



CHAPTER 1

Security Fundamentals for Oracle E-Business Suite

2 Oracle E-Business Suite Security



Oracle E-Business Suite is a mature application suite that has been around for many years, evolving through versions of functional and technical components and technology stacks. I still have memories of Oracle's First Report Writer (RPT) and Forms 2.0 and using the vi editor to perform changes. Back then, no customer would have considered security of an application to be important (perhaps other than controlling who could see what data in which cost center), and most customers I visited in the early 1990s never wanted to know anything more than how to change responsibilities or passwords. How the world has changed! Now it is becoming commonplace to see Oracle E-Business Suite software facing external customers, suppliers, retailers, employees using the Internet to access application modules, such as iExpenses, or external integration, such as XML Gateway, etc.

The fact that Oracle has its own technical suite of products that has been around even longer than Oracle E-Business Suite means Oracle has implemented good security practices from the ground up. "So," you may ask, "why is a book about Oracle E-Business Suite security even necessary?" Well, the answer is easy: With a product like Oracle E-Business Suite that has many components and a flexible architecture, it is critical to understand the techniques and approaches that can be used to implement secure applications that can change as the business demands. Current IT implementations generally take the "just in time" approach. Also, security is much more than just, "How do I harden that web server?"; companies have to think about data protection acts, Sarbanes-Oxley Act (SOX) compliance, data encryption, data integration, access control, and data access control. The list is endless...and growing by the day.

In recent years, Oracle E-Business Suite has gone from a back-office application that is used by the financial department to the core of the business's day-to-day operations, making its continuous operation critical to many companies. Consider what would happen if a salesperson could not get their bonus, if payroll stopped running, if the U.K.'s Bankers Automated Clearing System (BACS) transfers did not happen—it would not take very long for a business to stop functioning smoothly. Also, consider what would happen if external auditors or financial controllers wanted to see information for the massive amount of regulatory and compliance procedures that have to be followed, such as SOX, the Health Insurance Portability and Accountability Act (HIPAA) of 1996, the European Union's Data Privacy Directive, and so on.

Every Oracle E-Business Suite component has a security element, even if it is simply a matter of validating responsibilities or a set of books. In this book, I cover the important security features by providing a walkthrough of the E-Business Suite architecture and software components. My goal is to help customers and implementers understand important configuration issues and the decisions that they need to make

at each step. I concentrate on the functions and features that are needed in most organizations and companies to maintain regulatory compliance; what happens when “Autoconfig” settings are set; what to audit.

This chapter introduces you to the security concepts and elements that are important before we consider Oracle E-Business Suite functionality. Setting the ground rules of engagement is critical to successful security. Understanding how the corporate security policy will need to be adapted for Oracle E-Business Suite and knowing the threats to the organization are critical.

Security Foundations for Oracle E-Business Suite

This section defines the security fundamentals for Oracle E-Business Suite and how corporate policies will need to be impacted.

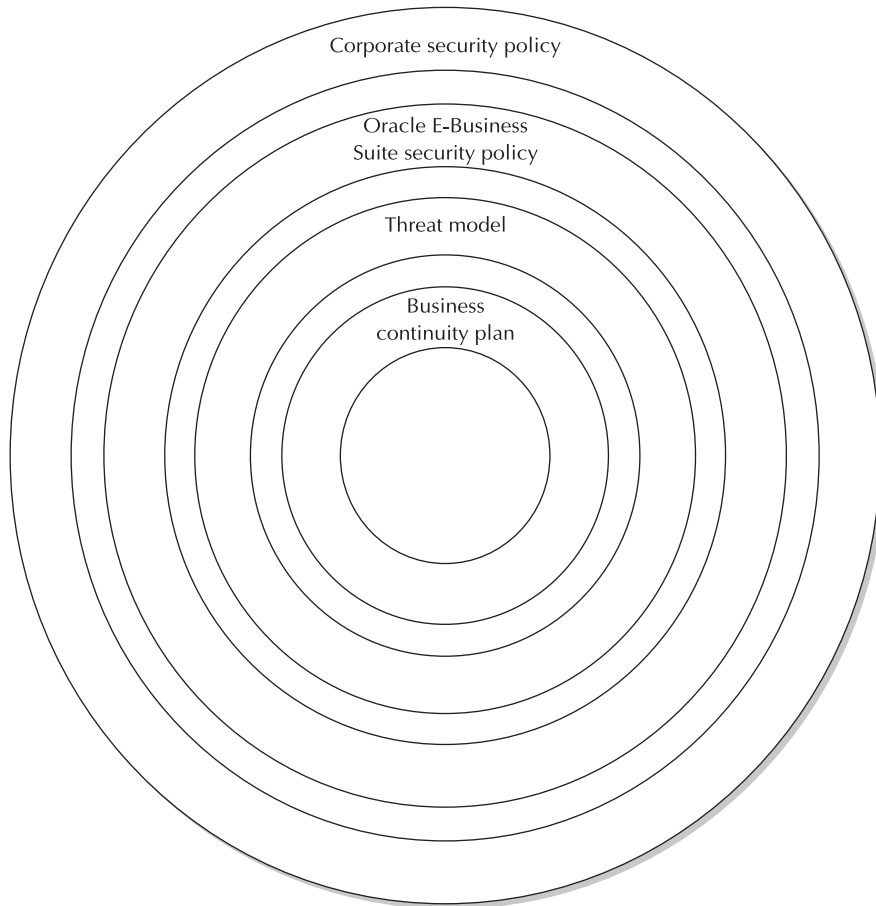
Obtain the corporate security policy to understand specific requirements and map this to Oracle E-Business Suite’s functional and technical elements. Every company has a security policy, although it may not be called that or formalized in writing; such things as requiring visitors to register at the reception desk, issuing a temporary ID card to visitors, locking doors at the end of the day, and so forth are policies that companies implement for security and access control.

