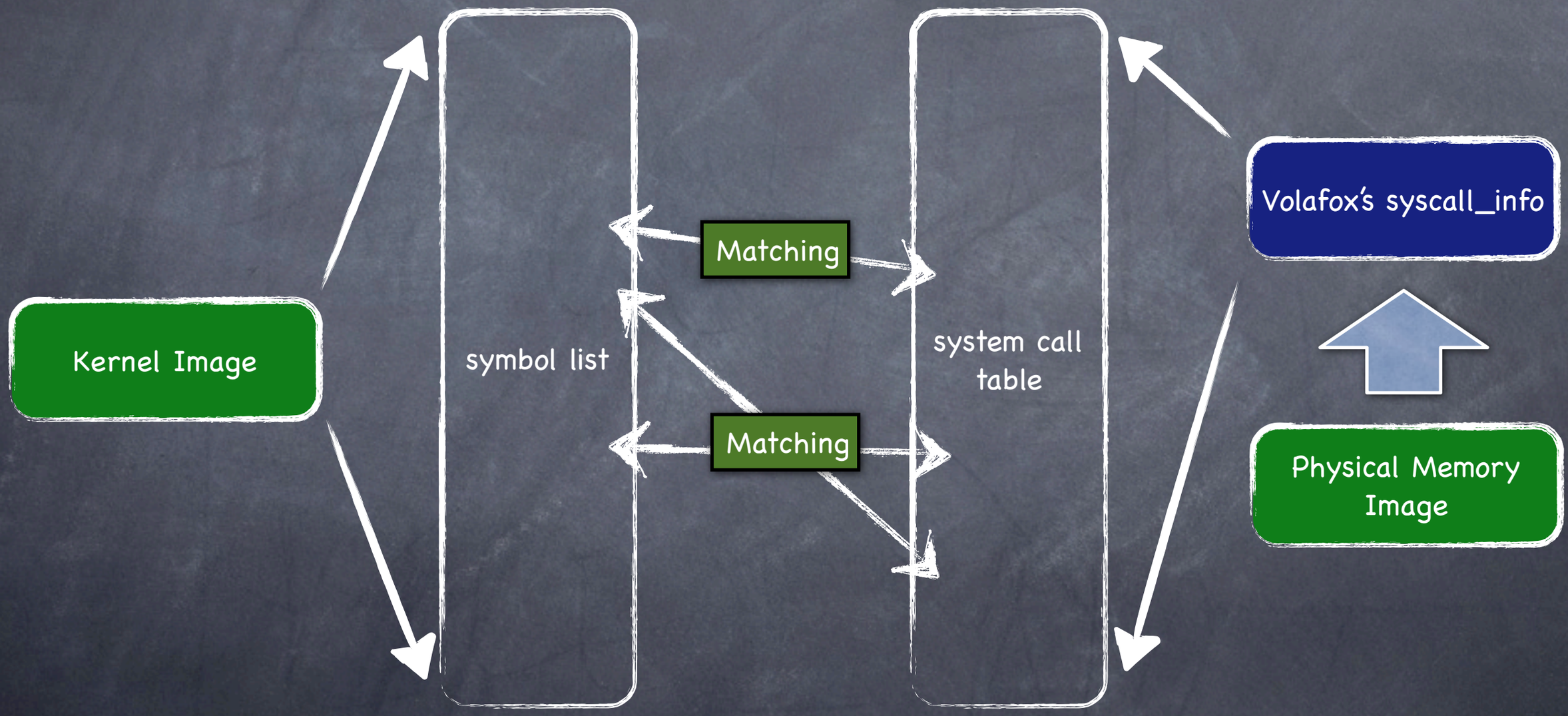
Detection


Kernel Image
symbol list
Matching
system call table
Volafox's syscall_info
Physical Memory Image

