

Better R11i Security In 3 Days - Keeping the Bad Guys Away (Part II)

Randy Giefer
Solution Beacon, LLC

Introduction

Terrorism. Identity theft. 9/11. It's no longer a safe world, and corporate data needs to be protected from not only the external bad guys, but also the internal ones as well! Many companies believe this won't happen to them – until it's too late. This paper discusses numerous ways to quickly and easily improve the security of your Release 11i environment and protect your valuable data. With minimal effort required to implement, real world examples of how substantial security benefits can be obtained by performing simple configuration tasks such as modifying the default values of select profile settings and timeout parameters. Security tips and tricks as well as best practices specific to the Oracle Applications will also be provided to enable administrators to protect their data from "the bad guys".

I am often asked, "Why is security so important? Why should I care?" The answer is not the same for everyone, but one answer that applies to some people is, "Because you can go to jail if you neglect security and don't abide by the rules". Yes, the above statement is intended to be an attention-grabber and does not apply to everyone, but the fact of the matter is that we all have a certain amount of responsibility to protect our organization's assets, and corporate or government data is most definitely an organizational asset.

Your job level and function will carry with it a responsibility to protect your organization's data. Executive-level personnel are held responsible for ensuring that data and accounting practices are followed and reported in accordance with numerous regulations (Sarbanes-Oxley has added more stringent reporting and audit controls over recent years), but the responsibility does not end with executive-level management. Applications System Administrators, OS Administrators and Database Administrators also have an obligation to protect the data and system from fraudulent activity and data theft. Among their many responsibilities, it is a fiduciary responsibility of your company's management to prevent fraud. Security violations are an invitation to commit fraud, and IT administrators are responsible for implementing security measures to prevent fraud from occurring.

If a security violation occurs, your management may be considered derelict in their duties and may be held liable. If your company is a publicly held corporation, the Oracle E-Business Suite of Applications are likely to be audited at periodically to ensure that you are following proper procedures to prevent fraud. The Applications System Administrators, OS System Administrators and Database Administrators are responsible for ensuring the security of your company's applications, operating system and database environment. In fact, some companies place Database Administrators and Applications System Administrators on their restricted stock list, which means that there are limitations on when these individuals can buy and sell company stock.

Statistics Support the Need for Security

If you pay attention to the news, it's hard to avoid hearing or reading about security incidents and their impact on our society. From stolen credit card information, to lost/stolen laptops with sensitive corporate or government data, to identity theft, thieves are creating the demand for organizations to better protect their data and systems. With increasing frequency, incidents like an employee selling account information on 92 million AOL accounts to a spammer, or laptops containing sensitive personal Veteran information being stolen, are hitting the newswire. It is apparent that the need for better security is increasing. Consider the following statistics:

- Businesses and financial institutions suffered \$48 billion in losses due to the 9.9 million Americans who had their identities stolen in 2003. (New York Times, October, 2004).

- IT security breaches doubled from 2001 to 2003. (CERT)
- In 2005, 20% of enterprises experienced a serious internet security incident. (Gartner)
- More than 25% of critical data within Fortune 1000 businesses will be inaccurate or incomplete through 2007. (Gartner)

Given that organization data doubles in volume every 18 months (Gartner), the effort required to protect this data also increases.

Internal Threats

Despite most people's fears that hackers will break into the company and destroy data or steal critical information, more often than not, security breaches come from the inside. Gartner estimates that more than 70% of unauthorized access to information systems is committed by employees, and that more than 95% of intrusions result in significant financial losses. The FBI also reports rampant insider hacking, which accounts for 60% to 80% of corporate computer crimes.

Statistics are what you make them to be. Even if the following statistics are exaggerated or off by a few degrees of accuracy, what is left presents a compelling enough reason to recognize that better security is vital.

- 80% of security breaches are done by insiders. (2003 CSI/FBI Survey)
- In 2005, 60% of security breach incident costs incurred by businesses will be financially or politically motivated. (Gartner)

Here are some quotes from other experts in the industry relating to security:

"Insider attacks are where most of the money's lost, where most of the vulnerabilities are."

Frank Huerta, Vice President Intrusion-Detection Product Delivery, Symantec

"Technological protection from external threats is indeed important, but human problems cannot be solved with [only] technological solutions."

Eric D. Shaw, Keven G. Ruby, & Jerrold M. Post, Security Awareness Bulletin / RAND

"In the Banking and Finance sector, fraud is typically perpetrated by a non-technical current or former employee. Sabotage, on the other hand, is typically led by a technical disgruntled employee, usually a former employee."

Dawn Cappelli, Carnegie Mellon University / CERT / Software Engineering Institute

Consider the contents of a recent HOMELAND SECURITY & GOVERNMENT SYSTEMS SECURITY newsletter below. It is still painfully obvious that there are many, many security issues still to be dealt with in today's world – largely because we think that we are more safe and secure than what we really are. The "It Won't Happen To Me" attitude is being disproved to more and more individuals and companies every day.

HOMELAND SECURITY & GOVERNMENT SYSTEMS SECURITY Newsletter

Report Indicates FBI Still has Problems with Lost Laptops (12 February 2007)

According to a report from the Justice Department inspector general's office, the FBI has lost 160 laptops in less than four years. At least 10 of the computers held "highly sensitive classified information" one held "personal identifying information on FBI personnel." Seven of the missing computers were assigned to counterintelligence and counterterrorism divisions...

VA Now Says Missing Hard Drive Holds Info. on 1.8 Million (12 February 2007)

The Department of Veterans Affairs (VA) has released additional information about the hard drive that was reported missing from the Birmingham (Ala.) VA Medical Center on January 22. The hard drive, which was used to back up data from an employee's work computer, may contain personally identifiable information of approximately 535,000 VA patients and as many as 1.3 million doctors, both living and deceased...

Indiana State Government Site Security Breach (10 February 2007)

An Indiana state government web site, www.IN.gov, experienced a security breach that exposed 5,600 credit card numbers of individuals and businesses. Normally, stored card information is encrypted or shortened to the last four digits, but in this case, the entire card numbers were stored in unencrypted form...

Parents Puzzled by University Data Breach Notification (9 February 2007)

A number of Radford, Virginia-area parents with young children have received letters from Radford University (RU) telling them their children's Social Security numbers (SSNs) and dates of birth may have been compromised in a security breach at the university's Waldron School of Health and Human Services...

Are you prepared? Do you have the controls in place to prevent becoming a statistic?

Common Security Controls

Often, people interact with security controls the moment they arrive at work. Physical security controls allow employees and authorized visitors access to the building they work in. These facility controls for individual users are usually along the lines of the three G's (Guards, Gates, and Gizmos), with Gizmos being card/badge readers, or even more sophisticated biometric devices like fingerprint, eye, or facial scans.

Other than IT department staff members, most users do not interact with systems-level security controls, other than providing a userid and password to log into the network or application. When securing IT systems, Technology Stack security controls are implemented in many different layers. Some of the more common controls are:

- Network Layer Security (Firewalls, Proxy Servers, Network Intrusion Detection (NIDS))
- Server Security (Antivirus, Host Intrusion Detection (HIDS))
- Database (Auditing)
- Application (Authentication and Authorization)

The problem with the overall majority of these controls is that they are designed to keep the external bad people out of our systems. But the real question that needs to be asked is "Who is keeping out the internal bad people?" The previously mentioned statistics relating to insider threats show what most organizations properly fail to prevent: protecting their systems from the internal Bad Guys. The internal threats are real!

Understanding Application Security

Application security is partly about technology, but mostly about controlling access. The basic starting point for a solid application security strategy begins with user security. If your system is not secure at the most basic user level, millions of dollars of advanced security hardware and software can be wasted if the system allows itself to be easily compromised at the user level. The best approach to Application Security is to use a multi-layered security model that increasingly adds controls over authentication, authorization and audit trail.

Authentication asks “Who are you?” Once the Application has determined you are who you say you are, Authorization asks “What privileges do you have?” Or another way of saying it, “What can you do?” Audit Trail asks “What did you do?” In this model, the accuracy and security of each level is dependant on the accuracy and security of the prior level. The effectiveness of each level is only as good as the prior level. For example, the effectiveness of an audit trail is almost useless if you can’t ensure that individual accounts are used and individuals are who they say they are. If every user (or even a subset of users) uses a shared or common account, what good does an audit trail do?

I recommend that you start with basic user controls and “work your way up”. It does no good to implement other security controls if the “internal bad guys” have easy access to your data by exploiting weak user access controls.

Recommended Ways to Maintain Applications Security

For complete and total Application Security, many, many items and areas need to be secure across your enterprise. Physical security, network security, and database security are all crucial to ensuring secure protection of the application and your data. This paper focuses on bringing attention to just the Application components of the system. This is not to say that physical, network, and database security aren’t important; they are, but for brevity and focus purposes, the content presented here is limited to Applications only.

