Status and Risk Analysis of Firefox Bug #360493

Chapin Information Services, Inc.
www.info-svc.com

March 9, 2007

Summary

This analysis concludes that the February 23 release of Firefox version 2.0.0.2 has resolved approximately 24% of the security risk related to Bug #360493. Timely resolution of the bugs blocking Bug #360493 would satisfy the current security needs of the average Internet user.

Introduction

Chapin Information Services (CIS) has conducted a detailed analysis of the Password Management patch released in version 2.0.0.2 of the Mozilla Firefox web browser. A simple approval or disapproval of this patch has been impossible due to a large number of circumstantial flaws that have not been corrected as of this version.

The following analysis is divided into lists of risk factors that may contribute to the overall success or failure of Password Management. Each risk is assigned a weight from zero to five such that zero means no risk and five means relatively high risk.

A discussion of each risk follows the analysis tables. Where this document mentions risks that are not yet disclosed to the public, only status information will be given.

Timeline

11/12/06	CIS Reports Bug #360493, Firefox Password Management Flaws
11/12/06	Bob Clary Changes Status to New on Bug #360493
11/21/06	Daniel Veditz Removes Security Lock on Bug #360493
12/01/06	Biju Reports Bug #362576, Autocomplete Attribute Broken
12/19/06	Mozilla Releases Firefox v2.0.0.1
01/29/07	CIS Recommends Disabling Firefox Password Management
02/01/07	CIS Reports Bug #368959, Additional Password Management Flaws
02/23/07	Mozilla Releases Firefox v2.0.0.2
02/24/07	Mauro Bartoccelli Reports Bug #371525, FillPassword Function Broken
03/04/07	Gavin Sharp Changes Status to Resolved on Bug #360493
03/05/07	Secunia Reports Bug #360493 Not Fixed Properly
03/06/07	CIS Reports Bug #372885, Additional Password Management Flaws

Risks Identified in Comments

#	Comment Numbers	Description	Weight	Status
1	#0, #31, #51, #80, #110, #118, #121, #135, #151, #160-179, #208, #216, #224, #228, #233, #267	The originating path of managed forms is ignored.	2	Open
2	#0, #51, #80, #92, #216	The "action" path of managed forms is ignored.	4	Open
3	#0, #110, #155, #307	The “action” attribute of managed forms is ignored when retrieving (filling) passwords.	4	Resolved
4	#0, #149, #326, #329	The "action" attribute of managed forms is ignored when saving passwords.	2	Open
5	#0, #11, #269	There is no programmatic feedback about the destination of unmanaged form submissions.	3	Open
6	#0, #15, #39, #205, #229, #238, #307	Automatic filling of saved credentials without user action.	3	Partial Solutions
7	#0, #70, #272	Mozilla has advertised "active protection from online scams", which was ineffective for this bug.	3	Open
8	#6, #38	“XSS” and scripting vulnerabilities.	0	N/A
9	#7, #114	The (in)visibility of managed forms is ignored.	3	Open
10	#22, #27, #28, #61, #91, #279, #331-346	Dynamic changes to the “action” attribute of managed forms are ignored.	4	Open
11	#49, #88, #358	Unpatched security risks reported to Mozilla are not included in the Firefox release notes.	3	Open
12	#68	Managed forms contain plain text passwords in memory.	2	Open
13	#81	The "method" attribute of managed forms is ignored.	2	Open
14	#253-261, #292-297, #305-309	The new patch does not affect all code paths.	5	Resolved
15	#274, #286-291, #294-296	The new patch uses the wrong "action" attribute variables.	2	Resolved
16	#285, #348, #349	The new "action" attribute check is at the wrong line in the patch.	1	Open
Subtotal			**43**	

Risks Identified in Bug Dependencies

#	Bug Number	Description	Weight	Status
17	362576	The form "autocomplete" attribute is broken.	1	Open
18	368959	Additional password management flaws	3	Open
19	371525	nsPasswordManager::FillPassword() was broken by the new patch.	5	Patch Pending
20	372885	Additional password management flaws	1	Open
Subtotal			**10**	

Other Risks Identified for This Analysis

#21. The Password Management module (nsPasswordManager) contains a large amount of Form Management code. This design issue is assigned a weight of 5. Bug #373154

#22. Additional form management flaws. This is assigned a weight of 1. Bug #373309

Descriptive Statistics of Weighted Risks

Average risk weight: 2.7
Sum of risk weights: 59
Sum of risks resolved: 14
Percent resolved: 24%

Discussion

Mozilla has requested discussion of all remaining matters be separate from the original bug. CIS has opened new bugs in the Bugzilla website for each of the unresolved risks. It is prudent to assume some of these discussions will lead to a “wontfix” status.

