Enterprise Governance in Azure

The need for companies to apply sound governance practices for Azure is at an all-time high. The complexities and intricacies to do this in an effective manner can be quite burdensome, so this whitepaper will explore the different guidelines, tools, and recommendations to get you started on the right path.
# White Paper: Enterprise Governance in Azure

## Table of Contents

- Introduction .................................................................................................................................................................. 2
- Azure Subscription Models ................................................................................................................................. 2
- Azure Governance Wheel ........................................................................................................................................ 3
- Hierarchy Patterns .................................................................................................................................................. 4
  - Azure Management Sites ....................................................................................................................................... 6
- Using Subscriptions and Resource Groups as Building Blocks ........................................................................... 7
  - Subscriptions .......................................................................................................................................................... 7
  - Resource Groups .................................................................................................................................................. 7
  - When to Use Subscriptions or Resource Groups ............................................................................................... 8
  - Planning Resource Groups ................................................................................................................................... 9
- Azure AD and RBAC: Better Together ..................................................................................................................... 9
- Custom RBAC Roles .................................................................................................................................................. 11
- Enforcing Standards With ARM Policies ............................................................................................................... 11
- Resource Tags .......................................................................................................................................................... 12
- Protecting Resources With Resource Locks ........................................................................................................ 13
- Azure Automation .................................................................................................................................................... 13
- Azure Security Center ............................................................................................................................................. 16
- Azure Advisor .......................................................................................................................................................... 17
- Recommendations ..................................................................................................................................................... 17
- Need Help? ............................................................................................................................................................... 18
- About the Authors..................................................................................................................................................... 19
Introduction

As enterprise adoption of public and hybrid cloud options continues to increase, organizations of all sizes are considering Microsoft Azure for their cloud-based workloads. Microsoft has further distanced itself from competitors as a leader according to Gartner’s 2016 Magic Quadrant for Cloud IaaS and Magic Quadrant for Enterprise Application PaaS reports, and it is continuing to develop at a rapid pace. The need to apply sound governance practices within Azure is essential to maintaining proper access controls, cost management, and organization of resources. This applies to mid-market companies and large enterprises. Microsoft has offered tools and guidelines to achieve effective governance, and in this white paper, we will explore many of them and make best practice recommendations to get you started on the right path.

Azure Subscription Models

To begin, we will look at the different ways to enroll for Azure services. Microsoft offers multiple types of subscriptions with different governance features available to them:

- Free trial
- Azure Direct (Pay-As-You-Go and 12-Month Prepay)
- BizSpark
- Open Licensing
- MPN Benefits
- Visual Studio (MSDN) Subscriptions
- CSP Program
- Enterprise Agreement (EA) subscriptions (either through an existing EA or through a dedicated Server and Cloud Enrollment (SCE) for Azure only)

Many of the subscription models are targeted at individuals, small businesses, and developers. Core management and security features within Azure are common across all of these. However, additional features become available at the Enterprise Agreement level. These features are designed for mid-size and large enterprises. This is the model we will primarily focus on as we consider the needs of organizations with multiple application environments, diverse business units, and distributed geographic locations.
Credera’s Azure Governance Wheel defines foundational components that together help organizations meet IT governance requirements while enabling business leaders and developers to meet objectives quickly. The primary goal of this framework is to mitigate risk by enforcing standardized governance principles to protect critical shared resources, while allowing business stakeholders, developers, and operations/engineering teams the access they need to work autonomously. The Enterprise Agreement enrollment forms the outermost boundary of the model, typically as the organization’s master agreement for Microsoft Azure cloud enrollment. All other governance items, starting with the Subscription as the base unit, frame Azure resources for the deployment and management of enterprise technology solutions.

