

Oracle® Adaptive Access Manager
API Integration Guide
10g (10.1.4.3.0)

December 2007

ORACLE®

Copyright © 2007, Oracle. All rights reserved.

The Programs (which include both the software and documentation) contain proprietary information; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or decompilation of the Programs, except to the extent required to obtain interoperability with other independently created software or as specified by law, is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. This document is not warranted to be error-free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose.

If the Programs are delivered to the United States Government or anyone licensing or using the Programs on behalf of the United States Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the Programs, including documentation and technical data, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement, and, to the extent applicable, the additional rights set forth in FAR 52.227-19, Commercial Computer Software--Restricted Rights (June 1987). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and we disclaim liability for any damages caused by such use of the Programs.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

The Programs may provide links to Web sites and access to content, products, and services from third parties. Oracle is not responsible for the availability of, or any content provided on, third-party Web sites. You bear all risks associated with the use of such content. If you choose to purchase any products or services from a third party, the relationship is directly between you and the third party. Oracle is not responsible for: (a) the quality of third-party products or services; or (b) fulfilling any of the terms of the agreement with the third party, including delivery of products or services and warranty obligations related to purchased products or services. Oracle is not responsible for any loss or damage of any sort that you may incur from dealing with any third party.

Contents

Preface.....	4
Documentation	4
Overview	6
Application (API) Integration	7
SOAP Services	7
Native API.....	7
Adaptive Risk Manager Online Native Client API – Web Services/ SOAP.....	8
Adaptive Risk Manager Online Native Client API – Static Linking.....	9
Integration Options.....	10
Adaptive Risk Manager Only Scenario	10
Adaptive Risk Manager, Adaptive Strong Authenticator and KBA Scenario	14
Adaptive Risk Manager and KBA Scenario	24
Troubleshooting	25

Preface

The Oracle® Adaptive Access Manager API Integration Guide provides information on natively integrating the client portion of the Adaptive Risk Manager Online solutions. In an API integration, the client application invokes the Oracle Adaptive Risk Manager Online APIs directly and manages the authentication and challenge flows.

Documentation

The Oracle Adaptive Access Manager 10g documentation includes the following:

- The Oracle® Adaptive Access Manager API Integration Guide, which provides information on natively integrating the client portion of the Adaptive Risk Manager Online solutions. In an API integration, the client application invokes the Adaptive Risk Manager Online APIs directly and manages the authentication and challenge flows.
- The Oracle® Adaptive Access Manager Database Installation Guide (Oracle), which provides information about installing the Adaptive Access Manager schema into an Oracle database. Access to the Adaptive Access Manager schema is a requirement of the Adaptive Access Manager Application Server, which hosts the Adaptive Strong Authenticator and the Adaptive Risk Manager. Note that the Adaptive Manager Access Manager schema needs to be installed into the Oracle database before proceeding to the installation of the proxy.
- The Oracle® Adaptive Access Manager Database Installation Guide for SQL Server 2005, which provides information about installing the Adaptive Access Manager schema into SQL Server 2005. Access to the Adaptive Access Manager schema is a requirement of the Adaptive Access Manager Application Server, which hosts the Adaptive Strong Authenticator and the Adaptive Risk Manager. Note that the Adaptive Manager Access Manager schema needs to be installed into SQL Server 2005 before proceeding to the installation of the proxy.
- The Oracle® Adaptive Access Manager Proxy Integration Guide, which provides programming information and instructions on the installation of the Adaptive Access Manager proxy, one of the components in the Adaptive Access Manager UIO deployment. The Oracle Adaptive Access Manager's Universal Installation Option (UIO) offers multi-factor authentication to Web applications without requiring any change to the application code. The Oracle® Adaptive Access Manager Proxy and The Oracle® Adaptive Access Manager Proxy Web Publishing Configuration are guides specific to the UIO deployment.
- The Oracle® Adaptive Access Manager Proxy Web Publishing Configuration, which provides information on creating web publishing rules and listeners so that Web applications and the WebUIO can be accessible from the Internet. The Oracle Adaptive Access Manager's Universal Installation Option (UIO) offers multi-factor authentication to Web applications without requiring any change to the application code. The Oracle® Adaptive Access Manager Proxy and The Oracle® Adaptive Access Manager Proxy Web Publishing Configuration are guides specific to the UIO deployment.

