

ПІДМІНА ВИДАЛЕНОГО СЕРВЕРУ

Лабораторна робота

Мета: Дослідити особливості підміни серверу.

Індивідуальне завдання:

Використовуючи другий проект роботи з ліцензійними ключами:

- створити додаток "що приматиме посилання", адресовані до оригінального серверу та повертає завжди позитивну відповідь
- зробити підміну а) dns адреси, б) ip адреси, таким чином, щоб інформація про верифікацію ліцензійного ключа надходила не на валідний сервер, а на підроблений

Запропонувати варіанти уникнення цих вразливостей.

ХІД РОБОТИ

Проксі-сервер (від англ. Proxy - представник, уповноважений; часто просто проксі, сервер-посередник) - проміжний сервер (комплекс програм) в комп'ютерних мережах, що виконує роль посередника між користувачем і цільовим сервером (при цьому про посередництво можуть хтозна, так і не знати обидві сторони), що дозволяє клієнтам як виконувати непрямі запити (приймаючи і передаючи їх через проксі-сервер) до інших мережних служб, так і отримувати відповіді. Спочатку клієнт підключається до проксі-сервера і запитує який-небудь ресурс (наприклад e-mail), розташований на іншому сервері. Потім проксі-сервер або підключається до вказаного серверу і отримує ресурс у нього, або повертає ресурс із власного кешу (у випадках, якщо проксі має свій кеш). У деяких випадках запит клієнта або відповідь сервера може бути змінений проксі-сервером в певних цілях. Проксі-сервер дозволяє захищати комп'ютер клієнта від деяких мережних атак і допомагає зберігати анонімність клієнта, але також може використовуватися шахраями для приховування адреси сайту, викритого в шахрайстві, зміни вмісту цільового сайту (підміна), а також перехоплення запитів самого користувача.

Лістинг функції перенаправлення запиту на іншу адресу:

```
private void forwardRequest(String method, HttpServletRequest req,
HttpServletRequest resp) {
    final boolean hasoutbody = (method.equals("POST"));

    try {
        final URL url = new URL(GlobalConstants.CLIENT_BACKEND_HTTPS
            + req.getRequestURI()
            + (req.getQueryString() != null ? "?" + req.getQueryString() : ""));
        HttpURLConnection conn = (HttpURLConnection) url.openConnection();
        conn.setRequestMethod(method);

        final Enumeration<String> headers = req.getHeaderNames();
        while (headers.hasMoreElements()) {
            final String header = headers.nextElement();
            final Enumeration<String> values = req.getHeaders(header);
            while (values.hasMoreElements()) {
                final String value = values.nextElement();
                conn.addRequestProperty(header, value);
            }
        }
        conn.setUseCaches(false);
        conn.setDoInput(true);
        conn.setDoOutput(hasoutbody);
        conn.connect();

        final byte[] buffer = new byte[16384];
        while (hasoutbody) {
```

```
        final int read = req.getInputStream().read(buffer);
        if (read <= 0) break;
        conn.getOutputStream().write(buffer, 0, read);
    }

    resp.setStatus(conn.getResponseCode());
    for (int i = 0; ; ++i) {
        final String header = conn.getHeaderFieldKey(i);
        if (header == null) break;
        final String value = conn.getHeaderField(i);
        resp.setHeader(header, value);
    }

    while (true) {
        final int read = conn.getInputStream().read(buffer);
        if (read <= 0) break;
        resp.getOutputStream().write(buffer, 0, read);
    }
} catch (Exception e) {
    e.printStackTrace();
}
}
```

Після перенаправлення необхідно на стороні серверу обробити запит та відправити інформацію про успішну валідацію.

Одним з варіантів уникнення подібних ситуацій є перевірка ір адреси з якої надіслано повідомлення про валідацію та додаванням токена.

Висновки: в ході лабораторної роботи було досліджено особливості серверу.