Introdução à Pesquisa em Vulnerabilidades no Kernel do Linux

Anderson Nascimento
anderson [at] allelesecurity [dot] com [dot] br

Agenda

ALLELE
SECURITY INTELLIGENCE

Sobre

Parte 1

- Arquitetura de Computadores
- Motivações
- Ambiente de Pesquisa
- Ecosistema
- Linux
- Subsistemas
- Ferramentas

Parte 2

- Vulnerabilidades
- Considerações Finais

Perguntas

sobre

ALLELE SECURITY INTELLIGENCE

anderson nascimento
ceo @ allele security intelligence

twitter: andersoncod3
github: andersoncod3
linkedin: andersoncod3
blog: https://andersoncod3.io

anteriormente

- administrador de sistemas
- analista de segurança da informação
- pesquisador em segurança da informação independente

estuda o kernel do freebsd desde ~ 2010
estuda o kernel do linux desde 2015

principais interesses

vulnerability research, rootkits, source code review, operating systems, computer science, mathematics, physics, language

Aviso Legal

ALLELE
SECURITY INTELLIGENCE

Este é o resultado de um trabalho pessoal e sem revisões prévias, nascido do interesse do apresentador como uma forma de estudar assuntos de seu interesse, sendo assim, erros poderão ser cometidos nesta apresentação e peço desculpas desde já.

Arquitetura de computadores

ALLELE
SECURITY INTELLIGENCE

"A arquitetura de computadores é o projeto conceitual e fundamental da estrutura operacional de um sistema computacional. Ela é o estudo dos requisitos necessários para que um computador funcione e de como organizar os diversos componentes para obter melhores desempenhos."

X86 X86-64/AMD64 ARM ARM64 SPARC Itanium MIPS POWERPC

Modos de operação de uma CPU x86/amd64

ALLELE SECURITY INTELLIGENCE

- Real mode
  - 16 bits
  - Sem rings de proteção
  - Acesso total ao hardware
- Protected mode
  - Rings - ring 0, 1, 2, 3
  - Segmentação
  - Paginação
  - 32 bits - 4G
  - Multitasking
- Virtual 8086
  - DOS
- Long mode
  - 4-level paginação
  - Suporte limitado a segmentação
  - AMD64 canonical address
  - 64 bits registradores
  - Mais registradores
  - Sub modo de compatibilidade

Modos de operação de uma CPU x86/amd64

ALLELE SECURITY INTELLIGENCE

- SMM
  - 16 bits
  - smram
  - Gerenciador SMI provido pela BIOS/UEFI
  - Usado normalmente para gerenciamento de erros críticos

Modos de operação de uma CPU x86

ALLELE SECURITY INTELLIGENCE

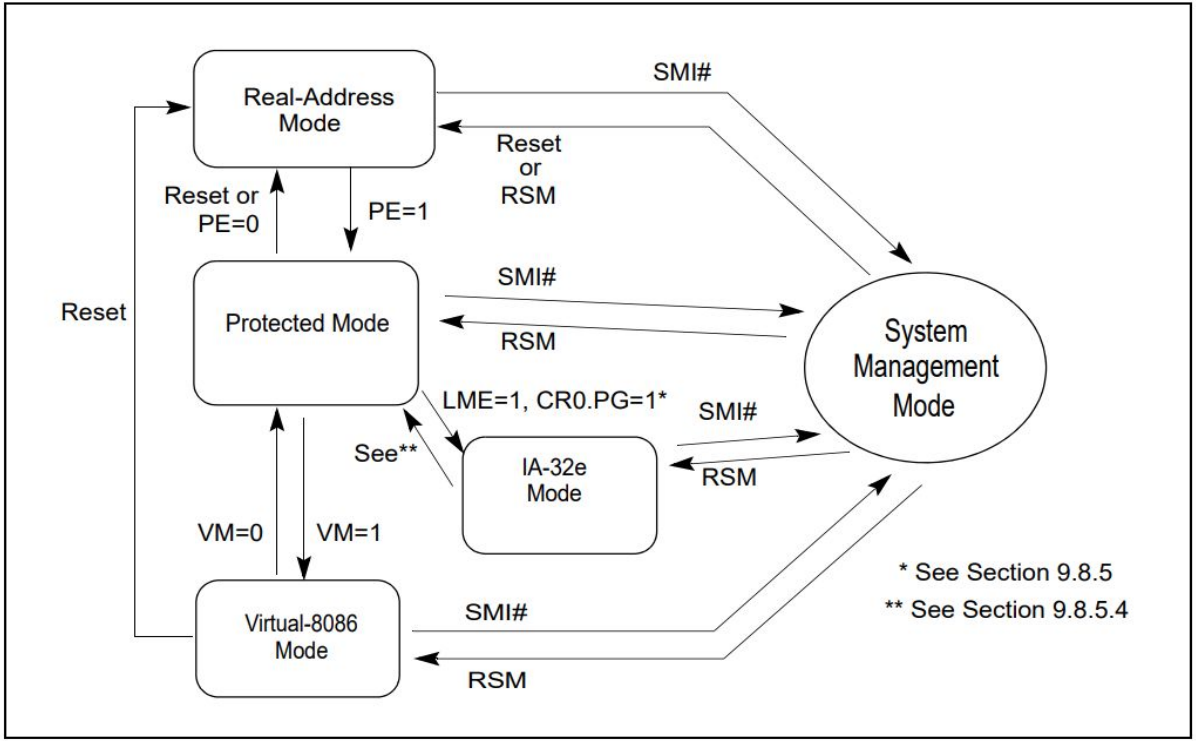
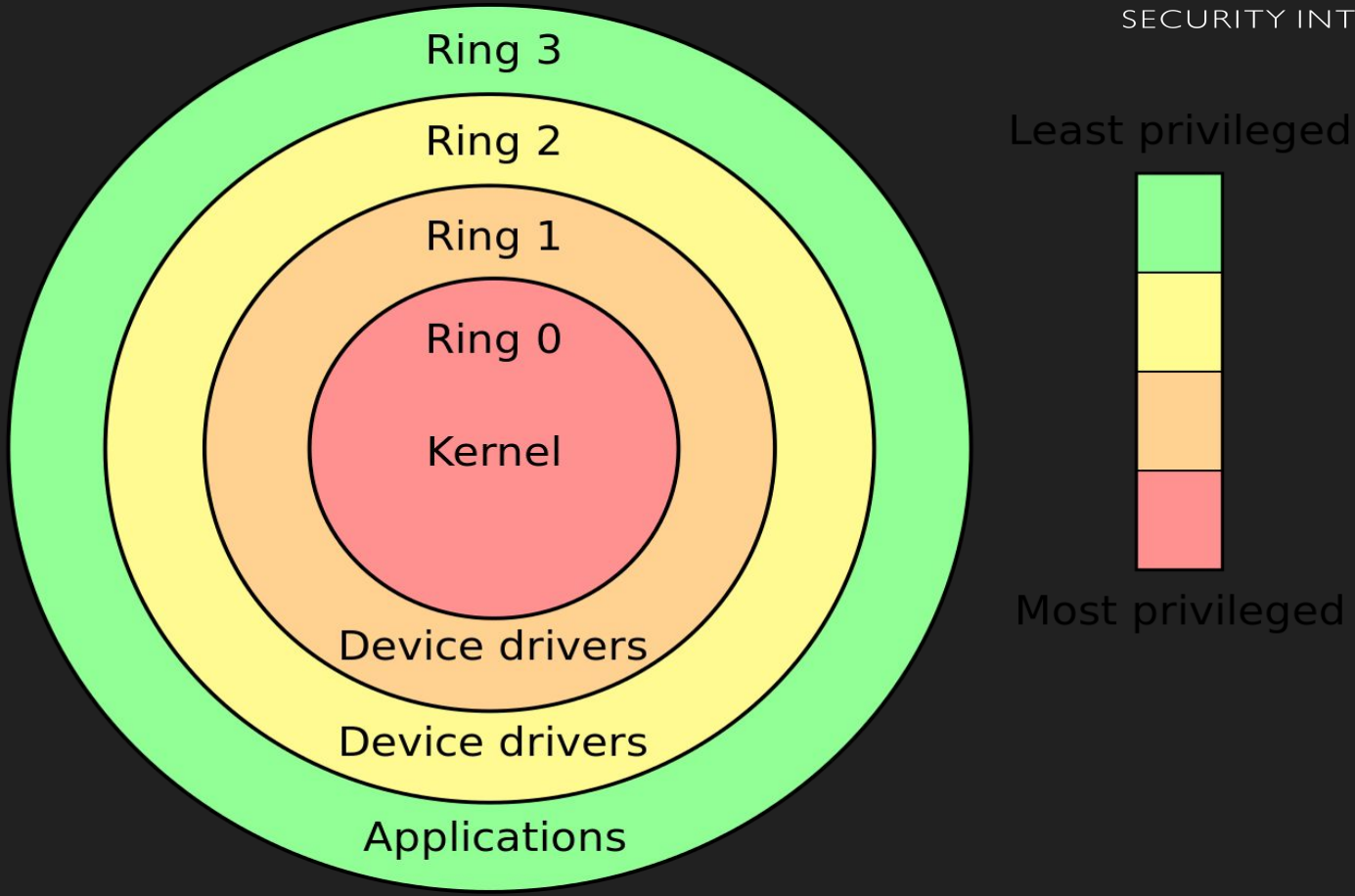