- The Oracle® Adaptive Risk Manager Online Installation Guide, which provides information on the installation of the administration user interface of Oracle Adaptive Access Manager. Adaptive Risk Manager Online is the administration user interface of Oracle Adaptive Access Manager, a set of web-based administration tools that provides sophisticated fraud monitoring, analysis, and tracking by user location, device, time of day, type of transaction, as well as a host of other factors, and evaluates these factors against a set of customizable rules.
- The Oracle® Adaptive Access Manager LDAP Configuration Guide, which provides information on how to configure the Oracle Adaptive Access Manager Application Server to allow a user to be authenticated via a user identifier and password. The intended audience of this manual are users of WebLogic and Tomcat who want to use LDAP to set up users instead of the functionality in WebLogic and Tomcat.
- The Oracle® Adaptive Access Manager Import/Export Manual, which provides information importing groups, rule templates, and models from the Adaptive Access Manager schema.
- The Oracle® Adaptive Risk Manager Online Customer Care API Guide, which provides information about the Adaptive Risk Manager Online Customer Care API and provides the XML definition for each of the APIs.
- The Oracle® Adaptive Access Manager Database Tables Archiving and Purging Procedure, which provides information on the purge and archive scripts in the Oracle Adaptive Access Manager Database Tables of Microsoft SQL Server 2005. The procedure to trigger the scripts and information on verification and validation of script results are also provided.
- The Oracle® Adaptive Access Manager SQL Server Maintenance Guide, which provides instructions to set up The Oracle Adaptive Access Manager Maintenance Plan to purge and archive scripts in the Oracle Adaptive Access Manager database tables of Microsoft SQL Server 2005. The manual also discusses in detail how to trigger the scripts and provides information on the verification and validation of script results.
- The Oracle® Adaptive Risk Manager™ Administrator's Guide, which provides step-by-step instructions for creating and managing groups, creating models that contain rules, and customizing and managing rules.
- The Oracle® Adaptive Risk Manager™ Dashboard and Reporting Guide, which provides detailed instructions on how to use the dashboard and reporting functionality within the Oracle® Adaptive Risk Manager Online. The Oracle® Adaptive Risk Manager Online includes a dashboard that provides a high-level overview of users and devices that have generated alerts and the alerts themselves, and it contains a comprehensive collection of reports on users, locations, devices, and security alerts.
- The Oracle® Adaptive Risk Manager™ Customer Care Administration Guide, which provides information on creating new customer cases and administering them.

Overview

This manual contains the guidelines to natively integrate the client portion of the Adaptive Risk Manager Online solutions. In an API integration, the client application invokes the Adaptive Risk Manager Online APIs directly and manages the authentication and challenge flows.

Oracle provides the APIs to fingerprint the devices, collect authentication/transaction logs, run security and business rules, and challenge the user by using Adaptive Risk Manager Online's KBA. Adaptive Risk Manager Online also provides the utility APIs to generate authentication pads like KeyPad, TextPad, and QuestionPad.

API integration of Adaptive Risk Manager Online provides various advantages—some of which are highlighted below:

- Flexibility in managing and controlling the authentication process flow.
- Ability to change the default user registration flow.
- Capability to share session data between existing applications and the Adaptive Risk Manager Online application. For example, the existing login session ID can be passed on to Adaptive Risk Manager Online API calls.

This manual contains the guidelines for integrating:

- Only the Adaptive Risk Manager
- Adaptive Risk Manager and KBA
- Adaptive Risk Manager, KBA and Authentication devices
- AuthentiPad (Keypad, PinPad, and other pads)
- Customer Care API

Application (API) Integration

Adaptive Risk Manager Online's components are software- and hardware-independent when deployed in a stand-alone environment using the published Web services API over SOAP. Support is also available for native (Java/.NET) environments.

Basic familiarity with SOAP, Java, or .NET is the skill set requirement for integration.

API integration is available in two flavors:

1. SOAP Service
2. Native APIs
 - a. SOAP Service wrapper (in Java or .NET)
 - b. Static-linked libraries (in Java)

SOAP Services

Adaptive Risk Manager Online SOAP services consists of five major modules:

- **VCryptCommon** contains the common APIs.
- **VCryptTracker** contains the APIs for fingerprinting and collecting authentication and transaction logs.
- **VCryptAuth** contains the APIs for accessing the Adaptive Strong Authenticator and KBA modules.
- **VCryptRulesEngine** contains the APIs for running the rules.
- **VCryptCC** contains the APIs for invoking customer-care-related requests.

Using direct SOAP services is preferred if the client does not want to include any of the Adaptive Risk Manager Online client jars or DLL within their application. However, to use the Adaptive Strong Authenticator functionality, native Java or .NET you must use the native Java or .NET integration.

For Web Service Definition Language (WSDL) and parameter definitions, refer to your Web Services guide.

Native API

The native API consists of a wrapper over the SOAP API that is published by the Adaptive Risk Manager Online Server and written in the client's native application language. The native APIs construct the SOAP bodies for the Adaptive Risk Manager Online request and also invoke the SOAP requests.

API integration can be done using the SOAP as the underlying mechanism or statically linking the Adaptive Risk Manager Online jars.

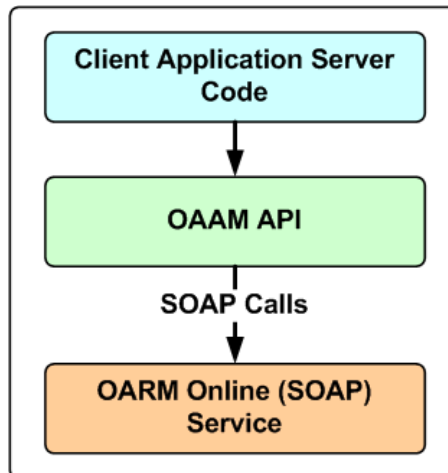
Adaptive Risk Manager Online Native Client API – Web Services/ SOAP

The construction of SOAP bodies and the SOAP calls help in abstracting the SOAP WSDL and other Web Services related issues.