The following are the four important documents or deliverables that you need to consider when implementing any software component for which security is a concern:

- **Corporate security policy** The overriding policy that will be referenced within the application or project security policy. This may be a policy that is required by statute and that must be supported by the application. The corporate security policy normally is referenced in the Oracle E-Business Suite security policy. An example of a security policy statement would be, “All users need to include numeric and alphabetical characters when setting their password.”
- **Oracle E-Business Suite security policy** Defines the security policy components that are important for Oracle E-Business Suite elements. This would include all areas of security that affect the Oracle E-Business Suite implementation. The corporate security policy example from the preceding paragraph would be cross-referenced in the Oracle E-Business Suite security policy as “Oracle E-Business Suite must have ‘Signon Password Hard to Guess’ set to True.” This shows how the corporate policy relates to the Oracle E-Business Suite security policy.

4 Oracle E-Business Suite Security

- **Threat model** Defines the threats to the Oracle E-Business Suite application. This is a typical extension of the corporate threat model as most threats are common. Some people consider the threat model the starting point for security. A threat is any action that is classified as a risk to the business from known and unknown sources. Each threat needs to be addressed by a security policy statement. When defining the threat, you need to analyze a number of elements, including the impact, risk, and forensic elements that can be used to validate whether the threat has occurred within the production application. So, using the previous password example, if we do not have the “Signon Password Hard to Guess” password policy set, then the risk is that users will have simple and guessable passwords, the impact of which is possible unauthorized access.
- **Business continuity plan (BCP)** Defines the plan to keep the business running as normally as possible during disasters, security compromises, and outages. For a mission-critical system, this is a critical document that covers how each policy or threat needs to be dealt with to make sure that the business continues to run normally and what process to follow during this difficult period using a proven and tested plan. Most customers consider the disaster recovery plans as the most important when thinking about the business continuity area, which are extreme outages. More normal outages could cause business “pains,” that is, loss of a disc. The new role in organizations of Chief Risk Officer (CRO), which is replacing the Chief Information Security Officer (CISO) and Chief Security Officer (CSO) roles, is showing how corporations see security as a risk to the business. All business areas need a BCP, to provide a clear approach to handling issues during various crises. The following illustrates the “rings” of security that should be considered in all implementations of Oracle E-Business Suites, even if it is simply a statement acknowledging that each area has been considered. Using the password example, the BCP would need a description of what action the company should take if a password has been compromised.



Oracle E-Business Suite Security Policy

This section addresses the important elements that you should consider for the Oracle E-Business Suite security policy. You need to understand at the outset that every E-Business Suite implementation is different, and thus not all of the points will be relevant to your implementation. Use the corporate security policy as the starting

point for the Oracle E-Business Suite security policy statements, reference the corporate security policy for generic application statements (e.g., session timeout), and extend where specifics to the application are required (e.g., Internet deployment and the use of a reverse proxy server).

Business Objectives and Goals

Before you start to consider the security elements and principles that you should apply, you need to understand the business's current implementation and what it plans to implement in the future, both short term and long term. Once you have gathered the details, you can evaluate the security criteria. An important point to remember with Oracle E-Business Suite is that after it is implemented, additional components can be implemented that will affect the security policies and procedures that have been put in place. The following are some examples of business changes that could affect the security policy:

- *The business decides to implement Oracle iReceivables for home-based workers.* It is becoming more common for home-based workers to access internal systems. In most cases, this is an extension of the corporate intranet using secure connections (e.g., Virtual Private Network Software), but in some cases it could be a Secure Sockets Layer (SSL) Internet interface.
- *The procurement department wants to implement "punch-out" calls via iProcurement.* Implementing this component of Oracle E-Business Suite requires a network connection from inside the corporation to the Internet via a web proxy. I am commonly asked, for example, "Why doesn't iProcurement work? We have carried out the setup as documented." I then find out that they forgot to inform the network team of the firewall rules that are required.
- *The HR department wants to start using iRecruitment for applicants to apply for jobs.* With this example, we have inbound Internet-based traffic, which again means that we need to take extra care to secure the Oracle E-Business Suite with the security that is applied to the web servers, database servers, firewall, denial of service (DoS) virus scanners, etc. This reduces the threats to the company.
- *The European operation requires access to the U.S. operation's E-Business Suite implementation, including access to the Professional and Self-Service interfaces, and direct SQL access.* Large corporate organizations may have firewalls separating each country and the firewall may not be compatible with SQL*Net for direct access to the database. So always understand how and where the user bases access Oracle E-Business Suite to make sure testing is carried out.

