

MWR InfoSecurity Security  
Advisory

xine-lib Free Uninitialised  
Variable

24<sup>th</sup> August 2010

MWR  INFOSECURITY

**Contents**

**1 Detailed Vulnerability Description..... 5**

    1.1 Introduction ..... 5

    1.2 Technical Background ..... 5

    1.3 Overview of Vulnerability..... 5

    1.4 Dependencies ..... 6

**2 Interim Workaround ..... 7**

**3 Recommendations..... 7**

**4 Credits..... 7**

**5 References ..... 7**

## xine-lib Free Uninitialised Variable

|                           |                                 |
|---------------------------|---------------------------------|
| <b>Package Name:</b>      | xine-lib                        |
| <b>Date Discovered:</b>   | May 2010                        |
| <b>Affected Versions:</b> | Confirmed in Version "1.1.18.1" |

|                     |  |
|---------------------|--|
| CVE Reference       | Not Yet Assigned   |
| Author              | Rafael Dominguez Vega  |
| Severity            | High Risk  |
| Local/Remote        | Remote   |
| Vulnerability Class | Uninitialised Variable / Remote Code Execution   |
| Vendor              | The xine project - <a href="http://www.xine-project.org/">http://www.xine-project.org/</a> |
| Vendor Response     | The vendor has implemented a fix in xine-lib 1.1.19  |

### Overview:

xine is an open source multimedia playback engine for Unix based operating systems. It is developed in the C programming language and released under the GNU General Public License.

The core engine of xine is xine-lib, which currently supports a number of different frontend player applications: -

|                    |                       |
|--------------------|-----------------------|
| xine-ui            | aaxine                |
| amarok             | xine-vcdx             |
| gxine              | Oxine                 |
| kaffeine           | sinek                 |
| GNOME media player | kxine                 |
| xine-plugin        | enix                  |
| sonic-rainbow      | opie                  |
| toxine             | XinePlayer (Mac OS X) |

xine-lib is affected by a memory corruption vulnerability because it uses a variable without initialising it first. The vulnerability can be exploited by opening a malicious media file in one of the affected players.

### Impact:

This could be exploited by an attacker in order to execute arbitrary code on the target system with the privileges of the user running the application. For this vulnerability to be successfully exploited the attacker will be required to make the user to open a specially crafted multimedia file.

### Cause:

Exploitation of this vulnerability is possible due to the freeing of an uninitialised pointer. An attacker could use this vulnerability to gain control over the pointer that was being freed by using a specially crafted ASF file.

**Solution:**

The vendor has implemented a fix for this vulnerability in xine-lib 1.1.19. Users should upgrade to the latest version of xine-lib. This can be found here [\[3\]](#)

## 1 Detailed Vulnerability Description

### 1.1 Introduction

“xine is a free (gpl-licensed) high-performance, portable and reusable multimedia playback engine. xine itself is a shared library with an easy to use, yet powerful API which is used by many applications for smooth video playback and video processing purposes.”

“xine-lib is the xine core engine. It is needed for all front ends and applications which use xine.” [\[1\]](#) The latest xine-lib release at the time of this vulnerability can be found here [\[2\]](#)

### 1.2 Technical Background

xine-lib is affected by a memory corruption vulnerability due to an uninitialised pointer variable which is later freed.

The value of uninitialised variables is undefined and would normally be “junk” data left in memory. The use of variables that have not been initialised could lead to unpredictable or unintended results that could be used by an attacker to control the program flow and execute arbitrary code.

### 1.3 Overview of Vulnerability

The function “`asf_header_parse_stream_properties`” in the `asfheader.c` source code file, is used by xine to obtain stream information from the header of the multimedia file.

A multimedia file handled by this function could meet the following condition when `private_data_length` is invalid and would go to “`exit_error`”: -

```
asf_stream->private_data = asf_reader_get_bytes(&reader, asf_stream-
>private_data_length);
if (!asf_stream->private_data)
    goto exit_error;
```

source code: `asfheader.c`

Once in “`exit_error`”, the pointer “`asf_stream->error_correction_data`” will be freed when not zero leading to unintended behaviour and a segmentation fault occurring in the process.

```
exit_error:
if (asf_stream) {
    if (asf_stream->private_data)
        free(asf_stream->private_data);
    if (asf_stream->error_correction_data)
        free(asf_stream->error_correction_data);
    free(asf_stream);
}
```

source code: `asfheader.c`

The use of the uninitialised variable in the free() function causes a memory corruption and xine-lib to behave unexpectedly, consequently crashing the application.

