AMS Accelerate User Guide

AMS Accelerate Concepts and Procedures

Version September 16 2021

AMS Accelerate User Guide: AMS Accelerate Concepts and Procedures

Table of Contents

What is AWS Managed Services?

Topics

Welcome to AWS Managed Services (AMS), infrastructure operations management for Amazon Web Services (AWS). AMS provides a range of operational services to help you achieve operational excellence on AWS. Whether you're just getting started in the cloud, looking to augment your current team, or need a long-term operational solution, AMS can help you meet your operational goals in the cloud. Leveraging AWS services and a library of automations, configurations, and run books, we provide an end-to-end operational solution for both new and existing AWS environments.

AMS operates infrastructure for some of the world’s largest enterprises. The service leverages a suite of native AWS services and features to provide a comprehensive set of infrastructure management capabilities. Within these AWS services AMS creates and maintains curated sets of monitoring controls, detection guardrails, automations, and runbooks to operate infrastructure in a compliant and secure way.

AMS operations plans

AWS Managed Services is available with two operations plans: AMS Accelerate and AMS Advanced. An operations plan offers a specific set of features and has differing levels of service, technical capabilities, requirements, price, and restrictions. Our operations plans give you the flexibility to select the right-sized operational capabilities for each of your AWS workloads. This section outlines the capabilities and differences, as well as the responsibilities, features, and benefits associated with each plan, so that you can understand which operations plan is best for your accounts.

For a detailed feature comparison of the two operations plans, see AWS Managed Services Features.

AMS Accelerate operations plan

AMS Accelerate is the AMS operations plan that helps you operate the day-to-day infrastructure management of your new or existing AWS environment. AMS Accelerate provides operational services, such as monitoring, incident management, and security. AMS Accelerate also offers an optional patch add-on for EC2-based workloads that require regular patching.

With AMS Accelerate, you decide which AWS accounts you want AMS Accelerate to operate, the AWS Regions you want AMS Accelerate to operate in, the add-ons you require, and the service-level agreements (SLAs) you need. For more details, see Using the AMS Accelerate operations plan and Service Description.

AMS Advanced operations plan

AMS Advanced provides full-lifecycle services to provision, run, and support your infrastructure. In addition to the operational services provided by AMS Accelerate, AMS Advanced also includes additional services, such as landing zone management, infrastructure changes and provisioning, access management, and endpoint security.

AMS Advanced deploys a landing zone to which you migrate your AWS workloads and receive AMS operational services. Our managed multi-account landing zones are pre-configured with the infrastructure to facilitate authentication, security, networking, and logging.

AMS Advanced also includes a change and access management system that protects your workloads by preventing unauthorized access or the implementation of risky changes to your AWS infrastructure. Customers need to create a Request for Change (RFC) using our Change Management system to implement most changes in your AMS Advanced accounts. You create RFCs from a library of automated changes that are pre-vetted by our security and operations teams or request manual changes that are reviewed and implemented by our operations team if they are deemed both safe and supported by AMS Advanced.

Using the AMS Accelerate operations plan

AMS Accelerate is the AMS operations plan that can operate AWS infrastructure supporting workloads. Whether your workloads are already in an AWS account or you're planning to migrate new ones, you can benefit from AMS Accelerate operational services such as monitoring and alerting, incident management, security management, and backup management, without going through a new migration, experiencing downtime, or changing how you use AWS. AMS Accelerate also offers an optional patch add-on for EC2 based workloads that require regular patching.

With AMS Accelerate you have the freedom to use, configure, and deploy all AWS services natively, or with your preferred tools. You can continue using your existing access and change mechanisms while AMS consistently applies proven practices that help scale your team, optimize costs, increase security and efficiency, and improve resiliency.

While AMS Accelerate can simplify your operations, you remain responsible for application development, deployment, test and tuning, and management. AMS Accelerate only makes changes in your account as a result of incidents, alarms, remediation, and some service requests. AMS Accelerate doesn't provision resources in the account on your behalf. AMS Accelerate provides troubleshooting assistance for infrastructure issues that impact applications, but AMS Accelerate doesn't access or validate your application configurations without your knowledge and approval. AMS Accelerate services and changes are provided directly in the AWS console and APIs, so you continue to leverage your existing accounts with AWS and available AWS marketplace solutions. AMS Accelerate doesn't modify code in your infrastructure-as-code templates (for example, AWS CloudFormation templates), but can guide your teams on which changes are required to follow best operational and security practices.

Key terms

- *AMS Advanced*: The services described in the "Service Description" section of the AMS Advanced Documentation. See Service Description.
- *AMS Advanced Accounts*: AWS accounts that at all times meet all requirements in the AMS Advanced Onboarding Requirements. For information on AMS Advanced benefits, case studies, and to contact a sales person, see AWS Managed Services.
- *AMS Accelerate Accounts*: AWS accounts that at all times meet all requirements in the AMS Accelerate Onboarding Requirements. See Getting Started with AMS Accelerate.
- *AWS Managed Services*: AMS and or AMS Accelerate.
- *AWS Managed Services Accounts*: the AMS Accounts and or AMS Accelerate Accounts.
- *Customer-Requested Configuration*: Any software, services or other configurations that are not identified in:
  - Accelerate: Supported Configurations or AMS Accelerate; Service Description.

  - AMS Advanced: Supported Configurations or AMS Advanced; Service Description.
- *Incident Communication*: AMS communicates an Incident to you or you request an Incident with AMS via an Incident created in Support Center for AMS Accelerate and in the AMS Console for AMS. The AMS Accelerate Console provides a summary of Incidents and Service Requests on the Dashboard and links to Support Center for details.
- *Managed Environment*: The AMS Advanced accounts and or the AMS Accelerate accounts operated by AMS.
- *Billing start date*: AWS Managed Services accounts are activated once you have granted access to AMS to a compatible account and AMS Activation notification occurs as defined in the AWS Managed Services Documentation. If the activation of the AWS Managed Services accounts, Add-on Service Request, or Account tier Service Request is received by AWS on or prior to the 20th day of the month, then the change will be effective as of the first day of the calendar month following the AMS Activation notification or such Service Request. If the activation or Service Request is received by AWS after the 20th day of the month, then the change will be effective as of the first day of the second calendar month following AMS Activation notification or such Service Request.

  AMS Activation Notification to the customer occurs when:

  1. Customer grants access to a compatible AWS account and hands it over to AWS Managed Services.
  2. AWS Managed Services designs and builds the AWS Managed Services Account.
- *Service Termination Date*: The last day of the calendar month in which the Customer provides the AMS Account Service Termination Request, or the last day of the calendar month following the end of the requisite notice period; provided that, if the Customer provides the AMS Account Service Termination Request after the 20th day of the calendar month, the Service Termination Date will be the last day of the calendar month following the calendar month that such AMS Account Service Termination Request was provided.
- *Provision of AWS Managed Services*: AWS will make available to Customer and Customer may access and use AWS Managed Services for each AWS Managed Services Account from the Service Commencement Date.
- *Termination for specified AWS Managed Services Accounts*: Customer may terminate the AWS Managed Services for a specified AWS Managed Services Account for any reason by providing AWS notice via a Service Request ("AMS Account Termination Request").
- *Effect of Termination of specified AWS Managed Services Accounts.*: On the Service Termination Date, AWS will (i) hand over the controls of all AMS Accounts or the specified AMS Account, as applicable, to Customer, or (ii) the parties will remove the AWS Identity and Access Management roles that give AWS access from all AMS Accelerate Accounts or the specified AMS Accelerate Account, as applicable.

Incident management terms:

- *Event*: A change in your AMS environment.
- *Alert*: Whenever an event from a supported AWS service exceeds a threshold and triggers an alarm, an alert is created and notice is sent to your contacts list. Additionally, an incident is created in your Incident list.
- *Incident*: An unplanned interruption or performance degradation of your AMS environment or AWS Managed Services that results in an impact as reported by AWS Managed Services or you.
- *Problem*: A shared underlying root cause of one or more incidents.
- *Incident Resolution* or *Resolve an Incident*:
  - AMS has restored all unavailable AMS services or resources pertaining to that incident to an available state, or
  - AMS has determined that unavailable stacks or resources cannot be restored to an available state, or
  - AMS has initiated an infrastructure restore authorized by you.
- *Incident Response Time*: The difference in time between when you create an incident, and when AMS provides an initial response by way of the console, email, service center, or telephone.

- *Incident Resolution Time*: The difference in time between when either AMS or you creates an incident, and when the incident is resolved.
- *Incident Priority*: How incidents are prioritized by AMS, or by you, as either Low, Medium, or High.
  - *Low*: A non-critical problem with your AMS service.
  - *Medium*: An AWS service within your managed environment is available but is not performing as intended (per the applicable service description).
  - *High*: Either (1) the AMS Console, or one or more AMS APIs within your managed environment are unavailable; or (2) one or more AMS stacks or resources within your managed environment are unavailable and the unavailability prevents your application from performing its function.

  AMS may re-categorize incidents in accordance with the above guidelines.
- *Infrastructure Restore*: Re-deploying existing stacks, based on templates of impacted stacks, and initiating a data restore based on the last known restore point, unless otherwise specified by you, when incident resolution is not possible.

Infrastructure terms:

- *Managed production environment*: A customer account where the customer's production applications reside.
- *Managed non-production environment*: A customer account that only contains non-production applications, such as applications for development and testing.
- *AMS stack*: A group of one or more AWS resources that are managed by AMS as a single unit.
- *Immutable infrastructure*: An infrastructure maintenance model typical for EC2 Auto Scaling groups (ASGs) where updated infrastructure components, (in AWS, the AMI) are replaced for every deployment, rather than being updated in-place. The advantages to immutable infrastructure is that all components stay in a synchronous state since they are always generated from the same base. Immutability is independent of any tool or workflow for building the AMI.
- *Mutable infrastructure*: An infrastructure maintenance model typical for stacks that are not EC2 Auto Scaling groups and contain a single instance or just a few instances. This model most closely represents traditional, hardware-based, system deployment where a system is deployed at the beginning of its life cycle and then updates are layered onto that system over time. Any updates to the system are applied to the instances individually, and may incur system downtime (depending on the stack configuration) due to application or system restarts.
- *Security groups*: Virtual firewalls for your instance to control inbound and outbound traffic. Security groups act at the instance level, not the subnet level. Therefore, each instance in a subnet in your VPC could have a different set of security groups assigned to it.
- *Service Level Agreements (SLAs)*: Part of AMS contracts with you that define the level of expected service.
- SLA *Unavailable* and *Unavailability*:
  - An API request submitted by you that results in an error .
  - A Console request submitted by you that results in a 5xx HTTP response (the server is incapable of performing the request).
  - Any of the AWS service offerings that constitute stacks or resources in your AMS-managed infrastructure are in a state of "Service Disruption" as shown in the Service Health Dashboard.
  - Unavailability resulting directly or indirectly from an AMS exclusion is not considered in determining eligibility for service credits. Services are considered available unless they meet the criteria for being unavailable.
- *Service Level Objectives (SLOs)*: Part of AMS contracts with you that define specific service goals for AMS services.

Patching terms:

- *Mandatory patches*: Critical security updates to address issues that could compromise the security state of your environment or account. A "Critical Security update" is a security update rated as "Critical" by the vendor of an AMS-supported operating system.
- *Patches announced versus released*: Patches are generally announced and released on a schedule. Emergent patches are announced when the need for the patch has been discovered and, usually soon after, the patch is released.
- *Patch add-on*: Tag-based patching for AMS instances that leverages AWS Systems Manager (SSM) functionality so you can tag instances and have those instances patched using a baseline and a window that you configure.
- *Patch methods*:
  - *In-place patching*: Patching that is done by changing existing instances.
  - *AMI replacement patching*: Patching that is done by changing the AMI reference parameter of an existing EC2 Auto Scaling group launch configuration.
- *Patch provider* (OS vendors, third party): Patches are provided by the vendor or governing body of the application.
- *Patch Types*:
  - *Critical Security Update (CSU)*: A security update rated as "Critical" by the vendor of a supported operating system.
  - *Important Update (IU)*: A security update rated as "Important" or a non-security update rated as "Critical" by the vendor of a supported operating system.
  - *Other Update (OU)*: An update by the vendor of a supported operating system that is not a CSU or an IU.
- *Supported patches*: AMS supports operating system level patches. Upgrades are released by the vendor to fix security vulnerabilities or other bugs or to improve performance. For a list of currently supported OSs, see Support Configurations.

Security terms:

- *Detective Controls*: A library of AMS-created or enabled monitors that provide ongoing oversight of customer managed environments and workloads for configurations that do not align with security, operational, or customer controls, and take action by notifying owners, proactively modifying, or terminating resources.

Service Request terms:

- *Service request*: A request by you for an action that you want AMS to take on your behalf.
- *Alert notification*: A notice posted by AMS to your **Service requests** list page when an AMS alert is triggered. The contact configured for your account is also notified by the configured method (for example, email). If you have contact tags on your instances/resources, and have provided consent to your cloud service delivery manager (CSDM) for tag-based notifications, the contact information (key value) in the tag is also notified for automated AMS alerts.
- *Service notification*: A notice from AMS that is posted to your **Service request** list page, usually to notify you of upcoming patching.

Miscellaneous terms:

- *AWS Managed Services Interface*: For AMS: The AWS Managed Services Advanced Console, AMS CM API, and AWS Support API. For AMS Accelerate: The AWS Support Console and AWS Support API.
- *Customer satisfaction (CSAT)*: AMS CSAT is informed with deep analytics including Case Correspondence Ratings on every case or correspondence when given, quarterly surveys, and so forth.
- *DevOps*: DevOps is a development methodology that strongly advocates automation and monitoring at all steps. DevOps aims at shorter development cycles, increased deployment frequency, and more

dependable releases by bringing together the traditionally-separate functions of development and operations over a foundation of automation. When developers can manage operations, and operations informs development, issues and problems are more quickly discovered and solved, and business objectives are more readily achieved.

- *ITIL*: Information Technology Infrastructure Library (called ITIL) is an ITSM framework designed to standardize the lifecycle of IT services. ITIL is arranged in five stages that cover the IT service lifecycle: service strategy, service design, service transition, service operation, and service improvement.
- *IT service management (ITSM)*: A set of practices that align IT services with the needs of your business.
- *Managed Monitoring Services (MMS)*: AMS operates its own monitoring system, Managed Monitoring Service (MMS), that consumes AWS Health events and aggregates AWS CloudWatch data, and data from other AWS services, notifying AMS operators (online 24x7) of any alarms created through an Amazon Simple Notification Service (Amazon SNS) topic.
- *Namespace*: When you create IAM policies or work with Amazon Resource Names (ARNs), you identify an AWS service by using a namespace. You use namespaces when identifying actions and resources.

Service description

AMS Accelerate is a service for managing operations of your AWS infrastructure.

AMS Accelerate features

AMS Accelerate offers the following features:

Incident management

AMS Accelerate proactively detects and responds to incidents and assists your team in resolving issues. You can reach out to AMS Accelerate operations engineers 24x7 using AWS Support Center, with response time SLAs depending on the level of response you selected for your account.

Monitoring

Accounts enrolled in AMS Accelerate are configured with a baseline deployment of CloudWatch events and alarms that have been optimized to reduce noise and to identify a possible upcoming incident. After receiving the alerts, the AMS team uses automated remediations, people, and processes, to bring the resources back to a healthy state and engage with your teams when appropriate to provide insights into learnings on the behavior and how to prevent it. If remediation fails, AMS starts the incident management process. You can change the baselines by updating the default configuration file.

Security management

AMS Accelerate maintains a library of AWS Config Rules and remediation actions to ensure that all your accounts comply with industry standards for security and operational integrity. AWS Config Rules test every configuration change among your all resources. If a change violates any rule conditions, AMS reports its findings, and allows you to remediate violations automatically or by request, according to the severity of the violation. AWS Config Rules facilitate compliance with standards set by: the Center for Internet Security (CIS), the National Institute of Standards and Technology (NIST) Cloud Security Framework (CSF), the Health Insurance Portability and Accountability Act (HIPAA), and the Payment Card Industry (PCI) Data Security Standard (DSS).

In addition, AMS Accelerate leverages Amazon GuardDuty to identify potentially unauthorized or malicious activity in your AWS environment. GuardDuty findings are monitored 24x7 by AMS. AMS collaborates with you to understand the impact of the findings and remediations based on best practice recommendations. AMS also supports Amazon Macie to protect your sensitive data such as personal health information (PHI), personally identifiable information (PII), and financial data.

Patch management

For an AWS account with the patch add-on, AWS Managed Services applies and installs vendor updates to EC2 instances for supported operating systems during your chosen maintenance windows. AMS creates a snapshot of the instance prior to patching, monitors the patch installation, and notifies you of the outcome. If the patch fails, AMS investigates the failure, tries to remediate it, or restores the instance as needed. AMS provides reports of patch compliance coverage and advises you of the recommended course of action for your business.

Backup management

AWS Managed Services creates, monitors, and stores snapshots for AWS services supported by AWS Backup. You define the backup schedules, frequency, and retention period by creating AWS Backup plans while onboarding accounts and applications. You associate the plans to resources. AMS tracks all backup jobs, and, when a backup job fails, alerts our team to run a remediation. AMS leverages your snapshots to perform restoration actions during incidents, if needed. AMS provides you with a backup coverage report and a backup status report.

Designated experts

AMS Accelerate also designates a Cloud Service Delivery Manager (CSDM) and a Cloud Architect (CA) to partner with your organization and drive operational and security excellence. Your CSDM and CA provide you guidance during and after configuration and onboarding AMS Accelerate, deliver a monthly report of your operational metrics, and work with you to identify potential cost savings using tools such as AWS Cost Explorer, Cost and Usage Reports, and Trusted Advisor.

Operations tools

AMS Accelerate can provide ongoing operations for your workload's infrastructure in AWS. Our patch, backup, monitoring, and incident management services depend on having resources tagged, and the AWS Systems Manager (SSM) and CloudWatch agents installed and configured on your EC2 instances with an IAM instance profile that authorizes them to interact with the SSM and CloudWatch services. AMS Accelerate provides tools like Resource Tagger to help you tag your resources based on rules, and automated instance configuration to install the required agents in your EC2 instances. If you're following immutable infrastructure practices, you can complete the prerequisites directly in the console or infrastructure-as-code templates.

All AMS Accelerate customers start with incident management, monitoring, security monitoring, log recording, prerequisite tools, backup management, and reporting capabilities. You can add AMS Patch add-on at an additional price.

Logging and Reporting

AWS Managed Services aggregates and stores logs generated as a result of operations in CloudWatch, CloudTrail, and VPC Flow Logs. Logging from AMS helps in faster incident resolution and system audits. AMS Accelerate also provides you with a monthly service report that summarizes key performance metrics of AMS, including an executive summary and insights, operational metrics, managed resources, AMS service level agreement (SLA) adherence, and financial metrics around spending, savings, and cost optimization. Reports are delivered by the AMS cloud service delivery manager (CSDM) designated to you.

Supported configurations

These are the configurations AMS Accelerate supports:

- Supported language: English.
- Supported AWS Regions: See the AWS Regions supported by AWS Managed Services in the AWS Regional Services webpage
- Supported AWS operating systems:
  - Amazon Linux 2 and Amazon Linux
  - CentOS 7.x, CentOS 6.5-6.10
  - Oracle Linux 7.5 and later minor versions
  - Red Hat Enterprise Linux (RHEL) 8.x, 7.x, 6.5-6.10
  - SUSE Linux Enterprise Server 15 SP0, SP1 and SAP specific versions, SUSE Linux Enterprise Server 12 SP4, SP5 and SAP specific versions.
  - Microsoft Windows Server 2019, 2016, 2012 R2, 2012

    Note

    Operating systems (OSs) that are outside of the general support period of the operating system manufacturer ("end of support" (EOS)) have an increased security risk and are considered as supported configuration, only if 1) you have extended support with the OS vendor that allows you to receive updates, or 2) any instances using EOS OS follow the security controls as specified by AMS in the user guide, or 3) you comply with any other compensating security controls required by AMS.

Supported services

AWS Managed Services provides operational management support services for the following AWS services. Each AWS service is distinct and as a result, AMS's level of operational management support varies depending on the nature and characteristics of the underlying AWS service. If you request that AWS Managed Services provide services for any software or service that is not expressly identified as supported in the following list, any AWS Managed Services provided for such customer-requested configurations will be treated as a "Beta Service" under the Service Terms.

- Incidents: All AWS services
- Service request: All AWS services
- Patching: EC2
- Backups and Restoration: EC2, RDS, EBS, Storage Gateway, Dynamo DB, Aurora, EFS
- Monitoring: EC2, RDS, Aurora, RedShift, ElasticSearch, NAT gateway (a Network Address Translation (NAT) service), Site-to-Site VPN, Elastic Load Balancer, Application Load Balancer, Personal Health Dashboard. To learn more about what AMS Accelerate is monitoring as part of a service, see Alerts from baseline monitoring in AMS (p. 122)
- Security controls: AWS Account, GuardDuty, Macie, API Gateway, Certificate Manager (ACM), Config, CloudTrail, CloudWatch, CodeBuild, Database Migration Service, DynamoDB, EC2, Elastic Block Store (EBS), Elastic File System (EFS), Elastic Load Balancing, ElastiCache, ElasticSearch, EMR, Identity and Access Management (IAM), Key Management Service, KMS), Lambda, Redshift, Relational Database Service (RDS), S3, SageMaker, Secrets Manager, Simple Notification Service (SNS), Systems Manager Agent (SSM), VPC (Security group, Volume, Elastic IP, VPN connection, Internet gateways), VPC Flow Logs.

Roles and responsibilities

AMS Accelerate manages your AWS infrastructure. The following table provides an overview of the roles and responsibilities for you and AMS Accelerate for activities in the lifecycle of an application running within the managed environment.

- **R** stands for Responsible party that does the work to achieve the task.

- **C** stands for Consulted; a party whose opinions are sought, typically as subject matter experts; and with whom there is bilateral communication.
- **I** stands for Informed; a party who is informed on progress, often only on completion of the task.

Activity	Customer	Accelerate
Application lifecycle		
Application development	R	I
Application infrastructure requirements, analysis, and design	R	I
Application deployment	R	I
AWS resource deployment	R	I
Application monitoring	R	I
Application testing/optimization	R	I
Troubleshoot and resolve application issues	R	I
Troubleshoot and resolve problems	R	I
AWS infrastructure monitoring	C	R
Incident response for AWS network issues	C	R
Incident response for AWS resource issues	C	R
Managed Account onboarding		
Grant access to the AWS Managed Account for the AMS team and tools	R	C
Implement changes in the account or environment to allow the deployment of tools in the account. For example, changes in Service Control Policies (SCPs)	R	C
Install SSM agents in EC2 instances	R	C
Install and configure tooling required to provide AMS services. For example, CloudWatch agents, scripts for patching, alarms, logs, and others	I	R
Manage access and identity lifecycle for AMS engineers	I	R
Collect all required inputs to configure AMS services. For example, patch maintenance windows duration, schedule and targets	R	I
Request the configuration of AMS services and provide all required inputs	R	I
Request the configuration of AMS services and provide all required inputs	R	I
Configure AMS services as requested by the customer. For example, patch maintenance windows, resource tagger, and alarm manager	C	R
Manage the lifecycle of users and their permissions, for local directory services, used to access AWS accounts and instances	R	I
Recommend reserved instances optimization	I	R
Patch management		

Activity	Customer	Accelerate
Configure maintenance windows and other parameters (for example maintenance window duration) for patching	R	I
Tag instances to associate them with custom maintenance windows and patch baselines	R	I
Define custom patch baselines to filter and exclude specific patches	R	I
Monitor for applicable updates to supported OS and software preinstalled with supported OS for EC2 instances	I	R
Report for missing updates to supported OS and maintenance window coverage	I	R
Take snapshots of instances before applying updates	I	R
Apply updates to EC2 instances per customer configuration	I	R
Investigate failed updates to EC2 instances	C	R
Update AMIs and stacks for Auto-Scaling groups (ASGs)	R	C
Patch development software (.NET, PHP, Perl, Python)	R	I
Patch and monitor middleware applications (for example, BizTalk, JBoss, WebSphere).	R	I
Patch and monitor custom and third-party applications	R	I
Backup		
Specify backup schedules and target resources	R	I
Perform backups per plan	I	R
Investigate failed backup jobs	I	R
Report for backup jobs status and backup coverage	I	R
Validate backups	R	I
Request backup restoration for resources supported AWS services resources as part of incident management	R	I
Perform backup restoration activities for resources of supported AWS services	I	R
Restore affected custom or third-party applications	R	I
Networking		
Provisioning and configuration of Managed Account VPCs, IGWs, Direct connect, and other AWS networking Services	R	I
Configure and operate AWS Security Groups/NAT/NACL inside the Managed account	R	I
Networking configuration and implementation within customer network (for example DirectConnect)	R	I

Activity	Customer	Accelerate
Networking configuration and implementation within AWS network	R	I
Monitor defined by AMS for network security, including security groups	I	R
Network-level logging configuration and management (VPC flow logs, ELB access log, and others)	I	R
Logging		
Record all application change logs	R	I
Record AWS infrastructure change logs	I	R
Enable and aggregate AWS audit trail	I	R
Aggregate logs from AWS resources	I	R
Monitoring and Remediation		
Configure monitored resources	R	I
Configure alarm manager and alarm thresholds	R	C
Deploy AMS CloudWatch baseline metrics and alarms per customer configuration	I	R
Monitor supported AWS resources using baseline CloudWatch metrics and alarms	I	R
Investigate alerts from AWS resources	C	R
Remediate alerts based on defined configuration, or create an incident	I	R
Define, monitor, and investigate customer-specific monitors	R	I
Investigate alerts from application monitoring	R	C
Security Architecture		
Review AMS resources and code for security issues and potential threats	I	R
Implement security controls in AMS resources and code to mitigate security risks	I	R
Enable supported AWS services for security management of the account and its AWS resources	I	R
Manage privileged credentials for account and OS access for AMS engineers	I	R
Security Risk Management		
Monitor supported AWS services for security management, like GuardDuty and Macie	I	R
Define and create AMS-defined Config Rules to detect if AWS resources comply with Center for Internet Security (CIS) and NIST security best practices.	I	R
Monitor AMS-defined Config Rules	I	R

Activity	Customer	Accelerate
Report conformance status of Config Rules	I	R
Define a list of required Config Rules and remediate them	I	R
Evaluate the impact of remediating AMS-defined Config Rules	R	I
Request remediation of AMS-defined Config Rules in the AWS account	R	I
Track resources exempted from AMS-defined Config Rules	R	I
Remediate supported AMS-defined Config Rules in the AWS account	C	R
Remediate non-supported AMS-defined Config Rules in the AWS account	R	I
Define, monitor, and investigate customer-specific Config Rules	R	I
Security monitoring and response		
Configure supported security management AWS services for alerting, alerts correlation, noise reduction, and additional rules	I	R
Monitor supported AWS services for security alerts	I	R
Install, update, and maintain endpoint security tools	R	I
Monitor for malware on instances using endpoint security	R	I
Incident Management		
Notify about incidents detected by AMS in AWS resources	I	R
Notify about incidents in AWS resources	R	I
Notify about incidents for AWS resources based on monitoring	I	R
Handle application performance issues and outages	R	I
Categorize incident priority	I	R
Provide incident response	I	R
Provide incident resolution or infrastructure restore for resources with available backups	C	R
Problem Management		
Correlate incidents to identify problems	I	R
Perform root cause analysis (RCA) for problems	I	R
Remediate problems	I	R
Identify and remediate application problems	R	I
Service Management		
Request information using service requests	R	I
Reply to service requests	I	R
Provide cost-optimization recommendations	I	R

Activity	Customer	Accelerate
Prepare and deliver monthly service report	I	R
Change Management		
Change management processes and tooling for provisioning and updating resources in the managed environment	R	I
Maintenance of application change calendar	R	I
Notice of upcoming maintenance Window	R	I
Record changes made by AMS Operations	I	R

Contact and escalation

You have a designated cloud service delivery manager (CSDM) who provides advisory assistance across AMS Accelerate, and has a detailed understanding of your use case and technology architecture for the managed environment. CSDMs work with account managers, technical account managers, AWS Managed Services cloud architects (CAs), and AWS solution architects (SAs), as applicable, to help launch new projects and give best practice recommendations throughout the software development and operations processes. The CSDM is the primary point of contact for AMS. Key responsibilities of your CSDM are:

- Organize and lead monthly service review meetings with customers.
- Provide details on security, software updates for environment and opportunities for optimization.
- Champion your requirements including feature requests for AMS Accelerate.
- Respond to and resolve billing and service reporting requests.
- Provide insights for financial and capacity optimization recommendations.

Contact hours

You can contact AMS Accelerate for different reasons at different times.

Feature	AMS Accelerate		AMS Advanced	
Service request	Monday to Friday: 08:00–18:00, local business hours	24/7	Monday to Friday: 08:00–18:00, local business hours	24/7
Incident management (P1)	24/7			
Incident management (P2-P3)	Monday to Friday: 08:00–18:00, local business hours	24/7	Monday to Friday: 08:00–18:00, local business hours	24/7
Backup and recovery	24/7			
Patch management	24/7			

Feature	AMS Accelerate	AMS Advanced
Monitoring and alerting	24/7	
Cloud service delivery manager (CSDM)	Monday to Friday: 08:00–17:00, local business hours	

Business hours

AMS contact hours

Feature	AMS Accelerate		AMS Advanced	
Service request	Monday to Friday from 8am- 6pm, local business hours	24 x 7	Monday to Friday from 8am- 6pm, local business hours	24 x 7
Incident management (P1)	24 x 7			
Incident management (P2-P3)	Monday to Friday from 8am- 6pm, local business hours	24 x 7	Monday to Friday from 8am- 6pm, local business hours	24 x 7
Backup and recovery	24 x 7			
Patch management	24 x 7			
Monitoring and alerting	24 x 7			
Cloud service delivery manager (CSDM)	Monday to Friday, 9am to 5pm, local business hours			

Escalation path

For escalation of issues or problems, reach out to your CSDM for the best path forward.

Getting Started with AWS Managed Services

If you do not have AWS Managed Services (AMS) operating an account already, start by contacting an Amazon Web Services (AWS) sales representative using our AWS Managed Services- Contact Sales page.

After you sign up for an AMS, the AMS Accelerate team guides you through the following onboarding process for each one of your AWS accounts.

Review the feature set here: AWS Managed Services Features

Account onboarding process

Onboarding an account into AMS Accelerate has four stages. Each stage requires inputs from you.

Account discovery
Account-level onboarding
Instance-level onboarding
Operations configuration

1. **Account discovery (p.   )** uses tools to discover what your needs are. You are assisted by AMS Accelerate personnel.
2. **Account-level onboarding (p.   )** asks you to accept the terms and conditions; and create an onboarding role for AMS Accelerate cloud architects (CAs), who will assist you with setting a security baseline, and resolving issues as needed.
3. **Instance-level onboarding (p.   )** configures your EC2 instances and enabling baseline monitoring.
4. **Operations configuration (p.   )** tailors AMS services to your needs.

Accelerate onboarding prerequisites

These are the technical dependencies that Accelerate components rely on.

AMS Accelerate VPC endpoints

A VPC endpoint enables private connections between your VPC and supported AWS services and VPC endpoint services powered by AWS. If you need to filter outbound internet connectivity, configure the following VPC service endpoints to ensure that AMS Accelerate has connectivity with it's service dependencies.

```
com.amazonaws.region.logs
com.amazonaws.region.monitoring
com.amazonaws.region.ec2
com.amazonaws.region.ec2messages
com.amazonaws.region.ssm
com.amazonaws.region.ssmmessages
com.amazonaws.region.s3
com.amazonaws.Region.events
```

For information about AWS VPC endpoints, see VPC endpoints.

Outbound internet connectivity

If your VPC outbound connectivity to the internet is restricted, the following must be opened for Accelerate to function properly:

EC2 patching endpoints. The following sections from the JSON file in the ZIP apply to Accelerate:

- WindowsPatching
- RedHatPatching
- AmazonLinuxPatching
- EPELRepository

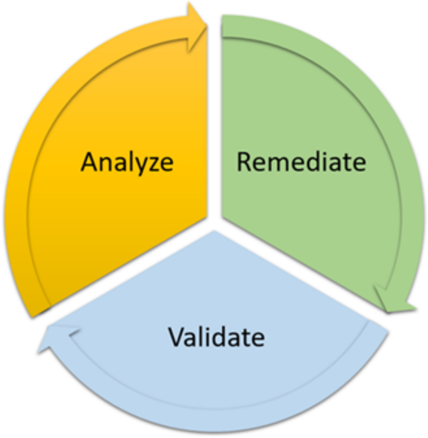
AWS Systems Manager on EC2 instances

You must install the AWS Systems Manager Agent (SSM Agent) on all of the EC2 instances you want AMS to manage. For details, see Instance-level onboarding (p. 24).

Account discovery

Account discovery is the stage when AMS Accelerate works with you to assess the current state of your account, evaluate that our service is a good fit for your account, and identify any major technical blockers for supporting your environment. During the Account Discovery stage, AMS Accelerate does not provide any operational services.

