6 Policy Types for AWS Governance

VAST

IT without Limits™

POWERED BY CloudHealth

WHY DOES GOVERNANCE MATTER?

For organizations with large cloud environments, managing your assets and monitoring the rapid rate of change is an extremely time consuming task. The best way to govern these dynamic environments at scale is to implement automated policies that allow you to manage your environment in a relatively hands-off manner. These policies consist of a set of customizable rules, which give you a simple and effective way to eliminate noise, gain consistency and control, and reclaim time that can be spent on more strategic projects.

This eBook outlines several types of policies and examples that you must put in place to centralize governance across your Amazon Web Services (AWS) cloud environment.

1 ASSET & CONFIGURATION MANAGEMENT

In most on-premises environments, IT teams have a solid process for understanding asset and configuration management. Using tools such as ConfigurationManagement Databases (CMDB) and frameworks like the IT Information Library (ITIL), organizations are able to tightly control deployments and ensure standardization. In the cloud world where virtually any user can provision infrastructure in a few clicks with a credit card, these previously developed frameworks fall apart. In order to bring asset and configuration management back under control, advanced IT shops realized they needed to manage their environments by exception: by setting up rules for non-approved configurations and assets and then closely monitoring for them.

TAG COMPLIANCE

Tagging is an essential way to accurately categorize assets to their appropriate business groups. By using tags, you might assign labels for categories such as department (e.g. Engineering), product, environment (e.g. Production), application (e.g. HRIS), customer, or application role (e.g. Cassandra). Once a resource is tagged, AWS ensures usage associated with this resource is reported by this tag, allowing you to more easily associate costs to different business groups. Setting policies to identify assets that do not conform with your organization's internal tagging standards can help you stay on top of tag compliance. This might include untagged assets, mistagged assets, or misspelled tags.

IDENTIFY NON-CONFORMING ASSETS

In any organization, there are asset types and configurations that are not-preferred or outright not allowed. There are many reasons that an organization may wish to set a policy preventing certain assets from running: the organization may receive special discounts on certain instance types or have decided that certain instance types are too costly. An organization may want to prevent users from launching instances in certain regions, such as China, for example, for security reasons. Launching an older AMI type or OS could lead to security or interoperability challenges. Whether is it certain instance types, regions, AMI types, OS, or network types, it's critical that you can quickly identify these and take action to correct them.

SAMPLE TAG COMPLIENCE POLICIES

If any asset is missing the tag "Environment", send notification and execute a Lambda function to tag the asset.

If any Amazon EC2 Instance is untagged, alert its owner via email and stop the instance.

SAMPLE NONCONFORMING ASSET POLICIES

If any X1 instance type is launched, send through approval workflow first.

If any instance is launched in a non-conforming region, send through approval workflow first.

2 FINANCIAL MANAGEMENT

In order to keep costs under control, the best practice is to implement financial management policies that will help identify which lines of business, cost centers, or projects are accountable for driving up costs, and will alert you when costs unexpectedly spike. Financial management policies primarily focus on budget and cost trend monitoring.

BUDGET

Creating a budget is easy. Staying within that budget, not so much. In order to help departments and lines of business stay within the allocated budget, set a policy to alert budget owners when projected spend is greater than their set budget. You may also want to set additional policies that send alerts when overall spend is nearing budget limits. These policies can help departments track their actual spend compared to allocated budget and avoid unpleasant surprises at the end of the quarter or year.

COST TREND

Closely related to budget policies, cost trend policies look for unexpected cost increases. You can have greater control over your costs by benchmarking the cost of each AWS asset type month over month and identifying variances by business group. You can get extremely granular with this policy type — for example, alert me when Amazon CloudFront costs grow by more than 10% in a given month for my production assets, or take a broader approach —alert me when the total cost of any department increases by 20%.

SAMPLE BUDGET POLICIES

If projected month to date cloud spend is greater than 100% of budget, then send an email notification to the budget owner.

If my total spend reaches 85% of my budget for a given month, then send an email notification to the budget owner.

SAMPLE COST TREND POLICY

If total Amazon S3 costs increase by more than 10% in 1 day, alert me.

