Critical Controls Implementation for Oracle E-Business Suite

Helping Organizations Securely Migrate to and Operate Oracle Applications in the Cloud
The permanent and official location for Software Defined Perimeter Working Group is
https://cloudsecurityalliance.org/research/working-groups/enterprise-resource-planning/
The Enterprise Resource Planning (ERP) Working Group (WG) seeks to develop best practices to enable organizations that run their business on large ERP implementations—such as systems applications and products (SAP) or Oracle applications—to migrate to and operate in cloud environments securely. Every ERP deployment is unique to each organization. In most cases, organizations spend months (if not years) customizing their SAP or Oracle implementations while also spending significant monies with third-party contractors to get the implementations completed. This undertaking makes standard security measures more challenging to implement due to the differences of each deployment. With the complexity of these large implementations, combined with the criticality of data and processes housed in these applications, industry best practices must be established to provide companies with security guidelines when migrating to the cloud to protect the organization’s critical infrastructure.
# Table of Contents

Introduction ......................................................................................................................................... 5  
How To Use This Document ................................................................................................................. 6  
Controls Implementation ..................................................................................................................... 7  
  - USR01 - Secure Authentication ...................................................................................................... 7  
  - USR02 - User Accounts Management ............................................................................................ 9  
  - USR03 - Role-Based Access Control ............................................................................................. 11  
  - USR04 - Emergency Access ......................................................................................................... 13  
  - USR05 - Segregation of Duties ..................................................................................................... 15  
  - USR06 - Secure User Provisioning/Deprovisioning .................................................................... 16  
  - USR07 - Enterprise Resource Planning (ERP) Accounts Security .................................................. 18  
  - APP01 - Secure Landscape ........................................................................................................... 20  
  - APP02 - Secure Baseline Configurations ...................................................................................... 22  
  - APP03 - Implementation of Security Patches ............................................................................... 27  
  - APP04 - Secure Communications ................................................................................................ 29  
  - APP05 - Change Management Controls ....................................................................................... 31  
  - APP06 - Secure Extensions .......................................................................................................... 33  
  - INT01 - Secure Integrations and Application Programming Interface (API) .............................. 35  
  - DAT01 - Continuous Monitoring .................................................................................................. 37  
  - DAT02 - Data Separation ............................................................................................................. 40  
  - DAT03 - Data Encryption .............................................................................................................. 43  
  - BUS01 - Inventory of Business Assets, Data, and Processes .......................................................... 45  
  - BUS02 - Business Process Controls ............................................................................................. 46  
  - BUS03 - Continuous Compliance .................................................................................................. 47
Introduction

This paper was created to help organizations determine what security changes are needed when deploying Oracle E-Business Suite (EBS) in the cloud. For clarity, this document will use the terms “cloud” and “Infrastructure as a Service (IaaS)” interchangeably.

The impact of the cloud is transformative on many levels, as it offers many benefits and opportunities. However, the cloud creates new challenges, too. EBS clients should address cloud migration as much more than a data center migration project. Cloud migration is a significant opportunity to “start over” regarding security by using best practices, tools, services, and techniques unique to the cloud. The net result of moving an EBS implementation to the cloud can significantly enhance and strengthen an organization’s security posture.

It can also bring severe risks if not done right.

Most crucially, EBS customers must recognize that Cloud Service Providers (CSPs) are NOT responsible for EBS application security. Public cloud vendors offer a shared responsibility model for Infrastructure-as-a-Service (IaaS) where they provide a virtual datacenter—but clients are responsible for the applications. Clients are also legally responsible for data wherever it is hosted (either on-premise or in the cloud).

Thus, several significant architectural changes are required to deploy EBS in the cloud securely. Network isolation and segmentation of database and web servers are mandatory; otherwise, the EBS implementation may be inadvertently accessible to the Internet. Encryption is no longer optional. Data must be encrypted at-rest, in-motion, and in-use. Auditing and monitoring (e.g., using security information and event management (SIEM) solutions) are often rarely enabled when on-premise, but to understand who “did what” it’s now required to provide evidence of control over cloud deployments. Lastly, vulnerability management is of paramount importance. Security patches must be quickly identified and applied, and new tools and processes are needed to guard against insecure configuration drift.
How To Use This Document

This document is a reference document to promote best practices for cloud deployments of the Oracle E-Business Suite. It does not replace Oracle Corporation's (the vendor's) documentation and specific instructions. Specifically, this document should be used as part of the Cloud Security Alliance (CSA) ERP Working Group's on-going dialogue. Please refer to the ERP Working Group's site for additional resources.

Each of the 20 controls presented in this document for the Oracle E-Business Suite is directly mapped to CSA's overarching Top 20 Critical Controls for Cloud Enterprise Resource Planning Customers.

Two parts define each control:

- **Control Implementation:** The control implementation defines the rationale for the control. Information technology (IT) leaders and information security and compliance professionals will benefit from understanding and mapping each control into their overarching IT operational and security and compliance controls.
- **Checklist:** To implement each control, an inventory of specific requirements and/or steps are identified. Database administrators (DBAs) and system administrators can use this section to implement the control.

To make full use of this document, readers should be familiar with CSA's Cloud Controls Matrix (CCM) and the Center for Internet Security's (CIS) Benchmarks for security hardening.
## Controls Implementation

<table>
<thead>
<tr>
<th>Domain</th>
<th>Cloud ERP Users</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control ID</td>
<td>USR01 - Secure Authentication</td>
</tr>
<tr>
<td>Technology Stack</td>
<td>Application, database</td>
</tr>
<tr>
<td>Versions</td>
<td>All versions</td>
</tr>
<tr>
<td>Control Implementation</td>
<td>Configure the authentication mechanism enabled for Oracle EBS users securely so no one can impersonate application users. Best practices for this calls for single sign-on (SSO), strong password policies, and additional factors during the authentication process.</td>
</tr>
<tr>
<td>Checklist/Steps in Order of Priority</td>
<td>The following attributes should be true for the authentication process:</td>
</tr>
<tr>
<td></td>
<td>1. To protect against replay and other common web attack vectors, transport layer security (TLS) must be used to encrypt all web-based hypertext transfer protocol (HTTP) communication, inclusive of authentication.</td>
</tr>
<tr>
<td></td>
<td>2. By default, Oracle EBS end-user passwords are encrypted but not additionally hashed. Passwords must also be hashed due to decryption risks because of the encryption algorithm's availability on the internet. Failure to hash end-user passwords also poses risks to critical database service accounts, such as the 'APPS' user.</td>
</tr>
<tr>
<td></td>
<td>3. Enforce strong password guidelines for all users—including end users, default seeded users, and service account—through the enablement of robust policies via the logon and session profile options.</td>
</tr>
<tr>
<td></td>
<td>4. Oracle Database profiles define password complexity policies. To effectively address risks for EBS database accounts, at least five profiles should manage different risk levels. The listing below is sorted by risk (highest risk to lowest risk):</td>
</tr>
<tr>
<td></td>
<td>• Seeded database accounts created by the installation of the database (use very complex requirements)</td>
</tr>
<tr>
<td></td>
<td>• Seeded EBS accounts created by the installation of EBS (except for the APPS account)</td>
</tr>
</tbody>
</table>
5. Change all default and weak database passwords.
6. Ensure production database accounts are changed when creating non-production databases for test and development.
7. Identify client-created, custom EBS and database accounts with passwords that have never been changed.