To assist with the analysis and discovery of an account, we ask you to run a pre-built script in the **AwsAccountDiscoveryCli** command line interface that generates a comprehensive picture of your account and resources, focusing on the services you are using. You can control which part of the report to share with AMS Accelerate and then AMS Accelerate starts an iterative process to remove technical blockers, if any, before moving to the Account-Level onboarding stage.



Important
The **AwsAccountDiscoveryCli** performs read-only calls and does not transmit data to AMS Accelerate during collection. Data is stored locally on the machine that runs the commands. AMS recommends that you review the collected data with your security team to determine whether or not you can share it with AMS for further analysis. Then, work with your AMS account team to determine the process for sharing your approved data with AMS.
For the latest changes, and to know if you need to update, see the Account Discovery Changelog zip file.

Using the AwsAccountDiscoveryCli for discovery

AwsAccountDiscoveryCli is a command line interface to discover AWS resources in a given account. You can perform discovery from a variety of platforms, including AWS CloudShell, Linux, and Windows.

Prerequisites for AwsAccountDiscoveryCli

Before you can use **AwsAccountDiscoveryCli**, you'll also need:

- (Recommended) Access to the AWS CloudShell with read-only permissions (for more information, see Managing AWS CloudShell access and usage with IAM policies). You’ll need the following AMS policies:
  - **arn:aws:iam::aws:policy/ReadOnlyAccess**
  - **arn:aws:iam::aws:policy/AWSCodeArtifactReadOnlyAccess**
  - **arn:aws:iam::aws:policy/AWSCloudShellFullAccess**

  Create and attach the following policy to the IAM entity (user, role) you are using to initiate the discovery:

```
{
  "Version":"2012-10-17",
  "Statement":[
    {
      "Effect":"Allow",
      "Action":"support:DescribeSeverityLevels",
      "Resource":"*"
    }
  ]
}
```

OR

- The latest AWS CLI configured with read-only permissions, see Configuring the AWS CLI. AMS recommends the following AWS Managed policies:
  - **arn:aws:iam::aws:policy/ReadOnlyAccess**
  - **arn:aws:iam::aws:policy/AWSCodeArtifactReadOnlyAccess**

  Create and attach the following policy to the IAM entity (user, role) you are using to initiate the discovery:

```
{
  "Version":"2012-10-17",
  "Statement":[
    {
      "Effect":"Allow",
      "Action":"support:DescribeSeverityLevels",
      "Resource":"*"
    }
  ]
}
```

- Python 3.6 and later (only required if not using AWS CloudShell. We highly recommend using CloudShell for discovery):
  - Linux: Python downloads
  - Mac OS X: Python Releases for Mac OS X
  - Windows: Python Releases for Windows

Important
If the account you want to discover is part of an AWS organization, in order to collect organization-level information, **AwsAccountDiscoveryCli** must be called from the organization's management account or by a member account that is a delegated administrator for an AWS service, otherwise the organization-level information will not be collected. To learn more about these concepts, see AWS Organizations terminology and concepts

Discovery of an AWS account from AWS CloudShell (recommended)

AWS CloudShell is a browser-based shell that makes it easy to securely manage, explore, and interact with your AWS resources. AWS CloudShell is pre-authenticated with your console credentials when you log in. Common development and operations tools are pre-installed, so no local installation or configuration is required. With AWS CloudShell, you can quickly run scripts with the AWS Command Line Interface (AWS CLI), experiment with AWS service APIs using the AWS SDKs, or use a range of other tools to be productive. You can use AWS CloudShell right from your browser at no additional cost.

Note
You can use the CloudShell AWS console from any other, or the closest, AWS Region where it is available, to perform resource discovery for all other AWS Regions. For example, to perform discovery in the Singapore region, open a CloudShell in the "US West(Oregon) us-west-2" AWS Region in the AWS Console and follow the instructions given next.

To use **AwsAccountDiscoveryCli** with AWS CloudShell:

1. From any page or AWS Region in the AWS Management Console, open the AWS CloudShell to run the account discovery script shown next. Ensure that you are logged into the AWS Management Console with the correct level of permissions, see Prerequisites for AwsAccountDiscoveryCli (p. 17).

   Note
   Do not change the "--domain-owner 354220221581" and " --region us-west-2" parts shown; copy the script as-is.

   ```
   python3 -m venv awsdiscovery
   source ~/awsdiscovery/bin/activate
   pip install pip --upgrade
   aws codeartifact login --tool pip \
     --repository AwsAccountDiscoveryCli \
     --domain aws-account-discovery-cli \
     --domain-owner 354220221581 \
     --region us-west-2
   pip install awsaccountdiscoverycli
   ```

2. Verify that the installation completed successfully:

   ```
   awsdiscover --version
   ```

3. Start the collection for the current account:

   ```
   awsdiscover
   ```

4. Discovery takes more time on large accounts. Once finished, compress the output folder to download the report:

```
tar -czvf DiscoveryReports.tar.gz /home/cloudshell-user/AwsAccountDiscoveryReports/
```

5. Select **Actions** in the top right corner, then choose **Download file**.
6. For the **Individual file path**, specify the following path and then choose **Download**.

   /home/cloudshell-user/DiscoveryReports.tar.gz

Discovery of an AWS account from Linux

Note
We highly recommend using Discovery of an AWS account from AWS CloudShell, as previously described. Use this method only if AWS CloudShell cannot be used due to IAM permission issues.

To discover an AWS account using Linux, follow these steps:

1. To authenticate to the AMS CodeArtifact repository and install **AwsAccountDiscoveryCli** on Linux, use the following script:

```
python3 -m venv awsdiscovery
source ~/awsdiscovery/bin/activate

pip install pip --upgrade

aws codeartifact login --tool pip --repository AwsAccountDiscoveryCli \
 --domain aws-account-discovery-cli --domain-owner 354220221581 --region us-west-2

pip install awsaccountdiscoverycli
```

2. Verify that the installation completed successfully (by checking that a version exists):

```
awsdiscover --version
```

3. Start the collection for the current account:

```
awsdiscover
```

4. Discovery takes more time on large accounts. After it's finished, you'll see the output location printed in the tool's output on screen.

Discovery of an AWS account from Windows

Note
We highly recommend using Discovery of an AWS account from AWS CloudShell, as previously described. Use this method only if AWS CloudShell cannot be used due to IAM permission issues.

To discover an AWS account using Windows, follow these steps:

1. To authenticate to the AMS CodeArtifact repository and install **AwsAccountDiscoveryCli** on Windows, use the following script:

```
py -m venv env
.\env\Scripts\activate

pip install pip --upgrade
```

```
$CODEARTIFACT_TOKEN = Get-CAAuthorizationToken -Domain aws-account-discovery-cli `
                        -DomainOwner 354220221581 `
                        -Region us-west-2
pip config set global.index-url https://aws:
$($CODEARTIFACT_TOKEN.AuthorizationToken)@aws-account-discovery-
cli-354220221581.d.codeartifact.us-west-2.amazonaws.com/pypi/AwsAccountDiscoveryCli/
simple/

pip install awsaccountdiscoverycli
```

2. Verify that the installation completed successfully:

```
awsdiscover --version
```

3. Start the collection for the current account:

```
awsdiscover
```

4. Discovery takes more time on large accounts. Once finished, you'll see the output location.

Discovery of an AWS account from a Hub account

This process involves creating a read-only IAM role in each account you want to discover (child accounts) from a central (hub) account.

Hub account - IAM configuration

Attach the following policy to the IAM entity (user, role) you are using to initiate the discovery:

```
{
 "Version": "2012-10-17",
 "Statement": [{
  "Effect": "Allow",
  "Action": [
   "sts:AssumeRole"
  ],
  "Resource": "arn:aws:iam::*:role/AwsAccountDiscoveryRole"
 }]
}
```

Child account - IAM configuration

1. Create a file: *trust_policy.json* (replace HUB_ACCOUNT_ID with the AWS account ID of your hub account):

```
{
 "Version": "2012-10-17",
 "Statement": [{
  "Effect": "Allow",
  "Principal": {
   "AWS": "arn:aws:iam::HUB_ACCOUNT_ID:root"
  },
  "Action": "sts:AssumeRole"
 }]
}
```

2. Configure your AWS CLI to use the child account. You'll need the following permissions:

```
iam:CreateRole
```

```
iam:AttachRolePolicy
```

3. Create a role called the **AwsAccountDiscoveryRole** in your child account with a trust to the hub account:

   ```
   aws iam create-role --role-name AwsAccountDiscoveryRole \
     --assume-role-policy-document file://trust_policy.json
   ```

4. Attach the ReadOnlyAccess policy to the role:

   ```
   aws iam attach-role-policy --role-name AwsAccountDiscoveryRole \
     --policy-arn arn:aws:iam::aws:policy/ReadOnlyAccess
   ```

Run Discovery from the Hub account

Open the AWS Management Console in the Hub account and open AWS CloudShell.

Configure your AWS CLI to HUB_ACCOUNT_ID

Run the following command with the desired CHILD_ACCOUNT_ID to discover

```
awsdiscover -a CHILD_ACCOUNT_ID
```

> **Note**
> After you have completed account discovery of a child account, AMS recommends deleting the **AwsAccountDiscoveryRole** role if you have no further use for it.

CLI reference

Use the **Help** menu of the tool to get the latest information about the available commands.

Account-level onboarding

The Account-level onboarding stage starts by accepting the terms and conditions and selecting the AWS Regions, add-ons, and Service Level Agreement (SLA) you need for the account. Your CSDM guides you through the acceptance process.

After you have accepted the AMS Terms and Conditions, you need to grant access to the account for the AMS team and tools. You first need to grant access to your Cloud Architect by creating an IAM role for AMS to use; to learn how, see Creating an IAM role for AMS to use (p. 22), in this section. Your Cloud Architect then creates additional roles so the AMS team and tools can access your account. For more details see Access management in AMS Accelerate (p. 85).

Your Cloud Architect also looks for possible configurations in the account, like Service Control Policies (SCPs), and security findings that might prevent AMS from deploying the tools and resources AMS needs to provide its service. Your Cloud Architect works with you to help you remediate findings and remove the blockers to the deployment of AMS tools and resources.

The AMS team starts deploying tools and AWS resources to provide the different services of AMS Accelerate. After it's completed, AMS has built the AWS Managed Services account and AMS notifies you that the service is active, which is the last prerequisite for the billing start date.

The Account-level onboarding stage enables you to continue with the rest of the onboarding process and the following tasks:

- Create incidents and service requests for AWS Infrastructure using the Support Center Console. See Incident reports and service requests in AMS Accelerate (p. 52).
- See the conformance status in your account of the Config Rules deployed by AMS, Compliance and conformance (p. 106).
- Locate and analyze GuardDuty and Macie (optional) findings. See GuardDuty (p. 101).
- Access and audit CloudTrail logs
- Track changes in your AMS Accelerate account. See Tracking changes in your AMS Accelerate accounts (p. 161).
- Use Resource Tagger to create tags. See Resource Tagger (p. 37).
- Request Patch, Backup, and Config Reports. See Reporting in AMS (p. 65).

The next two sections describe onboarding your EC2 instances and configuring Monitoring, Backup, Config Remediation, and Patch (if applicable, AMS Patch Orchestrator is an add-on that you must specifically request) according to your preferences.

Creating an IAM role for AMS to use

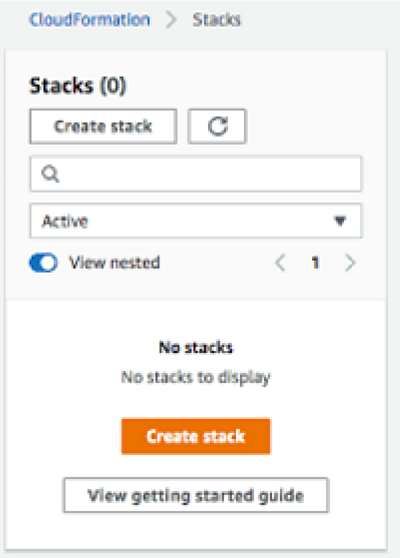
For AMS Accelerate, this is a sub-section of the account-level onboarding steps.

1. Your AMS Cloud Architect provides you with a JSON or YAML file that contains the IAM role AMS uses for creating infrastructure.

   Or you can use this to create the file yourself:

   ```
   {
    "AWSTemplateFormatVersion": "2010-09-09",
    "Description": "AMS Onboarding Role stack (for Prod)",
    "Parameters": {},
    "Conditions": {},
    "Resources": {
     "OnboardingRole": {
      "Type": "AWS::IAM::Role",
      "Properties": {
       "RoleName": "aws_managedservices_onboarding_role",
       "ManagedPolicyArns": ["arn:aws:iam::aws:policy/AdministratorAccess"],
       "AssumeRolePolicyDocument": {
        "Version": "2012-10-17",
        "Statement": [{
         "Action": "sts:AssumeRole",
         "Effect": "Allow",
         "Principal": {
          "AWS": ["328792436863"]
         }
        }]
       }
      }
     }
    }
   }
   ```

2. Sign in to the AWS Management Console and open the AWS CloudFormation console at https://console.aws.amazon.com/cloudformation.



CloudFormation > Stacks

Stacks (0)

Create stack

Active

View nested

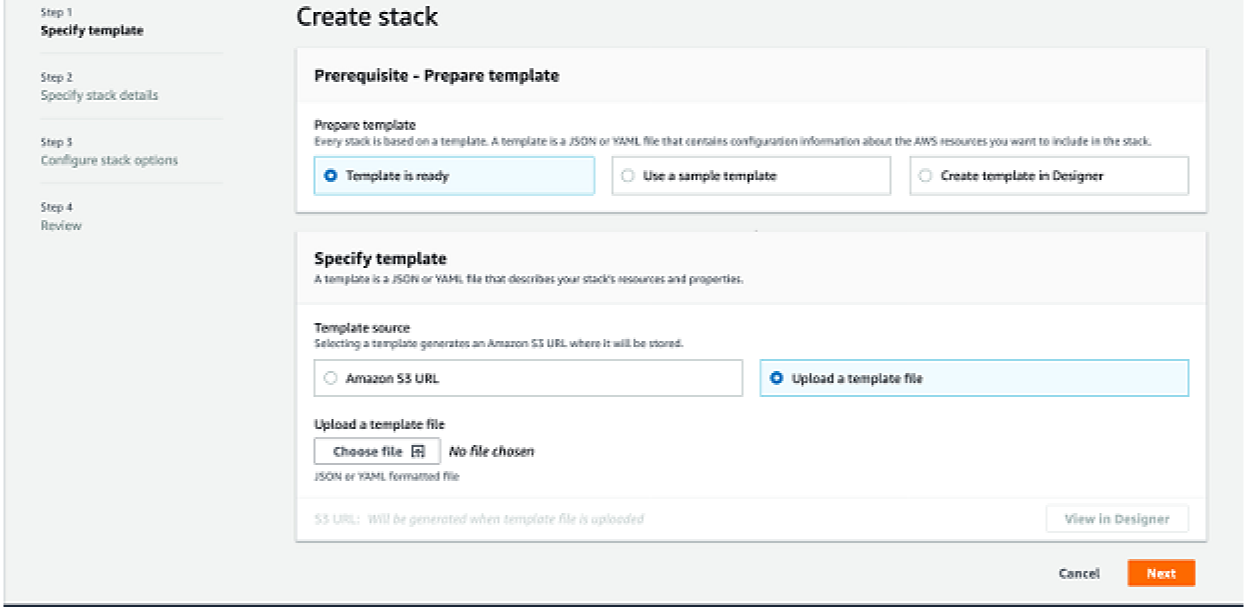
No stacks

No stacks to display

Create stack

View getting started guide

3. Choose **Create Stack**. You see the following page.



Step 1
Specify template

Step 2
Specify stack details

Step 3
Configure stack options

Step 4
Review

Create stack

Prerequisite - Prepare template

Prepare template
Every stack is based on a template. A template is a JSON or YAML file that contains configuration information about the AWS resources you want to include in the stack.

● Template is ready ○ Use a sample template ○ Create template in Designer

Specify template
A template is a JSON or YAML file that describes your stack's resources and properties.

Template source
Selecting a template generates an Amazon S3 URL where it will be stored.

○ Amazon S3 URL ● Upload a template file

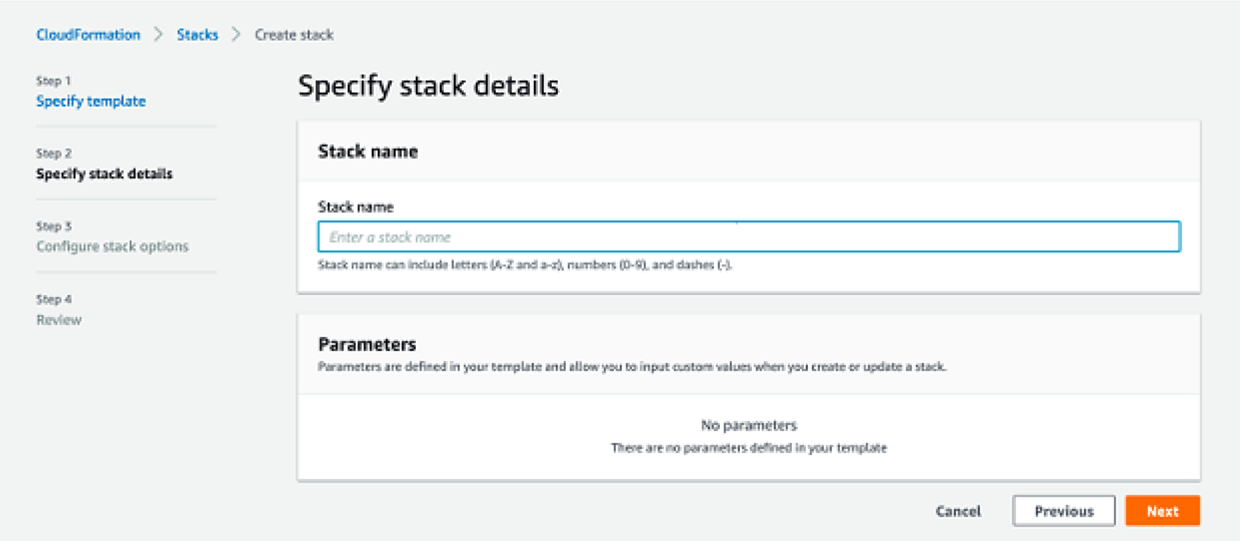
Upload a template file
Choose file *No file chosen*
JSON or YAML formatted file

S3 URL: *Will be generated when template file is uploaded*

View in Designer

Cancel Next

4. Choose **Upload a template file**, upload the JSON or YAML file of the IAM role, and then choose **Next**. You see the following page.



Specify stack details

Stack name

Stack name: Enter a stack name

Stack name can include letters (A-Z and a-z), numbers (0-9), and dashes (-).

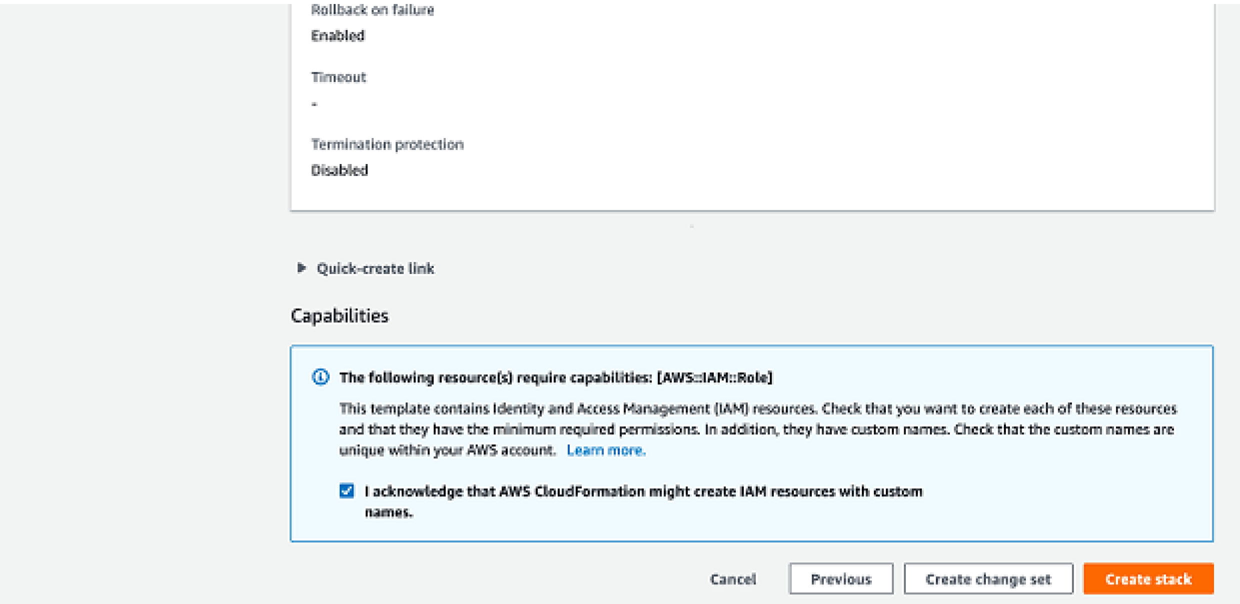
Parameters

Parameters are defined in your template and allow you to input custom values when you create or update a stack.

No parameters

There are no parameters defined in your template

5. Enter **`ams-onboarding-role`** into the **Stack name** section and continue scrolling down and selecting next until you reach this page.



Rollback on failure: Enabled

Timeout: -

Termination protection: Disabled

Capabilities

The following resource(s) require capabilities: [AWS::IAM::Role]

This template contains Identity and Access Management (IAM) resources. Check that you want to create each of these resources and that they have the minimum required permissions. In addition, they have custom names. Check that the custom names are unique within your AWS account. Learn more.

☑ I acknowledge that AWS CloudFormation might create IAM resources with custom names.

6. Make sure the check box is selected and then select **Create Stack**.
7. Make sure the stack was created successfully.

Work with your Cloud Architect (CA) to complete the account-level onboarding steps. After AMS Accelerate completes the account-level onboarding, you're ready to onboard your instances.

Instance-level onboarding

During the instance-level onboarding stage, AMS works with you to configure your EC2 instances and enable baseline monitoring.

You need to install the AWS Systems Manager Agent (SSM Agent) on all of the EC2 instances you want AMS to manage. The SSM Agent makes it possible for AMS and Systems Manager to update, manage, and configure EC2 instances. For details on how to install SSM Agents on EC2 instances, see Working with SSM Agent.

Once the EC2 instances have the SSM Agent installed and in a managed state, AMS needs to implement Automated Instance Configuration on your EC2 instances to start recording Operating System logs and metrics, enabling remote access for AMS engineers, and executing remote commands on the instance; which are essential for our Monitoring, Patch, and Log services and to enable remote access to AMS Operations for management and incident response. For details on setting up automated instance configuration, see Automated instance configuration in AMS Accelerate (p. 48).

To complete the deployment, you must tag the instances. Options for tagging are provided in detail in Tag management in AMS Accelerate (p. 31).

After AMS completes the deployment, you are able to:

- Create incidents and Service Requests for EC2 instances and Operating Systems using the Support Center Console. For more information, see Incident reports and service requests in AMS Accelerate (p. 52).
- Access and audit EC2 logs
- Obtain Patch Reports

Operations configuration

Operations configuration is the last onboarding stage. In this stage, you tailor the AMS services to your needs. You can either self-service the configuration of the services described below or request help from AMS to configure each service based on your inputs. If you need help from AMS, create a service request and provide all the required inputs to complete the task; remeber that service requests are not resolved immediately.

Be sure to perform the following actions for each account:

- Create additional rules to tag your resources, not previously defined. See Resource Tagger (p. 37).
- Configure which resources you want AMS to monitor using Alarm Manager. See Getting started with Alarm Manager (p. 129).
- Customize the threshold of the alarms provided by AMS default configuration. See Modifying the default configuration (p. 137).
- Define patch maintenance windows for your EC2 instances. See Create an SSM maintenance window for patching (p. 149).
- Create custom Patch Baselines. See AMS Accelerate patch baseline (p. 151).
- Define backup plans. See Backup management in AMS Accelerate (p. 143).
- Request Config rules remediation. See Remediation using AWS Config Rules (p. 107).

After you complete the mentioned actions, confirm with your CSDM that you're satisfied with the configuration of the account and you want to conclude the onboarding process. Make sure to share with your CSDM your account-level contact email and contact tag name at the resource level. You can continue adding and changing the configuration mentioned previously at any point.

For information on reports, see Reporting in AMS (p. 65).

Receiving AMS notifications

Communications between you and AMS occur for many reasons:

- Events created by monitoring alerts
- Patching service notifications, if you have opted-in to the Patch add on
- Service requests and incident reports
- Occasional important AWS announcements (your CSDM contacts you if any action on your part is required)

All of these notifications are sent to the default contact information (the root account email) that you provided AMS when you were onboarded. Because it's difficult to keep individual emails updated, we recommend that you use a group email that can be updated on your end. All notifications sent to you are also received by AMS operations and analyzed before making a response.

AMS notification service provides two additional ways to set up contacts for notifications:

- Tag your resources with contact tags (the tag Key Value being contact information) and provide the tag Key Name to your CSDM. Alarms on those resources will be sent to the contacts provided in the Key Value, in addition to the account contact created at onboarding. This is especially useful for application owners. For more information, see Tag-based alert notification (p. 122).
- (Required at onboarding) Send to your CSDM named lists of contacts for non-resource based notifications. For example, you might have a list named "SecurityContacts" and another named "OperationsContacts", and so forth. AMS adds the list to the notification service, and alarms that apply to that list's context are sent to those contacts. This is especially useful for organizational matters.

This advanced alert routing feature is active for most of the essential CloudWatch alarms such as Amazon EC2 instance failure, Amazon Elastic Block Store (Amazon EBS) volume capacity utilization - Root usage, Amazon EBS NonRoot usage, High Memory utilization, High Swap usage, and High CPU utilization for Amazon EC2.

Additionally, when you file a service request, or incident report, you have the option of adding "CC Emails" (highly recommended) and those email addresses receive notifications about the service request or incident.

AMS AMI notifications with SNS

AMS provides an AMI notification service. This service allows you to subscribe to an AWS Simple Notification Service (SNS) topic that notifies you when AMS AMI updates have been released. You can choose to receive notifications for only the AMS AMIs you use, or you can sign up to receive update notifications for all AMS AMIs. For more information on SNS topics, see What is Amazon Simple Notification Service?

Whenever AMIs are released, we send notifications to the subscribers of the corresponding topic; this section describes how to subscribe to the AMS AMI notifications.

Sample message

```
{
  "Type" : "Notification",
  "MessageId" : "example messageId",
  "TopicArn" : "arn:aws:sns:us-east-1:591688410472:customer-ams-windows2019",
  "Subject" : "New AMS AMIs are Now Available",
  "Message" : "{"v1": {"Message": "A new version of the AMS Amazon Machine Images has
 been released.n nYou are now able to launch new EC2 stacks from these AMIs.n nPlease use
 this time to update any dependencies such as CloudFormation or Autoscaling groups.n n
 nRelease Notesn n nWindowsn n n- Contains latest Windows Patches:n n nMicrosoft Windows
 Server 2008 R2 Datacentern n- (KB2819745, KB3018238, KB4507004, KB4507437)n n nMicrosoft
 Windows Server 2016 Datacenter Security Enhancedn n- (KB4509091, KB4507459)n n nMicrosoft
 Windows Server 2016 Datacentern n- (KB4509091, KB4507459)n n nMicrosoft Windows Server
 2012 R2 Security Enhancedn n- (KB3191564, KB3003057, KB3013172, KB3185319, KB4504418,
 KB4506996, KB4507463)n n nMicrosoft Windows Server 2012 R2 Standardn n- (KB3003057,
 KB3013172, KB3185319, KB4504418, KB4506996, KB4507463)n n n nLinuxn n n- Contains latest
 Linux patches n- All AMIs n
ow force domainjoin-cli leave before domainjoin-cli join for better stability in
 the domain join process.nn", "images": {"images": {"image_name": "customer-ams-
windows2019-2021.08-1", "image_id": "ami-05dfa45396fddaa5e"}}, "region": "us-east-1"}}",
  "Timestamp" : "2021-09-03T19:05:57.882Z",
  "SignatureVersion" : "1",
  "Signature" : "example sig",
  "SigningCertURL" : "example url",
  "UnsubscribeURL" : "example url"
}
```

Possible AMS AMI topics to subscribe to:

- **ALL**: Use `customer-ams-all-amis`. This topic subscription notifies you when any of the AMS AMIs are updated.
- **AMS AWS Linux AMIs**: Use `customer-ams-amazon1` (Amazon Linux) or `customer-ams-amazon2` (Amazon Linux 2).
- **AMS AWS RedHat AMIs**: Use `customer-ams-rhel6`, `customer-ams-rhel6-security-enhanced`, `customer-ams-rhel7`, `customer-ams-rhel7-security-enhanced`.
- **AMS AWS CentOs AMIs**: Use `customer-ams-centos7`, `customer-ams-centos7-security-enhanced`.
- **AMS AWS Windows AMIs**: Use `customer-ams-windows2012r2`, `customer-ams-windows2012r2-security-enhanced`, `customer-ams-windows2016`, `customer-ams-windows2016-security-enhanced`.

To subscribe to AMS new AMI notifications by using the Amazon SNS console:

1. Open the Amazon SNS console to the Dashboard.
2. In the upper-right corner, change to the AWS Region for the AMIs that you are subscribing to.
3. In the left-navigation pane, choose **Subscriptions**, and then choose **Create subscription**.
4. Provide the following information:

   a. **Topic ARN**: `arn:aws:sns:{`*`REGION`*`}:287847593866:{`*`AMS_AMI_NAME`*`}` where REGION is the selected AWS Region (where the SNS notification was created) and AMS_AMI_NAME is the AMI that you want notifications about. Examples:

      - To subscribe to notifications of new AMS Amazon Linux AMIs in AWS Region us-east-1, use this **Topic ARN** = `arn:aws:sns:us-east-1:287847593866:customer-ams-amazon1`.
      - To subscribe to notifications of new AMS Window Server 2016 AMIs in AWS Region us-west-2, use this **Topic ARN** = `arn:aws:sns:us-west-2:287847593866:customer-ams-windows2016`

   b. For **Protocol**, choose **Email**.

   c. For **Endpoint**, enter an email address that you can use to receive the notifications. We recommend a distribution list rather than an individual's email.
5. Choose **Create subscription**.
6. When you receive a confirmation email with the subject line "AWS Notification - Subscription Confirmation," open the email and choose **Confirm subscription** to complete your subscription.

To unsubscribe from AMS new AMI notifications by using the AWS SNS console:

1. Open the Amazon SNS console to the Dashboard.
2. In the navigation bar, change to the AWS Region of your choice. You must use the AWS Region in which you want to receive notifications for the corresponding AMIs.
3. In the navigation pane, choose **Subscriptions**, select the subscription, and then choose **Actions** -> **Delete subscriptions**.
4. When prompted for confirmation, choose **Delete**.