The preceding examples are different for every organization and can change fast as business demands. So, building security architectures that can be changed is a critical prerequisite before you start to think about the details to implement Oracle E-Business Suite. Most businesses have methods to request changes to security infrastructure (that is, firewall rules), but consider how a business user, when looking at new application components, will know the process of how to involve the relevant technologists within the organization.

Security Policy Principles

Use the following points as guides to establish the balance between security and using standard product. If you start to heavily customize the Oracle E-Business Suite, the cost of implementation and support may outweigh the security benefits.

- **Custom security** If a security policy statement needs a custom change to Oracle E-Business Suite, consider the cost to the business and any additional threats that may be introduced. Oracle has spent many years on security elements that one custom change can open up a new weakness. Building on top of an application means that extra development, testing, and support are required that would not otherwise be needed if a commonly supported standard approach were used. In some situations, custom security is required, in which case Oracle E-Business Suite has best practices and documents to assist you. An example of a custom solution that was needed in most implementations but was still intrusive in Oracle E-Business Suite before 11i.10 was the need to load suppliers from historic systems in bulk into the PO_VENDORS table. In E-Business Suite 11i.10, a PL/SQL API exists that allows upload of vendors showing policy may need version-specific information as the API is a more secure route to adding, changing vendor details.
- **Security policy content** Oracle E-Business Suite has a large number of functional security components that should be included in the security policy. Documenting in the security policy the application features that are required would assist the implementation team—for example, reviewing all user permissions before migrating into environments. Also consider the nonstandard elements that may be required, such as a direct API call for bulk upload of employees and responsibilities.
- **“Living” document** Never consider the security policy to be complete. When taking on major release functions, patches, etc., always consider how they impact the security policy and what changes or additions may need to be made. This is one of the important points I make clear to every customer. I recommend producing an approval form that includes a checkpoint to

ensure that the impact on the security policy has been evaluated. As an example, most customers will implement certain features and then say, for example, “I want to use punch-out Procurement (e.g., iProcurement) for supplier catalogs,” without considering the security implications. Businesses do not stand still, which is why most security documents need to be continually reviewed and updated.

- **Local legislation** The security policy must consider the impact of any legislation that applies to the company or organization. For example, U.S. companies must consider Sarbanes-Oxley, HIPAA, and the “safe harbor” provisions of the E.U.’s Directive on Data Protection; E.U. companies must consider the Directive on Data Protection; and U.K. companies must consider the Data Protection Act of 1988. Data ownership (what if it is an offshore implementation?) and global E-Business Suite implementations also need to be considered, because legislation from multiple jurisdictions may apply. For example, the security policy must consider how a shared service center works when dealing with local legislation and compliance. These are issues about which customers regularly have questions.
- **Product constraints** Because Oracle E-Business Suite is an “off-the-shelf” application, some constraints exist and need to be documented within the security policy so intrusive customizations are avoided in order to meet security requirements. An example of this is when Oracle audits user connection to Oracle E-Business Suite, it is only possible using standard features to monitor the user access, screen access, and which responsibilities have been used. The fields that are audited within Oracle E-Business Suite are also fixed.
- **Corporate policy** Most organizations have higher policy statements that may impact the implementation and may need to be taken into consideration when implementing Oracle E-Business Suite. For example, central support teams (such as DBA) will have standard support documents for environments that are not specific to Oracle E-Business Suite and will need to be extended to encompass E-Business Suite–specific methods to support the application suite.
- **Real-world policy** Never add a policy statement just because it is “a good idea”; it is easy to get carried away with the policy document. Employees, customers, and users need to believe that the policy is relevant to the application area. Also consider having both a full, internal policy and a basic “cheat sheet” for different styles of users. An example of a policy statement that may be considered “silly” by users is “Passwords must be 16 characters or more and include both upper- and lowercase letters and numbers.” Most users would forget their password, and the maintenance in changing them would outweigh the security benefits. Engage all parties (including support staff, managers, and the like) during the design of the policy. Do

not allow the policy to become a “shock,” because this will not be well received and in most cases will be ignored or worked around.

- **Security policy threat** The full policy should have limited circulation because it may include information about weaknesses that could compromise the integrity of the system if a vulnerable area is exploited. This is another reason for having different levels of policy documents for different user communities. The PO data entry clerks need to know the rules that apply to their job, rather than, for example, what happens if a buffer overflow occurs or a DoS attack is happening.
- **Security policy levels** Each policy statement needs to have an example and give a classification of the risk, threat, and impact. This should be referenced within the threat model.
- **System impact** Consider whether the policy statement will impact the capacity plan, database sizing, and performance. An example of a system impact is adding auditing on a very active table such as GL_ACTUALS. Because the number of rows could be in the hundreds of thousands, the additional latency that is incurred with the audit actions may miss critical business key performance indicators. That is why policy statements need to be added at the start of projects, not at the end. Who wants to retest twice?



TIP

A security policy is not just about access, hardening, and auditing. It should cover all areas of security that need to be addressed by the application. Also, the document is “living,” so do not ever consider it complete; the world does not stop just because your policy is complete. The last issue, and in my view the most critical, is to get the balance right, between user requirements and security.

Threat Model

A threat model should also be considered a “living” document, because a threat that the business views today will most likely change tomorrow. Before you consider the important parts of a threat model, take a moment to consider what *is* a threat.