Using native API, which is preferred over making direct SOAP calls, has a lot of advantages. A few advantages are listed below:

- The client library constructs the SOAP body and abstracts the SOAP nuances from the client application developer.
- Changes to any SOAP API signature does not require any code change from the application developer.
- Higher-level utility methods are available to extract parameters required by Adaptive Risk Manager Online directly from the HTTP Request and HTTP Session objects.
- APIs for encoding and decoding of some fingerprint data are available in native integration.

API libraries are available in Java, .NET and C++. In the Web Service configuration, these libraries have utility methods which make direct SOAP calls. The option requires lightweight client libraries (jars or dll) to be added to the client library part. For additional information on API integration deployment and configuration, refer to the installation guide.



Adaptive Risk Manager Online Native Client API – Static Linking

Clients using Java have the option to choose static linking. In static linking, the API doesn't make SOAP calls, instead they statically call the Adaptive Risk Manager Online engine APIs. With the static linking option, the client must include the Adaptive Risk Manager Online server jars and configure appropriate properties.

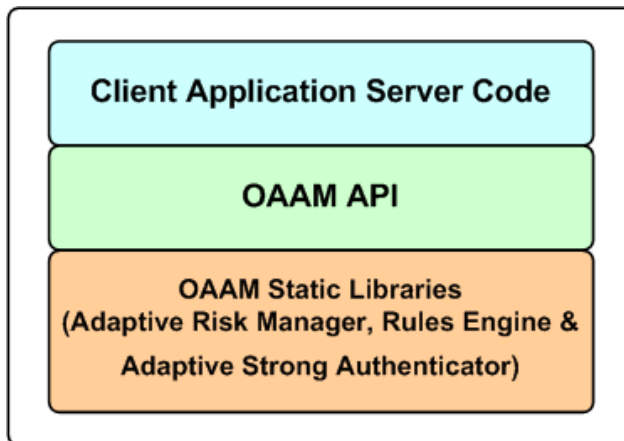
Although this option may provide slightly better performance, it may not be suitable for all clients.

Advantages of static linking are

- No SOAP calls; eliminates creating and tearing down of TCP/IP connections.
- No network latencies.
- Load balancer not required.

Disadvantages of static linking are

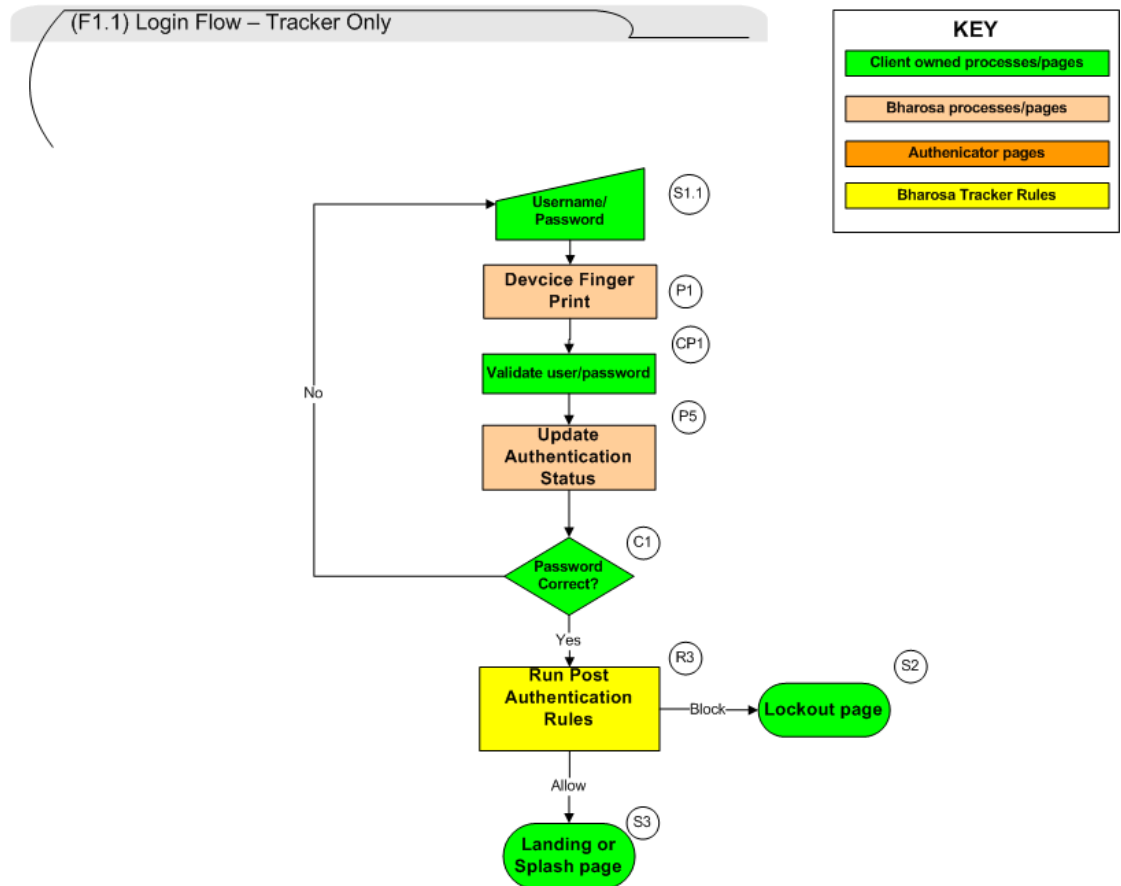
- The client server/application server has to accommodate the extra resource required by the Adaptive Risk Manager Online engine.
- The client server/application server may not be able to load balance the requests.



