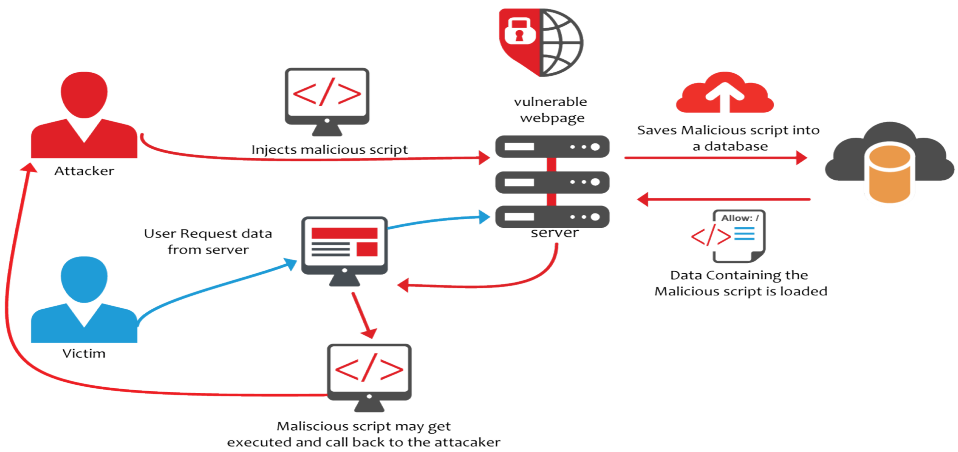
ROZŠÍRENÉ ZADANIE DIPLOMOVEJ PRÁCE

Názov práce:	Zraniteľnosti druhého rádu vo webových aplikáciách
Autor:	Bc.Oleh Sova
Vedúci práce:	RNDr. JUDr. Pavol Sokol, PhD.
Konzultant práce:	Mgr. Terézia Mezešová
Školiace pracovisko:	ÚINF - Ústav informatiky

Ciele práce:

1. Analyzovať zraniteľnosti druhého rádu vo webových aplikáciách a možnosti ich detekcie.
2. Analyzovať a porovnať aktuálne prístupy k detekcii zraniteľností druhého rádu vo webových aplikáciách.
3. Návrh, implementácia a otestovanie nástroja pre testovanie zraniteľností druhého rádu pre webové aplikácie vytvorené v jazyku Javascript.

Diplomová práca sa zaoberá výskumom zraniteľností druhého rádu vo webových aplikáciách. Zraniteľnosti druhého rádu vznikajú vtedy, keď aplikácia ukladá vstup do databázy bez overenia voči škodlivým príkazom a následne sa vstup z databázy zobrazí používateľom aplikácie, alebo sa použije pri vnútornej operácií. Jednou z takých zraniteľností je uložený cross-site scripting (XSS).



Obr1. Príklad zraniteľnosti druhého rádu využitej pri XSS útoku

Príklad zraniteľnosti druhého rádu je zobrazený na Obr. 1. Útočník (napr. hacker) nahrá škodlivý skript do aplikačného servera, ktorý sa potom uloží do databázy. Po následných požiadavkách servera na databázu, databáza bude vracať serveru dáta obsahujúce škodlivý skript. Používateľ (obeť) pri interakcii s aplikáciou (teda prístupom na aplikačný server) prijme dáta,

ktoré sú napadnuté škodlivým skriptom na svojom zariadení. Následne útočník (napr.hacker) môže získať kontrolu nad dátami obete. V niektorých prípadoch dokáže prevziať server. To v závisi od toho, na čo bol určený (pôvodne navrhnutý) škodlivý skript.

Cieľom tejto záverečnej práce je skúmať zraniteľnosti druhého stupňa, najmä však pre webové aplikácie. Na základe analýzy možností detekcie a aktuálnych riešení vytvoriť automatický systém kontroly štruktúry základného kódu pre programovací jazyk JavaScript.

Na základe výskumu v tejto oblasti môžeme určiť, že základný problém v zdrojovom kóde webových stránok (pri napadnutiach hackerov), je v neprítomnosti funkcie blokovania hrozieb (od XSS útokov a SQL injection) alebo v neprítomnosti funkcie kontroly vstupných údajov.

Súčasne pri realizácii automatizovaného systému kontroly zdrojového kódu pre zraniteľnosť druhého stupňa vzniká problém v štruktúre logiky kódu. Ak vo výslednom kóde bude chyba v štruktúre kódu, túto chybu je možné označiť v zodpovedajúcom okne na stránke systémovej analýzy.

Výsledkom analýzy zdrojového kódu bude zoznam nájdených chýb a zraniteľností v zdrojovom kóde. Analýza poskytne údaje o slabých miestach (popis slabého miesta, upozornenie na nebezpečenstvo, časť kódu so slabým miestom).

Literatúra:

[1] DAHSE, Johannes; HOLZ, Thorsten. Static Detection of Second-Order Vulnerabilities in Web Applications. In: USENIX Security Symposium. 2014. p. 989-1003.

[2] OLIVO, Oswaldo; DILLIG, Isil; LIN, Calvin. Detecting and exploiting second order denial-of-service vulnerabilities in web applications. In: Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security. ACM, 2015. p. 616-628.

[3] MUELLER, John Paul. Security for Web Developers: Using JavaScript, HTML, and CSS. " O'Reilly Media, Inc.", 2015

[4] BOULANGER, Jean-Louis (ed.). Static analysis of software: The abstract interpretation. John Wiley & Sons, 2013

[5] SCHOLTE, T., ROBERTSON, W., BALZAROTTI, D., AND KIRDA, E. An Empirical Analysis of Input Validation Mechanisms in Web Applications and Languages. In ACM Symposium On Applied Computing (SAC) (2012).