

OpenOffice DOC document Heap Overflow

Source: <http://www.derkeiler.com/Mailing-Lists/securityfocus/bugtraq/2005-04/0150.html>

From: lee xiaojun (*airsupply_at_segfault.cn*)

Date: 04/12/05

Date: 12 Apr 2005 00:04:38 -0000

To: bugtraq@securityfocus.com

('binary' encoding is not supported, stored as-is)

OpenOffice DOC document Heap Overflow
[Security Advisory]

Advisory:[AD_LAB-05001] OpenOffice DOC document Heap Overflow

Class: Design Error

DATE:30/3/2005

CVEID:CAN-2005-0941

Vulnerable:

<=OpenOffice OpenOffice 1.1.4

-OpenOffice OpenOffice 2.0dev

Unvulnerable:

Unknow

Vendor:

www.openoffice.org

I.DESCRPTION:

OpenOffice.org is an office productivity suite, including word processing, spreadsheets, presentations, drawings, data charting, formula editing, and file conversion facilities.

The vulnerability is caused due to a error within the .Doc document header processing.This can be exploited to cause a heap-based buffer overflow.

II.DETAILS:

There is a vulnerability in StgCompObjStream::Load() function, When reading DOC document information of format,memory is allocated by DOC provide length. DOC provided a 32 bits integer,and will use the low 16 bits of this number to allocate memory, but when reading doc information,still use the 32 bits number as length,this maybe cause heap overflow, and when free happened ,will cause write pointer,maybe cause arbitrary code excute .

```
BOOL StgCompObjStream::Load()
```

```
{
```

SecurityFocus Bugtraq: OpenOffice DOC document Heap Overflow

```
memset( &aClsId, 0, sizeof( ClsId ) );
nCbFormat = 0;
aUserName.Erase();
if( GetError() != SVSTREAM_OK )
    return FALSE;
Seek( 8L );
INT32 nMarker = 0;
*this >> nMarker;
if( nMarker == -1L )
{
    *this >> aClsId;
    INT32 nLen1 = 0;
    *this >> nLen1; // we can control this 32 bits int
    sal_Char* p = new sal_Char( USHORT ) nLen1 ]; //use low 16 bits value to allocate memory
    if( Read( p, nLen1 ) == (ULONG) nLen1 ) //still use 32 bits int as length,if failed,
        // will goto free step,maybe cause write pointer.
    {
        aUserName = String( p, gsl_getSystemTextEncoding() );
        ....
        nCbFormat = ReadClipboardFormat( *this );
    }
    else
        SetError( SVSTREAM_GENERALERROR );
    delete [] p; //free step,heap overflow cause write pointer.
}
return BOOL( GetError() == SVSTREAM_OK );
}
```

example:

if we provide 0x10000018 to nLen1,will allocate 0x18 length memory,
Read(p, nLen1) still use 0x10000018 as length,then, read will fail,
but readed length is bigger than allocated memory,and overwrite the next chunk.
when goto delete [] p;,write pointer happened. we had triggered this problem successful.
StartOffice maybe affected too. did not test.

III.CREDIT:

AD-LAB discovery this vuln:)
Vulnerability analysis and advisory by A1rsupply.
Special thanks to xalan's discussion.
Thank to Sam,icbm,liangbin and all Venustech AD-Lab guys:P.

V.DISCLAIMS:

The information in this bulletin is provided "AS IS" without warranty of any kind. In no event shall we be liable for any damages whatsoever including direct, indirect, incidental, consequential, loss of business profits or special damages.

Copyright 1996-2005 VENUSTECH. All Rights Reserved. Terms of use.

SecurityFocus Bugtraq: OpenOffice DOC document Heap Overflow

VENUSTECH Security Lab

VENUSTECH INFORMATION TECHNOLOGY CO.,LTD(<http://www.venustech.com.cn>)

Security

Trusted {Solution} Provider

Service