Oracle Entitlements Server

An Oracle White Paper
September 2008

Oracle Entitlements Server

Oracle Entitlements Server is a fine-grained authorization and entitlement management solution that can be used to more precisely control the protection of application resources. It simplifies and centralizes security for enterprise applications and SOA by providing comprehensive, reusable, and fully auditable authorization policies and a simple, easy to use administration model.

INTRODUCTION

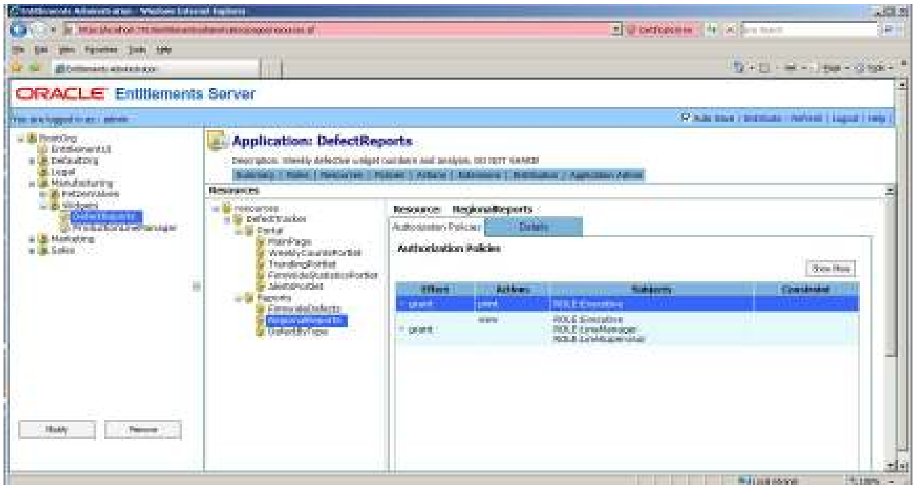
The requirements for providing secure access to internal and public-facing application resources have evolved dramatically in the past decade. Consider a typical Web application. Users access the application's presentation tier via a Web server that is protected by a single sign-on (SSO) application. Access to Web tier resources is managed by the Web-SSO application. Behind the Web tier is an application server, which manages access to the application's elements. Web-based security policies are typically configured outside of the application tier, however, many *application-centric* security decisions are hard-wired into the application logic itself. Application logic making security decisions is not centrally managed, governed or controlled by a security team. To make matters worse, runtime access control decisions are rarely audited. Until recently, most enterprises considered it sufficient to centrally manage security for the Web tier only – centralized application security could come later.

In the last few years however, changes in the enterprise application landscape have mandated a change in the approach to enterprise security. Sarbanes-Oxley mandates documented controls on who can access information systems that affect the finances of publicly held companies. Healthcare and privacy laws have placed stricter requirements on access to applications and auditing of that access. A rapid rise in the outsourcing of application development means that security logic embedded in the application tier is no longer directly controlled by the enterprise. These changes in the regulatory and development environments mandate a change in how access to the application tier is managed.

Oracle Entitlements Server (OES) enables centralized management of entitlements and authorization policies to more granularly determine access to both application components and application business objects. OES also provides a high performance runtime environment to enforce those policies within your application components and objects. The result is a solution which provides lower cost of management for policies across large numbers of applications and organizations without the associated performance penalties associated with a centralized architecture.

POLICY ADMINISTRATION

The policy management lifecycle will be different depending upon exactly how deeply and widely an enterprise is seeking to externalize access control. As such, Oracle Entitlements Server provides a flexible approach to the management of policy. Administrators use a web based console to author, edit and test policies for numerous applications.


