

# [NEWS] Mozilla XPCOM Library Race Condition

**Source:** <http://www.derkeiler.com/Mailing-Lists/Securiteam/2005-09/0032.html>

---

**From:** SecuriTeam ([support\\_at\\_securiteam.com](mailto:support_at_securiteam.com))

**Date:** 09/08/05

To: [list@securiteam.com](mailto:list@securiteam.com)

Date: 8 Sep 2005 12:35:49 +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>

-----

Mozilla XPCOM Library Race Condition

---

## SUMMARY

xpcom, or cross platform component object model is a framework for writing cross-platform, modular software. The xpcom library is used in many applications including a majority of the popular browsers such as FireFox, NetScape, Mozilla, Galeon, etc. It seems that there is a race condition of sorts in xpcom that makes it possible for an attacker to crash a victims browser by having them view a malformed HTML document. This issue is not believed to be exploitable by the Mozilla dev team, and will likely be addressed in full at a later date by the development team.

## DETAILS

XPCOM Race Condition:

It is possible for an attacker to create a race condition that will cause an access violation and result in a hard crash of the browser. One way to trigger this issue is by taking a decent sized HTML file and loading a DOM call within some nested divs that will cause part of the page currently being rendered to be deleted. If the page has not loaded by the time the DOM call is made then we can delete objects that have yet to be referenced, which will result in a crash as soon as the browser tries to reference the deleted object.

Securiteam: [NEWS] Mozilla XPCOM Library Race Condition

<<http://www.gulftech.org/wrecko.html>> <http://www.gulftech.org/wrecko.html>

The above link is a simple proof of concept James wrote a few months ago to show the developers how the issue could be used to cause a crash of the affected web browser. Due to time constraints James has not got to look into this issue very in depth, but it may be possible to use the race condition described here in combination with other DOM calls or JavaScript to produce different results than those demonstrated in his proof of concept HTML page.

Solution:

Mozilla have been aware of this issue for some months, and have fixed the issue on trunk, but not on branch. The reason for this as stated by one of the developers is "fixes for this stuff could easily cause regressions". James did test this issue on the latest copy of the Mozilla browser (Deer Park) this morning though, and it seemed to NOT be vulnerable. However, Firefox and the like are still affected.

ADDITIONAL INFORMATION

The information has been provided by <<mailto:security@gulftech.org>> GulfTech Security Research.

The original article can be found at:

<[http://www.gulftech.org/?node=research&article\\_id=00091-07212005](http://www.gulftech.org/?node=research&article_id=00091-07212005)>  
[http://www.gulftech.org/?node=research&article\\_id=00091-07212005](http://www.gulftech.org/?node=research&article_id=00091-07212005)

=====

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](mailto:list-unsubscribe@securiteam.com)

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