Risk #1. The most controversial assumption of the Password Manager design has been the use of domain names without validating the full URL. This was discussed at length in the current bug and should continue in Bug 263387.

The question at the heart of this issue: Is Firefox capable of determining a website requesting credentials is the same website for which credentials were saved?

It is our opinion that the lack of URL path checking and the lack of any option to change this behavior represents one of the greatest remaining risks to users.

Risk #2. The lack of URL path checking affects both the originating page address and the form “action” attribute. Because the latter value controls the destination of password transmission, we have assigned it a greater risk.

Risk #3. Changes made to Firefox v2.0.0.2 focused on validating the form "action" attribute before filling saved credentials into a login form. This change is working, however its effectiveness may hinge on the other risks listed in this document.

Risk #4. The changes made to resolve Risk #3 did not include validating the form "action" attribute when it is saved to disk. This means the Password Manager still does not aid the user in making decisions about the safety of typing passwords on websites. This also negates measures to mitigate Risk #3 when the user is unable to correctly evaluate new requests for credentials. Discussion of this risk should continue in Bug #373144.

Risk #5. Firefox does not provide any feedback about where the contents of a form will be transmitted. This is primarily the fault of relying on protocols that are common and standard, but are also flawed and known to be insecure. The Forms authentication standard does not require Firefox to use the website requesting a password as the transmission end point, yet it also does not prevent Firefox from informing the user.

Developers of major web browsers such as Internet Explorer and Firefox have a responsibility to also participate in the development of new security protocols. The existing protocols, which have been in use since the birth of the World Wide Web, have been considered "unacceptable" since 1999. The background, problems, and recommendations about these protocols are discussed at length in the CIS whitepaper, The Internet's Best Kept Secret: No Password Is Safe.

Risk #6. Users now have two options to disable automatic password filling without disabling the Password Manager. The first option is called "Use a master password", which can be found in the Options dialog box and requires the user to enter a password. The second option can be found by opening a new tab in the browser, typing "about:config" in the address bar, scrolling down to "signon.prefillForms", and then double clicking. Neither of the two options are enabled by default, so we do not consider them to be a complete solution. Several comments in this bug suggested adding a toolbar button that must be clicked before any passwords will be filled.

Risk #7. The "active protection from online scams" advertisement may create a false sense of security with regard to the Firefox Password Manager. We caution Mozilla against making unqualified claims to promote product security. We also caution users against accepting any product as being absolutely secure. As this risk does not represent a software bug, it will not be added to the Bugzilla website for discussion.

Risk #8. Vulnerabilities related to client-side scripting (XSS) injections are excluded from the scope of this bug and this analysis. More information can be found in the "Depends on" section of Bug #301375.

Risk #9. Firefox does not distinguish between forms that are visible and forms that are not visible to the user. The potential interaction of Risks 6 and 9 have been

demonstrated, and this is an excellent example of how circumstantial flaws can multiply the risk to security. Discussion of this risk should continue in Bug #373153.

Risk #10. This is a highly technical problem that should have been discussed in detail before the release of version 2.0.0.2. To put it simply, Firefox does not consistently decide at what time it should evaluate form data. Participants in the bug discussion cited legitimate uses for client-side scripts that would modify the form data, and specifically the "action" attribute. Therefore, this risk cannot be dismissed as an XSS problem. Discussion of this risk should continue in Bug #373151.

When a page first loads in Firefox, a function named nsPaswordManager:: FillDocument() is called upon to evaluate all of the forms on that page. This function is now designed to check the "action" attribute of each form and compare it to saved credentials. This immediately causes a problem. The "action" attribute is never saved to disk within this function, and this function has no role in the transmission of form data. Consequently, there are three separate events occurring at three different times.