The following items are best practice recommendations for maintaining security that your company should consider. Many of these recommendations are quick and easy to implement, and provide a high return on investment (i.e. “Big Bang for the Buck”), while others may take a few days to implement.

Note: This paper shows Release 12 screen shots to show what (if any changes) were introduced with Release 12. Unless otherwise noted, all functionality described in this paper is applicable to 11i as well as Release 12.

Best Practice: Follow Oracle’s Best Practices

MetaLink document 189367.1 describes some of the Oracle best practices for securing the E-Business Suite. The first versions of this document lacked breadth and depth, but Oracle has made significant improvements to it, and it is now a very, very good reference tool. I highly recommend reading it and understanding the details it presents. Many additional topics not covered in this paper are addressed in the Oracle Best Practice document such as:

- Advanced Security/Networking Option (ASO/ANO)
- Encrypt Credit Cards
- Auditing

Best Practice: Do Not Allow Shared Accounts

A very common security failure is failing to uniquely identify a user and audit their actions. When a shared or common account is used, it becomes exponentially more difficult with each additional “shared” user to track and audit who is making changes to data. Furthermore, a company’s technical staff, supported by tools to help proactively anticipate and resolve problems, will certainly struggle if they cannot identify which users are having those problems.

One way to reduce account sharing is to enforce “single user login” by disallowing multiple logins under the same username. MetaLink documents 375403.1 “How Can I Restrict Applications Users To Be Signed In Only Once At Any Time” and 270454.1 “How To Limit The Number Of Form User Open And The Number Of Session User” describe how to accomplish this:

When properly patched and configured, the E-Business Suite raises a Workflow event when the same user has multiple, open sessions. A subscription attached to this event may take some action

including closing the old session under the same user name or sending an email notification to the administrator. Patch 2128669 contains an example demonstrating how to write a custom event and/or additional subscriptions. The subscription calls a rule function that updates the ICX_SESSIONS table setting the DISABLED_FLAG='Y' for all other sessions for the user. This renders the other sessions invalid. The next user action returns the browser to a login screen indicating the session is invalid. User names appearing in the subscription's parameter list are excluded from this functionality. This functionality is disabled by default.

Best Practice: Do Not Use Generic Passwords

Another common security failure is the use of generic passwords – especially prevalent during password reset requests from users or on initial account creation. Think about it. If the standard procedure for creating a new account is to use the combination of the first initial of the first name and the entire last name for a userid, and the default password is set to ‘welcome’, how hard would it be for an “internal bad guy” to know the userid and password of a newly hired HR Director?

Note: The FNDLOAD utility can be used to migrate, among other entities, users, from one instance to another. If the user account being migrated does not exist on the target instance, FNDLOAD sets the password for that user to WELCOME, rather than the password used on the source instance. If you plan to migrate users using FNDLOAD, you should develop a procedure to deal with this security issue.

Best Practice: Use New Features Provided By The UMX Module

Oracle introduced a new module in Release 11.5.10 called UMX to help with User Management. It is a comprehensive module that provides a more secure user administration capability while at the same time providing additional features not found in the ultra-basic user administration forms found in the base 11i System Administrator functions.

With Release 11.5.10, Oracle User Management (UMX) implemented a Role Based Access Control (RBAC) model. Role Based Access Control (RBAC) is an ANSI standard that controls user access control. According to MetaLink Note 290525.1, Oracle User Management FAQ:

The RBAC standard supports mapping user access control based on the role that the user plays within the organization rather than upon the user's individual identity. The benefits of implementing RBAC include:

- *Reduced cost of administering user access*
- *Streamlined setup and implementation of security policies*
- *Structured user access control based on users' job functions*

The RBAC model augments the existing access control model in Oracle Applications by providing additional methods to organize your data security policies and existing function security (via roles). Security privileges in Oracle Applications have up to this point been managed on an individual user basis, with different types of privileges assigned to each user directly. For example, someone in a Support Agent position may have had to be assigned multiple responsibilities and several other types of access privileges in order to perform their job.

By leveraging the RBAC model, users will no longer need to be directly assigned the lower level permissions and responsibilities, as these can be implicitly inherited based upon the roles assigned to the user. Roles can now be defined to consolidate responsibilities and other roles through role inheritance, as well as lower level permissions (functions) and data security policies. This is accomplished through a one-time setup, where all the permissions are assigned to the role. In order to make a mass update in a production system a client only needs to change the permissions or role inheritance hierarchies defined for a role, then all of the users assigned to that role will instantly inherit the new permissions.

Suffice it to say, the UMX module with RBAC is a significant change to consider over the older, existing Oracle Applications security model.

What is the difference between a Role and a Responsibility? MetaLink Note 290525.1 states:

Responsibilities can now be considered a special type of role that represents the set of navigation menus contained within an application. Therefore, responsibilities loosely represent an application itself, whereas roles can be used to determine to what parts of that application (and data therein) a user has access. This represents a shift in the definition of a responsibility in Oracle Applications. Previously, a responsibility has been used not only to define the application navigation menus, but also to confer privileges and permissions within that application. Using this definition of responsibility, it was often necessary to create several similar responsibilities in order to effectively carve out data and functional security access for a group of users. This has increased the overall cost of ownership as the number of responsibilities has grown.

Oracle Applications follows the Role Based Access Control (RBAC) Reference Model (ANSI INCITS 359-2004) definition of a role as "a job function within the context of an organization with some associated semantics regarding the authority and responsibility conferred on the user assigned to the role." Roles can now be defined to determine what applications (responsibilities) as well as what data and functions within those applications a user has access to.

In addition to RBAC, UMX also introduces a new concept to the Oracle E-Business Suite called a Registration Process. A Registration process is a method by which an end-user can request varying levels of access to the system based on who they are in their eligibility. What is new or unique, is that the user doesn't request this via a help desk or by filling out a paper request; rather the system accepts the request, and routes the request through a workflow process, gathering and tracking the needed approvals, notifications and verifications along the way. This simplifies the System Administrator's job by providing streamlined flows for account administration and maintenance.

Oracle User Management supports three types of Registration Processes:

- Self-Service Account Requests. If you have ever made an on-line transaction, you probably did a self-registration and created an account to tie the order to. This Registration Process type is very similar in that the system provides a method for persons to request a new user account. This type of registration process also offers identity verification, which confirms the identity of the requester (via an email notification that requires a response) before the registration request is processed. If the recipient does not reply within a predetermined amount of time the request will be automatically rejected.
- Requests for Additional Access. Oracle UMX provides an Access Request Tool that enables existing users to request additional roles. Users can only request the additional roles that have been defined as appropriate based on their current roles.
- Creation by Administrators. While the name of this registration process does not sound all that intriguing, the reason it exists presents a much more interesting discussion point. In UMX, the definition of an Account Administrator is changing from that of one or two select individuals working in IT or on the helpdesk to users in the Business Units. With the concept of delegated administration, the ability to create user(s) can be extended beyond the traditional confines of an organization's IT department into the business, and even beyond the organization to business partners and clients, because each account creation registration process can be made available to select administrators.

Best Practice: Treat All Non-Production Instances With The Security As Production

Some companies go to great lengths to “lock down” and protect their production environment, but end up neglecting to secure their other non-production environments (e.g. DEV, TEST, QA) that hold copies of production data. To guard these copies of production data, you have two choices – either limit users who will access that data to the same abilities that they have in production, or, purge, obfuscate or encrypt sensitive data refreshed from production. The problem, of course, is that in order to develop or test functionality, users may need more privileges on non-production environments than you would normally allow on production.

There is no silver bullet available here to protect your data in your non-production instances. In some cases, purging data can suffice (i.e. for development of a new application extension), in other cases, you may have to develop your own obfuscation routines to change the data once it is refreshed from production. Which approach is best for you can only be determined by an analysis of your data and its associated sensitivity.

Best Practice: Set Application User Signon Profile Values

Your organization likely has IT standards and policies relating to authentication. For instance, your network login may require a password value longer than 5 characters, or your network account may be locked after three failed login attempts. The E-Business suite finally offers some of these same controls (although not until the later versions of 11i and Release 12). These controls are established within the E-Business Suite by setting system profile values.

The following E-Business Suite profile options address the basic User Authentication level in an Applications Security model.