3 COST OPTIMIZATION

While financial management policies are critical for keeping pace with budgets and trends, they don't help you optimize and reduce costs on their own. To achieve this, you need to create policies that will help you proactively reduce and optimize costs in your cloud environment. According to Gartner, through 2020, 80% of organizations will overshoot their cloud IaaS budgets due to a lack of cost optimization approaches.

In AWS, one of the most effective ways to reduce costs is to purchase Reserved Instances (RIs), therefore many of the best practice policies for cost optimization focus on managing and automating the full lifecycle of RIs.

SAMPLE RI OPPORTUNITY POLICY

If an instance is running On-Demand for more than 550 hours over the course of a month, send an email alert, potential RI purchase.

IDENTIFY RESERVATION OPPORTUNITIES

Purchasing Reserved Instances is a great way to receive a significant discount on the hourly prices for instances. The ROI for a typical RI purchase for an EC2 Instance that is running 24/7/365 is around six or seven months. Given that, if an instance is running On-Demand more than 450 to 550 hours in a single month, it's a good candidate for purchasing a reservation. If you want to purchase more conservatively, you may choose to watch the instance behavior over a few months before purchasing the reservation. Of course, rightsizing your instances must always be done before making any RI purchase. terminated if deemed nonessential. Take a snapshot, or point-in-time copy, of the asset before terminating or stopping it to ensure you can recover it if the asset is needed again.

1 Source: Gartner, How to Identify Solutions for Managing Costs in Public Cloud IaaS, Craig Lowery,

3 COST OPTIMIZATION

MODIFY RESERVED INSTANCES

It’s not enough to simply purchase Reserved Instances, you must also keep them optimized. AWS allows customers to modify Standard RIs in the following ways:

- Switching between Regional and an Availability Zone scope
- Switching between Availability Zones for reservations scoped to a specific zone within the same region
- Switching between Classic EC2 and Virtual Private Cloud
- Altering the instance size within the same family.

You can make modifications through the AWS console, directly through the API, or automatically with a cloud service management solution. Because modifications are free, mature organizations continuously look for modificationsto maximize their ROI from RI purchases.

SAMPLE RI MODIFICATION POLICY

If potential RI reallocation savings exceed $10, then modify Reserved Instance.

4 PERFORMANCE MANAGEMENT

Understanding and monitoring performance in your AWS environment is critical, but not always easy. It's important to consider core utilization metrics such as CPU, memory, disk, and network in/out, which can be gathered from Amazon CloudWatch and performance monitoring tools. Using these trended metrics over time, you can gain information on whether instances and volumes are sized properly and performing as expected. It is a best practice to have pre-defined thresholds for what constitutes normal behavior for your infrastructure. For example, if CPU is less than 20% then you deem that asset as underutilized. Underutilized assets should be downgraded for cost efficiency, while overutilized assets should be upgraded to avoid performance headaches.

SAMPLE UNDERUTILIZED INSTANCE POLICIES

If any RDS Instance has average read throughput AND write throughput AND swap usage less than 35% for over two weeks, then send an email notification - potential downgrade.

If any io1 volume type average reads are less than 10,000 AND average writes are less than 10,000 for 1 week, then send an email notification, potential downgrade.

RIGHTSIZING UNDERUTILIZED ASSETS

It's common for developers to spin up new instances that are substantially larger than necessary. This may be intentional to give themselves extra headroom during production or accidental because they don't know the performance requirements of the new workload yet. Over-provisioning Amazon EC2 or RDS Instances, EBS volumes, or S3 object storage, can lead to exponentially higher costs, so it's critical that you set up policies that will notify you when an asset is over-provisioned.

For example, the critical factors to consider with EBS volumes are capacity, IOPS, and throughput. AWS offers several types of EBS volumes, from Cold HDDs to Provisioned IOPS SSDs, each with their own set of pricing and performance. You can find candidates for downgrading by analyzing the read/writes on all volumes. If a volume barely has any read/writes, it is either attached to a zombie instance or the volume is unnecessary. Many organizations find they've deployed General Purpose SSD or Provisioned IOPS SSD volumes that barely have any read/writes for a long period of time. They can be downgraded to Throughput Optimized HDD or even Cold HDD volumes in order to reduce costs.