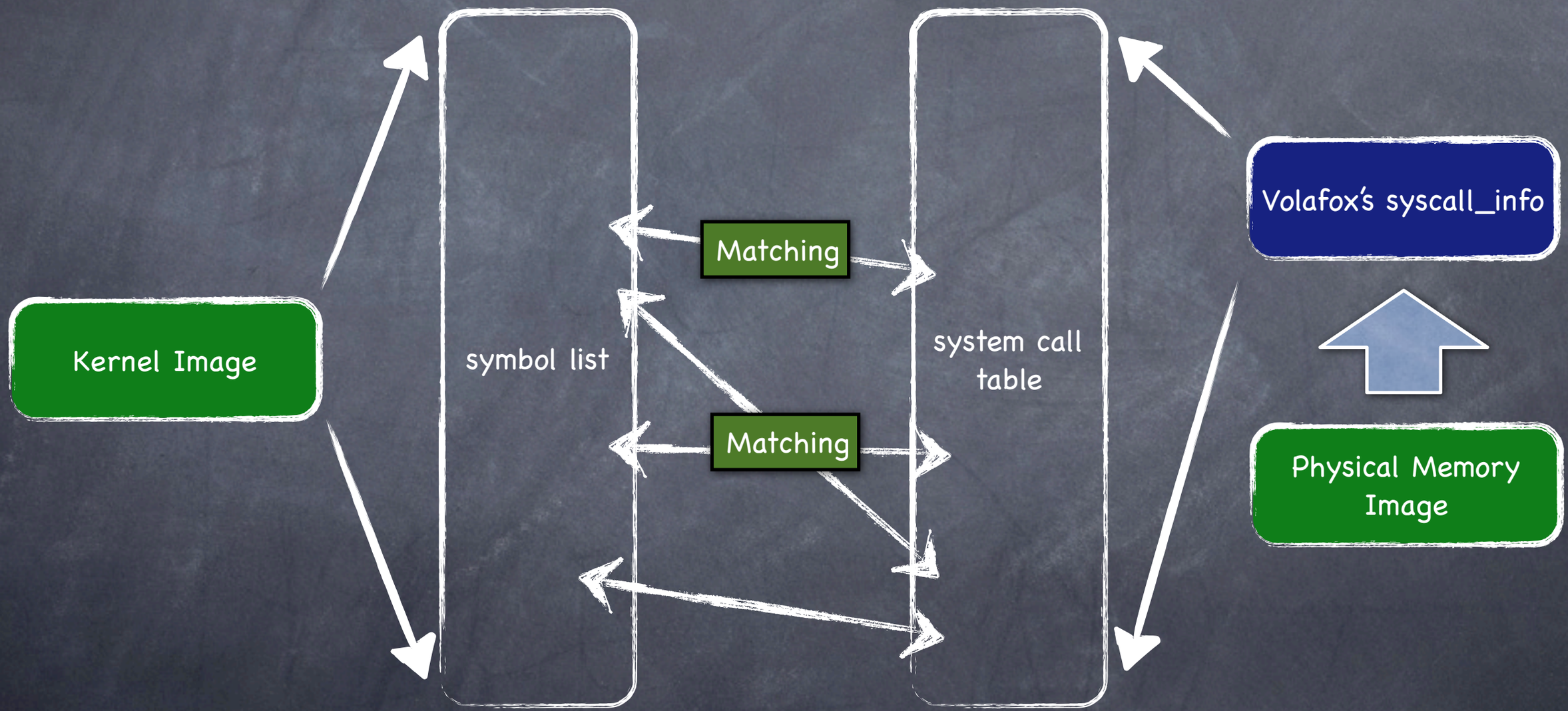
Detection


Kernel Image
symbol list
Matching
system call table
Volafox's syscall_info
Physical Memory Image