Under certain circumstances, it would be possible to control the value of the freed pointer by allocating and freeing a buffer of the same data length prior to the invocation of the free function that used the uninitialised variable. During the analysis of this vulnerability, it was possible to use this vulnerability to gain control over the pointer that was being freed by using a specially crafted ASF file. Such a file would force xine-lib to free the first data chunk from the header file (parse from the function "asf\_header\_parse\_header\_extension") with the same data length of the second data chunk which would be freed with the use of the uninitialised variable.

The output included below shows the control obtained over the pointer being freed:-

```
(gdb) bt
#0 * _GI__libc_free (mem=0x41414141) at malloc.c:3687
#1 0x019a2903 in asf_header_parse_stream_properties (header=0x8ea6310,
buffer=0x8ea53c8 "\221\ä·\267\251\317\021\216", <incomplete sequence \346>,
buffer len=54) at asfheader.c:360
#2 0x019a3649 in asf_header_new (buffer=0x8ea5308 "\002", buffer_len=4096) at
asfheader.c:740
#3 0x0199cefe in asf_read_header (this=0x8ea03c8) at demux_asf.c:417
#4 0x0199d955 in demux_asf_send_headers_common (this=0x8ea03c8) at demux_asf.c:608
#5 0x019a0aa1 in demux_asf_send_headers (this gen=0x8ea03c8) at demux_asf.c:1807
#6 0x0011cc4a in open_internal (stream=0x85ec528, mrl=0x815db30 "exploit.asf") at
xine.c:1258
#7 0x0011ce4d in xine_open (stream=0x85ec528, mrl=0x815db30 "exploit.asf") at
xine.c:1299
#8 0x0805307f in gui_xine_open_and_play (mrl=0x815db30 "exploit.asf", sub=0x0,
start pos=0, start time=0, av offset=0, spu offset=0, report error=1)
at actions.c:537
#9 0x0805409d in gui_play (w=0x0, data=0x0) at actions.c:741
#10 0x0805b5c7 in gui_execute_action_id (action=ACTID_PLAY) at event.c:562
#11 0x0805ceeb in on_start (data=0xbffffad9c) at event.c:1751
#12 0x080d0722 in xitk_run (cb=0x805ce90 <on start>, data=0xbffffad9c) at xitk.c:1988
#13 0x0805cc17 in gui_run (session_opts=0x0) at event.c:1903
#14 0x0806ee59 in main (argc=2, argv=0xbffff564) at main.c:2227
```

## 1.4 Dependencies

The exploitation of this vulnerability will be dependent on the target operating system and the heap protection mechanism implemented.

Additionally, the attacker will be required to make the user to open a specially crafted multimedia file.

## 2 Interim Workaround

The interim work around for this issue will require the source code of xine-lib to be modified and recompiled, for the value of the variable "error\_correction\_data" to be initialised.

## 3 Recommendations

The vendor has implemented a fix for this vulnerability in xine-lib 1.1.19. Users should upgrade to the latest version of xine-lib. This can be found at the location that is referenced here [\[3\]](#)

## 4 Credits

MWR InfoSecurity would like to formally acknowledge the support provided by VulnDev Ltd in the discovery of this vulnerability which was identified using their fuzzing and crash analysis toolkit.



## 5 References

[1] The xine project

<http://www.xine-project.org/>

[2] xine-lib 1.1.18.1

<http://prdownloads.sourceforge.net/xine/xine-lib-1.1.18.1.tar.bz2>

[3] xine-lib 1.1.19

<http://prdownloads.sourceforge.net/xine/xine-lib-1.1.19.tar.bz2>

MWR InfoSecurity  
St. Clement House  
1-3 Alencon Link  
Basingstoke, RG21 7SB  
Tel: +44 (0)1256 300920  
Fax: +44 (0)1256 844083  
[mwrinfosecurity.com](http://mwrinfosecurity.com)