ORACLE Entitlements Server
Application: DefectReports
Resources
Resource: RegionalReports
Details
Authorization Policies
Modify
Remove

Effect	Actions	Subjects	Constraint
grant	grant	ROLE Executive	
grant	view	ROLE Executive ROLE LineManager ROLE LineSupervisor	

From this single console, an administrator can set up access control policies for fine-grained application resources for specific users, groups or roles. For example, a simple policy which grants access to Account Reports for anyone who is in the **group** BankManagers would look as follows:

OES takes the responsibility for security policies away from developers and puts it into the hands of security administrators where they can be verified, tested and analyzed..

Grant (Get, //app/enterprise applications/AccountReports/, //group/BankManagers)

We can read this policy as follows, "*Grant the 'Get' action for 'AccountReports' to anyone who is in the user group 'BankManagers*'". Note that this policy is being written at the level of user groups. We can now ensure a user is granted access to Account Reports simply by placing them in the group BankManagers. OES also provides a rich application role mapping facility (also based upon policies) to allow for Role Based Access Control (RBAC) style entitlements:

Grant (Get, //app/enterprise applications/AccountReports/, //role/BankManager)

This is the same entitlement policy we saw earlier however it now is referring to a **role** called BankManager. We can assign users to this role using another type of OES policy for role mapping:

Grant(//role/BankManagers, //app/enterprise applications/AccountReports/, //group/BankManager)

Here we are indicating that anyone in the user group BankManagers will be automatically assigned the role BankManager. The distinction between groups and roles is subtle yet important. Group membership is typically static and changed through a provisioning exercise. Role membership is fluid and may change dynamically as a user interacts with applications. The choice of which to use is dependent upon the actual entitlement need. For example, we could have dynamically assigned someone to the role BankManager based upon some external attribute instead:

Grant(//role/BankManager, //app/enterprise applications/AccountReports/, everyone) if (UserType="BankManager")

Now the role mapping policy uses a **constraint** and an **attribute**. This policy reads "*Grant the role of BankManager for Account Reports on anyone who has an attribute called UserType with the value 'BankManager'*". This starts to reveal the power of OES policies to take advantage of external information during policy evaluation time. Any OES policy can incorporate a robust constraint expression to specify the exact conditions under which the policy applies. Using constraints OES policies can

- Have temporal properties (e.g. expire within certain time limits or have valid start/stop times)
- Use any user attribute (e.g. job location, salary level, customer-grade)
- Leverage external information (e.g. market conditions, risk factors, sales forecasts)
- Perform complex calculations (e.g. compute weighting factors, invoke external services)

Complex constraints can be written that help establish sophisticated authorization policies. For instance, we might want to only grant access to Account Reports for BankManagers past a certain salary level during normal business hours:

Grant (Get, //app/enterprise applications/AccountReports/, //role/BankManager) if (salary_level > 5) AND (currentHour > 8 AND currentHour < 17)

A common requirement with entitlements is the ability to handle exceptional scenarios elegantly. Using constraints we can combine multiple policies to provide fine grained access controls without disturbing existing entitlements. Consider the previous policy which granted selective accesss to Account Reports based upon a set of conditions. What if there was a new requirement to only permit this selective access for BankManagers who work outside of California? Oracle Entitlements Server implements a policy combining system whereby we can override Grants with Deny policies:

Deny (Get, //app/enterprise applications/AccountReports/, //role/BankManager) if state = "CA"

This policy will now override the previous policy only for those BankManager users who live in California. Notice that we did not have to make any changes to the previous policy to accommodate this new requirement.

HANDLING THE NEEDS OF ENTERPRISE POLICY MANAGEMENT

Since Oracle Entitlements Server provides centralized administration for numerous applications simultaneously, it provides a flexible organizational model whereby policies can be scoped to specific applications and isolated from each other. For example, a single OES policy store may be managing policies for multiple applications and organizations where similar resources, roles and actions must co-exist. To accommodate this, OES provides the ability to compartmentalize these into specific applications and organizations.

OES provides intuitive and flexible tools to scale up or down to meet a wide spectrum of administrative needs from single departments to large enterprises.