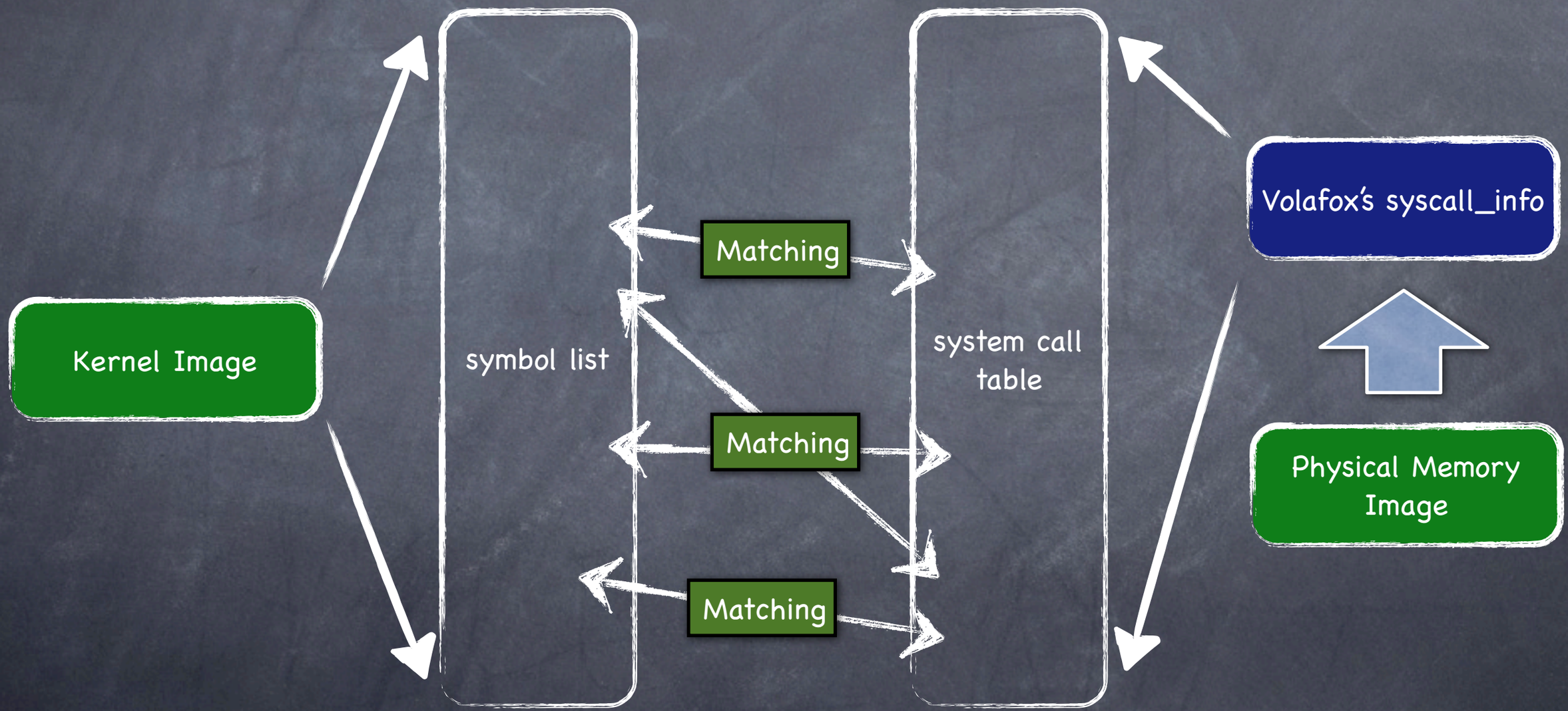
Detection


Kernel Image
symbol list
Matching
Matching
system call table
Volafox's syscall_info
Physical Memory Image