8. **Improved Security**
   - For high-privilege generic users not linked to a human resources (HR) record, such as the APPS account or the SYSTEM in the application or the database, passwords should be subject to significantly more stringent complexity requirements.
   - Create a custom password validation function for EBS end users using the oracle.apps.fnd.security.PasswordValidation Java interface.
   - If possible, SSO should be enabled so the user doesn't have to remember a specific password (or set of passwords) for each enterprise application. Since there will always be a subset of user accounts that can bypass single sign-on, checklist items No. 2 and No. 3 are still crucial in an SSO environment.
   - If possible, the authentication process should ask for more than one factor (i.e., the password and a time-based token). This suggestion is vital for high-privileged seeded users such as SYSTEM-accessing production environments. For Oracle (OCI) customers, both Oracle Access Manager (OAM) and Oracle Identity Cloud Service (IDCS) support multi-factor authentication (MFA).
   - For database accounts, use the password verification function “ora12c_strong_verify_function” to force passwords to meet strict United States (U.S) Department of Defense password complexity requirements.
   - Consider using SSO authentication for database support personnel.

**References**
- [https://docs.oracle.com/cd/E26401_01/doc.122/e22952/T156458T659601.htm](https://docs.oracle.com/cd/E26401_01/doc.122/e22952/T156458T659601.htm)
- [https://docs.oracle.com/cd/E18727_01/doc.121/e12843/T156458T465432.htm](https://docs.oracle.com/cd/E18727_01/doc.121/e12843/T156458T465432.htm)
<table>
<thead>
<tr>
<th>Domain</th>
<th>Cloud ERP Users</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control ID</td>
<td>USR02 - User Accounts Management</td>
</tr>
<tr>
<td>Technology Stack</td>
<td>Application, database</td>
</tr>
<tr>
<td>Versions</td>
<td>All</td>
</tr>
<tr>
<td>Control Implementation</td>
<td>Managing EBS user accounts is paramount for ensuring only the appropriate people have access. This control aims to assure that user accounts within the E-Business Suite are appropriately controlled and governed to avoid unauthorized access to business information.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Checklist/Steps in Order of Priority</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Oracle ships seeded user accounts with default passwords. The default passwords for the seeded accounts must never be used.</td>
</tr>
<tr>
<td>2. Disable (end-date) all seeded accounts unless otherwise documented by Oracle. For the EBS modules in-use, carefully consult Oracle’s documentation. Oracle customers must review accounts for all modules (both modules used and unused).</td>
</tr>
<tr>
<td>3. Create carefully designed custom responsibilities (roles) following the “principle of least privilege.” Try not to use seeded Oracle responsibilities created by the installation of the Oracle E-Business Suite. These responsibilities are over-privileged to simplify the implementation of EBS and should not be used afterward.</td>
</tr>
<tr>
<td>4. When copies of the production database are used to create test and development databases, EBS end-user accounts must be end-dated. Open “active” end-user accounts in non-production databases only for specific teams and/or individuals involved in support and development activities.</td>
</tr>
</tbody>
</table>
5. Oracle EBS end-user accounts can be created for employees, human resource candidates (in the hiring process), suppliers, customers, contractors, support infrastructure, and customizations (e.g., generic/service and seeded accounts). Procedures must exist to identify end-dated (closed) accounts that are no longer needed (i.e., ex-employees, customers, and/or suppliers no longer in business with the organization).

6. If the environment is internet deployed using EBS modules such as iSupplier, iSupport, and/or iStore, ensure that only those responsibilities, menus, and forms—security hardened for use on the internet—are exposed. Internal applications and “General Ledger” data must never be internet-accessible.

7. If the environment is internet-deployed using EBS modules such as iSupplier, iSupport, and/or iStore, disable internet-facing web servers for non-production situations. Enable these only for limited periods for specific testing or production support tasks.

8. Lock and expire seeded and default accounts created by installing the database and/or the E-Business Suite. Carefully refer to the respective EBS and database security guides for more information.

9. Disable (lock and expire) unused client-created custom applications, databases, and Linux accounts.

10. Improved Security
    - Take advantage of the CSP's offerings for risk-based security and authentication offerings such as geographic fencing (user's location), the environment's production status (e.g., production, testing, development), and/or temporal rules (day-of-week, time-of-day). Oracle OCI customers can deploy many of these features using the Oracle Identity Cloud Service with EBS.

References
- https://docs.oracle.com/cd/E26401_01/doc.122/e22952/T156458T659601.htm
Role-based access control (RBAC) must be used to manage EBS authorizations, especially for large and complex organizations. Oracle EBS provides RBAC functionality that allows users in the same functional or technical area to have the same authorizations bundled together in profiles assigned to single or composite roles. Roles can be further combined in composite roles. For example, “Employee” or “Employee-USA.”

The EBS needs to be continuously monitored to ensure that users have only the minimal number of appropriate permissions they need for day-to-day operations. This concept is the principle of least privilege.

The chief consideration for control implementation is to screen the change management process of users, authorization, roles, and responsibilities (e.g., when an employee changes position and the employee's authorization needs to be changed). A process should be established when an HR positional change occurs. A new hire, as well as a termination, should be monitored, with appropriate actions taken.

1. Regularly reconcile the “Active Users” report with the “HR Movements Report,” which identifies users who may require authorization modifications.
2. Within EBS, implement “User Manager” (UMX) RBAC roles and monitor workflows for exceptions, violations, and continued usage evidence.
3. Oracle User Manager Proxy User Access Control allows one user to grant another user the right to impersonate them. This feature must be carefully monitored and audited for abuse and appropriate use.

4. Improved Security
   - Monitor for new responsibilities and changes to responsibility menus. Pay close attention regarding access to security-sensitive forms, functions, and menus commonly assigned to highly privileged systems administrators (SYSADMIN) or those falling under the purview of “System Administration” responsibility processes.
   - Within the database, monitor for custom roles being assigned highly privileged standard roles or custom roles. For example, look for a custom database role being assigned the seeded standard role ‘DBA’ or for a custom role being assigned privileges, such as ‘SELECT ANY TABLE.’
   - For clients at OCI subscribing to the database “Enterprise Edition High-Performance” bundle, Oracle Database Vault is provided as a standard feature. Oracle Database Vault offers in-depth reporting and analysis of database roles and database user provisioning and activities.

References
- [https://docs.oracle.com/cd/E18727_01/doc.121/e12843/T156458T185608.htm#sg_rbac](https://docs.oracle.com/cd/E18727_01/doc.121/e12843/T156458T185608.htm#sg_rbac)
- 12.2.x: Manage Proxies- Query To See All Users That Have Set Up Proxy Access (Doc ID 2037701.1)
- [NIST RBAC model](https://docs.oracle.com/cd/E26401_01/doc.122/e22952/T156458T156461.htm)
<table>
<thead>
<tr>
<th>Domain</th>
<th>Cloud ERP Users</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control ID</td>
<td>USR04 - Emergency Access</td>
</tr>
<tr>
<td>Technology Stack</td>
<td>Application, database</td>
</tr>
<tr>
<td>Versions</td>
<td>All</td>
</tr>
</tbody>
</table>

**Control Implementation**

“Emergency Access” is granted to staff upon system failures, errors, or—in the ordinary course of business—personnel's unavailability. Allowing Emergency Access often results in privileges normally prohibited, exposing the entity to additional risk which must be appropriately managed.


2. Configure according to policy:
   - Configure roles: Who can approve and/or do what?
   - Grant access: Who can approve? And for what activities?
   - Monitor access: What gets reported, when, to whom, and how?
   - Terminate access: Who can approve?
   - Report and audit access: What, when, to whom, and how?

**Checklist/Steps in Order of Priority**

1. **Overall**
   - Require a service request (ticket) for all Emergency Access activities. Utilize approval workflows and audit for effectiveness.
   - Identify accounts within the application and supporting technologies as being in-scope for Emergency Access.
   - For accounts identified for emergency use, ensure that long, complex passwords are in use. This step will force the use of a password safe. Confirm the use of both complex passwords and an enterprise password safe.
   - Enable logging at all technologies inclusive of the application and all supporting technologies, especially the database. Stream audit logs to a SIEM for correlation to identify emergency use of identified accounts without service requests and/or a password-safe activity.
• Implement a regular reporting and audit procedure for all Emergency Access activities.