Figure 1 represents a typical example of such a structure. There are four top-level organizations defined for this policy store (Legal, Manufacturing, Marketing and Sales). Within the Manufacturing organization there are two sub-organizations defined (FetzerValves and Widgets). Inside each "leaf" organization are the applications themselves (ProjectDashboard, Project_X_Collaboration). Policies and their associated artifacts are created directly within an application to provide isolation between applications.

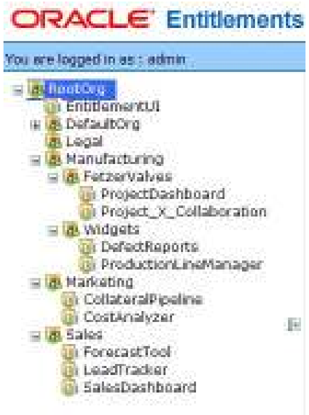


Figure 1 – Managing multiple organizations and applications with Oracle Entitlements Server

This structure also allows for delegated administration across organizations and applications. We can now establish specific administration roles in OES such that a ManufacturingAdmin has rights to only modify the policies within applications contained by Marketing organizations. Or we might establish a CollateralAdministrator who can only manage policies for the CollateralPipeline application within the Marketing organization. Even further, we could specify a CollateralReportingAdministrator role who can only **view** policies for the CollateralPipeline application and not change them.

Another typical challenge with large scale policy management is the ability to test that policy before submitting it to another team for verification or deployment. Oracle Entitlements Server provides a graphical policy simulation environment so that an administrator can test out various scenarios against their policy as if it were running directly inside a real application. This way they can identify policy issues or “boundary cases” well before a development team is engaged to do their own application centric testing.

ORACLE ENTITLEMENTS SERVER ARCHITECTURE

The Oracle Entitlements Server is made up of two major components, an administration portion and a runtime portion. The administration application acts as the **Policy Administration Point (PAP)** and is used to manage configurations, organizations, applications, policies, and roles. The runtime portion consists of one or more Security Modules (SMs). The Security Modules evaluate fine-grained access control polices at the **Policy Decision Point (PDP)** and enforce it at the **Policy Enforcement Point (PEP)**. The Security Modules are also the integration point for user identities and access to external attributes that may be incorporated into the policies. Security Modules can also be configured to retrieve information dynamically during policy evaluations from **Policy Information Points (PIPs)**. These information sources can be relational databases, identity directories, web services or any other source of data.

The OES architecture closely matches the architecture recommended by entitlement standards such as OASIS XACML.