Profile	Default	Recommendation
Signon Password Failure Limit	None	3 (attempts)
Signon Password Hard to Guess	No	Yes
Signon Password Length	5	8 (characters)
Signon Password No Reuse	None	180 (days)
Signon Password Custom	None	See Note Below
Signon Password Case	None ^{*1}	Sensitive

^{*1} In Release 12, the default value for this profile is ‘Sensitive’

- **Signon Password Failure Limit** - By default, there is no account lockout after a failed number of login attempts. This is just asking to be hacked! I recommend setting a failure limit using the Signon Password Failure Limit profile option. Prior to release 11.5.10, you needed to implement an alert (periodic), custom workflow or report to notify security administrators; now the system ‘locks’ the account. In addition, I recommend notifying security administrators of a lockout by monitoring FND_UNSUCCESSFUL_LOGINS and ICX.ICX_FAILURES tables. Both the FND_UNSUCCESSFUL_LOGINS and ICX.ICX_FAILURES tables capture failed login attempts from the Personal Home Page (Self Service/Web Interface), but failed Forms sessions are only logged to FND_UNSUCCESSFUL_LOGINS.
- **Signon Password Hard to Guess** - The Signon Password Hard to Guess profile option sets internal rules for verifying passwords to ensure that they will be "hard to guess." Oracle defines a password as hard-to-guess if it follows these rules:
 - The password contains at least one letter and at least one number
 - The password does not contain repeating characters.
 - The password does not contain the username.

- **Signon Password Length** - Signon Password Length sets the minimum length of an Oracle Applications password value. The default length is 5 and I recommended 8.
- **Signon Password No Reuse** - This profile option is set to the number of days that must pass before a user is allowed to reuse a password.
- **Signon Password Custom** - This profile option is used if you want to define your own password scheme (validated by custom Java code) in a custom Java class. This would be used if you have a more advanced and complex password value requirement that is not supported by the site profiles described in this paper. For example, your password policy could state that the password value must have a numeric value, an uppercase value, and a special character. If this were the case, you would not be able to enforce that password policy with the existing Oracle E-Business Suite profiles, so you would need to create a custom password java class and set the profile value to that class name for Signon Password Custom.
- **Signon Password Case** - This profile option is used to force case sensitivity in user passwords. By default in Release 11i, this profile is not populated and the system action defaults to being 'Insensitive'. This option allows for tighter security, as well as for better integration with Oracle Internet Directory, because it also allows case sensitive passwords. The Define Users form and the Signon form now accept case-sensitive passwords. I recommend setting the Signon Password Case profile value to "Sensitive" at the site level. Setting this profile on an existing system has no affect on existing passwords already stored in the system. The case sensitivity will start to take affect the next time a password value is changed – it is then that the rule is applied.

Note that this profile option was introduced with 11i.ATG_PF_H Rollup 4. There was an earlier attempt in 11i.ATG_PF_H Rollup 3 to implement a similar profile option, 'Set Password Case' with three possible values, 'Insensitive', 'Sensitive', and 'Mixed'. MetaLink Note 337274.1, "About Oracle Applications Technology 11i.ATG_PF_H Rollup 3 describes this deprecated profile. You will note that the profile option name for this profile did not match the naming convention used by the other password profiles. It appears that Oracle has corrected this with MetaLink Note 365228.1, "About Oracle Applications 11i.ATG_PF.H Rollup 4 (RUP 4)" says that "Mixed" is no longer supported. Also note that in Release 12, the default value of this parameter is set to "Sensitive".

Note that by default, Oracle generally does not set these parameters for you. You will need to set up or change the default values to explicitly configure/enable the profile parameters.

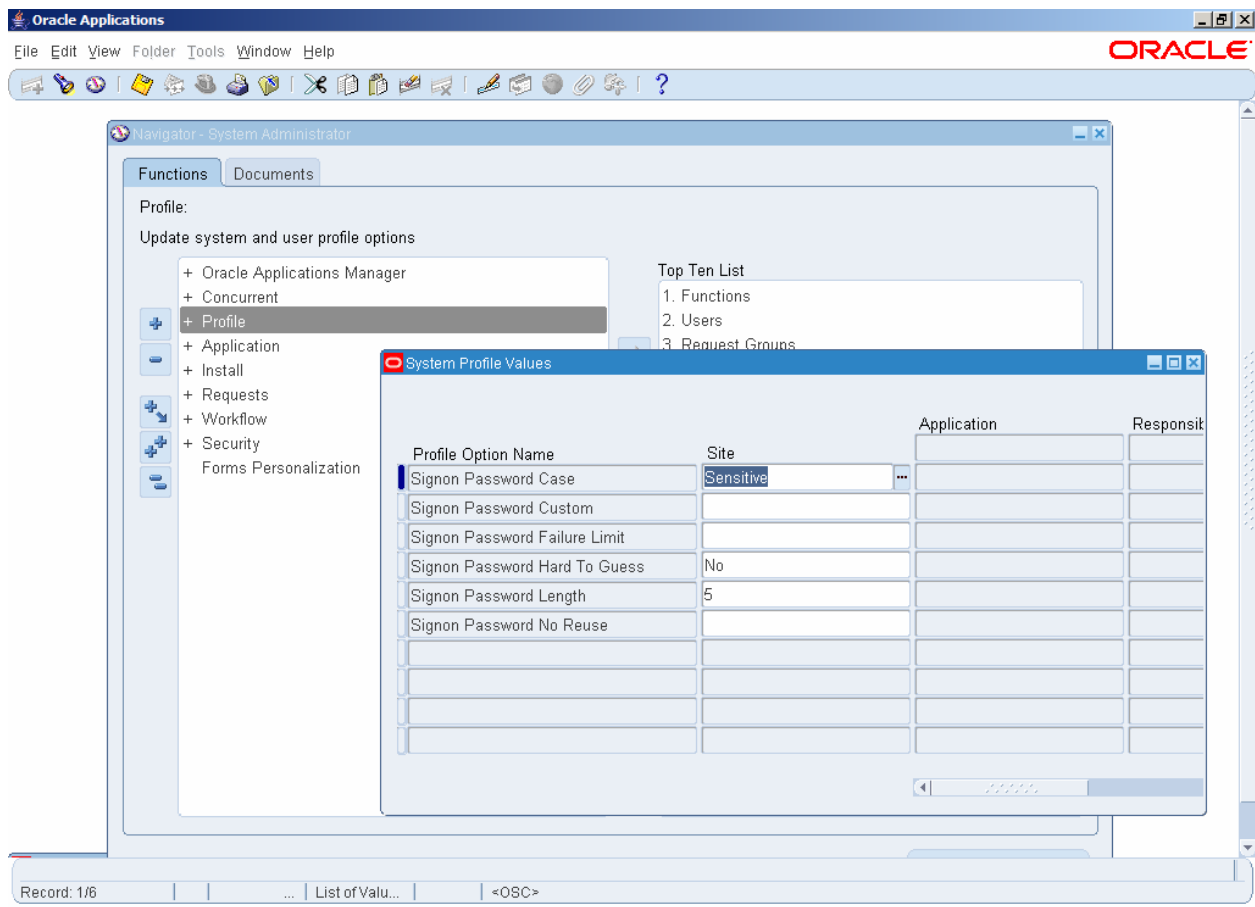


Figure 1 – Release 12 Default Signon System Profile Values

Best Practice: Expire Applications Users' Passwords Periodically

By default, users created within the E-Business Suite do not have their passwords set to expire (see the diagram below). In accordance with your organization IT policies, you should make user passwords expire periodically. As System Administrator, choose Security | Users | Define and set the Password Expiration. In the absence of an organizational IT policy that covers this action, I generally recommend forcing users to change their passwords every 30 or 60 days.

The screenshot shows the 'Users' Define screen. The user 'RGIEFER' is selected. The 'Password Expiration' section has 'None' selected. The 'Effective Dates' section shows 'From 18-FEB-2007'. The 'Direct Responsibilities' tab is active, showing a table with one row for 'System Administrator'.

Responsibility	Application	Description	Security Group	Effective Dates From	Effective Dates To
System Administrator	System Administrator		Standard	18-FEB-2007	

Figure 2 – Release 12 Security | Users | Define screen (Notice that this user's password never expires!)

Since Release 11.5.8, the Security | Users | Define screen has changed to add some additional functionality. Note on this screen the 'Indirect Responsibilities' and 'Securing Attributes' folders that were introduced by the afore-mentioned Oracle User Management (UMX) application module. Indirect responsibilities are used with UMX to allow a user to "inherit" an indirect responsibility through membership in a group to which the responsibility has been assigned. Securing attributes are used by the HTML-based applications to allow only select rows (records) of data to be visible to specified users or responsibilities based on the criteria (attribute values) contained in the row.

Release 12 did not change the Users screen or appear to change any of the associated functionality.

Best Practice: Set E-Business Suite Timeout Parameters and Profiles

An unattended PC without the screen locked poses a security risk. Likewise, an unattended or long running E-Business Suite user session can also pose a risk. The E-Business Suite provides many configuration parameters and profile settings to control user sessions. I recommend reviewing these against your existing corporate policies and setting them according to our recommendations after testing their impact. The following sections describe those items that I recommend setting.

▪ **ICX Timeout Profile Values**

The following E-Business Suite profile options control screen timeouts for Forms, as well as Self Service sessions. **Again, please note, some of the ICX profiles also control Forms Session timeouts!** This can be confusing since Inter-Cartridge Exchange (ICX) is often associated with Self Service applications. This is no longer the case since the release of Framework for the ICX Profiles control the timeout functionality.

