**Agrobank**

**UPASS Authentication**

**Technical Specification**

28 April 2011

Reference: AGRO/TECH/UPASS/20110428/v1.2.2

**Prepared By:**



Penril Datability (SEA) Sdn Bhd (384550-U)

Suite A-07-07 Plaza Mon’t Kiara

No. 2, Jalan Kiara, Mon’t Kiara

50480 Kuala Lumpur, Malaysia

Tel: (603) 6201 2622 Fax: (603) 6201 7622**Document Administration**

**Document Amendment Log**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Version No** | **Date Updated** | **Description of Change** | **Updated By**  **<Name, Dept>** | **Reviewed By**  **<Name, Dept>** | **Approved By <Name, Dept>** |
| 1.0 | 21/08/2008 | Initial Release | Jelson Low, Technical Services | Tan Lee Yong, Technical Services | Tan Lee Yong, Technical Services |
| 1.1 | 12/04/2009 | Amended table design, added OTP and updated penril logo | Jelson Low, Danniell Cheang, Professional Services | Tan Lee Yong, Professional Services | Tan Lee Yong, Professional Services |
| 1.2 | 17/01/2011 | Add application architecture | Tan Lee Yong, Professional Services | Jelson Low, Danniell Cheang, Professional Services | Tan Lee Yong, Professional Services |
| 1.2.1 | 31/01/2011 | Add error code | Tan Lee Yong, Professional Services |  |  |

Table of Contents

[System Overview 4](#_Toc291772354)

[Application Architecture 5](#_Toc291772355)

[Password Authentication Methodology 7](#_Toc291772356)

[Application Specification 9](#_Toc291772357)

[SHA-1 9](#_Toc291772358)

[UPASS Security Specification 9](#_Toc291772359)

[UPASS Available Parameter Settings 10](#_Toc291772360)

[UPASS Data Catalog 12](#_Toc291772361)

[Entity Relationship Diagram 12](#_Toc291772362)

[Data Dictionary 13](#_Toc291772363)

[Functional Specification 14](#_Toc291772364)

[Overview 14](#_Toc291772365)

[Create/Add User 15](#_Toc291772366)

[Verify User and Password 17](#_Toc291772367)

[Change Password 18](#_Toc291772368)

[Reset Password 20](#_Toc291772369)

[Generate OTP 22](#_Toc291772370)

[Verify OTP 23](#_Toc291772371)

[Lock User 24](#_Toc291772372)

[Suspend User 25](#_Toc291772373)

[Activate User 26](#_Toc291772374)

[Appendix A – Error Code 27](#_Toc291772375)

# System Overview

UPASS is a lightweight yet practical authentication system offers access control of multiple, related but independent software system. It offers authentication services such as static password verification, dynamic PIN such as One-Time PIN, session PIN as well as Vasco Hardware Token.