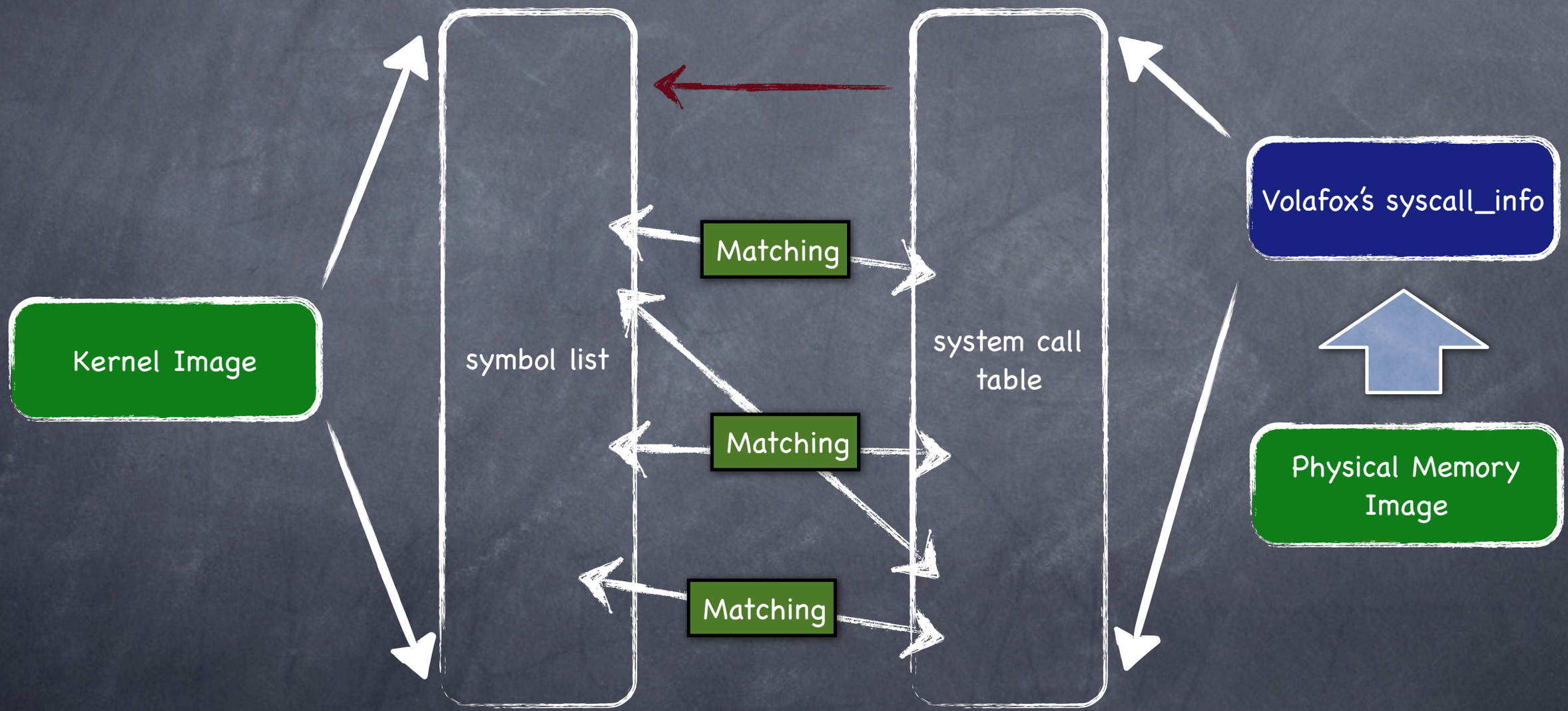
Detection


Kernel Image
symbol list
Matching
Matching
system call table
Volafox's syscall_info
Physical Memory Image

