

Izveštaji o proceni ranjivosti

Ime i prezime: Milica Đumić

Tim:7

Datum: 27.10.2024.

Scan Tool: Nessus (10.8.3)

Test okruženje: Metasploitable3

Ranjivost 1

1. Enumeracija CVE-a

- **CVE ID: 2016-9794**

- **Opis:**

Podsistem u jezgru Linux-a, pre 4.7, sadrži race condition koja dozvoljava korisnicima da izvedu denial of service (pad sistema) ili nespecificirani uticaj korišćenjem određenih komandi. Ovo je otkriveno u podsistemu Advanced Linux Sound Architecture (ALSA).

2. CVSS skor

- **CVSS skor (numerička vrednost): 7.8**
- **Vektor: CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H**

CVSS:3.1 = Ocena se dobija korišćenjem CVSS 3.1

AV = Attack Vector: Local (Eksploatacija se može dogoditi lokalno, npr. čitanje/pisanje u lokalnom fajl sistemu)

AC = Attack Complexity: Low (Ovaj napad ne zahteva mnogo tehničkog znanja, lako ga je izvesti)

PR = Privileges Required: Low (Napadaču su potrebne privilegije i napada neosetljive resurse)

UI = User Interaction: None (Za korišćenje ranjivog sistema nije potreban niko drugi sem napadača)

S = Scope: Unchanged (Opseg ranjivosti nije promenjen)

C = Confidentiality Impact: High (Napadač dobija informacije koje bi trebale biti zaštićene, narušava se poverljivost)

I = Integrity Impact: High (Napadač može da menja podatke i fajlove, narušen integritet)

A = Availability Impact: High (Napadač može da ograniči legitimni pristup sistemu, narušena dostupnost)

- **Opravljanje:**

Posledice ovog napada su izmena i pristup zaštićenim fajlovima, potencijalno pad kompletnog sistema. Kompleksnost napada je mala, a pomenute posledice mogu biti katastrofalne po sistem tako da je to doprinelo veličini ocene za ovaj napad. CIA trijada (poverljivost, integritet i dostupnost) su osobine koje su narušene na visokom nivou, a veoma su bitne za konzistentan i bezbedan rad sistema i one takođe učestvuju u formiranju velike ocene ovog napada. Vrednost exploitability scope je 1.8 što ocenjuje lakoću i tehnička sredstva potrebna da bi se ranjivost iskoristila. Impact scope je 5.9 što nam govori da bi bila velika šteta ukoliko bi se pomenuta ranjivost uspešno iskoristila.

3. Dostupnost eksploita

- **Postoji javno dostupan exploit (Da/Ne): Ne.**
 - **Opis eksploita:**
 - **Kod eksploita (ukoliko postoji):**
-

4. Analiza uzroka (root cause)

- **Uvođenje Greške (Commit/Verzija):**

Boazen Ding je otkrio ovu ranjivost. Commit je objavljen 14.04.2016. godine i odnosi se na sve verzije jezgra Linux-a pre 4.7. Link sa tačnim commit-om je:

<https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git/commit/?id=3aa02cb664c5fb1042958c8d1aa8c35055a2ebc4>.

U funkciji `snd_pcm_period_elapsed()` se poziva funkcija `kill_fasync()`, ali van zaključavanja toka podataka. Na ovom mestu je moguće da se dogodi trka do podataka (race condition). Konkretno funkcija izgleda ovako:

```

void snd_pcm_period_elapsed(struct snd_pcm_substream *substream)
{
    struct snd_pcm_runtime *runtime;
    unsigned long flags;

    if (PCM_RUNTIME_CHECK(substream))
        return;
    runtime = substream->runtime;

    snd_pcm_stream_lock_irqsave(substream, flags);
    if (!snd_pcm_running(substream) ||
        snd_pcm_update_hw_ptr0(substream, 1) < 0)
        goto _end;

#ifdef CONFIG_SND_PCM_TIMER
    if (substream->timer_running)
        snd_timer_interrupt(substream->timer, 1);
#endif
_end:
    snd_pcm_stream_unlock_irqrestore(substream, flags);
    kill_fasync(&runtime->fasync, SIGIO, POLL_IN);
}

```

- **Primer Koda (ako je primenljivo):**

Pomeranje poziva funkcije `kill_fasync()` nakon zaključavanja toka će na jednostavan način rešiti ovaj problem. Interfejs `fasync` se retko koristi, tako da ovo ne bi trebalo da ima veliki uticaj na performanse, ali bi trebalo da pokrije veliki broj slučajeva. U idealnom slučaju bi trebalo da se implementira mehanizam sinhronizacije za ispravan rad toka.

