**Standard Security Requirements for the Online banking Application**

## 1. Identification Requirements

Unique User ID Requirements.

| **Req. ID IDEN-OBA-001** | **Category: SECURITY** |
|---|---|
| **Subcategory(ies)/Tags** | Identification, User ID, Login |
| **Name** | Unique User ID |
| **Requirement** | The online banking application shall uniquely identified each user of the system with a unique user ID |
| **Use Case(s)** | Initial login to the system |
| **Rationale** | It is of utmost importance to identify individual users to provide reliable accountability for actions. Shared accounts prevent accountability and auditability of actions performed on the application. |
| **Priority Critical** | **Critical** /High/Medium/Low |
| **Constraints** | NA |
| **Comments** | A user may be a human, an automated process or another system that requests a session with the online banking application to perform a task. A user may have multiple user IDs as long as the multiple user IDs unambiguously and uniquely identify the user. |
| **Test Case Ref #** | OBA-IDEN-001 |

Preventing Backdoors in the Online Banking Authentication System

| **Req. ID IDEN-OBA-002** | **Category: SECURITY** |
|---|---|
| **Subcategory(ies)/Tags** | Identification, User ID, Login, Backdoor |
| **Name** | Backdoor Prevention |
| **Requirement** | All interfaces of the software that are accessed for performing any action shall have the capability to recognize the user ID. |

| | |
|---|---|
| **Use Case(s)** | Initial login to the system, batch jobs, API calls,N/W interface |
| **Rationale** | Identification must be applied across all the online banking application interfaces. In the event that a “backdoor” exists through which access is granted with no identification,<br>the security of the system would be compromised. |
| **Priority Critical** | Critical/**High**/Medium/Low |
| **Constraints** | NA |
| **Comments** | The term “interface” refers to the point of entry into a system. It can be a network interface, user interface, or other system interface, as appropriate. |
| **Test Case Ref #** | OBA-IDEN-002 |

**2. Authentication Requirements**

An authentication requirement is any security requirement that specifies the extent to which a business, application, component, or center shall verify the identity of its externals

**Credential Security**

| **Req. ID AUTH-OBA-001** | **Category: SECURITY** |
|---|---|
| **Subcategory(ies)/Tags** | Authentication, Credentials, Passwords, Hashing |
| **Name** | Credential Security |
| Requirement | The online banking application shall store the information used for authentication in a secure manner, using public and widely accepted cryptoalgorithms. |
| **Use Case(s)** | Password Storage |
| **Rationale** | Authenticating information must be stored in such a way so that a third party without authorization to do so cannot easily obtain it. For example, static passwords should be passed through a one-hash function, and only the hash should be stored. |
| **Priority Critical** | **Critical**/High/Medium/Low |

| | |
|---|---|
| **Constraints** | NA |
| **Comments** | Per-user salting is recommended for storing password hashing to provide additional level of security. |
| **Test Case Ref #** | OBA-AUTH-001 |

**Protect against Brute force attack-Credential Guessing**

| **Req. ID AUTH-OBA-002** | **Category: SECURITY** |
|---|---|
| **Subcategory(ies)/Tags** | Authentication, Credential Enumeration, Login |
| **Name** | Protect Credential Guessing |
| **Requirement** | The online banking application shall not provide feedback to the user during the authentication procedure other than “invalid” (i.e., it shall not reveal which part of the authentication [e.g., user ID or password] procedure is incorrect). |
| **Use Case(s)** | User Login |
| **Rationale** | Feedback that is too descriptive can inadvertently give out information regarding which part of an authentication procedure is incorrect, thus allowing an attacker to narrow his or her search. |
| **Priority Critical** | Critical/**High**/Medium/Low |
| **Constraints** | Some other application flows (user registration/ signup) may actually give up this information indirectly. |
| **Comments** | As long as strong brute-force detection and protection exists, the application shall be able to maintain good security levels against this type of attack. |
| **Test Case Ref #** | OBA-AUTH-002 |