Integration Options

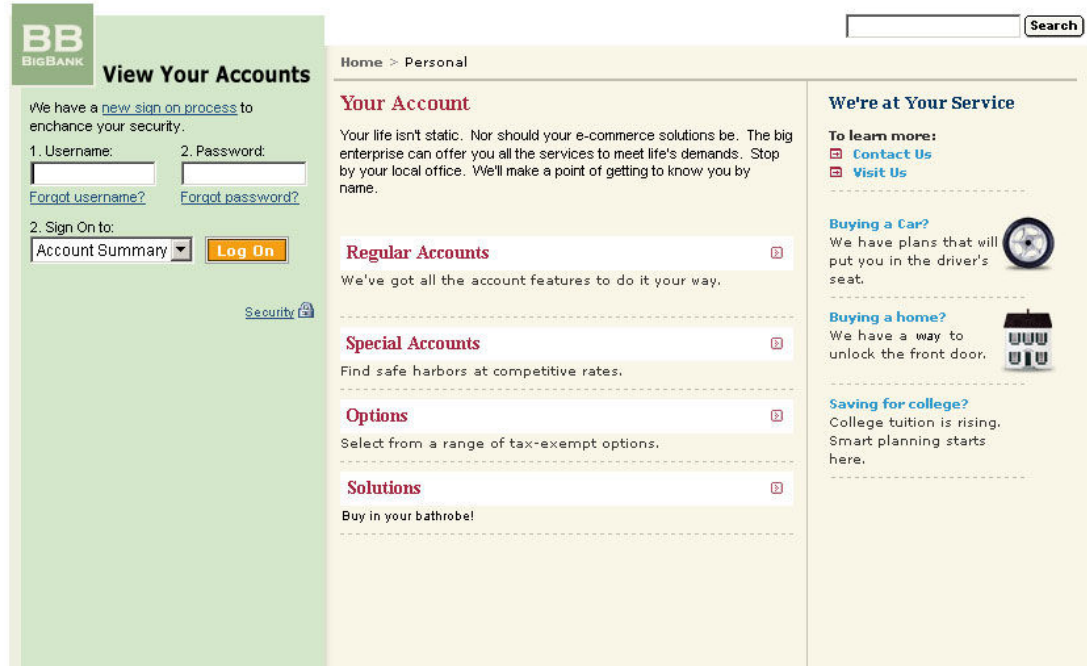
Clients can integrate Adaptive Risk Manager in a relatively short time frame and have their site secure from most fraud attacks. Integrating only the Adaptive Risk Manager doesn't require any change to the user experience.

Adaptive Risk Manager Only Scenario



User/Password Page (S1.1)

The User/Password Page is the existing page currently used by the client. It contains the text box for both the username and password. There are no changes required for this page; however, the post from this page should display a transient (intermediate) refresh page (P1).



Device Fingerprint Flow (F2)

This is the flow for fingerprinting the device. For more details, refer to the “Device Fingerprint Flow (F2)” section.



Reference APIs

Module	APIs	Notes
Server	VCryptTracker::updateLog()	
Sample	handleJump.jsp and handleFlash.jsp	

Validate User/ Passwd (CP1)

This is the client’s existing process. The client invokes the local API to validate the user. The result of the authentication is passed on to the Adaptive Risk Manager Online Server.



Reference APIs

Module	APIs	Notes
Sample	handlePassword.jsp	

Update Authentication Status (P5)

After validating the user password, the status is updated in the Adaptive Risk Manager.



Reference APIs

Module	APIs	Notes
Sample	handlePassword.jsp	
BharosaHelper	BharosaHelper::updateStatus()	

Password Status (C1)

Based on the authentication status, the user is either taken to the retry page or to post authentication rule processing.

Post Authentication Rules (R3)

The post authentication rules are run after the user password is authenticated. The post authentication runtime contains security rules.

The common actions returned are

- **Allow:** Allow the authentication.
- **Block:** Block the user.
- **Challenge:** Challenge is returned if the user has registered questions. The option may not be available for Adaptive Risk Manager Only deployments.



