#	Issues	Rating	Vulnerability Finding	Recommendation
1	SSL is Required (Enable and Enforce) - External::SSL is Required (Enable and Enforce) - External	Critical	SSL is not available or is not enforced on the web server, therefore, the entire session will be unencrypted. All traffic will be sent in plaintext, potentially exposing sensitive data on the network.	Enable and Enforce SSL on the web server. Enforce the use of SSL Either disable nonSSL traffic entirely or configure the web server to redirect all nonSSL traffic to SSL
2	Unauthenticated Reflected XSS Exists - External::Unauthenticated Reflected XSS Exists	Critical	The data that is displayed to the screen is directly pulled from the URL and/or POST data. Without proper input validation, cookie stealing, or other malicious attacks may occur at the application level. Since this exists on unauthenticated pages, this is a prime target for a phishing attack.	Apply input filtering and output encoding to user supplied data before returning it to the user's browser. Apply business logic filters to user input (including cookies, POSTs, GETs, and headers), allowing only characters necessary for the requested input. Apply HTML or URL encoding to any returned values to neutralize any remaining and potentially dangerous characters that would otherwise be interpreted as HTML or script by the browser (<>""'()=[];?*$%^&|/\{}) .
3	Insecure Configuration Management (generic)::Flash parameter AllowScriptAccess was set to always	Medium	It is possible to steal or manipulate customer session and cookies, which might be used to impersonate a legitimate user, allowing the hacker to view or alter user records, and to perform transactions as that user	Set the AllowScriptAccess parameter to 'sameDomain' which tells the Flash Player that only SWF files loaded from the same domain as the parent SWF will have script access to the hosting web page.
4	HTML has Hidden Comments::HTML has Hidden Comments	Medium	HTML has hidden contents/comments.	In general, there should be no hidden comments within HTML code because the contents can leak private information about the application (i.e. developer's names, email addresses,

This is a critical vulnerability. It should be remediated by turning on SSL during login.

Vulnerability 1: SSL is Required (Enable and Enforce) - External

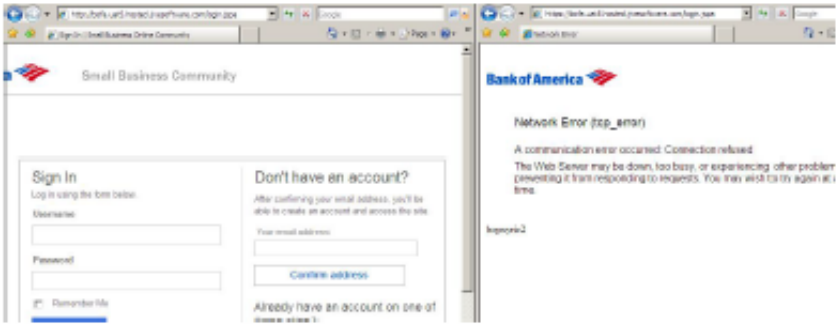
SSL is not available or is not enforced on the web server, therefore, the entire session will be unencrypted. All traffic will be sent in plaintext, potentially exposing sensitive data on the network.

Example 1:

The application does not have SSL enabled or enforced. This means that all HTTP requests/replies to/from the web server will be unencrypted. Because this is an external facing application, SSL must be enabled and enforced on every page of the application (especially any login pages that include usernames and passwords).

Steps to reproduce:

1. Browse to the application. http://bofa.uat5.hosted.jivesoftware.com
2. Note that the application responds with a HTTP connection rather than using a HTTPS (SSL) connection.
3. Replace http:// with https:// and notice the server accepts the request, hence SSL is enabled but not enforced.



This is a critical vulnerability. We need to remediate within 30 days (as of 8/27/12)

Vulnerability 2: Unauthenticated Reflected XSS Exists - External

The data that is displayed to the screen is directly pulled from the URL and/or POST data. Without proper input validation, cookie stealing, or other malicious attacks may occur at the application level. Since this exists on unauthenticated pages, this is a prime target for a phishing attack.