Parameter	Default	Recommendation
ICX:Session Timeout	None	30 (minutes)
ICX: Limit Time	4 (hours)	4 (hours)
ICX: Limit Connect	1000	2000

- **ICX:Session Timeout** - This profile option determines the length of time (in minutes) of inactivity in a user's form session before the session is disabled. Note that disabled does not mean terminated or killed. The user is provided the opportunity to re-authenticate and re-enable their timed-out session. If the re-authentication is successful, the disabled session is re-enabled and no work is lost. Otherwise, the session is terminated without saving pending work. This functionality is available via Patch 2012308 (included in 11.5.7, FND.E). Note: Setting the profile value to greater than 30 minutes can drain the JVM resources and cause 'out of memory' errors.
- **ICX: Limit time** - This profile option defines the maximum connection time for a connection – regardless of user activity. If 'ICX:Session Timeout' is set to NULL, then the session will last only as long as 'ICX: Limit Time', regardless of user activity.
- **ICX: Limit connect** - This profile option defines the maximum number of connection requests a user can make in a single session. Note that other EBS internal checks will generate connection requests during a user session, so it is not just user activity that can increment the count.

▪ **CRM Application Timeout Profile Values**

CRM applications use the afore-mentioned ICX timeout profiles (ICX:Session Timeout, ICX: Limit Time, and ICX: Limit Connect), but additionally, CRM also utilizes the JTF_INACTIVE_SESSION_TIMEOUT profile option.

Parameter	Default	Recommendation
JTF_INACTIVE_SESSION_TIMEOUT	None	30 (minutes)

JTF_INACTIVE_SESSION_TIMEOUT - This profile option affects CRM-based products only, and serves the same purpose as the ICX:Session Timeout profile. This profile option exists for legacy reasons, and its value should be set the same as ICX:Session Timeout.

▪ **Jserv (Java) Timeout Settings**

Parameter	Recommendation
disco4iviewer.properties:session.timeout	5400000 (milliseconds)
formservlet.ini:FORMS60_TIMEOUT	55 (minutes)
formservlet.properties:session.timeout	5400000 (milliseconds)
jserv.conf:ApJServVMTimeout	360 (seconds)
mobile.properties:session.timeout	5400000 (milliseconds)
zone.properties:session.timeout	5400000 (milliseconds)
zone.properties:servlet.framework.initArgs	5400000 (milliseconds)

These settings are located at: ../ora/iAS/Apache/Jserv/etc

JServ Timeout is specified by the value of the property session.timeout in the JServ configuration file zone.properties, and represents the number of milliseconds to wait before ending an idle JServ session (the default is 30 minutes). This timeout is used by products based on Oracle Applications Framework (OAF).

- **Apache HTTP Timeout Settings**

The following parameter settings control timeout behavior within Apache.

Parameter	Recommendation
httpd.conf:Timeout	300 (seconds)
httpd.conf:KeepAliveTimeout	15 (seconds)
httpd.conf:SSLSessionCacheTimeout	300 (seconds)

These settings are located: ../ora/iAS/Apache/Apache/conf

- **Forms 60 Environment Timeout Variables**

The following parameter settings control timeout behavior within Oracle Forms.

Parameter	Recommendation
FORMS60_TIMEOUT	55 (minutes)
FORMS60_CATCHTERM	0

You should modify the APPL_TOP/<SID>.env setting to include the following settings:

```
FORMS60_CATCHTERM=0
FORMS60_TIMEOUT=55 (minutes)
```

I recommend using a timeout value of 55 because it is less than the 60 minute value recommended for the web apache timeout values. Note that these values may vary depending on security policies.

- **Oracle Single Sign-On Server Timeouts**

The following parameter setting controls timeout behavior within Oracle Single Sign-On.

‘Single Sign-On Session Duration’ represents the number of hours a user can be logged in to the server without being timed out and having to log in again. This timeout value can be specified from the "Edit SSO Server Configuration" link on the SSO Server Administration page. When a user logs in to Release 11i via the Single Sign-On Server, an SSO login session is created and remains valid for the duration specified by this setting.

Best Practice: Properly Set Other Security-Related Profiles

In addition to Signon and Timeout profile options, there are other security-related E-Business Suite profile options that should also be set:

Profile	Default	Recommendation
Sign-On: Audit Level	(none)	FORM
Sign-on: Notification	No	Yes
Utilities: Diagnostics	No	No
FND: Diagnostics	Yes	No
Hide Diagnostics	No	Yes
Concurrent: Report Access Level	User	User
AuditTrail: Activate	No	Yes

- **Sign-On: Audit Level**
This profile option defines what level (User, Responsibility or Form) the system will perform auditing for user navigation. Note that this 'auditing' content is only information about t what the user navigation was – it does not audit what occurs while a user is on a form, for example.
- **Sign-on: Notification**
This profile option presents notifications to a user upon a successful login.
- **Utilities: Diagnostics**
The Utilities: Diagnostics profile option allows Forms users to examine (and change) individual database records. Setting Utilities: Diagnostics to "No" requires the user to enter the APPS password prior to using the Diagnostics Examine, which should mitigate their ability to change data within the database.
- **FND: Diagnostics**
Self Service has a similar profile option to the one above. If the profile option FND: Diagnostics is set to Yes, then anyone can use the Diagnostics Examine function which allows users to change database records.
- **Hide Diagnostics**
The Hide Diagnostics profile option hides the diagnostics menu from users. The diagnostics menu should be hidden from most users. Set the Hide Diagnostics profile option to Yes, the default value is No.
- **Concurrent: Report Access Level**
Set system profile option Concurrent: Report Access Level to 'User' to restrict a user from viewing other users' concurrent requests.
- **Audit Trail: Activate**
This profile option needs to be set to Yes for the EBS auditing to function.

Best Practice: Restrict Network Access – Set Password on Database Listener

For Oracle database releases prior to the 10g Release, it is very important to set a password for the Oracle TNS listener because any computer can access (and administer) the database listener remotely. For Oracle Database 10g Release 1 (and higher) the default authentication mode is local OS authentication, which requires the account executing the listener command to be a member of the local 'dba' group. It is a best practice to always place a password on the Oracle listener to prevent remote configuration of the Oracle listener (regardless of the version).

Using the `lsnrctl` utility, the `change_password` command is used to set the password for the first time, or to change an existing password.

```
$ lsnrctl
LSNRCTL> change_password
Old password: <hit Enter if no prior password value>
New password: <new password value>
Reenter new password: <new password value>
Connecting to
(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=<host>)(PORT=1521)))
Password changed for LISTENER
The command completed successfully
LSNRCTL>
```

The "Old password:" value should be left blank if the password is being set for the first time. Once the new password is set, the configuration should be saved using the `save_config` command.

```
LSNRCTL> save_config
Connecting to
(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=<host>)(PORT=1521)))
Saved <listener name> configuration parameters.
Listener Parameter File <oracle home>/listener.ora
Old Parameter File <oracle home>/listener.bak
LSNRCTL>
```

Once the password is set, subsequent attempts to perform privileged operations such as `save_config` and `stop` will fail unless the password is set using the `set password` command.

On another note, if the listener you are protecting does not have the default name of `LISTENER`, you must do `set current <listener name>` before issuing the `change_password` command.

Best Practice: Restrict Network Access - Limit Direct Access To The Database

One of the most significant controls that can be placed in an Oracle Applications environment is to not allow direct SQL*Net connections to your database. The underlying premise of this control is about knowing and controlling 'where' the access to the database is coming from, as well as (implicitly) creating a mechanism for identifying 'who' does the access.

Almost all database hardening and database security Best Practices in place today have a control established to identify and restrict direct database connections to the database. In the Oracle world, this is enforced primarily through the use of the `tcp.validnode_checking` and `tcp.invited_nodes` parameters in the `sqlnet.ora` file. Usage of these parameters restricts the IP addresses that can connect to the database listener.

Here is the syntax of the lines that need to be added to sqlnet.ora:

```
tcp.validnode_checking = YES
tcp.invited_nodes = (list of IP addresses)
tcp.excluded_nodes = (list of IP addresses)
```

For example:

```
tcp.validnode_checking = YES
tcp.invited_nodes = (192.168.1.91, visionhost.solutionbeacon.com)
```

Notes:

- You cannot specify a range, wildcard, partial IP or subnet mask (ouch!)
- TCP.INVITED_NODES takes precedence over the TCP.EXCLUDED_NODES if both lists are present (although if a range cannot be specified, you will likely only utilize the invited nodes option).

In a multi-node/server configuration, the E-Business Suite Web Node, Admin Node, Forms Node and Concurrent Processing Node servers would be included in the list of invited nodes, as well as any other administrative or monitoring servers (e.g. Oracle Enterprise Manager).

In both Release 11i and Release 12, you can also enable this capability via the Oracle Applications Manager (OAM) by clicking on the Security tab from the OAM dashboard and clicking on the 'Manage Security Options' on the main Security screen.