2. Improved Security
• To pull a password in a password safe, a service request (ticket) must be referenced. Ensure the service request is assigned to the same person pulling the password.

3. Application
• Identify EBS application accounts for being in-scope for Emergency Access. Focus on seeded accounts created when installing EBS (e.g., SYSTEM) and any custom responsibilities assigned either wholly or partially copied from the seeded System Administration responsibility. Also, include any seeded or custom users directly assigned the System Administration responsibility.
• End-date (where possible) application accounts identified as in scope for emergency use. Require workflow ticket system approval to use these accounts.
• Enable auditing per best practices, as documented in the EBS Security Guide.
• Monitor application audit logs for in-scope Emergency Access accounts. Alert on login activity.

4. Supporting Technologies
• Identify accounts within the supporting technologies as in-scope for Emergency Access.
• Use a bastion host to access all supporting technologies. Deploy all support, development and maintenance tools, and utilities on the bastion host and physically segment the network to force access solely through the bastion host, or carefully monitor audit logs to identify connections not using the bastion host.
• Utilize an audited workflow for granting Emergency Access to the bastion host.
• Enable logging within Linux, the database, and “WebLogic,” per Oracle’s recommended best practices.
• Monitor audit logs for in-scope Emergency Access accounts. Alert on login activity, especially if not associated with the bastion host.

References
• https://docs.oracle.com/cd/E26401_01/doc.122/e22952/T156458T663771.htm
• https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-123.pdf
The segregation of duties (SOD) control principle supports that no one should control a process from beginning to end. Within any process, no one user should have the ability to perform more than one incompatible task (e.g., transaction approval, accounting, and reconciliation). A combination of these tasks could, for example, allow someone to write off accounts receivable or make unreviewed changes to IT-related programs. An appropriately integrated SOD policy effectively walks the line between task/employee flexibility and security.

Within EBS, responsibilities are used to define authorization (what users have access to do). Oracle delivers Oracle EBS responsibilities through the installation of its suites. Most customers deploy custom responsibilities unique to business-specific processes.

### Checklist/Steps in Order of Priority

1. Audit custom users (users not created or maintained by Oracle Corporation) directly assigned the System Administration responsibility for appropriateness.
2. Audit any custom users (users not created or maintained by Oracle Corporation) with responsibilities whose menus are either wholly or partially copied from the seeded System Administration responsibility for appropriateness.
3. Carefully audit which users can access forms where they can (by design) inject code into the application. The code includes structured query language (SQL), HTML, operating system commands, and/or change environment variables. Depending on the version of EBS, there are more than 40 of these forms.

### References

- Document 1334930.1, *Sensitive Administrative Pages in Oracle E-Business Suite*
<table>
<thead>
<tr>
<th>Domain</th>
<th>Cloud ERP Users</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control ID</td>
<td>USR06 - Secure User Provisioning/Deprovisioning</td>
</tr>
<tr>
<td>Technology Stack</td>
<td>Application, database</td>
</tr>
<tr>
<td>Versions</td>
<td>All</td>
</tr>
</tbody>
</table>

### Control Implementation

Provisioning technical and functional users within an EBS application is a crucial security control from an operational perspective. The provisioning process needs to ensure:

1. There is a valid business reason and requirement behind the creation of the user.
2. The user is created in the appropriate system and client (tenant).
3. There is a responsible person associated with the account.
4. User permissions (authorizations) are assigned according to the least-privilege approach.
5. Authentication and password settings are provisioned accordingly and securely.
6. There is a validity period defined, when applicable, regarding the account.

Unmanaged and dormant accounts could be misused to access the system in an unauthorized way. Therefore, user accounts’ deprovisioning should be managed appropriately.

It is highly recommended to deploy the EBS as a subscribing application to SSO or a federated identity management solution. Such solutions will ensure that these designations are recognized across all systems when end users are provisioned (or deprovisioned).

### Checklist/Steps in Order of Priority

1. Ensure the provisioning and deprovisioning of users is appropriately managed and documented in company policy.
2. Review application end-user accounts (FND_LOGINS) to identify potential stale users. Different criteria will be needed for employees, contractors (contingent workers), suppliers, and/or customers. Pay attention to user accounts that are end-dated accounts when doing the analysis.

3. If using Oracle Human Resources, ensure that each end-user account is only associated with one human resources record. Also, ensure that terminated employees and consultants (contingent workers) no longer have open accounts.

4. Automate authorization provisioning tasks by deploying EBS User Management (UMX) RBAC roles and inheritance(s).

5. For customer-created custom database accounts, review the last login date (DBA_USERS.LAST_LOGIN). Lock accounts identified as stale and/or no longer needed.

References

- https://docs.oracle.com/cd/E26401_01/doc.122/e22952/T156458T185608.htm
- https://docs.oracle.com/database/121/REFRN/GUID-309FC-CB2-2E8D-4371-9FC5-7F3810E2A8C0.htm#REFRN23302
<table>
<thead>
<tr>
<th>Domain</th>
<th>Cloud ERP Users</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control ID</td>
<td>USR07 - ERP Accounts Security</td>
</tr>
<tr>
<td>Technology Stack</td>
<td>Application, database</td>
</tr>
<tr>
<td>Versions</td>
<td>All</td>
</tr>
<tr>
<td>Control Implementation</td>
<td>The security of ERP accounts is paramount, especially as ERP applications migrate to the cloud. Successfully develop login processes to make it more difficult for adversaries to enter the ERP system using valid (but compromised) credentials. Additionally, the ERP application should implement session management mechanisms according to the applicable standards. Organizations using cloud ERP applications should be sure to:</td>
</tr>
<tr>
<td>1.</td>
<td>Enable multi-factor authentication.</td>
</tr>
<tr>
<td>2.</td>
<td>Enable the passwords to rule, prioritizing password length rather than complexity.</td>
</tr>
<tr>
<td>3.</td>
<td>Analyze user login behavior to detect unfamiliar login locations, times, etc. Also, link to other user activities (e.g., data provided through cloud access and security broker providers, as well as conditional access).</td>
</tr>
<tr>
<td>4.</td>
<td>Implement SSO property.</td>
</tr>
<tr>
<td>5.</td>
<td>Limit access to ERP systems from specific networks.</td>
</tr>
<tr>
<td>6.</td>
<td>Ensure session tokens are dynamic, sufficiently random, encrypted, and expire on time.</td>
</tr>
<tr>
<td>7.</td>
<td>Enable audit logging of the user activities and transactions (“read-only” to all, including administrators).</td>
</tr>
<tr>
<td>Checklist/Steps in Order of Priority</td>
<td>1. Enable encryption between clients and servers to protect user credentials.</td>
</tr>
<tr>
<td></td>
<td>2. Enable multi-factor authentication for users.</td>
</tr>
<tr>
<td>3.</td>
<td>Where multi-factor authentication is not supported, accounts must use passwords unique to that system.</td>
</tr>
<tr>
<td>4.</td>
<td>Enable client certificate authentication for applications.</td>
</tr>
</tbody>
</table>
5. Limit access to ERP systems by network segmentation between clients and servers.

6. Limit access to ERP from specific networks. Consider limiting access to ERP systems by implementing a software-defined perimeter.

7. Use wired networks for high-security tasks and critical clients, such as IT operations management networks or back offices.