Example 1:

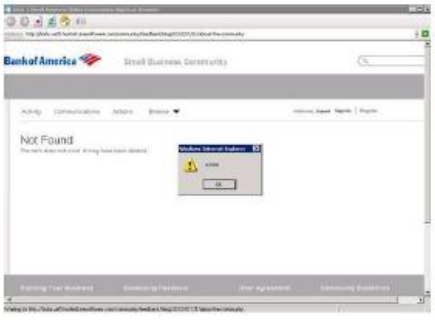
The application does not properly validate various pages and parameters throughout the application. This allows an attacker to perform a cross-site scripting attack against the application. An attacker will typically try to steal a user's session token (allowing the attacker to impersonate the user) or steal a user's credentials by displaying a fake login page.

Affected URLs:

This vulnerability is systemic throughout the application, but a sample list of URLs follow:

- http://bofa.uat5.hosted.jivesoftware.com/community/feedback/blog/2012/07/31/about-the-community
- http://bofa.uat5.hosted.jivesoftware.com/community/feedback/blog/2012/07/31/community-guidelines
- http://bofa.uat5.hosted.jivesoftware.com/community/feedback/blog/2012/07/31/getting-support-in-thecommunity
- http://bofa.uat5.hosted.jivesoftware.com/community/feedback/blog/2012/07/31/login.jspa
- http://bofa.uat5.hosted.jivesoftware.com/community/feedback/blog/2012/07/31/search.jspa

Parameters Affected : _patternParameter_Date__0
https://familycare.sapnagroup.net/baml/nanny-booking.html
Steps to reproduce:



1. Browse to http://bofa.uat5.hosted.jivesoftware.com/community/feedback/blog/2012/07/31%22/%3E%3CSTYLE%3E@import%22javascript:alert(63765)%22;%3C/STYLE%3E/about-the-community
2. Note that the Javascript gets executed at the end user's browser.

This is a “medium” issue. The “critical” vulnerabilities take precedence. However, we can remediate this if it is an easy fix.

Vulnerability 3: Insecure Configuration Management (generic)

It is possible to steal or manipulate customer session and cookies, which might be used to impersonate a legitimate user, allowing the hacker to view or alter user records, and to perform transactions as that user

Example 1:

The flash player accepts object parameters such as AllowScriptAccess. The AllowScriptAccess parameter determines whether the loaded SWF (or any SWF it subsequently loads) will be permitted to access the web page in which the SWF is embedded. If the parameter is set to 'always' then the SWF loaded from any domain could inject a script into the hosting web page.

Affected URL:

http://bofa.uat5.hosted.jivesoftware.com/gadgets/js/core:rpc.js

Steps to reproduce:

1. Navigate to the URL http://bofa.uat5.hosted.jivesoftware.com/gadgets/js/core:rpc.js
2. Save the file and view the source code of the file.
3. Observe the page source has <param name="allowScriptAccess" value="always">.

This is a “medium” issue. The “critical” vulnerabilities take precedence. However, we can remediate this if it is an easy fix.

Vulnerability 4: HTML has Hidden Comments

HTML has hidden contents/comments.

Example 1:

Many web application programmers use HTML comments to help debug the application when needed. While adding general comments is very useful, some programmers tend to leave important data, such as: filenames related to the web application, old links or links which were not meant to be browsed by users, old code fragments, etc. An attacker who finds these comments can map the application's structure and files, expose hidden parts of the site, and study the fragments of code to reverse engineer the application, which may help develop further attacks against the site.

Affected URLs:

This vulnerability is systemic throughout the application, but a sample list of URLs follow:

http://bofa.uat5.hosted.jivesoftware.com/forgot-username!input.jspa
http://bofa.uat5.hosted.jivesoftware.com/message/106999
http://bofa.uat5.hosted.jivesoftware.com/message/106999
http://bofa.uat5.hosted.jivesoftware.com/message/114119
http://bofa.uat5.hosted.jivesoftware.com/polls/1041

Steps to reproduce:

1. Log into the application.
3. Navigate to the URL http://bofa.uat5.hosted.jivesoftware.com/forgot-username!input.jspa
4. View the source code of the file.
5. Notice the page contains HTML comments.