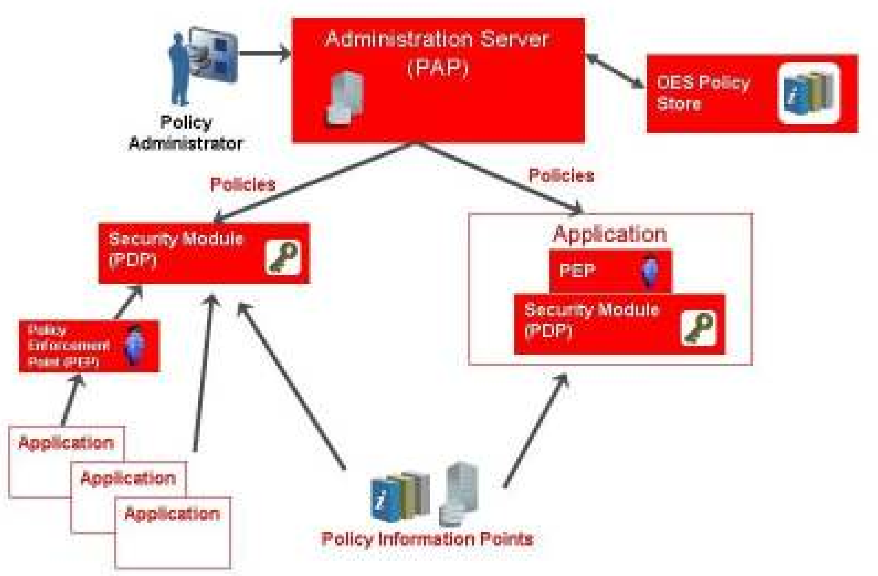


Figure 2 – Oracle Entitlements Server architecture

Oracle Entitlements Server is built upon a robust policy distribution mechanism whereby policy changes made at the PAP can be transactionally pushed to any number of PDP instances. These changes are made available to the PDP immediately and applications using the PDPs do not need to be rebooted or notified of the change. The policy distribution protocol has been designed so that the PDPs and PAP can work completely independent of each other. The PDPs will continue to provide authorization decisions even if the PAP is taken down. If policy changes are made while the PDP is unavailable, the PDP will checks with the PAP when it starts for any policy changes that may have occurred. If the PDP cannot connect to the PAP at startup to guarantee it has the latest policy, it will consult a lightweight file-based policy cache (signed and encrypted) so that it can begin handling authorization requests. The result is a highly scalable and fault tolerant architecture with no single point of failure.

Types of Security Modules

Oracle Entitlements Server provides several types of Security Modules to accommodate a diverse set of software environments. These Security Modules can be categorized into two types:

- Embedded PDP
- Centralized PDP

An Embedded PDP is a Security Module that is contained directly inside an application runtime, typically an application server like WebLogic Server, Tomcat or WebSphere. This configuration is ideal for situations where an application requires extremely high performance from the PDP and is expected to be making large numbers of authorization calls for each user. In this situation the application itself is instrumented to call the PDP directly using the PDP's API. In this case the application acts as the Policy Enforcement Point (PEP) since it will be directly using the resulting decisions from the PDP. Oracle Entitlements Server provides a Java based Security Module which can be embedded into any application server or Java runtime environment.

OES provides a flexible deployment architecture that can be adapted to small departmental and large enterprise scenarios.

The Centralized PDP is a stand-alone Security Module which services authorization requests on behalf of one or more remote client applications. This configuration is useful for applications that either cannot embed a PDP for technical reasons or desire a loosely coupled PDP. Centralized PDPs lend themselves well to a Service Oriented Architecture approach whereby authorization becomes a shared service for multiple types of applications and stakeholders.

Oracle Entitlements Server provides two types of Centralized PDPs, a Web Services Security Module and a Java RMI Security Module. When applications use either of these Centralized Security Modules they call the PDP's API through a **PDP Proxy**. The PDP Proxy takes care of many housekeeping tasks necessary for remote communication such as caching of decisions, handling failover situations and logging.

Applications can switch between the Embedded and Centralized PDP implementations without code changes; only a single configuration file must be modified. This means that applications can be re-architected to use a different PDP configuration without having to change a line of code.

Oracle Entitlements Server provides Policy Enforcement Point (PEP) support in several ways. For many custom applications it is not possible to create a generic PEP implementation since each application has its own way of describing resources and privileges. In these situations, Oracle Entitlements Server provides the PDP Proxy as a way to simplify how the application can implement their own PEP logic. For more structured runtimes it is possible to provide a pre-built PEP which can automatically begin protecting an application without the need for coding. For example, Oracle Entitlements Server provides a Microsoft SharePoint (MOSS) PEP and Security Module which protects web sites, parts, lists and documents automatically. In addition, there is an Oracle Database Security Module which sits beside the database server to intercept SQL requests and ensure queries match the access control policies defined within the Security Module.

Integrating Policies and Enterprise Data

It is impossible for a Policy Administrator to know all of the information necessary to articulate most real-world entitlement policies. As we saw in the earlier section it was necessary to sometimes leverage external data in the form of a policy constraint. Oracle Entitlements Server provides a simple facility in the PDP for describing these data sources for relational databases and LDAP identity directories. Once described, these **Attribute Retrievers** can then be employed to fetch data values which are then stored in attributes and used in policies.

