Identity and Access Management for the Cloud
What You Need to Know About Managing Access to Your Clouds
One of the biggest challenges in information security is Identity and Access Management (IdM). How do you control who has access to what systems and technology within your enterprise?

Operating systems and applications all have different ways of managing this. As a result, the more applications you use, the more challenging it is to safely and securely manage your users.

This problem becomes even more difficult in the public cloud. The ability to control the technology is limited and it’s difficult to leverage tools such as single sign on/federation products.

This paper provides best-practices for overcoming the challenges involved with safely and securely managing your users within public clouds, as well as your private clouds.
Access Control and the Cloud

Access control can be divided into two categories—authentication and authorization.

**Authentication = Successfully and Accurately Identifying Users**

Authentication has become easier over the past few years, as more operating systems and applications now support technologies such as Active Directory (AD), LDAP and single sign on/federation. However, it still can be problematic, especially for password management.

**Authorization = Mapping the Actions that a User is Allowed to Take**

Examples of actions you might want to control include the ability to create other users, remove users, start or stop compute instances or make changes to different sorts of data.

Authorization presents a larger issue than authentication, because most applications aren’t leveraging directory services. Rather, they have their own built-in authorization systems. In the best cases, they can map roles to AD or LDAP groups.

Both authentication and authorization are problematic for large enterprises. Users can end up in multiple directory servers at the same time, and tracking and managing this situation becomes exponentially more difficult with each new server added to the environment.

**IdM in the Public Cloud**

While there are definitely issues with IdM in the enterprise, those issues pale by comparison to those in the public cloud. With enterprises subscribing to more and more cloud services, successfully managing authentication is becoming increasingly difficult, if not impossible, without some sort of centralized or federated system to manage users’ identities.

Very few cloud providers have support for third party authentication. Those that do are almost always found in the more traditional SaaS (Software as a Service) market, such as SalesForce.com. They are not typically found in the PaaS (Platform as a Service) or IaaS (Infrastructure as a Service) markets. There are even fewer options if you eliminate non-enterprise friendly options, such as OpenID. The providers that are left invariably only support authentication—not authorization.
Challenges of Authorization

It is even more difficult for enterprises to manage authorization in the cloud. One of the benefits of using a public cloud is that it exposes the inner workings of the infrastructure in ways that are usually limited to the staff at a physical datacenter. This is very powerful because it gives developers, and even regular users, the ability to self-service and provides users much more quickly with the resources they request. Unfortunately, most cloud providers don't limit who can do that. This means that once you grant a user access, they have access to all infrastructure and applications. Although this makes access easy, it can be disastrous. Suddenly, you have a company full of sysadmins with the equivalent of root access.

Authorization Capabilities Differ Among Cloud Providers

Even the cloud providers who do provide authorization tend to do so in a way that is different for each service. For example, Amazon's AWS has some very granular access control mechanisms for services, such as S3, but when it comes to their flagship product, EC2, it's an all-or-nothing scenario. What can be controlled with Access Control Rules varies dramatically from one cloud provider to another. This makes consistent application of authorization even more difficult.

Some companies have attempted to solve this lack of authorization by creating separate accounts with their cloud provider(s) for each project. This way only the relevant developers are allowed access to the cloud account. Anyone with access to the account could still do a lot of damage by mistake, but this limits the scope to just that one project. This works when only a few accounts are being managed, but becomes extremely cumbersome as the number grows.

One Cloud Provider—Hundreds of Accounts

Some companies have hundreds of accounts with the same cloud provider. Managing these accounts without authorization presents a number of challenges:

- **Time:** Managing the accounts individually becomes a full-time job. It's even more time-consuming if the provider doesn't have an option for consolidating all of the billing into one monthly statement.
- **Reliability:** In many companies, some of their developers have access to more than one account at a time. This is not only a huge headache to manage, but also constantly switching accounts increases the chances of someone making a mistake.
- **User Management:** Working with a large number of accounts makes it nearly impossible to correctly handle authorization when someone changes roles or leaves the company.

---

**Real-World Example:**

One company let all of their senior developers have access to an account that had been set up with their external cloud provider. One developer, as part of software testing, provisioned servers and terminated test instances over the course of several days. Later one afternoon, he accidentally terminated several key development databases, instead of his test instances. Fortunately for both the developer and the company, those databases were backed up. Although recovery was relatively simple, all development work was completely halted for several hours until the databases could be recreated.

---

Identity & Access Management

Copyright © 2012 enStratus Networks, Inc.
How Authorization Should Work

When evaluating software and services, it is vital to ensure that there is a way to implement a robust role-based access control system that allows administrators to create fine-grained Access Control Lists (ACLs). More specifically, it is necessary to control which users/groups can do what, for every available action. In the case of cloud, providers are not making this available today. This means that customers either need to build their own solution or engage with a third-party software product.

Using a Proxy Between You and the Cloud Provider

Essentially, what is needed is a proxy between the consumer and the cloud provider that allows users to create and maintain levels of authorization and monitoring far beyond what is currently available from most providers. Once this proxy is in place, it is very easy to log every action users are taking. Having a user action log contributes to overall security, potential recovery scenarios, and will help pass compliance audits as well.