“Capabilities, intentions, and attack methods of adversaries to exploit, or any circumstance or event with the potential to cause harm to, information or an information system. Any circumstance or event with the potential to harm an information system (IS) through unauthorized access, destruction, disclosure, modification of data, and/or denial of service.” National Information Systems Security (INFOSEC).

Based on this definition we need to consider what threats exist that are relevant, such as capture password. Within this section we define the high-level principles that apply and an approach towards threat modeling.

Threat Model Principles

The following are the important principles that you need to consider for your threat model:

- **Threat** All known threats that the business needs to protect against should be included in the threat model and will need updating as new threats are uncovered by security experts and hackers. When updating the threat model, consider how many security fixes are released by software companies to counter the threats and how you should be prepared for this additional effort. A threat could be an internal or external party and could be a physical threat or an electronic threat. An example of a threat is someone obtaining credit card information from the database.
- **Impact** Once you have listed the known threats, consider the impact and the importance of the threat. In some cases, the threat may have a low impact and only forensic evidence needs to be captured. But all threats that are known need to include an impact statement. Using the credit card example, the impact assigned to the threat would be high, because if credit card information is stolen, the impact would at the very least be negative press reports and loss of confidence in the company's ability to protect personal information.
- **Mitigation** When defining the threat, any ways to mitigate the threat should be included. For example, if the threat is a virus, a mitigation statement may be, "The corporation expects all Internet-facing servers to be protected from viruses by the use of up-to-date virus-scanning software." If mitigation is not possible, then the threat should be considered a business risk and the cost and likelihood of the risk need to be defined. The mitigation plan for the credit card example is to make sure data is secure and encrypted in the database and in transit.
- **Cost** Implementing security features will cost the business both money and time. However, this cost must be weighed against the cost to the business if the threat is not mitigated and impacts the business. If the cost of protecting against the threat is not warranted, then the threat should be considered an acceptable business risk. When considering the cost, consider not only the monetary outlay but also the impact on the business with regard

to customers, suppliers, and so on, if the threat occurs. The credit card example may not cause major financial implications, but the bad PR would cause a large impact on future business and current customers, as they will consider any breach as a serious lack of security.

- **Likelihood of threat** When a threat has been identified, consider and document the likelihood of the threat occurring. This is difficult with certain threats, but you should also provide cost estimates to the business to remove the possibility of the threat. In the credit card example, sending unencrypted data over a nonsecure communication channel presents a high likelihood of theft, whereas sending encrypted data over a secure channel presents a low likelihood of theft.
- **Electronic forensic evidence** The most important element of the threat model is to document each compromise, detail what evidence is required when the threat has occurred, and provide evidence that could assist in identifying the originator of the threat. The evidence that is captured may be used to prosecute an individual or group that caused the threat. Using the example of credit card information, make sure audit records and transmission data (such as electronic files) are retained so that you have an audit trail to show if any security issues existed.



TIP

Electronic forensic evidence could be a web server log, but in some cases it could be current runtime memory and operating system state. So, when investigating security threat issues, do not panic and shut everything down, as some critical information could be lost. Consider restricting access to the server. For example, if a firewall is between the users and Oracle E-Business Suite, then deny access at this level.

Threat Model Approach

After you define the high-level principles, you need to decide on the approach to threat modeling. Each individual organization should consider the approach that suits its business and E-Business Suite implementation. A threat model will grow with the organization and will need to be updated at regular intervals. It may be worth considering placing the threat model within an electronic store so that searching and access is made easier (a number of tools are on the market to control the threat model). The threats included in the threat model should only be relevant

to implementation and should not be cluttered with “nonevents.” The following scenarios explain the different approaches and how each threat model can be either very simple or very complex:

- **Internal Oracle E-Business Suite implementation with no external access outside the local office, and only ten back-office users, using core Enterprise Resource Planning (ERP)** The threat model for this scenario will be lower in complexity and may only require a small number of threats to be defined. Example threats are “Loss of data due to backup corruption,” “Self-approval of an invoice,” “Default passwords,” and “Open accounts after an employee has left the organization.”
- **Medium-size corporation performing core ERP and limited intranet self-service access, such as iExpenses, across the entire organization, but still local within a physical country** The basic threats listed in the first scenario still are required, but additional threats are required, such as “Firewall impact across offices,” “Secure communication for electronic data transfer,” “Risk of central server failure,” “Disaster recovery approach,” and “Lack of local system support.”
- **Large multinational corporation implementing Internet- and intranet-facing self-service components, Customer Relationship Management (CRM) and core ERP, and multilingual support** The threat model required for this approach will be the most detailed and complex of all three options. Also, threats will need to be defined in some cases on a per-country basis. Some examples of corporate threats are “Loss of intercompany network,” “Local country access control policy/monitoring,” and “Corruption/support of local multilingual implementation.” Note that within this scenario some threats are not relevant to software/hardware components, but it is still critical to understand other outside issues that could be considered a threat.



TIP

When defining the threat model, involve core people representing all aspects of the business. You may know one complete area of the implementation but have limited knowledge of another. Also, consider using an external consultant to assist in “fast tracking” a threat model.