Figure 2-3. Transitions Among the Processor's Operating Modes

https://software.intel.com/sites/default/files/managed/a4/60/325384-sdm-vol-3abcd.pdf

Níveis de Privilégios Modo Protegido

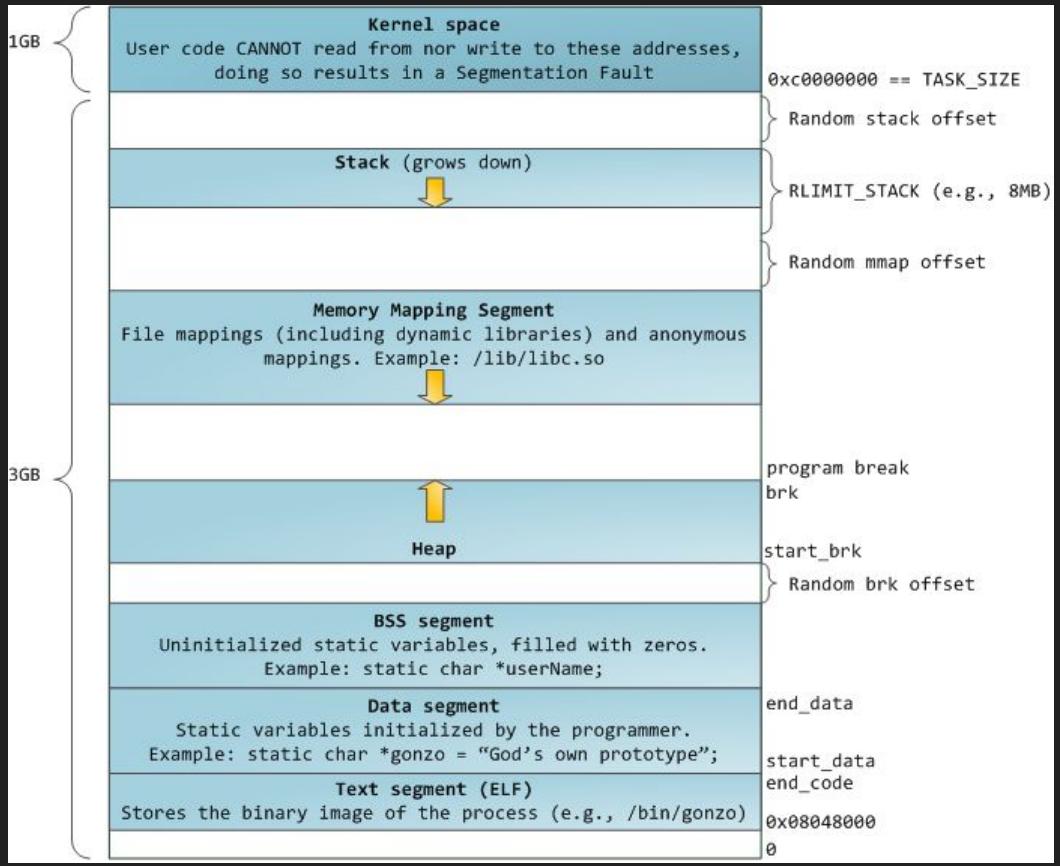
ALLELE SECURITY INTELLIGENCE



https://upload.wikimedia.org/wikipedia/commons/thumb/2/2f/Priv_rings.svg/1280px-Priv_rings.svg.png

Espaço de endereçamento x86

ALLELE SECURITY INTELLIGENCE



https://i.stack.imgur.com/pVeNT.png

Motivações

ALLELE
SECURITY INTELLIGENCE

Porque estudar o Kernel do Linux?

Motivações

ALLELE
SECURITY INTELLIGENCE

Porque estudar o Kernel do Linux?

Conhecimento

Motivações

ALLELE
SECURITY INTELLIGENCE

Porque estudar o Kernel do Linux?

Conhecimento

Aprender como um kernel funciona permite ao programador entender melhor a máquina.

Motivações

ALLELE SECURITY INTELLIGENCE

Porque estudar o Kernel do Linux?