Logically, for a client script to affect the "action" attribute, it must make changes to the form at some time between the page loading and the form data transmission. We will use the onSubmit event as an example. According to Comment #341, the onSubmit event would happen after the credentials are saved to disk. In this scenario, the client script would change the "action" attribute after the value was saved to disk, but before the form is transmitted. The "action" attribute being saved to disk does not correspond to the transmission end point, and the Password Manager will be unaware of any changes that occur during the onSubmit event.

Worse yet, disabling JavaScript in this scenario would cause the "action" attribute to remain static, potentially causing credentials to transmit over an insecure connection or to arrive at the wrong server.

Risk #11. There were multiple complaints about the difficulties in learning about this type of bug. It was pointed out that both the Mozilla website and the Firefox release notes had no documentation of this bug at any time before February 23, 2007, more than three months after it was made public in the Bugzilla website. This lack of documentation and communication creates an opportunity for Mozilla to mitigate this type of security risk in the future. As this risk does not represent a software bug, it will not be added to the Bugzilla website for discussion.

Risk #12. A theory mentioned in Comment #68 hinted at never filling real passwords into password fields. This would have obvious advantages in that it would significantly reduce the attack surface presented by Password Management. In fact, the real password is not needed by any part of the program until transmission time. Discussion of this risk should continue in Bug #373149.

Risk #13. See Risk #5. Again, the Forms authentication standard does not prevent Firefox from warning or stopping the user when the "get" method is used. Discussion of this risk should continue in Bug #371515.

Risk #14. In some drafts of the patch, the "action" attribute was not validated before all instances of passField->SetValue(). This was corrected before the release of version 2.0.0.2.

Risk #15. In some drafts of the patch, the "action" attribute was not correctly referenced. This was corrected before the release of version 2.0.0.2.

Risk #16. For both performance and security reasons, the "action" attribute should be validated at the top of the credentials DataEntry loop. In version 2.0.0.2, this validation occurs in the middle of the DataEntry loop after retrieving the form signature and validating the username and password fields. Discussion of this risk should continue in Bug #373145.

Risk #17: The "autocomplete" attribute, which is supposed to allow webmasters to disable the Password Manager, does not work in all cases. As a result, the Password Manager fills password fields when it should not. This is Bug #362576.

Risk #18: CIS reported several problems related to the Password Manager in Bug #368959, and included a request that they be fixed before Bug #360493. The status of this bug is currently, "new".

Risk #19: Mozilla and CIS have confirmed that the nsPasswordManager::FillPassword() function does not work correctly in version 2.0.0.2. As a result, this version of the Password Manager is partially crippled and may be difficult to use in many instances. This is Bug #371525.

Risk #20: CIS reported a problem related to the Password Manager in Bug #372885. A test case has been included. The status of this bug is currently, "unconfirmed".

Risk #21: One of the most substantial risks remaining with this bug is the overall design of the Password Manager module, nsPasswordManager. Its code includes a large amount of form management routines that are not essential to the logical function of this module, and cause or exacerbate all of the above risks.

To eliminate this risk would mean:

1. Creating two, logically separate modules for Form Management and Password Management.
2. Allowing the Form Management module to unlock passwords only by calling on a Password Management function.
3. Requiring the Form Management module to include in its function call all of the necessary data needed to validate an authentication request.

4. Requiring the Password Management module to hand-off validated credential requests directly to the transmission function, so that the Form Management module is never in possession of saved credentials. (Note: This is an idealistic feature that would exclude scripting of saved credentials.)
5. Eliminating all Form-related routines from the Password Management module.

Discussion of this risk should continue in Bug #373154.

Risk #22. CIS reported a problem related to the Form Manager in Bug #373309. A test case has been included. The status of this bug is currently, "unconfirmed".

Conclusion

Given the 22 risks identified as relating to this bug, and the actual amount of corrective action taken so far, it is impossible to change our recommendation that the Firefox Password Manager be disabled by users. By our estimation, less than 25% of the risk represented by these problems has been resolved in the release of version 2.0.0.2.

We suggest that Mozilla correct 40% of these weighted risks before its Password Manger is deemed safe for common use. This means, at minimum, resolving the risks identified in the four bug dependencies, or providing the level of security needed to reasonably protect the average Internet user. We also suggest considering goals greater than 40%, which could position the Firefox Password Manager ahead of competitive products.