To subscribe to AMS New AMI notifications using the Deployment | Ingestion | Stack from CloudFormation Template | Create (ct-36cn2avfrrj9v):

1. To subscribe to the AmazonLinuxSubscription, create and save an execution parameters JSON file; this example names it CreateSubscribeAmiParams.json:

```
{
    "AWSTemplateFormatVersion": "2010-09-09",
    "Resources": {
        "AmazonLinuxSubscription":{
              "Type" : "AWS::SNS::Subscription",
              "Properties": {
                "TopicArn": "arn:aws:sns:{REGION}:287847593866:{AMS_AMI_NAME}",
                "Protocol": "email",
                "Endpoint": "username@yourdomain.com"
            }
        }
      }
}
```

2. Create and save the RFC parameters JSON file with the following content; this example names it CreateSubscribeAmiRfc.json file:

```
{
   "ChangeTypeId": "ct-36cn2avfrrj9v",
   "ChangeTypeVersion": "1.0",
   "Title": "cfn-ingest-subscribe-ami"
}
```

3. Create the RFC, specifying the CreateSubscribeAmiRfc file and the CreateSubscribeAmiParams file:

```
aws amscm create-rfc --cli-input-json file://CreateSubscribeAmiRfc.json  --execution-
parameters file://CreateSubscribeAmiParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

For information on consuming AMIs programmatically, see EC2 stack: creating.

Service notifications

AMS sends outbound service requests, or service notifications, when you need to act on, or be aware of, something that might impact your account or resources, including:

- Infrastructure impact: AMS sends a service notification when there is an underlying AWS service impacting your infrastructure, and you need to take action before a certain date, or you may have an outage.

- EC2 Hardware issues: AMS sends service notifications out for EC2 hardware issues that require you to reboot an EC2 instance before a certain date, or letting you know that AMS will reboot the instance for you. This is an important notice because reboot can cause an outage and you must respond with an acceptable date, or create an RFC with ct-09qbhy7kvtxqw, to reboot the instance yourself. A service notification like this automatically closes in five days if you do not respond.

Using the AMS consoles

The AMS consoles in the AWS Management Console are available for you to interact with AMS and operate your AMS Advanced-managed and AMS Accelerate resources. The AMS consoles generally behave like any AWS console; however, because AMS is a private organization, only accounts enabled for AMS can access the console. Once AMS is enabled in your account, you can access the console by searching for "Managed Services" in the unified search bar.

> **Note**
> Depending on your account role, you access the AMS Advanced console or the AMS Accelerate console.

When using the AMS consoles, be aware of the following caveats:

- The AMS console is account specific. So, if you are in a "Test" account for your organization, you won't be able to see resources in the "Prod" account for that organization. Likewise, you must have an AMS Advanced role to access the AMS Advanced console.
- The AMS consoles apply an IAM policy when you authenticate that determines which console you can access and what you can do there. Your administrator may apply additional polices to the default AMS policy to restrict what you can see and do in the console.

The AMS Accelerate console has these features:

- Opening page: The opening page has information boxes and links to facilitate your access to your existing RFCs, incidents, service request, and reports.
- Feature pages, links in the left-hand navigation pane:
  - **Dashboard**: Provides an overview of the current status of your account including:
    - **Incidents on your resources**: A button for opening an incident case in AWS Support Center, plus how many incident cases are **Awaiting approval** and require your attention and how many are **Open**
    - **Compliance status**: Links to **Rules** and **Resources** that are noncompliant or compliant
    - **Service requests**: A button for opening a service request case in AWS Support Center, plus how many are **Awaiting approval** and require your attention and how many are **Open**
    - **Account-level security**: Links to details on **Real time threat detection** GuardDuty findings and **Data security and privacy** Macie findings
    - **Quick actions**: Open your **Backup vaults** or **Patch instances** configuration pages
  - **Reports**: Opens the **Reports** page and the default reports, **Daily Backup** and **Daily Patch** and **Monthly Billling**
  - **Configuration**: Ensure your resources are being managed successfully and according to your specifications.
    - **Install SSM agent**: The SSM agent is required
    - **Configure tagging rules**: Opens AMS Resource Tagger
    - **Configure alarms**: Opens AMS CloudWatch alarm configuration
    - **Configure patch schedule**: Opens the AWS Systems Manager console
    - **Configure patch baselines**: Opens the AWS Patch Manager console
    - **Configure backup plans**: Opens the AWS Backup console

- **Feature spotlight**: Information on the latest updates to the console
- **Documentation**: The AWS Managed Services documentation landing page

Tag management in AMS Accelerate

Topics

AMS Accelerate uses tags for function and reporting purposes. Tags serve as your primary method of grouping and differentiating your resources, controlling how AMS Accelerate manages your resources, and which resources AMS Accelerate manages.

Each of the AMS Accelerate service offerings have tagging requirements that are outlined in this chapter. Some of the service offerings require you to use specific tags, while others allow you to use any of your own. For information about required tags, see AMS Accelerate required tags (p. 34).

What are tags?

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value, both of which you define.

You use tags to categorize your AWS resources in different ways, for example, by purpose, owner, or environment. For example, you could define a set of tags for your account's Amazon EC2 instances, which helps you track each instance's owner and stack level.

Tags don't have any semantic meaning to Amazon EC2 and are interpreted strictly as a string of characters.

To learn more, see Tagging AWS resources.

How tagging works

There are multiple ways to apply tags to your resources. You can tag resources directly in the console of each AWS service when you create the resource; use AWS Tag Editor to add, remove, or edit tags for multiple resources; or use provisioning tools such as AWS CloudFormation Resource tag. AMS Accelerate also provides the AMS Accelerate Resource Tagger that you use to define rules for an automated tag lifecycle manager. For information about using Resource Tagger in AMS Accelerate, see Resource Tagger (p. 37).

Using Resource Tagger to create tags

The AMS Accelerate Resource Tagger is a component that is deployed in your account during AMS Accelerate onboarding. Resource Tagger has a configurable set of rules that define how your resources should be tagged, then it enforces those rules, automatically adding and removing tags on resources to ensure they comply with your rules.

If you want to use the Resource Tagger to tag your resources, see Resource Tagger (p. 37).

The following is an example Resource Tagger configuration snippet that adds the tag **ams:rt:ams-managed** with the value **true** to all Amazon EC2 instances. The **ams:rt:ams-managed** tag opts you in to having your resources monitored by AMS Accelerate.

```
{
    "AWS::EC2::Instance": {
        "SampleConfigurationBlock": {
            "Enabled": true,
            "Filter": {
                "Platform": "*"
            },
            "Tags": [
                {
                    "Key": "ams:rt:ams-managed",
                    "Value": "true"
                }
            ]
        }
    }
}
```

Using AWS CloudFormation to create tags

If you don't want to use the AMS Accelerate Resource Tagger, you can apply your own tags by using AWS CloudFormation.

> **Important**
> Resource Tagger controls all tags in your account with the **ams:rt:** prefix, and deletes any tags with that prefix if its configuration rules don't include the tag. Some of the AMS Accelerate service components require tags with this prefix, which means that you need to deploy both the Resource Tagger configuration and tags through AWS CloudFormation. You apply the AWS CloudFormation tags to your resources, while your Resource Tagger configuration prevents the Resource Tagger from deleting your tags.

Using AWS CloudFormation, you can apply tags at the stack level (see AWS CloudFormation documentation, Resource tag) or at the individual resource level (for example, see Tagging your Amazon EC2 resources).

The following is an example of how you can apply the tag **ams:rt:ams-managed** with the value **true** to an Amazon EC2 instance managed by AWS CloudFormation. The **ams:rt:ams-managed** tag opts you in to having your resources monitored by AMS Accelerate.

```

 Type: AWS::EC2::Instance

Properties:
  InstanceType: "t3.micro"

  # ...other properties...

  Tags:
    - Key: "ams:rt:ams-managed"
      Value: "true"
```

The following is an example of how you can apply the tag **ams:rt:ams-managed** with the value **true** to an Auto Scaling group managed by AWS CloudFormation. Note that the Auto Scaling group will propagate its tags to Amazon EC2 instances that are created by it. The **ams:rt:ams-managed** tag opts you in to having your resources monitored by AMS Accelerate.

```
  Type: AWS::AutoScaling::AutoScalingGroup
Properties:
  AutoScalingGroupName: "SampleASG"

  # ...other properties...

  Tags:
    - Key: "ams:rt:ams-managed"
      Value: "true"
```

Using Terraform to create tags

If you don't want to use AMS Accelerate Resource Tagger, you can apply your own tags using Terraform. However, if you don't want to use Resource Tagger because of its drift from your Terraform definitions, there is a way for you to use the Resource Tagger and ignore the drift it causes; see Configuring Terraform to ignore Resource Tagger tags (p. 47).

> **Important**
>
> The Resource Tagger controls all tags in your account with the **ams:rt:** prefix, and deletes any tags with this prefix if its configuration rules don’t include the tag. Some of the AMS Accelerate service components require applying tags with this prefix, which means that you need to deploy both the Resource Tagger configuration and tags with Terraform. You apply the Terraform tags to your resources, while your Resource Tagger configuration prevents the Resource Tagger from deleting your tags.

The following is an example of how you can apply the tag **ams:rt:ams-managed** with the value **true** to an Amazon EC2 instance managed by Terraform. The **ams:rt:ams-managed** tag opts you in to having your resources monitored by AMS Accelerate.

```
  resource "aws_instance" "sample_linux_instance" {
    # ...ami and other properties...

    instance_type = "t3.micro"

    tags = {
        "ams:rt:ams-managed" = "true"
    }
}
```

The following is an example of how you can apply the tag **ams:rt:ams-managed** with the value **true** to an Auto Scaling group managed by Terraform. Note that the Auto Scaling group propagates its tags to the Amazon EC2 instances that are created by it. The **ams:rt:ams-managed** tag opts you in to having your resources monitored by AMS Accelerate.

```
  resource "aws_autoscaling_group" "sample_asg" {
    # ...other properties...

    name = "terraform-sample"

    tags = {
        "ams:rt:ams-managed" = "true"
    }
}
```

For a description of how to manage Terraform-created resource tags, see Configuring Terraform to ignore Resource Tagger tags (p. 47).

AMS Accelerate required tags

Certain tags are required to trigger various AMS Accelerate actions.

Monitoring tags

AMS Accelerate monitors supported resources for health, availability, and reliability. For more information about this service offering, see Monitoring and event management in AMS Accelerate (p. 120).

AMS Accelerate periodically onboards additional AWS services to baseline monitoring. If you use the Resource Tagger default configuration, these updates are automatically deployed to your accounts, and changes are reflected to the supported resources.

To opt-in to have your Amazon EC2 instances managed by AMS Accelerate, you must apply the following tag via Customization profile in AppConfig; for more information, see Step 3: Creating a configuration and a configuration profile.

Apply the following tag to your resources:

Key	Value
ams:rt:ams-managed	true

For example, you can create a Customized configuration document like this one to apply the tags to all your AMS-supported EC2 resources:

```
{
    "AWS::EC2::Instance": {
        "AllEC2": {
            "Enabled": true,
            "Filter": {
                "Platform": "*"
            },
            "Tags": [
                {
                    "Key": "ams:rt:ams-managed",
                    "Value": "true"
                }
            ]
        }
    }
}
```

Important
Remember to deploy your configuration changes after you have made them. In SSM AppConfig, you must deploy a new version of the configuration after creating it.

Services other than Amazon EC2 will have default baseline monitoring. In order to *opt out* your resources to be monitored by AMS Accelerate, you can use the customization configuration profile to exclude specific resources or AWS services. This allows you to control which resources should have monitoring tags to deploy baseline alarm definitions. See Working with Resource Tagger (p. 41).

Using Resource Tagger

The AMS Accelerate Resource Tagger configuration in your account ensures that the following tags are deployed automatically, if you apply this one tag (**ams:rt:ams-managed**).

You will see the following tags being applied to your supported resources for baseline monitoring.

Key	Value	Rule
ams:rt:ams-monitoring-policy	ams-monitored	Applies to all EC2 resources supported by AMS
ams:rt:ams-monitoring-policy-platform	ams-monitored-linux	Applies to all Amazon EC2 instances running Linux OS
ams:rt:ams-monitoring-policy-platform	ams-monitored-windows	Applies to all Amazon EC2 instances running Windows OS

For other supported services

Apply the following tags to your resources, according to the given rules:

Key	Value	Rule
ams:rt:ams-monitoring-policy	ams-monitored	Applies to all resources supported by AMS Accelerate monitoring.
ams:rt:ams-monitoring-with-kms	ams-monitored-with-kms	Elasticsearch Domain with KMS
ams:rt:ams-monitoring-with-master	ams-monitored-with-master	Elasticsearch Domain with Dedicated Master Node

If you're not using Resource Tagger

To monitor AMS-supported resources, the resources must have the specific tags defined by AMS. You can use alternate methods of applying tags to your resources, such as AWS CloudFormation; for details, see Using AWS CloudFormation to create tags (p. 32). Or you can use Terraform to apply tags; for details see Using Terraform to create tags (p. 33). To avoid conflicting with Resource Tagger, you need to disable the default configuration; for details, see Disabling the default configuration (p. 44).

Apply the following tags to your resources, according to the given rules:

Key	Value	Rule
ams-managed	true	Apply to supported EC2 instances
ams-monitoring-policy	ams-monitored	Apply to all resources supported by AMS Accelerate monitoring.
ams-monitoring-policy-platform	ams-monitored-linux	Apply to all Amazon EC2 instances running Linux OS

Key	Value	Rule
ams-monitoring-policy-platform	ams-monitored-windows	Apply to all Amazon EC2 instances running Windows OS
ams-monitoring-with-kms	ams-monitored-with-kms	Elasticsearch Domain with KMS
ams-monitoring-with-master	ams-monitored-with-master	Elasticsearch Domain with Dedicated Master Node

Backup management tags

AMS Accelerate manages the backing up of supported resources. For more information about this service offering, see Backup management in AMS Accelerate (p. 143).

AMS Accelerate backup management uses tags to identify which resources should be automatically backed up (and also provides manual backup capabilities). You can use any tag key:value combination to associate your resources with backup plans. To opt in to automated backups using the **ams-default-backup-plan** AWS Backup plan, you must apply the following tag to your supported resources:

Key	Value
ams:rt:backup-orchestrator	true

Note
During onboarding, AMS Accelerate tags all resources with **ams:rt:backup-orchestrator-onboarding** with value **true** for short interval, short retention snapshots. This is managed by the **ams-onboarding-backup-plan** backup plan. For more information about AMS Accelerate-managed AWS Backup plans, see AMS Accelerate backup configuration (p. 145).

Instance configuration automation tags

AMS Accelerate manages agents on your Amazon EC2 instances, such as the SSM agent and the CloudWatch agent. For more information about this service offering, see Automated instance configuration in AMS Accelerate (p. 48)

To opt-in to have your Amazon EC2 instances managed by AMS Accelerate, you must apply the following tag to your Amazon EC2 instances:

Key	Value
ams:rt:ams-managed	true

Patch management tags

AMS Accelerate manages the patching of supported resources. For more information about this service offering, see Patch management in AMS Accelerate (p. 148).

Note
AMS Accelerate patching is an optional add-on service.

You can use any tag **key:value** combination to associate your resources with your patch maintenance windows. AMS Accelerate patch management uses tags to identify which resources should be patched in the default patch cycle. AMS Accelerate provides a default patch cycle when you onboard to patching. To make use of the default patch cycle, add the following tag to your supported resources:

Key	Value
AmsDefaultPatchKey	True

Note
This is the default tag for the default patch cycle. You can change the tag that is used by following instructions in Default patch cycle (p. 151).

AMS Accelerate infrastructure tags

During onboarding to AMS Accelerate, several AWS resources are deployed to your account. So you can identify them, these resources are tagged with the following:

Key	Value
ams:resourceOwner	AMS
ams:resourceOwnerService	A description of which AMS Accelerate service offering this resource comes from, for instance, AMS Deployment, Backup, Controls, Monitoring, Patch, and so forth.
AppId	AMSInfrastucture
AppName	
Environment	

Note
These tags are applied using AWS CloudFormation stack-level tags, and rely on AWS CloudFormation propagating the tags to created resources. For more information, see Resource tag.

Resource Tagger

Tags are an important input for many of our services like patch, backup, and monitoring. You use tags to group and differentiate your resources and map them with different configurations. With Resource Tagger, you can specify rules to govern how AWS resources are tagged in your account. While onboarding an account, AMS Accelerate deploys your tagging policy to ensure resources within your managed accounts are tagged.

Topics

What is Resource Tagger?

Resource Tagger is an AMS Accelerate service offering you use to specify rules to govern how AWS resources are tagged in your account. It aims to provide you with complete visibility into how your tags are applied to your AWS resources.

Resource Tagger automatically creates, updates, and deletes tags on supported AWS resources, based on the tagging rules you specify in your configuration profiles. For example, you can specify a rule that applies a tag to a collection of Amazon EC2 instances, indicating that they should be managed by AMS Accelerate, which results in the instances being monitored or backed up. You can use tags like this to identify compliance status for the AWS resources based on the defined policy in your AWS AppConfig configuration profiles. For more information, see AWS AppConfig.

AMS Accelerate provides a default managed tagging configuration so you can have your resources monitored by AMS Accelerate. You define which resources should be managed by AMS Accelerate, and the managed tagging rules ensure that the resources having the appropriate tags are monitored by AMS Accelerate.

With Resource Tagger, if you choose, you can override or deactivate the default AMS Accelerate managed tags, provide your own tagging rules to meet your policies, and use other mechanisms, such as Terraform, to avoid drift. You can define the exceptions to scale, based on your operations. For example, you could define policy to apply tags for all Amazon EC2 instances with supported platforms (such as Windows and Linux), and exclude from tagging specific instance IDs.

> **Important**
> Resource Tagger controls all tags in your account that have the **ams:rt:** prefix. Any tags that begin with this prefix are deleted unless they are present in Resource Tagger's configuration rules. To summarize, any tag on supported resources that starts with **ams:rt:** is considered owned by Resource Tagger. If you manually tag something with, for example, **ams:rt:**, that tag would automatically be removed if it wasn't specified in one of the Resource Tagger configuration profiles.

How Resource Tagger works

When your account is onboarded to AMS Accelerate, two JSON configuration documents are deployed to your account in AWS AppConfig. The two documents, called *Configuration profiles*, are **AMSManagedTags**, referred to as the **default configuration profile**, and **CustomerManagedTags**, referred to as the **customization configuration profile**. You use the customization configuration profile to define your own policies and rules for your accounts, and those are not overwritten by AMS Accelerate.

Both profiles reside in the **AMSResourceTagger** application, and in the **AMSInfrastructure** environment. All tags applied by the resource tagger have the key prefix **ams:rt:**.

Customization configuration profile:

The customization configuration profile is initially empty at the time of account onboarding; however, any rules placed in the profile document are enforced, in addition to the rules in the default configuration profile. Any configuration in the customization configuration profile is entirely managed by you, and is not overwritten by AMS Accelerate, except by your request.

You can specify any custom tagging rules you want in the custom configuration profile for the supported AWS resources, and you can also specify modifications to the AMS Accelerate-managed default configuration here, see Working with Resource Tagger (p. 41).

> **Important**
> If you update this profile, the Resource Tagger automatically enforces the changes across all relevant resources in your AWS account. The changes are enacted automatically, but they may take up to 60 minutes to take effect.

You can update this profile by using the AWS Management Console, or through AWS CLI/SDK tools. For information about updating a customization configuration profile, see the AWS AppConfig user guide: What Is AWS AppConfig?

Default configuration profile:

The default configuration profile document is internal to AMS Accelerate and it contains AMS Accelerate-supplied default rules that you can't modify or delete permanently. This profile can be updated at any time by AMS Accelerate and made available to you for review; any changes you have made to it are automatically deleted. If you want to modify or disable any of the default configuration rules you use the customization configuration profile, see Working with Resource Tagger (p. 41).

Configuration profile document format

The default configuration profile document and the customization configuration profile document have the same format and follow the same structure:

```
{
    "Options": {
        "ReadOnly": false
    },
    "ResourceType": {
  "ConfigurationID": {
   "Enabled": true,
   "Filter": {
    ...
   },
   "Tags": [
    ...
   ]
  },
 "ConfigurationID": {
   ...
  }
 },
 "ResourceType": {
  ...
 }
}
```

Definitions:

Options: (optional) Specify options for how you would like the ResourceTagger to behave. Omitting the block is equivalent to setting all options to their default values. See below for available **Options** settings:

- **ReadOnly**: (optional, defaults to false): Specifies ReadOnly mode for Resource Tagger. Set ReadOnly to true to disable Resource Tagger creating or removing tags on AWS resources. For more information, see Preventing Resource Tagger from modifying resources (p. 42).

ResourceType: This key must be one of the following supported strings, and represents all configuration related to the resource type indicated:

- AWS::EC2::Instance
- AWS::RDS::DBInstance
- AWS::RDS::DBCluster
- AWS::Elasticsearch::Domain
- AWS::Redshift::Cluster
- AWS::ElasticLoadBalancing::LoadBalancer

- AWS::ElasticLoadBalancingV2::LoadBalancer
- AWS::EC2::NatGateways
- AWS:EC2::VPNConnection

ConfigurationID: This key must be unique in the profile document, and uniquely names the following block of configuration. If two configuration blocks in the same **ResourceType** block have the same **ConfigurationID**, the one that appears last in the profile takes effect. If you specify a **ConfigurationID** in your customization profile that is the same as one specified in the default document, the configuration block defined in the customization profile takes effect.

> **Important**
>
> The **ConfigurationID** should ***not*** overlap with the AMS Accelerate profile; for example, it should not be **AMSMonitoringLinux** or **AMSMonitoringWindows**, otherwise it disables the respective configuration of the **AMSManagedTags** configuration profile.

Enabled (optional, defaults to **true**): Specifies if the configuration block takes effect. Set this to **false** to disable a configuration block. A disabled configuration block has no effect.

Filter: Specifies the resources that the configuration applies to. Each filter object can have any one (but only one) of the following fields:

- **AWS::EC2::Instance**
  - **InstanceId**: The filter matches an EC2 instance with the specified instance ID. This field supports wildcard matching, so **i-00000*** would match any instance that has an instance ID starting with **i-00000**.
  - **Platform**: The filter matches an EC2 instance with the specified platform. Valid values are **windows**, **linux** or the wildcard * (to match any platform).
- **AWS::EC2::NatGateways**:
  - **NatGatewayId**: The ID of the NAT Gateway. This field supports wildcard matching.
  - **State**: The state of the NAT gateway (pending | failed | available | deleting | deleted or wildcard "*")
  - **VpcId**: The ID of the VPC in which the NAT Gateway resides. This field supports wildcard matching.
  - **SubnetId**: The ID of the Subnet in which the NAT Gateway resides. This field supports wildcard matching
- **AWS::EC2::VPNConnection**:
  - **VpnConnectionId**: The ID of the connection.
- **AWS::RDS::DBCluster**
  - **DBClusterIdentifier**: The filter matches an RDS cluster identifier with the specified identifier. This field does not support wildcard matching, so a cluster identifier must be specified.
  - **Engine**: The engine in use by the RDS Instance. This field supports wildcard matching.
  - **EngineVersion**: The engine version. This field supports wildcard matching.
- **AWS::RDS::DBInstance**
  - **DBInstanceIdentifier**: The filter matches an RDS instance with the specified instance ID. This field does not support wildcard matching, so an instance identifier must be specified.
  - **Engine**: The engine in use by the RDS Instance. This field supports wildcard matching.
  - **EngineVersion**: The engine version. This field supports wildcard matching.
- **AWS::Elasticsearch::Domain**
  - **DomainId**: The DomainId of the Elasticsearch resource. This field supports wildcard matching.
  - **DomainName**: The DomainName of the Elasticsearch resource. This field supports wildcard matching.
  - **HasMasterNode**: Boolean; If the Domain has a dedicated master node, this can be set to true.
  - **HasKmsKey**: If the Domain has a KMS key for encryption at rest, this can be set to true.

- **AWS::Redshift::Cluster**
  - **ClusterIdentifier**: The Cluster Identifier. This field supports wildcard matching.
- **AWS::ElasticLoadBalancing::LoadBalancer (Classic Load Balancer)**
  - **LoadBalancerName**: The LoadBalancer Name. This field supports wildcard matching.
  - **Scheme**: Can be either "internet-facing", "internal" or wildcard "*".
  - **VPCId**: The VPCId in which the loadbalancer is deployed, can be wildcard "*".
- **AWS::ElasticLoadBalancingV2::LoadBalancer (Application Load Balancer (ALB))**
  - **LoadBalancerArn**: The LoadBalancer Amazon Resource Name (ARN).
  - **DNSName**: The DNSName of the LoadBalancer. This field supports wildcard matching.
  - **LoadBalancerName**: The LoadBalancer Name. This field supports wildcard matching.

Other Filter properties:

- **Tag**: The filter applies to any resource that already has the given tag applied. The value for this property must be a JSON object with the following fields:
  - **Key**: Must be an exact string, and specifies that the resources must have a tag with that exact key.
  - **Value**: Specifies the matching value for the tag. Supports wildcards, so a value of **Sample** matches any value that ends with the string **Sample**.
- **Fn::AND**: A JSON array of JSON objects. Each object follows the same rules as the **Filter** configuration block. This specifies that the filter match any resource that matches all of the sub-filters.
- **Fn::OR**: A JSON array of JSON objects. Each object follows the same rules as the **Filter** configuration block. This specifies that the filter match any resource that matches any of the sub-filters.
- **Fn::NOT**: A JSON object that follows the same rules as the **Filter** configuration block. This specifies that the filter explicitly not match any resource that matches the sub-filter. Use this to specify exclusions to your tagging rules.

Tags: The tags to be applied to the matched resources. This field is a JSON array, with each element being a JSON object containing the following fields:

- **Key**: The key for the tag to be applied.
- **Value**: The value for the tag to be applied.

Note

All tags applied by Resource Tagger always have the prefix **ams:rt:**. If you do not specify this prefix on your tag value here, it's automatically applied for you. This is to maintain a record of which tag values have been applied by Resource Tagger, as opposed to those applied by any other means.

For important information on tags, see Tag naming and usage conventions

Working with Resource Tagger

How to use AMS Accelerate Resource Tagger.

Topics

- Viewing or making changes to the Resource Tagger configuration (p. 46)
- Deploying configuration changes (p. 46)
- Configuring Terraform to ignore Resource Tagger tags (p. 47)

Viewing the tags applied by Resource Tagger

All tags applied by Resource Tagger have the key prefix **ams:rt:**. For example, the following tag definition results in a tag with key **ams:rt:sampleKey** and value **sampleValue**. All tags with this prefix are treated as being part of Resource Tagger.

```
{
 "Key": "sampleKey",
 "Value": "sampleValue"
}
```

Important
If you manually create your own tag with the **ams:rt:** prefix, it's considered managed by Resource Tagger. This means that if the resource is managed by Resource Tagger, but the configuration profiles do not indicate that the tag should be applied, then Resource Tagger removes your manually added tag. If you do manually tag resources managed by Resource Tagger, do not use the **ams:rt:** prefix for tag keys.

Preventing Resource Tagger from modifying resources

Resource Tagger can be set to a read-only mode that prevents it from adding or removing any tags on your resources. This is useful if you want to provide your own tagging mechanism.

When in read-only mode, Resource Tagger still examines the tagging rules that are being specified in the managed and customer configuration profiles, and scans for resources that do not meet these tagging rules. Any non-compliant resources are surfaced with AWS Config. The AWS Config rules that you can look for have the `AMSResourceTagger-` prefix. For example the `AMSResourceTagger-EC2Instance` Config rule evaluates if appropriate tags are created for `AWS::EC2::Instance` resources based on the configuration profile.

Resource Tagger stops at this point, and does not make any changes to your resources (does not add or remove tags).

You can enable the read-only mode by modifying the customer configuration profile to include the **ReadOnly** key in the **Options** block. For example, the following configuration profile snippet shows how this might look:

```
{
    "Options": {
        "ReadOnly": true
    },
    "AWS::EC2::Instance": {
        [... the rest of your configuration ...]
    }
}
```

Resource Tagger would react to this new configuration as soon as it has finished deploying, and stop adding and removing tags on resources.

Note
To re-enable tag modification, change the **ReadOnly** value to **false**, or remove the key altogether, since the default value is **false**.

For more on the **Options** setting, see Configuration profile document format (p. 39), next.

Example configuration profile

The following example profile document specifies that all Windows EC2 instances that are part of a stack-* CloudFormation stack be managed by AMS Accelerate; however, explicitly excludes a particular EC2 instance with ID i-00000000000000001.

```
{
    "AWS::EC2::Instance": {
        "AMSMonitoringWindows": {
            "Enabled": true,
            "Filter": {
                "Fn::AND": [
                    {
                        "Platform": "Windows"
                    },
                    {
                        "Tag": {
                            "Key": "aws:cloudformation:stack-name",
                            "Value": "stack-*"
                        }
                    },
                    {
                        "Fn::NOT": {
                            "InstanceId": "i-00000000000000001"
                        }
                    }
                ]
            },
            "Tags": [
                {
                    "Key": "ams:rt:ams-managed",
                    "Value": "true"
                }
            ]
        }
    }
}
```

Merging the default configuration

The default configuration profile is supplied by AMS Accelerate at the time of account onboarding. This profile provides default rules that are deployed in your account.

While you can't modify the default configuration profile, you can provide overrides to the defaults by specifying a configuration block in your customization configuration profile with the same **ConfigurationID** as the default configuration block. If you do this, your configuration block overwrites the default configuration block.

For example, consider the following default configuration document:

```
{
 "AWS::EC2::Instance": {
  "AMSManagedBlock1": {
   "Enabled": true,
   "Filter": {
    "Platform": "Windows"
   },
   "Tags": [{
    "Key": "my-tag",
    "Value": "SampleValueA"
   }]
  }
 }
}
```

In order to change the tag value applied here from `SampleValueA` to `SampleValueB`, and have the tag applied to all instances, not just Windows instances, you would provide the following customization configuration profile:

```
{
 "AWS::EC2::Instance": {
  "AMSManagedBlock1": {
   "Enabled": true,
   "Filter": {
    "Platform": "*"
   },
   "Tags": [{
    "Key": "my-tag",
    "Value": "SampleValueB"
   }]
  }
 }
}
```

Important
Remember to deploy your configuration changes after you have made them; for information, see Deploying configuration changes (p. 46). In SSM AppConfig, you must deploy a new version of the configuration after creating it.

Disabling the default configuration

You can disable a default configuration rule by adding a configuration block with the same **ConfigurationID** to your customization configuration profile and giving the **Enabled** field for a value of **false**.

For example, if the following configuration were present in the default configuration profile:

```
{
 "AWS::EC2::Instance": {
  "AMSManagedBlock1": {
   "Enabled": true,
   "Filter": {
    "Platform": "Windows"
   },
   "Tags": [{
    "Key": "my-tag",
    "Value": "SampleValueA"
   }]
  }
 }
}
```

You could disable this tagging rule by including the following in your customization configuration profile:

```
{
 "AWS::EC2::Instance": {
  "AMSManagedBlock1": {
   "Enabled": false
  }
 }
}
```

Important
Remember to deploy your configuration changes after you have made them; for information, see Deploying configuration changes (p. 46). In SSM AppConfig, you must deploy a new version of the configuration after creating it.

Removing tags applied by Resource Tagger

Any tags prefixed with **ams:rt** are removed by Resource Tagger if the tags do not exist in the configuration profiles, or, if they do exist, where the filter doesn't match. This means that you can remove tags applied by Resource Tagger by doing one of the following:

- Modifying the customization configuration section that defines the tag.
- Adding an exception for the specific resources so they no longer match the filter.

For example: if a **Linux** instance has the following tags:

```
"Tags": [{
    "Key": "ams:rt:MyOwnTag",
    "Value": true
},{
    "Key": "myTag",
    "Value": true
}]
```

And you deploy the following Resource Tagger configuration profile:

```
{
    "AWS::EC2::Instance": {
        "AMSMonitoringWindows": {
            "Enabled": true,
            "Filter": {
                "Platform": "Windows"
            },
            "Tags": [{
                "Key": "ams:rt:ams-managed",
                "Value": "true"
            }]
        }
    }
}
```

Resource Tagger reacts to the new configuration changes, and the only tag on the instance becomes:

```
 "Tags": [{
     "Key": "myTag",
     "Value": true
 }]
```

Important
Remember to deploy your configuration changes after you have made them; for information, see Deploying configuration changes (p. 46). In SSM AppConfig, you must deploy a new version of the configuration after creating it.

Viewing or making changes to the Resource Tagger configuration

The two JSON configuration profiles, **AMSManagedTags** and **CustomerManagedTags**, deployed to your account in AWS AppConfig at onboarding and residing in the AMSResourceTagger application, and in the **AMSInfrastructure** environment, can be reviewed through AppConfig's GetConfiguration API.

The following is an example of this GetConfiguration call:

```
aws appconfig get-configuration
 --application AMSResourceTagger
 --environment AMSInfrastructure
 --configuration AMSManagedTags
 --client-id ANY_STRING
 outfile.json
```

Application: AppConfig logical unit to provide capabilities, for the Resource Tagger, this is AMSResourceTagger.

- **Environment**: AMSInfrastructure.
- **Configuration**: To view AMS Accelerate default tag definitions, the value is AMSManagedTags, while to view customer tag definitions, the value is CustomerManagedTags.
- **Client ID**: The unique application instance identifier, this can be any string.
- The tag definitions can then be viewed in the specified output file, in this case, outfile.json.

The alarm definitions can then be viewed in the specified output file, in this case, outfile.json.

You can see which version of configuration is deployed to your account by viewing the past deployments in the **AMSInfrastructure** environment.

To override tag rules:

Any of the existing tag rules can be overridden by updating the customization profile using AppConfig's **CreateHostedConfigurationVersion** API. Using the same **ConfigurationID** as a default configuration tag rule overrides the default rule, and applies the custom rule in its place.

To deploy changes made to the **CustomerManagedTags** document:

After you make changes to the customization configuration profile, you must deploy the changes for them. To deploy the new changes, AppConfig's StartDeployment API must be run using the AWS AppConfig Console or the CLI.

Deploying configuration changes

Once the customization is completed, these changes must be deployed through the AWS AppConfig StartDeployment API. The following instructions show how to deploy using the AWS CLI. Additionally, you can use the AWS Management Console to make these changes. For information, see Step 5: Deploying a configuration.