**Server Authentication**

| **Req. ID AUTH-OBA-003** | **Category: SECURITY** |
|---|---|
| **Subcategory(ies)/Tags** | Authentication, Credentials, Login |
| **Name** | Server Authentication |
| **Requirement** | The system shall have the ability to authenticate itself to the user and to other systems during session establishment. |
| **Use Case(s)** | Login over SSL into a Web application |
| **Rationale** | In most cases, authentication focuses solely on the client authentication to the server. However, without proper server-to-client authentication, it may be possible for a third party to impersonate a server and obtain client's authentication credentials. |
| **Priority Critical** | **Critical**/High/Medium/Low |
| **Constraints** | With mechanisms such as SSH, Time of First Use (TOFU) problems exist. |
| **Comments** | Other authentication mechanisms where this requirement is not met shall be disabled or redirected to this form authentication (e.g., http to https). |
| **Test Case Ref #** | OBA-AUTH-003 |

**Reauthentication**

| **Req. ID AUTH-OBA-004** | **Category: SECURITY** |
|---|---|
| **Subcategory(ies)/Tags** | Authentication, Credentials, Login |
| **Name** | Reauthentication |
| **Requirement** | The system shall have the ability to reauthenticate the user during an active session. |
| **Use Case(s)** | Before performing critical transactions |
| **Rationale** | Periodic reauthentication improves a system's ability to |

| | |
|---|---|
| | withstand session “hijacking” attacks, in which a third party assumes control of a previously authenticated session. |
| **Priority Critical** | Critical/**High**/Medium/Low |
| **Constraints** | N/A |
| **Comments** | Other risk management and fraud detection/prevention controls shall exist for the primary protection of the application data. |
| **Test Case Ref #** | OBA-AUTH-004 |

**Password Complexity**

| **Req. ID AUTH-OBA-005** | **Category: SECURITY** |
|---|---|
| **Subcategory(ies)/Tags** | Authentication, Credentials, Login |
| **Name** | Password Complexity |
| **Requirement** | The system shall require that the authentication information is configurable to administrator specified<br>Characteristics for minimum length, alphabetic characters and numeric or special characters. |
| **Use Case(s)** | N/A |
| **Rationale** | Use of trivial and predictable authenticators makes it easier for a third party to obtain an authenticator through brute-force attacks, such as dictionary attacks and other cracking methods. |
| **Priority Critical** | Critical/High/**Medium**/Low |
| **Constraints** | N/A |

| | |
|---|---|
| **Comments** | N/A |
| **Test Case Ref #** | OBA-AUTH-005 |

## 3. Authorization Requirements

An authorization requirement is any security requirement that specifies the access and usage privileges of authenticated users and the online banking application

| **Req. ID AUTHR-OBA-001** | **Category: SECURITY** |
|---|---|
| **Subcategory(ies)/Tags** | Authorization, Credentials, Access Control |
| **Name** | Access Rights |
| **Requirement** | The system shall not allow access to system resources without checking the assigned rights and privileges of the authenticated user. |
| **Use Case(s)** | Access to different functionalities of the application. |
| **Rationale** | Authorization is useless unless tied to something that maps identification to rights or privileges.<br>Authorization controls must be applied across all users, resources, and interfaces. |
| **Priority Critical** | **Critical**/High/Medium/Low |
| **Constraints** | N/A |
| **Comments** | For a Web application, this would translate to every Web page request having a routine that checks for access rights before processing the request. |
| **Test Case Ref #** | OBA-AUTHR-001 |

### Session Timeout

| **Req. ID AUTHR-OBA-001** | **Category: SECURITY** |
|---|---|
| **Subcategory(ies)/Tags** | Session Management, Authorization, Authentication |
| **Name** | Session Timeout |
| **Requirement** | The system shall provide a “timeout” feature so that if during an active session there |

| | |
|---|---|
| | has not been any exchange of messages across the connection for an administrator-specified period of time, the system shall drop the connection and require a successful reauthentication to regain<br>access. |
| **Use Case(s)** | N/A |
| **Rationale** | Leaving open active sessions increases the possibility of session hijacking as well as disclosure of data. |
| **Priority Critical** | **Critical**/High/Medium/Low |
| **Constraints** | N/A |
| **Comments** | N/A |
| **Test Case Ref #** | OBA-AUTHR-002 |

