Česká zemědělská univerzita v Praze

Provozně ekonomická fakulta

Katedra informačního inženýrství

Teze k bakalářské práci

Útoky SQL Injection

Jan Zamazal

Souhrn

Bakalářská práce se zabývá tématem útoků typu SQL Injection na webové aplikace napojené na databázový systém a obranou proti nim. V teoretické části předkládá stručnou historii jazyka SQL a základy práce s SQL databázemi. Dále obsahuje přehled možných typů napadení webové aplikace způsobem SQL Injection. Rozebírá a porovnává způsoby ochrany proti tomuto typu útoku. V praktické části testuje reálnou hrozbu užitím popsaných postupů na nezabezpečené webové aplikaci. Je zde také navržena nejvhodnější ochrana WWW aplikace, která je implementována a testováním je ověřena její funkčnost v praktickém prostředí.

Klíčová slova: SQL Injection, webová aplikace, útok na databáze, databázové systémy, WWW aplikace, SQL, zabezpečení databáze, zneužití dat

1 Úvod

Od prvního zveřejnění možného nebezpečí spojeného s možným útokem typu SQL Injection uplynulo již přes deset let, stále je však napříč internetem možné nalézt vysoké množství aplikací, které jsou veřejně přístupné a nemají implementovánu ochranu proti tomuto typu útoku.

SQL Injection je technika, při které útočník využije uživatelských vstupů aplikace pro neoprávněný přístup a nakládání s daty uložených v SQL databázi napojené na aplikaci. Jedná se především o HTML formuláře nebo parametry v adresním řádku, které je možné měnit. Útočník může zadat do těchto vstupů parametry, které aplikace neočekává a které mohou přímo změnit chování aplikace. Pokud není program zabezpečen, uživatelské vstupy jsou přímo vkládány do dotazu SQL, který je odeslán databázi k vykonání. Díky tomu může útočník pozměnit dotaz tak, aby se aplikace zachovala podle jeho vůle, bez ohledu na svůj zdrojový kód. Může si takto zobrazit informace, ke kterým nemá oprávnění, data v databázi upravit či zcela vymazat.

Přes vysoké nebezpečí, které tato zranitelnost představuje, je stále velké množství programátorů či administrátorů WWW aplikací, kteří se rozhodnou ignorovat poznatky o tomto tématu zveřejněné, nebo o dané problematice ani nevědí, neboť se nezajímají o zabezpečení svých programů a vyvíjí je pouze z funkčního hlediska, ovšem bez ohledu na bezpečnost. Většina z nich předpokládá, že jejich aplikace nebude pro potenciálního útočníka lukrativní a k napadení nedojde. Neuvědomují si, že kvůli této zranitelnosti je možné nejen zcizit jakákoliv citlivá data uložená v databázi, ale také ovládnout celý stroj, na kterém webová aplikace běží, a využít ho k útočení na další počítače, či k dalším ilegálním činnostem.

2 Cíle a metodika

2.1 Cíle práce

Tato práce se zabývá tématem zabezpečení databází napojených na webové aplikace proti útokům typu SQL Injection. V teoretické častí si klade za cíl stručně shrnout historii jazyka SQL a na příkladech prezentovat základní pracovní postupy s databází SQL. Především základy práce s těmito databázemi jsou důležité pro pochopení teoretických principů problematiky tématu této práce. Cílem druhé poloviny teoretické části práce je zobecnit možnosti útočníka při napadení SQL databáze a na příkladech popsat teorii útoku na nezabezpečenou webovou aplikaci. Zároveň si práce klade za úkol zmapovat současnou situaci v této problematice, jelikož je stále dostupné velké množství aplikací bez jakékoliv ochrany.

Praktická část této práce se zaměřuje na navržení účinného řešení tohoto problému. Jejím cílem je navrhnout zabezpečení webových aplikací proti útokům typu SQL Injection, implementovat nejdůležitější funkce do testovací aplikace a otestovat toto řešení praktickými testy v běžném prostředí. V závěru jsou výsledky testování okomentovány a zobecněny pro další použití v praxi.

2.2 Metodika práce

Metodika práce je založena na studiu a analýze zdrojů týkajících se této problematiky. Takto získané informace a techniky jsou využity v praktické části při návrhu řešení. Syntézou teoretických poznatků a výsledků dosažených v praktické části práce bude zobecněn závěr této práce.

3 Výsledky a diskuse

Cílem práce bylo použít teoretické poznatky pro navržení řešení zabezpečení WWW aplikace proti napadení. Navržené řešení zabezpečení, bylo stejně jako původní nezabezpečená aplikace, otestováno v reálném prostředí. Porovnání výsledků testů dokazuje možnost snadno proniknout do databázového systému a zneužít data v něm uložená, pokud není aplikace zabezpečena. Preferované zabezpečení používá metodu svazování proměnných knihovnou *PHP Data Objects*. Toto řešení je účinné a testováním byla ověřena jeho funkčnost. Po implementaci tohoto zabezpečení již nebylo možné zcizit data uložená v databázi. Použití této knihovny také mění veškerou práci aplikace s databází a zvyšuje nejen bezpečnost, ale i robustnost aplikace v podobě univerzálnosti umožňující bez změny metod a funkcí používat program s širokým portfoliem databázových systémů.

Při vývoji nových aplikací je možné doporučit použití této metody zabezpečení již od začátku vývoje. Pro již spuštěné projekty, které nepoužívají tento typ zabezpečení, lze doporučit co nejdříve dané části aplikace aktualizovat a tuto ochranu implementovat. Ve stávajícím stavu hrozí vysoké nebezpečí neoprávněného vniknutí do databáze a zneužití dat.

4 Seznam použitých zdrojů

1. CLARKE, Justin. SQL injection attacks and defense. Waltham, MA: Elsevier, c2012, xvviii, 547 p. ISBN 9781597499637.
2. GROFF, James R a Paul N WEINBERG. *SQL: kompletní průvodce*. Vyd. 1. Brno: CP Books, 2005, 936 s. ISBN 80-251-0369-2.
3. ANSI X3.135:1986. Database Language SQL. Washington: ANSI, 1986
4. CODD, E. F. A relational model of data for large shared data banks. *Communications of the ACM*. 1970, vol. 13, issue 6, s. 377-387. DOI: 10.1145/362384.362685.
5. NT Web Technology Vulnerabilities. *Phrack*. 1998, vol. 8, issue 54. Dostupné z: http://phrack.org/issues/54/8.html
6. HTTP Methods: GET vs. POST. *W3Schools.com* [online]. [cit. 2015-8-28]. Dostupné z: http://www.w3schools.com/tags/ref_httpmethods.asp
7. VESELÝ, Ondřej. Ochrana proti SQL Injection v PHP. *Birknet* [online]. 2013 [cit. 2015-02-15]. Dostupné z: http://www.birknet.eu/node/ochrana-proti-sql-injection-v-php/