1888	1888	_end:	
1889	-		<code>snd_pcm_stream_unlock_irqrestore(substream, flags);</code>
1890	1889		<code>kill_fasync(&runtime->fasync, SIGIO, POLL_IN);</code>
	1890	+	<code>snd_pcm_stream_unlock_irqrestore(substream, flags);</code>
1891	1891	}	

5. Preporuke za mitigaciju

- **Da li je dostupan Vendor Fix ili patch (Da/Ne):** Da
- **Mitigation Strategy:**
Mitigaciju je moguće sprovesti update-ovanjem pomoću komande: **sudo apt update**. Ova komanda će povući noviju verziju paketa sa repozitorijuma.

Zatim je potrebno uraditi upgrade pomoću funkcije: **sudo apt upgrade**. Ova komanda će instalirati najnoviju verziju jezgra.

Radi provere, da li se promenila verzija, koristi se komanda: **uname -r**.

- **Alternativni fix (ukoliko ne postoji vendorski):**

Ranjivost 2

1. Enumeracija CVE-a

- **CVE ID: 2019-6974**
- **Opis:**

U jezgru Linux, pre 4.20.8, u virt/kvm/kvm_main.c u funkciji kvm_iocti_create_device postoji pogrešno rukovanje brojačem referenci, tj. race condition (trka do podataka) koji uzrokuje use-after-free.

2. CVSS skor

- **CVSS skor (numerička vrednost): 8.1**
- **Vektor: CVSS:3.1/AV:N/AC:H/PR:N/UI:N/S:U/C:H/I:H/A:H**

CVSS:3.1 = Ocena se dobija korišćenjem CVSS 3.1

AV = Attack Vector: Network (Eksploatacija se može dogoditi na mreži, napadač je potencijalno bilo ko ko koristi internet)

AC = Attack Complexity: High (Ova vrsta napada zahteva tehnička znanja i nije je jednostavno izvesti ili ponoviti)

PR = Privileges Required: None (Napadaču nisu potrebne privilegije niti pristup osetljivim dokumentima)

UI = User Interaction: None (Za korišćenje ranjivog sistema nije potreban niko drugi sem napadača)

S = Scope: Unchanged (Opseg ranjivosti nije promenjen)

C = Confidentiality Impact: High (Napadač dobija informacije koje bi trebale biti zaštićene, narušava se poverljivost)

I = Integrity Impact: High (Napadač može da menja podatke i fajlove, narušen integritet)

A = Availability Impact: High (Napadač može da ograniči legitimni pristup sistemu, narušena dostupnost)

- **Opravdanje:**

Potencijalni napadač je bilo ko sa mreže (interneta), to ranjivu komponentu izlaže velikom broju napadača i doprinosi velikoj oceni celokupnog napada. CIA trijada (poverljivost, integritet i dostupnost) su osobine koje su narušene na visokom nivou, a veoma su bitne za konzistentan i bezbedan rad sistema i one takođe učestvuju u formiranju velike ocene ovog napada. Vrednost exploitability scope je 2.2 što ocenjuje lakoću i tehnička sredstva potrebna da bi se ranjivost iskoristila. Impact scope je 5.9 što nam govori da bi bila velika šteta ukoliko bi se pomenuta ranjivost uspešno iskoristila.

3. Dostupnost eksploita

- **Postoji javno dostupan exploit (Da/Ne): Da**

- **Opis eksploita:**

Verifikovan exploit je commit-ovan 15.02.2019. i može se naći na linku

<https://www.exploit-db.com/exploits/46388>.

U datom kodu postoje 4 koraka:

1. Kreiranje uređaja koji ima referencu na objekat virtuelne mašine (VM), to je pozajmljiva referenca
2. Inicijalizacija uređaja
3. Prenosi referencu na uređaj u datoteku sa tabelom deskriptora pozivaoca
4. Poziva `kvm_get_kvm()` funkciju da pozajmljenu referencu na VM pretvori u stvarnu referencu.

Prenos vlasništva u koraku 3 ne sme da se realizuje pre nego što je pozajmljena referenca na VM postala stvarna, a to se događa u koraku 4. Posle koraka 3, napadač može da zatvori fajl, tabelu deskriptora, i da izbriše pozajmljenu referencu. Time bi napadač prouzrokovao da refcount kvm-a bude 0.