Business Continuity Plan

The BCP is not technically a security document, but it is very important. The business must consider the threats and include in the BCP any that have enough

impact to cause a continuity element to be implemented. When defining the BCP, consider the financial and time costs of implementing business continuity as a “must do”; if not implemented, then serious issues, such as loss of critical systems, will arise. A denial-of-service (DoS) attack could be a major vulnerability that requires the server to fail over onto a backup server or even a complete new environment at a different location due to a system outage.

Some businesses confuse business continuity planning and disaster recovery planning. Both are critical, but the BCP will be used more often than the disaster recovery plan (e.g., whereas a broken payroll printer will be part of the BCP, total site failure will be part of the DRP). In some cases, both documents will be defined with a high-level statement in the BCP and the low-level implementation details in the DRP (e.g., total site failure).

Business continuity and disaster recovery plans are usually complex in nature and cover all serious scenarios that affect the business operations. The team that looks after the BCP elements needs to know the risks that require the most attention, because a risk has to be classified in the same way a threat is classified.

A DRP is used only during a critical period, so make sure that it is clear and follows an easily understandable flow. Most plans also indicate the management escalation route to fast track issues.

Oracle E-Business Suite Primer

Now that we have set the scene with the important security principles that are required, we need to look at the Oracle E-Business Suite to understand how these security principles relate to its components and security. At this point, a recap of Oracle E-Business Suite key areas and product versions is also worthwhile.

Evolution of Oracle E-Business Suite Security

Recent versions of Oracle E-Business Suite are increasingly keeping pace with the latest Oracle technology stack (e.g., Oracle Real Application Clusters support and Oracle Database 10g). For many years, it was normal for the Oracle E-Business Suite to be a version behind the Oracle technology stack of the time. The application technology lagged behind the most up-to-date version of the core technology product due to retesting and impact on customers. New methods exist that allow development to separate components of Oracle E-Business Suite (e.g., Oracle Application Server 10g and Oracle Real Application Clusters 10g), so new functions can be utilized within implementations. Table 1-1 provides an outline of the key security features that have been released over the years; too many security features exist across Oracle E-Business Suite to list them all.

E-Business Suite Version	Key Security Components Introduced in Version
Pre-11i.8	<p>Username and password support the basis of access to Oracle E-Business Suite.</p> <p>Responsibilities to allow grouping of functions (e.g., AP Manager).</p> <p>Oracle E-Business Suite auditing and database auditing within AUD\$.</p> <p>User information is stored when creating or updating Oracle E-Business Suite tables that can be used for auditing purposes.</p> <p>Privileges to control access to functions and menu items. Restrict by user or responsibility.</p> <p>User/system profiles and report grouping to control system access.</p> <p>Features to assist with HIPAA implementation. (This is only one element of HIPAA compliance; the business still needs to implement procedures, etc.)</p> <p>FSG Hierarchical Security Support when producing FSG reports.</p> <p>Hierarchical Security Support to allow access at certain levels within the chart of accounts for posting of journals to the general ledger.</p> <p>Segment Value Security for Journals Reports to make sure users can only report on the business areas to which they are allowed access.</p>
11i.8	<p>Enable Oracle Label Security and Virtual Private Database (DBMS_RLS) for logically splitting the database.</p> <p>Implement reverse proxy configuration for Internet deployment. A reverse proxy is when a web server (such as Oracle Application Server 10g) will proxy requests from one source to Oracle E-Business Suite application tier.</p> <p>Implement single sign-on (SSO) and OID with Oracle Application Server 10g technology stack, providing integration with external authentication and directory stores (using this feature requires a number of patches).</p> <p>Oracle Assets "Security by Set of Book" feature allows segregation of fixed assets.</p>

TABLE 1-1 *Evolution of E-Business Suite Key Security Components*

E-Business Suite Version	Key Security Components Introduced in Version
11i.9	<p>Sarbanes-Oxley compliance (a patch can be applied for previous versions); the software elements assist when implementing SOX, but the business still needs to implement the procedures, etc.</p> <p>Account Hierarchy Manager Security for defining account rules to be applied against each account.</p> <p>Internal Controls Manager (ICM); important component for SOX compliance. This allows corporate governance to be controlled and provides a single point for internal controls.</p> <p>Cross Instance Data Security allows data to be shared from one database to another (for example, general ledger records) and maintains the security attributes.</p>
11i.10+	<p>Oracle User Management introduces Role Based Access Control (RBAC) model. This allows users to be given a role rather than a responsibility, and includes heritage of function permissions.</p> <p>Support for customizations in Apache/OHS that are not removed by AutoConfig.</p> <p>Oracle Application Manager security wizards for advanced topology configurations. These include SSL Certificates, SSL Appliance, and Load Balancing options.</p> <p>Oracle Database 10g and RAC support providing high availability and new security components, including DBMS_CRYPTO and column-level security (using DBMS_RLS PL/SQL package).</p> <p>URL Firewall is the ability to use Apache Directives to restrict access to specific web pages. This is a shipped list of restrictions allowing for a fast start when implementing Oracle E-Business Suite.</p>

TABLE 1-1 Evolution of E-Business Suite Key Security Components (continued)



TIP

If you need to use one of the features listed in Table 1-1 within your business, obtain the relevant patches that are required. Also, compliance and regulatory features are only the software element, not the full requirement for being compliant with applicable legislation.