```
aws appconfig start-deployment
--application-id <application_id>
--environment-id <environment_id>
--deployment-strategy-id <deployment_strategy_id>
--configuration-profile-id <configuration_profile_id>
--configuration-version 1
```

- **Application ID**: The application ID of the application AMSResourceTagger. Get this with the ListApplications API call.
- **Environment ID**: The environment ID; get this with the ListEnvironments API call.
- **Deployment Strategy ID**: The deployment strategy ID; get this with the ListDeploymentStrategies API call.
- **Configuration Profile ID**: The configuration profile ID of CustomerManagedTags; get this with the ListConfigurationProfiles API call.
- **Configuration Version**: The version of the configuration profile you intend to deploy.

Important
Resource Tagger applies the tags as specified in the configuration profiles. Any manual modifications you make (with the AWS Management Console, or CloudWatch CLI/SDK) to the resource tags are automatically reverted back, so ensure your changes are defined through Resource Tagger. To know which tags are created by the Resource Tagger, look for tag keys prefixed with `ams:rt:`.

Restrict access to the deployment with the StartDeployment and the StopDeployment API actions to trusted users who understand the responsibilities and consequences of deploying a new configuration to your targets.

To learn more about how to use AWS AppConfig features to create and deploy a configuration, see the documentation at Working with AWS AppConfig.

Configuring Terraform to ignore Resource Tagger tags

If you use Terraform to provision your resources, and you want to use Resource Tagger to tag your resources, the Resource Tagger tags may be identified as drift by Terraform.

You can configure Terraform to ignore all Resource Tagger tags using the **lifecycle** configuration block, or the **ignore_tags** global configuration block. For more information, see the Terraform documentation on Resource Tagging at Resource Tagging.

The following example shows how to create a global configuration to ignore all tags that begin with the Resource Tagger tag prefix `ams:rt:`:

```
provider "aws" {
  # ... potentially other configuration ...