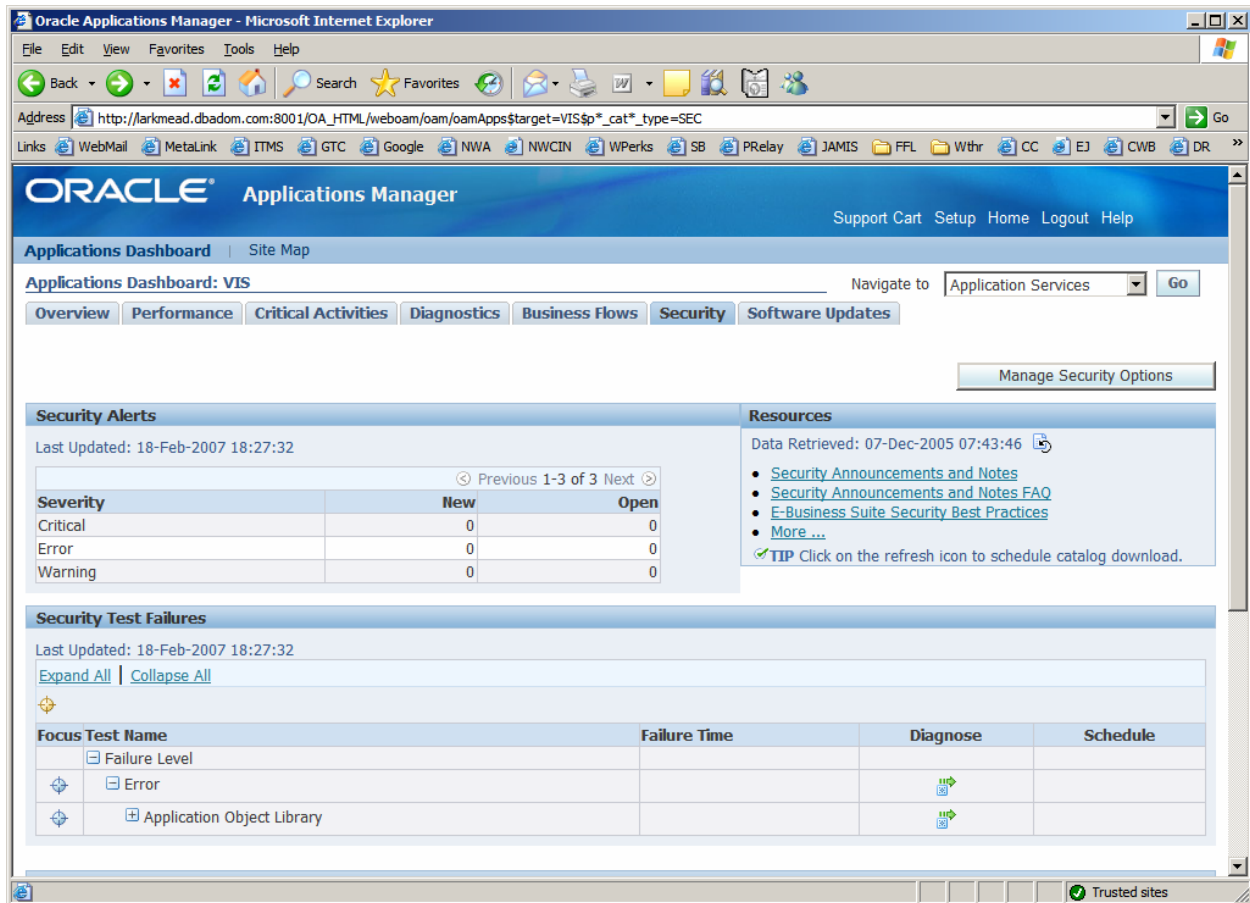


Figure 3 – Release 12 Oracle Applications Manager Security Main Screen

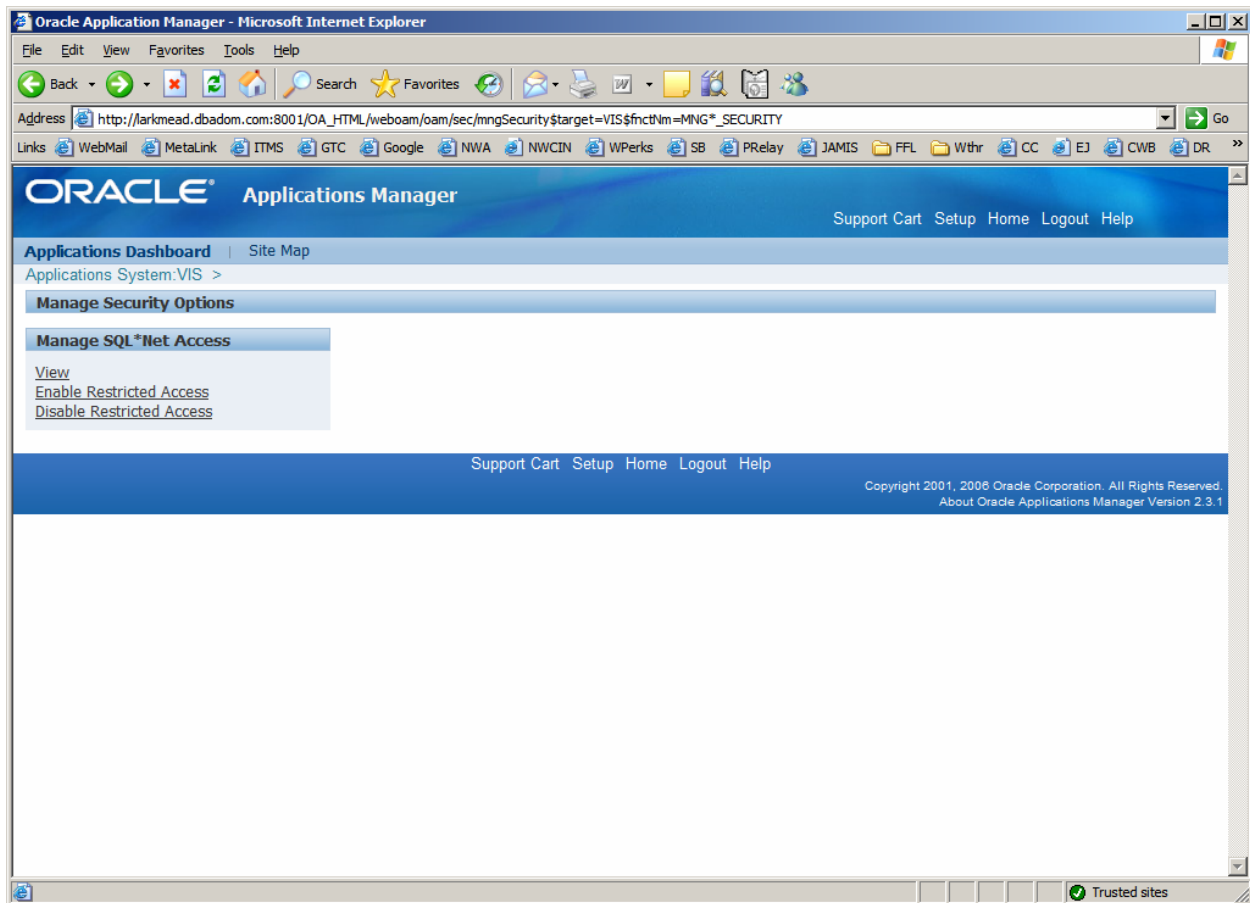


Figure 4 – Release 12 Oracle Applications Manager Manage Security Options

Clicking on the ‘Enable Restricted Access’ link will launch a wizard where you can specify a list of hosts that can access the Oracle Applications Database via SQL*Net. You need to complete the following tasks.

- Run this wizard and respond to the questions
- Run AutoConfig on Database Tier
- Bounce the TNS Listener

Specific to this practice of managing the direct database access, Oracle released a formal recommendation on security for client/server tools directly accessing any E-Business Suite environment. MetaLink Note 277535.1’s pertinent statements are:

Oracle recommends that all components requiring direct connection to the E-Business Suite database are deployed on servers rather than on end user desktop machines. The E-Business Suite architecture mostly supports this requirement natively through a three-tier deployment in which end user browser sessions connect to a middle tier of servers running Oracle 9i Application Server. For the few exception cases in which Oracle E-Business Suite components or associated development tools are not directed through Oracle Application Server, it is recommended that they are deployed in a remote server environment using either Windows Server Terminal Services, Citrix or Tarantella.

If you are asking yourself, “What does this really mean? Does this mean that my developers and/or production support group won’t be able to use SQL tools like TOAD? Can’t we use our PCs anymore to connect to the database?”, then you aren’t alone. For an explanation, let’s start with a security best practice statement:

"Good security controls reduce what can be attacked (the vulnerability surface area), as well as reducing the ways an attack can take place (an attack vector)."