Conhecimento

Oportunidade de emprego

Motivações

ALLELE SECURITY INTELLIGENCE

Porque estudar o Kernel do Linux?

Conhecimento

Oportunidade de emprego

Ter conhecimentos de como um Kernel funciona é um grande diferencial. Pode não haver muitas oportunidades mas as que existem podem ser bastantes interessantes.

Motivações

ALLELE
SECURITY INTELLIGENCE

Porque estudar o Kernel do Linux?

Conhecimento

Oportunidade de emprego

Retorno financeiro

Motivações

ALLELE SECURITY INTELLIGENCE

Porque estudar o kernel do Linux?

Conhecimento

Oportunidade de emprego

Retorno financeiro

Retorno financeiro através do mercado de compra e venda de vulnerabilidades.

- Pwn2Own, $15,000, Ubuntu
- Zimperium, n-days, preço desconhecido
- Zerodium, CentOS Ubuntu Tails, > $30,000
- Internet Bug Bounty, $10,000, Comex, towelroot

Motivações

ALLELE
SECURITY INTELLIGENCE

Porque estudar o Kernel do Linux?

Conhecimento

Oportunidade de emprego

Retorno financeiro

Furtividade

Motivações

ALLELE SECURITY INTELLIGENCE

Porque estudar o Kernel do Linux?

Conhecimento

Oportunidade de emprego

Retorno financeiro

Furtividade

O Kernel é uma das principais partes de um sistema. Se um atacante tem controle do Kernel, seu único limite será sua imaginação.

Motivações

ALLELE
SECURITY INTELLIGENCE

Porque estudar o Kernel do Linux?

Conhecimento

Oportunidade de emprego

Retorno financeiro

Furtividade

Superfície de ataque/escalação de privilégios/sandbox

Motivações

ALLELE SECURITY INTELLIGENCE

Porque estudar o Kernel do Linux?

Conhecimento

Oportunidade de emprego

Retorno financeiro

Furtividade

Superfície de ataque/escalação de privilégios/sandbox

- O Kernel contém por padrão uma grande superfície de ataque.

Ambiente de Pesquisa/Estudo

ALLELE
SECURITY INTELLIGENCE

P: Qual ambiente mais adequado para se iniciar os estudos do Kernel do Linux?

Ambiente de Pesquisa/Estudo

ALLELE SECURITY INTELLIGENCE

P: Qual ambiente mais adequado para se iniciar os estudos do Kernel do Linux?

R: Depende de cada pesquisador e suas motivações

Ambiente de Pesquisa/Estudo

ALLELE SECURITY INTELLIGENCE

P: Qual ambiente mais adequado para se iniciar os estudos do Kernel do Linux?

R: Depende de cada pesquisador e suas motivações

Porém, existem algumas possibilidades:

Ambiente de Pesquisa/Estudo

ALLELE SECURITY INTELLIGENCE

P: Qual ambiente mais adequado para se iniciar os estudos do Kernel do Linux?

R: Depende de cada pesquisador e suas motivações

Porém, existem algumas possibilidades:

Maquinas reais

Ambiente de Pesquisa/Estudo

ALLELE SECURITY INTELLIGENCE

P: Qual ambiente mais adequado para se iniciar os estudos do Kernel do Linux?

R: Depende de cada pesquisador e suas motivações

Porém, existem algumas possibilidades:

Maquinas reais

Maquinas virtuais

Ambiente de Pesquisa/Estudo

ALLELE SECURITY INTELLIGENCE

P: Qual ambiente mais adequado para se iniciar os estudos do Kernel do Linux?

R: Depende de cada pesquisador e suas motivações

Porém, existem algumas possibilidades:

Maquinas reais

Maquinas virtuais

Núvem

Ambiente de Pesquisa/Estudo

ALLELE
SECURITY INTELLIGENCE

Maquinas reais

- Vantagem
  - Segurança
- Desvantagem
  - Gerenciamento
  - Custo
  - Hardware
  - Mobilidade

Ambiente de Pesquisa/Estudo

ALLELE
SECURITY INTELLIGENCE

Maquinas Virtual

- Vantagem
  - Segurança
  - Custo
- Desvantagem
  - Espaço em disco
  - Mobilidade

Ambiente de Pesquisa/Estudo

ALLELE
SECURITY INTELLIGENCE

Núvem

- Vantagem
  - Mobilidade
  - Hardware
- Desvantagem
  - Segurança
  - Custo

Ecosistema

ALLELE SECURITY INTELLIGENCE

Linus Torvalds

- Criador, mantenedor e principal desenvolvedor

Greg Kroah-Hartman

- Desenvolvedor e mantenedor de vários subsistemas
- Mantenedor do branch -stable

Desenvolvedores

Empresas

- Red Hat
- Google
- SUSE
- Canonical
- Oracle

Distribuições

Ecosistema

ALLELE
SECURITY INTELLIGENCE

Empresas:

- Red Hat
  - Red Hat Enterprise Linux
  - Fedora
  - CentOS
- Google
  - Android
  - ChromeOS/ChromiumOS
- SUSE
  - openSUSE
- Canonical
  - Ubuntu
- Oracle
  - Oracle Linux

Linux

ALLELE SECURITY INTELLIGENCE

Kernel release 4.14.0

- 61,000 files
- 25,000,000 lines

4,394 developers

≈530 companies kernel

releases 4.9.0 - 4.14.0

December 2016 - November 2017

https://github.com/gregkh/kernel-development/blob/master/kernel-development.pdf

Linux

ALLELE SECURITY INTELLIGENCE

10,000 lines added

2,700 lines removed

2,000 lines modified kernel

releases 4.9.0 - 4.14.0

December 2016 - November 2017

Every day

https://github.com/gregkh/kernel-development/blob/master/kernel-development.pdf

Como obter o kernel do Linux

ALLELE SECURITY INTELLIGENCE

GitHub

http://github.com/torvalds/linux

git clone http://github.com/torvalds/linux

kernel.org

https://git.kernel.org/

git clone https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git/

git.centos.org

https://git.centos.org/summary/rpms!kernel.git

git clone https://git.centos.org/git/rpms/kernel.git && cd kernel && git checkout c7 / git checkout [tag]

git clone https://git.centos.org/git/centos-git-common.git
get_sources.sh

Como obter o kernel do Linux

ALLELE SECURITY INTELLIGENCE

Continuous release - permite acesso previamente a pacotes que estarão disponíveis nas próximas versões

AdditionalResources » Repositories » CR

Text | Titles

The Continuous Release (CR) Repository

Contents

1. Purpose

The continuous release (CR) repository makes generally available packages that will appear in the next point release of CentOS, on a testing and **hotfix** basis until formally released.

