

MDKSA-2005:120 – Updated mozilla-firefox packages fix multiple vulnerabilities

Source: <http://www.derkeiler.com/Mailing-Lists/securityfocus/bugtraq/2005-07/0232.html>

From: Mandriva Security Team (security_at_mandriva.com)

Date: 07/14/05

To: bugtraq@securityfocus.com

Date: Wed, 13 Jul 2005 21:50:44 -0600

-----BEGIN PGP SIGNED MESSAGE-----

Hash: SHA1

Mandriva Linux Security Update Advisory

Package name: mozilla-firefox
Advisory ID: MDKSA-2005:120
Date: July 13th, 2005

Affected versions: 10.2

Problem Description:

A number of vulnerabilities were reported and fixed in Firefox 1.0.5 and Mozilla 1.7.9. The following vulnerabilities have been backported and patched for this update:

In several places the browser UI did not correctly distinguish between true user events, such as mouse clicks or keystrokes, and synthetic events generated by web content. The problems ranged from minor annoyances like switching tabs or entering full-screen mode, to a variant on MFSA 2005-34 Synthetic events are now prevented from reaching the browser UI entirely rather than depend on each potentially spoofed function to protect itself from untrusted events (MFSA 2005-45).

Scripts in XBL controls from web content continued to be run even when Javascript was disabled. By itself this causes no harm, but it could be combined with most script-based exploits to attack people running vulnerable versions who thought disabling javascript would protect them. In the Thunderbird and Mozilla Suite mail clients Javascript is

disabled by default for protection against denial-of-service attacks and worms; this vulnerability could be used to bypass that protection (MFSA 2005-46).

If an attacker can convince a victim to use the "Set As Wallpaper" context menu item on a specially crafted image then they can run arbitrary code on the user's computer. The image "source" must be a javascript: url containing an eval() statement and such an image would get the "broken image" icon, but with CSS it could be made transparent and placed on top of a real image. The attacker would have to convince the user to change their desktop background to the exploit image, and to do so by using the Firefox context menu rather than first saving the image locally and using the normal mechanism provided by their operating system. This affects only Firefox 1.0.3 and 1.0.4; earlier versions are unaffected. The implementation of this feature in the Mozilla Suite is also unaffected (MFSA 2005-47).

The InstallTrigger.install() method for launching an install accepts a callback function that will be called with the final success or error status. By forcing a page navigation immediately after calling the install method this callback function can end up running in the context of the new page selected by the attacker. This is true even if the user cancels the unwanted install dialog: cancel is an error status. This callback script can steal data from the new page such as cookies or passwords, or perform actions on the user's behalf such as make a purchase if the user is already logged into the target site. In Firefox the default settings allow only <http://addons.mozilla.org> to bring up this install dialog. This could only be exploited if users have added questionable sites to the install whitelist, and if a malicious site can convince you to install from their site that's a much more powerful attack vector. In the Mozilla Suite the whitelist feature is turned off by default, any site can prompt the user to install software and exploit this vulnerability. The browser has been fixed to clear any pending callback function when switching to a new site (MFSA 2005-48).

Sites can use the _search target to open links in the Firefox sidebar. A missing security check allows the sidebar to inject data: urls containing scripts into any page open in the browser. This could be used to steal cookies, passwords or other sensitive data (MFSA 2005-49).

When InstallVersion.compareTo() is passed an object rather than a string it assumed the object was another InstallVersion without verifying it. When passed a different kind of object the browser would generally crash with an access violation. shutdown has demonstrated that different javascript objects can be passed on some OS versions to get control over the instruction pointer. We assume this could be developed further to run arbitrary machine code if the attacker can get exploit code loaded at a predictable address (MFSA 2005-50).

The original frame-injection spoofing bug was fixed in the Mozilla Suite 1.7 and Firefox 0.9 releases. This protection was accidentally bypassed by one of the fixes in the Firefox 1.0.3 and Mozilla Suite 1.7.7 releases (MFSa 2005-51).

A child frame can call `top.focus()` even if the framing page comes from a different origin and has overridden the `focus()` routine. The call is made in the context of the child frame. The attacker would look for a target site with a framed page that makes this call but doesn't verify that its parent comes from the same site. The attacker could steal cookies and passwords from the framed page, or take actions on behalf of a signed-in user. This attack would work only against sites that use frames in this manner (MFSa 2005-52).

Several media players, for example Flash and QuickTime, support scripted content with the ability to open URLs in the default browser. The default behavior for Firefox was to replace the currently open browser window's content with the externally opened content. If the external URL was a `javascript: url` it would run as if it came from the site that served the previous content, which could be used to steal sensitive information such as login cookies or passwords. If the media player content first caused a privileged `chrome: url` to load then the subsequent `javascript: url` could execute arbitrary code. External `javascript: urls` will now run in a blank context regardless of what content it's replacing, and external apps will no longer be able to load privileged `chrome: urls` in a browser window. The `-chrome` command line option to load chrome applications is still supported (MFSa 2005-53).

Alerts and prompts created by scripts in web pages are presented with the generic title [JavaScript Application] which sometimes makes it difficult to know which site created them. A malicious page could attempt to cause a prompt to appear in front of a trusted site in an attempt to extract information such as passwords from the user. In the fixed version these prompts will contain the hostname from the page which created it (MFSa 2005-54).

Parts of the browser UI relied too much on DOM node names without taking different namespaces into account and verifying that nodes really were of the expected type. An XHTML document could be used to create fake `` elements, for example, with content-defined properties that the browser would access as if they were the trusted built-in properties of the expected HTML elements. The severity of the vulnerability would depend on what the attacker could convince the victim to do, but could result in executing user-supplied script with elevated "chrome" privileges. This could be used to install malicious software on the victim's machine (MFSa 2005-55).

