

[UNIX] Multiple Vendor xpdf PDF Viewer Buffer Overflow Vulnerability

Source: <http://www.derkeiler.com/Mailing-Lists/Securiteam/2004-12/0072.html>

From: SecuriTeam (support_at_securiteam.com)

Date: 12/21/04

To: list@securiteam.com

Date: 21 Dec 2004 18:35:37 +0200

The following security advisory is sent to the securiteam mailing list, and can be found at the SecuriTeam web site: <http://www.securiteam.com>

-- promotion

The SecuriTeam alerts list – Free, Accurate, Independent.

Get your security news from a reliable source.

<http://www.securiteam.com/maillinglist.html>

Multiple Vendor xpdf PDF Viewer Buffer Overflow Vulnerability

SUMMARY

Xpdf is "an open-source viewer for Portable Document Format (PDF) files".

Remote exploitation of a buffer overflow vulnerability in the xpdf PDF viewer, as included in multiple Linux distributions, could allow attackers to execute arbitrary code as the user viewing a PDF file.

DETAILS

Vulnerable Systems:

- * xpdf version 3.00

Immune Systems:

- * xpdf version 3.00pl2

The offending code can be found in the Gfx::doImage() function in the source file xpdf/Gfx.cc.

```
void Gfx::doImage(Object *ref, Stream *str, GBool inlineImg) {  
    Dict *dict;  
    int width, height;
```

Securiteam: [UNIX] Multiple Vendor xpdf PDF Viewer Buffer Overflow Vulnerability

```
int bits;
GBool mask;
GBool invert;
GfxColorSpace *colorSpace;
GfxImageColorMap *colorMap;
Object maskObj;
GBool haveMask;
int maskColors[2*gfxColorMaxComps];
Object obj1, obj2;
int i;

...
// get the mask
haveMask = gFalse;
dict->lookup("Mask", &maskObj);
if (maskObj.isArray()) {
    for (i = 0; i < maskObj.arrayGetLength(); ++i) {
        maskObj.arrayGet(i, &obj1);
[!] maskColors[i] = obj1.getInt();
        obj1.free();
    }
    haveMask = gTrue;
}
...
}
```

Due to the fact that the loop boundaries are not less than the storage area, the maskColors array is eventually filled up. After that, local variables and other stack memory is overwritten. This ultimately leads to control of program flow and arbitrary code execution.

Analysis:

The severity of this issue is mitigated by the fact that several of the local overwritten variables in doImage() are referenced prior to EIP being restored; therefore, before the attack gains control of the target process. However, an attacker with knowledge of the remote operating system can construct and validate a malicious payload before attempting exploitation, thus increasing the chances of success. An attacker must convince a target user to open the malicious file to exploit this vulnerability.

Vendor response:

A patch to address this vulnerability is available from:

ftp://ftp.foolabs.com/pub/xpdf/xpdf-3.00p12_patch
ftp://ftp.foolabs.com/pub/xpdf/xpdf-3.00p12_patch

Updated binaries (version 3.00p12) are available from:

<http://www.foolabs.com/xpdf/download.html>
<http://www.foolabs.com/xpdf/download.html>

Securiteam: [UNIX] Multiple Vendor xpdf PDF Viewer Buffer Overflow Vulnerability

CVE Information:

<<http://www.cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2004-1125>>
CAN-2004-1125

Disclosure timeline:

11/23/2004 – Initial vendor notification
11/29/2004 – Initial vendor response
12/21/2004 – Coordinated public disclosure

ADDITIONAL INFORMATION

The information has been provided by
<<mailto:idlabs-advisories@idefense.com>> iDEFENSE.

The original article can be found at:

<<http://www.idefense.com/application/poi/display?id=172&type=vulnerabilities>>
<http://www.idefense.com/application/poi/display?id=172&type=vulnerabilities>

=====

This bulletin is sent to members of the SecuriTeam mailing list.

To unsubscribe from the list, send mail with an empty subject line and body to:

list-unsubscribe@securiteam.com

In order to subscribe to the mailing list, simply forward this email to: list-subscribe@securiteam.com

=====

=====

DISCLAIMER:

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.