Reference APIs

Module	APIs	Notes
Server	VCryptRulesEngine::processRules()	
Sample	handlePassword.jsp	
BharosaHelper	BharosaHelper::runPostAuthRules()	

Lock Out Page (S2)

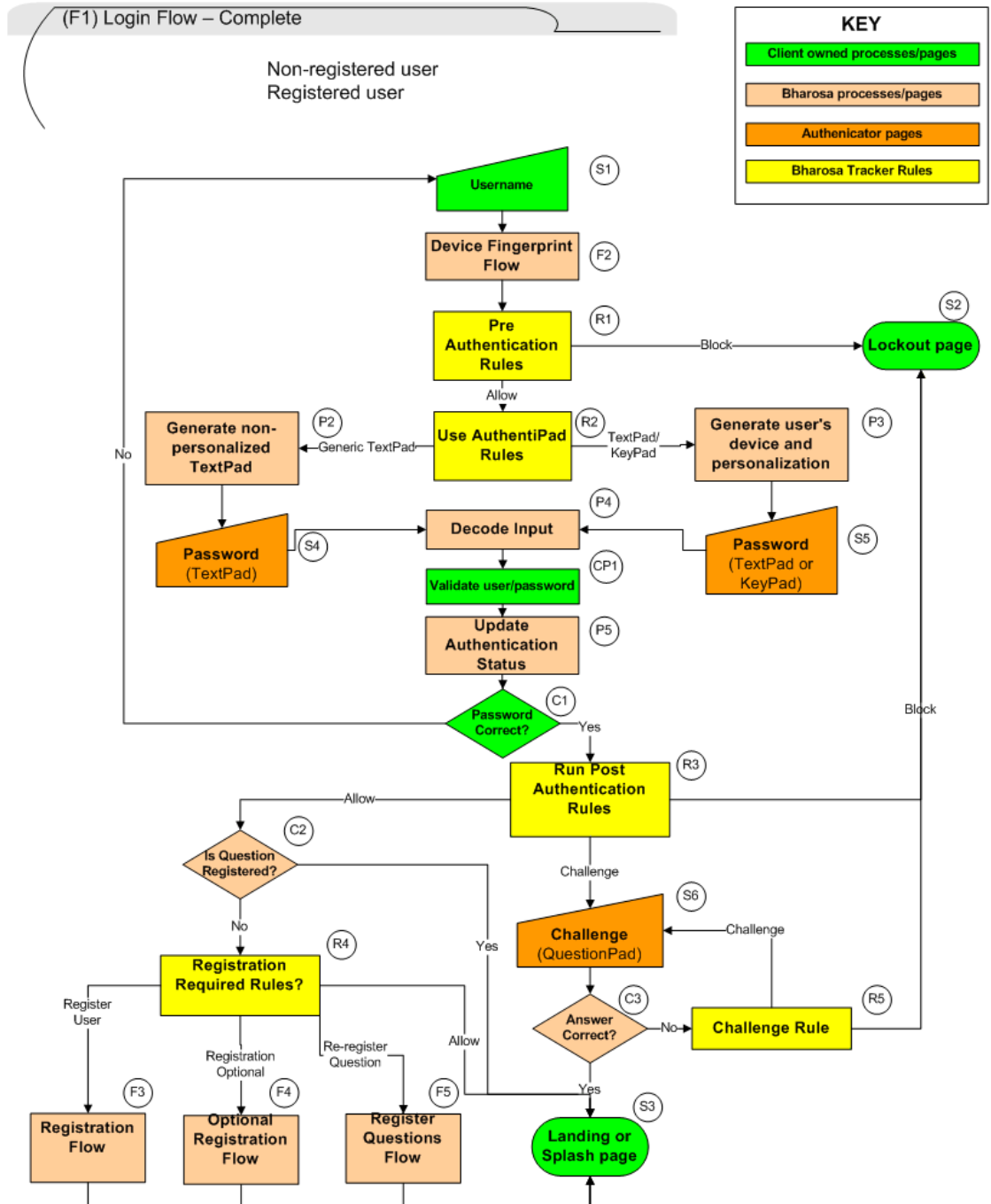
The Lock Out Page is the page that the user is generally redirected to if he is blocked from authentication or if he is carrying out a transaction.

Landing or Splash Page (S3)

The Landing or Splash Page is the page where the user lands after a successful login.

