УДК 004.056.53

Захар Георгійович ДЕМИДОВ,
науковий співробітник науково-дослідної лабораторії захисту інформації та кібербезпеки Харківського національного університету внутрішніх справ

ОСНОВНІ ВИДИ КІБЕРАТАК НА WEB-САЙТИ

Основне правило при створенні web-ресурсу - ніколи не довіряти даним, введеним користувачем. Про це знають багато web-програмістів, але не всі до кінця дотримуються цього, з думкою: "Що там на цьому сайті ламати, та кому це потрібно". Але найчастіше на таких сайтах і вчаться зламувати адмінку й бази даних. Винятки становлять сайти, зроблені на основі якогось сучасного фреймворку (Наприклад: LARAVEL, SYMFONY, YII 2.0), бо там система безпеки вже протестована досвідченими фахівцями в цьому напрямку [1].

У інтернет-мережі зараз маса інформації щодо різних видів хакерських атак на web-сайти, але давайте їх ще раз узагальнимо, пояснимо та визначимо способи боротьби та запобігання.

SQL ін'єкція (англ. SQL injection) - один з найпоширеніших способів злому web-ресурсів, програм, які працюють з даними. Його мета - впровадити в запит сторонній, чужий SQL-код.

Впровадження SQL запитів, в залежності від виду використовуваної СУБД і умов, дає можливість хакерам виконати сторонній запит до бази даних (наприклад, видалити, прочитати, додати або змінити дані з будь-яких таблиць бази), отримати можливість працювати з локальними файлами і виконання своїх команд на сервері, що атакується.

Здебільшого, такий вид нападу можна бачити на прикладі сайтів, де використовуються параметри командного рядка (змінні URL), щоб побудувати SQL-запити до баз даних, без відповідної перевірки.

Для виключення таких випадків потрібно фільтрувати лапки та інші спецсимволи, вони можуть порушити логіку вашого запиту. Також, коли у вас є число, обов'язково явно приводити його до числового типу.

Деякі фахівці радять застосовувати для цього спеціальні конструктори SQL-запитів, які самі забезпечують необхідний поділ запиту і даних.

PHP ін'єкція – теж спосіб злому веб-сайтів, які написані на PHP. Основна думка - впровадження свого розробленого сценарію в код на серверній стороні ресурсу, що призводить до виконання сторонніх команд. Відомо - в багатьох розповсюджених движках та на форумах, які працюють на PHP (найчастіше це застарілі версії), є не зовсім продумані модулі та конструкції з уразливими місцями. Хакери шукають ці уразливості, аналізують їх, та користуються ними.

XSS (Cross Site Scripting, "міжсайтовий скриптинг") являє собою атаку, при якій зловмисник публікує на сайті, що атакується, скрипт, який виконується у користувачів при відкритті ними сторінок. Оскільки цей скрипт виконується в браузері у користувача, то він має доступ до інформації в його cookie, й може виробляти дії на сайті від імені користувача (якщо той "залогінився"), наприклад, писати, читати та видаляти повідомлення.

Розрізняють активні та пасивні XSS атаки. При першому типі нападу шкідливий скрипт зберігається на сервері та починає свою діяльність при завантаженні сторінки сайту в браузері клієнта. При другому виді атаки скрипт не зберігається на сервері, а шкідливий вплив починає виконуватися тільки в

разі будь-якого дії користувача, наприклад, при натисканні на сформоване посилання.

Основним способом протидії XSS-атак є фільтрація надходячих ззовні та публікуємих на сайті даних. Як правило, достатньо замінювати символи "<" та ">" на "& lt;" та "& gt;" відповідно (php-функція htmlspecialchars), при цьому введений відвідувачем текст втрачає HTML-оформлення, а скрипти, що містяться в ньому, втрачають шкідливість.

CSRF (англ. Cross Site Request Forgery - «міжсайтова підробка запиту», також відома як XSRF) - вид атак на відвідувачів веб-сайтів, який використовує недоліки протоколу HTTP. Коли користувач заходить на сайт, який створив зловмисник, від користувача таємно відправляється запит на інший сервер (наприклад, на сервер платіжної системи), який здійснює якусь шкідливу операцію (наприклад, переказ грошей на рахунок зловмисника). Для здійснення даної атаки жертва повинна бути автентифікована на тому сервері, на який відправляється запит, і цей запит не повинен вимагати будь-якого підтвердження з боку користувача, яке не може бути проігноровано або підроблено атакуючим скриптом.

На відміну від інших вразливостей CSRF виникає не в результаті помилок програмування, а є нормальною поведінкою Web-сервера і браузера. Тобто більшість сайтів, які використовують стандартну архітектуру, уразливі «за замовчуванням».

Спосіб боротьби - питати введення CAPTCHA при здійсненні критичних дій на зразок зміни пароля або переказу грошей (як зроблено в WebMoney).

DOS

Відокремлено в ряду загроз безпеки стоять вразливості DOS - Denial Of Service (відмова в обслуговуванні). Як правило, до цього класу нападів належать події, описувані в новинах "Хакери атакували сайт X, порушивши його роботу. Сайт не працював протягом Y годин". Тобто це саме "атака", а не "злом". На сервер виробляються запити, які він не може (не встигає) обробити, в результаті чого він не встигає обробити запити звичайних відвідувачів і виглядає для них як непрацюючий.

Ці атаки не мають на меті викрасти дані з бази, але можуть допомогти почати інші види атак, тобто звільнити шлях. Наприклад, деякі програми через помилки в своєму коді можуть викликати виняткові ситуації, і при відключенні сервісів здатні виконувати код, наданий зловмисником. Або атаки ла-

винного типу, коли сервер не може обробити величезну кількість вхідних пакетів.

При бажанні (бюджеті) можна «завалити» будь-який сервер. Обмежили кількість звернень з однієї IP-адреси? Отримайте DDOS (distributed - розподілений), коли звернення проводяться не з одного комп'ютера. DDoS використовується там, де звичайний DoS неефективний. Для цього кілька комп'ютерів об'єднуються, далі кожен створює DoS атаку на систему жертви.

Проти DOS атаки неможливо захиститися на 100%, але можна зробити обмеження на кількість спроб логіна з однієї IP-адреси в деяку кількість часу. Наприклад - не більше 5 в 10 хвилин. При вичерпанні показувати повідомлення "почекайте" або пропонувати ввести CAPTCHA. Деякі системи просять ввести CAPTCHA взагалі при кожній спробі логіна.

Список використаних джерел:

1. Лучшие PHP фреймворки в 2017 году. URL: blog.liveedu.tv/список-лучших-php-фреймворков/ (дата звернення: 21.10.2017).
2. Виды взломов сайтов и их предотвращение. URL: http://captcha.ru/articles/antihack/ (дата звернення: 10.10.2017).
3. Виды хакерских атак. URL: https://sites.google.com/site/hakerskieataki/home/vidy-hakerskih-atak (дата звернення: 10.10.2017).

Одержано 27.10.2017