This model is highly flexible and customizable within each component to the specific requirements of each organization. It is likely that if you are considering this issue, governance policies for on-premises systems already exist and will serve as a good starting point for how to apply those principles within these features. It is also important to mention that this should apply whether the organization primarily runs “traditional IT” workloads (e.g., servers/VMs, monolithic applications, etc.) or “agile IT” workloads (e.g., microservices, containerized apps, etc.), from the perspective of preventing unauthorized access and ensuring budgets are adhered to. IT teams, risk and security management, and business leaders should work together to establish how the business will approach governance so there is consensus between all stakeholders. Doing this will greatly increase the chances of successfully leveraging Azure in a way that aligns with your organization’s goals.
Hierarchy Patterns

<table>
<thead>
<tr>
<th>FUNCTIONAL</th>
<th>BUSINESS UNIT</th>
<th>GEOGRAPHIC</th>
<th>RESPONSIBLE PARTIES</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enterprise Enrollment</td>
<td>Enterprise Enrollment</td>
<td>Enterprise Enrollment</td>
<td>Enterprise Administrators</td>
</tr>
<tr>
<td>Accounting</td>
<td>IT</td>
<td>US East</td>
<td>Department Administrators</td>
</tr>
<tr>
<td>Account Owner</td>
<td>Account Owner</td>
<td>US West</td>
<td>Account Administrators</td>
</tr>
<tr>
<td>App 1 Dev</td>
<td>App 1 Test</td>
<td>App 1</td>
<td>Service Administrators</td>
</tr>
<tr>
<td>Production Apps</td>
<td>App 2</td>
<td>App 2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>App 3</td>
<td>Project A</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Project B</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Project C</td>
<td></td>
</tr>
</tbody>
</table>

Azure enrollment hierarchies define how services are structured within an Enterprise Agreement. The Enterprise Portal allows customers to divide access to Azure resources associated with an Enterprise Agreement based on flexible hierarchies customizable to an organization’s unique needs. The hierarchy pattern should match an organization’s management and geographic structure so that the associated billing and resource access can be accurately accounted for. The three high-level patterns are Functional, Business Unit, and Geographic, using departments as an administrative construct for account groupings. Within each department, accounts can be assigned subscriptions, which create silos for billing and several key limits in Azure (e.g., number of VMs, storage accounts, etc.).

The Azure Governance Wheel will be used within subscriptions to apply the organization’s governance principles to each environment. Although subscriptions provide billing segregation and generally form a security boundary, it is possible to enable private communication between virtual networks in different subscriptions using VNet Peering.
# Enterprise Governance in Azure

<table>
<thead>
<tr>
<th>Unit</th>
<th>Description</th>
<th>Users</th>
<th>Roles/Responsibilities</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enterprise Enrollment</td>
<td>• The root level element of governance, tied to an Azure Enterprise Agreement&lt;br&gt;• May contain multiple Departments, Accounts and/or Subscriptions</td>
<td>Enterprise Administrators</td>
<td>• Full access to add/remove Departments, Accounts, and Subscriptions; lower-level Administrators applicable to the Enterprise Portal; and billing information&lt;br&gt;• Multiple Enterprise Administrators can exist</td>
</tr>
<tr>
<td>Department</td>
<td>• An administrative division of organizational hierarchy, based on the selected hierarchy pattern&lt;br&gt;• Owns one or more Accounts</td>
<td>Department Administrator</td>
<td>• Edit department level properties such as name and cost center&lt;br&gt;• Add/remove Accounts within the department</td>
</tr>
<tr>
<td>Account</td>
<td>• An individual or group associated with an email address, which may belong to either an Azure AD account or Microsoft Account&lt;br&gt;• Given a descriptive name within the Enterprise Portal for administrative purposes&lt;br&gt;• Holds one or more Azure Subscriptions</td>
<td>Account Administrator</td>
<td>• Creates and manages Subscriptions&lt;br&gt;• Also known as the Account Owner within the Enterprise Portal, this user will become the Account Administrator of its Subscriptions through the Azure Account Portal&lt;br&gt;• By default, this user also becomes the Service Administrator for associated Subscriptions</td>
</tr>
</tbody>
</table>
Enterprise enrollments and Enterprise Administrators are each managed in the Enterprise Portal. Here, you can also create read-only Administrators who have access to the account for billing or reporting purposes but cannot make any changes. Departments and Department Administrators are both managed in the Enterprise Portal as well. Accounts and Account Administrators (aka Account Owners) are created in the Enterprise Portal, but these users will usually manage subscriptions through the Account Portal.