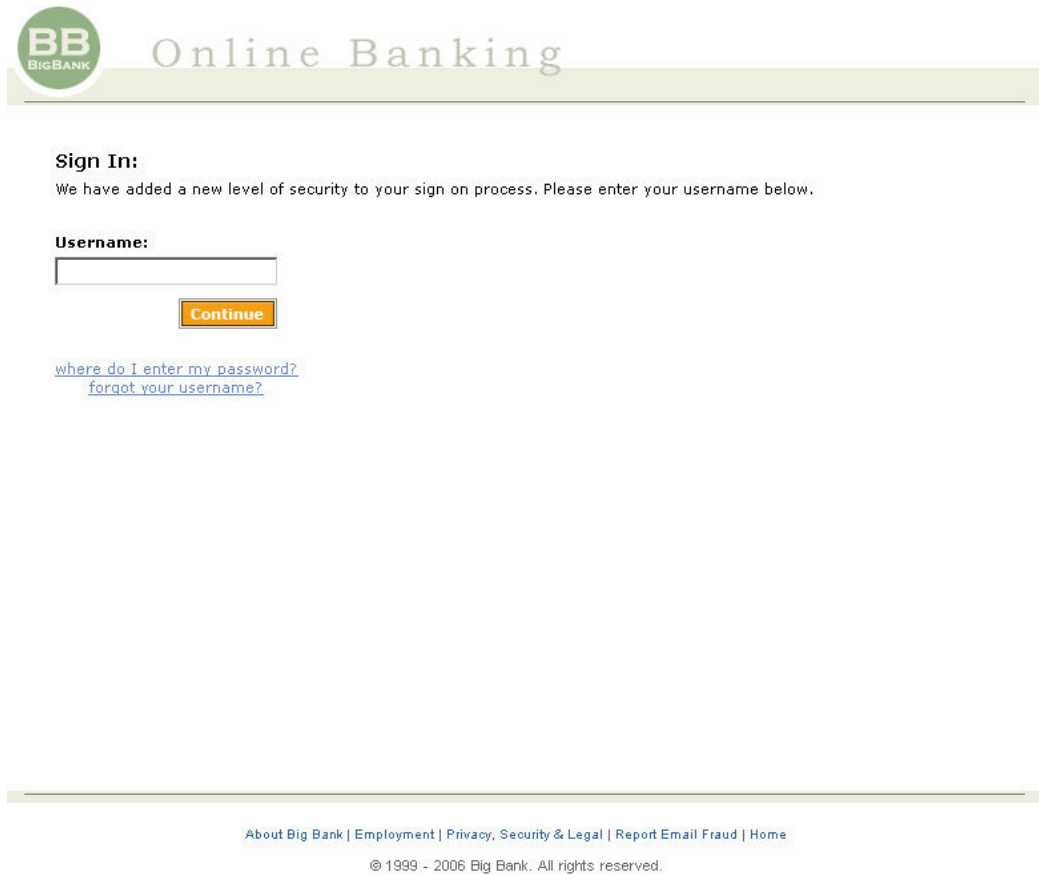
Adaptive Risk Manager, Adaptive Strong Authenticator and KBA Scenario

This flow is a consolidation of the Adaptive Risk Manager, AuthentiPads and KBA solutions. The flows are determined by the rules that are run at different runtimes.



Username Page (S1)

When personalization (image and/or phrase) is used, the login page must be split into two pages. The username (login ID) is accepted from the first page and stored in the HTTP session. The username page is followed by a transient page for capturing the flash and secure cookies and for fingerprinting the device.



Sign In:
We have added a new level of security to your sign on process. Please enter your username below.

Username:

Continue

[where do I enter my password?](#)
[forgot your username?](#)

[About Big Bank](#) | [Employment](#) | [Privacy, Security & Legal](#) | [Report Email Fraud](#) | [Home](#)
© 1999 - 2006 Big Bank. All rights reserved.

Device Fingerprint Flow (F2)

This is the flow for fingerprinting the device. For more details, refer to the “Device Fingerprint Flow (F2)” section.



Reference APIs

Module	APIs	Notes
Server	VCryptTracker::updateLog()	
Sample	handleJump.jsp and handleFlash.jsp	

Pre Authentication Rules (R1)

Pre Authentication rules are run before the user is authenticated or shown his personal device and/or phrase.

The common actions are

- **Allow:** Allow the authentication flow to proceed.
- **Block:** Block the user.



Reference APIs

Module	APIs	Notes
Server	VCryptRulesEngine::processRules()	
Sample	handleJump.jsp	
BharosaHelper	BharosaHelper::runPreAuthRules()	

Use AuthentiPad Rules (R2)

This runtime runs the rules for determining which AuthentiPad is used. If the user has not registered, the rule returns the Generic TextPad. If the user is registered with Adaptive Risk Manager Online, either the personalized TextPad or KeyPad action will be returned.

The common actions are

- **Generic TextPad:** Use default generic TextPad.
- **TextPad:** Use personalized TextPad with phrase.
- **KeyPad:** Use personalized KeyPad with phrase.



Reference APIs

Module	APIs	Notes
Server	VCryptRulesEngine::processRules()	
Sample	password.jsp	
BharosaHelper	BharosaHelper::getAuthentiPad()	

Generate Non-Personalized TextPad (P2)



Online Banking

Sign In:

Please use this secure TextPad to enter your password.



[what's this?](#)
[forgot your password?](#)

[About Big Bank](#) | [Employment](#) | [Privacy, Security & Legal](#) | [Report Email Fraud](#) | [Home](#)

© 1999 - 2006 Big Bank. All rights reserved.

Generic TextPad and non-personalized TextPad are used for users who have not yet registered with Adaptive Risk Manager Online.



Reference APIs

Module	APIs	Notes
Server	VCryptAuth::getUserByLoginId()	
Sample	Password.jsp	
BharosaHelper	BharosaHelper::createPersonalizedAuthentiPad () BharosaHelper::createAuthentiPad()	
Client	AuthentiPad::getHTML()	

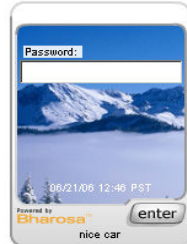
Generate Personalized TextPad or KeyPad (P3)



Online Banking

Sign In:

Please use this secure TextPad to enter your password.



[not your image & phrase?](#)
[what's this?](#)
[forgot your password?](#)

Send me to user preferences after login.

[About Big Bank](#) | [Employment](#) | [Privacy, Security & Legal](#) | [Report Email Fraud](#) | [Home](#)

© 1999 - 2006 Big Bank. All rights reserved.



