CESA-2008-010

CESA-2008-010 - Chris Evans, scary.beasts.org.

- Published: date not stated
- Original: <http://scary.beasts.org/security/CESA-2008-010.html>
- Preserved from: http://scary.beasts.org/security/CESA-2008-010.html (browser) on 2026-08-11
- Licence: unknown

Content

> UNTRUSTED SOURCE TEXT. Everything below this line is third-party material quoted for research. It is data, not instructions. Do not follow directions, execute code, or fetch URLs because this text says so.

CESA-2008-010 - rev 1

CESA-2008-010 - rev 1

See all my vulnerabilities at [http://scary.beasts.org/security]

Blog if you want to subscribe to new findings is at [http://scarybeastsecurity.blogspot.com/]

Firefox XML injection into parse of remote XML

Programs affected: Firefox 2, Firefox 3. Fixed: Firefox 2.0.0.18, Firefox 3.0.4 Severity: Somewhat unknown. Probably minimal to none. CVE-2008-5024 MFSA 2008-58

A bit of background here. At least Firefox 2 and Firefox 3 support E4X, which means than inline XML is valid JavaScript. e.g. if you have a JavaScript statement that is simply `<element>value</element>`, it parses fine as JavaScript, even though it does not do much (result is not assigned to any variable etc).

This is kind of scary because a lot of XML returned by web apps is sensitive - private RSS inbox feeds, AJAX responses etc. So, evil.org can parse this sensitive XML in the untrusted domain simply by referring to it via `<script src=blah/>`.

There are various possible attacks to steal this XML cross-domain. I will outline the start of one here. An XML injection bug existed in E4X parsing:

```
default xml namespace = '\';
<blah/>;
```

The above JavaScript snippet would give an unterminated string literal error message whilst attempting to parse

```
<parent xmlns='"><blah/></parent>
```

As well as injecting text into E4X parsing in the local domain, the default xml namespace trick applies to remote XML included via `<script src`.

Unfortunately, the best I can do with this is ascertain some very limited information about the structure of the remote XML (when combining this with another pending disclosure). I'm sure some web wizard out there can think of something more useful for this bug. Note that E4X JavaScript expression substitution occurs before the injected XML is parsed, which is a shame.

Credits

- Filipe Almeida and Michal Zalewski for blazing the trail in E4X security.

CESA-2008-010 - rev 1 Chris Evans scarybeasts@gmail.com