### Azure Management Sites

<table>
<thead>
<tr>
<th>Unit</th>
<th>Description</th>
<th>Users</th>
<th>Roles/Responsibilities</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subscription</td>
<td>• A billing container that also serves as a security boundary and defines many Azure limits (e.g., number of cores and resources, etc.)&lt;br&gt;• Contains and organizes all resources and establishes governance principles over them</td>
<td>Service Administrator,&lt;br&gt;Co-Administrators,&lt;br&gt;Subscription Owners</td>
<td>• Manage resources and resource governance&lt;br&gt;• A single Service Administrator is defined through the Azure Account Portal&lt;br&gt;• Multiple Co-Administrators may be added in the Classic Portal, and multiple Subscription Owners may be defined in the Azure Portal&lt;br&gt;• The Service Administrator and Co-Administrators are automatically added as Subscription Owners in the Azure Portal</td>
</tr>
<tr>
<td>Resource Groups</td>
<td>• Logical containers within a subscription that contain related Azure resources sharing a common lifecycle&lt;br&gt;• Defines access for various roles and users to associated resources</td>
<td>Service Administrator,&lt;br&gt;Co-Administrators,&lt;br&gt;Subscription Owners, RBAC Users</td>
<td>• Abilities vary based on access assigned by Administrators/Owners&lt;br&gt;• Administrator/Owner level users have full access to all resources</td>
</tr>
</tbody>
</table>

Enterprise enrollments and Enterprise Administrators are each managed in the Enterprise Portal. Here, you can also create read-only Administrators who have access to the account for billing or reporting purposes but cannot make any changes. Departments and Department Administrators are both managed in the Enterprise Portal as well. Accounts and Account Administrators (aka Account Owners) are created in the Enterprise Portal, but these users will usually manage subscriptions through the Account Portal.

### Azure Management Sites

<table>
<thead>
<tr>
<th>Unit</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enterprise Portal</td>
<td><a href="https://ea.azure.com">https://ea.azure.com</a></td>
</tr>
<tr>
<td>Account Portal</td>
<td><a href="https://account.windowsazure.com">https://account.windowsazure.com</a></td>
</tr>
<tr>
<td>Classic Portal</td>
<td><a href="https://manage.windowsazure.com">https://manage.windowsazure.com</a></td>
</tr>
<tr>
<td>Azure Portal</td>
<td><a href="https://portal.azure.com">https://portal.azure.com</a></td>
</tr>
</tbody>
</table>
Using Subscriptions and Resource Groups as Building Blocks

Once you have modeled your organization’s hierarchy and mapped it to an Azure hierarchy pattern, you can consider where to draw the line between subscriptions and resource groups based on your governance needs.

Subscriptions

There can be multiple Azure subscriptions within an Enterprise Agreement. Subscriptions can be created through either the Enterprise Portal or the Account Portal. Through the Account Portal, the account administrator can rename the subscription or reassign the service administrator to another user. Also at the subscription level, if additional full admin-level users are required for any reason, service administrators can log on to the Classic Portal to add co-administrators, who cannot access the Account Portal but otherwise have full access within the subscription. In Azure, the ability to view resources is segregated at the subscription level. Billing for Azure consumption is also reported at the subscription level.