Oracle E-Business Suite Technical Components

Oracle E-Business Suite has a large array of technical components, detailed in Figure 1-1. It is important to understand how different types of protocols, connections, user interfaces, and processes will work within a full E-Business Suite implementation of all functional components. Detailed discussions in Chapter 3 cover the configuration and key implementation issues when securing each layer.

The following sections summarize the components shown in Figure 1-1 and describe the wide range of interfaces, connections, processes, and application fragments that make up Oracle E-Business Suite.

Oracle Professional User Interface

Oracle Professional User Interface is used mostly by back-office personnel who require a desktop-style interface for data processing. E-Business Suite has utilized Oracle Forms since the beginning of the product. The forms process runs as a Java plug-in (JInitiator). A desktop tool called Web Applications Desktop Integrator (commonly known as Web ADI) can also be used within the browser or as a standalone component for management or data manipulation.

Different styles of protocols, including HTTP, HTTPS, and socket mode, are supported and you need to consider the implications of implementing each protocol. An example of this is that socket mode may not be allowed over the corporate firewall infrastructure. Also, the overhead on the server of HTTPS (SSL) may mean additional hardware will be required. We will discuss this in Chapter 2 in more detail.

Self-Service User Interface

Self-service user interfaces cover a wide range of Oracle E-Business Suite products that are used within intranet and Internet deployments. The self-service user interface is generally the one most clients want to Internet-enable and thus usually requires the most security considerations when deploying. Self-service user interfaces are deployed using the Oracle Application Development Framework (ADF), which is customizable and extendable and has a personalization framework. This means that when you are deploying custom changes, you need to consider whether any security issues may arise for customized code. An example of this may be the inclusion of a new field that potentially allows a cross-site scripting (CSS) attack.

Development User Interface

Oracle E-Business Suite is customizable using a number of development and enhancement applications. Most customers in general are concerned about data loading or simple customizations, which are implemented by using Oracle Forms, Oracle Reports, PL/SQL, and SQL*Loader. Also, some tools are used by users, such as Oracle Discoverer and SQL*Plus, to query data content directly within the