2. The Issue

At a point release stage, new sources have been released by the upstream vendor, but binary updates are not formally issued as **complete** until the ISOs are ready (e.g., here, during the 5.8 release build process). This entails, in part building the updated RPMs with any needed patches; solving installer issues that crop up in **anaconda**; verification by the QA group of content; the delay while mirrors synchronize sufficiently to handle the update load; and finally formal release of a new point release.

For new major releases (i.e., the **.0** point release), there is no prior point release in the major release series which may need security issues addressed, and so no need for streaming out 6.0 series updates before 6.0 itself was formally released; however for existing installs of a given major release, some administrators may desire, after review of announced updates from the upstream vendor, to update sooner than the complete CentOS point release rebuild process formerly permitted.

https://wiki.centos.org/AdditionalResources/Repositories/CR

Como obter o kernel do Linux

continuous release - como usar

ALLELE SECURITY INTELLIGENCE

4. Usage

Usage of the **CR** repository is entirely optional to a system administrator, and is based on their express actions to opt-in.

Installation instructions for CentOS 5 and 6: You can install the CR repository by running **yum install centos-release-cr**. The package is included in the CentOS Extras repository, enabled by default.

Installation instructions for CentOS 7: The repository configuration file is included in the newest centos-release package. First update your system with **yum update** to get the new centos-release package, then run **yum-config-manager --enable cr** to enable the CR repository.

You can verify that your repository configuration is correct by running **yum repolist**. You should see a CR repository in the repository list. The CR repository will be empty and ignored until packages are placed in it at point release roll-over time. It will be emptied once the new point release is complete, but you can remain with the CR repository enabled if you so choose. This will allow you to be set up to receive CR updates for the next point release roll-over cycle.

https://wiki.centos.org/AdditionalResources/Repositories/CR

Subsistemas

ALLELE
SECURITY INTELLIGENCE

IPC (Inter Process Communication)

MM (Memory Management)

VFS (Virtual File Systems)

Networking

Security

Auditing

Process

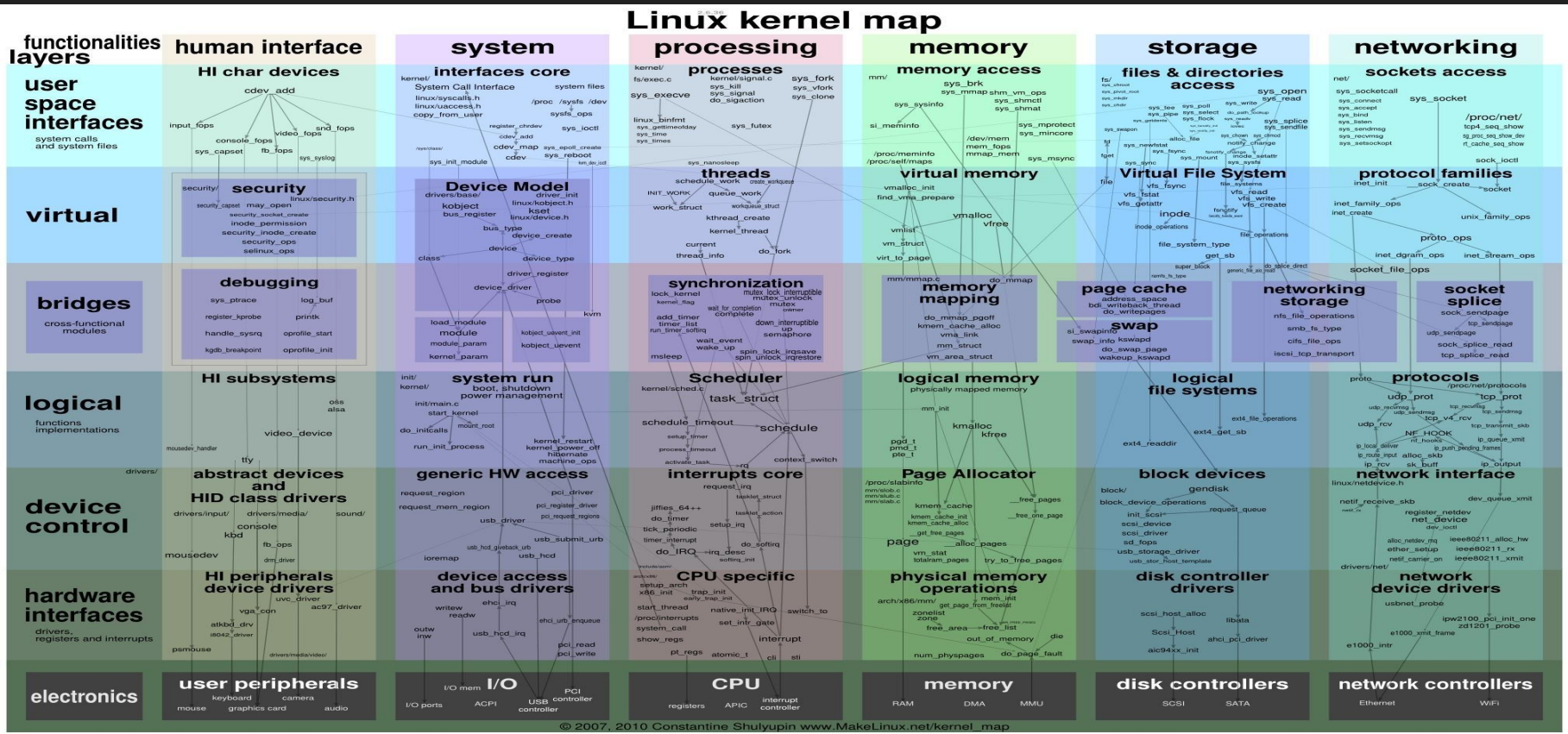
Scheduling

Power management

...

subsistemas

ALLELE SECURITY INTELLIGENCE



Ferramentas

ALLELE SECURITY INTELLIGENCE

Editor de texto

- vim

Navegadores de base de código

- cscope

Diffing

- meld

Análise estática

- smatch
- coccinelle
- sparse

Debugging

- gdb + vmware

Ferramentas

ALLELE
SECURITY INTELLIGENCE

Dynamic Tracing

- SystemTap
- Kprobes

Fuzzers

- Syzkaller
- Trinity

Ferramentas - cscope

ALLELE SECURITY INTELLIGENCE

```
Functions calling this function: __sys_recvmmsg

  File         Function            Line
0 net/compat.c compat_sys_recvmmsg  786 return __sys_recvmmsg(fd, (struct mmsghdr __user *)mmsg, vlen,
1 net/compat.c compat_sys_recvmmsg  791 return __sys_recvmmsg(fd, (struct mmsghdr __user *)mmsg, vlen,
2 net/compat.c compat_sys_recvmmsg  797 datagrams = __sys_recvmmsg(fd, (struct mmsghdr __user *)mmsg, vlen,
3 net/socket.c SYSCALL_DEFINE5     2420 return __sys_recvmmsg(fd, mmsg, vlen, flags, NULL);
4 net/socket.c SYSCALL_DEFINE5     2425 datagrams = __sys_recvmmsg(fd, mmsg, vlen, flags, &timeout_sys);

Find this C symbol:
Find this global definition:
Find functions called by this function:
Find functions calling this function:
Find this text string:
Change this text string:
Find this egrep pattern:
Find this file:
Find files #including this file:
Find assignments to this symbol:
```