- **Kod eksploita (ukoliko postoji):**

Na jednostavan način se sprečava zloupotreba tako što se funkcija `kvm_get_kvm` premeti na poziciju koraka broj 3, tj. iznad if uslova.

```
3003  +      kvm_get_kvm(kvm);
3003  3004      ret = anon_inode_getfd(ops->name, &kvm_device_fops, dev, O_RDWR | O_CLOEXEC);
3004  3005      if (ret < 0) {
3006  +      kvm_put_kvm(kvm);
3005  3007      mutex_lock(&kvm->lock);
3006  3008      list_del(&dev->vm_node);
3007  3009      mutex_unlock(&kvm->lock);
3008  3010      ops->destroy(dev);
3009  3011      return ret;
3010  3012  }
3011  3013
3012  -      kvm_get_kvm(kvm);
3013  3014      cd->fd = ret;
3014  3015      return 0;
3015  3016  }
```

4. Analiza uzroka (root cause)

- **Uvođenje Greške (Commit/Verzija):**

Ovu ranjivost je pronašao Jonn Horn 26.01.2019. i objavljen je na sledećem linku: <https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git/commit/?id=cfa39381173d5f969daf43582c95ad679189cbc9>. Data ranjivost se odnosi na sve verzije jezgra Linux-a pre 4.20.8. On je dao konkretan predlog rešenja, koji je kasnije commit-ovan kao verifikovan exploit.

- **Primer Koda (ako je primenljivo):**

Ovo je konkretno kod kako je izgledala prvobitno funkcija.

```
static int kvm_ioctl_create_device(struct kvm *kvm,
                                  struct kvm_create_device *cd)
{
    struct kvm_device_ops *ops = NULL;
    struct kvm_device *dev;
    bool test = cd->flags & KVM_CREATE_DEVICE_TEST;
    int ret;

    if (cd->type >= ARRAY_SIZE(kvm_device_ops_table))
        return -ENODEV;

    ops = kvm_device_ops_table[cd->type];
    if (ops == NULL)
        return -ENODEV;

    if (test)
        return 0;

    dev = kzalloc(sizeof(*dev), GFP_KERNEL);
    if (!dev)
        return -ENOMEM;

    dev->ops = ops;
    dev->kvm = kvm;

    ret = ops->create(dev, cd->type);
    if (ret < 0) {
        mutex_unlock(&kvm->lock);
        kfree(dev);
        return ret;
    }
    list_add(&dev->vm_node, &kvm->devices);
    mutex_unlock(&kvm->lock);

    if (ops->init)
        ops->init(dev);

    ret = anon_inode_getfd(ops->name, &kvm_device_fops, dev, O_RDWR | O_CLOEXEC);
    if (ret < 0) {
        mutex_lock(&kvm->lock);
        list_del(&dev->vm_node);
        mutex_unlock(&kvm->lock);
        ops->destroy(dev);
        return ret;
    }

    kvm_get_kvm(kvm);
    cd->fd = ret;
    return 0;
}
```

5. Preporuke za mitigaciju

- **Da li je dostupan Vendor Fix ili patch (Da/Ne):** Da

- **Mitigation Strategy:**

Mitigaciju je moguće sprovesti update-ovanjem paketa virt pomoću komande: **virt-install**. Ukoliko ova komanda nije pronađena, potrebno je prvo instalirati virt-manager pomoću komande: **sudo apt-get install virt-manager**.

- **Alternativni fix (ukoliko ne postoji vendorski):**

Ranjivost 3

1. Enumeracija CVE-a

- **CVE ID: 2016-9754**
- **Opis:**

Funkcija `ring_buffer_resize()` u `kernel/trace/ring_buffer.c` u podsistemu za profilisanje jezgra Linux-a pre 4.6.1 pogrešno rukuje određenim računanjima sa integer vrednostima. To uzrokuje da napadač, na lokalnom nivou, stekne privilegije upisom malicioznog sadržaja u fajl `/sys/kernel/debug/tracing/buffer_size_kb`.

2. CVSS skor

- **CVSS skor (numerička vrednost): 7.8**
Vektor: CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H
CVSS:3.1 = Ocena se dobija korišćenjem CVSS 3.1
AV = Attack Vector: Local (Eksploatacija se može dogoditi lokalno, npr. čitanje/pisanje u lokalnom fajl sistemu)
AC = Attack Complexity: Low (Ovaj napad ne zahteva mnogo tehničkog znanja, lako ga je izvesti)
PR = Privileges Required: Low (Napadaču su potrebne privilegije i napada neosetljive resurse)
UI = User Interaction: None (Za korišćenje ranjivog sistema nije potreban niko drugi sem napadača)
S = Scope: Unchanged (Opseg ranjivosti nije promenjen)
C = Confidentiality Impact: High (Napadač dobija informacije koje bi trebale biti zaštićene, narušava se poverljivost)
I = Integrity Impact: High (Napadač može da menja podatke i fajlove, narušen integritet)
A = Availability Impact: High (Napadač može da ograniči legitimni pristup sistemu, narušena dostupnost)
- **Opravdanje:**
Posledice ovog napada su izmena i pristup zaštićenim fajlovima, potencijalno pad kompletnog sistema. Kompleksnost napada je mala, a pomenute posledice mogu biti katastrofalne po sistem tako da je to doprinelo veličini ocene za ovaj napad. CIA trijada (poverljivost, integritet i dostupnost) su osobine koje su narušene na visokom nivou, a veoma su bitne za konzistentan i bezbedan rad sistema i one takođe učestvuju u formiranju velike ocene ovog napada. Vrednost exploitability scope je 1.8 što ocenjuje lakoću i tehnička sredstva potrebna da bi se ranjivost iskoristila. Impact scope je 5.9 što nam govori da bi bila velika šteta ukoliko bi se pomenuta ranjivost uspešno iskoristila.