4 PERFORMANCE MANAGEMENT

RIGHTSIZING OVERUTILIZED ASSETS

Rightsizing is not only important for identifying cost savings in underutilized assets, but it's also critical for identifying assets that are overutilized and could be impacting performance and causing a poor experience for the user. Over-utilization policies will look similar to the underutilization policies, except with different thresholds and more inclusive clauses: instead of "AND" clauses, use "OR" to find any area of resource constraint.

SAMPLE OVERUTILIZED INSTANCE POLICIES

If any gp2 volume type throughput averages more than 150 MiB/s for 1 week, then send an email notification, potential upgrade.

If any S3 Infrequent Access Object was retrieved more than three times in the last 30 days, send an email alert, potential migration to S3.

5 OPERATIONAL GOVERNANCE

Automating basic operational tasks is one of the best ways to reclaim time to focus on more strategic business projects. Whether you are automating the detection and elimination of zombies or unused infrastructure, flagging older instance types, or even scheduling environments to turn off and on again, these policies can yield significant time savings.

IDENTIFY AND TERMINATE ZOMBIE INFRASTRUCTURE

Zombie assets are infrastructure components running in your cloud, but are not in use. For example, this could be an EC2 Instance once used for a project that has since ended, but the instance was never turned off. Zombie assets can also come in the form of an unattached ElasticIP, an empty Elastic Load Balancer (ELB), or an idle Relational Database Service (RDS) instance. No matter the cause, AWS will charge for them as long as these assets are in a running state. They must be isolated, evaluated, and immediately terminated if deemed nonessential. It is recommended that you take a snapshot, or point-in-time copy, of the asset before terminating or stopping it to ensure you can recover it, if the asset is needed again.

INSTANCE SCHEDULING

For instances running 24/7, AWS will bill for 672 – 744 hours per instance, depending on the month. If an instance is turned off between 5pm and 9am on weekdays and stopped weekends and holidays, then total billable hours per month would range from 152 – 184 hours per instance, saving you 488 – 592 instance hours per month. The most cost efficient environments dynamically stop and start instances based on a set schedule. Each cluster of instances can be treated a different way. These types of lights on/lights off policies can often be even more cost effective than RI purchases, so it’s crucial to analyze where this type of policy can be implemented.

SAMPLE ZOMBIE TERMINATION POLICIES

If an EBS volume is unattached for 1 week, then trigger Snapshot, delete volume, send an email notification.

If a Snapshot is older than 2 months, send an email notification, delete.

If an ElasticIP is unattached for two weeks, then send an email notification, and release ElasticIP.

SAMPLE INSTANCE SCHEDULING POLICY

Stop ‘Development’ EC2 Instances at 7pm on Friday, start ‘Development’ at 6am on Monday.

6 SECURITY & INCIDENT MANAGEMENT

In a rapidly evolving cloud environment, it is important to keep up with changes that might impact your security posture. The best way to do this is with automated security policies, which can monitor for issues and flag them before they become catastrophic. There are many different types of security policies to set across access control, network security, application security, data security, log management, and resiliency.

SAMPLE ACCESS CONTROL POLICY

If any accounts have root account API access, then send an email notification, and execute a Lambda function to revoke user access.

ACCESS CONTROL

Cloud security starts with users and access controls. Without proper access controls and identity management, users can intentionally or unintentionally create security flaws with serious implications. Set policies that will validate that you have properly and securely configured access to your cloud and to help you stay ahead of breaches by monitoring for leading indicators such as:

- Misconfigured users (i.e., users not in a group)
- Users with too broad of a span of control (i.e., root accounts enabled for API access, too broad privileges, etc.)
- Users with vulnerable accounts (i.e., not compliant with password policies, IAM user access keys in need of rotation, multi-factor authentication disabled, etc.)
- Inactive users (i.e., IAM user with access keys that are not being used, etc.)

While it's always best to proactively catch security vulnerabilities before they are exploited, it's prudent to also monitor for events that could turn into security incidents, or lagging indicators, like:

- Suspicious activity (i.e., a large volume of instances are launched outside of normal usage patterns, new IP address for login on IAM user accounts, etc.)
- Changes to security groups or users (i.e., new IAM group or user recently created or changed, Root account recently used, etc.)

[2] Source: Gartner, Clouds Are Secure: Are You Using Them Securely? Jay Heiser, 31 January 2018.