Resource Groups

Azure Resource Manager introduced the concept of resource groups as a more robust and flexible way to deploy and manage resources in Azure. Resource groups serve as containers for Azure resources within a subscription and are a flexible way to define resource lifecycles, policies, and access control. Resources can be allocated to resource groups using different deployment models, depending on the needs of the organization. Applications are commonly segregated into resource groups because they share a common lifecycle of being provisioned, updated, and decommissioned. The newer Azure Portal also allows role-based access control (RBAC) for more fine-grained access in Azure Resource Manager deployments. RBAC can be applied at the subscription level to be inherited by resource groups, or to individual resource groups for more granular delegation.
When to Use Subscriptions or Resource Groups

The decision of whether to divide resources into separate subscriptions or resource groups often takes billing and departmental chargebacks into account, as well as the overall management structure of the organization. A common functional hierarchy pattern includes pre-production environments assigned to application development teams within functional groups (e.g., accounting, HR, etc.) and production environments assigned to an IT engineering group. Business unit and geographic patterns may divide subscriptions between account holders within discrete business units, subsidiaries, or regions, with application owners and IT engineering teams assigned granular access within each subscription. Common reasons for dividing between separate subscriptions, rather than resource groups within a single subscription, are department groupings on billing statements, grouping resources into silos, separating access control between environments, and limiting access to services only available in the Classic Portal where RBAC does not apply.

From a cost management perspective, the Enterprise Portal offers significant insight into commitment usage and charges at department, account, and subscription levels. In addition to the reporting directly visible in the portal, you can download detailed usage reports with configurable date ranges, view your organization’s full Azure price sheet for creating plans and projections, and you can also import your consumption usage into Power BI to perform deeper analysis. Power BI allows you to drill down further to identify trends as well as build dashboards and shared reports. With these more advanced capabilities, you can get a better idea of how closely your usage is aligned to your budgeted commitment to make more informed business decisions, such as raising or lowering spending limits applied to various departments. These billing and reporting capabilities should influence planning of whether to place items into separate subscriptions.
Planning Resource Groups

Once you have taken billing and administrative factors into account to devise a subscription strategy, then the next step is to plan an approach to resource groups. When deploying resource groups, there are two primary approaches to their design:

- Resource groups that encompass all resources and core infrastructure components for an application deployment, including storage accounts, virtual networks, subnets, VMs, web apps, load balancers, etc.
- Centralized resource groups for core components such as storage accounts and virtual networks, with application resources such as VMs, web apps, load balancers, etc., set up in their own resource groups.

The centralized approach makes it easier to build and maintain hybrid network connectivity, protect data sovereignty, and enforce compliance requirements within the environment. Both "traditional IT" and "agile IT" workloads can be designed on top of the centralized resource group approach. “Traditional IT” workloads tend to be organized by overall application or system, when those resources share a common lifecycle. “Agile IT” workloads tend to be organized by layer (e.g., web tier, app tier, etc.) to support deployment lifecycles. With the centralized approach, application teams are empowered to develop and implement solutions efficiently while minimizing risk and optimizing the use of shared infrastructure. The alternative approach requires each solution to maintain its own core infrastructure and virtual networks, introducing additional management overhead and greater potential for conflicts.

Bearing these factors in mind, it is important to consider how they will apply to your organization, because ultimately any governance model will need to reflect the company’s strategic, compliance, and budgetary goals and requirements. Starting with sound governance principles to guide separation of resources into subscriptions and resource groups can prevent costly reorganization efforts later. After establishing this critical framework, the next step is to establish guidelines around RBAC, ARM policies, and auditing within your subscriptions.

Azure AD and RBAC: Better Together

The goal of security-minded organizations should be to give employees the exact permissions they need to do their jobs. Excessive permissions expose your cloud environment to risk from without and within. However, too few permissions mean that employees can’t get their work done efficiently. Properly utilizing Azure Active Directory (AD) and role-based access control (RBAC) within the Azure hierarchy facilitates segregation of duties and least privilege.