  ignore_tags {
    key_prefixes = ["ams:rt:"]
  }
}
```

Automated instance configuration in AMS Accelerate

Topics

AMS Accelerate provides an automated instance configuration service. These configurations ensure that the instance is writing the correct logs and emitting the correct metrics in order for AMS to properly manage the instance. In order for this service to activate, AMS Accelerate requires that specific conditions are met that enable AMS to configure the instance.

The steps required to enable the automatic configuration of the required settings on the instance are provided in this section.

How automated instance configuration works

Automated instance configuration enables AMS Accelerate to perform certain configurations on a daily basis on instances that you indicate by adding particular agents and tags.

Automated instance configuration changes

The AMS Accelerate instance configuration automation makes the following changes:

1. IAM Permissions

   Adds the IAM-managed Policies required to grant the instance permission to use the agents installed by AMS Accelerate, described next.
2. Agents
   1. The Amazon CloudWatch Agent is responsible for emitting OS logs and metrics. The instance configuration automation ensures that the CloudWatch agent is installed and running the AMS Accelerate minimum version.
   2. The Amazon SSM Agent is responsible for running remote commands on the instance. The instance configuration automation ensures that the SSM Agent is running the AMS Accelerate minimum version.
3. CloudWatch Configuration
   1. To ensure that the required metrics and logs are emitted, AMS Accelerate customizes the CloudWatch configuration. For more information, see the following section, CloudWatch configuration (p. 50).

Automated instance configuration prerequisites

These conditions must be met to enable AMS Accelerate to perform the previously described automated actions on managed instances.

The SSM Agent is installed

AMS Accelerate automated instance configuration requires that the Amazon SSM Agent is installed. For more information, see the following:

- Linux: Manually install SSM Agent on EC2 instances for Linux - AWS Systems Manager
- Windows: Manually install SSM Agent on EC2 instances for Windows Server - AWS Systems Manager

The SSM Agent is in the managed state

AMS Accelerate automated instance configuration requires an operational SSM agent. The Amazon SSM Agent must be installed, and the Amazon EC2 instance must be in the managed state. For more information, see the AWS documentation, Working with SSM Agent.

Automated instance configuration setup

Assuming the prerequisites have been met, adding a specific Amazon EC2 instance tag automatically initiates the AMS Accelerate automated instance configuration. Use one of the following methods to add this tag:

1. (Strongly recommended) Use the AMS Accelerate Resource Tagger

   To configure the tagging logic for your account, see How tagging works (p. 31). After tagging is complete, tags and automated instance configuration are handled automatically.
2. Manually add tags

   Manually add the following tag to the Amazon EC2 instances:

   Key:**ams:rt:ams-managed**, Value:**true**.

   > **Note**
   > The instance configuration service attempts to apply the required AMS configurations once the **ams:rt:ams-managed** tag is applied to the instance. The service asserts the AMS required configurations whenever an instance is started, and when a the AMS daily configuration check occurs.

Automated instance configuration details

Automated instance configuration makes particular changes or additions to your IAM instance profiles and CloudWatch configuration.

IAM permissions

Instance profiles attached to instances are checked to ensure they contain the correct managed policies. The following managed policies are added to the existing role if they aren't present.

- arn:aws:iam::aws:policy/AmazonSSMManagedInstanceCore
- arn:aws:iam::aws:policy/CloudWatchAgentServerPolicy
- arn:aws:iam::aws:policy/AMSInstanceProfileBasePolicy

If an instance profile is not attached to the instance, the workflow attaches the following instance profile with the needed permissions to run the AMS Accelerate OS automated instance configuration workflow: **AMSOSConfigurationCustomerInstanceProfile-<REGION>**

CloudWatch configuration

Additional detail on the CloudWatch configuration.

- CloudWatch configuration file location on the instance:
  - Windows: %ProgramData%\Amazon\AmazonCloudWatchAgent\amazon-cloudwatch-agent.json
  - Linux: /opt/aws/amazon-cloudwatch-agent/etc/amazon-cloudwatch-agent.d/ams-accelerate-config.json
- CloudWatch configuration file location in Amazon S3:
  - Windows: https://ams-configuration-artifacts-<region-name>.s3-<region-name>.amazonaws.com/configurations/cloudwatch/latest/windows-cloudwatch-config.json
  - Linux: https://ams-configuration-artifacts-<region-name>.s3-<region-name>.amazonaws.com/configurations/cloudwatch/latest/linux-cloudwatch-config.json
- Metrics collected:
  - Windows:
    - Amazon SSM Agent (CPU_Usage)
    - CloudWatch Agent (CPU_Usage)
    - Disk space utilization for all disks (% free space)
    - Memory (% committed bytes in use)
  - Linux:
    - Amazon SSM Agent (CPU_Usage)
    - CloudWatch Agent (CPU_Usage)
    - CPU (cpu_usage_idle, cpu_usage_iowait, cpu_usage_user, cpu_usage_system)
    - Disk (used_percent, inodes_used, inodes_total)
    - Diskio (io_time, write_bytes, read_bytes, writes, reads)
    - Mem (mem_used_percent)
    - Swap (swap_used_percent)
- Logs collected:
  - Windows:
    - AmazonSSMAgentLog
    - AmazonCloudWatchAgentLog
    - AmazonSSMErrorLog
    - AmazonCloudFormationLog
    - ApplicationEventLog
    - EC2ConfigServiceEventLog
    - MicrosoftWindowsAppLockerEXEAndDLLEventLog
    - MicrosoftWindowsAppLockerMSIAndScriptEventLog
    - MicrosoftWindowsGroupPolicyOperationalEventLog
    - SecurityEventLog
    - SystemEventLog
  - Linux:
    - /var/log/amazon/ssm/amazon-ssm-agent.log
    - /var/log/amazon/ssm/errors.log
    - /var/log/audit/audit.log
    - /var/log/cloud-init-output.log
    - /var/log/cloud-init.log
    - /var/log/cron

- /var/log/maillog
- /var/log/messages
- /var/log/secure
- /var/log/spooler
- /var/log/yum.log
- /var/log/zypper.log

Incident reports and service requests in AMS Accelerate

Topics

With AMS Accelerate, you can request help with operational issues and requests at any time through the AWS Support Center in the AWS console. AMS Accelerate operations engineers are available to respond to your incidents and service requests 24x7, with response time Service Level Agreements (SLAs) and Service Level Objectives (SLOs), dependent on your selected account Service Tier (Plus, Premium). AMS Accelerate operations engineers proactively notify you of important alerts and questions using the same mechanisms.

Incident management

In AMS Accelerate, you use the **AWS Support Center** in the **AWS Console** to file incident reports. Incidents are AWS service performance issues that impact your managed environment, as determined by AMS Accelerate or you. Incidents identified by the AMS Accelerate team are first received as "events" (a change in system state captured by monitoring). If a configured threshold is breached, the event triggers an alarm, also called an alert. The AMS Accelerate operations team determines if the event is non-impacting, or an incident (a service interruption or degradation), or a problem (the underlying root cause of one or more incidents).

> **Note**
> The AMS Accelerate team also receives incidents created by you programmatically using the AWS Support API with service code `service-ams-operations-report-incident`.

For information about using AWS Support, see Getting started with AWS Support.

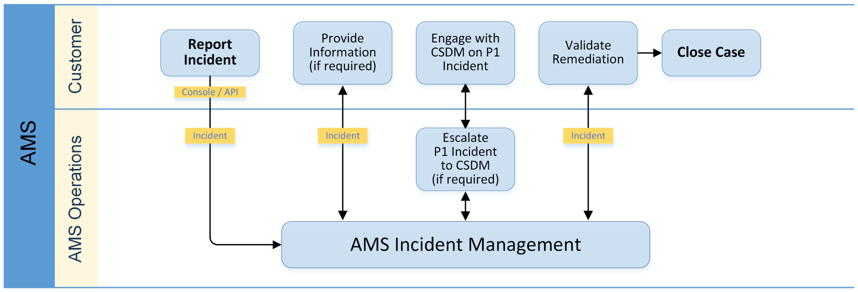
What is incident management?

Incident management is the process AMS uses to record, act on, communicate progress of, and provide notification of, active incidents.

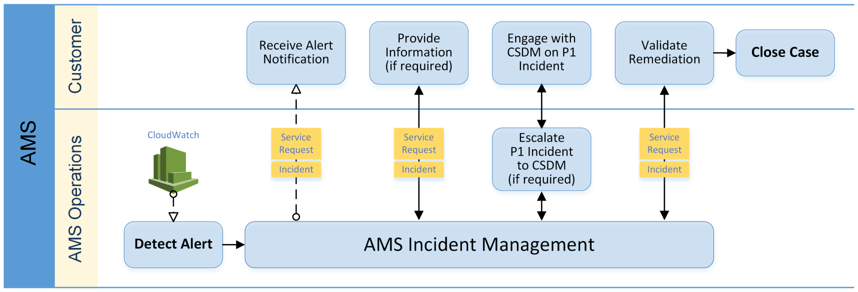
The goal of the incident management process is to ensure that normal operation of your managed service is restored as quickly as possible, the business impact is minimized, and all concerned parties are kept informed.

Examples of incidents include (but are not restricted to) loss of or degradation of network connectivity, a non-responsive process or API, or a scheduled task not being performed (for example, a failed backup).

The following graphic depicts the workflow of an incident reported by you to AMS.



This graphic depicts the workflow of an incident reported by AMS to you.



Incident priority

Incidents created in AWS Support center, console or Support API (SAPI), have different classifications than incidents created in the AMS console.

- Low: Non-critical functions of your business service, or application, related to AWS or AMS resources are impacted.
- Medium: A business service or application related to AWS and/or AMS resources is moderately impacted and is functioning in a degraded state.
- High: Your business is significantly impacted. Critical functions of your application related to AWS and/or AMS resources are unavailable. Reserved for the most critical outages affecting production systems.

> **Note**
> The AWS Support Console offers five levels of incident priority that we translate to the three AMS levels.

How incident response and resolution work

AMS Accelerate uses IT service management (ITSM) incident management best practices to restore service, when needed, as quickly as possible.

We provide 24/7/365 follow-the-sun support through five operations centers around the world with dedicated operators actively watching monitoring dashboards and incident queues.

Our operations engineers use internal incident tracking tools to identify, log, categorize, prioritize, diagnose, resolve, and close incidents; we provide you with updates on all of these activities through AWS Support Center and through the AWS Support API. Our operators leverage a variety of internal AWS support tools to help with all of those activities. These operators are deeply familiar with AMS Accelerate-supported infrastructure and have expert-level technical skills to address identified support issues. In the event our operators need assistance, the Premium Support and AWS service teams are available.

After your incident is received by the AMS Accelerate operations team, we validate the priority and classification working with you if there are any clarifications required. For example, if the incident report is better classified as a service request, it's reclassified and the AMS Accelerate service request team takes over and you're notified. If the incident can be resolved by the receiving operator, steps are taken to quickly resolve the incident. AMS Accelerate operators consult internal documentation for a resolution and, if needed, escalate the incident to other support resources until the incident is resolved. After it's resolved, the AMS Accelerate operations team documents the incident and resolution for future use.

In cases where critical severity incidents are impacting your critical workloads, AMS Accelerate may recommend an infrastructure restore. There is often a trade-off between troubleshooting an issue and simply restoring from a known functional backup, and your risks and impacts from service downtime are the deciding factors. If you have time to devote to troubleshooting issues, AMS Accelerate will assist you, and your cloud service delivery manager (CSDM) may get involved, but if the urgency to restore is high, AMS Accelerate can initiate a restore right away.

Working with incidents

From AWS Support Center, you can perform the following tasks:

- Report and update an incident. To report an AMS Accelerate incident, choose **AMS Operations -- Report Incident** from the **Services** menu.
- Get a list of, and detailed information about, all of your submitted incidents.
- Narrow your search for incidents by status and other filters.
- Add communications and file attachments to your incidents, and add email recipients for case correspondence.
- Initiate a live chat or request a call back on your incident.
- Resolve incidents.
- Rate incident communications.

The following examples describe using Support Center to submit an incident. After it's submitted, the AMS Accelerate team works with you to resolve the incident per the standard AMS Accelerate SLA.

Submitting an incident

To report an incident using Support Center, refer to the support documentation: Creating a support case

To report an incident using AWS Support center:

1. Click **Create case**. The create incident case page opens.
2. Open the **Technical support issue type** menu and choose **AMS Operations -- Report Incident**. Supply information about your incident and choose **Create**.
3. To be kept informed by email at each step of the incident resolution process, be sure to fill in the **CC Emails** option; if you connect by federation, log in before following the link in the email that AMS Accelerate sends you about the incident.

> **Note**
> Make your description as detailed as possible. Include relevant resource information, along with anything else that might help us understand your issue. For example, to troubleshoot performance, include timestamps and logs. For feature requests or general guidance questions, include a description of your environment and purpose. In all cases, follow the Description Guidance that appears on your case submission form.
> When you provide as much detail as possible, you increase the chances that your case can be resolved quickly.

You can also use the AWS Support API with service code `service-ams-operations-report-incident` to report an incident.

Monitoring and updating an incident

You can update, monitor, and review incident reports and service requests, both called *cases*, by using Support Center, or programmatically using the AWS Support API, `DescribeCases` operation.

To monitor a case (incident or service request) using AWS Support Center, follow these steps.

1. In the AWS Management console, browse to **Support**.
2. From the left navigation, select **Your support cases**, browse to a case and choose the **Subject** link to open a details page with current status and correspondences.

   If you want to use phone or chat at this point, click **Open case in Support Center** to open the case **Create** page in the AWS Support Center, auto-populated with the AMS service type.

   When a reported incident or service request case is updated by the Accelerate operations team, you receive an email and a link to the incident in the Support Center so you can respond.

   > **Note**
   > You can't respond to case correspondence by replying to the email.

   If there are many cases in the dashboard, you can use the **Filter** option:

   - **Subject**: Use this filter to search on keywords in the subject of the case.
   - **Severity**: Use this to filter cases by severity by selecting a severity from the list.
   - **Case type**: Use this to see all cases of a particular case type. Accelerate incidents and service requests appear under the Technical Support Case Type along with any service-specific cases.
   - **Status**: Use this to filter cases by status by selecting a specific status from the list.
3. To check the latest status, refresh the page.
4. If there are so many correspondences that they do not all appear on the page, choose **Load More**.
5. To provide an update to the case status, choose **Reply**, enter the new correspondence, and then choose **Submit**.
6. To close out the case after it has been resolved to your satisfaction, choose **Close case**.

   Be sure to rate the service through the 1-5 star rating to let AMS know how we're doing.

Managing incidents with the support API

You can use the AWS Support API to create incidents and add correspondence with AWS Support staff during investigations into your issues. The AWS Support API models much of the behavior of the AWS Support Center.

For information about you can use this AWS support service, see Programming the Life of an AWS Support Case.

> **Note**
> The AMS Accelerate team receives incidents created by you programmatically using the with service code `service-ams-operations-report-incident`.

Responding to an AMS Accelerate-generated incident

AMS Accelerate proactively monitors your resources. For more information, see Monitoring and event management in AMS Accelerate (p. 120). Sometimes AMS Accelerate identifies and creates an incident, most often to notify you of an event. If action on your part is required to resolve an incident, notification is sent by the AMS Accelerate team to the contact information you have provided for the account. You respond to this notification in the same way as for any other incident—usually through Support Center, though in some cases contact through email or phone is required.

Important
To receive state change notifications for an incident case or service request, enter an email address in the addresses field.

Service request management

AMS Accelerate uses service request management to record, act on, communicate progress of, and provide notification of active service requests.

The goal of the service request management process is to ensure that your managed service is delivering what you need.

For billing-related queries, create a service request.

Note
The AMS Accelerate team receives service requests created by you programmatically using the AWS Support API with service code `service-ams-operations-service-request`.

How service request management works

Service requests are handled by the on-call AMS Accelerate operations team.

After your service request is received by the AMS Accelerate operations team, it's reviewed to ensure that the request is properly classified as a service request or an incident. If it's reclassified as an incident, the AMS Accelerate incident management process begins and you're notified.

If the AMS Accelerate operator can resolve the service request, steps to do so are taken immediately. For example, if the service request is for architecture advice, or other information, the operator refers you to the appropriate resources or answers the question directly.

If the service request is out of scope for AMS Accelerate operations, the operator either sends the request to your cloud service delivery manager, so they can communicate with you, or to the appropriate AWS support team, along with an email to you as to what steps are being taken.

The service request is not resolved until you have indicated that you are satisfied with the outcome.

Note
We recommend you provide a contact email, name, and phone number in all cases to facilitate communications.

Working with service requests

Using the AWS Support Center, you can perform the following tasks:

- Report and update a service request. To an AMS Accelerate service request, choose **AMS Operation -- Service Request** from the **Services** menu.
- Get a list of, and detailed information about, all of your submitted service requests.
- Narrow your search for service requests by status and other filters.
- Add communications and file attachments to your requests, and add email recipients for case correspondence.
- Resolve service requests.
- Rate service request communications.

The following examples describe using the AWS Management console to create a service request for AMS Accelerate. After it's submitted, the AMS Accelerate team works with you to resolve the request per your AMS SLA.

Creating a service request

To create a service request using the AWS Management console, follow these steps:

1. Browse to **Support**.
2. Choose **Create Case**.
3. Then open the **Technical support issue type** menu, and choose **AMS Operations – Service Request**. Supply information about your service request and choose **Create**.
4. Choose a **Category**:
    - **Backup related**: Use this for any questions or requests related to backup activities.
    - **Monitoring related:** Use this for any questions or requests related to monitoring activities.
    - **Other**: Use this to request non-resource-specific help or ask a how-to question.
    - **Patch related**: Use this for any questions or requests related to patch activities.
    - **Reporting Query**: Use this to request AMS-specific report data.
    - **Resource Tagger**: Use this for any questions or requests related to Resource Tagger.
5. Choose a **Severity**:
    - **General Guidance**: Non-critical functions of your business service or application related to AWS or AMS Accelerate resources are impacted. This is the default for most service requests.
    - **System Impaired**: A non-production business service or application related to AWS/AMS Accelerate resources is moderately impacted and functioning in a degraded state due to one of the categories listed in step 4.
    - **Production System Impaired**: A production business service or application related to AWS or AMS Accelerate resources is moderately impacted and functioning in a degraded state due to one of the categories mentioned in the previous step.
    - **Production System Down**: Critical functions of a production business service or application related to AWS or AMS Accelerate resources are unavailable. In most cases, we recommend using the incident form instead.
    - **Business Critical System Down**: Your business is significantly impacted. Critical functions of your application related to AWS or AMS Accelerate resources are unavailable. In most cases, we recommend using the incident form instead.
6. Enter information for:
    - **Subject**: A descriptive title for the service request.
    - **Details**: A comprehensive description of the service request, the systems impacted, and the expected outcome of a resolution.
7. To add an attachment, choose **Add Attachment**, browse to the attachment you want, and choose **Open**. To delete the attachment, choose the delete icon ⊗
8. **Contact Method**: The default contact method is through the web. To select other options:
    - **Preferred contact language**: English is the supported language for AMS Accelerate service requests.
    - **Web**: Your service request is submitted through the web and handled by the AMS operations team.
    - **Additional contacts**: Enter any additional email addresses you want copied on your service request.
9. Choose **Submit**.

    A case details page opens with information on the service request, such as **Type**, **Subject**, **Created**, **ID**, and **Status**. Plus, a **Correspondence** area that includes the description of the request you create.

To open a correspondence area and provide additional details or updates in status, choose **Reply**.

After the service request has been resolved, choose **Resolve Case**.

If there are so many correspondences that they don't all appear on the page, choose **Load More**.

Be sure to rate the service through the 1-5 star rating to let AMS know how we're doing.

Note
If you're going to test service request functionality, we recommend you add a no-action flag to your service request's subject, such as `AMSTestNoOpsActionRequired`. Then you can test without starting the service request resolution process.
The AMS Accelerate team receives service requests created by you programmatically using the AWS Support API with service code `service-ams-operations-service-request`.

Monitoring and updating a service request

To monitor a case (incident or service request) using AWS Support Center, follow these steps.

1. In the AWS Management console, browse to **Support**.
2. From the left navigation, select **Your support cases**, browse to a case and choose the **Subject** link to open a details page with current status and correspondences.

   If you want to use phone or chat at this point, click **Open case in Support Center** to open the case **Create** page in the AWS Support Center, auto-populated with the AMS service type.

   When a reported incident or service request case is updated by the Accelerate operations team, you receive an email and a link to the incident in the Support Center so you can respond.

   Note
   You can't respond to case correspondence by replying to the email.

   If there are many cases in the dashboard, you can use the **Filter** option:

   - **Subject**: Use this filter to search on keywords in the subject of the case.
   - **Severity**: Use this to filter cases by severity by selecting a severity from the list.
   - **Case type**: Use this to see all cases of a particular case type. Accelerate incidents and service requests appear under the Technical Support Case Type along with any service-specific cases.
   - **Status**: Use this to filter cases by status by selecting a specific status from the list.
3. To check the latest status, refresh the page.
4. If there are so many correspondences that they do not all appear on the page, choose **Load More**.
5. To provide an update to the case status, choose **Reply**, enter the new correspondence, and then choose **Submit**.
6. To close out the case after it has been resolved to your satisfaction, choose **Close case**.

   Be sure to rate the service through the 1-5 star rating to let AMS know how we're doing.

Managing service requests with the support API

You can use the AWS Support API to create service requests and add correspondence to them throughout investigations of your issues and interactions with AWS support staff. The AWS support API models much of the behavior of the AWS Support Center.

The AMS team also receives service requests created by you programmatically using the AWS Support API with the service code **service-ams-operations-service-request**.

For more information about how you can use this AWS support service, see Programming the Life of an AWS Support Case.

Responding to an AMS Accelerate-generated service request

AMS Accelerate proactively monitors your resources; for more information, see Monitoring and event management in AMS Accelerate (p. 120). Sometimes AMS Accelerate creates a service request, or service notification for you, typically if action on your part is required to resolve a service request. In that case, the AMS Accelerate team sends a notification to the contact you have provided for the account. You respond to this service request in the same way as any other case—usually through the Support Center, though in some cases, email or phone correspondence is required.

> **Important**
>
> To receive state change notifications for a service request or incident case, you must have entered an email address in the addresses field. Notifications go only to the email address added to the case when it's created.

The link in the notification email works only if you're using an email server on your AMS Accelerate federated network. Otherwise, you can respond to the correspondence by going to your AMS Accelerate console and using the case details page.

> **Note**
>
> AMS Accelerate sends communications to your primary email address on your AWS account; we recommend adding an alternate operations contact email alias to facilitate the service request/notification management process. This is covered during the AMS Accelerate onboarding process and within the related onboarding documentation.

AWS Managed Services Operations On Demand

Topics

Operations on Demand is an AWS Managed Services (AMS) service feature that extends the standard scope of your AMS operations plan by providing operational services that are not currently offered natively by the AMS operations plans or AWS. Once selected, the catalog offering is delivered by a combination of automation and highly skilled AMS resources. There are no long term commitments or additional contracts, allowing you to extend your existing AMS and AWS operations and capabilities as needed. Customers agree to purchase blocks of hours (20 hours per block) on a monthly or one-time basis. Billing is block-based; unused whole blocks will not be billed.

You can select from the catalog of standardized offerings and initiate a new Operations on Demand engagement through a service request. Examples of Operations on Demand offerings include assisting with the maintenance of Amazon EKS, operations of AWS Control Tower, and management of SAP clusters. New catalog offerings are added regularly based on demand and the operational use cases we see most often.

Operations on Demand is available for both AMS Advanced and AMS Accelerate Operations Plans and is available in all AWS Regions where AMS is available.

Operations on Demand catalog of offerings

Operations on Demand offers you the services described in the following table.

Title	Description	Expected Outcomes	Operations Plan
Amazon EKS Cluster Maintenance	AMS frees your container developers by handling the ongoing maintenance of your Amazon Elastic Kubernetes Service (Amazon EKS) deployments. While Amazon EKS simplifies the provisioning, scaling, and management of Kubernetes clusters and nodes, customers are still responsible for ongoing maintenance of the underlying system. For example, the Kubernetes project releases updates	Assist customer teams with the underlying operations work of updating Amazon EKS clusters.	AMS Advanced and AMS Accelerate

	regularly and will only support branches for up to a year. AMS will handle updates to the control plane, add-ons, and nodes so that your container developers can focus on their applications.		
AWS Control Tower Operations	Ongoing operations and management of your AWS Control Tower landing zone, including AWS Transit Gateway and AWS Organizations - providing a comprehensive landing zone solution. We handle account vending, SCP and OU management, drift remediation, SSO user management, and AWS Control Tower upgrades with our library of custom controls and guardrails.	Assist customer teams with some of the underlying operations work of managing AWS Control Tower, AWS Transit Gateway, and AWS Organizations.	AMS Accelerate
SAP Cluster Assist	Dedicated alarming, monitoring, cluster patching, backup, and incident remediation for your SAP clusters. This catalog item allows you to offload some of the ongoing operational work from your SAP operations team so that they can focus on capacity management and performance tuning.	Assist customer or partner SAP teams with some of the underlying operations work. Still requires the customer to provide other SAP capabilities such as capacity management, performance tuning, DBA, and SAP basis administration.	AMS Accelerate

Legacy OS Upgrade	Avoid an instance migration by upgrading instances to a supported operating system version. We can perform an in-place upgrade on your selected instances leveraging automation and the upgrade capabilities of the software vendors (for example, Microsoft Windows 2008 R2 to Microsoft Windows 2012 R2). This approach is ideal for legacy applications that cannot be easily re-installed on a new instance and provides additional protection from known and unmitigated security threats on older OS versions.	Solution for applications that can no longer be re-installed on a new instance (e.g. lost the source code, ISV out of business, etc.). Failed upgrades can be rolled back to their original state. From an operational perspective, this is preferred as it puts the instance in a more supportable state with the latest security patches.	AMS Advanced and AMS Accelerate
Curated Change Execution	Work with our skilled operations engineers to translate your business requirements into validated change requests that can be executed safely within your AWS environment. Take advantage of our unique approach to automation and knowledge of operational best practices (e.g. impact assessment, roll backs, two-person rule), whether it is a simple change at scale or a complex action with downstream impacts.	Work with customers to define, create, and execute custom change requests. Changes can be manual or automated (CFN, SSM). Includes consultation with AWS Support for configuration guidance when necessary. Not intended for changes to application code, application installation/ deployment, data migration, or OS configuration changes.	AMS Accelerate

Priority RFC Execution	Designated AMS operations engineer capacity to prioritize the execution of your requests for change (RFC). All submissions will receive a higher level of response and priority order can be adjusted by interacting directly with engineers via an Amazon Chime meeting room.	Customers receive a response SLO of 8 hours for RFCs.	AMS Advanced

Note
For definitions of key terms refer to the AWS Managed Services Documentation Key Terms.

Requesting AMS Operations On Demand

AWS Managed Services; (AMS) Operations on Demand (Operations on Demand) is available for all AWS accounts that have been onboarded to AMS. To take advantage of Operations on Demand, request additional information from your cloud service delivery manager (CSDM), Solutions Architect (SA), account manager, or Cloud Architect (CA). Available Operations on Demand offerings are listed in the Operations on Demand catalog of offerings (p. 60). Once the engagement scoping is completed, submit a service request to AMS Operations to initiate an engagement for Operations on Demand.

Each Operations on Demand service request must contain the following detailed information pertaining to the engagement:

- The specific Operations on Demand offerings requested, and for each specific Operations on Demand offering:
  - The number of blocks (one block is equal to 20 hours of operational resource time in a given calendar month, to be charged at AWS's then-current standard rate for the applicable Operations on Demand offering) to allocate to the specific Operations on Demand offering
  - The account ID for each AWS Managed Services account for which the specific Operations on Demand offering is being requested

After the Operations on Demand service request is received, AMS Operations will review and update with their approval, partial approval, or denial.

Once approved, AMS and you coordinate to begin the engagement. No Operations on Demand offerings requested through an Operations on Demand service request are initiated until the service request is approved.

Operations on Demand service requests must be submitted by you through either:

- The AWS Managed Services account that will receive the applicable Operations on Demand offerings, or
- An AWS Managed Services account that is an AWS Organizations Management account in **all features** mode, on behalf of any of its member accounts that are AWS Managed Services accounts.

Engagements for Operations on Demand offerings begin on the first day of the first calendar month after the Operations on Demand service request is approved, except in cases where the approval occurs after the 20th day of a given calendar month, in which case the engagement begins on the first day of the second calendar month following the month in which approval occurs, unless mutually agreed by AWS and the customer.

Making changes to Operations on Demand offerings

To request changes to ongoing engagements for Operations on Demand offerings, submit a service request containing the following information:

- The modification(s) being requested, and
- The requested date for the modifications to become effective.

After receiving the Operations on Demand service request, AMS Operations reviews the request and either updates with their approval or requests that the assigned CSDM work with you to determine the scope and implications of the modification. If the modification is determined to require a scoping effort with the CSDM, you are required to submit a second Operations on Demand service request to initiate the modified engagement following the completion of the scoping exercise. Once approved, the modified engagement becomes effective on the first day of the first calendar month after the Operations on Demand service request is approved, except in cases where the approval occurs after the 20th day of a given calendar month in which case the engagement begins on the first day of the second calendar month following the month in which approval occurs, unless mutually agreed by AWS and the customer.

Reporting in AMS

Topics

AMS collates data from various native AWS services to provide value added reports on major AMS offerings.

AWS Managed Services (AMS) offers two types of detailed reporting:

- On request reporting: Certain reports can be requested ad hoc through your cloud service delivery manager (CSDM)
- Self-service reporting: You can generate some reports yourself

On-request reporting

Topics

AMS collates data from various native AWS services to provide value added reports on major AMS offerings. For a copy of these reports, make a request to your Cloud Service Delivery Manager (CSDM).

Patch reporting

Topics

Instance Details Summary

The objective of this report is to provide instance details gathered for instances that are onboarded to reporting. This is an informational report that helps identify all the instances onboarded, account status, instance details, maintenance window coverage, maintenance window execution time, stack details, and platform type.

This report provides:

1. Insights into Production and Non-Production Instances of an account. Note: Production and Non-Production stage is derived from the Account Name and not from the Instance Tags.

2. Insights into distribution of instances by platform type. Note: 'N/A' platform type is when AWS Systems Manager (SSM) is not able to get the platform information.
3. Insights into distribution of state of instances, number of instances running/stopped/terminating.

Field Name	Definition
Report Datetime	The date and time the report was generated.
Account Id	AWS Account ID to which the instance ID belongs
Account Name	AWS account name
Production Account	Identifier of AMS prod, non-prod accounts, depending on whether account name include value 'PROD', 'NONPROD'. Example: PROD, NONPROD, Not Available
Account Status	AMS account status. For example: ACTIVE, INACTIVE
AMS account service commitment	PREMIUM, PLUS
Landing Zone	Flag for account landing zone type. For example: MALZ, NON-MALZ
Access Restrictions	Regions to which access is restricted. For example: US SOIL
Instance Id	ID of EC2 instance
Instance Name	Name of EC2 instance
Instance Platform Type	Operating System (OS) type. For example: Windows, Linux, and so forth
Instance Platform Name	Operating System (OS) name. For example: MicrosoftWindowsServer2012R2Standard, RedHatEnterpriseLinuxServer
Stack Name	Name of stack that contains instance
Stack Type	AMS stack (AMS infrastructure within customer account) or Customer stack (AMS managed infrastructure that supports customer applications). Examples: AMS, CUSTOMER
Auto Scaling Group Name	Name of Auto Scaling Group (ASG) that contains the instance
Instance Patch Group	Patch group name used to group instances together and apply the same maintenance window. If the patch group is unassigned the value will be “Unassigned”
Instance Patch Group Type	Patch group type. DEFAULT: default patch group w/ default maintenance window, determined by AMSDefaultPatchGroup:True tag on the instance CUSTOMER: customer created patch group NOT_ASSIGNED: no patch group assigned

Field Name	Definition
Instance State	State within the EC2 instance lifecycle. Examples: TERMINATED, RUNNING, STOPPING, STOPPED, SHUTTING-DOWN, PENDING. For more information, see Instance lifecycle.
Maintenance Window Coverage	If there is a future Maintenance Window on this instance. Examples: COVERED or NOT_COVERED
Maintenance Window Execution Datetime	Next time the maintenance window is expected to execute. If NULL, single window execution, i.e. not recurring

Patch Details

The objective of this report is to provide patch details and maintenance window coverage of various instances.

This report provides:

1. Insights on Patch groups and its types.
2. Insights on Maintenance Windows, duration, cutoff, future dates of maintenance window executions (schedule) and instances impacted in each window.
3. Insights on all the operating systems under the account and number of instances that operating system is installed.

Field Name	Definition
Report Datetime	The date and time the report was generated.
Account Id	AWS Account ID to which the instance ID belongs
Account Name	AWS account name
Instance Id	ID of EC2 instance
Production Account	Identifier of AMS prod, non-prod accounts, depending on whether account name include value 'PROD', 'NONPROD'. If data is not available value will be “Not Available”
Account Status	AMS account status. For example: ACTIVE, INACTIVE
Instance Platform Type	Operating System (OS) type. For example: Windows, Linux
Instance Platform Name	Operating System (OS) name. For example: MicrosoftWindowsServer2012R2Standard, RedHatEnterpriseLinuxServer
Stack Type	AMS stack (AMS infrastructure within customer account) or Customer stack (AMS managed

Field Name	Definition
	infrastructure that supports customer applications). For example: AMS, CUSTOMER
Instance Patch Group	Patch group name used to group instances together and apply the same maintenance window. If the patch group is unassigned the value will be “Unassigned”
Instance Patch Group Type	Patch group type. DEFAULT: default patch group w/ default maintenance window, determined by AMSDefaultPatchGroup:True tag on the instance CUSTOMER: customer created patch group UNASSIGNED: no patch group assigned
Instance State	State within the EC2 instance lifecycle. For example: TERMINATED, RUNNING, STOPPING, STOPPED, SHUTTING-DOWN, PENDING For more information, see Instance lifecycle.
Maintenance Window Id	Maintenance window identifier
Maintenance Window State	Possible values are ENABLED or DISABLED.
Maintenance Window Type	Maintenance window type
Maintenance Window Next Execution Datetime	Next time the maintenance window is expected to execute. If NULL, single window execution, i.e. not recurring
Last Execution Maintenance Window	The latest time the maintenance window was executed
Maintenance Window Duration (hrs)	The duration of the maintenance window in hours
Maintenance Window Coverage	The maintenance window coverage
Patch Baseline Id	Patch baseline currently attached to instance
Patch Status	Overall patch compliance status. For example: COMPLIANT, NON_COMPLIANT. If there is at least one missing patch, instance is considered noncompliant, otherwise compliant.
Compliant - Total	Count of compliant patches (all severities)
Noncompliant - Total	Count of noncompliant patches (all severities)
Compliant - Critical	Count of compliant patches with "critical" severity
Compliant - High	Count of compliant patches with "high" severity
Compliant - Medium	Count of compliant patches with "medium" severity
Compliant - Low	Count of compliant patches with "low" severity
Compliant - Informational	Count of compliant patches with "informational" severity

Field Name	Definition
Compliant - Unspecified	Count of compliant patches with "unspecified" severity
Noncompliant - Critical	Count of noncompliant patches with "critical" severity
Noncompliant - High	Count of noncompliant patches with "high" severity
Noncompliant - Medium	Count of noncompliant patches with "medium" severity
Noncompliant - Low	Count of noncompliant patches with "low" severity
Noncompliant - Informational	Count of noncompliant patches with "informational" severity
Noncompliant - Unspecified	Count of noncompliant patches with "unspecified" severity

Instances that missed patches

The objective of this report is to provide details on instances that missed patches during the last maintenance window execution.

This report provides:

1. Insights on missing patches at the patch ID level.
2. Insights on all the instances which have at least one patch missing along with attributes such as patch severity, unpatched days, range, and release date of the patch.

Field Name	Definition
Report Datetime	The date and time the report was generated.
Account Id	AWS Account ID to which the instance ID belongs
Account Name	AWS account name
Production Account	Identifier of AMS prod, non-prod accounts, depending on whether the account name includes the value 'PROD', 'NONPROD'.
Account Status	AMS account status. For example: ACTIVE or INACTIVE
AMS account service tier	PREMIUM or PLUS
Instance Id	ID of EC2 instance
Instance Platform Type	Operating System (OS) type. For example: Windows
Instance State	State of the EC2 instance lifecycle. For example: TERMINATED, RUNNING, STOPPING, STOPPED,

Field Name	Definition
	SHUTTING-DOWN, PENDING For more information, see Instance lifecycle.
Patch Id	ID of released patch. For example: KB3172729
Patch Severity	Severity of patch per publisher. For example: CRITICAL, IMPORTANT, MODERATE, LOW, UNSPECIFIED
Patch Classification	Classification of patch per publisher. For example: CRITICALUPDATES, SECURITYUPDATES, UPDATEROLLUPS, UPDATES, FEATUREPACKS
Patch Release Datetime (UTC)	Release date of patch per publisher
Patch Install State	Install state of patch on instance per SSM. For example: INSTALLED, MISSING, NOT APPLICABLE
Days Unpatched	Number of days instance unpatched since last SSM scanning
Days Unpatched Range	Bucketing of days unpatched. For example: <30 DAYS, 30-60 DAYS, 60-90 DAYS, 90+ DAYS

Backup reporting

Topics

Backup snapshot success/failure

Backup snapshot success/failure reporting

This report provides:

1. Insights on number of distinct snapshots taken.
2. The backup success rate.

Field Name	Definition
Report Datetime	The date and time the report was generated
AWS Account ID	AWS Account ID to which the resource belongs
Account Name	AWS account name
Backup Type	The type of backup if there is a plan
Backup Plan Name	User defined backup plan name
Backup Vault Name	The name of the backup vault

Field Name	Definition
Resource Type	The type of resource that is being backed up
# of Resources	The number of resources that were backed up
Resource Region	The region of the backed up resource
Backup State	The state of the backup
Recovery Point ID	The unique identifier of the recovery point

Backup summary

Backup summary reporting

This report provides: Insights on important backup metrics.

Field Name	Definition
Customer Name	Customer name for situations where multiple sub-customers are
Backup Month	Month of the backup
Backup Year	Year of the backup
Resource Type	The type of resource that is being backed up
# of Resources	The number of resources that were backed up
Distinct Snapshots	Number of distinct snapshots
Backup Success Rate	The rate of successful backups
Max Snapshot Age	The maximum snapshot age
Backups Greater Than 30 Days Old	The count of backups that are over 30 days old

Backup snapshot aged

Backup snapshot aged reporting

This report provides:

1. Aging of backup snapshots.
2. Classify backup snapshots into different aging buckets.
3. Understand which resources are out of backup compliance.

Field Name	Definition
Report Datetime	The date and time the report was generated
AWS Account ID	AWS Account ID to which the resource belongs

Field Name	Definition
Account Name	AWS account name
Backup Type	The type of backup if there is a plan
Backup Plan Name	User defined backup plan name
Backup Vault Name	The name of the backup vault
Resource Type	The type of resource that is being backed up
# of Resources	The number of resources that were backed up
Resource Region	The region of the backed up resource
Backup State	The state of the backup
Recovery Point ID	The unique identifier of the recovery point
Distinct Snapshots	The number of distinct snapshots
Snapshot Age (days)	The age in days of the snapshot
Backups Greater Than 30 Days Old	The number of backups that are over 30 days old
Backups 15-30 Days Old	The number of backups that are between 15 and 30 days
Backups Less Than 15 Days Old	The number of backups that are less than 15 days old

AWS Config reporting

Topics

AWS Config reporting

Provides an in-depth look at resource and config rule compliance of AMS accounts.

This report provides:

- Insights on top non-compliant resources in your environment to discover potential threats and misconfigurations.
- Insights on compliance of resources and config rules over time.
- Insights on config rule description, recommended severity of rule, and remediation steps to fix non-compliant resources.

Field	Description
Date	Report date
Customer name	Customer name
AWS account ID	Associated AWS account ID for customer

Field	Description
Source identifier	AWS Config rule unique source identifier
Rule Description	AWS Config rule description
Rule Type	AWS Config rule type
Compliance Flag	AWS Config rule compliance state
Resource Type	AWS resource type
Resource Name	AWS resource name
Severity	Default recommended severity defined by AMS for the AWS config rule
Remediation Category	Associated remediation response category for a config rule
Remediation Description	Remediation action explained to make config rule to be compliant
Customer action	Customer action required to make the config rule to be compliant
Delta metrics report	Changes for compliance of a rule between given 2 dates

Billing reporting

Topics

AMS Billing Charges Details reporting

The objective of this report is to provide details about AMS billing charges with linked accounts and respective AWS services.

This report provides:

1. Insights on AMS service-level charges, uplift percentages, account-level AMS service tiers and AMS fees.
2. Insights on linked accounts and AWS usage charges

Field Name	Definition
Billing Month	The month and year of the service billed
Payer Account Id	The 12 digit id identifying the account that will be responsible for paying the ams charges
Linked Account Id	The 12 digit id identifying the AMS account that consumes services that generates expanses

Field Name	Definition
AWS Service Name	The AWS service that was used
AWS Charges	The AWS charges for the AWS service name in AWS Service Name
Pricing Plan	The pricing plan associated with the linked account
Uplift Proportion	The uplift percentage (as a decimal V.WXYZ) based on pricing_plan, SLA, and AWS service
Adjusted AWS Charges	AWS usage adjusted for AMS
Uplifted AWS Charges	The percentage of AWS charges to be charged for AMS; adjusted_aws_charges * uplift_percent
Instances EC2 RDS Spend	Spend on EC2 and RDS instances
AMS Charges	Total ams charges for the product; uplifted_aws_charges + instance_ec2_rds_spend + uplifted_ris + uplifted_sp
Prorated Minimum Fee	The amount we charge to meet the contractual minimum
Minimum Fee	AMS Minimum Fees (if applicable)
Linked Account Total AMS Charges	Sum of all charges for the linked_account
Payer Account Total AMS Charges	Sum of all charges for payer account

Self-service reporting

Topics

AWS Managed Services (AMS) Self-Service Reporting (SSR) feature collates data from various native AWS services and provides access to reports on major AMS offerings. It also provides the information needed to support operations, configuration management, asset management, security management and compliance.

Use SSR to access the reports from the AMS console and report datasets through S3 buckets (one bucket per account) so you can plug it into your favorite Business Intelligence (BI) tool for customizing the reports based on your unique needs. AMS creates this S3 bucket in your primary AWS Region, and the data is shared from the AMS control plane hosted in us-east-1.

Daily Patch reports

These reports provide patching details.

Topics

Instance details summary (Patch Orchestrator)

This is an informational report that helps identify all the instances onboarded to Patch Orchestrator (PO), account status, instance details, maintenance window coverage, maintenance window execution time, stack details, and platform type.

This dataset provides:

- Insights into Production and Non-Production instances of an account. Production and Non-Production stage is derived from the account name and not from the instance tags.
- Insights into distribution of instances by platform type. The 'N/A' platform type is when AWS Systems Manager (SSM) is not able to get the platform information.
- Insights into distribution of state of instances, number of instances running, stopped, or terminating.

Console Field Name	Dataset Field Name	Definition
Report Datetime	dataset_datetime	The date and time the report was generated.
Account Id	aws_account_id	AWS Account ID to which the instance ID belongs
Account Name	account_name	AWS account name
Production Account	prod_account	Identifier of AMS prod, non-prod accounts, depending on whether account name include value 'PROD', 'NONPROD'.
Account Status	account_status	AMS account status
	account_sla	AMS account service commitment
Landing Zone	malz_flag	Flag for MALZ-related account
Account Type	malz_role	MALZ role
Access Restrictions	access_restrictions	Regions to which access is restricted
Instance Id	instance_id	ID of EC2 instance
Instance Name	instance_name	Name of EC2 instance
Instance Platform Type	instance_platform_type	Operating System (OS) type
Instance Platform Name	instance_platform_name	Operating System (OS) name
Stack Name	instance_stack_name	Name of stack that contains instance

Console Field Name	Dataset Field Name	Definition
Stack Type	instance_stack_type	AMS stack (AMS infrastructure within customer account) or Customer stack (AMS managed infrastructure that supports customer applications)
Auto Scaling Group Name	instance_asg_name	Name of Auto Scaling Group (ASG) that contains the instance
Instance Patch Group	instance_patch_group	Patch group name used to group instances together and apply the same maintenance window
Instance Patch Group Type	instance_patch_group_type	Patch group type
Instance State	instance_state	State within the EC2 instance lifecycle
Maintenance Window Coverage	mw_covered_flag	If an instance has at least one enabled maintenance window with a future execution date, then it’s considered covered, otherwise not covered
Maintenance Window Execution Datetime	earliest_window_execution_time	Next time the maintenance window is expected to execute

Patch details

This report provides patch details and maintenance window coverage of various instances.

This report provides:

- Insights on Patch groups and its types.
- Insights on Maintenance Windows, duration, cutoff, future dates of maintenance window executions (schedule) and Instances impacted in each window.
- Insights on all the operating systems under the account and number of instances that operating system is installed.

Field Name	Dataset Field Name	Definition
Report Datetime	dataset_datetime	The date and time the report was generated.
Account Id	aws_account_id	AWS Account ID to which the instance ID belongs
Account Name	account_name	AWS account name
Instance Id	instance_id	ID of EC2 instance
Instance Name	instance_name	Name of EC2 instance
Production Account	prod_account	Identifier of AMS prod, non-prod accounts, depending on

Field Name	Dataset Field Name	Definition
		whether account name include value 'PROD', 'NONPROD'.
Account Status	account_status	AMS account status
	account_sla	AMS account service tier
Instance Platform Type	instance_platform_type	Operating System (OS) type
Instance Platform Name	instance_platform_name	Operating System (OS) name
Stack Type	instance_stack_type	AMS stack (AMS infrastructure within customer account) or Customer stack (AMS managed infrastructure that supports customer applications)
Instance Patch Group Type	instance_patch_group_type	DEFAULT: default patch group w/ default maintenance window, determined by AMSDefaultPatchGroup:True tag on the instance CUSTOMER: customer created patch group NOT_ASSIGNED: no patch group assigned
Instance Patch Group	instance_patch_group	Patch group name used to group instances together and apply the same maintenance window
Instance State	instance_state	State within the EC2 instance lifecycle
Maintenance Window Id	window_id	Maintenance window ID
Maintenance Window State	window_state	Maintenance window state
Maintenance Window Type	window_type	Maintenance window type
Maintenance Window Next Execution Datetime	window_next _execution_time	Next time the maintenance window is expected to execute
Last Execution Maintenance Window	last_execution_window	The latest time the maintenance window was executed
	window_next_exec_yyyy	Year part of window_next_execution_time
	window_next_exec_mm	Month part of window_next_execution_time
	window_next_exec_D	Day part of window_next_execution_time

Field Name	Dataset Field Name	Definition
	window_next _exec_HHMI	Hour:Minute part of window_next_execution_time
Maintenance Window Duration (hrs)	window_duration	The duration of the maintenance window in hours
Maintenance Window Coverage	mw_covered_flag	If an instance has at least one enabled maintenance window with a future execution date, then it’s considered covered, otherwise not covered
Patch Baseline Id	patch_baseline_id	Patch baseline currently attached to instance
Patch Status	patch_status	Overall patch compliance status. If there is at least one missing patch, instance is considered noncompliant, otherwise compliant.
Compliant - Critical	compliant_critical	Count of compliant patches with "critical" severity
Compliant - High	compliant_high	Count of compliant patches with "high" severity
Compliant - Medium	compliant_medium	Count of compliant patches with "medium" severity
Compliant - Low	compliant_low	Count of compliant patches with "low" severity
Compliant - Informational	compliant_informational	Count of compliant patches with "informational" severity
Compliant - Unspecified	compliant_unspecified	Count of compliant patches with "unspecified" severity
Compliant - Total	compliant_total	Count of compliant patches (all severities)
Noncompliant - Critical	noncompliant_critical	Count of noncompliant patches with "critical" severity
Noncompliant - High	noncompliant_high	Count of noncompliant patches with "high" severity
Noncompliant - Medium	noncompliant_medium	Count of noncompliant patches with "medium" severity
Noncompliant - Low	noncompliant_low	Count of noncompliant patches with "low" severity
Noncompliant - Informational	noncompliant _informational	Count of noncompliant patches with "informational" severity

Field Name	Dataset Field Name	Definition
Noncompliant - Unspecified	noncompliant _unspecified	Count of noncompliant patches with "unspecified" severity
Noncompliant - Total	noncompliant_total	Count of noncompliant patches (all severities)

Instances that missed patches

This report provides details on instances that missed patches during the last maintenance window execution.

This report provides:

- Insights on missing patches at the patch id level.
- Insights on all the instances which have at-least one patch missing along with attributes such as patch severity, unpatched days, range, and release date of the patch.

Field Name	Dataset Field Name	Definition
Report Datetime	dataset_datetime	The date and time the report was generated.
Account Id	aws_account_id	AWS Account ID to which the instance ID belongs
Account Name	account_name	AWS account name
Customer Name Parent	customer_name_parent	
Customer Name	customer_name	
Production Account	prod_account	Identifier of AMS prod, non-prod accounts, depending on whether account name include value 'PROD', 'NONPROD'.
Account Status	account_status	AMS account status
Account Type	account_type	
	account_sla	AMS account service tier
Instance Id	instance_id	ID of EC2 instance
Instance Name	instance_name	Name of EC2 instance
Instance Platform Type	instance_platform_type	Operating System (OS) type
Instance State	instance_state	State within the EC2 instance lifecycle
Patch Id	patch_id	ID of released patch
Patch Severity	patch_sev	Severity of patch per publisher

Field Name	Dataset Field Name	Definition
Patch Classification	patch_class	Classification of patch per publisher
Patch Release Datetime (UTC)	release_dt_utc	Release date of patch per publisher
Patch Install State	install_state	Install state of patch on instance per SSM
Days Unpatched	days_unpatched	Number of days instance unpatched since last SSM scanning
Days Unpatched Range	days_unpatched_bucket	Bucketing of days unpatched

Monthly billing report

Monthly billing report.

Billing charges details

This report provides details about AMS billing charges with linked accounts and respective AWS services.

This report provides:

- Insights on AMS service-level charges, uplift percentages, account-level AMS service tiers and AMS fees.
- Insights on linked accounts and AWS usage charges.

Field Name	Dataset Field Name	Definition
Billing Date	date	The month and year of the service billed
Payer Account Id	payer_account_id	The 12 digit id identifying the account that will be responsible for paying the ams charges
Linked Account Id	linked_account_id	The 12 digit id identifying the AMS account that consumes services that generates expanses
AWS Service Name	product_name	The AWS service that was used
AWS Charges	aws_charges	The AWS charges for the AWS service name in AWS Service Name
Pricing Plan	pricing_plan	The pricing plan associated with the linked account
AMS Service Group	tier_uplifting_groups	AMS service group code that determines uplift percentage

Field Name	Dataset Field Name	Definition
Uplift Proportion	uplift_percent	The uplift percentage (as a decimal V.WXYZ) based on pricing_plan, SLA, and AWS service
Adjusted AWS Charges	adjusted_aws_usage	AWS usage adjusted for AMS
Uplifted AWS Charges	uplifted_aws_charges	The percentage of AWS charges to be charged for AMS; adjusted_aws_charges * uplift_percent
Instances EC2 RDS Spend	instances_ec2_rds_spend	Spend on EC2 and RDS instances
Reserved Instance Charges	ris_charges	Reserved instance charges
Uplifted Reserved Instance Charges	uplifted_ris	The percentage of reserved instance charges to be charged for AMS; ris_charges * uplift_percent
Savings Plan Charges	sp_charges	SavingsPlan usage charges
Uplifted Savings Plan Charges	uplifted_sp	The percentage of savings plans charges to be charged for AMS; sp_charges * uplift_percent
AMS Charges	ams_charges	Total ams charges for the product; uplifted_aws_charges + instance_ec2_rds_spend + uplifted_ris + uplifted_sp
Prorated Minimum Fee	prorated_minimum	The amount we charge to meet the contractual minimum
Linked Account Total AMS Charges	linked_account_total _ams_charges	Sum of all charges for the linked_account
Payer Account Total AMS Charges	payer_account_total _ams_charges	Sum of all charges for payer account
Minimum Fee	minimum_fees	AMS Minimum Fees (if applicable)
Reserved Instance and Savings Plan discount	adj_ri_sp_charges	RI/SP discount to be applied against RI/SP charges (applicable under certain circumstances)

Daily backup report

This report provides details about the status of backup (success/failure) and insights into snapshots taken.

This report provides:

- Backup status
- Number of snapshots taken
- Recovery point
- Backup plan and vault information

Field Name	Dataset Field Name	Definition
Report Datetime	dataset_datetime	The date and time the report was generated.
Account Id	aws_account_id	AWS Account ID to which the instance ID belongs
Account Name	account_name	AWS account name
Account SLA	account_sla	AMS account service commitment
	malz_flag	Flag for MALZ-related account
	malz_role	MALZ role
	access_restrictions	Regions to which access is restricted
Resource ARN	resource_arn	The Amazon resource name
Resource Id	resource_id	The unique resource identifier
Resource Region	resource_region	The region of the resource
Resource Type	resource_type	The type of resource
Recovery Point ARN	recovery_point_arn	The ARN of the recovery point
Recovery Point Id	recovery_point_id	The unique identifier of the recovery point
Backup snapshot scheduled start datetime	start_by_dt_utc	Timestamp when snapshot is scheduled to begin
Backup snapshot actual start datetime	creation_dt_utc	Timestamp when snapshot actually begins
Backup snapshot completion datetime	completion_dt_utc	Timestamp when snapshot is completed
Backup snapshot expiration datetime	expiration_dt_utc	Timestamp when snapshot expires
Backup Job status	backup_job_status	State of the snapshot
Backup Type	backup_type	Type of backup
Backup Job Id	backup_job_id	The unique identifier of the backup job
Backup Size In Bytes	backup_size_in_bytes	The backup size in bytes

Field Name	Dataset Field Name	Definition
Backup Plan ARN	backup_plan_arn	The backup plan ARN
Backup Plan Id	backup_plan_id	Backup plan unique identifier
Backup Plan Name	backup_plan_name	The Backup Plan name
Backup Plan Version	backup_plan_version	The backup plan version
Backup Rule Id	backup_rule_id	The backup rule id
Backup Vault ARN	backup_vault_arn	Backup vault ARN
Backup Vault Name	backup_vault_name	The backup vault name
IAM Role ARN	iam_role_arn	The IAM role ARN
Recovery Point Status	recovery_point_status	Recovery point status
Recovery Point Delete After Days	recovery_point_delete_after_days	Recovery point delete after days
Recovery point move to cold storage after days	recovery_point_move_to_cold_storage_after_days	Number of days after completion date when backup snapshot is moved to cold storage
Recovery Point Encryption Status	recovery_point_is_encrypted	Recovery point encryption status
Recovery Point Encryption Key ARN	recovery_point_encryption_key_arn	Recovery point encryption key ARN
Volume State	volume_state	Volume State
Instance Id	instance_id	Unique instance Id
Instance State	instance_state	Instance state
Stack Id	stack_id	Cloudformation stack unique identifier
Stack Name	stack_name	Stack Name
Tag: AMS Default Patch Group	tag_ams_default_patch_group	Tag Value: AMS Default Patch Group
Tag: App Id	tag_app_id	Tag Value: App ID
Tag: App Name	tag_app_name	Tag Value: App Name
Tag: Backup	tag_backup	Tag Value: Backup
Tag: Compliance Framework	tag_compliance_framework	Tag Value: Compliance Framework
Tag: Cost Center	tag_cost_center	Tag Value: Cost Center
Tag: Customer	tag_customer	Tag Value: Customer
Tag: Data Classification	tag_data_classification	Tag Value: Data Classification

Field Name	Dataset Field Name	Definition
Tag: Environment Type	tag_environment_type	Tag Value: Environment Type
Tag: Hours of Operation	tag_hours_of_operation	Tag Value: Hours of Operation
Tag: Owner Team	tag_owner_team	Tag Value: Owner Team
Tag: Owner Team Email	tag_owner_team_email	Tag Value: Owner Team Email
Tag: Patch Group	tag_patch_group	Tag Value: Patch Group
Tag: Support Priority	tag_support_priority	Tag Value: Support Priority

Data retention policy

AMS SSR has a data retention policy per report after the period reported, the data is cleared out and no longer available.

Report name	Data Retention SSR Console	Data Retention SSR S3 Bucket
Instance Details Summary (Patch Orchestrator)	2 Months	2 Years
Patch Details	2 Months	2 Years
Instances that missed patches during maintenance window execution	2 Months	2 Years
AMS Billing Charges Details	2 Years	2 Years
Daily Backup Report	1 Month	2 Years

Offboarding from SSR

To offboard from the SSR service please create a service request (SR) through the AMS console an AMS operations engineers will help you offboard from SSR. In the ticket please provide the reason for offboarding.

If you are offboarding and account and want to do a cleanup please create an SR through the AMS console and AMS operations engineers will help delete SSR S3 bucket.

If you are leaving AMS you will automatically be offboarded from the AMS SSR console. AMS will automatically stop sending data to your account. AMS deletes your SSR S3 bucket as part of offboarding process.

Access management in AMS Accelerate

With AMS Accelerate, you manage your user lifecycle, permissions in directory services, and federated authentication system to access the AWS console or AWS APIs.

Topics

AMS Accelerate console access

When you onboard with AMS Accelerate, you automatically have access to the AMS Accelerate console. The console gives you a summarized view into the services you have with AMS Accelerate. This view includes individual components presented on the dashboard and the configuration pages. You can choose which components your different IAM roles have access to by defining an appropriate IAM policy. The following JSON document shows an example of the permissions you can assign to a role. In addition to these, you can also assign 'Describe', 'List', 'Get*' permissions for all listed services to use their native consoles.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AmsMaciePermissions",
      "Action": [
        "macie2:GetFindingStatistics"
      ],
      "Effect": "Allow",
      "Resource": "*"
    },
    {
      "Sid": "AmsGuardDutyPermissions",
      "Action": [
        "guardduty:GetFindingsStatistics",
        "guardduty:ListDetectors"
      ],
      "Effect": "Allow",
      "Resource": "*"
    },
    {
      "Sid": "AmsSupportPermissions",
      "Action": "support:*",
      "Effect": "Allow",
      "Resource": "*"
    },
    {
      "Sid": "AmsConfigPermissions",
      "Action": [
        "config:GetComplianceSummaryByConfigRule",
        "config:GetComplianceSummaryByResourceType"
      ],
      "Effect": "Allow",
      "Resource": "*"
    },
    {
      "Sid": "AmsAppConfigPermissions",
      "Action": [
        "appconfig:CreateHostedConfigurationVersion",
        "appconfig:GetConfiguration",
        "appconfig:GetDeployment",
        "appconfig:ListApplications",
        "appconfig:ListConfigurationProfiles",
        "appconfig:ListDeploymentStrategies",
        "appconfig:ListDeployments",
        "appconfig:ListEnvironments",
        "appconfig:ListHostedConfigurationVersions",
        "appconfig:StartDeployment",
        "appconfig:ValidateConfiguration"
      ],
      "Effect": "Allow",
      "Resource": "*"
    },
    {
      "Sid": "AmsCloudFormationStacksPermissions",
      "Action": [
        "cloudformation:DescribeStacks",
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

Access roles are controlled by internal group membership, to learn more, including the IAM roles that AMS uses, see Identity and Access Management (p. 102).

Granting permissions for AMS Accelerate administration

To allow your users to read and configure AMS Accelerate capabilities, like accessing the AMS Console or configuring backups, you must grant explicit permissions to their IAM roles to perform those actions. The following AWS CloudFormation template contains the policies required to read and configure services associated with AMS so you can assign them to your IAM roles. They are designed to closely align with common job responsibilities in the IT industry, where Administrator or Read-Only permissions are required; however, if you need to grant different permissions to users, you can edit the policy to include or exclude specific permissions. You can also create your own custom policy.

The **AMSAccelerateAdminAccess** policy is meant to be used for setting up and operating the AMS Accelerate components (along with **AWSBackupFullAccess**, **CloudWatchFullAccess**, and **AWSConfigUserAccess** AWS-managed policies), whereas **AMSAccelerateReadOnly** grants minimum required permissions (along with **CloudWatchReadOnlyAccess** AWS-managed policy) for viewing AMS Accelerate-related resources.

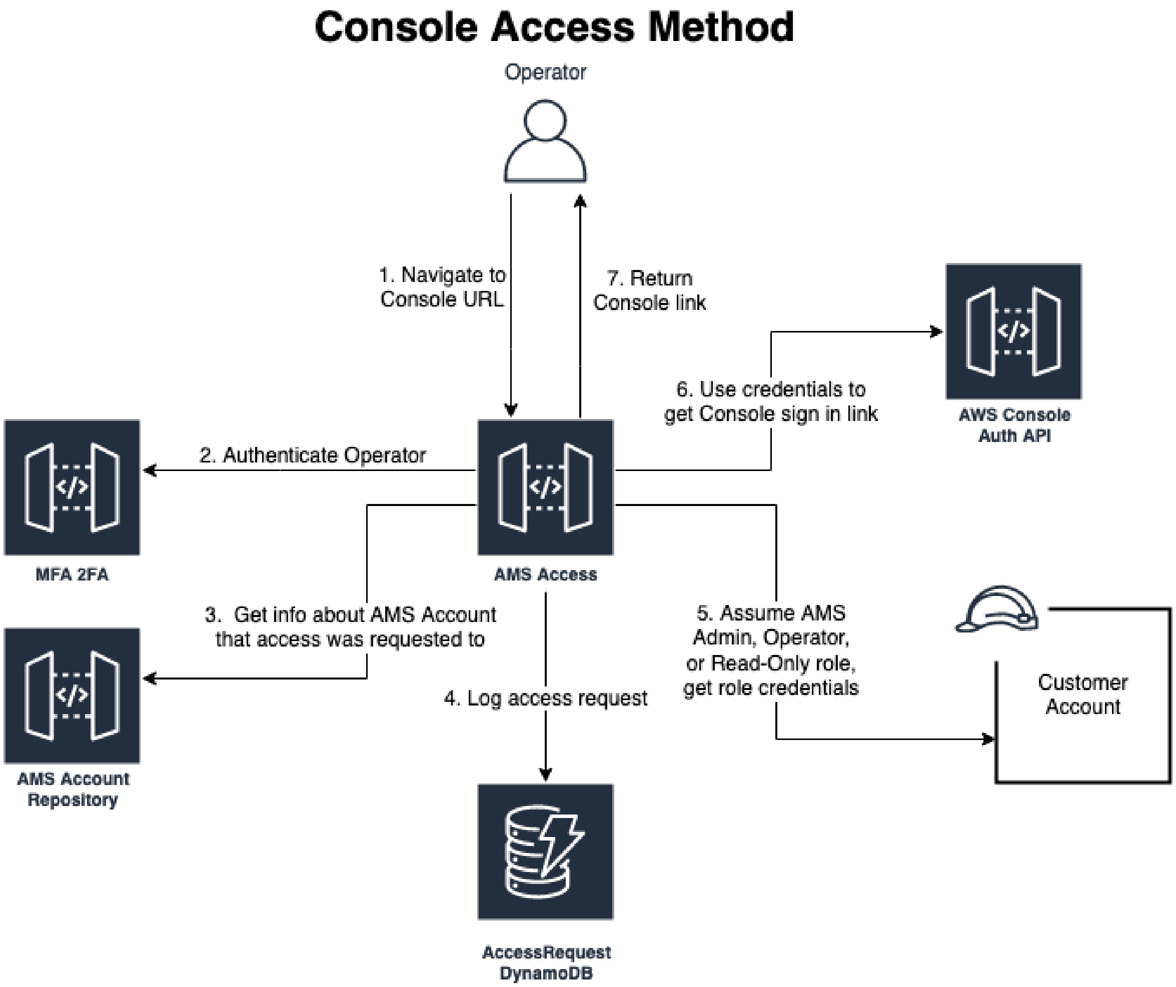
```
AWSTemplateFormatVersion: 2010-09-09
Description: AMSAccelerateCustomerAccessPolicies
```