Randy Giefer, 2006

The main point of the Oracle recommendation is to limit the number of direct connections to the database, which in turn can reduce the number of vulnerability surfaces and attack vectors. The basic, underlying premise of this control is “knowing and controlling” *which* servers can have SQL access to the database, as well as (implicitly) creating an additional mechanism for identifying *who* does the access. By placing an intermediate server with software like Citrix running on it in-between the database and the end-user/developer client PC, additional authentication, authorization, and auditing can be performed. In our TOAD example, TOAD would be installed on the intermediate server and that server IP address would be included in the invited node list. Developers, DBAs and others would access the intermediate server from their PC devices.

Note: Be sure to research and adhere to licensing requirements for licensed software products when installed on Terminal Server products.

Note: Specifically related to our TOAD example, the TOAD product has additional server-side security features that can be configured to restrict TOAD user capabilities. It requires the DBA to install the TOAD server objects in the database and perform some additional configurations. It needs to be noted that TOAD Security by itself is not a replacement for the "direct access" control as implementing just TOAD Security would still leave the database listener port exposed to an unmanageable and uncontrollable number of connectivity points. However a much more secure alternative is to implement ‘invited nodes’, TOAD Security, *and* have the TOAD client installed on a hardened server running Windows Terminal Services or Citrix.

Note also that not all security alerts or recommendations from Oracle can be dealt with by simply applying a patch – this recommendation by Oracle to restrict direct database access is a perfect example. As always, you may have to investigate whether an Oracle recommendation fits with your company’s needs, and find your own way to do what it suggests.

The task to protect the database and yet allow authorized individuals appropriate access will require time and research for your unique circumstances.

Best Practice: Follow the Principle of Least Privilege

The principle of least privilege requires that a user be given no more privilege than necessary to perform a job. Ensuring least privilege requires identifying what the user's job is, determining the minimum set of privileges required to perform that job, and restricting the user to only those privileges. Here are some examples:

- Limit the number of users who have the System Administration responsibility. In most cases, there is no justifiable need for the Help Desk, OS System Administrators, or individual users to have this broad responsibility – subsets of the entire System Administration capabilities can be split out by creating custom responsibilities with a subset of System Administration menu paths that will not compromise security. The same principle applies to Application Developer and “Super User” responsibilities.
- Limit OS access to only those who need it and limit their access to only specific directories and files, not the entire APPL_TOP and ORACLE_HOME, for example.

- Limit the users who have the Oracle SYS or SYSTEM password to the minimal number of personnel required. In some larger Database Administration Groups, not all DBAs may need to have these accounts and their inherit privileges.
- Limit users who can access the other Oracle database accounts (GL, PO, etc.) to a small and finite group. A read-only account with APPS-like access can be created for development and production support personnel.

Best Practice: Regularly Change System Passwords

Regularly change the Applications default passwords on the Oracle database (APPS, GL, PO, etc.) at least quarterly. Also change the SYS and SYSTEM Oracle database passwords and the OS Applications password(s) for your *oracle* and *applmgr* accounts quarterly. A best practice is to set every account to have a unique password. However, the reality is that most organizations don't do that, and will set multiple accounts to the same password value for ease of administration. While I don't recommend that as a Security consultant, I do recognize that this 'practice' is used quite frequently. Here are some guidelines if you do:

- Set the SYS, SYSTEM passwords to different password values
- Set the oracle and applmgr OS accounts to different password values
- Set APPLSYS and APPS passwords to be different from the Application Module passwords like INV, AR, AP, etc.

The FNDCPASS program, described in MetaLink Note 159244.1, "How To Use FNDCPASS to Change The Oracle Users, APPS, APPLSYS and Application Module Passwords (INV, AR, AP, etc.) For Applications 11.5 in UNIX", greatly simplifies the password change process. While in earlier 11i versions you originally had to change these passwords one at a time (even with FNDCPASS), Oracle added has added new functionality to this utility with the "ALLORACLE mode" introduced with 11i.ATG_Pf.H.RUP4. The ALLORACLE mode enhances the FNDCPASS functionality so that you can change almost all Oracle schema passwords at the same time, using the syntax:

```
FNDCPASS apps/pass 0 Y system/manager ALLORACLE welcome1
```

There are a few caveats, of course:

1. You should run FNDCPASS from the database tier to avoid any possible problems with database encryption.
2. You should change the APPLSYS password first. It will automatically change the APPS password, as they have to be the same. The syntax to change the APPLSYS password is:

```
FNDCPASS apps/pass 0 Y system/pass SYSTEM APPLSYS <new password>
```

3. If you run FNDCPASS against the other Oracle accounts without first updating APPLSYS, the APPLSYS password will be "undecodable by the Applications". In other words, you'll break the ability to log in via the APPLSYS account. You might recover from such an error by importing the *fn_users* table from an export, or recovering the database from a backup.
4. Schemas that are not changed by the ALLORACLE mode are APPLSYS, APPS, the external account and the public account. Although those passwords aren't changed by the ALLORACLE command, that doesn't mean that you shouldn't change those passwords as well.
5. There are a number of files documented in MetaLink Note 159244.1 that need to have the APPS password manually updated after you've run FNDCPASS to change the APPS password.

Notes Regarding Oracle Database Password Values

Remember that the Applications passwords are set both within the E-Business Suite and the database itself. Database passwords have more constraints and restrictions than the Applications, so you have to be

careful when changing these passwords that you select a password value that is a valid database password. Here are some notes to remember regarding database password values:

- If the database password is not enclosed in quotes, then it can include any letter, any digit, or any of the three following special characters: "_", "#", or "\$". Only a letter can be used in the first character, the other characters can be used after that.
- It is also important to remember that Oracle passwords are not case sensitive (i.e. "a" is the same as "A"), so the effective alphabet is reduced by 26 characters.

Best Practice: Minimize Passwords Contained In Files

To prevent someone gaining access to a file they weren't intended to, it's a good idea to not hard code usernames and passwords in files. This will minimize maintenance issues when you change your Oracle database account passwords. Consider creating concurrent requests for scripts that you might normally run outside of the applications. By running your programs through the concurrent manager, you won't need to include usernames and passwords in your programs. If you must pass usernames and passwords in files, use your operating system's security controls to limit who can read the files and have your scripts read an encrypted password file to get the account password.

Also, be sure to limit the occurrences of sensitive passwords in log files. To prevent adpatch from writing password information in the log files, use the 'flags=hidepw' option on the adpatch call. For example:

```
$ adpatch flags=hidepw
```

In order to make sure that this option is always added to the adpatch command, you may wish to create your own 'adpatch wrapper' script that adds the 'flags=hidepw' option (as well as any others that you may wish) on the adpatch call. Place this wrapper script in a directory and make sure that the directory is included in your PATH statement *before* \$FND_TOP/bin, and you should be set to go next time you want to 'adpatch'.

Best Practice – Secure Default Database Accounts

One of the most common ways to "hack" a database is to utilize a default database account that has the default password. Unfortunately, Oracle provides many, many default accounts (in addition to SYS and SYSTEM) when a database is installed (depending on the installation options). The E-Business Suite adds another 200+ accounts to this default account list.

Product feature accounts (e.g. CTXSYS), as well as other administrative and application accounts all should have the passwords changed immediately upon installation. Of course, these passwords should also be changed on a regular basis.

Demonstration accounts (e.g. QS_xyz), should be dropped (recommended). Other accounts (e.g. system/product accounts) should be locked and expired.

```
alter user OUTLN identified by gr#8w1n3s account lock password expire;
```

The following table shows the database schemas that are shipped with a fresh install of the 11i E-Business suite. The second column defines if the account password should be changed, and the third column defines if FNDCPASS should be used to change the password instead of just changing the password at the database level.

Database Schemas Shipped with E-Business Suite

Schema	Change?	FNDCPASS?	Description
SYS	Y	N	Initial schema in any Oracle database. Owns the data dictionary.
SYSTEM	Y	N	Initial DBA User.
DBSNMP	Y	N	Used for database status monitoring.
SCOTT	Y	N	Demo account delivered with RDBMS.
SSOSDK	Y	N	Single Sign On SDK.
JUNK_PS, MDSYS, ODM_MTR, OLAPSYS, ORDPLUGINS, ORDSYS, OUTLN, OWAPUB	Y	N	Miscellaneous
PORTAL30_DEMO, PORTAL30_PUBLIC, PORTAL30_PS, PORTAL30_SSO_PUBLIC	Y	N	Oracle Portal and Portal Single Sign On, v3.0.9
PORTAL30, PORTAL30_SSO	Y	Y	Oracle Portal and Portal Single Sign On, v3.0.9
CTXSYS	Y	Y	InterMedia schema used by Online Help and CRM service products for indexing knowledge base data.
EDWREP	Y	Y	Embedded Data Warehouse Metadata Repository
ODM	Y	Y	Oracle Data Manager
APPLSYSPUB	N	Y	Initial, pre-authentication user with minimal privileges to assist with APPS (FND) user authentication.
APPLSYS	Y	Y	Contains shared APPS foundation objects. Need to run Autoconfig after changing this password.
APPS	Y	Y	Runtime user for E-Business Suite. Owns all of the applications code. Need to run Autoconfig after changing this password.
APPS_mrc	Y	Y	Optional, additional APPS schemas for the (now obsolete) Multiple Reporting Currencies feature. Defaults to APPS_MRC, but country code suffixes may be used, e.g. APPS_UK, APPS_JP. Need to run Autoconfig after changing this password.
AD_MONITOR	Y	N	Used by Oracle Applications Manager (OAM) to monitor patching.
ABM, AHL, AHM, ... AP, AR...GL, ... ZX	Y	Y	These schemas belong to individual EBS base products. By default the password is the same as the SCHEMA name. Changing the password for these schemas does not affect any configuration files.