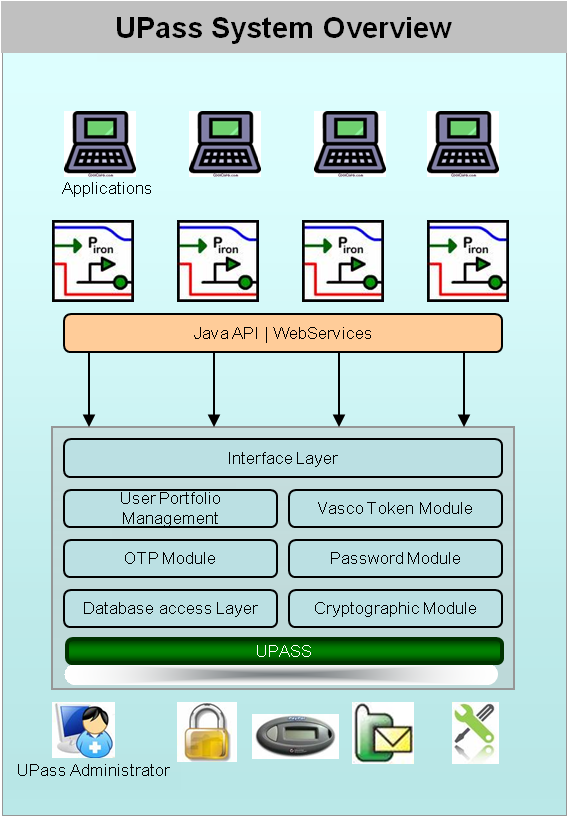


Figure 1 UPASS Overview

# Application Architecture

**Architecture of Using Web Service Integration and as a standalone authentication system**



Figure 2 System Architecture 1

**Architecture of Using Direct Java API Integration and as an integrated authentication system**



Figure 3 System Architecture 2

**System Specifications**

|  |  |  |
| --- | --- | --- |
| No | Items | Description |
|  | Programming Language Used | Java |
|  | Integration Method | Web Service, Java API |
|  | Supported Application Server | Oracle WebLogic 11g, IBM WebSphere 7, JBoss 5, Apache TomCat 5.5 |
|  | Supported Web Server | Sun One Web Server 6.1 & 7, Apache |
|  | Supported Database | Oracle 10g |
|  | Supported OS | Sun Solaris, RedHat Linux |
|  | Used Cryptography Algorithm | SHA-1 |

**System Minimum Requirements**

|  |  |  |
| --- | --- | --- |
| No | Items | Description |
|  | Application Server | RAM: 4GB  Disk Space: 50GB  CPU: 1.2 Ghz and above |
|  | Database Server | RAM: 16GB  Disk Space: 100GB  CPU: 1.2 Ghz and above |

## Password Authentication Methodology

1. Uses SHA-1 Hashing Algorithm
2. Adds Noise
3. Apply another layer hashing – Base16 Encode
4. Scramble the result string



Figure 4 Password Hashing Methodology

**Cryptographic Hash Function Used Java Classes**

|  |  |  |
| --- | --- | --- |
| No | Java Method | Description |
|  | **java.security.MessageDigest** | For creating digests. This class allows specifying the digest algorithm we wish to use. In this case, we use SHA-1. |
|  | **java.lang.String.getBytes** | For obtaining a byte sequence from the input String, specifying a fixed encoding ("iso-8859-1") |
|  | **java.lang.String.StringTokenizer**  **java.lang.String.StringBuffer** | To scramble the digested bytes |

# Application Specification

UPASS addresses financial Institution security requirements on handling password leveraging on industrial standard Java Cryptography Extension framework. Particularly, SHA-1 hashing algorithm is uses for the password hashing.

### SHA-1

SHA stands for Secure Hash Algorithm. SHA-1 is the most widely used of the existing SHA hash functions, and is employed in several widely-used security applications and protocols. In 2005, security flaws were identified in SHA-1, namely that a mathematical weakness might exist, indicating that a stronger hash function would be desirable. SHA-1 produces a 160-bit message digest based on the design of the MD4 and MD5 message digest algorithms, but has a more conservative design.

## UPASS Security Specification

|  |  |  |
| --- | --- | --- |
| User ID and Password Control | Minimum Requirements | Remark |
| 1. User ID | Single or combination of numeric, alpha or alphanumeric. | The rule is not parameterized. |
| 1. Maximum password expiration | **90 days** (The system will force a reset after this period has elapsed) | System wide support and can be parameterized via the system configuration file |
| 1. Minimum password length | **8** characters | This is standard feature. |
| 1. Maximum failed log-in attempts | **3 times.** (The User ID will subsequently be locked (disabled)) | The maximum number of attempts can be parameterized via the system configuration file. If the User ID is locked, the user must send a request to the administrator to reset the password. The application will force the user to change their password upon login. |
| 1. Inactive/Dormant ID for more than 90 days | User ID is automatically disabled by the system | The number of days can be parameterized via the system configuration file. |
| 1. Duplicate password control | For a minimum of **6** generations, the same passwords cannot be reused. | The number of generations can be parameterized via the system configuration file. |
| 1. Last sign on information displayed | The last signed on details will be displayed to the user upon logging into the system | This is standard feature. |
| 1. First time sign on must force a password change | First time sign on user must be forced to change the defaulted password. | This is standard feature. |
| 1. Changing of passwords by user | Users may change their passwords anytime they wish | This is standard feature. |
| 1. Password composition | Must be combination of alphanumeric, special characters, lower/upper case (if possible) | This is standard feature.  The password must be combination of at least 1 alphabet and 1 numeric value and is case sensitive. |
| 1. Deletion of User IDs from application | User IDs that are obsolete must be able to be deleted (instead of just suspended) from the application | The same user id cannot be reused even if deleted. The deleted user will not show on user enquiry screen, but the record will still be stored in the database. |
| 1. Password storage | Requires Hashing | Combination of Login ID and Password is hashed using SHA-1 before being stored into Database. The hashed value will be used for authentication |