Each Azure subscription is associated with a single Azure AD instance. Azure AD more closely resembles Active Directory Lightweight Directory Services (AD LDS) than on-premises Active Directory in terms of directory service features and functionality, it is just more robust. Administrators can assign access rights using the Azure portal, Azure command-line tools, and Azure Management APIs to allow users and groups from that directory to manage resources in the Azure subscription. Leveraging Azure AD user accounts instead of Microsoft IDs is crucial because Microsoft IDs are privately owned by individuals and not managed by the organization. Therefore, they are not subject to password policies and cannot be quickly disabled in the event they are compromised.
To get started with Azure AD, there are a few different paths to consider. First, if your organization already uses Microsoft Office 365, then you already have an Azure AD tenant. You can use identities from this directory when establishing your Azure Enterprise enrollment, creating subscriptions, and assigning RBAC permissions. If you are starting from scratch, you can also create a new Azure AD tenant in the Azure Account Portal. In most circumstances, identity management and billing for existing Pay-As-You-Go Azure subscriptions can be centralized by associating the existing account with your Enterprise Agreement and transferring ownership to an Azure AD account through the Azure Enterprise Portal.

If your organization uses traditional on-premises Active Directory, Azure AD allows you to synchronize your on-premises identities, most commonly using Azure AD Connect. Azure AD Connect is the successor of DirSync, Azure AD Sync, and Forefront Identity Manager with the Azure Active Directory Connector configured. It synchronizes on-premises Active Directory users, groups, and other objects into an Azure AD instance. This process opens the doors for centralized identity management, federation, and single sign-on, which reduces administrative overhead and improves the user experience.

It is important to understand how Azure AD fits into the Azure hierarchy. Each subscription in Azure trusts only one Azure AD directory. However, within an Enterprise Agreement multiple subscriptions can be associated with the same Azure AD directory. This is important because it allows an organization to use a uniform set of users and groups across the enterprise. As you move down the hierarchy a resource group can belong to only one subscription, and each resource belongs to only one resource group. Administrators grant access by assigning the appropriate role to users and groups at a specified scope. The scope of a role assignment can be anywhere in this hierarchy: a subscription, a resource group, or a single resource. Bear in mind that access granted at a parent scope is inherited by its child scopes. For example, a user with access to a resource group can manage all the resources it contains, like websites, virtual machines, and subnets.
Leveraging RBAC empowers administrators to segregate duties within the organization and grant users only the amount of access they need to do their jobs. Instead of the old classic mode model that gave everybody unrestricted permissions to Azure subscriptions and resources, Resource Manager and RBAC can delegate specific tasks. For example, RBAC can allow one employee to manage virtual machines in a subscription, while another can only manage SQL databases within the same subscription.

- Azure RBAC comes in three general flavors:
  - Owner - Has full access to all resources including the right to delegate access to others.
  - Contributor - Can create and manage all types of Azure resources but can’t grant access to others.
  - Reader - Can view existing Azure resources.

Each subscription has over 30 built-in roles with granular resource-based permissions. Some examples of these are Network Contributor, SQL DB Contributor, and Virtual Machine Contributor.

### Custom RBAC Roles

However, if Azure’s built-in roles don’t meet your organization’s specific delegation needs, you can also create custom roles. Custom roles are stored in the Azure AD directory and can therefore be shared across all subscriptions within an Enterprise Agreement. Just like built-in roles, custom roles can be assigned to users, groups, and applications at the subscription, resource group, and resource level. Custom roles can be created using Azure PowerShell, the Azure Command-Line Interface (CLI), or via the REST API. The most straightforward way to create a custom role is to use the New-AzureRmRoleDefinition PowerShell cmdlet and a JSON input file. The easiest way to build the JSON input file is to use Get-AzureRmRoleDefinition to export an existing role to a JSON file and customize it for the new role you want to create. The custom properties that must be configured in the JSON file are these:

- Actions - A collection of operation strings that identify securable operations of Azure resource providers to which the role grants access.
- NotActions - Used if the set of operations that you wish to allow is more easily defined by excluding restricted operations (not a deny).
- AssignableScopes - Specifies the scopes (subscriptions, resource groups, or resources) within which the custom role is available for assignment and controls who can view, modify, and delete the role.