Online Banking

Sign In:

Please use your mouse to enter your password on this secure KeyPad.



[not your image & phrase?](#)
[what's this?](#)
[forgot your PIN?](#)

Send me to user preferences after login.

[About Big Bank](#) | [Employment](#) | [Privacy, Security & Legal](#) | [Report Email Fraud](#) | [Home](#)

© 1999 - 2006 Big Bank. All rights reserved.

Personalized KeyPad or TextPad are similar to the Generic TextPad, except they utilize user-selected phrases and background images.



Reference APIs

Module	APIs	Notes
Server	VCryptAuth::getUserByLoginId()	
Sample	password.jsp	
BharosaHelper	BharosaHelper:: createPersonalizedAuthentiPad () BharosaHelper::createAuthentiPad()	
Client	AuthentiPad::getHTML()	

Display TextPad or KeyPad (S4 and S5)

The HTML snippet should be embedded in the password page. The HTML renders the TextPad or KeyPad using Javascript. There is a tag, which makes a HTTP request to the server to get the TextPad or KeyPad image.



Reference APIs

Module	APIs	Notes
Server	VCryptAuth::getUserByLoginId()	
Sample	password.jsp kbimage.jsp	
BharosaHelper	BharosaHelper:: createPersonalizedAuthentiPad () BharosaHelper::createAuthentiPad() BharosaHelper::imageToStream()	
Client	AuthentiPad::getHTML() KeyPadUtil::encryptImageToStream()	

Decode AuthentiPad Input (P4)

The data entered by the user is decoded by the Adaptive Risk Manager Online Utility API. The decoded value is in raw text format. The AuthentiPad, which had been used to generate the image, is used during the decoding. It is recommended that it be saved in the HTTP Session. The AuthentiPad object is serializable and can be stored in the database or file system.



Reference APIs

Module	APIs	Notes
Sample	handlePassword.jsp	
BharosaHelper	BharosaHelper::decodePadInput()	This method removes the AuthentiPad object from the HTTP Session
Client	KeyPadUtil::decodeKeyPadCode	

Validate User/ Passwd (CP1)

This is the client's existing process. The client invokes the local API to validate the user. The result of the authentication is passed on to the Adaptive Risk Manager Online Server.



Reference APIs

Module	APIs	Notes
Sample	handlePassword.jsp	

Update Authentication Status (P5)

After validating the user password, the status is updated in the Adaptive Risk Manager.



Reference APIs

Module	APIs	Notes
Server	VCryptTracker::updateAuthStatus()	
Sample	handlePassword.jsp	
BharosaHelper	BharosaHelper::updateStatus()	

Password Status (C1)

Based on the authentication status, the user is either taken to the retry page or to post authentication rule processing.

Post Authentication Rules (R3)

The post authentication rules are run after the user password is authenticated. The post authentication runtime contains security rules.

The common actions returned are

- **Allow:** Allow the authentication.
- **Block:** Block the user.
- **Challenge:** Challenge is returned if the user has registered questions. The option may not be available for Adaptive Risk Manager Only deployments.



Reference APIs

Module	APIs	Notes
Server	VCryptRulesEngine::processRules()	
Sample	handlePassword.jsp	
BharosaHelper	BharosaHelper::runPostAuthRules()	

Check Question Registration for User (C2)

If the user is already verified, it is not necessary to continue to Registration Required Rules.

Registration Required Rules (R4)

This runtime runs the rules for determining whether the user is asked to register. The registration requirement is based on the business and security requirements. The business requirement dictates whether the registration is mandatory or optional.

The common actions are

- **Register:** Force the user to register.
- **Registration Optional:** Make the registration optional for the user.
- **Skip Registration:** Skip registration for this session.



Reference APIs

Module	APIs	Notes
Server	VCryptRulesEngine::processRules()	
Sample	password.jsp	
BharosaHelper	BharosaHelper::getAuthentiPad()	

Challenge (QuestionPad) (S6)



Online Banking

Security Question:

For your safety please answer the following security question.



[not your image & phrase?](#)
[what's this?](#)
[forgot your answer?](#)

[About Big Bank](#) | [Employment](#) | [Privacy, Security & Legal](#) | [Report Email Fraud](#) | [Home](#)

© 1999 - 2006 Big Bank. All rights reserved.

Challenge pad is similar to TextPad, only the question is embedded in the pad.



Reference APIs

Module	APIs	Notes
Server	VCryptAuth::getSecretQuestions()	
Sample	Challenge.jsp	
BharosaHelper	BharosaHelper:: createPersonalizedAuthentiPad () BharosaHelper::createAuthentiPad()	
Client	AuthentiPad::getHTML()	

Check Challenge Question Answer (C3)

This step calls the server to validate whether the answer provided by the user is correct.



Reference APIs

Module	APIs	Notes
Server	VCryptAuth::authenticateQuestion() VCryptRulesEngine::processRules() VCryptTracker::updateAuthStatus()	
Sample	handleChallenge.jsp	
BharosaHelper	BharosaHelper:: validateAnswer()	

Run Challenge Rules (R5)

If the user fails to answer the question correctly, this runtime is invoked to determine whether the user is given another chance to answer the question or to block the user.