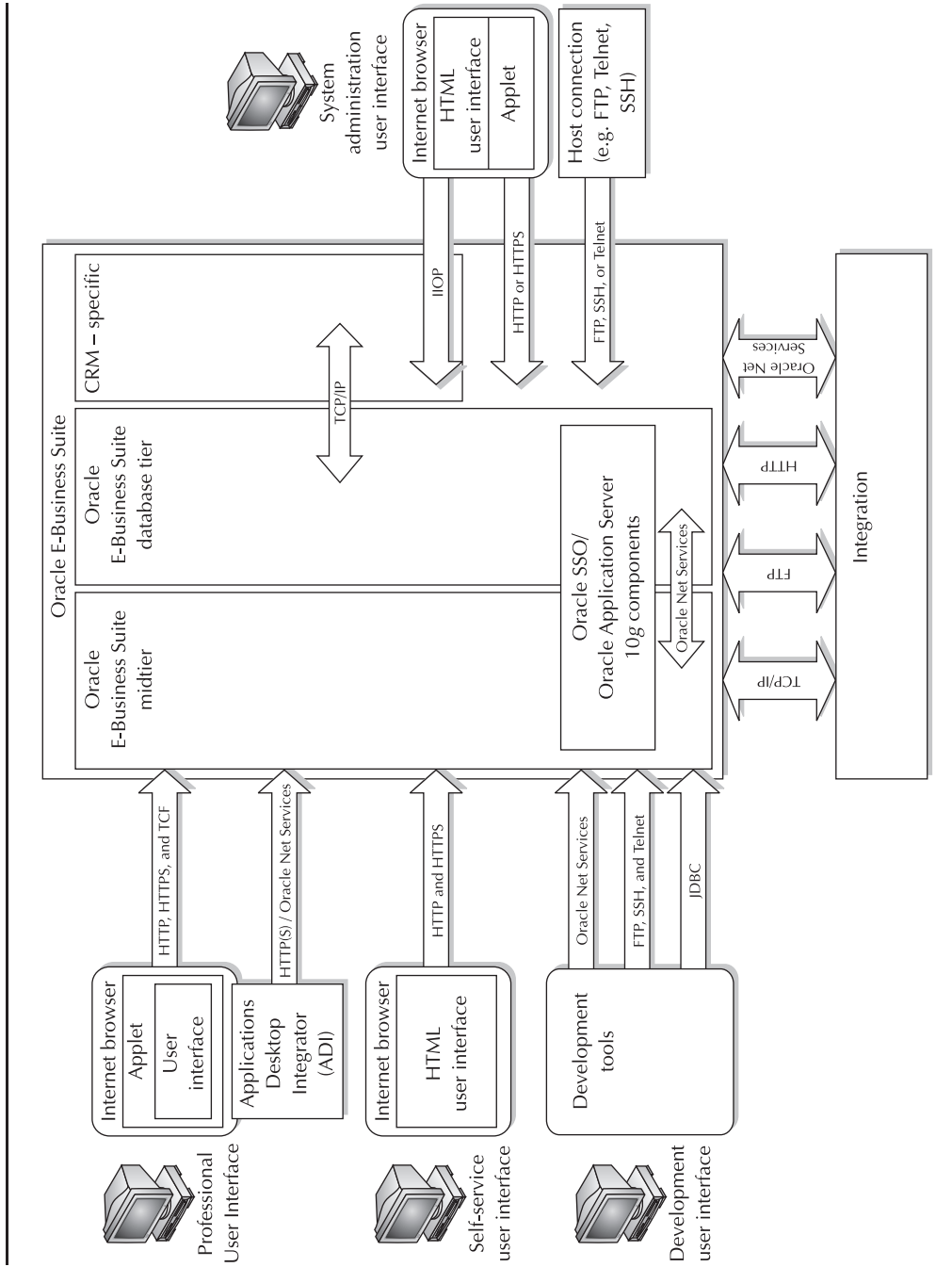


FIGURE 1-1 Oracle E-Business Suite components

database. Oracle JDeveloper can be used to customize the Oracle ADF (that is, self-service user interface). Due to the nature of the development tools, numerous protocols will need to be supported, such as HTTP, SQL*Net, JDBC, and so on.

Oracle E-Business Suite Application Tier

Within Figure 1-1, the application-tier component is split from the database server as a logical split, but it is possible to implement the database and application tiers together, or a number of application-tier servers together, for resilience and scalability. Most application-tier components provide user interface, such as self-service user interface or professional user interface. In some cases the concurrent managers are installed on application-tier server(s) to process batch work, but in the figure we have left them on the database server, which is the normal approach. Oracle HTTP Server (OHS) is the center pin for all initial connections; OHS is based on Apache and was enhanced by Oracle.

Oracle Application Server 10g components can be used for SSO, Oracle Portal, and LDAP integration and usually run on their own server, but in small implementations they may run on a midtier or database server.

Other Oracle9iAS R1 components include Oracle Portal, Oracle Forms, web-based Oracle Discoverer, and a J2EE user interface. Each component can be deployed using different protocols/configurations (for example, Oracle Forms can be deployed as CGI or Servlet).

Oracle E-Business Suite Database Tier

The database tier that is shown in Figure 1-1 has a number of components and, as stated with regard to the application tier, could contain all Oracle E-Business Suite components or just the database-related items.

Concurrent managers cover a wide range of background processes that manage user process requests or standard background tasks such as purging. In general, concurrent managers receive requests and produce output, run application components, or perform housekeeping task.

Oracle Workflow notification mailer is used to handle flow logic of Oracle Workflow and to send e-mail notifications to users for action, such as the approval of a purchase order.

The database can be a single database with a single database instance or multiple instances within an Oracle Real Application Clusters (RAC) configuration. Each database will have a net service listener handling connections. RAC will provide a high-availability option if required, but consider the more complex topology before deployment in some situations.

Oracle Identity Management infrastructure provides the LDAP service through Oracle Internet Directory and integration with SSO and other LDAP services.

Oracle Customer Relationship Management (CRM)–Specific Functions

When implementing Oracle CRM, and specifically integration with an external telephony service, integration will be required to support “screen pops,” call routing, and physical phone switches. Oracle eMail Server is used for marketing or outbound campaigns; this would normally be integrated into the corporate e-mail system. Consider the security issues in having marketing information over the corporate e-mail system (for example, marketing campaign could impact other production systems).

Integration

Oracle integration products provide the facility to integrate data, send messages, and provide external APIs into Oracle E-Business Suite. It is becoming increasingly common to have distributed applications and systems that need integration. Different styles of deployments are required within businesses, such as application-to-application (A2A), product-to-product (P2P), business-to-business (B2B), or a combination of methods. Also, the supported network protocol could be X.25, TCP/IP, HTTP, HTTPS, FTP, and so forth.

Application-to-application integration is becoming a common request for both off-the-shelf and custom-developed applications.

Invoking an interface point from Oracle E-Business Suite can be either direct, via a “data hook,” or indirect, via a common message. The interface can be multidirectional, calling an API to retrieve data and pass back to the calling system. Outbound data feeds use either database triggers, CUSTOM.pll (Forms method of customization), or “data hooks” from the foundation layer.

It is even possible to support Business Process Execution Language (BPEL), which is an emerging standard for developing and deploying discrete services for process flow. Oracle now has BPEL within Oracle Application Server 10g that could be used to extend Oracle E-Business Suite and enhance the automated business product.

System Administration User Interface

Management interfaces within Oracle E-Business Suite could be either Oracle Applications Manager (OAM), Oracle Enterprise Manager (OEM), also known as Grid Control within Oracle Application Server 10g and Oracle Database 10g, or local administration tools that assist the day-to-day running of the application server. Tools can be used over standard protocols like Oracle Net Services, HTTP, and TCP/IP.

Additional E-Business Suite Implementation Security Considerations

This section deals with additional areas that you need to consider during an E-Business Suite implementation. Each area may or may not be important to your implementation, but you should still consider each to see if it is relevant.