8. Enable security audit logging for the application and supporting technologies.

9. Send audit logs to a centralized log server (a server not accessible by EBS administrators, DBAs, and/or users).

10. Monitor the centralized log server using a SIEM product.

References

- CIS 20 Critical Security Controls
- NCSC Cloud Security Guidance
The cloud’s impact is transformative on many levels, as the cloud offers many benefits while also creating new challenges. The impact of moving the full EBS landscape (all environments) or partially moving EBS landscapes (e.g., just development) to the cloud requires organizations to reassess their formal governance policies. This assessment focuses on safeguarding production data and overall business operations, including testing and supporting the production environment.

Oracle EBS cloud migrations create challenges for formal governance policies because of:

1. New technologies employed by cloud providers
2. Changes to existing infrastructure and technologies
3. Interactions with new vendors and internal teams
4. New changes to IT processes, controls, and policies
5. Impacts on legal and compliance processes and reporting
6. Data provenance and sovereignty issues and challenges
7. New or updated versions of EBS (if upgraded as part of the cloud migration)

<table>
<thead>
<tr>
<th>Domain</th>
<th>Cloud ERP Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control ID</td>
<td>APP01 - Secure Landscape</td>
</tr>
<tr>
<td>Technology Stack</td>
<td>Application and supporting technologies</td>
</tr>
<tr>
<td>Versions</td>
<td>All</td>
</tr>
</tbody>
</table>
| Control Implementation | The cloud’s impact is transformative on many levels, as the cloud offers many benefits while also creating new challenges. The impact of moving the full EBS landscape (all environments) or partially moving EBS landscapes (e.g., just development) to the cloud requires organizations to reassess their formal governance policies. This assessment focuses on safeguarding production data and overall business operations, including testing and supporting the production environment. Oracle EBS cloud migrations create challenges for formal governance policies because of:
  1. New technologies employed by cloud providers
  2. Changes to existing infrastructure and technologies
  3. Interactions with new vendors and internal teams
  4. New changes to IT processes, controls, and policies
  5. Impacts on legal and compliance processes and reporting
  6. Data provenance and sovereignty issues and challenges
  7. New or updated versions of EBS (if upgraded as part of the cloud migration) |
| Checklist/Steps in Order of Priority | 1. Inventory all in-scope governance policies and controls for the Oracle EBS.
  2. Determine ownership (team) of each policy. Each control requires an owner responsible for the effectiveness of the control and to be assigned any issues and/or questions.
  3. Assess the effectiveness of the current policies. |
<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>4.</td>
<td>Identify changes required for the cloud. For example, what impact does the CSP’s data centers’ physical location have on data sovereignty requirements for personally identifiable information (PII)?</td>
</tr>
<tr>
<td>5.</td>
<td>Set up a regular review and update process to adapt to new compliance requirements.</td>
</tr>
<tr>
<td>6.</td>
<td>Implement automated controls where applicable.</td>
</tr>
</tbody>
</table>

**References**

- CSA [Top 20 Critical Controls for Cloud ERP Customers](#)
Oracle EBS environments are complex, as the EBS is built using several vital technologies that closely interact. These components are extensively customizable and configurable. While the extensibility of the EBS is a compelling advantage, a significant percentage of the possible number of configuration options have a material security impact. Therefore, a best practice security process should consist of:

1. Define baselines for all components and technologies
2. Enforce the baselines
3. Automate baseline enforcement and monitoring where applicable
4. Setup review and update procedures

1. **Application**
   - Change all default passwords and end-date seeded accounts, per the *EBS Security Guide*.
   - Apply all application security-critical patch updates (CPUs).
   - Ensure the EBS Security Console is clean of all security scan findings. Refer to the *EBS Security Guide* for more information on the Security Console.
   - Deploy all EBS client/server tools and utilities, most notably the "Application Desktop Integrator" (ADI) Excel plugin on the "Virtual Desktop" instead of individual laptops. General Ledger users commonly and heavily utilize the ADI to post journals and run financial reports. The Virtual Desktop should be referred to as a jump box or bastion host.
   - Activate “Server Security” and “Valid Node Checking” to restrict physical access to EBS servers and services.
• Configure the concurrent manager to “start/stop” without the APPS password.
• Enable auditing per recommended best practices in the EBS Security Guide. Forward audit logs to a centralized log management solution.
• Additional EBS security configurations that must be set include:
  • Profile Option: “Security: Allowed Resources” set to “CONFIG”
  • Profile Option: “Allow Unrestricted Redirects” set to “NO”
  • Restricting end user from free-form page queries (FORMS_RESTRICT_ENTER_QUERY set to TRUE)
  • Set a file size upload limit for attached files
  • Whitelist what types of files may be uploaded (attached)
  • Carefully review and set the HTTP security header configuration settings for X-Frame-Options: SAMEORIGIN, X-Content-Type-Options: nosniff, HTTP Strict-Transport-Security (HSTS), Cookie Domain Scoping, secure Cookie Attribute, httpOnly Cookie Attribute, samesite Cookie Attribute

2. Application Tier
• Restrict physical access within the CSP. Use network firewalls and segmentation to allow only HTTP connections from internal to organizational domains or from demilitarized zone (DMZ) domains built to support internet EBS modules (such as iSupplier or iStore). Allow access from support personnel only from a dedicated bastion host using secure shell (SSH).
• Change all default WebLogic passwords.
• Apply all WebLogic security patches (CPUs).
• Apply all “Fusion Middleware” security patches (CPUs).
• Restrict and protect access to EBS admin pages by configuring trusted.conf.
• WebLogic: Only allow access to Oracle WebLogic server administration ports from trusted hosts.
• WebLogic: Only allow direct access to Oracle WebLogic Server from trusted hosts.
• WebLogic: Disable Web Services Atomic Transactions.
• Sign custom Java code with a certificate from a certificate authority.
3. Database

- Restrict physical access to database servers within the CSP. Only allow SSH and transparent network substrate (TNS) access from EBS servers, the bastion host Virtual Desktop personally used by support, and/or dedicated interface servers. Enforce by “Valid Node” checking and/or by network firewall policies.
- Change all default and weak database passwords.
- Apply all database security patches (CPUs).
- Lock and expire seeded and default accounts created by installing the database and/or the EBS. Carefully refer to the respective EBS and database security guides for more information.
- Oracle Database profiles define password complexity policies. To effectively address risk for EBS database accounts, at least five profiles should be used to manage different risk levels. The listing below is sorted highest to lowest risk:
  - Seeded database accounts created by the installation of the database; use very complex requirements
  - Seeded EBS accounts created by the installation of EBS except for the APPS account
  - A profile specifically for the EBS APPS account
  - Accounts supporting customizations and interfaces
  - Accounts assigned to humans and EBS support personnel
- Disable XDB.
- Set and enforce access control lists (ACLs) within the database to limit standard Oracle RDBM functional to the appropriate organizational network domains.
- Enable auditing per EBS security best practices, as documented in the EBS Security Guide. Stream audit logs to a centralized log management solution.
- Harden external procedures (EXTPROC) services.

4. Improved Security

- Scan the database using Oracle's free database security assessment tool (DBStat).
- Scan and measure the overall database security using the Center for Internet Security (CIS) Oracle Database Security Benchmark (e.g., CIS Oracle 12c Database Benchmark). DBStat has a function to scan for both CIS and U.S. Department of Defense security technical implementation guide (STIG) Oracle Database benchmarks.
5. **Operating System**

- Restrict physical access to the servers. For support personnel, only allow SSH from the bastion host Virtual Desktop.
- Apply all Linux security patches (CPUs).
- Provision dedicated accounts. Do not share accounts among support personnel.
- Utilize single sign-on (SSO) to authenticate support personnel.
- Only allow SSH, remove Telnet, remote shell (RSH), and file transfer protocol (FTP).
- Follow **CIS Benchmark** recommendations for Oracle Relational Database Management System (RDBMS) Linux recommendations for file and group permissions.
- Limit root access to need to know.
- Set network time protocol (NTP) per organizational standards.
- Enable auditing and stream audit logs to a centralized log management solution.