```
Resources:
  AMSAccelerateAdminAccess:
    Type: 'AWS::IAM::ManagedPolicy'
    Properties:
      ManagedPolicyName: AMSAccelerateAdminAccess
      Path: /
      PolicyDocument:
        Fn::Sub:
        - |
          {
            "Version": "2012-10-17",
            "Statement": [
            {
                 "Sid": "AmsSelfServiceReport",
                 "Effect": "Allow",
                 "Action": "amsssrv:*",
                 "Resource": "*"
               }
              {
                "Sid": "AmsBackupPolicy",
                "Effect": "Allow",
                "Action": "iam:PassRole",
                "Resource": "arn:aws:iam::${AWS::AccountId}:role/ams-backup-iam-role"
              },
              {
                "Sid": "AmsChangeRecordKMSPolicy",
                "Effect": "Allow",
                "Action": [
                  "kms:Encrypt",
                  "kms:Decrypt",
                  "kms:GenerateDataKey"
                ],
                "Resource": [
                  "arn:aws:kms:${AWS::Region}:${AWS::AccountId}:key/*"
                ],
                "Condition": {
                  "ForAnyValue:StringLike": {
                    "kms:ResourceAliases": "alias/AMSCloudTrailLogManagement"
                  }
                }
              },
              {
                "Sid": "AmsChangeRecordAthenaReadPolicy",
                "Effect": "Allow",
                "Action": [
                  "athena:BatchGetNamedQuery",
                  "athena:Get*",
                  "athena:List*",
                  "athena:StartQueryExecution",
                  "athena:UpdateWorkGroup",
                  "glue:GetDatabase*",
                  "glue:GetTable*",
                  "s3:GetAccountPublicAccessBlock",
                  "s3:ListAccessPoints",
                  "s3:ListAllMyBuckets"
                ],
                "Resource": "*"
              },
              {
                "Sid": "AmsChangeRecordS3ReadPolicy",
                "Effect": "Allow",
                "Action": [
                  "s3:Get*",
                  "s3:List*"
                ],
                "Resource": [
                  "arn:aws:s3:::ams-a${AWS::AccountId}-athena-results-${AWS::Region}",
                  "arn:aws:s3:::ams-a${AWS::AccountId}-athena-results-${AWS::Region}/*",
                  "arn:aws:s3:::ams-a${AWS::AccountId}-cloudtrail-${AWS::Region}",
                  "arn:aws:s3:::ams-a${AWS::AccountId}-cloudtrail-${AWS::Region}/*"
                ]
              },
              {
                "Sid": "AmsChangeRecordS3WritePolicy",
                "Effect": "Allow",
                "Action": [
                  "s3:PutObject",
                  "s3:PutObjectLegalHold",
                  "s3:PutObjectRetention"

                ],
                "Resource": [
                  "arn:aws:s3:::ams-a${AWS::AccountId}-athena-results-${AWS::Region}/*"
                ]
              },
              {
                "Sid": "MaciePolicy",
                "Effect": "Allow",
                "Action": [
                  "macie2:GetFindingStatistics"
                ],
                "Resource": "*"
              },
              {
                "Sid": "GuardDutyPolicy",
                "Effect": "Allow",
                "Action": [
                  "guardduty:GetFindingsStatistics",
                  "guardduty:ListDetectors"
                ],
                "Resource": "*"
              },
              {
                "Sid": "SupportPolicy",
                "Effect": "Allow",
                "Action": "support:*",
                "Resource": "*"
              },
              {
                "Sid": "ConfigPolicy",
                "Effect": "Allow",
                "Action": [
                  "config:Get*",
                  "config:Describe*",
                  "config:Deliver*",
                  "config:List*",
                  "config:StartConfigRulesEvaluation"
                ],
                "Resource": "*"
              },
              {
                "Sid": "AppConfigReadPolicy",
                "Effect": "Allow",
                "Action": [
                  "appconfig:List*",
                  "appconfig:Get*"
                ],
                "Resource": "*"
              },
              {
                "Sid": "AppConfigPolicy",
                "Effect": "Allow",
                "Action": [
                  "appconfig:StartDeployment",
                  "appconfig:StopDeployment",
                  "appconfig:CreateHostedConfigurationVersion",
                  "appconfig:ValidateConfiguration"
                ],
                "Resource": [
                  "arn:aws:appconfig:*:${AWS::AccountId}:application/
${AMSAlarmManagerConfigurationApplicationId}",
                  "arn:aws:appconfig:*:${AWS::AccountId}:application/
${AMSAlarmManagerConfigurationApplicationId}/configurationprofile/
${AMSAlarmManagerConfigurationCustomerManagedAlarmsProfileID}",
                  "arn:aws:appconfig:*:${AWS::AccountId}:application/
${AMSAlarmManagerConfigurationApplicationId}/environment/*",
                  "arn:aws:appconfig:*:${AWS::AccountId}:application/
${AMSResourceTaggerConfigurationApplicationId}",
                  "arn:aws:appconfig:*:${AWS::AccountId}:application/
${AMSResourceTaggerConfigurationApplicationId}/configurationprofile/
${AMSResourceTaggerConfigurationCustomerManagedTagsProfileID}",
                  "arn:aws:appconfig:*:${AWS::AccountId}:application/
${AMSResourceTaggerConfigurationApplicationId}/environment/*",
                  "arn:aws:appconfig:*:${AWS::AccountId}:deploymentstrategy/*"
                ]
              },
              {
                "Sid": "CloudFormationStacksPolicy",
                "Effect": "Allow",
                "Action": [
                  "cloudformation:DescribeStacks"
                ],
                "Resource": "*"
              },
              {
                "Sid": "EC2Policy",
                "Action": [
                  "ec2:DescribeInstances"
                ],
                "Effect": "Allow",
                "Resource": "*"
              },
              {
                "Sid": "SSMPolicy",
                "Effect": "Allow",
                "Action": [
                  "ssm:AddTagsToResource",
                  "ssm:CancelCommand",
                  "ssm:CancelMaintenanceWindowExecution",
                  "ssm:CreateAssociation",
                  "ssm:CreateAssociationBatch",
                  "ssm:CreateMaintenanceWindow",
                  "ssm:CreateOpsItem",
                  "ssm:CreatePatchBaseline",
                  "ssm:DeleteAssociation",
                  "ssm:DeleteMaintenanceWindow",
                  "ssm:DeletePatchBaseline",
                  "ssm:DeregisterPatchBaselineForPatchGroup",
                  "ssm:DeregisterTargetFromMaintenanceWindow",
                  "ssm:DeregisterTaskFromMaintenanceWindow",
                  "ssm:Describe*",
                  "ssm:Get*",
                  "ssm:List*",
                  "ssm:PutConfigurePackageResult",
                  "ssm:RegisterDefaultPatchBaseline",
                  "ssm:RegisterPatchBaselineForPatchGroup",
                  "ssm:RegisterTargetWithMaintenanceWindow",
                  "ssm:RegisterTaskWithMaintenanceWindow",
                  "ssm:RemoveTagsFromResource",
                  "ssm:SendCommand",
                  "ssm:StartAssociationsOnce",
                  "ssm:StartAutomationExecution",
                  "ssm:StartSession",
                  "ssm:StopAutomationExecution",
                  "ssm:TerminateSession",
                  "ssm:UpdateAssociation",
                  "ssm:UpdateAssociationStatus",
                  "ssm:UpdateMaintenanceWindow",
                  "ssm:UpdateMaintenanceWindowTarget",
                  "ssm:UpdateMaintenanceWindowTask",
                  "ssm:UpdateOpsItem",
                  "ssm:UpdatePatchBaseline"
                ],
                "Resource": "*"
              },
              {
                "Sid": "AmsPatchRestrictAMSResources",
                "Effect": "Deny",
                "Action": [
                  "ssm:DeletePatchBaseline",
                  "ssm:UpdatePatchBaseline"
                ],
                "Resource": [
                  "arn:aws:ssm:${AWS::Region}:${AWS::AccountId}:patchbaseline/*"
                ],
                "Condition": {
                  "StringLike": {
                    "aws:ResourceTag/ams:resourceOwner": "*"
                  }
                }
              },
              {
                "Sid": "AmsPatchRestrictAmsTags",
                "Effect": "Deny",
                "Action": [
                  "ssm:AddTagsToResource",
                  "ssm:RemoveTagsFromResource"
                ],
                "Resource": "*",
                "Condition": {
                  "ForAnyValue:StringLike": {
                    "aws:TagKeys": [
                      "AMS*",
                      "Ams*",
                      "ams*"
                    ]
                  }
                }
              },
              {
                "Sid": "TagReadPolicy",
                "Effect": "Allow",
                "Action": [
                  "tag:GetResources",
                  "tag:GetTagKeys"
                ],
                "Resource": "*"
              },
              {
                "Sid": "CloudtrailReadPolicy",
                "Effect": "Allow",
                "Action": [
                  "cloudtrail:DescribeTrails",
                  "cloudtrail:GetTrailStatus",
                  "cloudtrail:LookupEvents"
                ],
                "Resource": "*"
              },
              {
                "Sid": "EventBridgePolicy",
                "Effect": "Allow",
                "Action": [
                  "events:Describe*",
                  "events:List*",
                  "events:TestEventPattern"
                ],
                "Resource": "*"
              }
            ]
          }
        - AMSAlarmManagerConfigurationApplicationId: !ImportValue "AMS-Alarm-Manager-
Configuration-ApplicationId"
          AMSAlarmManagerConfigurationCustomerManagedAlarmsProfileID: !ImportValue "AMS-
Alarm-Manager-Configuration-CustomerManagedAlarms-ProfileID"
          AMSResourceTaggerConfigurationApplicationId: !ImportValue "AMS-ResourceTagger-
Configuration-ApplicationId"
          AMSResourceTaggerConfigurationCustomerManagedTagsProfileID: !ImportValue "AMS-
ResourceTagger-Configuration-CustomerManagedTags-ProfileID"

  AMSAccelerateReadOnly:
    Type: 'AWS::IAM::ManagedPolicy'
    Properties:
      ManagedPolicyName: AMSAccelerateReadOnly
      Path: /
      PolicyDocument: !Sub |
        {
          "Version": "2012-10-17",
          "Statement": [
          {
                 "Sid": "AmsSelfServiceReport",
                 "Effect": "Allow",
                 "Action": "amsssrv:*",
                 "Resource": "*"
               }
            {
               "Sid": "AmsBackupPolicy",
               "Effect": "Allow",
               "Action": [
                 "backup:Describe*",
                 "backup:Get*",
                 "backup:List*"
               ],
               "Resource": "*"
            },
            {
                "Action": [
                    "rds:DescribeDBSnapshots",
                    "rds:ListTagsForResource",
                    "rds:DescribeDBInstances",
                    "rds:describeDBSnapshots",
                    "rds:describeDBEngineVersions",
                    "rds:describeOptionGroups",
                    "rds:describeOrderableDBInstanceOptions",
                    "rds:describeDBSubnetGroups",
                    "rds:DescribeDBClusterSnapshots",
                    "rds:DescribeDBClusters",
                    "rds:DescribeDBParameterGroups",
                    "rds:DescribeDBClusterParameterGroups",
                    "rds:DescribeDBInstanceAutomatedBackups"
                ],
                "Effect": "Allow",
                "Resource": "*"
            },
            {
                "Action": [
                    "dynamodb:ListBackups",
                    "dynamodb:ListTables"
                ],
                "Effect": "Allow",
                "Resource": "*"
            },
            {
                "Action": [
                    "elasticfilesystem:DescribeFilesystems"
                ],
                "Resource": "arn:aws:elasticfilesystem:*:*:file-system/*",
                "Effect": "Allow"
            },
            {
                "Action": [
                    "ec2:DescribeSnapshots",
                    "ec2:DescribeVolumes",
                    "ec2:describeAvailabilityZones",
                    "ec2:DescribeVpcs",
                    "ec2:DescribeAccountAttributes",
                    "ec2:DescribeSecurityGroups",
                    "ec2:DescribeImages",
                    "ec2:DescribeSubnets",
                    "ec2:DescribePlacementGroups",
                    "ec2:DescribeInstances",
                    "ec2:DescribeInstanceTypes"
                ],
                "Effect": "Allow",
                "Resource": "*"
            },
            {
                "Action": [
                    "tag:GetTagKeys",
                    "tag:GetTagValues",
                    "tag:GetResources"
                ],
                "Effect": "Allow",
                "Resource": "*"
            },
            {
                "Effect": "Allow",
                "Action": [
                    "storagegateway:DescribeCachediSCSIVolumes",
                    "storagegateway:DescribeStorediSCSIVolumes"
                ],
                "Resource": "arn:aws:storagegateway:*:*:gateway/*/volume/*"
            },
            {
                "Effect": "Allow",
                "Action": [
                    "storagegateway:ListGateways"
                ],
                "Resource": "arn:aws:storagegateway:*:*:*"
            },
            {
                "Effect": "Allow",
                "Action": [
                    "storagegateway:DescribeGatewayInformation",
                    "storagegateway:ListVolumes",
                    "storagegateway:ListLocalDisks"
                ],
                "Resource": "arn:aws:storagegateway:*:*:gateway/*"
            },
            {
                "Action": [
                    "iam:ListRoles",
                    "iam:GetRole"
                ],
                "Effect": "Allow",
                "Resource": "*"
            },
            {
                "Effect": "Allow",
                "Action": "organizations:DescribeOrganization",
                "Resource": "*"
            },
            {
                "Action": "fsx:DescribeBackups",
                "Effect": "Allow",
                "Resource": "arn:aws:fsx:*:*:backup/*"
            },
            {
                "Action": "fsx:DescribeFileSystems",
                "Effect": "Allow",
                "Resource": "arn:aws:fsx:*:*:file-system/*"
            },
            {
                "Action": "ds:DescribeDirectories",
                "Effect": "Allow",
                "Resource": "*"
            },
            {
              "Sid": "AmsChangeRecordKMSPolicy",
              "Effect": "Allow",
              "Action": [
                "kms:Encrypt",
                "kms:Decrypt",
                "kms:GenerateDataKey"
              ],
              "Resource": [
                "arn:aws:kms:${AWS::Region}:${AWS::AccountId}:key/*"
              ],
              "Condition": {
                "ForAnyValue:StringLike": {
                  "kms:ResourceAliases": "alias/AMSCloudTrailLogManagement"
                }
              }
            },
            {
              "Sid": "AmsChangeRecordAthenaReadPolicy",
              "Effect": "Allow",
              "Action": [
                "athena:BatchGetNamedQuery",
                "athena:Get*",
                "athena:List*",
                "athena:StartQueryExecution",
                "athena:UpdateWorkGroup",
                "glue:GetDatabase*",
                "glue:GetTable*",
                "s3:GetAccountPublicAccessBlock",
                "s3:ListAccessPoints",
                "s3:ListAllMyBuckets"
              ],
              "Resource": "*"
            },
            {
```

```
      "Sid": "AmsChangeRecordS3ReadPolicy",
      "Effect": "Allow",
      "Action": [
        "s3:Get*",
        "s3:List*"
      ],
      "Resource": [
        "arn:aws:s3:::ams-a${AWS::AccountId}-athena-results-${AWS::Region}",
        "arn:aws:s3:::ams-a${AWS::AccountId}-athena-results-${AWS::Region}/*",
        "arn:aws:s3:::ams-a${AWS::AccountId}-cloudtrail-${AWS::Region}",
        "arn:aws:s3:::ams-a${AWS::AccountId}-cloudtrail-${AWS::Region}/*"
      ]
    },
    {
      "Sid": "AmsChangeRecordS3WritePolicy",
      "Effect": "Allow",
      "Action": [
        "s3:PutObject",
        "s3:PutObjectLegalHold",
        "s3:PutObjectRetention"
      ],
      "Resource": [
        "arn:aws:s3:::ams-a${AWS::AccountId}-athena-results-${AWS::Region}/*"
      ]
    },
    {
      "Sid": "MaciePolicy",
      "Effect": "Allow",
      "Action": [
        "macie2:GetFindingStatistics"
      ],
      "Resource": "*"
    },
    {
      "Sid": "GuardDutyReadPolicy",
      "Effect": "Allow",
      "Action": [
        "guardduty:GetFindingsStatistics",
        "guardduty:ListDetectors"
      ],
      "Resource": "*"
    },
    {
      "Sid": "SupportReadPolicy",
      "Effect": "Allow",
      "Action": "support:Describe*",
      "Resource": "*"
    },
    {
      "Sid": "ConfigReadPolicy",
      "Effect": "Allow",
      "Action": [
        "config:Get*",
        "config:Describe*",
        "config:List*"
      ],
      "Resource": "*"
    },
    {
      "Sid": "AppConfigReadPolicy",
      "Effect": "Allow",
      "Action": [
        "appconfig:List*",
        "appconfig:Get*"
      ],
      "Resource": "*"
      },
      {
       "Sid": "CloudFormationReadPolicy",
       "Effect": "Allow",
       "Action": [
        "cloudformation:DescribeStacks"
       ],
       "Resource": "*"
      },
      {
       "Sid": "EC2ReadPolicy",
       "Effect": "Allow",
       "Action": [
        "ec2:DescribeInstances"
       ],
       "Resource": "*"
      },
      {
       "Sid": "SSMReadPolicy",
       "Effect": "Allow",
       "Action": [
        "ssm:Describe*",
        "ssm:Get*",
        "ssm:List*"
       ],
       "Resource": "*"
      },
      {
       "Sid": "TagReadPolicy",
       "Effect": "Allow",
       "Action": [
        "tag:GetResources",
        "tag:GetTagKeys"
       ],
       "Resource": "*"
      },
      {
       "Sid": "CloudtrailReadPolicy",
       "Effect": "Allow",
       "Action": [
        "cloudtrail:DescribeTrails",
        "cloudtrail:GetTrailStatus",
        "cloudtrail:LookupEvents"
       ],
       "Resource": "*"
      },
      {
       "Sid": "EventBridgePolicy",
       "Effect": "Allow",
       "Action": [
        "events:Describe*",
        "events:List*",
        "events:TestEventPattern"
       ],
       "Resource": "*"
      }
     ]
    }
```

AMS Accelerate operator access

AMS Accelerate operators can access your account console and instances, in certain circumstances.


Console Access Method
Operator
1. Navigate to Console URL
7. Return Console link
6. Use credentials to get Console sign in link
AWS Console Auth API
2. Authenticate Operator
MFA 2FA
AMS Access
3. Get info about AMS Account that access was requested to
AMS Account Repository
4. Log access request
5. Assume AMS Admin, Operator, or Read-Only role, get role credentials
Customer Account
AccessRequest DynamoDB

AMS Accelerate operator console access

Various AMS Accelerate personnel, such as operations engineers and cloud architects (CAs), occasionally require access to your accounts to respond to service requests or reported incidents. AMS Accelerate access is governed by an internal AMS Accelerate access service that enforces controls on access. These controls include, business justification (current supported business justifications include: service requests, OpsItems, support cases). Access defaults to read only, and access is tracked and recorded. Access roles are controlled by internal group membership, which is controlled by AMS Accelerate Operations management and periodically reviewed. The following IAM roles are used:

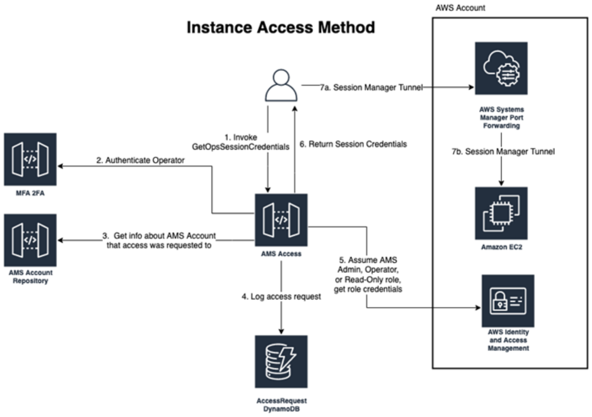
- **ams-access-admin**: The AMS Accelerate admin role has full permissions to operate in your account without restrictions. AMS Accelerate feature services (with scoped down session policy), and only a few select individuals can assume the admin role.
- **ams-access-operations**: The AMS Accelerate operations role has full permissions to operate in your account with the exception of IAM write permissions. Individuals with certain group membership can assume this role.
- **ams-access-read-only**: The AMS Accelerate read-only role has read-only permissions in your account and is available to AMS Accelerate Operations and AMS Accelerate cloud architects (CAs).

AMS Accelerate personnel can assume one of the previously mentioned AMS Accelerate IAM roles deployed in your account:

- Through direct federation into the AWS Management Console to perform manual, browser-based work, such as host access through SSM session manager to an Amazon EC2 instance or apply SSM documents from the SSM console or OpsCenter.
- By obtaining session credentials, with scoped down session policy, to programmatically (with AWS APIs) interact with the account and the resources within such as applying AWS SSM documents or deploying AWS resources.

AWS instance access

Access to your instances within your account is managed in much the same way as internal account access. The internal AMS Accelerate access service is used as the broker to instance access and, after access is granted, the AMS Accelerate operator uses SSM session manager to gain access by using session credentials.


AWS Account
Instance Access Method
7a. Session Manager Tunnel
AWS Systems Manager Port Forwarding
1. Invoke GetOpsSessionCredentials
6. Return Session Credentials
7b. Session Manager Tunnel
2. Authenticate Operator
MFA 2FA
Amazon EC2
3. Get info about AMS Account that access was requested to
AMS Access
AMS Account Repository
5. Assume AMS Admin, Operator, or Read-Only role, get role credentials
4. Log access request
AWS Identity and Access Management
AccessRequest DynamoDB

How and when to use the root user account

AWS Managed Services (AMS) Security and Operations provide robust security of customer accounts. The "root user" account is the superuser, or administrator, account within your AWS account, and its use is strongly discouraged and watched by AMS. However, there are some tasks that require root access including changing your account settings, activating AWS Identity and Access Management (IAM) access to billing and cost management, changing your root password, and enabling multi-factor authentication (MFA). Root should not be used otherwise. For more information on when to use the root user account, see Tasks that require root user credentials. For information about how MFA is configured, see Secure New Account with Multi-Factor Authentication

> **Note**
> MFA is created at onboarding to specifically disallow root access. Root access in AMS accounts is different from other AWS accounts, and is critical to the security of your entire AMS-managed environment. When MFA is configured, the token is immediately deleted, ensuring that neither you nor AMS retains the ability to log in as root. AMS expects such access to be used only when absolutely necessary.

When root access is required, the process varies slightly between AMS account types but always triggers an AMS Security and Operations team response. AMS monitors API calls for root access, and alarms are triggered if such access is detected.

Root with AMS Advanced single-account landing zone:

If you have a single-account landing zone, contact your cloud service deliver manager (CSDM) and cloud architects (CAs) to advise them of the root access work that you require. It is best to give twenty-four hours notice before the proposed activity.

Root with AMS Advanced multi-account landing zone:

For multi-account landing zone Application, Shared Services, Security, or Networking accounts, use the Management | Other | Other (ct-1e1xtak34nx76) change type. Include the date, time, and the purpose of using the root user credentials and schedule the RFC to be sure to give twenty-four hours notice before the proposed activity. Use your multi-account landing zone Management account to submit the RFC.

Additionally, contact your CSDM and CAs twenty-four hours in advance, to advise them of the root access work you require.

Root with AMS Accelerate:

As an AMS Accelerate account, AMS cannot prohibit you from using your root user account. However, AMS Operations and Security does treat its usage as an issue to investigate and we will reach out to your Security team with every use.

If you have an AMS Accelerate account, contact your CSDM and CAs twenty-four hours in advance, to advise them of the root access work you require.

To learn about AWS root user account usage, see AWS account root user.

AMS operations and security response to root usage:

The AMS Operations team receives an alarm when the root user account is used. If the root credentials usage is unscheduled, they contact the AMS Security team, and your account team, to verify if this is expected activity. If it is not expected activity, AMS works with your Security team.

Security in AMS Accelerate

AWS Managed Services protects your information assets and helps keep your AWS infrastructure secure by using multiple controls. AMS Accelerate maintains a library of AWS Config Rules and remediation actions to ensure that all your accounts comply with industry standards for security and operational integrity. AWS Config Rules test every configuration change among your all resources. If a change violates any rule conditions, AMS reports its findings, and allows you to remediate violations automatically or by request, according to the severity of the violation. AWS Config Rules facilitate compliance with standards set by: the Center for Internet Security (CIS), the National Institute of Standards and Technology (NIST) Cloud Security Framework (CSF), the Health Insurance Portability and Accountability Act (HIPAA), and the Payment Card Industry (PCI) Data Security Standard (DSS).

In addition, AMS leverages Amazon GuardDuty to identify potentially unauthorized or malicious activity in your AWS environment. GuardDuty findings are monitored 24x7 by AMS. AMS collaborates with you to understand the impact of the findings and remediations based on best practice recommendations. AMS also uses Amazon Macie to protect your sensitive data such as personal health information (PHI), personally identifiable information (PII) and financial data.

> **Note**
> Amazon Macie is an optional service and is not enabled by default.

Infrastructure security

During onboarding, AMS Accelerate deploys the following AWS Config baseline infrastructure and set of rules that AMS Accelerate uses to monitor your accounts:

- **AWS Config service-linked role**: AMS Accelerate deploys the service-linked role named **AWSServiceRoleForConfig**, which is used by AWS Config to query the status of other AWS services. The **AWSServiceRoleForConfig** service-linked role trusts the AWS Config service to assume the role. The permissions policy for the **AWSServiceRoleForConfig** role contains read-only and write-only permissions on AWS Config resources and read-only permissions for resources in other services that AWS Config supports. If you already have a role configured with AWS Config Recorder, AMS Accelerate validates that the existing role has an AWS Config managed-policy attached. If not, AMS Accelerate replaces the role with the service-linked role **AWSServiceRoleForConfig**.
- **AWS Config recorder and delivery channel**: AWS Config uses the configuration recorder to detect changes in your resource configurations and capture these changes as configuration items. AMS Accelerate deploys the configuration recorder in all service AWS Regions, with recording of all resources. AMS Accelerate also creates the config delivery channel, an Amazon S3 bucket, which is used to record changes that occur in your AWS resources; it updates configuration states through the delivery channel. The config recorder and delivery channel are required for AWS Config to work. AMS Accelerate creates the recorder in all AWS Regions, and a delivery channel in a single AWS Region. If you already have a recorder and delivery channel in an AWS Region, AMS Accelerate does not delete the existing AWS Config resources, instead AMS Accelerate utilizes your existing recorder and delivery channel after validating that they are properly configured.
- **AWS Config rules**: AMS Accelerate maintains a library of AWS Config Rules and remediation actions to ensure that all your accounts comply with industry standards for security and operational integrity. AWS Config Rules test every configuration change among your all resources. If a change violates any rule conditions, AMS reports its findings, and allows you to remediate violations automatically or by request, according to the severity of the violation. AWS Config Rules facilitate compliance with standards set by: the Center for Internet Security (CIS), the National Institute of Standards and Technology (NIST) Cloud Security Framework (CSF), the Health Insurance Portability and Accountability Act (HIPAA), and the Payment Card Industry (PCI) Data Security Standard (DSS).

- **AWS Config aggregator authorization**: An aggregator is an AWS Config resource type that collects AWS Config configuration and compliance data from multiple accounts and multiple Regions. AMS Accelerate onboards your account to a config aggregator from which AMS Accelerate aggregates your account's resource configuration information and config compliance data and generates the compliance report. If there are existing aggregators configured in the AMS-owned account, AMS Accelerate deploys an additional aggregator and the existing aggregator is not modified.

  Note
  The Config aggregator is not set up in your accounts; rather, it is set up in AMS-owned accounts and your account(s) are onboarded to it.

To learn more about AWS Config, see:

- AWS Config: What Is Config?
- AWS Config Rules: Evaluating Resources with Rules
- AWS Config Rules: Dynamic Compliance Checking: AWS Config Rules – Dynamic Compliance Checking for Cloud Resources
- AWS Config Aggregator: Multi-Account Multi-Region Data Aggregation

For information on reports, see AWS Config reporting (p. 72).

Data protection

AMS Accelerate continuously monitors your managed accounts by leveraging native AWS services such as Amazon GuardDuty, Amazon Macie (optionally), and other internal proprietary tools and processes. After an alarm is triggered, AMS Accelerate assumes responsibility for the initial triage and response to the alarm. Our response processes are based on NIST standards. AMS Accelerate regularly tests its response processes using Security Incident Response Simulation with you to align your workflow with existing customer security response programs.

When AMS Accelerate detects any violation, or imminent threat of violation, of AWS or your security policies, we gather information, including impacted resources and any configuration-related changes. AMS Accelerate provides 24/7/365 follow-the-sun support with dedicated operators actively reviewing and investigating monitoring dashboards, incident queue, and service requests across all of your managed accounts. AMS Accelerate investigates the findings with our security experts to analyze the activity and notify you through the security escalation contacts listed in your account.

Based on our findings, AMS Accelerate engages with you proactively. If you believe the activity is unauthorized or suspicious, AMS works with you to investigate and remediate or contain the issue. There are certain finding types generated by GuardDuty that require you to confirm the impact before AMS Accelerate is able to take any action. For example, the GuardDuty finding type **UnauthorizedAccess:IAMUser/ConsoleLogin**, indicates that one of your users has logged in from an unusual location; AMS notifies you and asks that you review the finding to confirm if this behavior is legitimate.

Amazon Macie

We recommend, and AMS Accelerate supports, Macie to detect a large and comprehensive list of sensitive data, such as personal health information (PHI), personally identifiable information (PII), and financial data.

Macie can be configured to run periodically on any Amazon S3 bucket, automating the evaluation of any new or modified objects within a bucket over time. As security findings are generated, AMS will notify you and work with you to remediate as needed.

For more information, see Analyzing Amazon Macie findings.

GuardDuty

GuardDuty is a continuous security monitoring service that uses threat intelligence feeds, such as lists of malicious IP addresses and domains, and machine learning to identify unexpected and potentially unauthorized and malicious activity within your AWS environment. This can include issues like escalations of privileges, uses of exposed credentials, or communication with malicious IP addresses, or domains. GuardDuty also monitors AWS account access behavior for signs of compromise, such as unauthorized infrastructure deployments, like instances deployed in a Region that has never been used, or unusual API calls, like a password policy change to reduce password strength. For more information, refer to the GuardDuty User Guide.

To view and analyze your GuardDuty findings, use the following procedure.

1. Open the GuardDuty console.
2. Choose **Findings**, and then choose a specific finding to view details. The details for each finding differ depending on the finding type, resources involved, and nature of the activity.

For more information on available finding fields, see GuardDuty finding details.

GuardDuty suppression rules

A suppression rule is a set of criteria, consisting of a filter attribute paired with a value, used to filter findings by automatically archiving new findings that match the specified criteria. Suppression rules can be used to filter low-value findings, false positive findings, or known activities, you do not intend to act on, to make it easier to recognize the security threats with the most impact to your environment.

AMS has a defined set of criteria to identify suppression rules for your managed accounts. AMS implements suppression rule to filter false positive findings and reduce frequent unactionable notifications. When a managed account meets this criteria, AMS will apply the filters and notify you with the details of the suppression filter deployed via a service request (SR).

You can communicate with AMS via a service request (SR) to modify or revert the suppression filters.

Suppressed findings are not sent to AWS Security Hub, Amazon S3, or CloudTrail Events, reducing finding of unactionable data if you consume GuardDuty findings via Security Hub or a third-party SIEM, alerting and ticketing applications.

GuardDuty continues to generate findings even when they match your suppression rules, however, those findings automatically marked as **archived**. The archived finding is stored in GuardDuty for 90-days and can be viewed at any time during that period. You can view suppressed findings in the GuardDuty console by selecting **Archived** from the findings table, or through the GuardDuty API using the ListFindings API with a **findingCriteria** of **service.archived equal** to **true**.

Common Use Cases for Suppression Rules:

The following are finding types with common use cases for applying suppression rules, select the finding name to learn more about how to apply a suppression rules for that use case.

- **Recon:EC2/Portscan**: Use a suppression rule to automatically archive findings when using an autorized vulnerability scanner.
- **UnauthorizedAccess:EC2/SSHBruteForce**: Use a suppression rule to automatically archive findings when it is targeted to bastion instances.
- **Recon:EC2/PortProbeUnprotectedPort**: Use a suppression rule to automatically archive findings when it is targeted to intentionally exposed instances.

Data encryption

AMS Accelerate uses several AWS services for data encryption.

Amazon Simple Storage Service offers several object encryption options that protect data in transit and at rest. Server-side encryption encrypts your object before saving it on disks in its data centers and then decrypts it when you download the objects. As long as you authenticate your request and you have access permissions, there is no difference in the way you access encrypted or unencrypted objects. For more information, see Data protection in Amazon S3.

Identity and Access Management

AWS Identity and Access Management is a web service that helps you securely control access to AWS resources. You use IAM to control who is authenticated (signed in) and authorized (has permissions) to use resources. During AMS Accelerate onboarding, you are responsible for creating cross-account IAM administrator roles within each of your managed accounts.

With AMS Accelerate, you're responsible for managing access to your AWS accounts and their underlying resources, such as access management solutions, access policies, and related processes. This means that you manage your user lifecycle, permissions in directory services, and federated authentication system, to access the AWS console or AWS APIs. In order to help you manage your access solution, AMS Accelerate deploys AWS Config rules that detect common IAM misconfigurations, and then deliver remediation notifications. For more information, see AWS Config Managed Rules.

Authenticating with identities

AMS uses IAM roles, which is a type of IAM identity. An IAM role is very similar to a user, in that it is an identity with permissions policies that determine what the identity can and cannot do in AWS. However, a role doesn't have credentials associated with it and, instead of being uniquely associated with one person, a role is intended to be assumable by anyone who needs it. An IAM user can assume a role to temporarily take on different permissions for a specific task.

Access roles are controlled by internal group membership, which is administered and periodically reviewed by Operations Management. AMS uses the following IAM roles:

Role name	Description
Used by (entity): **AMS Access Service only**	
ams-access-management	Deployed manually by you during onboarding. Assumed only by AMS access to deploy or update access roles. Remains in your account after onboarding for any future updates to the access roles.
Used by (entity): **AMS Operations**	
ams-access-operations	This AMS operations role has full privileges to operate in your AMS account, with the exception of IAM write permissions.
ams-access-read-only	This AMS read-only role is limited to read-only permissions in your AMS account.
Used by (entity): **AMS Operations and AMS Services**	

Role name	Description
ams-access-admin	This AMS admin role has full permissions to operate in accounts without restrictions. Only AMS internal services (with a scoped-down session policy) and only a very few select AMS individuals can assume the admin role.
customer_ssm_automation_role	Assumed by AWS Systems Manager to execute SSM Automation documents within your account.
ams_ssm_automation_role	
Used by (entity): **AWS Services**	
ams-opscenter-eventbridge-role	Assumed by Amazon EventBridge to create AWS System Manager OpsItems as a part of AMS-specific AWS Config Rules remediation workflow.
AMSOSConfigurationCustomerInstanceRole	This IAM role is applied to your Amazon EC2 instances when AMS OS-Configuration service discovers that the required IAM policies are missing. It allows your Amazon EC2 instances to interact with AWS Systems Manager, Amazon CloudWatch, and Amazon EventBridge services. It also has attached the AMS custom-managed policy to enable RDP access to your Windows instances.
mc-patch-glue-service-role	Assumed by AWS Glue ETL workflow to perform data transformation and prepare it for AMS Patch report generator.
Used by (entity): **AMS Service**	
ams-alarm-manager-AWSManagedServicesAlarmManagerDe-<8-digit hash>	Assumed by AMS alarm manager infrastructure within your AMS account to perform AWS Config Rules evaluation for a new AWS AppConfig deployment.
ams-alarm-manager-AWSManagedServicesAlarmManagerRe-<8-digit hash>	Assumed by AMS alarm manager remediation infrastructure within your AMS account to allow the creation or deletion of alarms for remediation.
ams-alarm-manager-AWSManagedServicesAlarmManagerSS-<8-digit hash>	Assumed by AWS Systems Manager to invoke the AMS alarm manager remediation service within your AMS account.
ams-alarm-manager-AWSManagedServicesAlarmManagerTr-<8-digit hash>	Assumed by AMS alarm manager infrastructure within your AWS account to conduct periodic AMS AWS Config Rules evaluation.
ams-alarm-manager-AWSManagedServicesAlarmManagerVa-<8-digit hash>	Assumed by AMS alarm manager infrastructure within your AMS account to ensure that the required alarms exists in the AWS account.
ams-backup-config-rule-st-amsBackupAlertConfigRule-<8-digit hash>	Assumed by AMS backup infrastructure in your AMS account to evaluate the AWS Config Rules 'amsBackupAlertEval' rule.

Role name	Description
ams-backup-config-rule-st-amsBackupPlanConfigRuleH-<8-digit hash>	Assumed by AMS backup infrastructure in your AMS account to execute the AWS Config Rules 'amsBackupPlanEval' rule.
ams-backup-iam-role	This role is used to run AWS Backup within your accounts.
ams-log-management-AWSManagedServicesCloudTrailLog-<8-digit hash>	Assumed by AWS CloudTrail to write logs into AMS-specific Amazon CloudWatch Logs groups.
ams-monitoring-AWSManagedServicesLogGroupLimitLamb-<8-digit hash>	Assumed by AMS Logging & Monitoring infrastructure in your AMS account to evaluate Amazon CloudWatch Logs groups limit and compare with the service quotas.
ams-monitoring-AWSManagedServicesRDSMonitoringRDSE-<8-digit hash>	Assumed by AMS Logging & Monitoring infrastructure in your AMS account to forward Amazon RDS events to Amazon CloudWatch Events.
ams-monitoring-AWSManagedServicesRedshiftMonitorin-<8-digit hash>	Assumed by AMS Logging & Monitoring infrastructure in your AMS account to forward Amazon Redshift events (CreateCluster and DeleteCuster) to Amazon CloudWatch Events.
ams-monitoring-infrastruc-AWSManagedServicesMonito-<8-digit hash>	Assumed by AMS Logging & Monitoring infrastructure in your AMS account to publish messages to Amazon Simple Notification Service to validate that the account is reporting all necessary data.
ams-opscenter-role	Assumed by AMS Notification Management system in your AMS account to manage AWS System Manager OpsItems related to alerts in your account.
ams-opsitem-autoexecution-role	Assumed by AMS Notification Management system to handle automated remediation using SSM documents for monitoring alerts related to resources in your account.
ams-patch-infrastructure-amspatchconfigruleroleC1-<8-digit hash>	Assumed by AWS Config to evaluate AMS patch resources and detect drift in its AWS CloudFormation stacks.
ams-patch-infrastructure-amspatchcwruleopsitemams-<8-digit hash>	Assumed by Amazon EventBridge to create AWS System Manager OpsItems for patching failures.
ams-patch-infrastructure-amspatchservicebusamspat-<8-digit hash>	Assumed by Amazon EventBridge to send an event to the AMS Patch orchestrator event bus for AWS Systems Manager Maintenance Windows state change notifications.
ams-patch-reporting-infra-amspatchreportingconfigr-<8-digit hash>	Assumed by AWS Config to evaluate AMS Patch reporting resources and detect drift in its AWS CloudFormation stacks.

Role name	Description
ams-resource-tagger-AWSManagedServicesResourceTagg-<8-digit hash>	Assumed by AMS Resource Tagger infrastructure within your AMS account to perform AWS Config Rules evaluation upon new AWS AppConfig deployment.
ams-resource-tagger-AWSManagedServicesResourceTagg-<8-digit hash>	Assumed by AMS Resource Tagger infrastructure within your AMS account to validate that required AWS tags exist for the managed resources.
ams-resource-tagger-AWSManagedServicesResourceTagg-<8-digit hash>	Assumed by AWS Systems Manager to invoke AMS Resource Tagger remediation workflow in your AMS account.
ams-resource-tagger-AWSManagedServicesResourceTagg-<8-digit hash>	Assumed by AMS Resource Tagger remediation infrastructure within your AMS account to create or delete AWS tags for the managed resources.
ams-resource-tagger-AWSManagedServicesResourceTagg-<8-digit hash>	Assumed by AMS Resource Tagger infrastructure within your AWS account to conduct periodic AMS Config Rule evaluation.
ams_os_configuration_event_rule_role-<AWS Region>	Assumed by Amazon EventBridge to forward events from your account to AMS OS-Configuration service EventBus in the correct Region.
mc-patch-reporting-service	Assumed by AMS patch data aggregator and report generator.

To learn more about AWS Cloud Development Kit (CDK) identifiers, including hashes, see UniqueIDs.

AMS Accelerate feature services assume the **ams-accelerate-admin** role for programmatic access to the account, but with a session policy scoped down for the respective feature service (for example, patch, backup, monitoring, and so forth).

AMS Accelerate follows industry best practices to meet and maintain compliance eligibility. AMS Accelerate access to your account is recorded in CloudTrail and also available for your review through change tracking. For information about queries that you can use to get this information, see Tracking changes in your AMS Accelerate accounts (p. 161).

Managing access using policies

Various AMS Accelerate support teams such as Operations Engineers, Cloud Architects, and Cloud Service Delivery Managers (CSDMs), sometimes require access to your accounts in order to respond to service requests and incidents. Their access is governed by an internal AMS access service that enforces controls, such as business justification, service requests, operations items, and support cases. The default access is read-only, and all access is tracked and recorded; see also Tracking changes in your AMS Accelerate accounts (p. 161).

Validation of IAM resources

The AMS Accelerate access system periodically assumes roles in your accounts (at least every 24 hours) and validates that all of our IAM resources are as expected.

In order to ensure uninterrupted access into your accounts, AMS Accelerate has a "canary" that monitors and alarms on the presence and status of the IAM roles, as well as their attached policies, mentioned above. Periodically, the canary assumes the **ams-accelerate-read-only** role and initiates CloudFormation and IAM API calls against your accounts. The canary evaluates the status of the AMS Accelerate access roles to make sure they are always unmodified and up-to-date. This activity creates CloudTrail logs in the account.

The AWS Security Token Service (AWS STS) session name of the canary is **AMS-Access-Roles-Auditor-{uuid4()}** as seen in CloudTrail and the following API calls occur:

- Cloud Formation API Calls: `describe_stacks()`
- IAM API Calls:
  - `get_role()`
  - `list_attached_role_policies()`
  - `list_role_policies()`
  - `get_policy()`
  - `get_policy_version()`
  - `get_role_policy()`

Security event logging and monitoring

Accounts enrolled in AMS Accelerate are configured with a baseline deployment of CloudWatch Events and Alarms that have been optimized to reduce noise and to identify indications of a true incident. AMS Accelerate also employs GuardDuty for account monitoring; see GuardDuty (p. 101) for more detail.

Compliance and conformance

AMS Accelerate maintains a library of AWS Config Rules and remediation actions to ensure that all your accounts comply with industry standards for security and operational integrity. AWS Config Rules test every configuration change among your all resources. If a change violates any rule conditions, AMS reports its findings, and allows you to remediate violations automatically or by request, according to the severity of the violation. AWS Config Rules facilitate compliance with standards set by: the Center for Internet Security (CIS), the National Institute of Standards and Technology (NIST) Cloud Security Framework (CSF), the Health Insurance Portability and Accountability Act (HIPAA), and the Payment Card Industry (PCI) Data Security Standard (DSS). The rules that AMS Accelerate deploys and manages have the `AMS` prefix in the rule name.

As an example, when an Amazon S3 bucket is created, AWS Config can evaluate the Amazon S3 bucket against a rule that requires Amazon S3 buckets to deny public read access. If the Amazon S3 bucket policy or bucket access control list (ACL), allows public read access, AWS Config flags both the bucket and the rule as noncompliant. These AWS Config Rules mark resources as either Compliant, Noncompliant, or Not Applicable, based on the result of their evaluation. For more information about AWS Config service, see the AWS Config Developer Guide.

You can use the AWS Config console, AWS CLI, or AWS Config API to view the rules deployed in your account and the compliance state of your rules and resources. For more information, see the AWS Config documentation: Viewing Configuration Compliance.

Config Reports

AMS generates a report that provides an in-depth look at resource and config rule compliance of AMS accounts. This report provides insights into:

- the top non-compliant resources in your environment, to discover potential threats and misconfigurations
- compliance of resources and config rules over time
- recommended config rule descriptions, severity of rules, and remediation steps to fix non-compliant resources

See the AWS Config Reporting section for more details.

Resource Exception in AWS Config Rules

The Resource Exception feature in AWS Config Rules allows customers to eliminate a specific non-compliant resource for a specific Config Rule in the report. Please note that the exempted resources will still show up as non-compliant in your AWS Config Service console.

You can create a Service request against your account with following inputs:

```
[
    {
        "resource_name": "resource_name_1",
        "config_rule_name": "config_rule_name_1",
        "business_justification": "REASON_TO_EXEMPT_RESOURCE",
        "resource_type": "resource_type"
    },
    {
        "resource_name": "resource_name_2",
        "config_rule_name": "config_rule_name_2",
        "business_justification": "REASON_TO_EXEMPT_RESOURCE",
        "resource_type": "resource_type"
    }
]
```

Remediation using AWS Config Rules

AMS Accelerate has a library of AWS Systems Manager Automation documents and runbooks to assist in remediating noncompliant resources. AMS has defined remediation response plan to be implemented for each rule when a resource goes into noncompliance. Response plans are described below:

- **Automatic Remediation**

  When a resource goes into the noncompliant state, AMS automatically remediates the rules using automated SSM documents. Noncompliance of these rules may strongly impact the security and availability of your accounts. When the resource goes into noncompliance, you are notified with an incident report. When the resource is returned to the compliant state, you receive an update to the incident report. If the SSM document fails, then you receive an update on the incident, stating that automated remediation has failed, and an AMS engineer will be investigating the issue.
- **Automatic Incident**

  AMS automatically creates an incident report to notify you that a resource has gone into a noncompliant state and asks which actions you would like to be performed. You have the following options when responding to the incident:

  - Request that AMS remediate the noncompliant resources listed in the incident. Then, we attempt to remediate the noncompliant resource, and notify you once the underlying incident has been resolved.
  - You can resolve the noncompliant item manually in the console or through your automated deployment system (for example, CI/CD Pipeline template updates); then, you can resolve the incident. The noncompliant resource is re-evaluated as per the rule’s schedule and, if the resource is evaluated as noncompliant, a new incident report is created.

  - You can choose to not resolve the noncompliant resource and simply resolve the incident. If you update the configuration of the resource later, AWS Config will trigger a re-evaluation and you will again be alerted to evaluate the noncompliance of that resource.
- **Config Report Only**

  If your resource goes into a noncompliant state, AMS does not automatically remediate or notify you. You can review the compliance states of these rules using the AWS Config console, AWS CLI, or AWS Config API. Additionally, your CSDM will share a report of all rules in your environment upon request. This report highlights the compliant and noncompliant resources in your account.

  For more details, see the following table, AMS AWS Config Rules Remediation.

  > **Note**
  > You cannot modify the remediation category for the config rule at this time.

AMS Accelerate AWS Config Rules Inventory

Rule Name	Identifier	Trigger Type	Compliance Framework
Remediation Category: Automatic Remediation			
ams-nist-cis-guardduty-enabled-centralized	GUARDDUTY_ENABLED_CENTRALIZED	Periodic	CIS, NIST, HIPAA, PCI
ams-nist-cis-vpc-flow-logs-enabled	VPC_FLOW_LOGS_ENABLED	Periodic	CIS, NIST, HIPAA, PCI
Remediation Category: Auto Incident			
ams-nist-cis-vpc-default-security-group-closed	VPC_DEFAULT_SECURITY_GROUP_CLOSED	Config Changes	CIS, NIST, HIPAA, PCI
ams-nist-cis-iam-password-policy	IAM_PASSWORD_POLICY	Periodic	NIST, HIPAA, PCI
ams-nist-cis-iam-root-access-key-check	IAM_ROOT_ACCESS_KEY_CHECK	Periodic	CIS, NIST, HIPAA, PCI
ams-nist-cis-iam-user-mfa-enabled	IAM_USER_MFA_ENABLED	Periodic	CIS, NIST, HIPAA, PCI
ams-nist-cis-restricted-ssh	INCOMING_SSH_DISABLED	Config Changes	CIS, NIST, HIPAA, PCI

Rule Name	Identifier	Trigger Type	Compliance Framework
ams-nist-cis-restricted-common-ports	RESTRICTED_INCOMING_TRAFFIC	Config Changes	CIS, NIST, HIPAA, PCI
ams-nist-cis-s3-account-level-public-access-blocks	S3_ACCOUNT_LEVEL_PUBLIC_ACCESS_BLOCKS	Config Changes	CIS, NIST, HIPAA, PCI
ams-nist-cis-s3-bucket-public-read-prohibited	S3_BUCKET_PUBLIC_READ_PROHIBITED	Config Changes	CIS, NIST, HIPAA, PCI
ams-nist-cis-s3-bucket-public-write-prohibited	S3_BUCKET_PUBLIC_WRITE_PROHIBITED	Config Changes	CIS, NIST, HIPAA, PCI
ams-nist-cis-s3-bucket-server-side-encryption-enabled	S3_BUCKET_SERVER_SIDE_ENCRYPTION_ENABLED	Config Changes	CIS, NIST, HIPAA, PCI
ams-nist-cis-securityhub-enabled	SECURITYHUB_ENABLED	Periodic	CIS, NIST, HIPAA
Remediation Category: Config Report Only			
ams-nist-cis-ec2-instance-managed-by-systems-manager	EC2_INSTANCE_MANAGED_BY_SSM	Config Changes	CIS, NIST, HIPAA, PCI
ams-nist-cis-cloudtrail-enabled	CLOUD_TRAIL_ENABLED	Periodic	CIS, NIST, HIPAA, PCI
ams-nist-cis-access-keys-rotated	ACCESS_KEYS_ROTATED	Periodic	CIS, NIST, HIPAA, PCI
ams-nist-cis-acm-certificate-expiration-check	ACM_CERTIFICATE_EXPIRATION_CHECK	Config Changes	CIS, NIST, PCI

Rule Name	Identifier	Trigger Type	Compliance Framework
ams-nist-cis-alb-http-to-https-redirection-check	ALB_HTTP_TO_HTTPS_REDIRECTION_CHECK	Periodic	CIS, NIST, HIPAA, PCI
ams-nist-cis-api-gw-cache-enabled-and-encrypted	API_GW_CACHE_ENABLED_AND_ENCRYPTED	Config Changes	CIS, NIST, HIPAA, PCI
ams-nist-cis-api-gw-execution-logging-enabled	API_GW_EXECUTION_LOGGING_ENABLED	Config Changes	CIS, NIST, HIPAA, PCI
ams-nist-autoscaling-group-elb-healthcheck-required	AUTOSCALING_GROUP_ELB_HEALTHCHECK_REQUIRED	Config Changes	NIST, HIPAA, PCI
ams-nist-cis-cloud-trail-cloud-watch-logs-enabled	CLOUD_TRAIL_CLOUD_WATCH_LOGS_ENABLED	Periodic	CIS, NIST, HIPAA, PCI
ams-nist-cis-cloud-trail-encryption-enabled	CLOUD_TRAIL_ENCRYPTION_ENABLED	Periodic	CIS, NIST, HIPAA, PCI
ams-nist-cis-cloud-trail-log-file-validation-enabled	CLOUD_TRAIL_LOG_FILE_VALIDATION_ENABLED	Periodic	CIS, NIST, HIPAA, PCI
ams-nist-cis-cloudtrail-s3-dataevents-enabled	CLOUDTRAIL_S3_DATAEVENTS_ENABLED	Periodic	CIS, NIST, HIPAA, PCI
ams-nist-cis-cloudwatch-alarm-action-check	CLOUDWATCH_ALARM_ACTION_CHECK	Config Changes	CIS, HIPAA

Rule Name	Identifier	Trigger Type	Compliance Framework
ams-nist-cis-cloudwatch-log-group-encrypted	CLOUDWATCH_LOG_GROUP_ENCRYPTED	Periodic	CIS, HIPAA, PCI
ams-nist-cis-codebuild-project-envvar-awscred-check	CODEBUILD_PROJECT_ENVVAR_AWSCRED_CHECK	Config Changes	CIS, NIST, HIPAA, PCI
ams-nist-cis-codebuild-project-source-repo-url-check	CODEBUILD_PROJECT_SOURCE_REPO_URL_CHECK	Config Changes	CIS, NIST, HIPAA, PCI
ams-nist-cis-db-instance-backup-enabled	DB_INSTANCE_BACKUP_ENABLED	Config Changes	CIS, NIST, HIPAA, PCI
ams-nist-cis-dms-replication-not-public	DMS_REPLICATION_NOT_PUBLIC	Periodic	CIS, NIST, HIPAA, PCI
ams-nist-dynamodb-autoscaling-enabled	DYNAMODB_AUTOSCALING_ENABLED	Periodic	NIST, HIPAA
ams-nist-cis-dynamodb-pitr-enabled	DYNAMODB_PITR_ENABLED	Periodic	CIS, NIST, HIPAA
ams-nist-dynamodb-throughput-limit-check	DYNAMODB_THROUGHPUT_LIMIT_CHECK	Periodic	HIPAA
ams-nist-ebs-optimized-instance	EBS_OPTIMIZED_INSTANCE	Config Changes	HIPAA
ams-nist-cis-ebs-snapshot-public-restorable-check	EBS_SNAPSHOT_PUBLIC_RESTORABLE_CHECK	Periodic	CIS, NIST, HIPAA, PCI

Rule Name	Identifier	Trigger Type	Compliance Framework
ams-nist-ec2-instance-detailed-monitoring-enabled	EC2_INSTANCE_DETAILED_MONITORING_ENABLED	Config Changes	CIS, NIST, HIPAA, PCI
ams-nist-cis-ec2-instance-no-public-ip	EC2_INSTANCE_NO_PUBLIC_IP	Config Changes	CIS, NIST, HIPAA, PCI
ams-nist-cis-ec2-managedinstance-association-compliance-status-check	EC2_MANAGEDINSTANCE_ASSOCIATION_COMPLIANCE_STATUS_CHECK	Config Changes	CIS, NIST, HIPAA, PCI
ams-nist-cis-ec2-managedinstance-patch-compliance-status-check	EC2_MANAGEDINSTANCE_PATCH_COMPLIANCE_STATUS_CHECK	Config Changes	CIS, NIST, HIPAA, PCI
ams-nist-cis-ec2-security-group-attached-to-eni	EC2_SECURITY_GROUP_ATTACHED_TO_ENI	Config Changes	CIS, NIST, HIPAA, PCI
ams-nist-cis-ec2-stopped-instance	EC2_STOPPED_INSTANCE	Periodic	CIS, NIST
ams-nist-cis-ec2-volume-inuse-check	EC2_VOLUME_INUSE_CHECK	Config Changes	CIS, NIST
ams-nist-cis-efs-encrypted-check	EFS_ENCRYPTED_CHECK	Periodic	CIS, NIST, HIPAA, PCI
ams-nist-cis-eip-attached	EIP_ATTACHED	Config Changes	CIS, NIST, HIPAA, PCI

Rule Name	Identifier	Trigger Type	Compliance Framework
ams-nist-cis-elasticache-redis-cluster-automatic-backup-check	ELASTICACHE_REDIS_CLUSTER_AUTOMATIC_BACKUP_CHECK	Periodic	CIS, NIST, HIPAA
ams-nist-cis-elasticsearch-encrypted-at-rest	ELASTICSEARCH_ENCRYPTED_AT_REST	Periodic	CIS, NIST, HIPAA, PCI
ams-nist-cis-elasticsearch-in-vpc-only	ELASTICSEARCH_IN_VPC_ONLY	Periodic	CIS, NIST, HIPAA, PCI
ams-nist-cis-elb-acm-certificate-required	ELB_ACM_CERTIFICATE_REQUIRED	Config Changes	CIS, NIST, HIPAA, PCI
ams-nist-elb-deletion-protection-enabled	ELB_DELETION_PROTECTION_ENABLED	Config Changes	CIS, NIST, HIPAA, PCI
ams-nist-cis-elb-logging-enabled	ELB_LOGGING_ENABLED	Config Changes	CIS, NIST, HIPAA, PCI
ams-nist-cis-emr-kerberos-enabled	EMR_KERBEROS_ENABLED	Periodic	CIS, NIST, HIPAA, PCI
ams-nist-cis-emr-master-no-public-ip	EMR_MASTER_NO_PUBLIC_IP	Periodic	CIS, NIST, HIPAA, PCI
ams-nist-cis-encrypted-volumes	ENCRYPTED_VOLUMES	Config Changes	CIS, NIST, HIPAA, PCI
ams-nist-cis-guardduty-non-archived-findings	GUARDDUTY_NON_ARCHIVED_FINDINGS	Periodic	CIS, NIST, HIPAA, PCI
ams-nist-iam-group-has-users-check	IAM_GROUP_HAS_USERS_CHECK	Config Changes	NIST, HIPAA, PCI

Rule Name	Identifier	Trigger Type	Compliance Framework
ams-nist-cis-iam-policy-no-statements-with-admin-access	IAM_POLICY_NO_STATEMENTS_WITH_ADMIN_ACCESS	Config Changes	CIS, NIST, HIPAA, PCI
ams-nist-cis-iam-user-group-membership-check	IAM_USER_GROUP_MEMBERSHIP_CHECK	Config Changes	CIS, NIST, HIPAA, PCI
ams-nist-cis-iam-user-no-policies-check	IAM_USER_NO_POLICIES_CHECK	Config Changes	CIS, NIST, HIPAA, PCI
ams-nist-cis-iam-user-unused-credentials-check	IAM_USER_UNUSED_CREDENTIALS_CHECK	Periodic	CIS, NIST, HIPAA, PCI
ams-nist-cis-ec2-instances-in-vpc	INSTANCES_IN_VPC	Config Changes	CIS, NIST, HIPAA, PCI
ams-nist-cis-internet-gateway-authorized-vpc-only	INTERNET_GATEWAY_AUTHORIZED_VPC_ONLY	Periodic	CIS
ams-nist-cis-kms-cmk-not-scheduled-for-deletion	KMS_CMK_NOT_SCHEDULED_FOR_DELETION	Periodic	CIS, NIST, PCI
ams-nist-lambda-concurrency-check	LAMBDA_CONCURRENCY_CHECK	Config Changes	HIPAA
ams-nist-lambda-dlq-check	LAMBDA_DLQ_CHECK	Config Changes	HIPAA

Rule Name	Identifier	Trigger Type	Compliance Framework
ams-nist-cis-lambda-function-public-access-prohibited	LAMBDA_FUNCTION_PUBLIC_ACCESS_PROHIBITED	Config Changes	CIS, NIST, HIPAA, PCI
ams-nist-cis-lambda-inside-vpc	LAMBDA_INSIDE_VPC	Config Changes	CIS, NIST, HIPAA, PCI
ams-nist-cis-mfa-enabled-for-iam-console-access	MFA_ENABLED_FOR_IAM_CONSOLE_ACCESS	Periodic	CIS, NIST, HIPAA, PCI
ams-nist-cis-multi-region-cloudtrail-enabled	MULTI_REGION_CLOUD_TRAIL_ENABLED	Periodic	CIS, NIST, HIPAA, PCI
ams-nist-rds-enhanced-monitoring-enabled	RDS_ENHANCED_MONITORING_ENABLED	Config Changes	NIST, HIPAA
ams-nist-cis-rds-instance-public-access-check	RDS_INSTANCE_PUBLIC_ACCESS_CHECK	Config Changes	CIS, NIST, HIPAA, PCI
ams-nist-rds-multi-az-support	RDS_MULTI_AZ_SUPPORT	Config Changes	NIST, HIPAA
ams-nist-cis-rds-snapshots-public-prohibited	RDS_SNAPSHOTS_PUBLIC_PROHIBITED	Config Changes	CIS, NIST, HIPAA, PCI
ams-nist-cis-rds-storage-encrypted	RDS_STORAGE_ENCRYPTED	Config Changes	CIS, NIST, HIPAA, PCI
ams-nist-cis-redshift-cluster-configuration-check	REDSHIFT_CLUSTER_CONFIGURATION_CHECK	Config Changes	CIS, NIST, HIPAA, PCI

Rule Name	Identifier	Trigger Type	Compliance Framework
ams-nist-cis-redshift-cluster-public-access-check	REDSHIFT_CLUSTER_PUBLIC_ACCESS_CHECK	Config Changes	CIS, NIST, HIPAA, PCI
ams-nist-cis-redshift-require-tls-ssl	REDSHIFT_REQUIRE_TLS_SSL	Periodic	CIS, NIST, HIPAA, PCI
ams-nist-cis-root-account-hardware-mfa-enabled	ROOT_ACCOUNT_HARDWARE_MFA_ENABLED	Periodic	CIS, NIST, HIPAA, PCI
ams-nist-cis-root-account-mfa-enabled	ROOT_ACCOUNT_MFA_ENABLED	Periodic	CIS, NIST, HIPAA, PCI
ams-nist-cis-s3-bucket-default-lock-enabled	S3_BUCKET_DEFAULT_LOCK_ENABLED	Config Changes	CIS, NIST
ams-nist-cis-s3-bucket-logging-enabled	S3_BUCKET_LOGGING_ENABLED	Config Changes	CIS, NIST, HIPAA, PCI
ams-nist-cis-s3-bucket-replication-enabled	S3_BUCKET_REPLICATION_ENABLED	Config Changes	CIS, NIST, HIPAA, PCI
ams-nist-cis-s3-bucket-ssl-requests-only	S3_BUCKET_SSL_REQUESTS_ONLY	Config Changes	CIS, NIST, HIPAA, PCI
ams-nist-cis-s3-bucket-versioning-enabled	S3_BUCKET_VERSIONING_ENABLED	Periodic	CIS, NIST, HIPAA, PCI
ams-nist-cis-sagemaker-endpoint-configuration-kms-key-configured	SAGEMAKER_ENDPOINT_CONFIGURATION_KMS_KEY_CONFIGURED	Periodic	CIS, NIST, HIPAA, PCI

Rule Name	Identifier	Trigger Type	Compliance Framework
ams-nist-cis-sagemaker-notebook-instance-kms-key-configured	SAGEMAKER_NOTEBOOK_INSTANCE_KMS_KEY_CONFIGURED	Periodic	CIS, NIST, HIPAA, PCI
ams-nist-cis-sagemaker-notebook-no-direct-internet-access	SAGEMAKER_NOTEBOOK_NO_DIRECT_INTERNET_ACCESS	Periodic	CIS, NIST, HIPAA, PCI
ams-nist-cis-secretsmanager-rotation-enabled-check	SECRETSMANAGER_ROTATION_ENABLED_CHECK	Config Changes	CIS, NIST, HIPAA
ams-nist-cis-secretsmanager-scheduled-rotation-success-check	SECRETSMANAGER_SCHEDULED_ROTATION_SUCCESS_CHECK	Config Changes	CIS, NIST, HIPAA
ams-nist-cis-sns-encrypted-kms	SNS_ENCRYPTED_KMS	Config Changes	CIS, NIST, HIPAA, PCI
ams-nist-cis-vpc-sg-open-only-to-authorized-ports	VPC_SG_OPEN_ONLY_TO_AUTHORIZED_PORTS	Config Changes	CIS, NIST, HIPAA, PCI
ams-nist-vpc-vpn-2-tunnels-up	VPC_VPN_2_TUNNELS_UP	Config Changes	NIST, HIPAA
ams-cis-ec2-ebs-encryption-by-default	EC2_EBS_ENCRYPTION_BY_DEFAULT	Periodic	CIS, NIST, HIPAA, PCI
ams-cis-rds-snapshot-encrypted	RDS_SNAPSHOT_ENCRYPTED	Config Changes	CIS, NIST, HIPAA, PCI

Rule Name	Identifier	Trigger Type	Compliance Framework
ams-cis-redshift-cluster-maintenancesettings-check	REDSHIFT_CLUSTER_MAINTENANCESETTINGS_CHECK	Config Changes	CIS, NIST, HIPAA, PCI

Incident response

Upon receiving an alert, the AMS team uses automated and manual remediations to bring the resources back to a healthy state. If remediation fails, AMS starts the incident management process to collaborate with your team. You can change the baselines by updating the default configuration in a configuration file.

Incident response and onboarding in AMS Accelerate

During onboarding, AMS Accelerate suppresses automatic incident creation for your existing noncompliant resources; instead, your Cloud Service Deliver Manager (CSDM) provides you with a report that contains all the noncompliance rules and resources for your review. After you have identified the rules that you want AMS to remediate, create a service request in the AWS Support Center console indicating those rule and resources. The following Service Request template is an example of a customer request to AMS to manually remediate noncompliant resources. If AMS has additional questions, we work with you in the Service Request to gather the information required.