Oracle Entitlements Server also provides a custom Attribute Retriever API so that custom retrieval mechanisms can be developed. For example, an enterprise may store information necessary for an entitlements decision in a proprietary system or service. Instead of needing to copy the data to a database or LDAP entry, OES can be instrumented to access the information where it resides.

Building data driven entitlement policies means having to write fewer policies. This has a direct impact on lowering the cost of policy management.

TYPICAL DEPLOYMENT SCENARIO

The following section will highlight how Oracle Entitlements Server can be deployed into an organization to solve a discrete fine grained authorization problem. The scenario will then be expanded to demonstrate how Oracle Entitlements Server can scale to handle the needs of a larger enterprise set of entitlement challenges.

Figure 4 below shows a Customer Care Portal which has been developed for our example company, Acme Bank. Acme has developed a J2EE based portal which is deployed across several data centers to handle the telephone and online chat service requests from their retail banking customers. The Customer Care Portal acts as a central “hub” application for both Customer Service Representatives (CSRs) and Branch personnel (Tellers, Branch Managers and Branch Executives). Because the Portal is used by two different types of users it must provide adequate controls so that CSRs do not access information that Branch management requires (for example customer credit risk scores and customer profitability reports). Similarly, there are some entitlements required for Tellers that CSRs should not have (for example home telephone number and social security number).

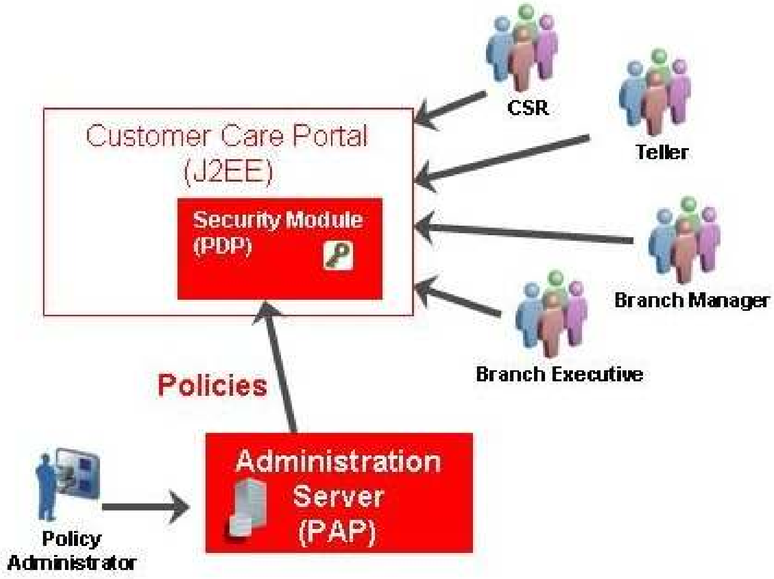


Figure 3 - the Acme Bank Customer Care Portal

Acme has deployed their Portal using the OES embedded Security Module. The Portal application makes direct in-memory calls to the PDP as pages are rendered to ensure that specific data columns or UI components are only provided to the users who can access them. Each Branch and Call Center is able to remotely connect to a specific Portal instance running in a data center.

Within Acme's IT organization a Policy Administrator works with the Portal development team to define and manage policies that meet the needs of Acme's Corporate Security group. Whenever a policy is changed by Corporate Security or the Portal application itself is versioned, the Policy Administrator makes and tests the appropriate changes in the PAP and propagates them to a Portal test environment. The verified policies are then propagated to the production servers in the data centers.

Let's suppose that Acme's IT organization receives requirements to secure additional applications; namely a .NET based account sign-up application and a J2EE based statement generator. The decision is made to deploy an additional set of PDPs into the data center to provide the authorization policies for these applications. A Web Services and an RMI based PDP are deployed to protect the .NET and J2EE applications respectively. Figure 5 below shows the new architecture.