6. **Network**

- Physically isolate the EBS environment(s). Segment and enclave database servers from WebLogic servers and production from non-production servers. Force support personnel to use a bastion host to access all servers and services.
  - Create separate network subnets/virtual cloud networks (VCNs) for each environment/instance, the bastion host(s), databases, application servers, and load balancer(s).
  - Database servers must always be placed in a private subnet.

7. **Improved Security**

- Employ web application firewalls (WAFs) to filter traffic, especially if internet is deployed with modules such as iSupport, iSupplier, and/or with web services with trading partners. Refer to the Oracle documentation for how to enable the open-source “ModSecurity WAF.” Most CSPs also provide options for WAFs.
- Take full advantage of CSP advanced network security features such as blocking and alerting based on threat intelligence for “bad IP addresses.”
- Consider the production system enumerating all hosts that are part of the Oracle E-Business Suite instance in the host’s file on each system; this reduces the dependency on the domain name system (DNS) for the core system.
• https://docs.oracle.com/cd/E26401_01/doc.122/e22952/toc.htm
• https://www.oracle.com/database/technologies/security/db-sat.html
• https://www.cisecurity.org/cis-benchmarks/
<table>
<thead>
<tr>
<th>Domain</th>
<th>Cloud ERP Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control ID</td>
<td>APP03 - Implementation of Security Patches</td>
</tr>
<tr>
<td>Technology Stack</td>
<td>Application, Fusion Middleware, WebLogic, Database, Java, Linux</td>
</tr>
<tr>
<td>Versions</td>
<td>All</td>
</tr>
<tr>
<td>Control Implementation</td>
<td></td>
</tr>
</tbody>
</table>

Oracle provides patches to fix security vulnerabilities in its code through its critical patch update (CPU) program. A valid support contract is required to download the patches. The patches are released quarterly on the Tuesday closest to the 17th day of January, April, July, and October. Also, Oracle will release security alerts for issues that are too critical to wait for the next CPU release date.

Oracle EBS Cloud is always operated as an IaaS scenario; CSPs will, in general, not apply security patches to the EBS or its supporting technologies. Therefore, EBS customers must establish a security patch management process.

Utilizing the latest security patches is critical to securing the EBS, and successful deployment of security patches requires an understanding of the following concepts.

1. CPUs are cumulative bundles: The entire bundle is applied. Individual vulnerabilities (e.g., common vulnerabilities and exposures (CVEs)) cannot be separately patched. The latest security patches are engaged with all previous missing security patches by applying the most recent CPU. If CPUs are not utilized for a quarter or two, catching up is as easy as employing the latest CPU.
2. CPUs are mutually exclusive: CPUs are separately released for each Oracle product, application, and/or technology. The application must be patched with each supporting technology to secure the EBS. Applying the latest CPU to the EBS does not patch the supporting technologies, nor does engaging CPUs for any supporting technologies patch the EBS application. For example, applying a CPU to the database does not patch the EBS.
<table>
<thead>
<tr>
<th>Checklist/Steps in Order of Priority</th>
<th>References</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. A continuous vulnerability management process must be set up for CPU bundles and security alerts.</td>
<td>• <a href="https://www.oracle.com/security-alerts/">https://www.oracle.com/security-alerts/</a></td>
</tr>
<tr>
<td>2. All CPU and security alerts must be evaluated and applied promptly to the systems.</td>
<td>• <a href="https://www.cvedetails.com/vendor/93/Oracle.html">https://www.cvedetails.com/vendor/93/Oracle.html</a></td>
</tr>
<tr>
<td>3. Ensure the latest CPU has been applied to the EBS.</td>
<td>• <a href="https://docs.oracle.com/middleware/1212/webtier/HSADM/config_mod_sec.htm#HSADM1163">https://docs.oracle.com/middleware/1212/webtier/HSADM/config_mod_sec.htm#HSADM1163</a></td>
</tr>
<tr>
<td>4. Ensure the latest CPU has been applied to the database. Depending on the version, the Oracle Java Virtual Machine (OJVM) within the database may need to be separately patched.</td>
<td>• <a href="https://blogs.oracle.com/cloudsecurity/enhancing-ebs-security-in-oracle-cloud-part-3">https://blogs.oracle.com/cloudsecurity/enhancing-ebs-security-in-oracle-cloud-part-3</a></td>
</tr>
<tr>
<td>5. Ensure the latest CPU has been applied to the three separate “Fusion Middleware” installations.</td>
<td></td>
</tr>
<tr>
<td>6. Ensure the latest CPU has been applied to WebLogic.</td>
<td></td>
</tr>
<tr>
<td>7. Ensure the latest CPU has been applied to the server operating systems (Linux).</td>
<td></td>
</tr>
<tr>
<td>8. Review the last Java version applied. Both the EBS and its supporting technologies make extensive use of Java. The EBS can quickly deploy more than five different (and possibly more) separate Java installations depending on the version.</td>
<td></td>
</tr>
</tbody>
</table>

3. Cloud service providers (CSPs) do not automatically utilize CPUs: Ensuring that the latest security patches are applied is the customer's sole responsibility. If assistance is required, EBS patch management services may be available directly or through a third party (depending on the CSP).
### Domain
Cloud ERP Application

### Control ID
APP04 - Secure Communications

### Technology Stack
Network, WebLogic (Apache Web Server), Database, Linux

### Versions
All

### Control Implementation
In an EBS environment, there are multiple ways to communicate with the application, inclusive of internet browsers (HTTP), mobile applications (iPhone), desktop/Excel (application development and integration (ADI)), web services, and direct database connections. All EBS-related communications must be secured.

The following are the most critical concepts to highlight for securing communications:

1. Web application security (HTTP(s)/TLS/Secure Sockets Layer (SSL))
2. Database communication (encryption)
3. Linux (SSH)

### Checklist/Steps in Order of Priority

#### 1. Application
- Encrypt end-to-end network traffic, including internal and external communications.
  - Use an internet protocol security (IPSec) virtual private network (VPN) or CSP-specific infrastructure options.
  - Encrypt all HTTP traffic end-to-end for the application, secondary interfaces (e.g., mobile/iPhone), and all supporting utilities.
  - Use TLS and avoid weak TLS/SSL ciphers and protocols. The EBS is currently certified for TLS 1.2, 1.1, and 1.0. Avoid critical sizes of less than 128-bit and any Rivest Cipher 4 (RC4)-based ciphers.
  - Deploy a load balancer or reverse-proxy to offload SSL encryption/decryption.
  - If using internet-facing EBS applications such as iSupplier, iStore, or iSupport, use an external web server. It is highly recommended to use more robust SSL security for these external web servers.
- Manage SSL certificates within a centrally-managed strong vault.

2. Improved Security
   - Enable HTTP strict transport security (HSTS) to instruct web browsers to use only secure connections.

3. Supporting Technologies
   - Physically isolate all servers on the network (enclave) deploying the EBS’s supporting technologies. Only allow support personnel direct access to the servers via a bastion host. Bastion host access should use two-factor MFA, and communications must be encrypted.
   - Use SSH on all EBS servers to access the bastion host. Secure shell encrypts all traffic between clients and servers.
   - Support personnel should not locally store SSH keys in clear text on the bastion host and/or their laptops (especially keys without passphrases). Use an SSH key management solution, such as a password safe and/or SSH-agent.
   - For all database connections, encrypt the TNS protocol. Set the SQL*Net configuration encryption option to “REQUIRED.” This will force the encryption of all database connections, including WebLogic web server(s), developers, DBAs, or any person, team, or tool seeking direct database access.
   - Oracle Database networking supports the Federal Information Processing Standard (FIPS) encryption algorithm, the Advanced Encryption Standard (AES). It is recommended to use AES 256 bit or higher encryption with an integrity checksum configuration of “REQUIRED SHA1.” Also, remember to set the “SQLNET.CRYPTO_SEED” to be a random 70-character string.

