Message-ID: <CABU6YOa0q0=aWxSqcWN7AbNAZS1Yx6nuqBUkAfsfbzJKywwBhw@mail.gmail.com>
Date: Wed, 25 Aug 2021 21:20:09 +0100
From: Mark J Cox <mark@...nssl.org>
To: oss-security@...ts.openwall.com
Subject: OpenSSL SM2 Decryption Buffer Overflow (CVE-2021-3711), Read buffer
 overruns processing ASN.1 strings (CVE-2021-3712)

OpenSSL Security Advisory [24 August 2021]
==========================================

SM2 Decryption Buffer Overflow (CVE-2021-3711)
==============================================

Severity: High

In order to decrypt SM2 encrypted data an application is expected to call the
API function EVP_PKEY_decrypt(). Typically an application will call this
function twice. The first time, on entry, the "out" parameter can be NULL and,
on exit, the "outlen" parameter is populated with the buffer size required to
hold the decrypted plaintext. The application can then allocate a sufficiently
sized buffer and call EVP_PKEY_decrypt() again, but this time passing a non-NULL
value for the "out" parameter.

A bug in the implementation of the SM2 decryption code means that the
calculation of the buffer size required to hold the plaintext returned by the
first call to EVP_PKEY_decrypt() can be smaller than the actual size required by
the second call. This can lead to a buffer overflow when EVP_PKEY_decrypt() is
called by the application a second time with a buffer that is too small.

A malicious attacker who is able present SM2 content for decryption to an
application could cause attacker chosen data to overflow the buffer by up to a
maximum of 62 bytes altering the contents of other data held after the
buffer, possibly changing application behaviour or causing the application to
crash. The location of the buffer is application dependent but is typically
heap allocated.

OpenSSL versions 1.1.1k and below are affected by this issue. Users of these
versions should upgrade to OpenSSL 1.1.1l.

OpenSSL 1.0.2 is not impacted by this issue.

OpenSSL 3.0 alpha/beta releases are also affected but this issue will be
addressed before the final release.

This issue was reported to OpenSSL on 12th August 2021 by John Ouyang. The fix
was developed by Matt Caswell.

Read buffer overruns processing ASN.1 strings (CVE-2021-3712)
=============================================================

Severity: Moderate

ASN.1 strings are represented internally within OpenSSL as an ASN1_STRING
structure which contains a buffer holding the string data and a field holding
the buffer length. This contrasts with normal C strings which are repesented as
a buffer for the string data which is terminated with a NUL (0) byte.

Although not a strict requirement, ASN.1 strings that are parsed using OpenSSL's
own "d2i" functions (and other similar parsing functions) as well as any string
whose value has been set with the ASN1_STRING_set() function will additionally
NUL terminate the byte array in the ASN1_STRING structure.

However, it is possible for applications to directly construct valid ASN1_STRING
structures which do not NUL terminate the byte array by directly setting the
"data" and "length" fields in the ASN1_STRING array. This can also happen by

```
using the ASN1_STRING_set0() function.

Numerous OpenSSL functions that print ASN.1 data have been found to assume that
the ASN1_STRING byte array will be NUL terminated, even though this is not
guaranteed for strings that have been directly constructed. Where an application
requests an ASN.1 structure to be printed, and where that ASN.1 structure
contains ASN1_STRINGs that have been directly constructed by the application
without NUL terminating the "data" field, then a read buffer overrun can occur.

The same thing can also occur during name constraints processing of certificates
(for example if a certificate has been directly constructed by the application
instead of loading it via the OpenSSL parsing functions, and the certificate
contains non NUL terminated ASN1_STRING structures). It can also occur in the
X509_get1_email(), X509_REQ_get1_email() and X509_get1_ocsp() functions.

If a malicious actor can cause an application to directly construct an
ASN1_STRING and then process it through one of the affected OpenSSL functions
then this issue could be hit. This might result in a crash (causing a Denial of
Service attack). It could also result in the disclosure of private memory
contents (such as private keys, or sensitive plaintext).

OpenSSL versions 1.1.1k and below are affected by this issue. Users of these
versions should upgrade to OpenSSL 1.1.1l.

OpenSSL versions 1.0.2y and below are affected by this issue. However OpenSSL
1.0.2 is out of support and no longer receiving public updates. Premium support
customers of OpenSSL 1.0.2 should upgrade to 1.0.2za. Other users should upgrade
to 1.1.1l.

An initial instance of this issue in the X509_aux_print() function was reported
to OpenSSL on 18th July 2021 by Ingo Schwarze. The bugfix was developed by Ingo
Schwarze and first publicly released in OpenBSD-current on 10th July 2021 and
subsequently in OpenSSL on 20th July 2021 (commit d9d838ddc). Subsequent
analysis by David Benjamin on 17th August 2021 identified more instances of the
same bug. Additional analysis was performed by Matt Caswell. Fixes for the
additional instances of this issue were developed by Matt Caswell.

Note
====

OpenSSL 1.0.2 is out of support and no longer receiving public updates. Extended
support is available for premium support customers:
https://www.openssl.org/support/contracts.html

OpenSSL 1.1.0 is out of support and no longer receiving updates of any kind.
The impact of these issues on OpenSSL 1.1.0 has not been analysed.

Users of these versions should upgrade to OpenSSL 1.1.1.

References
==========

URL for this Security Advisory:
https://www.openssl.org/news/secadv/20210824.txt

Note: the online version of the advisory may be updated with additional details
over time.

For details of OpenSSL severity classifications please see:
https://www.openssl.org/policies/secpolicy.html
```