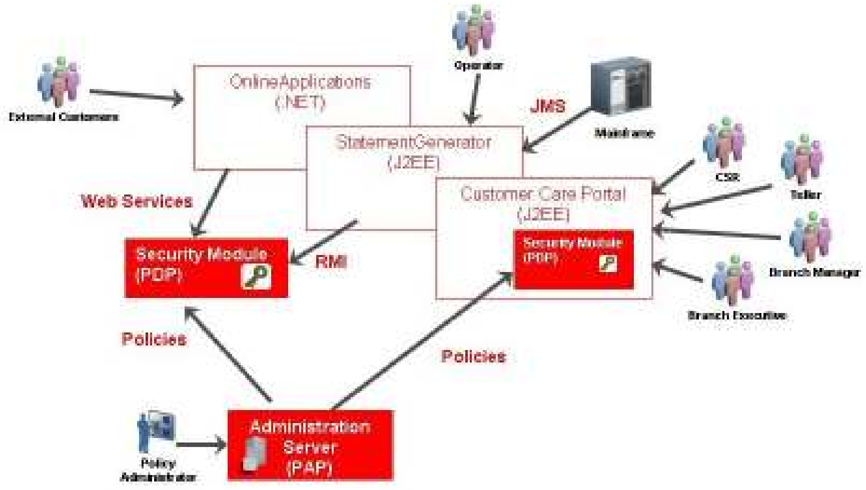


Figure 4 - Acme Bank utilizing OES across multiple applications

For the .NET application there is a need to selectively provide various product combinations and rates depending upon the geographic region a customer is based within. Policies are written to ensure that Acme doesn't accidentally offer a customer a particular account combination or statement of terms that is not legal in their state.

For the Statement Generator there is a need to make sure that only certain Operators can begin and view statement jobs based upon their level and region. Because of identity theft concerns and frequent employee turnover, Acme finds itself continuously modifying the access controls on their statement generator application. For example, an Operator with higher seniority is allowed to produce and view statements across a wider set of customers than a newly hired Operator. But for certain situations Acme may need to revoke or grant new abilities to Operators as they quit or exit a probationary period. This can be handled by either modifying the policies directly in OES or changing user attributes through their provisioning system.

As the OES deployment expands across multiple applications, the Policy Administrator will also configure each application's policies into their own OES Application to ensure proper isolation and simplify management. It might also be necessary to set up additional Administrator roles so that additional Policy Administrators can assist with the new workload.

The Retail Bank is quite successful in securing their applications and they gain the attention of Acme's CEO. The CEO asks that other lines of business work with Retail Banking to understand how they might gain similar benefits. It is eventually decided by the Acme IT organization to establish a cross-organizational Authorization Service. This Authorization Service will provide a common end point for multiple types of applications at Acme to leverage fine grained entitlements without having to write or host those entitlement decisions in their own environments. Figure 6 below shows the logical architecture for this approach.