References
- Enabling SSL or TLS in Oracle E-Business Suite Release 12.2 (Doc ID 2143101.1)
- https://docs.oracle.com/database/121/DBSEG/asoconfg.htm#DBSEG9599
- NIST SP 800-53 Revision 4: CM-6 b
<table>
<thead>
<tr>
<th>Domain</th>
<th>Cloud ERP Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control ID</td>
<td>APP05 - Change Management Controls</td>
</tr>
<tr>
<td>Technology Stack</td>
<td>Application, database</td>
</tr>
<tr>
<td>Versions</td>
<td>All</td>
</tr>
</tbody>
</table>
| Control Implementation | Implement the appropriate change management controls across all components of the EBS so no unmanaged changes can be implemented in the production system through unauthorized access. Mechanisms exist to assist in enforcing EBS change management processes, including:  
1. Separation of Duties (SoD) for production  
2. Customizations inventory and source code control |
| Checklist/Steps in Order of Priority | 1. Avoid using the production database for production support troubleshooting and testing. Use a regularly scheduled cloning process to create copies of production for production support. For example, this may include a nightly or weekly copy of production.  
2. Restrict access to production. Developers should not have access to promote customizations to production. Install separation of duties between the DBA and development teams for production code promotion.  
3. Create an inventory of customizations, and store the source code for customizations in a source code control solution. The production database should not be the source code control solution.  
4. Setup a source code control solution separately from the EBS implementation (e.g., GIT). |
5. **Improved Security**

- Restrict access to production and automate configurations and customizations among EBS databases using the generic loader (FNDLOAD) utility (for example, to move setups for customization from test to production). The generic loader downloads data from a database according to a configuration (.lct) file and converts it into a data file (.ldt file). The generic loader can then upload this data to another database using a configuration file.

**References**

- [https://docs.oracle.com/cd/E26401_01/doc.122/e22953/T174296T575361.htm](https://docs.oracle.com/cd/E26401_01/doc.122/e22953/T174296T575361.htm)
<table>
<thead>
<tr>
<th>Domain</th>
<th>Cloud ERP Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control ID</td>
<td>APP06 - Secure Extensions</td>
</tr>
<tr>
<td>Technology Stack</td>
<td>Application, database</td>
</tr>
<tr>
<td>Versions</td>
<td>All</td>
</tr>
<tr>
<td>Control Implementation</td>
<td>Oracle EBS customizations are often referred to as “CEMLIs” (Customizations, Extensions, Modifications, Localizations, and Integrations) or “RICE” (Reports, Interfaces, Customizations, and Extensions) items. Consult the Oracle E-Business Suite Developer’s Guide carefully before considering any customization.</td>
</tr>
</tbody>
</table>
| Checklist/Steps in Order of Priority | 1. Create and enforce a secure configuration, development, and coding guidelines document.  
2. Setup a secure software development lifecycle, including a mandatory review process for all developed artifacts  
3. Register all customizations, per the EBS Developer’s Guide. A custom application must own all customizations.  
4. Carefully manage users who have access to sensitive administration pages that can design inject code into the application. Possible code includes SQL, HTML, operating system commands, and/or change environment variables. Depending on the version of EBS, there are more than 40 of these forms. Tightly control access and usage of these forms.  
5. For Linux shell scripts registered as concurrent requests, mask the APPS password. This process is not done by default and is an additional configuration checkbox.  
6. The core functionality of Oracle Human Resources, Oracle Payroll, and Oracle Advanced Benefits all depend on “Fast Formulas.” Fast Formulas is a scripting language. Unfortunately, the menus to enter and/or modify Fast Formulas are not widely distributed in the implementation menus and should be tightly restricted. Likewise, secure all Fast Formulas as structured code in a source code control solution. |
7. Never grant database customizations using database links to “PUBLIC.” Grant all custom database links to an explicit user.
8. Database customizations using database links should never exist between a production database and a non-production database. Disable production database links when copies are made to create test or development databases.
9. “UTL_FILE_DIR” should never be used for EBS interfaces that read or write files to the file system. Use Oracle Database Directories and/or external tables instead.
10. If UTL_FILE_DIR must be used, it should never allow full access to the entire file system. If ULT_FILE_DIR must be used, it should be limited to a specific directory.
11. Never grant custom Oracle Database Directories and/or external tables to “PUBLIC.”
12. Create custom database objects according to the EBS Developer’s Guide to use a naming convention.
13. Never grant custom database objects to the “PUBLIC” unless there is a valid reason.
14. Create custom database accounts according to the principle of least privilege. Avoid granting elevated privileges such as the “DBA” role, “SELECT ANY TABLE,” “READ ALL DATA,” or “EXECUTE ANY PRIVILEGE.”

References

- https://docs.oracle.com/cd/E26401_01/doc.122/e22961/toc.htm
- https://docs.oracle.com/cd/E26401_01/doc.122/e22952/toc.htm
- Document 1334930.1, Sensitive Administrative Pages in Oracle E-Business Suite
<table>
<thead>
<tr>
<th>Domain</th>
<th>Integrations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control ID</td>
<td>INT01 - Secure Integrations and Application Programming Interface (API)</td>
</tr>
<tr>
<td>Technology Stack</td>
<td>Application, Database, Concurrent Manager, Web Services</td>
</tr>
<tr>
<td>Versions</td>
<td>All</td>
</tr>
</tbody>
</table>

**Control Implementation**

The extensive integration of ERP applications (such as EBS) with external applications and data sources is common practice because of the nature of processes supported by these systems.

In a typical EBS environment, there are interfaces and connections between different solutions and different environments. If improperly secured, these integrations are ripe for abuse, and production information and data risks may be easily compromised.

The management of EBS interfaces should address the following considerations:

1. Maintain an inventory of all interfaces, including the type of exchanged data and the technical details of the connections (such as protocol, user, business owner, authorizations, and encryption details).
2. Set up a continuous process for security monitoring of the interfaces.
   - Avoid using insecurely provisioned interfaces, such as broad trust relationships or usernames and passwords that others can leverage. Dedicate one EBS user account (service account) to each interface, per best practice.
   - Always apply the least privilege principle to define the privileges that technical users will be granted for various interfaces.
   - If possible, encrypt all interfaces that exchange regulated or sensitive data between applications. If enabled, alert if encryption is disabled.
   - Avoid setting up interfaces from lower security (such as development) to systems with higher security (such as production) whenever possible.
• If secrets are used to set up the interfaces (i.e., API keys, passwords, certificates), establish the proper management process to govern those secrets (maintain/change/rotate if needed).
• Detect and remove obsolete interfaces and integrations.

Checklist/Steps in Order of Priority

1. Document all interfaces in the centralized information technology infrastructure library (ITIL) configuration management database (CMDB). This location thoroughly documents the EBS and its supporting technologies, components, and servers. Define a unique identifier for each interface. Be sure to include:
   • Concurrent requests
   • Deployed native EBS web services
   • Service-oriented architecture (SOA) suite web services
   • Database links and external tables
   • Data classification (e.g., PII, confidentiality, etc.)
   • Authentication type (password, certificate)
   • Type of interface (file, web service representational state transfer (REST), SOAP)
   • Contact details for internal owner and target (e.g., the bank)

2. Use the security principles by design and security by default; if possible, design for mutual authentication between applications using client certificates.

3. For internet-facing components, follow all Oracle guidance for deploying the EBS or Oracle SOA Suite in a DMZ.
   • Fully configure the URL firewall and create separate DMZ network segments.
   • Secure all external traffic with a web application firewall (WAF).