3. Dostupnost eksploita

- Postoji javno dostupan exploit (Da/Ne): Ne
 - Opis eksploita:
 - Kod eksploita (ukoliko postoji):
-

4. Analiza uzroka (root cause)

- Uvođenje Greške (Commit/Verzija):

Ranjivost je objavio Steven Rostedt 13.05.2016. godine, odnosi se na sve verzije jezgra Linux-a pre 4.6.1. Commit se nalazi na linku

<https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git/commit/?id=59643d1535eb220668692a5359de22545af579f6>. Ukoliko bi došlo do toga da je u funkciji

ring_buffer_resize() vrednost size veća od max_long – buf_page_size tada bi DIV_ROUND_UP vraćala vrednost 0 za broj potrebnih buffer-a. To dovodi do neupešne logike i sruši rad kernela (denial of service). Ovo je objašnjenje koda:

Vrednost BUF_PAGE_SIZE= 4080.

Funkcija DIV_ROUND_UP(a, b) = (a+b-1)/b.

Vrednost za size = 18446744073709547520 (18014398509481980 vrednost /sys/kernel/debug/tracking/buffer_size_kb koju tracking_entries_write() pretvori u bite).

$size = DIV_ROUND_UP(size, BUF_PAGE_SIZE) = (18446744073709547520 + 4080 - 1) / 4080 = 4521260802379792$

$size = size * BUF_PAGE_SIZE = 4521260802379792 * 4080 = 18446744073709551599$

$size < 2 * BUF_PAGE_SIZE = 4521260802379792 < 16648400 = false$ (ne ulazimo u if granu)

$nr_page = DIV_ROUND_UP(size, BUF_PAGE_SIZE) = (18446744073709551599 + 4080 - 1) / 4080 = 3823 / 4080 = 0$ (ovde je nastala greška)

Objašnjenje leži u tome da se za vrednost unsigned long prevaziđe vrednost u bitima tako da se dobije ostatak koji preliva preko vrednosti ovog ograničenja. Na taj način dobijemo da je potreban broj buffer-a 0.

- **Primer Koda (ako je primenljivo):**

U suštini nije ni potrebno 2 puta raditi izračunavanje `DIV_ROUND_UP`, to je rezultat istorijskih izmena koda. Kod je potrebno izmeniti tako što bi se vrednost `nr_pages` računala na početku, njena vrednost mora biti minimalno 2 i zatim se vrednost promenljive `size` računa na kraju.

1660	-	<code>size = DIV_ROUND_UP(size, BUF_PAGE_SIZE);</code>
1661	-	<code>size *= BUF_PAGE_SIZE;</code>
1660	+	<code>nr_pages = DIV_ROUND_UP(size, BUF_PAGE_SIZE);</code>
1662	1661	
1663	1662	<code>/* we need a minimum of two pages */</code>
1664	-	<code>if (size < BUF_PAGE_SIZE * 2)</code>
1665	-	<code>size = BUF_PAGE_SIZE * 2;</code>
1663	+	<code>if (nr_pages < 2)</code>
1664	+	<code>nr_pages = 2;</code>
1666	1665	
1667	-	<code>nr_pages = DIV_ROUND_UP(size, BUF_PAGE_SIZE);</code>
1666	+	<code>size = nr_pages * BUF_PAGE_SIZE;</code>
1668	1667	
1669	1668	<code>/*</code>
1670	1669	<code>* Don't succeed if resizing is disabled, as a reader might be</code>

5. Preporuke za mitigaciju

- **Da li je dostupan Vendor Fix ili patch (Da/Ne):** Da.
- **Mitigation Strategy:**

Mitigaciju je moguće sprovesti update-ovanjem pomoću komande: **sudo apt update**. Ova komanda će povući noviju verziju paketa sa repozitorijuma.

Zatim je potrebno uraditi upgrade pomoću funkcije: **sudo apt upgrade**. Ova komanda će instalirati najnoviju verziju jezgra.

Radi provere, da li se promenila verzija, koristi se komanda: **uname -r**.

- **Alternativni fix (ukoliko ne postoji vendorski):**