Ferramentas - smatch

ALLELE SECURITY INTELLIGENCE

```
git clone git://repo.or.cz/smatch.git

cd smatch

make
```

Now let's create a small test.c file:

```
#include "check_debug.h"
int var;
void function(void)
{

if (var < 0 || var >= 10)

return;

__smatch_implied(var);
}
```

The "check_debug.h" file can be included into any .c file. It is used to display internal Smatch information which helps with debugging. If you run `./smatch test.c`, then that prints the value of the "var" variable.

```
test.c:9 function() implied: var = '0-9'
```

https://blogs.oracle.com/linuxkernel/smatch-static-analysis-tool-overview,-by-dan-carpenter

ALLELE
SECURITY INTELLIGENCE

PARTE 2

VULNERABILIDADES

Vulnerabilidades

ALLELE SECURITY INTELLIGENCE

CVE-2015-7613

"Race condition in the IPC object implementation in the Linux kernel through 4.2.3 allows local users to gain privileges by triggering an IPC_ADDID call that leads to uid and gid comparisons against uninitialized data, related to msg.c, shm.c, and util.c."

https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2015-7613

IPC SUBSYSTEM - SYSTEMV IPC

ALLELE SECURITY INTELLIGENCE

SVIPC(7) Linux Programmer's Manual SVIPC(7)

NAME

svipc - System V interprocess communication mechanisms

SYNOPSIS

```
#include <sys/msg.h>
#include <sys/sem.h>
#include <sys/shm.h>
```

DESCRIPTION

This manual page refers to the Linux implementation of the System V interprocess communication (IPC) mechanisms: message queues, semaphore sets, and shared memory segments. In the following, the word resource means an instantiation of one among such mechanisms.

man svipc

Vulnerabilidades - CVE-2015-7613

ALLELE SECURITY INTELLIGENCE

A vulnerabilidade afetava todo o subsistema (3 mecanismos) mas quando ela foi corrigida inicialmente em upstream, somente em um mecanismo foi corrigido (semáforo). Após a correção, ainda era possível explorar a mesma vulnerabilidade utilizando os outros dois mecanismos do subsistema IPC (message, shared memory).

https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2015-7613

Initialize msg/shm IPC objects before doing ipc_addid()

Browse files

As reported by Dmitry Vyukov, we really shouldn't do ipc_addid() before having initialized the IPC object state. Yes, we initialize the IPC object in a locked state, but with all the lockless RCU lookup work, that IPC object lock no longer means that the state cannot be seen.

We already did this for the IPC semaphore code (see commit e8577d1: "ipc/sem.c: fully initialize sem_array before making it visible") but we clearly forgot about msg and shm.

Reported-by: Dmitry Vyukov <dvyukov@google.com>
Cc: Manfred Spraul <manfred@colorfullife.com>
Cc: Davidlohr Bueso <dbueso@suse.de>
Cc: stable@vger.kernel.org
Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>

master (#115) v4.16 ... v4.3-rc4

torvalds committed on Sep 30, 2015 1 parent 3225031 commit b9a532277938798b53178d5a66af6e2915cb27cf

Vulnerabilidades - CVE-2015-7613

ALLELE SECURITY INTELLIGENCE

ipc/sem.c: fully initialize sem_array before making it visible

Browse files

```
ipc_addid() makes a new ipc identifier visible to everyone.  New objects
start as locked, so that the caller can complete the initialization
after the call.  Within struct sem_array, at least sma->sem_base and
sma->sem_nsems are accessed without any locks, therefore this approach
doesn't work.

Thus: Move the ipc_addid() to the end of the initialization.

Signed-off-by: Manfred Spraul <manfred@colorfullife.com>
Reported-by: Rik van Riel <riel@redhat.com>
Acked-by: Rik van Riel <riel@redhat.com>
Acked-by: Davidlohr Bueso <dave@stgolabs.net>
Acked-by: Rafael Aquini <aquini@redhat.com>
Signed-off-by: Andrew Morton <akpm@linux-foundation.org>
Signed-off-by: Linus Torvalds <torvalds@linux-foundation.org>
```

master (#4) v4.16 ··· v3.18

manfred-colorfu authored and **torvalds** committed on Dec 2, 2014 1 parent 92788ac commit e8577d1f0329d4842e8302e289fb2c22156abef4

Vulnerabilidades - CVE-2015-7613

ALLELE SECURITY INTELLIGENCE

Lições

Programadores experientes, também erram (esquecem).

Conhecimento do funcionamento interno do subsistema ajuda bastante durante o processo de pesquisa, entendimento e exploração de vulnerabilidades.

Vulnerabilidades

ALLELE SECURITY INTELLIGENCE

CVE-2017-10661

"Race condition in fs/timerfd.c in the Linux kernel before 4.10.15 allows local users to gain privileges or cause a denial of service (list corruption or use-after-free) via simultaneous file-descriptor operations that leverage improper might_cancel queueing."

https://nvd.nist.gov/vuln/detail/CVE-2017-10661

TIMERFD SUBSYSTEM

ALLELE SECURITY INTELLIGENCE

TIMERFD_CREATE(2) Linux Programmer's Manual TIMERFD_CREATE(2)

NAME

timerfd_create, timerfd_settime, timerfd_gettime - timers that notify via file descriptors

SYNOPSIS

```
#include <sys/timerfd.h>

int timerfd_create(int clockid, int flags);

int timerfd_settime(int fd, int flags,
                    const struct itimerspec *new_value,
                    struct itimerspec *old_value);

int timerfd_gettime(int fd, struct itimerspec *curr_value);
```

DESCRIPTION

These system calls create and operate on a timer that delivers timer expiration notifications via a file descriptor. They provide an alternative to the use of setitimer(2) or timer_create(2), with the advantage that the file descriptor may be monitored by select(2), poll(2), and epoll(7).