5. Enforce the principle of least privilege on the technical account used for the integration to reduce consequences in case of a security breach.

6. Perform a penetration test ("pen test") of the published integrations before the “business go-live,” including an initial vulnerability scan. Run a basic network vulnerability scan that supports a common vulnerability scoring system (CVSS) rating. Measure the current situation, and remediate as needed (based on risks).

7. Set up continuous security monitoring of the interfaces.

References

• CIS 20 Critical Security Controls
<table>
<thead>
<tr>
<th>Domain</th>
<th>Cloud ERP Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control ID</td>
<td>DAT01 - Continuous Monitoring</td>
</tr>
<tr>
<td>Technology Stack</td>
<td>Application, HTTP Server, Database</td>
</tr>
<tr>
<td>Versions</td>
<td>All</td>
</tr>
<tr>
<td>Control Implementation</td>
<td></td>
</tr>
<tr>
<td>A complex bundle of multiple technologies supports the Oracle E-Business Suite. Multiple log and audit sources must be enabled and analyzed to understand what is happening within EBS. For the EBS to be effective, these sources must be driven by a continuous monitoring program, which includes an incident response program working within the following framework of directives:</td>
<td></td>
</tr>
<tr>
<td>1. Implement a centralized logging solution.</td>
<td></td>
</tr>
<tr>
<td>2. Enable application logging and auditing, per best practices.</td>
<td></td>
</tr>
<tr>
<td>3. Enable logging and auditing with the supporting technologies, especially the web server and database.</td>
<td></td>
</tr>
<tr>
<td>4. Forward all audit logs to a centralized log and secure server.</td>
<td></td>
</tr>
<tr>
<td>5. Implement a process to review the logs periodically, preferably using a SIEM tool, so a timely response is possible.</td>
<td></td>
</tr>
<tr>
<td>6. Implement an incident response process so when an incident is identified, qualified teams are involved in incident containment.</td>
<td></td>
</tr>
<tr>
<td>Consider physically removing the centralized log server and/or SIEM solution from the CSP hosting EBS when EBS is deployed in the cloud. For example, this migration could be on-premise or at a third-party CSP. This action will help ensure the confidentiality, integrity, and availability of the security log data.</td>
<td></td>
</tr>
<tr>
<td>Checklist/Steps in Order of Priority</td>
<td></td>
</tr>
<tr>
<td>1. Application</td>
<td></td>
</tr>
<tr>
<td>• Unsuccessful logins are tracked by default but only for users authenticated by EBS. EBS will not track unsuccessful logins for users authenticated outside EBS (for example, SSO users). Consolidate and analyze all unsuccessful login attempts for the EBS.</td>
<td></td>
</tr>
</tbody>
</table>
• Sign-on-audit: Ensure that sign-on-audit is set to ‘Forms Level’ to track and log what professional forms users visit. Deliver this log to the centralized log solution.
• Page access tracking (PAT): Enable PAT for forms built with the application framework (AOF) and CRM (JTF) technologies. Enable per EBS Security Guide recommendations. Also, ensure this log is forwarded to the centralized log solution.
• Database connection tagging: Enable the profile option “FND:Connection Tagging (FND_CONNECTION_TAGGING).” EBS end users who trigger audit events can be identified captured in native database auditing and the Oracle Database Vault when database connection tagging is enabled.
• EBS Audit Trail: Enable the EBS Audit Trail per EBS Security Guide recommendations to track a history of all changes to security-sensitive records and tables. Also, ensure this log data is forwarded to the centralized log solution.

2. WebLogic and Fusion Middleware
• By default, WebLogic and Fusion Middleware generate many log files. Set the severity level appropriately to catch both errors and failures. Forward these logs to the centralized log solution. The logs include:
  • Oracle HTTP Server (Apache) logs
  • Oracle Process Manager and Notifications Server logs
  • WebLogic Server server deployment logs

3. Database
• Database alert log: The database alert log is a chronological log of messages and errors that should be monitored for unexpected errors.
• Listener logging: Ensure “TNS Listener Logging” is enabled to collect information about database client connection requests. Forward the log to the centralized log solution.
• Database auditing: Ensure native database auditing is enabled, per recommended best practices in the EBS Security Guide. Forward the native database audit logs via syslog to the centralized log solution. The EBS Security Guide recommended best practices for database auditing address database sessions, system changes, schema changes, schema auditing, and targeted auditing of sensitive EBS objects (e.g., tables defining security and flexfield configurations).

4. Virtual Data Center
• By default, most CSP infrastructure consoles provide extensive logging options. For example, these logs can
alert on the creation of new virtual servers and servers being stopped and/or restarted. The CSP infrastructure logs must be collected and centralized with the application logs.

5. Improved Security

• Fine-grained auditing (FGA): Fine-grained auditing within the Oracle RDBMS allows detailed conditions to trigger auditing. Fine-grained auditing is standard with the EBS database license and defines targeted audit policies within the database. For example, an FGA policy could be created to audit all accesses to the developer tool-generated salary table, but not by the EBS application. Fine-grained auditing is an ideal tool for monitoring sensitive PII data. Fined-grained auditing logs can be streamed via syslog to centralized log solutions.

• Oracle Database Vault (DV): The Oracle DV is a purpose-built solution to protect application data from unauthorized access while meeting continuous monitoring requirements. With DV, controls can be deployed to block privileged account access to application data while logging and controlling sensitive operations inside the database using multi-factor authorization. Database Vault logs can be streamed via syslog to centralized log solutions.

• The DV comes with default, out-of-the-box rules for the EBS, including rules to “blind” DBAs and developers to seeing sensitive PII data.

• For OCI clients subscribing to the database “Enterprise Edition High-Performance” bundle, Oracle DV is a standard feature.

References

• https://docs.oracle.com/cd/E26401_01/doc.122/e22952/T156458T660093.htm
• 12.2 E-Business Suite - Collecting Fusion Middleware Log Files (Doc ID 1362900.1)
• Oracle Applications E-Business Suite 12.2 Fusion Middleware Log Files: Locate, View, and Control (Doc ID 1366187.1)
• https://www.oracle.com/database/vm-cloud-pricing.html
<table>
<thead>
<tr>
<th>Domain</th>
<th>Cloud ERP Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control ID</td>
<td>DAT02 - Data Separation</td>
</tr>
<tr>
<td>Technology Stack</td>
<td>Database</td>
</tr>
<tr>
<td>Versions</td>
<td>All</td>
</tr>
<tr>
<td>Control Implementation</td>
<td>Oracle EBS implementations are usually composed of multiple environments with diverse roles, such as sandbox, development, quality assurance (QA), or production. Overall, environment security is paramount to data security in the production system. Likewise, organizations have fiduciary responsibilities to protect data wherever it exists. In general terms, access to lower-risk environments (such as development) should not place sensitive data at risk; separate production and non-production environments (with different access controls) as much as possible. Unless sensitive information in non-production environments is appropriately secured (e.g., masked, scrambled, and/or redacted), secure these environments with the same levels and standards of protection.</td>
</tr>
</tbody>
</table>
| Checklist/Steps in Order of Priority | 1. Production data must be legally protected wherever it exists. When production data is copied to a non-production system, data must be equally treated and protected as in the production system. For example: if all financial data is copied to a test system, the test system must be treated similarly to the production system.  
2. Physically separate and isolate production servers, databases, and all other supporting technologies from a non-production test, development, and support assets. Enclave production EBS environments into physically separate network segments from non-production environments.  
3. Change all passwords for supporting technologies when creating copies of production for test, development, and support databases. |
4. End-date (close) all active Oracle EBS end-user accounts when creating copies of production for test, development, and support databases. Open only those accounts minimally required for development and support activities.