### Enforcing Standards With ARM Policies

Another means of applying controls to your organization’s Azure environment are Azure Resource Manager (ARM) custom policies. ARM policies are an explicit deny mechanism which can prevent users from breaking standards needed to manage your organization’s resources. Custom policies are commonly used to enforce naming conventions, control the types of resources that can be provisioned, require resource tags, and restrict the locations in which the resources can be provisioned.
The first step in applying a policy is creating a policy definition. Policy definitions describe the actions or resources that are specifically denied. Policy definitions are created using PowerShell (with or without JSON input files) that includes parameters, condition/logical operators, and a deny or audit effect (logging). For example, an administrator can create a policy that enforces a server naming convention. Policy definitions can be created by using the New-AzureRmPolicyDefinition PowerShell cmdlet and then applying it at a scope such as the subscription, resource group, or an individual resource. The example below requires a new virtual machine name to end in "-pvm" or the provisioning process will fail.

New-AzureRmPolicyDefinition -Name vmNamePolicyDefinition -Description "Policy to deny production RG VM creation if the name does not end in -pvm" -Policy '{
  "if": {
    "allOf": [
      {
        "field": "type",
        "equals": "Microsoft.Compute/virtualMachines"
      },
      {
        "not": {
          "allOf": [
            {
              "field": "name",
              "like": "*-pvm"
            }
          ]
        }
      ]
    },
    "then": {
      "effect": "deny"
    }
  }
}'

Like RBAC, inheritance principles apply to ARM policies. If a policy is applied to a subscription or resource group, it is applicable to all the resources within it.

**Resource Tags**

Tags are key/value pairs that you can apply to resources created in the ARM model for logical organization. An example of this would be a key of “Environment” and a value of “Development” on a development VM. Each resource is limited to 15 directly applied tags, which are visible across multiple resource groups for a wide context view. ARM policies can be used to enforce consistent application of tags, which is useful for ensuring that a department or application name is tagged on all resources in a resource group, for example. This can help with automating processes against logical groupings of resources as well as with billing chargebacks in environments with multiple departments or business units. Tags are available on billing reports in the Enterprise Portal for resource grouping and filtering purposes.
We recommend designing a tagging strategy to identify the metadata useful to the organization (e.g., Owner, BillTo, Environment, etc.) and requiring tags for each of the following:

- Resource groups
- Storage
- Virtual Machines
- Application Service Environments/web servers

Tags can be applied in the Azure Portal, using PowerShell, using the Azure CLI, or in resource templates. For example, to apply a tag to a resource without existing tags using PowerShell, run the following:

```
Set-AzureRmResourceTag @
{ ProjectName="Project A"; Environment="Production" } -ResourceName DatabaseA -ResourceGroupName AppDatabases-Prod-rg -ResourceType Microsoft.Sql/Servers/Databases
```

**Protecting Resources With Resource Locks**

When we discussed the centralized resource group model, one of the primary benefits was that shared infrastructure resources like Virtual Networks would be segregated into resource groups controlled by centralized IT engineering teams. Resource Locks represent another layer of protection to critical resources that could cause significant business disruption if modified or deleted. For example, an organization may want to prevent their ExpressRoute Gateway from being removed when their critical apps rely on hybrid network connectivity or prevent the deletion of a storage account that hosts critical data. Two types of locks are available in Azure Resource Manager, CanNotDelete and ReadOnly. They can be implemented in the Resource Manager portal, with PowerShell using the New-AzureRmResourceLock cmdlet, or via ARM templates. Locks can be applied at the subscription, resource group, or individual resource level, and are typically created by users with the Owner or User Access Administrator roles. Once they are added, they apply to all resources or policies within that scope for all users within the organization.