6 SECURITY & INCIDENT MANAGEMENT

AUDIT TRAIL

Without proper audit trails and logs in place, it can be extremely challenging to identify security incidents, policy violations, fraudulent activity, and operational problems. In short, root cause analysis and troubleshooting are greatly helped by log management. AWS CloudTrail is one the primary source of logs, as the service provides a record of all API calls on your account, including the identity of the API caller, the time of the API call, the source IP address of the API caller, the request parameters, and the response elements returned by the AWS service. Other AWS services, such as EBS and S3 generate access logs as well. Audit trail policies will ensure that logs are collected, stored securely for the proper amount of time, and are available for analysis when needed.

SAMPLE AUDIT TRAIL POLICIES

If CloudTrail is not enabled for all regions, send an email alert, and enable CloudTrail for all regions.

If CloudTrail S3 bucket is publicly accessible, send an email alert, restrict access to bucket, and encrypt bucket.

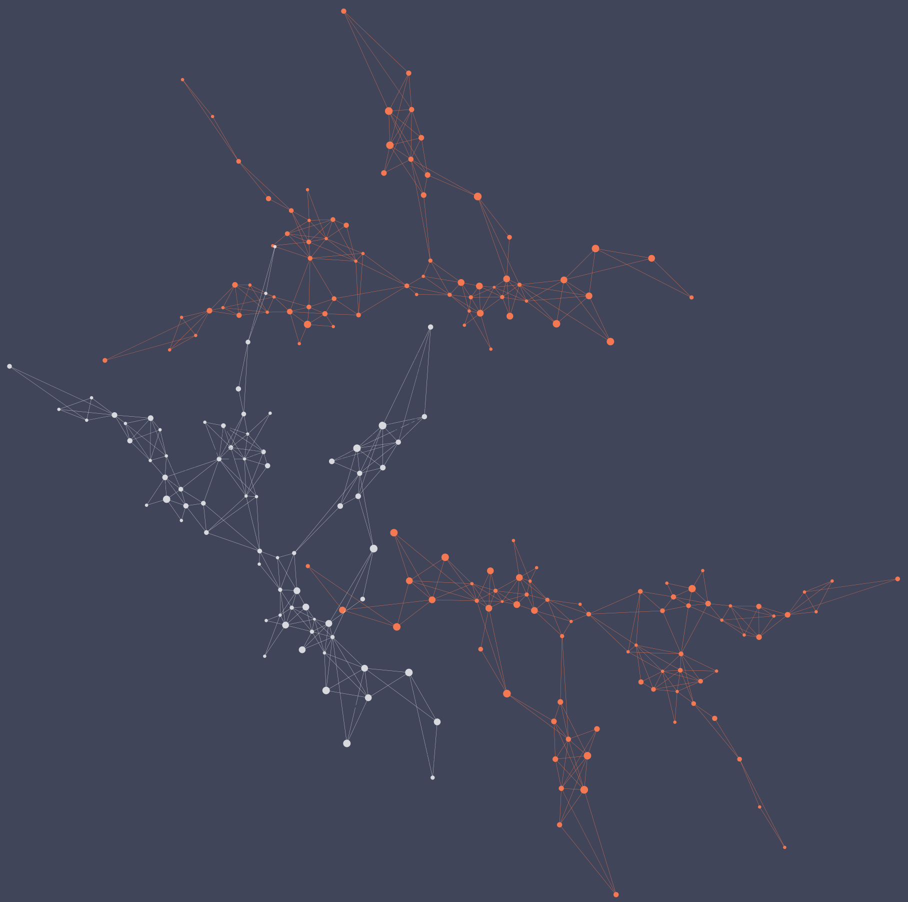
CONCLUSION

It’s important to remember that these best practice policies are not meant to be one-time activities, but ongoing processes. Because of the dynamic and ever changing nature of the cloud, governance policies should ideally be automated so they can take place continuously. It’s also critical to periodically revisit policies to ensure they still make sense for your organization. Learn how you can implement and automate governance policies across your AWS environment.

Learn more by visiting **www.vastITservices.com**

POWERED BY CloudHealth

VAST View™ is a trademark of VAST IT Services.



VAST
IT without Limits™