**User and group Priviledges**

| **Req. ID AUTHR-OBA-003** | **Category: SECURITY** |
|---|---|
| **Subcategory(ies)/Tags** | Authorization, Credentials, Access Control |
| **Name** | User and Group Privileges |
| **Requirement** | The system shall have features to assign user and group privileges (i.e., access permissions) to user IDs (not authentication information). |
| **Use Case(s)** | N/A |
| **Rationale** | Assigning user privileges to authenticators may compromise their confidentiality. Instead, assigning privileges to a user enables authorization checking without requiring disclosure of authentication. |
| **Priority Critical** | Critical/**High**/Medium/Low |
| **Constraints** | N/A |
| **Comments** | N/A |
| **Test Case Ref #** | OBA-AUTHR-003 |

**Role based Access Control**

| **Req. ID AUTHR-OBA-004** | **Category: SECURITY** |
|---|---|

| | |
|---|---|
| **Subcategory(ies)/Tags** | Authorization, Credentials, Access Control |
| **Name** | Role-Based Access Control (RBAC) |
| **Requirement** | The system shall provide an enforceable mechanism through which users can be segmented into roles (e.g., administrator), involving access to security features and other administrative functions. |
| **Use Case(s)** | N/A |
| **Rationale** | Providing for role-based access control allows individuals to have access based on a specific purpose, rather than just their identity. This minimizes the risk associated with providing superuser<br>or other privileged access to individual users. |
| **Priority Critical** | **Critical**/High/Medium/Low |
| **Constraints** | N/A |
| **Comments** | N/A |
| **Test Case Ref #** | OBA-AUTHR-004 |

## 4. Integrity Requirements

### Source Identification

| **Req. ID INT-OBA-001** | **Category: SECURITY** |
|---|---|
| **Subcategory(ies)/Tags** | Data Integrity, Data Protection, Audit |
| **Name** | Source Identification |
| **Requirement** | The system shall have the capability to propagate, when requested, the original user identifier to the destination. |
| **Use Case(s)** | N/A |
| **Rationale** | The source identification should be available to further back-end systems for audit purposes. |
| **Priority Critical** | Critical/High/**Medium**/Low |
| **Constraints** | N/A |
| **Comments** | N/A |

| | |
|---|---|
| **Test Case Ref #** | OBA-INT-001 |

**Integrity of Logs**

| **Req. ID INT-OBA-002** | **Category: SECURITY** |
|---|---|
| **Subcategory(ies)/Tags** | Data Integrity, Data Protection, Logging |
| **Name** | Integrity of Logs |
| **Requirement** | The system shall have the capability to protect the integrity of audit log records by generating integrity checks (e.g., checksums or secure hashes) when the log records are created, and by verifying the integrity check data when the record is accessed. |
| **Use Case(s)** | N/A |
| **Rationale** | A common technique, as part of an attack, is to alter the log and audit records on a system to hide unauthorized activity. Integrity checks on these records can help prevent such activity. |
| **Priority Critical** | **Critical**/High/Medium/Low |
| **Constraints** | N/A |
| **Comments** | N/A |
| **Test Case Ref #** | OBA-INT-002 |

## 5. Non- Repudiation requirements

**Time Stamping**

| **Req. ID NONR-OBA-001** | **Category: SECURITY** |
|---|---|
| **Subcategory(ies)/Tags** | Nonrepudiation, Logging, Accountability |
| **Name** | Time Stamping |
| **Requirement** | The system shall have the capability to securely link received information with the originator of the information and other characteristics such as time and date. |
| **Use Case(s)** | N/A |

| | |
|---|---|
| **Rationale** | To enforce nonrepudiation, it is necessary to tie specific data to a user or system, as well as to the time at which it was sent. This supports accountability of actions and is a core concept of non-repudiation. |
| **Priority Critical** | Critical/High/**Medium**/Low |
| **Constraints** | N/A |
| **Comments** | N/A |
| **Test Case Ref #** | OBA-NONR-001 |