Ways to Leverage Role-based Access Control

There are various methods for incorporating role-based access control into public and private clouds. Each of these can be deployed individually, or together, to help companies meet their security requirements.

- **User-based:** Limit login access to specific users. More specifically, limit admin access for all boxes to just the administrators groups. Further restrict users’ access to only the instance or group of instances that is necessary to perform their job functions. For example, application developers are only allowed access to systems in the application tier and DBAs are only allowed access to the SQL tier.

- **Filters:** A well-designed system will allow the creation of rules that can be applied not only to the zone or network, but also based on the meta-data related to the instance. For example, in development groups, it is often useful to restrict the ability to terminate instances to only the user who started that instance. This will help prevent a user from accidentally terminating a database they are not working on.

- **Granular Controls:** The most complete access control method is to provide a granular level of access to each component of the infrastructure. As an example, a company can grant a person in first tier support the ability to reboot instances but not start, stop or terminate them. Another example is allowing a person in QA access to development systems but not to production systems.

No matter which of these access control mechanisms are utilized, it is important to incorporate logging and alerting into the operations of public and private clouds. With the appropriately configured logging and alerting system in place, the operations team will be able to track all activity and be notified when specific actions fail.
Example of Using Role-based Access Control

Before:
Once you grant a user access, they have complete access to all systems and can perform any action.

After:
With role-based access control, users are only allowed access to certain systems, and can only perform certain actions based on their defined role.

No Controls/Complete Access

Granular Controls
What About Directory Services?

Access control in any application is important. As new applications are added to the portfolio, it is essential to minimize new parts that need to be managed.

User management should be approached with caution, especially with users who have administrative responsibilities or similar levels of access to critical data. Each additional system with uniquely created users increases the likelihood that access will not be removed or updated when a user changes roles. The number of places a user appears may scale linearly, but the complexity of management scales exponentially. As a result, one of the most common security breaches is a user whose access wasn’t properly adjusted or removed when he or she changed jobs or left the organization.

Using LDAP/AD for Role Maintenance

The chosen solution should allow synchronization with Active Directory/LDAP to eliminate this issue. This way not only can users be authenticated, but also groups can be mapped to roles within the proxy. As a result, maintenance of roles will be minimal, once the initial setup is complete, because all changes happen within the directory server. LDAP and Active Directory can be leveraged to dynamically create users within guest VMs.

Conclusion

Using enStratus to Improve IdM

enStratus enables customers to significantly improve your Identity and Access Management strategy with their cloud deployments. enStratus uses the existing tools from cloud providers and expands that coverage with fine-grained, role-based access control that is cloud independent. enStratus supports a variety of authentication methods to meet unique requirements of enterprises. Customers also gain auditing and logging of all user actions—something that cloud providers don't make available. And, existing directories, such as Active Directory and LDAP, can be leveraged to minimize the complexity of deployments and maintain fewer points of user management.
About the enStratus Enterprise Cloud Management Solution

enStratus provides cloud governance, automation and independence for enterprises. Think of enStratus as the enterprise console to the world of cloud computing.

The Enterprise Cloud Management Solution

enStratus™ helps you manage your cloud infrastructure. We support the provisioning, management and automation of applications in all leading public and private clouds.

We do this while retaining the ability for developers and application operators to choose the configuration management, monitoring and other operation tools that make the most sense for each application.

enStratus integrates into the leading operations tools and your internal systems to ensure your IT policies and procedures extend into the cloud.

enStratus is available as Software as a Service, or as on-premises software that enables you to control the cloud from within your own data centers. enStratus provides:

Governance - enStratus enables you to meet your governance needs with flexible access controls, logging, financial controls and integration into your internal management systems and access directories.

Automation - enStratus helps you meet the economic and operational advantages of cloud computing through a variety of automation tools including auto-provisioning, auto-scaling, automated backups, and more.

Independence - enStratus supports over 20 of the leading public clouds and private cloud platforms.

- **Public compute**: AWS EC2, Bluelock, CloudSigma, GoGrid, Joyent Cloud, Rackspace, SoftLayer, Tata InstaCompute, Terremark
- **Public storage**: AT&T Synaptic Storage, Azure, Google, AWS S3
- **Private compute**: Citrix CloudStack, Eucalyptus, Joyent Cloud, Nimbla, OpenStack Nova, vCloud Director
- **Private storage**: EMC Atmos, OpenStack Swift, Eucalyptus Walrus
- **Direct virtualization**: vSphere

Across these clouds, enStratus enables enterprises to leverage leading configuration management solutions, such as Chef and Puppet, as well as PaaS solutions, such as Cloud Foundry.

enStratus also provides Consulting Services to assist you in your migration into the cloud. We can help you design a deployment to meet your target SLAs and address issues such as scaling parameters, security and compliance.

To learn more, visit [http://www.enstratus.com](http://www.enstratus.com).