**Azure Automation**

Another standardization tool targeted toward operations is Azure Automation. Rather than performing common, repetitive tasks individually through the Azure Portal or PowerShell, you can use Azure Automation Runbooks to automate resource management and processes, reducing implementation time and the possibility of error. In addition, PowerShell Desired State Configuration (DSC) is available to help automate configuration management, ensuring standardized and enforced settings in deployed nodes.

Runbooks can take the form of text-based PowerShell or PowerShell Workflow, as well as graphical versions of each of those. A convenient way to get started with runbooks is with the Runbook Gallery. From there, you can take runbooks for common tasks such as shutting down or deploying VMs and customize them to your specific needs. While those are simple examples, runbooks are also scalable up to complex workflows requiring multiple steps.
Taking this example one step further, a PowerShell-based runbook that shuts down all VMs with a specified tag (e.g., Environment="Development", etc.) could take a script as shown below and run at a scheduled time of day to reduce consumption.

```
<#
.SYNOPSIS
   Stops all VMs in a resource group

.DESCRIPTION
   This runbook stops all VMs in a specified resource group. It applies only to VMs in Azure Resource Manager.

.PARAMETER AzureCredential
   Required with default of "AzureCredential". This specifies an Automation credential with an Azure AD account for the connection.
```
.PARAMETER AzureSubscriptionId
    Required with default of "AzureSubscriptionId". This specifies an Automation variable with the Azure subscription GUID.

.PARAMETER TagName
    Required. Identifies the tag name to target for VM shutdown.

.PARAMETER TagValue
    Required. Identifies the value of the named tag to target for VM shutdown.

.NOTES
    Sample Script
    #>

param (
    [Parameter(Mandatory=$false)]
    [String]  $AzureCredential = 'AzureCredential',

    [Parameter(Mandatory=$false)]
    [String] $AzureSubscriptionId = 'AzureSubscriptionId',

    [Parameter(Mandatory=$true)]
    [String] $TagName, 

    [Parameter(Mandatory=$true)]
    [String] $TagValue
)

$Cred = Get-AutomationPSCredential -Name $AzureCredential -ErrorAction Stop
$null = Login-AzureRmAccount -Credential $Cred -ErrorAction Stop -ErrorVariable err
if($err) {
    throw $err
}

$SubscriptionId = Get-AutomationVariable -Name $AzureSubscriptionId -ErrorAction Stop
$Resources = Find-AzureRmResource -TagName $TagName -TagValue "$TagValue"

$VMs = @()
foreach ($resource in $Resources) {
    if ($resource.ResourceType -eq "Microsoft.Compute/virtualMachines") {
        $VMs += $resource
    }
}

$VMs | ForEach-Object {
    Stop-AzureRmVM -Name $_.Name -ResourceGroupName $_.ResourceGroupName -Force -ErrorAction Continue
}
Azure Security Center

Azure Security Center represents a combination of best practice analysis and security policy management for all resources within an Azure subscription. This powerful and easy-to-use tool allows security teams and risk officers to prevent, detect, and respond to security threats as it automatically collects and analyzes security data from your Azure resources, the network, and partner solutions like anti-malware programs and firewalls. Azure Security Center applies advanced analytics, including machine learning and behavioral analysis, while leveraging global threat intelligence from Microsoft products and services, the Microsoft Digital Crimes Unit (DCU), the Microsoft Security Response Center (MSRC), and external feeds. Security governance can be applied broadly at the subscription level or narrowed down to specific, granular requirements applied to individual resources through policy definition. Azure Security Center analyzes resource security health based on those policies and uses this to provide insightful dashboards and alerting for events such as malware detection or malicious IP connection attempts. It can also tie in alerts to centralized logging systems such as Microsoft’s own Log Analytics service within the Operations Management Suite (OMS), which enables collection and correlation of events from on-premises and multiple cloud (e.g., Azure, AWS, etc.) sources.
Azure Advisor