The following tables show for each version of the database the default accounts that are possible, and the default status upon installation. Note that these passwords need to be checked regularly, as patches and other DBA actions will often reset them back to their default value! Demonstration accounts (e.g. SCOTT, QS_*), as well as any other unneeded accounts, should be dropped from the database if not utilized.

Oracle 10g (R1 and R2) EE – Default Accounts and Status

Username	Account Status
ANONYMOUS	EXPIRED & LOCKED
CTXSYS	EXPIRED & LOCKED
DBSNMP	EXPIRED & LOCKED
DIP	EXPIRED & LOCKED
DMSYS	EXPIRED & LOCKED
EXFSYS	EXPIRED & LOCKED
HR	EXPIRED & LOCKED
LBACSYS	EXPIRED & LOCKED
MDDATA	EXPIRED & LOCKED
MDSYS	EXPIRED & LOCKED
MGMT_VIEW	EXPIRED & LOCKED
ODM	EXPIRED & LOCKED
ODM_MTR	EXPIRED & LOCKED
OE	EXPIRED & LOCKED
OLAPSYS	EXPIRED & LOCKED
ORDPLUGINS	EXPIRED & LOCKED
ORDSYS	EXPIRED & LOCKED
OUTLN	EXPIRED & LOCKED
PM	EXPIRED & LOCKED
QS	EXPIRED & LOCKED
QS_ADM	EXPIRED & LOCKED
QS_CB	EXPIRED & LOCKED
QS_CBADM	EXPIRED & LOCKED
QS_CS	EXPIRED & LOCKED
QS_ES	EXPIRED & LOCKED
QS_OS	EXPIRED & LOCKED
QS_WS	EXPIRED & LOCKED
RMAN	EXPIRED & LOCKED
SCOTT	EXPIRED & LOCKED
SH	EXPIRED & LOCKED
SI_INFORMTN_SCHEMA	EXPIRED & LOCKED
SYS	OPEN
SYSMAN	EXPIRED & LOCKED
SYSTEM	OPEN
TSMSYS (New in 10g R2)	EXPIRED & LOCKED
WK_TEST	EXPIRED & LOCKED
WKPROXY	EXPIRED & LOCKED
WKSYS	EXPIRED & LOCKED
WMSYS	EXPIRED & LOCKED
XDB	EXPIRED & LOCKED

Oracle 9i R2 EE - Default Accounts and Status

Username	Account Status
ADAMS	EXPIRED & LOCKED
CTXSYS	EXPIRED & LOCKED
DBSNMP	OPEN
HR	EXPIRED & LOCKED
LBACSYS	EXPIRED & LOCKED
MDSYS	EXPIRED & LOCKED
ODM	EXPIRED & LOCKED
ODM_MTR	EXPIRED & LOCKED
ORDPLUGINS	EXPIRED & LOCKED
ORDSYS	EXPIRED & LOCKED
OUTLN	EXPIRED & LOCKED
PM	EXPIRED & LOCKED
QS	EXPIRED & LOCKED
QS_ADM	EXPIRED & LOCKED
QS_CB	EXPIRED & LOCKED
QS_CBADM	EXPIRED & LOCKED
QS_CS	EXPIRED & LOCKED
QS_ES	EXPIRED & LOCKED
QS_OS	EXPIRED & LOCKED
QS_WS	EXPIRED & LOCKED
SCOTT	OPEN
SH	EXPIRED & LOCKED
SYS	OPEN
SYSTEM	OPEN
WKPROXY	EXPIRED & LOCKED
WKSYS	EXPIRED & LOCKED
WMSYS	EXPIRED & LOCKED
XDB	EXPIRED & LOCKED

Oracle 9i R1 EE – Default Accounts and Status

Username	Account Status
ADAMS	EXPIRED & LOCKED
AURORA\$JIS\$UTILITY\$	OPEN
AURORA\$ORB\$UNAUTHENTICATED	OPEN
BLAKE	EXPIRED & LOCKED
CLARK	EXPIRED & LOCKED
CTXSYS	EXPIRED & LOCKED
DBSNMP	OPEN
JONES	EXPIRED & LOCKED
OE	EXPIRED & LOCKED
HR	EXPIRED & LOCKED

LBACSYS	EXPIRED & LOCKED
MDSYS	EXPIRED & LOCKED
OLAPDBA	EXPIRED & LOCKED
OLAPSVR	EXPIRED & LOCKED
OLAPSYS	EXPIRED & LOCKED
ORDPLUGINS	EXPIRED & LOCKED
ORDSYS	EXPIRED & LOCKED
OSE\$HTTP\$ADMIN	OPEN
OUTLN	OPEN
PM	EXPIRED & LOCKED
QS	EXPIRED & LOCKED
QS_ADM	EXPIRED & LOCKED
QS_CB	EXPIRED & LOCKED
QS_CBADM	EXPIRED & LOCKED
QS_CS	EXPIRED & LOCKED
QS_ES	EXPIRED & LOCKED
QS_OS	EXPIRED & LOCKED
QS_WS	EXPIRED & LOCKED
SCOTT	OPEN
SH	EXPIRED & LOCKED
SYS	OPEN
SYSTEM	OPEN

Best Practice: Be Proactive

Proactively monitor security sources like the US Computer Emergency Readiness Team (CERT). CERT is a center of Internet security expertise, located at the Software Engineering Institute, a federally funded research and development center operated by Carnegie Mellon University. They study Internet security vulnerabilities, research long-term changes in networked systems, and develop information and training to help you improve security. You can subscribe to CERT alerts via: <http://www.us-cert.gov/cas/index.html>

Subscribe to Oracle Security Alerts. In order to start receiving Oracle Security alerts by e-mail, you need to follow these steps as outlined below. Also, if you had previously signed up for this alert, you should double check that your electronic subscriptions are up to date.

1. Go to the login page to sign into your Oracle/OTN account
2. Login to your account or create a new account as necessary.
3. On the personal information page, scroll down and find the 'Electronic Subscriptions' section and be sure to check the box next to the Oracle Security Alerts and click 'Continue' to confirm.

Best Practice: Apply all prior, and plan in advance for new Oracle Security Patches

Plan ahead. Plan for Quarterly updates for Security patches and integrate plans to put these Critical Patch Updates (CPUs) into your Release Management and Release Planning process. Here are some important notes from MetaLink Note 360470.1, “Oracle Critical Patch updates and Security Alerts Frequently Asked Questions”:

“In January of 2005, we changed the method and schedule by which we deliver security patch updates and security fixes for all of their products. A Critical Patch Update (CPU) is a collection of patches for multiple security vulnerabilities. It also includes non-security fixes that are

required (because of interdependencies) by those security patches. Oracle provides CPUs for all product offerings on a quarterly schedule. Customers prefer to have a regular, planned schedule for patching their systems. After surveying customers across a variety of industries, we found that a quarterly process strikes a balance between issuing patches so frequently that customers cannot keep up with them, and so infrequently that customers may be exposed to an un-patched and serious security vulnerability.”

Prior to the Critical Patch Update Program, the Oracle Security Alert was the primary means of releasing security fixes for Oracle products. After the introduction of the Critical Patch Update Program, Oracle may occasionally issue a Security Alert in cases where we are releasing an interim (one-off) security patch in advance of a Critical Patch Update. I strongly recommend applying Oracle’s Critical Patch Updates (CPUs) on a quarterly basis, shortly after they are released. Note that extensive testing is often required, as these “updates” often are product upgrades to the Applications technology stack and need to be thoroughly tested. You should also watch for Oracle’s occasional security alerts – if they are released separately from the CPU, then the security issue is likely to be a very serious one. The Oracle Alerts are located at: <http://www.oracle.com/technology/deploy/security/alerts.htm> Keep in mind that if you’ve received notification of a security issue, so have potential hackers!

An important note is that in general, CPU patches for Oracle technology stack products are cumulative – you can apply the most recent patch and you get patches for all of the prior CPUs. However, patches for the E-Business Suite are NOT cumulative, so if you get behind, you have to apply all of the patches from the prior CPUs.

