УДК 004.056
Дмитришин С. – ст. гр. PI-21
Тернопільський державний технічний університет імені Івана Пулюя

ЗАХИСТ WEB СЕРВІСІВ ВІД XSS

Науковий керівник: асистент Луцків А.М.

Вразливість, типу XSS (XSS або Cross Site Scripting «Міжсайтові скрипти») , виникає за умов, коли дані введені користувачем виводяться без належної фільтрації в текст HTML.

Прикладом може бути ситуація, коли дані, відправлені одним користувачем без фільтрації виводяться іншим користувачам. Типовими системами такого типу являються чати, форуми, книги відгуків тощо. Як наслідок зловмисник може скомпрометувати користувачів певного сервісу використовуючи їх персональні дані, ідентифікатори сесій і т.п. Простий приклад: *<body onLoad="alert(document.cookie)">*, якщо зловмиснику вдасться розмістити цей тег наприклад в гостьовій книзі, то кожен користувач що її відкриє буде бачити повідомлення з власними “куками” (*Cookies*). Зауважимо, що тег буде опрацьований браузером навіть тоді, якщо в документі він зустрічається двічі.

Самодостатній *Cross-Site Scripting* – це відносно новий спосіб реалізації XSS атак, його набагато важче виявити і відфільтрувати. Тому що старі методи, які фільтрують класичний XSS в цьому випадку не діють.

Типовим прикладом такого типу атаки може служити URI:

data:text/html;base64, PHNjcmlwdD5hbGVydCgndGVzdCcpPC9zY3JpcHQ+

при посиланні на цю URI браузером буде виведено повідомлення «test».

Для реалізації атаки з самодостатнім XSS використовується протокол data (RFC2397), який призначений для передачі даних в URI, після чого йде визначення типу MIME, і варіанту запису: base64 - що ускладнює процес фільтрування. Шкідливий код закодований по base64, у вищезгаданому прикладі було використано наступний уривок скрипта: *<script>alert('test')</script>*.

Не менш небезпечними є поєднання атак SQL Injection і XSS (або SiXSS) – це приклад пасивної XSS атаки, також реалізація XSS можлива з використанням Macromedia Flash, XSS як спосіб «дефейсу»(deface) сайтів(активна XSS), Visual Basic – як основна мова сценаріїв, зміна кодування інформації (для ускладнення фільтрації), тощо.

Наведемо основні способи захисту від XSS атак.

Найефективнішим методом було і залишається фільтрування будь-якої вхідної чи вихідної інформації. Деколи помилково вважають що захистити себе можна фільтруючи тільки спеціальні символи чи оператори JavaScript – це не так, і цьому є ряд доказів.

Також перспективним методом захисту клієнтських “куків” (Cookies) вважається нововведення від Microsoft - директива *HttpOnly*. Нажаль, вона реалізована тільки в MSIE починаючи з SP1 для версії 6, хоча і тут є ряд недоліків.

Література

- RFC 2397 [ftp://ftp.rfc-editor.org/in-notes/rfc2397.txt]
- http://www.securitylab.ru/
- http://www.securityfocus.com/