## UPASS Available Parameter Settings

|  |  |  |
| --- | --- | --- |
| Parameter | Setting | Remarks |
| PASSWORD\_MAX\_ERROR | 3 | Maximum password failed attempt before user is locked/suspend |
| PASSWORD\_GENERATION | 3 | Number of used password in history that user cannot set as desire password |
| PASSWORD\_EXPIRY\_DAY | 0 | Number of days before password is expired and need to be change to success in password verification |
| PASSWORD\_MIN\_LENGTH | 8 | Minimum length of passowrd |
| PASSWORD\_COMPLEXITY | 0/1 | 0=Alphanumeric only  1=At least an upper case [A-Z] and at least a lower case [a-z] and at least a numeric [0-9] and at least a special character [!@#$%^&\*()\_+<>?] |
| SUPERVISOR\_ID\_SUSPEND | 30 | Suspend Root or Admin ID for max password error, in minute |
| TAC\_DEFAULT\_TIMEOUT | 7200 | Denotes validity of life span in second for TAC |
| TAC\_LIFTSPAN\_TIMEOUT | 7200 | Denotes the validity of life span second after the 1st used |
| TAC\_REPEAT\_TIMEOUT | 300 | Denotes the time in second allowed for next TAC generation |
| TAC\_MAX\_ERROR | 3 | Maximum TAC verification failed before user is locked |
| TAC\_MAX\_USE | 3 | Maximum number used for TAC |
| SMTP\_SERVER\_URL |  | Mail server SMTP URL |
| SMTP\_PORT |  | The Mail server SMTP port |
| SMTP\_USER\_NAME |  | The Mail server SMTP login username |
| SMTP\_USER\_PSW |  | The Mail server SMTP login password |

# UPASS Data Catalog

## Entity Relationship Diagram

## upass ER Diagram.jpg

## Data Dictionary

**TB\_AM\_USERMAS**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Column name** | **Data type** | **Unique** | **Len** | **Null?** | **Description** | **Default** | **PK** |
| M\_USER\_ALIAS | VARCHAR2 | Yes | 20 | Not null | User Alias |  | YES |
| M\_USER\_ID | NUMBER |  | 9 | Not null | User ID No |  |  |

**TB\_AM\_USER**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Column name** | **Data type** | **Unique** | **Len** | **Null?** | **Description** | **Default** | **PK** |
| U\_USER\_ID | NUMBER | Yes | 9 | NOT NULL | User ID No |  | YES |
| U\_USER\_ALIAS | VARCHAR2 |  | 20 | NOT NULL | User Alias |  |  |
| U\_USER\_TYPE | NUMBER |  | 1 | NOT NULL | User Type | 2 |  |
| U\_USER\_DESC | VARCHAR2 |  | 40 | YES | User Description |  |  |
| U\_DATE\_CREATED | TIMESTAMP |  | 6 | NOT NULL | User Date Creation | SYSTIMESTAMP |  |
| U\_DATE\_LAST\_USED | TIMESTAMP |  | 6 | YES | User Date Last Used |  |  |
| U\_USE\_COUNT | NUMBER |  | 6 | NOT NULL | User Use Count | 0 |  |
| U\_DATE\_LAST\_ACT | TIMESTAMP |  | 6 | YES | User Date Last Active |  |  |
| U\_DATE\_LAST\_LOCK | TIMESTAMP |  | 6 | YES | User Date Last Lock |  |  |
| U\_USER\_STATE | NUMBER |  | 1 | NOT NULL | User Starte | 1 |  |
| U\_DATE\_LOCK\_FROM | TIMESTAMP |  | 6 | YES | User Date Lock From |  |  |
| U\_DATE\_LOCK\_TO | TIMESTAMP |  | 6 | YES | User Date Lock To |  |  |
| P\_PASSWD\_CIPHERTEXT | VARCHAR2 |  | 64 | YES | Password in Ciphertext |  |  |
| P\_PASSWD\_STATE | NUMBER |  | 1 | NOT NULL | Password State | 0 |  |
| P\_DATE\_GENERATED | TIMESTAMP |  | 6 | NOT NULL | Password Date Generated | SYSTIMESTAMP |  |
| P\_DATE\_FIRST\_USED | TIMESTAMP |  | 6 | YES | Password Date First Used |  |  |
| P\_DATE\_LAST\_USED | TIMESTAMP |  | 6 | YES | Password Date Last Used |  |  |
| P\_USE\_COUNT | NUMBER |  | 6 | NOT NULL | Password Use Count | 0 |  |
| P\_ERROR\_COUNT | NUMBER |  | 2 | NOT NULL | Password Error Count | 0 |  |
| P\_EXPIRED\_STATUS | NUMBER |  | 1 | NOT NULL | password Expired Status | 1 |  |
| P\_DATE\_EXPIRED | TIMESTAMP |  | 6 | YES | Password Date Expired |  |  |
| P\_PASSWD\_HIST | VARCHAR2 |  | 256 | YES | Password History. |  |  |