The pertinent MetaLink notes on past CPUs are located below:

Critical Patch Update	MetaLink Note	Latest Version/Date
Critical Patch Update - January 2007	403335.1	Rev 1, 16 Jan 2007
Critical Patch Update - October 2006	391558.1	Rev 3, 20 Nov 2006
Critical Patch Update - July 2006	372927.1	Rev 1, 18 Jul 2006
Critical Patch Update - April 2006	360044.1	Rev 1, 18 Apr 2006
Critical Patch Update - January 2006	343382.1	Rev 1, 17 Jan 2006
Critical Patch Update - October 2005	333953.1	Rev 2, 19 Dec 2005
Critical Patch Update - July 2005	311034.1	Rev 1, 12 Jul 2005
Critical Patch Update - April 2005	301040.1	Rev 2, 13 Apr 2005
Critical Patch Update - January 2005	293953.1	Rev 2, 15 Mar 2005

The CPU patches are released on the Tuesday closest to the 15th day of January, April, July and October. The next four dates are listed below and you should plan these into your release management schedule.

- 17 April 2007
- 17 July 2007
- 16 October 2007
- 15 January 2008

Best Practice: Limit Access To Forms Allowing SQL Entry

Believe it or not, the E-Business Suite actually has numerous forms that allow users to enter SQL statements that get executed directly to the database, which means that no controls are in place to prevent an internal bad guy from abusing this privilege. This is a significant attack vector that needs to be mitigated if at all possible!

The table below shows the Forms that allow users to enter SQL statements, edit code, add code or otherwise affect executable code. Access to these forms should be (in order of preference) eliminated or restricted to a small group of users. If the business justification means that the access cannot be eliminated, then I would strongly suggest that auditing be turned on for those tables.

EBS Forms That Accept SQL Statements

Form Function	Form Name	Table Name
ALR_ALRALERT	ALRALERT	ALR_ALERTS
FND_FNDCPMCP_SYS	FNDCPMCP	FND_CONCURRENT_PROGRAMS
FND_FNDCPMPE	FNDCPMPE	FND_EXECUTABLES
FND_FNDFFMDC	FNDFFMDC	FND_DESCRIPTIVE_FLEXS FND_DESCR_FLEX_CONTEXTS FND_DESCR_FLEX_COLUMN_USAGES
FND_FNDFFMVS	FNDFFMVS	FND_FLEX_VALUE_SETS FND_DESCR_FLEX_COL_USAGE FND_ID_FLEX_SEGMENTS FND_FLEX_VALIDATION_TABLES FND_FLEX_VALIDATION_EVENTS
FND_FNDPOMPO	FNDPOMPO	FND_PROFILE_OPTIONS
FND_FNDSCAPP	FNDSCAPP	FND_APPLICATION
FND_FNDSCDDG	FNDSCDDG	FND_DATA_GROUPS FND_DATA_GROUP_UNITS
FND_FNDSCMOU	FNDSCMOU	FND_ORACLE_USERID
PSB_PSBSTPTY	PSBSTPTY	PSB_ATTRIBUTE_TYPES
MSDCSDFN	MSDCSDFN	MSD_CS_DEFINITIONS
MSDCSDFA	MSDCSDFA	MSD_CS_DEFINITIONS
MSD_MSDAUDIT	MSDAUDIT	MSD_AUDIT_SQL_STATEMENTS
JTFRSDGR	JTFRSDGR	JTF_RS_DYNAMIC_GROUPS_B JTF_RS_DYNAMIC_GROUPS_TL
JTFBRWKB	JTFBRWKB	JTF_BRM_RULES_B
ONT_OEXPCFVT	OEXPCFVT	OE_PC_CONSTRAINTS OE_PC_CONDITIONS OE_PC_ASSIGNMENTS OE_PC_VTMPLTS
ONT_OEXDEFWK, QP_OEXDEFWK	OEXDEFWK	OE_DEF_ATTR_DEF_RULES
JTFTKOBT	JTFTKOBT	JTF_OBJECTS_B JTF_OBJECTS_TL JTF_OBJECT_USAGES
JTF_GRID_ADMIN	JTFGRDMD	JTF_GRID_DATASOURCES_B JTF_GRID_COLS_B
JTFGDIAG	JTFGDIAG	JTF_GRID_DATASOURCES_B JTF_GRID_COLS_B
JTFGANTT	JTFGANTT	JTF_RS_RESOURCE_EXTNS JTF_RS_GROUPS_B JTF_RS_TEAMS_B
QP_QPXPRFOR	QPXPRFOR	QP_PRICE_FORMULAS_B
QP_QPXPTMAP	QPXPTMAP	QP_ATTRIBUTE_SOURCING
GMAWFPCL_F	GMAWFPCL	GMA_PROCDEF_WF

Best Practice: Perform Security Assessments

Security Assessments differ from Audits, as these assessments are designed to help the organization to improve overall security by not only identifying vulnerabilities, but also bringing together the assessor and the assessed to work as a team to provide organization-compatible recommendations for mitigating the vulnerabilities. This is in contrast to an audit findings report, where a vulnerability may be identified (sometimes accurately and sometimes not) without any recommended action. There are several ways that Security Assessments can be performed, but successful security-oriented organizations use all of the following in order to ensure that their systems and data are protected:

- **Intra-Team Assessments.** Within a team, periodic reviews are performed (as a team, not as an individual) on various areas of the system. An example would be a production support team performing a periodic review of DBA Administrator access levels and controls. Note that the participants in this type of assessment are members of the organization that generally best know the system (and the vulnerabilities), so encouraging this type of review and proper execution can provide tremendous returns.
- **Intra-Organization Assessments.** Having another team within your organization perform an internal review can be beneficial because it usually stimulates a lot of “why?” questions. “Why” questions are a beneficial tool to identifying and resolving vulnerabilities when they are asked (and answered) in a safe and open environment without fear or retribution. When handled and facilitated properly, these questions often highlight breakdowns in procedures and security issues.

A suggestion to make any of the above Assessments more interesting is to make the process a “fun” competition. Break up into teams and make it a contest to see who can point out the most vulnerabilities. You will be surprised as to how many can be found!

- **Third Party Assessment.** A third party assessment can be especially beneficial because it brings in knowledgeable security (and preferably E-Business Suite) experts to review your system security and work together as a team to improve security – not just produce a report stating perceived vulnerabilities. As you already know, the E-Business Suite is a complex series of products, tools and utilities delivered as a single system that takes years of practice and experience to master. The reality is that most auditors do not have this extensive domain knowledge and experience about the E-Business Suite (e.g. What do you mean every database user is APPS?), and therefore, audits are often not a true indication of the security of your system.

Unlike an external audit where an outside party lists what they perceive to be shortcomings, a third party assessment can bring missing E-Business Suite knowledge and a independent perspective to the team to work with the organization to suggest real world security improvements that may not otherwise be detected by audits or internal reviews.

Additional Security Controls and Actions

Here are some additional security controls that can be started after ‘Day 3’:

- **Be Paranoid!** Think like one of the bad people and assume the worst case scenario, rather than the optimistic approach of “It will never happen to me”. Remember, each of the “bad guys” in the case studies described in this paper had co-workers. Almost all of these co-workers couldn’t believe what the “villans” did to the company. Their fallacy – they trusted them.

- Review/Update/Create Security Processes, Procedures and Policies. Many organizations do not have defined and documented security policies and procedures. These are needed to provide consistent, repeatable security standards to ensure compliance, to provide to auditors, and also to serve as an educational tool for new employees.
- “Harden” your system software. The Operating System, Database, and E-Business Suite Technical Stack all can be configured, changed, and modified to be more secure. “Hardening” consists of disabling unneeded services and changing default configurations (passwords, ports, etc.). Many documents, papers, and books exist on how to accomplish the hardening for each of the software tiers. A hardening procedure should be created and documented as a corporate standard, and adapted only with proper approval when appropriate.
- If you expose all or part of your E-Business Suite environment to the internet, then be sure you are following the recommended best practices by Oracle. They have published a several very good whitepapers on this topic and have some very good practices to follow:
 - 287176.1 DMZ Configuration with Oracle E-Business Suite 11i
 - 380490.1 Oracle E-Business Suite R12 Configuration in a DMZ
- Strive for continuous process improvement. Security is not just a one-time event, it is a continuous process that needs to change and adapt with the organization.

Better R11i Security in Three Days - Conclusion

The world is long past simple username/password control for protecting an organization’s data. There is no single answer or single solution. "Security" vs. “usability” comes down to the concept of having multiple layers of controls and the extent of those controls is a delicate balancing act between letting the right people in and keeping the wrong people out. In order to manage this you have to weigh the value of the information against the barriers you need to erect in order to secure it.

Remember, the internal threats are still very real, so it pays to be prepared *and* paranoid to keep the bad, and the “badder” guys away!