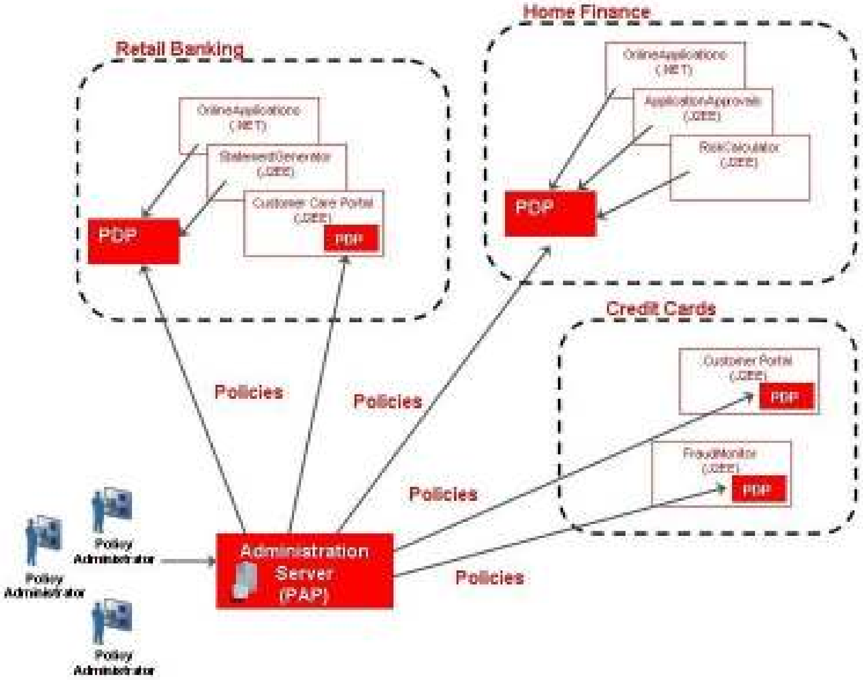


Figure 5 - Authorization as a shared service across multiple Acme Bank organizations

Home Finance and Credit Cards each establish PDPs in their architectures and adapt their existing and new applications to use a PDP for user authorizations. These PDPs contain only the policies pertinent to their respective client applications. Each line of business can choose the PDP deployment style (embedded or centralized) that meets their specific technology, performance and scalability requirements.

A centralized PAP is created to handle the needs of these different organization's policies. Using Oracle Entitlement Server's policy management facilities, the Policy Administration team sets up various Organizations to house the multiple Applications within each line of business. Separate Administration roles are established to allow different individuals in the new Policy Administration team to deal with the constant policy management tasks required by so many applications. Delegated Administration within the OES administration console ensure that a Home Finance administrator cannot view or modify policies for Retail Banking or Credit Cards. Similarly there are specific sub-administrative roles created so that certain people can only view policies while others have full edit and policy distribution responsibilities.

To summarize, Oracle Entitlements Server provides the ability to easily serve the needs of different sized deployments:

1. Single application, departmental use. Requires easy integration into existing runtimes and access for one Policy Administrator.
2. Multiple applications, single organizational use. Requires a more flexible set of deployment options without compromising functionality. Needs to support ability to separate each application's policies elegantly and simply by one or more Policy Administrators.
3. Multiple organizations, enterprise use. Requires highly scalable runtimes with diverse deployment choices. Must integrate with heterogeneous IT infrastructures. Strong delegated administration features required to support large Policy Administration teams serving different organizational needs and timelines.

Oracle Entitlements Server is an enabling technology to help evolve the notion of authorization from a *capability* to a *competence* to eventually a *shared service*.

SUMMARY

The requirements for application security have evolved. No longer can organizations be satisfied to just simply entitle access to critical business functions by authorizing whole applications. Competitive pressures dictate that firms do more with less and as roles become consolidated and applications are shared across wider sets of audiences the need for fine grained authorizations continues to grow.

Oracle Entitlements Server allows users to centrally define and manage policy for assigning roles, delegating administration and defining access. In addition, it allows users to protect not only application specific objects but also business objects such as accounts, contracts, reports and patient records.

Oracle Entitlements Server scales to meet the runtime needs of hundreds of applications while keeping a simplified administration model that cleanly separates massive policy stores for multiple organizations and numerous applications. Strong policy simulation features allow administrators to test policy before it is put into production without involving developers.

Built upon a framework designed for integration, OES works in diverse IT ecosystems involving multiple security and application infrastructures. OES is a key component to Oracle's industry leading Identity Management product family which provides a complete suite of solutions for enterprise Identity and Access management challenges.

Oracle Entitlements Server
September 2008
Author: Bill Dettelback
Contributing Authors: Eric Leach

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
oracle.com