5. End-date (disable) responsibilities (roles) and/or functional authorization grants for users whose authorizations differ between production and non-production. End-date elevated privileges for test and/or development in production. Likewise, if users have reduced test and/or development privileges, end-date these responsibilities as part of the cloning process.

6. Disable responsibilities (roles) used only in test and development in production. Support and development roles are commonly created with elevated authorizations not allowed in production. End-date test and development responsibilities in production.

7. Sensitive information—including PII data such as national identified numbers (e.g., U.S. Social Security numbers), cell phone numbers, and street addresses—must be secured when creating copies of production for test, development, and support databases. Oracle Corporation (and third parties) offer solutions for securing PII data, and custom solutions can also mask, scramble, or redact PII data.

8. Interfaces between non-production environments and production systems should not exist. Interfaces may be used on a case-by-case evaluation and must meet all compliance requirements (e.g., the encryption and least privilege principle). Point interfaces from production to non-production (push data) and not from non-production to production (pull data). This includes web services (EBS Native and SOA Suite) and database access control lists (ACLs), database links, database external tables, Oracle Database Directories, and o/s file read/write settings for UTL_FIL_DIR.

9. Improved Security

   • Oracle Data Masking Packs: Oracle Data Masking Packs are standard for OCI customers using the data Enterprise Edition High-Performance bundle. The EBS Data Masking Templates are available to de-identify (mask and/or scramble) sensitive PII data stored in standard EBS tables.

   • Oracle Advanced Security Option (ASO) Dynamic Redaction: The ASO with dynamic redaction is provided as a standard feature for OCI clients subscribing to the database Enterprise Edition High-Performance bundle. “Data Redaction” or “Dynamic Data Masking” is the SQL run-time process obfuscating or hiding sensitive data
elements. The Data Redaction feature secures PII data in production and/or non-production databases from support and development personnel. For example, while Social Security numbers are visible in the application if using Data Redaction, a developer directly connecting to the database would only see “#” instead of the actual numbers. With Data Redaction, there is no need to scramble non-production databases.

- Oracle Database Vault (DV): The Oracle Database Vault is a purpose-built solution to protect application data from unauthorized access and to meet continuous monitoring requirements. With DV, deploy controls to block privileged account access to application data and log and control sensitive operations inside the database using multi-factor authorization. Stream DV logs via syslog to centralized log solutions.
- The DV comes with default, out-of-the-box rules for the EBS, including rules to “blind” DBAs and developers to seeing sensitive PII data.
- Oracle DV is a standard feature for OCI clients subscribing to the database Enterprise Edition High-Performance bundle.

References

Business data stored and processed by the Oracle EBS is its most crucial component. According to predefined rules and policies, sensitive data at rest and in-motion must be encrypted and classified to avoid unauthorized access.

Adhere to the following guidelines concerning ERP data.

1. Encrypt data while at-rest and when stored in the database or any other location.
2. Use transparent data encryption (TDE) to encrypt data stored within the database. Transparent data encryption allows only authorized database access either when in-use (mounted) or offline, including backup files.
3. Ensure the proper process is in place to maintain, issue, revoke, and control access to encryption keys, wallets, and certificates for TDE and server images. Use a key vault or hardware security module (HSM). Most CSPs offer a key vault. For example, Oracle OCI has the cloud “Infrastructure Key Management Vault.”
4. Rotate the TDE master encryption keys for production databases regularly (ideally every 90 days or less).
5. Take care to use separate TDE encryption keys for production databases. Production databases must never share encryption keys with non-production databases.
6. Transparent data encryption database encryption is coarse-grained and does not provide fine-grained security. Transparent data encryption only ensures that valid database users can access encrypted data. Fine-grained security within the database is provided through standard database features and/or the Oracle DV (optional license feature).
1. Encrypt all EBS tablespaces using TDE.
2. Confirm key vault or HSM usage to secure the master encryption key.
3. Confirm when the TDE Master encryption key was last changed.
4. Confirm that production and non-production databases do not share a master encryption key.
5. Additional security: Consider full disk encryption and encrypt offline server images, clones, and backups. Most CSPs provide robust key management solutions as part of their basic service.

For Oracle OCI clients, TDE is provided by default (free). For EBS deployments on Azure, AWS, and other cloud providers, TDE should be obtained through an additional ASO license.

The Oracle Cloud Infrastructure Key Management Vault is free for the first twenty (20) key versions per month.

References
- https://docs.oracle.com/cd/E26401_01/doc.122/e22952/T156458T193106.htm
- https://www.oracle.com/cloud/security/pricing.html
<table>
<thead>
<tr>
<th>Domain</th>
<th>Business Processes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control ID</td>
<td>BUS01 - Inventory of Business Assets, Data, and Processes</td>
</tr>
<tr>
<td>Technology Stack</td>
<td>Application and supporting technologies</td>
</tr>
<tr>
<td>Versions</td>
<td>All</td>
</tr>
<tr>
<td>Control Implementation</td>
<td>Operating business applications at scale presents many challenges, including establishing appropriate levels of data visibility and establishing processes to support system applications. Start with a clear inventory of these components to understand where an organization’s “crown jewels” are and provide the right governance and controls around those components. Implement an inventory of applications, data, and processes which serve as a single source of truth regarding business processes.</td>
</tr>
</tbody>
</table>

**Checklist/Steps in Order of Priority**

1. Ensure a single source of truth exists (inventory/repository), such as an “ITIL CMDB.” Make sure that all EBS applications and components within the CMDB are accurate and regularly updated.
2. Identify and thoroughly document all the supporting technologies and components that build up the production and non-production EBS environments. Incorporate critical relationships with other applications such as Advanced Supply Chain, Agile, or Demantra for manufacturing.
3. Define service level agreements for all EBS applications and supporting technologies.
4. Make risk management scorecards available for each EBS environment, including sensitive data protection requirements.

**References**

- Get ISO 27001 reference for ITIL MDB
<table>
<thead>
<tr>
<th>Domain</th>
<th>Business Processes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control ID</td>
<td>BUS02 - Business Process Controls</td>
</tr>
<tr>
<td>Technology Stack</td>
<td>Application, Database</td>
</tr>
<tr>
<td>Versions</td>
<td>All</td>
</tr>
<tr>
<td>Control Implementation</td>
<td>The EBS supports critical business processes. Controls must ensure that no fraudulent activities can be executed by abusing existing or elevated privileges.</td>
</tr>
</tbody>
</table>
| Checklist/Steps in Order of Priority | 1. Identify critical steps in each critical business process—in partnership with business process owners—within the EBS.  
2. For each process, ensure all requirements for the organization's fraud protection program are met.  
3. Implement a solution for automated fraud monitoring of interfaces. |
<table>
<thead>
<tr>
<th>Domain</th>
<th>Business Processes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control ID</td>
<td>BUS03 - Continuous Compliance</td>
</tr>
<tr>
<td>Technology Stack</td>
<td>Application and supporting technologies</td>
</tr>
<tr>
<td>Versions</td>
<td>All</td>
</tr>
<tr>
<td>Control Implementation</td>
<td>It is critical to adhere to regulatory compliance levels related to data, processes, and an organization's industry due to the data and processes that EBS applications support. Implement a process to ensure continuous compliance that functions as a centralized view for monitoring control effectiveness in real time.</td>
</tr>
<tr>
<td>Checklist/Steps in Order of Priority</td>
<td>1. Identify compliance, regulatory standards, and contractual agreements affecting EBS applications. 2. Identify the required specific critical controls. 3. Identify the required testing procedures to validate the operating effectiveness of those specific critical controls. 4. Develop automated testing procedures to validate control effectiveness 24 hours per day, seven days per week. 5. Implement an alerting mechanism to address audit findings when they happen.</td>
</tr>
</tbody>
</table>