Improper cloning of base objects allowed web content scripts to walk up the prototype chain to get to a privileged object. This could be used to execute code with enhanced privileges (MFSa 2005-56).

The updated packages have been patched to address these issue.

References:

<http://www.mozilla.org/security/announce/mfsa2005-45.html>
<http://www.mozilla.org/security/announce/mfsa2005-46.html>
<http://www.mozilla.org/security/announce/mfsa2005-47.html>
<http://www.mozilla.org/security/announce/mfsa2005-48.html>
<http://www.mozilla.org/security/announce/mfsa2005-49.html>
<http://www.mozilla.org/security/announce/mfsa2005-50.html>
<http://www.mozilla.org/security/announce/mfsa2005-51.html>
<http://www.mozilla.org/security/announce/mfsa2005-52.html>
<http://www.mozilla.org/security/announce/mfsa2005-53.html>
<http://www.mozilla.org/security/announce/mfsa2005-54.html>
<http://www.mozilla.org/security/announce/mfsa2005-55.html>
<http://www.mozilla.org/security/announce/mfsa2005-56.html>
<http://secunia.com/advisories/15489/>
<http://secunia.com/advisories/15549/>
<http://secunia.com/advisories/15601/>

Updated Packages:

Mandrakelinux 10.2:

e1b405c9ba89903ac57fa8ef1849f9e0 10.2/RPMS/libnss3-1.0.2-7.1.102mdk.i586.rpm
5d06976462d9f0cf9cdc42b7f3449b13 10.2/RPMS/libnss3-devel-1.0.2-7.1.102mdk.i586.rpm
881b159dc065c1822f4084a0022c4654 10.2/RPMS/libnspr4-1.0.2-7.1.102mdk.i586.rpm
0f8273f507c95688351402f120517f52 10.2/RPMS/libnspr4-devel-1.0.2-7.1.102mdk.i586.rpm
4be2d65eaf5baf43eb52bdec806040bb 10.2/RPMS/mozilla-firefox-1.0.2-7.1.102mdk.i586.rpm
a134e6e29f9b0aca55fcd0d8708e9630 10.2/RPMS/mozilla-firefox-devel-1.0.2-7.1.102mdk.i586.rpm
4d1968b656af129405977a9aff3be145 10.2/SRPMS/mozilla-firefox-1.0.2-7.1.102mdk.src.rpm

Mandrakelinux 10.2/X86_64:

27214cb9ac9d2ddbcd40f2ee3934c1b8 x86_64/10.2/RPMS/lib64nss3-1.0.2-7.1.102mdk.x86_64.rpm
2104fd1c3dc3a0fc95c1f69cd2b3bcdd x86_64/10.2/RPMS/lib64nss3-devel-1.0.2-7.1.102mdk.x86_64.rpm
e1b405c9ba89903ac57fa8ef1849f9e0 x86_64/10.2/RPMS/libnss3-1.0.2-7.1.102mdk.i586.rpm
5d06976462d9f0cf9cdc42b7f3449b13 x86_64/10.2/RPMS/libnss3-devel-1.0.2-7.1.102mdk.i586.rpm
47ec9f1c56391a073847e6b5ef8be0b7 x86_64/10.2/RPMS/lib64nspr4-1.0.2-7.1.102mdk.x86_64.rpm
05530693d7b048d721ac16caea859c07
x86_64/10.2/RPMS/lib64nspr4-devel-1.0.2-7.1.102mdk.x86_64.rpm
881b159dc065c1822f4084a0022c4654 x86_64/10.2/RPMS/libnspr4-1.0.2-7.1.102mdk.i586.rpm
0f8273f507c95688351402f120517f52 x86_64/10.2/RPMS/libnspr4-devel-1.0.2-7.1.102mdk.i586.rpm
e271265e3395b746ad812c93896346b9 x86_64/10.2/RPMS/mozilla-firefox-1.0.2-7.1.102mdk.x86_64.rpm
e253b6883f45647ea3c8e546bf8000d9
x86_64/10.2/RPMS/mozilla-firefox-devel-1.0.2-7.1.102mdk.x86_64.rpm
4d1968b656af129405977a9aff3be145 x86_64/10.2/SRPMS/mozilla-firefox-1.0.2-7.1.102mdk.src.rpm

SecurityFocus Bugtraq: MDKSA-2005:120 – Updated mozilla-firefox packages fix multiple vulnerabilities

To upgrade automatically use MandrakeUpdate or urpmi. The verification of md5 checksums and GPG signatures is performed automatically for you.

All packages are signed by Mandriva for security. You can obtain the GPG public key of the Mandriva Security Team by executing:

```
gpg --recv-keys --keyserver pgp.mit.edu 0x22458A98
```

You can view other update advisories for Mandriva Linux at:

<http://www.mandriva.com/security/advisories>

If you want to report vulnerabilities, please contact

security_(at)_mandriva.com

```
Type Bits/KeyID Date User ID
pub 1024D/22458A98 2000-07-10 Mandriva Security Team
<security*mandriva.com>
```

```
-----BEGIN PGP SIGNATURE-----
```

```
Version: GnuPG v1.2.4 (GNU/Linux)
```

```
iD8DBQFC1eEUmqjQ0CJFipgRAh8CAKCFh+nHmVdmfp7QAQSFUi0WEnXVcgCcCwvF
lBj66NXyt+VZLyBPBQqAK+M=
=ai/5
```

```
-----END PGP SIGNATURE-----
```