- **Support** When you implement E-Business Suite, support teams need to access the database and may need to apply correction scripts. Document the approach and business practices that need to be followed. Consider the access method and tools that could be used. Certain data access tools may allow data changes that cause an issue when base data is changed and saved, which could lead to a system crash. Consider your data protection and data security rules. Some large clients are now implementing “shared service centers” that handle the entire corporate business processes, even across countries.
- **Capacity and sizing** This is a security issue if the server was sized for a small number of users with low headroom. A DoS attack would cause an impact earlier because the machine’s resources will be flooded, whereas a machine with higher headroom would allow administrators to resolve what is causing the excess usage of machine resources. Also consider database growth, and so forth, and how this is monitored. Could a potential hacker flood the inbox for notifications by capturing a notification and replaying when using an insecure connection? This also highlights the importance of DoS and intrusion detection devices.
- **“Cutting corners”** The modern world wants applications deployed faster and cheaper. That is a great idea, but consider if cutting corners will result in a less-secure environment. When you consider what elements are going to be altered during a project, always think of the risk now and in the future. An example of cutting corners would be storing keys for encryption in a PL/SQL program and not even wrapping the code or taking a further step by removing the keys from the PL/SQL program.
- **Hardening** A common term that is used during any implementation, hardening means removing unwanted services, not allowing certain functions, and de-installing some components. This is one of the most difficult areas to cover during the implementation because hardening techniques may impact your support agreement. So, consider how you will implement the hardening features at the operating system level, hardware level, and Oracle E-Business Suite level and still maintain support. A typical approach for testing standard product issues if problems occur is to have standard implementation that is not hardened that matches production. This means any issue can be replicated to

provide test cases to support. When hardening, consider if the user functions will still be possible and whether access control has not impacted the system ability to perform required functions. Getting the balance right and having a method that ensures new hardening recommendations are implemented is the key to a structured, controlled approach. Another possible approach (but consider the support implications) is to have an external firm perform the hardening for your organization; some companies have hardening tools for Oracle E-Business Suite. Also Oracle provides hardening scripts within standard product, for instance, URL Firewall.

- **Penetration testing** When you are happy with your implementation, have a penetration test carried out on Internet-enabled products. Penetration testing can be either white box or black box. White box is when the penetration tester has full knowledge of the topology and environment, including code and IP addresses. Black box testing assumes no prior knowledge of the implementation, with only limited information provided (e.g., URL for Oracle E-Business Suite). A good approach is to start with a black box review, which is generally faster and easier to perform, and then, if this shows issues, move on to a full white box review to dig deeper into the workings. Some firms offer Oracle E-Business Suite validation tools, such as AppSentry from Integrigy, and professional consultants who understand Oracle E-Business Suite components.
- **Control procedures** Implement strict control procedures and a management approval process for proposed changes to Oracle E-Business Suite. For example, suppose that a manager's e-mail account was compromised by hackers and they sent an e-mail, purportedly from the manager, telling the IT department to shut down Oracle E-Business Suite and purge all data that was more than a year old because it was not needed anymore. In this scenario, another system with lesser security than the E-Business Suite threatens the E-Business Suite. So, using strict control procedures and security confirmations, namely, returning phone calls to confirm that the system does need to be taken down, can be a big benefit.
- **Offsite storage** Most companies have offsite storage for backup, disaster recovery machines, and archive storage, so you need to consider the security requirements for offsite storage. For example, consider what would happen if someone were to get possession of the backup tapes and acquire the credit card numbers of customers. Protecting against this may require, for example, encryption of secure data (such as credit cards) and secure storage for backup tapes.

Every implementation and most components have a security consideration. Some are small and some are complex, but you should consider them all and their impact during the implementation of Oracle E-Business Suite.

Another important implementation consideration that may be affected by security decisions and needs to be understood at the start is what configuration options Oracle standard policy considers Certified, Supported, and Customization. The following list describes how these categories work in the context of Oracle E-Business Suite:

- **Certified** Refers to anything that Oracle has explicitly tested, which includes all material covered by published MetaLink Notes. This represents Oracle's minimum recommended level of system configuration.
- **Supported** Covers variants on the certified configurations that Oracle believes will work but hasn't explicitly tested. For example, Oracle certifies the E-Business Suite to work generically with load balancers but does not test every single third-party load balancer available. As such, if there are problems with a specific load balancer, Oracle checks whether it's a generic problem for all load balancers and, if so, fixes it. If it's specific to that load balancer, then it's the third-party vendor's issue, and Cisco defers support responsibilities to the third-party vendor.
- **Customization** Falls into two categories: supported and unsupported. If you follow published methods (for instance, by using Oracle Application Framework [OAF] personalization to hide a field on a given HR form, or using AutoConfig to change a given setting), that's supported. Unsupported, on the other hand, refers to any other changes outside of this (which will include some elements of this book). Oracle's standard support policy for customizations is to verify whether they were done through "authorized" mechanisms and documented in official MetaLink Notes. If so, Cisco replicates the issue and fixes the problem.

Summary

This chapter provided an introduction to security principles that you need to consider when implementing Oracle E-Business Suite or any other IT application. It is critical to lay down a structured approach and always provide a clear path on implementation considerations that need to be addressed by the security policy. The threat model and business continuity plan need to state what your risks are and define how important the security is on a relative scale. This chapter also provided a brief summary of the Oracle E-Business Suite components that you need to consider from a security standpoint; not all will be relevant, but consider the important pieces when implementing E-Business Suite. The last section dealt with external implementation factors that could cause issues during the deployment.

You will not need to implement all of the areas covered in this book, but remember to get the right balance for your implementation. It is better to detail out everything at the conceptual phase, so you can determine what you can maintain at a reasonable cost.

We can now move onto the key infrastructure components that may be impacted by Oracle E-Business Suite implementations.