The use of these three system calls is analogous to the use of timer_create(2), timer_settime(2), and timer_gettime(2). (There is no analog of timer_getoverrun(2), since that functionality is provided by read(2), as described below.)

man timerfd_create

Vulnerabilidades - CVE-2017-10661

ALLELE SECURITY INTELLIGENCE

Vulnerabilidade publicamente conhecida que ainda afeta, pelo menos, a versão do CentOS 7 (linux-3.10.0-693.21.1.el7)

https://nvd.nist.gov/vuln/detail/CVE-2017-10661

Vulnerabilidades - CVE-2017-10661

ALLELE SECURITY INTELLIGENCE

```
➜ linux-3.10.0-693.21.1.el7 git:(187831e) ✗ uname -a
Linux research01 3.10.0-693.21.1.el7 #0 SMP Mon Mar 5 11:34:50 -03 2018 x86_64 x86_64 x86_64 GNU/Linux
➜ linux-3.10.0-693.21.1.el7 git:(187831e) ✗ /tmp/poc > /dev/null
^C
➜ linux-3.10.0-693.21.1.el7 git:(187831e) ✗ dmesg|grep -A3 WARNING|tail -f
--
[64423.042382] WARNING: CPU: 2 PID: 101718 at lib/list_debug.c:62 __list_del_entry+0x82/0xd0
[64423.042383] list_del corruption. next->prev should be ffff880079336ca8, but was dead000000000200
[64423.042384] Modules linked in: coretemp snd_seq_midi snd_seq_midi_event iosf_mbi crc32_pclmul ghash_clmulni_intel aesni_intel l
rw gf128mul glue_helper ablk_helper cryptd snd_ens1371 snd_rawmidi snd_ac97_codec ac97_bus snd_seq snd_seq_device snd_pcm snd_time
r nfit snd ppdev pcspkr joydev libnvdimm vmw_balloon parport_pc soundcore parport sg vmw_vmci shpchp i2c_piix4 ip_tables ext4 mbca
che jbd2 sr_mod cdrom ata_generic pata_acpi vmwgfx sd_mod crc_t10dif drm_kms_helper crct10dif_generic syscopyarea sysfillrect sysi
mgblt fb_sys_fops ttm drm ata_piix libata mptspi e1000 scsi_transport_spi mptscsih crct10dif_pclmul crct10dif_common mptbase crc32
c_intel i2c_core serio_raw
[64423.042417] CPU: 2 PID: 101718 Comm: poc Tainted: G        W      ------------   3.10.0-693.21.1.el7 #0
--
[64444.630195] WARNING: CPU: 1 PID: 26296 at lib/list_debug.c:91 __list_add_rcu+0x5b/0x80
[64444.630197] list_add_rcu corruption. next->prev should be prev (ffffffff81a98d60), but was dead000000000200. (next=ffff88007933
6ca8).
[64444.630198] Modules linked in: coretemp snd_seq_midi snd_seq_midi_event iosf_mbi crc32_pclmul ghash_clmulni_intel aesni_intel l
rw gf128mul glue_helper ablk_helper cryptd snd_ens1371 snd_rawmidi snd_ac97_codec ac97_bus snd_seq snd_seq_device snd_pcm snd_time
r nfit snd ppdev pcspkr joydev libnvdimm vmw_balloon parport_pc soundcore parport sg vmw_vmci shpchp i2c_piix4 ip_tables ext4 mbca
che jbd2 sr_mod cdrom ata_generic pata_acpi vmwgfx sd_mod crc_t10dif drm_kms_helper crct10dif_generic syscopyarea sysfillrect sysi
mgblt fb_sys_fops ttm drm ata_piix libata mptspi e1000 scsi_transport_spi mptscsih crct10dif_pclmul crct10dif_common mptbase crc32
c_intel i2c_core serio_raw
[64444.630242] CPU: 1 PID: 26296 Comm: poc Tainted: G        W      ------------   3.10.0-693.21.1.el7 #0
➜ linux-3.10.0-693.21.1.el7 git:(187831e) ✗ _
```

Vulnerabilidades - CVE-2017-10661

ALLELE SECURITY INTELLIGENCE

about summary refs log tree commit diff stats

author	Thomas Gleixner <tglx@linutronix.de>	2017-01-31 15:24:03 +0100
committer	Thomas Gleixner <tglx@linutronix.de>	2017-02-10 11:15:09 +0100
commit	1e38da300e1e395a15048b0af1e5305bd91402f6 (patch)	
tree	9d5c20e4fb1aba203e59dbd6146eee723f258a3a	
parent	7551b02b94ad1daee3a79d667dc3c46d08328f87 (diff)	
download	linux-1e38da300e1e395a15048b0af1e5305bd91402f6.tar.gz	

timerfd: Protect the might cancel mechanism proper

```
The handling of the might_cancel queueing is not properly protected, so
parallel operations on the file descriptor can race with each other and
lead to list corruptions or use after free.

Protect the context for these operations with a seperate lock.

The wait queue lock cannot be reused for this because that would create a
lock inversion scenario vs. the cancel lock. Replacing might_cancel with an
atomic (atomic_t or atomic bit) does not help either because it still can
race vs. the actual list operation.

Reported-by: Dmitry Vyukov <dvyukov@google.com>
Signed-off-by: Thomas Gleixner <tglx@linutronix.de>
Cc: "linux-fsdevel@vger.kernel.org"
Cc: syzkaller <syzkaller@googlegroups.com>
Cc: Al Viro <viro@zeniv.linux.org.uk>
Cc: linux-fsdevel@vger.kernel.org
Link: http://lkml.kernel.org/r/alpine.DEB.2.20.1701311521430.3457@nanos
Signed-off-by: Thomas Gleixner <tglx@linutronix.de>
```

Vulnerabilidades - CVE-2017-10661

ALLELE SECURITY INTELLIGENCE

First Last Prev Next This bug is not in your last search results.

Format For Printing - XML - Clone This Bug - Last Comment

Bug 1481136 -

Summary: **CVE-2017-10661** kernel: Handling of might_cancel queueing is not properly pret...

Status: NEW

Aliases: CVE-2017-10661

Product: Security Response
Component: vulnerability (Show other bugs)
Version: unspecified
Hardware: All Linux

Priority medium **Severity** medium
Target Milestone: ---
Target Release: ---
Assigned To: Red Hat Product Security
QA Contact:
Docs Contact:

URL:
Whiteboard: impact=moderate,public=20170210,repor...
Keywords: Security

Depends On: 1485404 1485405 1485406 1485407 1485408 1485409 1485410
Blocks: ~~1481154~~
Show dependency tree / graph

Reported: 2017-08-14 04:16 EDT [redacted]
Modified: 2018-02-07 18:36 EST (History)
CC List: 37 users
[redacted]

See Also:
Fixed In Version:
Doc Type: Bug Fix
Doc Text: A race condition was found in the Linux kernel before version 4.11-rc1 in 'fs/timerfd.c' file which allows a local user to cause a kernel list corruption or use-after-free via simultaneous operations with a file descriptor which leverage improper 'might_cancel' queuing. An unprivileged local user could use this flaw to cause a denial of service of the system. Due to the nature of the flaw, privilege escalation cannot be fully ruled out, although we believe it is unlikely.
Clone Of:
Environment:
Last Closed:

Vulnerabilidades - CVE-2017-10661

ALLELE SECURITY INTELLIGENCE

Attachments	(Terms of Use)
Add an attachment (proposed patch, testcase, etc.)	