**TB\_AM\_TAC**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Column name** | **Data type** | **Unique** | **Len** | **Null?** | **Description** | **Default** | **PK** |
| T\_USER\_ID | NUMBER | Yes | 9 | NOT NULL | User Id No |  | YES |
| T\_TAC\_CIPHERTEXT | VARCHAR2 |  | 64 | YES | TAC Ciphertext |  |  |
| T\_TAC\_STATE | NUMBER |  | 1 | NOT NULL | TAC State | 0 |  |
| T\_DATE\_GENERATED | TIMESTAMP |  | 6 | NOT NULL | Date Generated | SYSTIMESTAMP |  |
| T\_DATE\_FIRST\_USED | TIMESTAMP |  | 6 | YES | Date On First TAC Use |  |  |
| T\_DATE\_LAST\_USED | TIMESTAMP |  | 6 | YES | Date On Last TAC Use |  |  |
| T\_USE\_COUNT | NUMBER |  | 6 | NOT NULL | Use Count of TAC | 0 |  |
| T\_ERROR\_COUNT | NUMBER |  | 2 | NOT NULL | Error Count of TAC | 0 |  |
| T\_SESSION\_ID | VARCHAR2 |  | 32 | YES | Session ID No |  |  |

# Functional Specification

## Overview

UPASS System provides the following module features, each module feature will be explain further in details at the following sections.

1. Create/Add User
2. Verify User and Password
3. Change Password
4. Reset Password
5. Generate OTP
6. Verify OTP
7. Lock User
8. Suspend User
9. Unlock User

### Create/Add User

**Overview**

This function is to create a new user provided with new User Alias and Password.

**Process Flow**



**Fields Specification**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field | Description | M/O | Format | Remarks |
| User Alias | User desires user alias | M | Alphanumeric | Minimum length = 8 |
| Password | User desires password | M | Alphanumeric | Validity on composition setting  Normal=Alphanumeric only  Complex= at least an upper case [A-Z] and at least a lower case [a-z] and at least a numeric [0-9] and at least a special character [!@#$%^&\*()\_+<>?] |
| Description | User description | O | Alphanumeric |  |
| User Type | UPASS system user type | M | Numeric | 0 – Root user (Super admin) \*\*  1 – Admin user (UPASS admin user)  2 – Normal user (System user)  \*\* Note: Root user creation function are not valid after 1st time setup |
| Root User Alias | Root user alias for UPASS System | M | Alphanumeric | For admin user creation, User Type = ‘1’ |
| Root User Password | Root user password | M | Alphanumeric | For admin user creation, User Type = ‘1’ |
| Admin User Alias | Admin user alias for UPASS System | M | Alphanumeric | For normal user creation, User Type = ‘2’ |
| Admin User Password | Admin user password | M | Alphanumeric | For normal user creation, User Type = ‘2’ |

### Verify User and Password

**Overview**

This function is to verify user ID and password that registered to UPASS System.

**Process Flow**



**Fields Specification**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field | Description | M/O | Format | Remarks |
| User Alias | User alias | M | Alphanumeric |  |
| Password | User password | M | Alphanumeric |  |

### Change Password

**Overview**

This function allow user to change password provided with successful old password verification.

**Process Flow**



**Fields Specification**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field | Description | M/O | Format | Remarks |
| User Alias | User alias | M | Alphanumeric |  |
| Old Password | User current password | M | Alphanumeric |  |
| New Password | User new password | M | Alphanumeric | Validity on composition setting  Normal=Alphanumeric only  Complex= at least an upper case [A-Z] and at least a lower case [a-z] and at least a numeric [0-9] and at least a special character [!@#$%^&\*()\_+<>?] |

### Reset Password

**Overview**

This function allow user to reset password in case user has forgotten his/her current password.