Common rule actions are

- **Challenge:** Challenge the user again.
- **Block:** Block the user.



Reference APIs

Module	APIs	Notes
Server	VCryptRulesEngine::processRules()	
Sample	handleChallenge.jsp	
BharosaHelper	BharosaHelper::validateAnswer()	

Lock Out Page (S2)

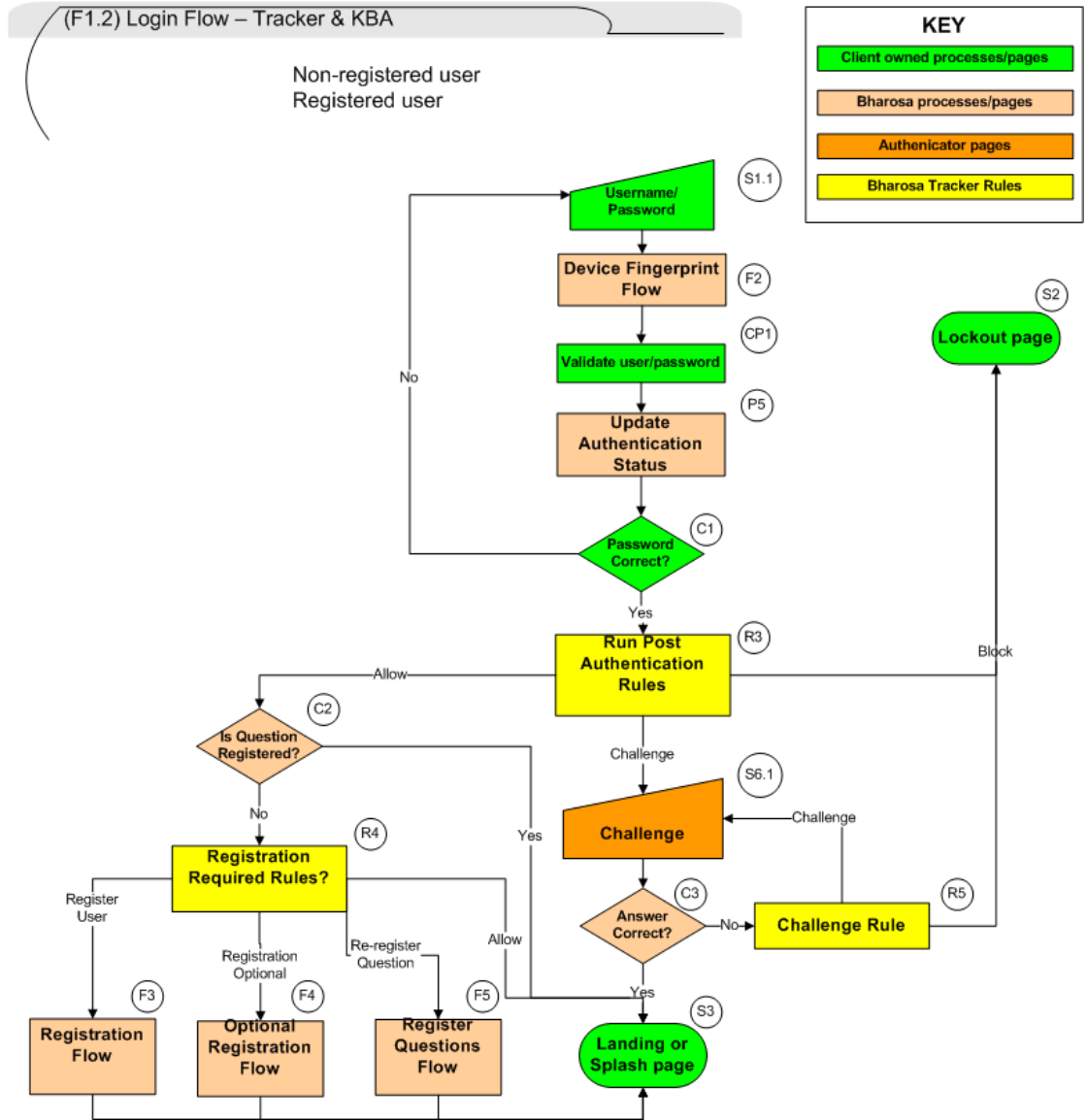
The Lock Out Page is the page the user is generally redirected to if he is blocked from authentication or if he is carrying out a transaction.

Landing or Splash Page (S3)

The Landing or Splash Page is the page the user lands on after a successful login.

Adaptive Risk Manager and KBA Scenario

This scenario is the same as the previous scenario, except it doesn't have a split login flow and there are no personalizations or AuthentiPads.



Troubleshooting

This section describes the steps to take if you experience any problems with Adaptive Risk Manager after the integration.

1. Confirm that `bharosa_properties` is in the `classes` directory.
2. Confirm that you have customized `bharosa_client.properties`.
3. Make sure only one copy of the `bharosa_client.properties` appears in the classpath. If multiple property files are needed, ensure that they are all identical.
4. Make sure the directory specified in `log4j.xml` for logfiles is present and write-accessible.
5. Update `log4j.xml` to different levels of logging for troubleshooting application issues.
6. Check the log levels in `log4j.xml` for recommended levels in case of space issues.