[redacted] 2017-08-14 04:16:55 EDT Description

The handling of the might_cancel queueing is not properly protected, so parallel operations on the file descriptor can race with each other and lead to list corruptions or use after free.

References:

https://marc.info/?l=linux-fsdevel&m=148587265720603&w=2

https://marc.info/?t=148587273100007&r=1&w=2

https://source.android.com/security/bulletin/2017-08-01#kernel-components

Upstream patch:

https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git/commit/?id=1e38da300e1e395a15048b0af1e5305bd91402f6

[redacted] 2017-08-25 11:20:24 EDT Comment 4

Statement:

This issue does not affect Red Hat Enterprise Linux 5 as the code with the flaw is not present in the products listed.

This issue affects Red Hat Enterprise Linux 6, 7 and Red Hat Enterprise MRG 2. Future updates for the respective releases may address this issue.

Note

You need to log in before you can comment on or make changes to this bug.

First Last Prev Next This bug is not in your last search results.

Format For Printing - XML - Clone This Bug - Top of page

Vulnerabilidades - CVE-2017-10661

ALLELE SECURITY INTELLIGENCE

Lições

Backporting

Vulnerabilidades corrigidas em upstream podem levar tempo para serem corrigidas nas distribuições.

Conclusion

This exploit shows how much impact the kernel configuration can have on how easy it is to write an exploit for a kernel bug. While simply turning on every security-related kernel configuration option is probably a bad idea, some of them - like the `kernel.dmesg_restrict` sysctl - seem to provide a reasonable tradeoff when enabled.

The fix timeline shows that the kernel's approach to handling severe security bugs is very efficient at quickly landing fixes in the git master tree, but leaves a window of exposure between the time an upstream fix is published and the time the fix actually becomes available to users - and this time window is sufficiently large that a kernel exploit could be written by an attacker in the meantime.

Posted by Ben at 10:16 AM

Vulnerabilidades

ALLELE SECURITY INTELLIGENCE

CVE-2016-7117

"Use-after-free vulnerability in the _sys_recvmmsg function in net/socket.c in the Linux kernel before 4.5.2 allows remote attackers to execute arbitrary code via vectors involving a recvmmsg system call that is mishandled during error processing."

https://nvd.nist.gov/vuln/detail/CVE-2016-7117

recvmmsg()

ALLELE SECURITY INTELLIGENCE

RECVMMSG(2) Linux Programmer's Manual RECVMMSG(2)

NAME

recvmmsg - receive multiple messages on a socket

SYNOPSIS

```
#define _GNU_SOURCE
#include <sys/socket.h>

int recvmmsg(int sockfd, struct mmsghdr *msgvec, unsigned int vlen,
             unsigned int flags, struct timespec *timeout);
```

DESCRIPTION

The **recvmmsg**() system call is an extension of **recvmsg**(2) that allows the caller to receive multiple messages from a socket using a single system call. (This has performance benefits for some applications.) A further extension over **recvmsg**(2) is support for a timeout on the receive operation.

The *sockfd* argument is the file descriptor of the socket to receive data from.

The *msgvec* argument is a pointer to an array of *mmsghdr* structures. The size of this array is specified in *vlen*.

The *mmsghdr* structure is defined in *<sys/socket.h>* as:

man recvmmsg

Vulnerabilidades - CVE-2016-7117

ALLELE SECURITY INTELLIGENCE

Vulnerabilidade afetou uma parte principal do kernel do Linux, adicionada em 2009 e foi somente corrigida em 2016 (7 anos)

https://nvd.nist.gov/vuln/detail/CVE-2016-7117

Vulnerabilidades - CVE-2016-7117

ALLELE SECURITY INTELLIGENCE

```
Functions calling this function: __sys_recvmmsg

  File         Function           Line
0 net/compat.c compat_sys_recvmmsg  786 return __sys_recvmmsg(fd, (struct mmsghdr __user *)mmsg, vlen,
1 net/compat.c compat_sys_recvmmsg  791 return __sys_recvmmsg(fd, (struct mmsghdr __user *)mmsg, vlen,
2 net/compat.c [illegible]          797 datagrams = __sys_recvmmsg(fd, (struct mmsghdr __user *)mmsg, vlen,
3 net/socket.c SYSCALL_DEFINE5     2420 return __sys_recvmmsg(fd, mmsg, vlen, flags, NULL);
4 net/socket.c SYSCALL_DEFINE5     2425 datagrams = __sys_recvmmsg(fd, mmsg, vlen, flags, &timeout_sys);
```

```
Find this C symbol:
Find this global definition:
Find functions called by this function:
Find functions calling this function:
Find this text string:
Change this text string:
Find this egrep pattern:
Find this file:
Find files #including this file:
Find assignments to this symbol:
```

Vulnerabilidades - CVE-2016-7117

ALLELE SECURITY INTELLIGENCE

net: Fix use after free in the recvmmsg exit path

Browse files

```
The syzkaller fuzzer hit the following use-after-free:

  Call Trace:
   [<ffffffff8175ea0e>] __asan_report_load8_noabort+0x3e/0x40 mm/kasan/report.c:295
   [<ffffffff851cc31a>] __sys_recvmmsg+0x6fa/0x7f0 net/socket.c:2261
   [<     inline     >] SYSC_recvmmsg net/socket.c:2281
   [<ffffffff851cc57f>] SyS_recvmmsg+0x16f/0x180 net/socket.c:2270
   [<ffffffff86332bb6>] entry_SYSCALL_64_fastpath+0x16/0x7a
  arch/x86/entry/entry_64.S:185

And, as Dmitry rightly assessed, that is because we can drop the
reference and then touch it when the underlying recvmsg calls return
some packets and then hit an error, which will make recvmmsg to set
sock->sk->sk_err, oops, fix it.

Reported-and-Tested-by: Dmitry Vyukov <dvyukov@google.com>
Cc: Alexander Potapenko <glider@google.com>
Cc: Eric Dumazet <edumazet@google.com>
Cc: Kostya Serebryany <kcc@google.com>
Cc: Sasha Levin <sasha.levin@oracle.com>
Fixes: a2e2725 ("net: Introduce recvmmsg socket syscall")
http://lkml.kernel.org/r/20160122211644.GC2470@redhat.com
Signed-off-by: Arnaldo Carvalho de Melo <acme@redhat.com>
Signed-off-by: David S. Miller <davem@davemloft.net>
```

master v4.16 ... v4.6-rc1

Arnaldo Carvalho de Melo authored and davem330 committed on Mar 14, 2016 1 parent b6e4038 commit 34b88a68f26a75e4fded796f1a49c40f82234b7d