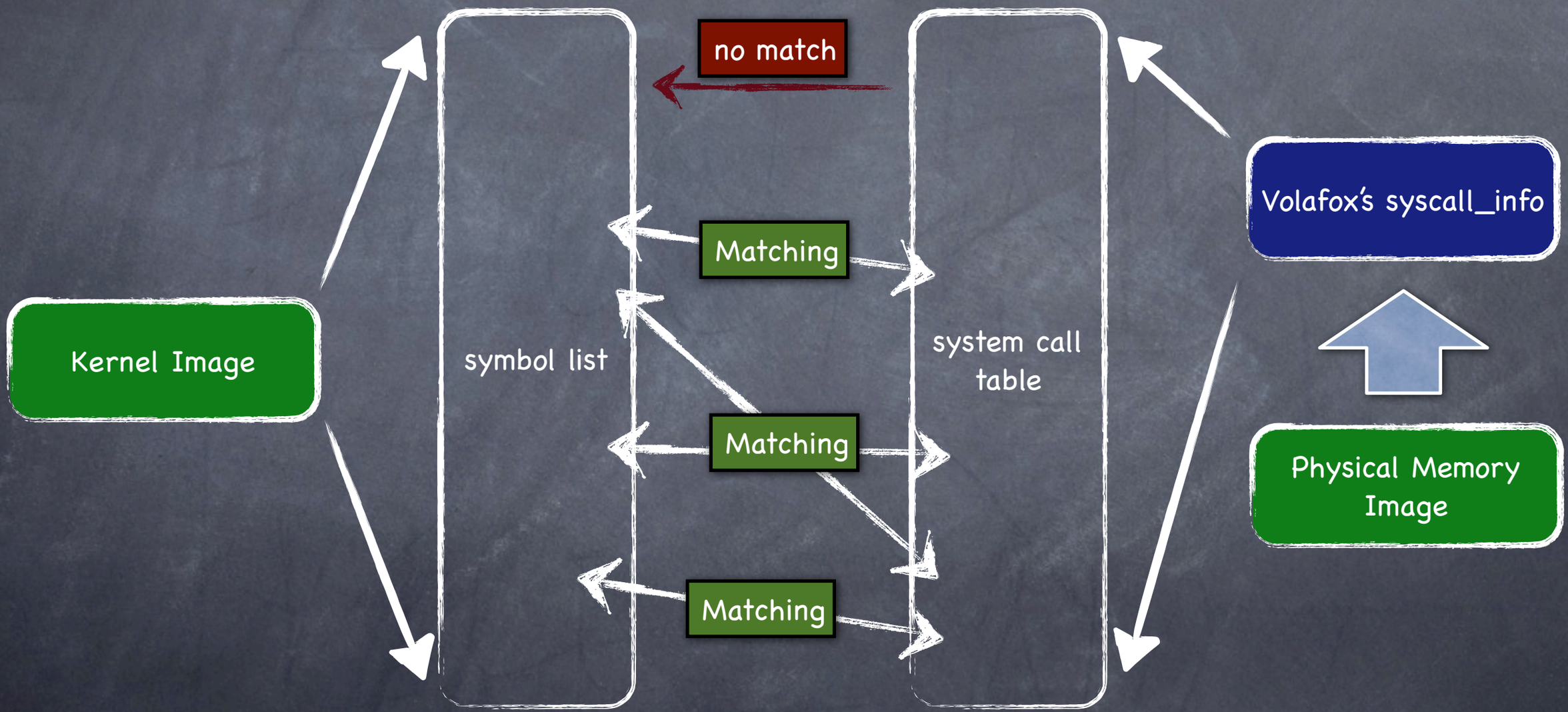
Detection


Kernel Image
symbol list
system call table
Matching
Matching
Matching
Volafox's syscall_info
Physical Memory Image

Detection


Kernel Image
symbol list
system call table
Matching
Matching
Matching
Volafox's syscall_info
Physical Memory Image

Detection


no match
Matching
Matching
Matching
Kernel Image
symbol list
system call table
Volafox's syscall_info
Physical Memory Image

Demo

Terminal — bash — 80×24

```
n0fate:volafox n0fate$ python volafox.py
Memory analyzer for OS X 0.5 - n0fate
Contact: rapfer@gmail.com
usage: python volafox.py -i MEMORY_IMAGE -s KERNEL_IMAGE -o INFORMATION

-= CAUTION =-
this program need to physical memory image, kernel image(mach_kernel)
and it support to Intel x86 Architecture only :(

INFORMATION:
os_version          Dawin kernel detail version
machine_info        Kernel version, cpu, memory information
mount_info          Mount information
kern_kext_info      Kernel KEXT(Kernel Extensions) information
kext_info           KEXT(Kernel Extensions) information
proc_info           Process list
syscall_info        Kernel systemcall information
n0fate:volafox n0fate$
```

Question?

mail: rapfer@gmail.com
blog: http://feedbeef.blogspot.com