**Process Flow**



**Fields Specification**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field | Description | M/O | Format | Remarks |
| User Alias | User alias | M | Alphanumeric |  |
| New Password | User new password | M | Alphanumeric | Validity on composition setting  Normal=Alphanumeric only  Complex= at least an upper case [A-Z] and at least a lower case [a-z] and at least a numeric [0-9] and at least a special character [!@#$%^&\*()\_+<>?] |
| Root User Alias | Root user alias for UPASS System | M | Alphanumeric | For admin user reset password only, User Type = ‘1’ |
| Root User Password | Root user password | M | Alphanumeric | For admin user reset password only, User Type = ‘1’ |
| Admin User Alias | Admin user alias for UPASS System | M | Alphanumeric | For normal user reset password only, User Type = ‘2’ |
| Admin User Password | Admin user password | M | Alphanumeric | For normal user reset password only, User Type = ‘2’ |

### Generate OTP

**Overview**

This function is to generate a One Time Password (OTP). The generated OTP is used for OTP verification within a valid period and meeting our security setting

**Process Flow**



**Fields Specification**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field | Description | M/O | Format | Remarks |
| User Alias | User alias | M | Alphanumeric |  |

### Verify OTP

**Overview**

This function is to verify the user’s OTP generated by UPASS System.

**Process Flow**



**Fields Specification**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field | Description | M/O | Format | Remarks |
| User Alias | User alias | M | Alphanumeric |  |
| OTP | One time password generated by UPASS System | M | Numeric | A 6 digit length OTP |

### Lock User

**Overview**

This function allows UPASS System to lock use. Locked user will not be able to perform any other function in UPASS System.

**Process Flow**



**Fields Specification**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field | Description | M/O | Format | Remarks |
| User Alias | User alias | M | Alphanumeric |  |
| Root User Alias | Root user alias for UPASS System | M | Alphanumeric | To lock admin user only, User Type = ‘1’ |
| Root User Password | Root user password | M | Alphanumeric | To lock admin user only, User Type = ‘1’ |
| Admin User Alias | Admin user alias for UPASS System | M | Alphanumeric | To lock normal user only, User Type = ‘2’ |
| Admin User Password | Admin user password | M | Alphanumeric | To lock normal user only, User Type = ‘2’ |

### Suspend User

**Overview**

This function allows UPASS System to suspend user within a period of time. This function is normally used to suspend root user and admin user who have an invalid login within a period of time.

**Process Flow**



**Fields Specification**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field | Description | M/O | Format | Remarks |
| User Alias | User alias | M | Alphanumeric |  |
| Root User Alias | Root user alias for UPASS System | M | Alphanumeric | To suspend admin user only, User Type = ‘1’ |
| Root User Password | Root user password | M | Alphanumeric | To suspend admin user only, User Type = ‘1’ |
| Admin User Alias | Admin user alias for UPASS System | M | Alphanumeric | To suspend normal user only, User Type = ‘2’ |
| Admin User Password | Admin user password | M | Alphanumeric | To suspend normal user only, User Type = ‘2’ |

### Activate User

**Overview**

This function allows UPASS System to activate locked and suspended user. Enabling UPASS user to perform other functions in UPASS System.

**Process Flow**



**Fields Specification**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field | Description | M/O | Format | Remarks |
| User Alias | User alias | M | Alphanumeric |  |
| Root User Alias | Root user alias for UPASS System | M | Alphanumeric | To activate admin user only, User Type = ‘1’ |
| Root User Password | Root user password | M | Alphanumeric | To activate admin user only, User Type = ‘1’ |
| Admin User Alias | Admin user alias for UPASS System | M | Alphanumeric | To activate normal user only, User Type = ‘2’ |
| Admin User Password | Admin user password | M | Alphanumeric | To activate normal user only, User Type = ‘2’ |

# Appendix A – Error Code

|  |  |  |
| --- | --- | --- |
| No | Error Code | Description |
| 1 | 0 | Successful |
| 2 | 1 | Invalid credential |
| 3 | 2 | Exceeded maximum login attempts |
| 4 | 3 | Password has been previously used (within password history list) |
| 5 | 4 | Password expired |
| 6 | 5 | Invalid status |
| 7 | 6 | User not found |
| 8 | 7 | VASCO not assigned |
| 9 | 8 | VASCO already assigned |
| 10 | 9 | Token does not exist |
| 11 | 10 | Invalid credential to application |
| 12 | 11 | User already exist |
| 13 | 20 | Reset password is required |
| 14 | 21 | Reset password and self registration is required |
| 15 | 96 | Password does not comply with complexity rule |
| 16 | 97 | Invalid input |
| 17 | 98 | System unavailable |
| 18 | 99 | Unknown error |