One final governance tool is Azure Advisor, an automated consulting resource that can examine current configurations and make practical recommendations in the following areas:

- High availability (HA)
- Security
- Performance
- Cost

Even though many of these are based on common best practices, like placing virtual machines in availability sets for HA, security recommendations originate from Azure Security Center. While actual billing data is restricted to the Azure Enterprise Portal, Azure Advisor can look at usage and identify points of potential cost savings, such as underutilized VMs. Together, these provide a straightforward way to evaluate adherence to many governance principles within an Azure subscription.

Recommendations

Based on our experience with a diverse range of Azure enterprise clients, we have several best practice recommendations to offer. This is not exhaustive but should serve as a starting point to developing a sound governance approach aligned with the standards of your organization.

- Define your organizational hierarchy and map this to a pattern for the Azure Enterprise Portal such that billing, subscription management, and resource group design are aligned with this logical hierarchy.
- When establishing an Azure governance model, input from business leaders, security and risk management, and IT should all be considered.
- Use consistent, standardized naming conventions throughout the Azure Enterprise Portal and Azure resources, using ARM policies to enforce these standards within subscriptions (see Recommended Naming Conventions for Azure Resources).
- Sign up for Azure and assign access using Azure AD organizational accounts (i.e., work or school accounts) whenever possible.
  - If you have on-premises Active Directory, we recommend synchronizing this to Azure AD using Azure AD Connect.
  - If you have an Office 365 subscription, this includes an Azure AD tenant that we recommend using to sign up for Azure.
- Use centralized resource group design to minimize risk, protect critical core infrastructure, and ease management of cross-premises hybrid connectivity, while enabling application teams the access they need to achieve business objectives quickly.
- Leverage Azure Active Directory, Role-Based Access Control, and ARM policies within the Azure hierarchy to facilitate sound security practices such as segregation of duties and least privilege and enforce organizational standards across all cloud resources.
- Automate common tasks and virtual machine configuration whenever possible to ensure consistent baselines and reduce the possibility of error.
- Tag resources appropriately to facilitate access control, resource identification, and billing consolidation.
Need Help?

Are you interested in exploring Microsoft Azure but concerned about governance and how to make public cloud fit within your IT model? Credera has extensive experience in cloud infrastructure design and implementation. If you have questions or would like to discuss cloud and infrastructure solutions, contact us at findoutmore@credera.com.
About the Authors

Adrian Romo is a Senior Architect with Credera’s Cloud & Infrastructure Services Practice. He has 19 years of experience delivering IT infrastructure solutions to medium and large companies throughout the U.S. He attained his expertise in the consulting field supporting industries that include financial services, oil and gas, and healthcare and also in traditional operations roles managing multi-national networks. Mr. Romo is a subject matter expert in Active Directory, Exchange, Office 365, Microsoft Azure, VMware, Cisco networking, information security and data center management.

Bryan Sakowski is an Architect with Credera’s Cloud & Infrastructure Services Practice. He has over 8 years of experience designing and implementing technology infrastructure solutions. He gained systems and network engineering expertise working with a diverse range of industries, including financial services, commodities trading, refining, manufacturing, and retail. His proficiencies include Microsoft Azure, AWS, Active Directory, Office 365, PowerShell scripting, configuration management and automation, Exchange Server, SQL Server, Microsoft deployment services, VMware and Hyper-V virtualization, networking technologies and security, and Linux operating systems.
WE KEEP OUR PROMISES

Our clients hire us to own their toughest challenges. They retain us because we keep our promises.

GET IN TOUCH
Has this paper been helpful? Need additional information on our services? Please reach out and let us know how we can help:
findoutmore@credera.com