Common Vulnerabilities and Exposures

2014-3470

Pavel Raav
Tartu 2015
Introduction

On 28th May 2014 an issue was reported to OpenSSL by Felix Gröbert and Ivan Fratric: OpenSSL clients who enabled anonymous Elliptic Curve Diffie Hellman cipher suites were subject to a denial of service attack. The problem affected the following OpenSSL versions: 0.9.8, 1.0.0, 1.0.1.

OpenSSL is a cryptography library that provides an open source implementation of the Secure Sockets Layer (SSL) and Transport Layer Security (TLS) protocols. SSL and TLS are cryptographic protocols that are used to provide communications security over a computer network. They encrypt the data of network connections in the application layer.

Elliptic Curve Diffie Hellman (ECDH) is an Elliptic Curve variant of the standard Diffie Hellman algorithm.

The Diffie-Hellman algorithm makes it possible for two communicating parties to agree upon a shared secret between them which can be used as the basis for some encryption key to be used for further communication. An anonymous Diffie-Hellman uses Diffie-Hellman, but without authentication.

Elliptic Curve algorithms are used to reduce key size and thus increase speed.

A cipher suite is a named combination of authentication, encryption, message authentication code (MAC) and key exchange algorithms used to negotiate the security settings for a network connection using the TLS/SSL network protocol.

A denial of service (DoS) attack is a malicious attempt to make a server or a network resource unavailable to users, usually by temporarily interrupting or suspending the services of a host connected to the Internet.

The vulnerability

The vulnerability allows attackers to cause a server crash by triggering a NULL certificate value using NULL pointer dereference programming error in the ssl3_send_client_key_exchange function in s3_clnt.c file in the OpenSSL library. This enables the attacker to disrupt the availability of services, which can result in a loss of productivity and interrupt critical operations.
The `ssl3_send_client_key_exchange` function is used in the client key exchange. There are various different methods for exchanging keys. If anonymous ECDH has been chosen as the method, then the attacker (client) is able to send a NULL value in place of a certificate. This can happen because there is no authentication when using anonymous ECDH. This enables the attacker to send a certificate without a digital signature from a certificate authority, which normally would attest that the key certificate is valid.

In this case the NULL pointer dereference is caused by triggering the NULL certificate value in `sess_cert` as shown below:

```c
if (s->session->sess_cert->peer_ecdh_tmp != NULL)
```

### The fix

Since the problem is caused by not handling the possibility of a certificate value being NULL, it is possible to completely avoid the problem by actually handling a NULL certificate value with a simple `if` statement, thus avoiding the NULL pointer dereference and the subsequent client crash. A fix made by Stephen Henson of the OpenSSL team is provided below:

```c
if (s->session->sess_cert == NULL)
{
    ssl3_send_alert(s,SSL3_AL_FATAL,SSL_AD_UNEXPECTED_MESSAGE);
    SSLerr(SSL_F_SSL3_SEND_CLIENT_KEY_EXCHANGE,SSL_R_UNEXPECTED_MESSAGE);
    goto err;
}
```

The fix forces the program to check if the `sess_cert` value is NULL, before doing anything else. Because of that the vulnerability is permanently eliminated.

If the value is indeed NULL, the program sends an alert that there has been an error in the client key exchange (`ssl3_send_alert(...)`), generates an error file (`SSLerr(...)`) and aborts the exchange by triggering an error state (`goto err`).

```c
err:
    return(0);
```

3
Referenced materials

1 https://github.com/openssl/openssl/blob/a5362db4603910b1bb978163e0e7dc8890727300/ssl/s3_clnt.c
2 https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-3470
4 http://www.openssl.org/news/secadv_20140605.txt
5 http://whatis.techtarget.com/definition/OpenSSL
6 http://en.wikipedia.org/wiki/Transport_Layer_Security
7 http://en.wikipedia.org/wiki/Cipher_suite
8 https://wiki.openssl.org/index.php/Diffie_Hellman
12 https://github.com/openssl/openssl/commit/a5362db4603910b1bb978163e0e7dc8890727300
13 https://www.openssl.org/docs/crypto/err.html