Vulnerabilidades - CVE-2016-7117

ALLELE SECURITY INTELLIGENCE

net: Introduce recvmmsg socket syscall

Browse files

```
Meaning receive multiple messages, reducing the number of syscalls and
net stack entry/exit operations.

Next patches will introduce mechanisms where protocols that want to
optimize this operation will provide an unlocked_recvmsg operation.

This takes into account comments made by:

. Paul Moore: sock_recvmsg is called only for the first datagram,
  sock_recvmsg_nosec is used for the rest.

. Caitlin Bestler: recvmmsg now has a struct timespec timeout, that
  works in the same fashion as the ppoll one.

  If the underlying protocol returns a datagram with MSG_OOB set, this
  will make recvmmsg return right away with as many datagrams (+ the OOB
  one) it has received so far.

. Rémi Denis-Courmont & Steven Whitehouse: If we receive N < vlen
  datagrams and then recvmsg returns an error, recvmmsg will return
  the successfully received datagrams, store the error and return it
  in the next call.

This paves the way for a subsequent optimization, sk_prot->unlocked_recvmsg,
where we will be able to acquire the lock only at batch start and end, not at
every underlying recvmsg call.

Signed-off-by: Arnaldo Carvalho de Melo <acme@redhat.com>
Signed-off-by: David S. Miller <davem@davemloft.net>
```

master v4.16 ··· v2.6.33-rc1

Arnaldo Carvalho de Melo authored and **davem330** committed on **Oct 13, 2009** 1 parent c05e85a commit a2e2725541fad72416326798c2d7fa4dafb7d337

Vulnerabilidades - CVE-2016-7117

ALLELE SECURITY INTELLIGENCE

Bug 1382268 -

Summary: CVE-2016-7117 kernel: Use-after-free in the recvmmsg exit path

Status: CLOSED ERRATA

Aliases: CVE-2016-7117

Product: Security Response
Component: vulnerability (Show other bugs)
Version: unspecified
Hardware: All Linux

Priority high **Severity** high
Target Milestone: ---
Target Release: ---
Assigned To: Red Hat Product Security
QA Contact:
Docs Contact:

URL:
Whiteboard: impact=important,public=20160314,repo...
Keywords: Security

Duplicates: ~~CVE-2017-8281~~ (view as bug list)
Depends On: ~~1382269~~ ~~1390044~~ ~~1390046~~ ~~1390047~~ ~~1390048~~ ~~1390805~~ ~~1390806~~ ~~1390807~~ ~~1390808~~ ~~1399113~~ ~~1399114~~ ~~1399115~~ ~~1399116~~ ~~1399117~~ ~~1399118~~ ~~1399119~~
Blocks: ~~1382270~~
Show dependency tree / graph

Reported: 2016-10-06 03:54 EDT [redacted]
Modified: 2017-09-07 04:53 EDT (History)
CC List: 54 users

agordeev
alekcejk
amaris
aquini
arm-mgr

See Also:
Fixed In Version:
Doc Type: If docs needed, set a value
Doc Text: A use-after-free vulnerability was found in the kernel's socket recvmmsg subsystem. This may allow remote attackers to corrupt memory and may allow execution of arbitrary code. This corruption takes place during the error handling routines within __sys_recvmmsg() function.
Clone Of:
Environment:
Last Closed: 2017-02-15 05:58:17 EST

Vulnerabilidades - CVE-2016-7117

ALLELE SECURITY INTELLIGENCE

um pouco mais que 3 meses para ser corrigida

errata-xmlrpc **2017-01-17 13:19:30 EST** **Comment 30**

This issue has been addressed in the following products:

  Red Hat Enterprise Linux 7

Via RHSA-2017:0086 https://rhn.redhat.com/errata/RHSA-2017-0086.html

Vulnerabilidades - CVE-2016-7117

ALLELE SECURITY INTELLIGENCE

Lições

Validade da Lei de Linus (Linus's Law)

"Dados olhos suficientes, todos os erros são óbvios (Given enough eyeballs, all bugs are shallow)"

Profundidade da base de código

Essa vulnerabilidade afetou uma parte principal do kernel do Linux, a chamada de sistema recvmmsg()

https://pt.wikipedia.org/wiki/Lei_de_Linus

Vulnerabilidades

ALLELE SECURITY INTELLIGENCE

CVE-2018-6554

"Memory leak in the irda_bind function in net/irda/af_irda.c and later in drivers/staging/irda/net/af_irda.c in the Linux kernel before 4.17 allows local users to cause a denial of service (memory consumption) by repeatedly binding an AF_IRDA socket."

CVE-2018-6555

"The irda_setsockopt function in net/irda/af_irda.c and later in drivers/staging/irda/net/af_irda.c in the Linux kernel before 4.17 allows local users to cause a denial of service (IAS_OBJECT use-after-free and system crash) or possibly have unspecified other impact via an AF_IRDA socket."

https://nvd.nist.gov/vuln/detail/CVE-2018-6554
https://nvd.nist.gov/vuln/detail/CVE-2018-6555

IRDA Linux Driver UAF

ALLELE SECURITY INTELLIGENCE

Vulnerabilities summary

The following advisory describes two vulnerabilities in the Linux kernel. By combining these two vulnerabilities a privilege escalation can be achieved. The two vulnerabilities are quite old and have been around for at least 17 years, quite a few long term releases of Linux have them in their kernel. While the assessment of the Linux kernel team is that they only pose a denial of service, that is incorrect, we will provide here proof that they can run code with a bit of effort and some luck (the probability of success of gaining root privileges is above 50%).

https://blogs.securiteam.com/index.php/archives/3759

Vulnerabilidades
CVE-2018-6554/CVE-2018-6555

ALLELE
SECURITY INTELLIGENCE

Lições

Vulnerabilidades lógicas

Chamar bind() múltiplas vezes

Explorabilidade

A vulnerabilidade isoladamente pode não permitir escalação de privilégios mas pode fornecer primitivas necessárias para serem usadas em conjunto com outras vulnerabilidades. Toda vulnerabilidade pode ser interessante

https://pt.wikipedia.org/wiki/Lei_de_Linus

Considerações Finais

ALLELE
SECURITY INTELLIGENCE

- Módulos de Terceiros
- Capabilities
- Carregamento automático de módulos
- Namespaces
- Changelog / Novos recursos / Beta
- Lista de discussões

OBRIGADO!

DÚVIDAS?