```
Hello,
Please remediate the following resources for the Config Rule “ENCRYPTED_VOLUMES”.
Resource List:
    "Vol-12345678"
    "Vol-87654312"
Thank you
```

After the onboarding process is completed, AMS Accelerate automatically creates an incident for each noncompliant resource for the rules marked as Automatic Incident below.

Resilience

The AWS global infrastructure is built around AWS Regions and Availability Zones. AWS Regions provide multiple physically separated and isolated Availability Zones, which are connected with low-latency, high-throughput, and highly redundant networking. With Availability Zones, you can design and operate applications and databases that automatically fail over between zones without interruption. Availability Zones are more highly available, fault tolerant, and scalable than traditional single or multiple data center infrastructures.

For more information about AWS Regions and Availability Zones, see AWS global infrastructure.

For information about AMS Accelerate continuity management, see Backup management in AMS Accelerate (p. 143).

Security best practices

AMS Accelerate uses conformance packs, which provide a general-purpose compliance framework designed to enable you to create security, operational, or cost-optimization governance checks using managed or custom AWS Config Rules and AWS Config remediation actions. For information on how to best configure these conformance packs, please refer to AWS Config's Operational Best Practices for NIST CSF and Operational Best Practices for CIS Top 20.

Monitoring and event management in AMS Accelerate

Topics

The AMS Accelerate monitoring system monitors your AWS resources for failures, performance degradation, and security issues.

As a managed account, AMS Accelerate configures and deploys alarms for applicable AWS resources, monitors these resources, and performs remediation when needed.

The AMS Accelerate monitoring system relies on internal tools, such as Resource Tagger and Alarm Manager, and leverages native AWS services, such as AWS AppConfig, Amazon CloudWatch (CloudWatch), Amazon EventBridge(formerly known as CloudWatch), Amazon GuardDuty, Amazon Macie, and AWS Health.

What is monitoring?

AMS Accelerate monitoring provides these benefits:

- A default configuration that creates, manages, and deploys policies across your managed account for all or supported AWS resources that you select.
- A monitoring baseline so that you have a default level of protection, even if you don't configure any other monitoring for your managed accounts. For more information, see Alerts from baseline monitoring in AMS (p. 122).
- The ability to customize the baseline resource alarms to meet your requirements.
- Automatic remediation of alerts by AMS Operations, when possible, to prevent or reduce the impact to your applications. For example, if you are using a standalone Amazon EC2 instance and it fails the system health check, AMS attempts to recover the instance by stopping and restarting it. For more information, see AMS automatic remediation of alerts (p. 140).
- Visibility into active, and previously resolved, alerts using OpsCenter. For example, if you have an unexpected high CPU utilization on an Amazon EC2 instance, you can request access to the AWS Systems Manager console (which includes access to the OpsCenter console) and view the OpsItem directly in the OpsCenter console.
- Investigating alerts to determine the appropriate actions. For more information, see Incident management (p. 52).
- Alerts generated based on the configuration in your account and supported AWS services. The monitoring configuration of an account refers to all the resource parameters in the account that create an alert. The monitoring configuration of an account includes CloudWatch Alarm definitions, and

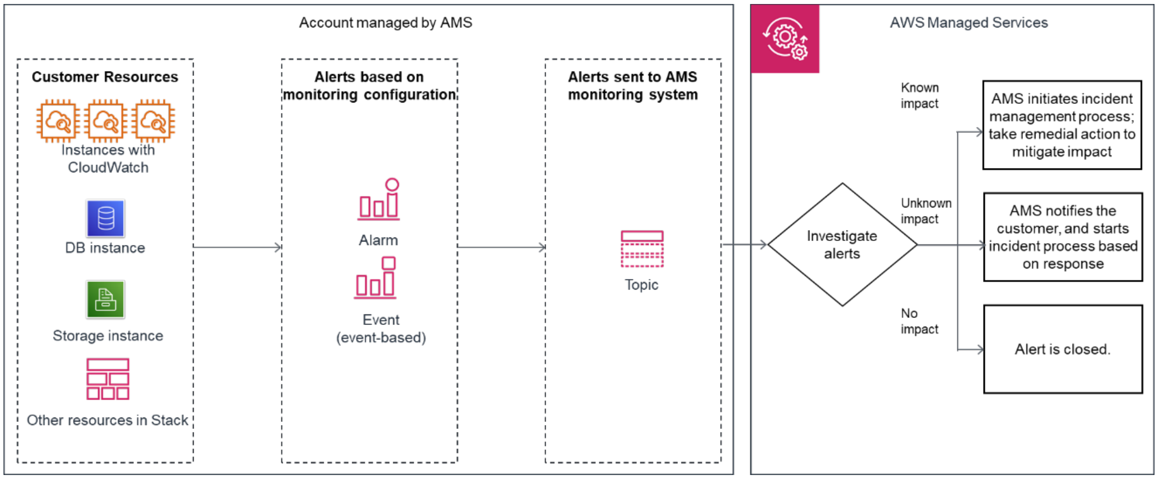
EventBridge (formerly known as CloudWatch Events) that generate the alert (alarm or event). For more information about the resource parameters, see Alerts from baseline monitoring in AMS (p. 122).

- Notification of imminent, on-going, receding, or potential failures; performance degradation; or security issues generated by the baseline monitoring configured in an account (known as an alert). Examples of alerts include a CloudWatch Alarm, an Event, or a Finding from an AWS service, such as GuardDuty or AWS Health.

How monitoring works

See the following graphics on monitoring architecture in AMS.

The following diagram depicts the **AMS Accelerate** monitoring architecture.


Account managed by AMS
Customer Resources
Instances with CloudWatch
DB instance
Storage instance
Other resources in Stack
Alerts based on monitoring configuration
Alarm
Event (event-based)
Alerts sent to AMS monitoring system
Topic
AWS Managed Services
Investigate alerts
Known impact
AMS initiates incident management process; take remedial action to mitigate impact
Unknown impact
AMS notifies the customer, and starts incident process based on response
No impact
Alert is closed.

After your resources are tagged based on the policy defined using Resource tagger, and alarm definitions are deployed, the following diagram depicts the AMS monitoring architecture.

- Generation: At the time of account onboarding, AMS configures baseline monitoring (a combination of CloudWatch (CW) alarms, and CW event rules) for all your resources created in a managed account. The baseline monitoring configuration generates an alert when a CW alarm is triggered or a CW event is generated.
- Aggregation: All alerts generated by your resources are sent to the AMS monitoring system by directing them to an SNS topic in the account.
- Processing: AMS analyzes the alerts and processes them based on their potential for impact. Alerts are processed as described next.
  - Alerts with known customer impact: These lead to the creation of a new incident report and AMS follows the incident management process.

    Example alert: An Amazon EC2 instance fails a system health check, AMS attempts to recover the instance by stopping and restarting it.
  - Alerts with uncertain customer impact: For these types of alerts, AMS sends a service notification that posts to your **Service Requests** page, asking you to verify the impact before we classify the alert as an incident.

    For example: An alert for >85% CPU utilization for more than 10 minutes on an Amazon EC2 instance can't be immediately categorized as an incident since this behavior may be expected based on usage. For such alerts, AMS sends an alert notification with the details and checks if the alert needs mitigating action. Alert notifications are discussed in detail in this section. We offer options for mitigating actions in the notification, and your reply that confirms that the alert is an incident triggers the creation of a new incident report and the AMS incident management process. Any service notification that receives a response of "no customer impact," or no response at all for three days, is marked as resolved and the corresponding alert is marked as resolved.

- Alerts with no customer impact: If, after evaluation, AMS determines that the alert doesn't have customer impact, the alert is closed.

  For example, AWS Health notifies of an EC2 instance requiring replacement but that instance has since been terminated.

Alert notification

As a part of the alert processing, based on the impact analysis, AMS creates an incident and initiates the incident management process for remediation, when impact can be determined. If impact can't be determined, AMS sends an alert notification to the email address associated with your account by way of a service notification; see the diagram on AMS monitoring architecture for alert handling process in How monitoring works (p. 121).

Tag-based alert notification

We recommend tag-based alert notifications because notifications sent to a single email address can cause confusion when multiple teams use the same account. You can use tags to get alert notifications for different resources sent to different email addresses. For resources with alerts that need to be sent to a specific email address, tag that resource with the key = OwnerTeamEmail, value = EMAIL_ADDRESS (use a group email; do not put personal information in tags). You can also use a custom tag key, but you must provide the custom tag key name to your CSDM with your explicit consent to use it in an email in order to activate automated notification for the tag-based communication. We recommend using the same tagging strategy for contact tags across all your instances and resources.

Note
The tag key value **OwnerTeamEmail** does not have to be in camel case. However, tags are case sensitive and it's best to use the recommended format. The email address must be specified in full, with the "at sign" (@) to separate the local part from the domain. Examples of invalid email addresses: Team.AppATabc.xyz or john.doe. For general guidance on your tagging strategy, see Tagging AWS resources. Do not add personally identifiable information (PII) in your tags, use distribution lists or aliases wherever possible.

Alerts from baseline monitoring in AMS

This section describes AMS Accelerate monitoring defaults; for more information, see Monitoring and event management in AMS Accelerate (p. 120).

The following table shows what is monitored and the default alerting thresholds. You can change the alerting thresholds with a custom configuration document, or submit a service request. For instructions on changing your custom alarm configuration, see Changing the configuration (p. 137). To be notified directly when alarms cross their threshold, in addition to AMS's standard alerting process, follow these instructions about how to overwrite alarm configurations, Tag-based Alarm Manager (p. 128).

Amazon CloudWatch provides extended retention of metrics. For more information, see CloudWatch Limits.

Note
AMS Accelerate calibrates its baseline monitoring on a periodic basis. New accounts are always onboarded with the latest baseline monitoring and the table describes the baseline monitoring for an account that is newly onboarded. AMS Accelerate updates the baseline monitoring in existing accounts on a periodic basis and you may experience a delay before the updates are in place.

Alerts from baseline monitoring

Resource	Alert name and trigger condition	Notes
For starred (*) alerts, AMS proactively assesses impact and remediates when possible; if remediation is not possible, AMS creates an incident. Where automation fails to remediate the issue, AMS informs you of the incident case and an AMS engineer is engaged. In addition, these alerts can be sent directly to your email (if you have opted in to the Direct-Customer-Alerts SNS topic).		
ALB instance	HTTPCode_Target_5XX_Count sum > 0% for 1 min, 5 consecutive times.	CloudWatch alarm on excess number of HTTP 5XX response codes generated by the targets.
	RejectedConnectionCount sum > 0% for 1 min, 5 consecutive times.	CloudWatch alarm if the number of connections that were rejected because the load balancer reached its maximum
ALB target	TargetConnectionErrorCount sum > 0% for 1 min, 5 consecutive times.	CloudWatch alarm if number of connections were unsuccessfully established between the load balancer and the registered instances.
Aurora	Average CPU utilization > 90% for 20 mins, 5 consecutive times.	CloudWatch Alarm.
Site-to-Site VPN	VPNTunnelDown TunnelState <= 0 for 1 min, 20 consecutive times.	TunnelState is 0 when both tunnels are down, .5 when one tunnel is up, and 1.0 when both tunnels are up.
EC2 instance - all OSs	CPUUtilization* > 95% for 5 mins, 6 consecutive times.	CloudWatch alarm. High CPU utilization is an indicator of a change in application state such as deadlocks, infinite loops, malicious attacks, and other anomalies. This is a Direct-Customer-Alerts alarm.
	StatusCheckFailed > 0% for 5 minute , 3 consecutive times.	
EC2 instance - Linux	Minimum mem_used_percent >= 95% for 5 minutes, 6 consecutive times.	
	Average swap_used_percent >= 95% for 5 minutes, 6 consecutive times.	
	Maximum disk_used_percent >= 95% for 5 minutes, 6 consecutive times.	
EC2 instance - Windows	Minimum Memory % Committed Bytes in Use >= 95% for 5 minutes, 6 consecutive times.	
	Maximum LogicalDisk % Free Space <= 5% for 5 minutes, 6 consecutive times.	
OpenSearch cluster	ClusterStatus	CloudWatch alarm. The KMS encryption key that is used to encrypt

Resource	Alert name and trigger condition	Notes
	red maximum is >= 1 for 1 minute, 1 consecutive time.	data at rest in your domain is disabled. Re-enable it to restore normal operations. To learn more, see Red Cluster Status.
OpenSearch domain	KMSKeyError >= 1 for 1 minute, 1 consecutive time.	CloudWatch alarm. At least one primary shard and its replicas are not allocated to a node. To learn more, see Encryption of Data at Rest for Amazon OpenSearch Service.
	KMSKeyInaccessible >= 1 for 1 minute, 1 consecutive time.	
	ClusterStatus yellow maximum is >= 1 for 1 minute, 1 consecutive time.	At least one replica shard is not allocated to a node. To learn more, see Yellow Cluster Status.
	FreeStorageSpace minimum is <= 20480 for 1 minute, 1 consecutive time.	A node in your cluster is down to 20 GiB of free storage space. To learn more, see Lack of Available Storage Space.
	ClusterIndexWritesBlocked >= 1 for 5 minutes, 1 consecutive time.	The cluster is blocking write requests. To learn more, see ClusterBlockException.
	Nodes minimum < x for 1 day, 1 consecutive time.	x is the number of nodes in your cluster. This alarm indicates that at least one node in your cluster has been unreachable for one day. To learn more, see Failed Cluster Nodes.
	CPUUtilization average >= 80% for 15 minutes, 3 consecutive times.	100% CPU utilization isn't uncommon, but sustained high averages are problematic. Consider right-sizing an existing instance types or adding instances.
	JVMMemoryPressure maximum >= 80% for 5 minutes, 3 consecutive times.	The cluster could encounter out of memory errors if usage increases. Consider scaling vertically. Amazon ES uses half of an instance's RAM for the Java heap, up to a heap size of 32 GiB. You can scale instances vertically up to 64 GiB of RAM, at which point you can scale horizontally by adding instances.
	MasterCPUUtilization average >= 50% for 15 minutes, 3 consecutive times.	Consider using larger instance types for your dedicated master nodes. Because of their role in cluster stability and blue/green deployments, dedicated master nodes should have lower average CPU usage than data nodes.
	MasterJVMMemoryPressure maximum >= 80% for 15 minutes, 1 consecutive time.	

Resource	Alert name and trigger condition	Notes
OpenSearch instance	AutomatedSnapshotFailure maximum is >= 1 for 1 minute, 1 consecutive time.	CloudWatch alarm. An automated snapshot failed. This failure is often the result of a red cluster health status. To learn more, see Red Cluster Status.
ELB instance	SpilloverCountBackendConnectionErrors > 1 for 1 minute , 15 consecutive times.	CloudWatch alarm if an excess number of requests that were rejected because the surge queue is full.
	SurgeQueueLength > 100 for 1 minute, 15 consecutive times.	CloudWatch alarm if an excess number of requests are pending routing.
GuardDuty Service	Not applicable; all findings (threat purposes) are monitored. Each finding corresponds to an alert. Changes in the GuardDuty findings. These changes include newly generated findings or subsequent occurrences of existing findings.	List of supported GuardDuty finding types are on GuardDuty Active Finding Types.
Health	AWS Personal Health Dashboard	Notifications sent when there are changes in the status of AWS Personal Health Dashboard (AWS Health) events.Service event. Example: Scheduled EC2 instance store retirement.
Macie	Newly generated alerts and updates to existing alerts. Macie finds any changes in the findings. These changes include newly generated findings or subsequent occurrences of existing findings.	Amazon Macie alert. For a list of supported Amazon Macie alert types, see Analyzing Amazon Macie findings. Note that Macie is not enabled for all accounts.
NATGateways	PacketsDropCount : Alarm if packetsdropcount is > 0 over 15 minutes period	A value greater than zero may indicate an ongoing transient issue with the NAT gateway.
	ErrorPortAllocation : Alarm if NAT Gateways could not allocate port for over 15 minutes evaluation period	The number of times the NAT gateway could not allocate a source port. A value greater than Zero indicates that too many concurrent connecations are open..
RDS	Average CPU utilization > 75% for 15 mins, 2 consecutive times.	CloudWatch alarms.
	Sum of DiskQueueDepth > 75% for 1 mins, 2 consecutive times.	

Resource	Alert name and trigger condition	Notes
	Average FreeStorageSpace < 1,073,741,824 bytes for 5 mins, 2 consecutive times.	
	Average ReadLatency >= 1.001 seconds for 5 mins, 2 consecutive times.	
	Average WriteLatency >= 1.005 seconds for 5 mins, 2 consecutive times.	
	Low Storage alert Triggers when the allocated storage for the DB instance has been exhausted.	RDS-EVENT-0007, see details at Using Amazon RDS event notification.
	DB instance fail The DB instance has failed due to an incompatible configuration or an underlying storage issue. Begin a point-in-time-restore for the DB instance.	RDS-EVENT-0031, see details at Amazon RDS Event Categories and Event Messages.
	RDS -0034 failover not attempted. RDS is not attempting a requested failover because a failover recently occurred on the DB instance.	RDS-EVENT-0034, see details at Amazon RDS Event Categories and Event Messages.
	RDS - 0035 DB instance invalid parameters For example, MySQL could not start because a memory-related parameter is set too high for this instance class, so your action would be to modify the memory parameter and reboot the DB instance.	RDS-EVENT-0035, see details at Amazon RDS Event Categories and Event Messages.
	Invalid subnet IDs DB instance The DB instance is in an incompatible network. Some of the specified subnet IDs are invalid or do not exist.	Service event. RDS-EVENT-0036, see details at Amazon RDS Event Categories and Event Messages.
	RDS-0045 DB instance read replica error An error has occurred in the read replication process. For more information, see the event message. For information on troubleshooting Read Replica errors, see Troubleshooting a MySQL Read Replica Problem.	RDS-EVENT-0045, see details at Amazon RDS Event Categories and Event Messages.

Resource	Alert name and trigger condition	Notes
	RDS-0057 Error create statspack user account Replication on the Read Replica was ended.	Service event. RDS-EVENT-0057, see details at Amazon RDS Event Categories and Event Messages.
	RDS-0058 DB instance read replication ended Error while creating Statspack user account PERFSTAT. Drop the account before adding the Statspack option.	Service event. RDS-EVENT-0058, see details at Amazon RDS Event Categories and Event Messages.
	DB instance partial failover recovery complete The instance has recovered from a partial failover.	Service event. RDS-EVENT-0065 see details at Amazon RDS Event Categories and Event Messages.
	DB instance recovery start The SQL Server DB instance is re-establishing its mirror. Performance will be degraded until the mirror is reestablished. A database was found with non-FULL recovery model. The recovery model was changed back to FULL and mirroring recovery was started. (<dbname>: <recovery model found>[,…])	Service event. RDS-EVENT-0066 see details at Amazon RDS Event Categories and Event Messages.
	DB instance without enhanced monitoring Enhanced Monitoring can't be enabled without the enhanced monitoring IAM role. For information about creating the enhanced monitoring IAM role, see To create an IAM role for Amazon RDS Enhanced Monitoring.	Service event. RDS-EVENT-0079 see details at Amazon RDS Event Categories and Event Messages.
	DB instance enhanced monitoring disabled Enhanced Monitoring was disabled due to an error making the configuration change. It's likely that the enhanced monitoring IAM role is configured incorrectly. For information about creating the enhanced monitoring IAM role, see To create an IAM role for Amazon RDS Enhanced Monitoring.	Service event. RDS-EVENT-0080 see details at Amazon RDS Event Categories and Event Messages.
	Invalid permissions recovery S3 bucket The IAM role that you use to access your Amazon S3 bucket for SQL Server native backup and restore is configured incorrectly. For more information, see Setting Up for Native Backup and Restore.	Service event. RDS-EVENT-0081 see details at Amazon RDS Event Categories and Event Messages.

Resource	Alert name and trigger condition	Notes
	Low storage alert when the DB instance has consumed more than 90% of its allocated storage.	Service event. RDS-EVENT-0089 see details at Amazon RDS Event Categories and Event Messages.
	Notification service when scaling failed for the Aurora Serverless DB cluster.	Service event. RDS-EVENT-0143 see details at Amazon RDS Event Categories and Event Messages.
RedShift cluster	The health of the cluster <= 0 for 5 min, 2 consecutive times	For more information, see Monitoring Amazon Redshift using CloudWatch metrics.
	The maintenance mode of the cluster >= 1 for 5 min, 1 consecutive time	
	The average amount of time taken for disk read >= 1 for 5 min, 1 consecutive time	
	The average amount of time taken for disk write >= 1 for 5 min, 1 consecutive time	

For information on remediation efforts, see AMS automatic remediation of alerts (p. 140).

Tag-based Alarm Manager

AMS Accelerate applies alarms to your AWS resources using the tag-based Alarm Manager to implement a baseline monitoring strategy and ensure that all your AWS resources are monitored and protected. By integrating with the tag-based Alarm Manager, you can customize the configuration of your AWS resources based on their type, platform, and other tags, to ensure the resources are monitored. Alarm Manager is deployed to your account during onboarding.

How Alarm Manager works

When your account is onboarded to AMS Accelerate, two JSON documents, called configuration profiles, are deployed in your account in AWS AppConfig. Both profile documents reside in the Alarm Manager application and in the AMS Accelerate infrastructure environment.

The two configuration profiles are named **AMSManagedAlarms** (the default configuration profile) and **CustomerManagedAlarms** (the customization configuration profile).

- Default configuration profile:
  - The configuration found in this profile contains the default configuration that AMS Accelerate deploys in all customer accounts. This configuration contains the default AMS Accelerate monitoring policy, which you should not modify because AMS Accelerate can update this profile at any time, erasing any changes you have made.
  - If you want to modify or disable any of these definitions, see Modifying the default configuration (p. 137) and Disabling the default configuration (p. 139).
- Customization configuration profile:

- Any configuration in this profile is entirely managed by you; AMS Accelerate does not overwrite this profile, unless you explicitly request it.
- You can specify any custom alarm definitions you want in this profile, and you can also specify modifications to the AMS Accelerate-managed default configuration. For more information, see Modifying the default configuration (p. 137) and Disabling the default configuration (p. 139).
- If you update this profile, Alarm Manager automatically enforces your changes across all relevant resources in your AWS account. Note that while your changes are enacted automatically, they may take up to 60 minutes to take effect.
- You can update this profile using the AWS Management Console or AWS CLI/SDK tools. See the AWS AppConfig User Guide for instructions about updating a configuration.
- The customization profile is initially empty; however, any alarm definitions placed in the profile document are enforced, in addition to the default configuration.

All CloudWatch alarms created by the Alarm Manager contain the tag key **ams:alarm-manager:managed** and tag value **true**. This is to ensure that the Alarm Manager manages only those alarms that it creates, and won't interfere with any of your own alarms. You can see these tags using the Amazon CloudWatch ListTagsForResource API.

> **Important**
> If custom alarm definitions and default alarm definitions are specified with the same ConfigurationID (see Configuration profile document format for monitoring (p. 133)), the custom definitions take priority over default rules.

Getting started with Alarm Manager

By default, when you onboard with AMS Accelerate, your configuration is deployed to AWS AppConfig, defining an alarm baseline for your resources. The alarm definitions are applied only to resources with the **ams:rt:*** tags. We recommend that these tags be applied using the Resource Tagger (p. 37): you set up a basic Resource Tagger configuration in order to let AMS Accelerate know which resources you want managed.

Use Resource Tagger to apply the tag key **ams:rt:ams-managed** with tag value **true** to any resources you want AMS Accelerate to monitor.

The following is an example Resource Tagger customization profile that you can use to opt in to monitoring for all of your Amazon EC2 instances. For general information, see Resource Tagger (p. 37).

```
{
    "AWS::EC2::Instance": {
        "AMSManageAllEC2Instances": {
            "Enabled": true,
            "Filter": {
                "InstanceId": "*"
            },
            "Tags": [
                {
                    "Key": "ams:rt:ams-managed",
                    "Value": "true"
                }
            ]
        }
    }
}
```

For information about how to apply this Resource Tagger configuration, see Viewing or making changes to the Resource Tagger configuration (p. 46).

Providing your own tags using Resource Tagger

The tag-based Alarm Manager manages the lifecycle of per-resource CloudWatch alarms; however, it requires that the managed resources have specific tags defined by AMS Accelerate. To use the Resource Tagger to apply the default set of AMS-managed alarms to both Linux and Windows based instances, follow these steps.

1. Browse to the AppConfig console within your account.
2. Select the ResourceTagger application.
3. Select the **Configuration profiles** tab, and then select **CustomerManagedTags**.
4. Click **Create** to create a new profile.
5. Select **JSON** and define your configuration. The following example associates the backup plan with instances across all platforms. For more examples of filter and platform definition, see Resource Tagger (p. 37).

```
{
    "AWS::EC2::Instance": {
        "AccelerateBackupPlan": {
            "Enabled": true,
            "Filter": {
                "Fn::AND": [
                    {
                        "Platform": "*"
                    }
                ]
            },
            "Tags": [
                {
                    "Key": "ams:rt:ams-managed",
                    "Value": "true"
                }
            ]
        }
    }
}
```

6. Click **Create hosted configuration version**.
7. Click **Start deployment**.
8. Define the following deployment details:

```
Environment: AMSInfrastructure
          Hosted configuration version: <Select the version that you have just created>
            Deployment Strategy: AMSNoBakeDeployment
```

9. Click **Start deployment**.

Your instances become tagged with `"ams:rt:ams-managed": "true"` which ensures that additional `"ams:rt:ams-monitoring-policy": "ams-monitored"` and `"ams:rt:ams-monitoring-policy-platform": "ams-monitored-linux"` are applied to the instances. These tags then result in the appropriate alarms being created for the instance. For more information about this process, see Monitoring tags (p. 34).

Providing your own tags (without using Resource Tagger)

The tag-based Alarm Manager manages the lifecycle of per-resource CloudWatch alarms; however, it requires that the managed resources have specific tags defined by AMS Accelerate. AMS Accelerate provides a default configuration profile that assumes that your tags have been applied by Resource Tagger.

If you want to use an alternate method of applying tags to your resources, such as AWS CloudFormation or Terraform, and not Resource Tagger, you need to disable the Resource Tagger so that it doesn't apply tags to your resources and compete with your chosen tagging method. For instructions on changing your custom Resource Tagger configuration profile to enable read-only mode, see Preventing Resource Tagger from modifying resources (p. 42).

After the Resource Tagger has been set to read-only mode, and the configuration profile is deployed, use your chosen tagging method to apply tags to your resources according to the following guidelines:

Resource type	Tag key	Tag value
All supported resources (described in this table)	ams:rt:ams-monitoring-policy	ams-monitored
EC2 instances (Linux)	ams:rt:ams-monitoring-policy-platform	ams-monitored-linux
EC2 instances (Windows)	ams:rt:ams-monitoring-policy-platform	ams-monitored-windows
Elasticsearch Domain with KMS	ams:rt:ams-monitoring-with-kms	ams-monitored-with-kms
Elasticsearch Domain with Dedicated Master Node	ams:rt:ams-monitoring-with-master	ams-monitored-with-master

Resources that have these tag keys and values are managed by the AMS Accelerate Alarm Manager.

Providing tags using AWS CloudFormation

Note
Make sure you have set Resource Tagger to read-only mode first before applying tags using AWS CloudFormation, otherwise Resource Tagger may modify the tags based on the configuration profile. For information on setting Resource Tagger to read-only mode, and guidelines on providing your own tags, see Providing your own tags (without using Resource Tagger) (p. 131).

To apply tags using AWS CloudFormation, you can apply tags at the stack level (see CloudFormation Resource Tags) or, at the individual resource level, (for example, see Creating EC2 Instance Tags).

The following is an example of how you can apply AMS Accelerate alarm management tags to an Amazon EC2 instance managed by AWS CloudFormation:

```
Type: AWS::EC2::Instance
Properties:
  InstanceType: "t3.micro"
  # ...other properties...

  Tags:
    - Key: "aws:rt:ams-monitoring-policy"
      Value: "ams-monitored"
    - Key: "aws:rt:ams-monitoring-policy-platform"
      Value: "ams-monitored-linux"
```

The following is an example of how you can apply AMS Accelerate alarm management tags to an Auto Scaling group managed by AWS CloudFormation. Note that the Auto Scaling group will propagate its tags to Amazon EC2 instances that are created by it:

```
Type: AWS::AutoScaling::AutoScalingGroup
Properties:
  AutoScalingGroupName: "TestASG"

  # ...other properties...

  Tags:
    - Key: "aws:rt:ams-monitoring-policy"
      Value: "ams-monitored"
    - Key: "aws:rt:ams-monitoring-policy-platform"
      Value: "ams-monitored-linux"
```

Providing tags using Terraform

> **Note**
> Make sure you have set Resource Tagger to read-only mode first before applying tags using AWS CloudFormation, otherwise Resource Tagger may modify the tags based on the configuration profile. For information on setting Resource Tagger to read-only mode, and guidelines on providing your own tags, see Providing your own tags (without using Resource Tagger) (p. 131).

For a description of how to manage resource tags using Terraform, see the Terraform documentation Resource Tagging.

The following is an example of how you can apply AMS Accelerate alarm management tags to an Amazon EC2 instance managed by Terraform.

```
resource "aws_instance" "test_linux_instance" {
  # ...ami and other properties...

  instance_type = "t3.micro"

  tags = {
    "aws:rt:ams-monitoring-policy" = "ams-monitored"
    "aws:rt:ams-monitoring-policy-platform" = "ams-monitored-linux"
  }
}
```

The following is an example of how you can apply AMS alarm management tags to an Auto Scaling group managed by Terraform. Note that the Auto Scaling group propagates its tags to EC2 instances that are created by it:

```

  resource "aws_autoscaling_group" "test_asg" {
  name = "terraform-test"
  # ...other properties...
  tags = {
    "aws:rt:ams-monitoring-policy" = "ams-monitored"
    "aws:rt:ams-monitoring-policy-platform" = "ams-monitored-linux"
  }
}
```

Configuration profile document format for monitoring

Both the default configuration profile document and the customization configuration profile document follow the same structure:

```
  {
    "<ResourceType>": {
        "<ConfigurationID>": {
            "Enabled": true,
            "Tag": {
                "Key": "...",
                "Value": "..."
            },
            "AlarmDefinition": {
                ...
            }
        },
        "<ConfigurationID>": {
            ...
        }
    },
    "<ResourceType>": {
        ...
    }
}
```

- **ResourceType**: This key must be one of the following supported strings. The configuration within this JSON object will relate only to the specified AWS resource type. Supported resource types:
  - AWS::EC2::Instance
  - AWS::EC2::Instance::Disk
- **ConfigurationID**: This key must be unique in the profile, and uniquely names the following block of configuration. If two configuration blocks in the same **ResourceType** block have the same **ConfigurationID**, the one that appears latest in the profile takes effect. If you specify a **ConfigurationID** in your customization profile that is the same as one specified in the default profile, the configuration block defined in the customization profile takes effect.
  - **Enabled**: (optional, default=true) Specifies if the configuration block will take effect. Set this to false to disable a configuration block. A disabled configuration block behaves as if it's not present in the profile.
  - **Tag**: Specifies the tag that this alarm definition applies to. Any resource (of the appropriate resource type) that has this tag key and value will have a CloudWatch alarm created with the given definition. This field is a JSON object with the following fields:
    - **Key**: The key of the tag to match. Keep in mind that if you're using Resource Tagger to apply the tags to the resource, the key for the tag will always begin with **ams:rt:**.
    - **Value**: The value of the tag to match.
  - **AlarmDefinition**: Defines the alarm to be created. Alarm Manager currently only supports single-metric alarms. This is a JSON object whose fields are passed as is to the CloudWatch `PutMetricAlarm` API call (with the exception of pseudoparameters; for more information, see Configuration profile - pseudoparameter substitution (p. 135)). For information about what fields are required, see the PutMetricAlarm documentation.

OR

CompositeAlarmDefinition: Defines a composite alarm to be created. When you create a composite alarm, you specify a rule expression for the alarm that takes into account the alarm state of other alarms that you have created. This is a JSON object whose fields are passed as-is to the `CloudWatchPutCompositeAlarm`. The composite alarm goes into ALARM state only if all conditions of the rule are met. The alarms specified in a composite alarm's rule expression can include metric alarms and other composite alarms. For information about what fields are required, see the PutCompositeAlarm documentation.

Both options provide the following fields:

- **AlarmName**: Specifies the name of the alarm you want to create for the resource. This field has all of the same rules as specified in the PutMetricAlarm documentation; however, since the alarm name must be unique in a Region, the Alarm Manager has one additional requirement: you must specify the unique identifier pseudoparameter in the name of the alarm (otherwise, Alarm Manager appends the unique identifier of the resource to the front of the alarm name). For example, for the **AWS::EC2::Instance** resource type, you must specify `${EC2::InstanceId}` in the alarm name, or it's implicitly added at the start of the alarm name. For the list of identifiers, see Configuration profile - pseudoparameter substitution (p. 135).

  All other fields are as specified in the PutMetricAlarm or the PutCompositeAlarm documentation.
- **AlarmRule**: Specifies which other alarms are to be evaluated to determine this composite alarm's state. For each alarm that you reference, they have to be either exist in CloudWatch or specified in Alarm Manager configuration profile in your account.

Important

You can specify either **AlarmDefinition** or **CompositeAlarmDefinition** in your Alarm Manager configuration document, But they both can’t be used at the same time.

In the following example, the system creates an alarm when two specified metric alarms exceeds its threshold:

```
{
  "AWS::EC2::Instance": {
    "LinuxResourceAlarm": {
      "Enabled": true,
      "Tag": {
        "Key": "ams:rt:mylinuxinstance",
        "Value": "true"
      },
      "CompositeAlarmDefinition": {
        "AlarmName": "${EC2::InstanceId} Resource Usage High",
        "AlarmDescription": "Alarm when a linux EC2 instance is using too much CPU and too
 much Disk",
        "AlarmRule": "ALARM(\"${EC2::InstanceId}: Disk Usage Too High -
 ${EC2::Disk::UUID}\") AND ALARM(\"${EC2::InstanceId}: CPU Too High\")"
      }
    }
  }
}
```

Important

When Alarm Manager is not able to create or delete an alarm due to broke configuration, it sends the notification to the **Direct-Customer-Alerts** SNS topic. This alarm is called **AlarmDependencyError**.

We highly recommend that you have confirmed your subscription to this SNS topic. To receive messages published to a topic, you must subscribe an endpoint to the topic. For details, see Step 1: Create a topic.

Note
Many of the AMS Accelerate-provided baseline alarm definitions list the SNS topic, **MMS-Topic**, as a target. This is for use in the AMS Accelerate monitoring service, and is the transport mechanism for your alarm notifications to get to AMS Accelerate. Do not specify **MMS-Topic** as the target for any alarms other than those provided in the baseline (and overrides of the same), as the service ignores unknown alarms. It **does not** result in AMS Accelerate acting on your custom alarms.

Configuration profile - pseudoparameter substitution

In either of the configuration profiles, you can specify pseudoparameters that are substituted in place as follows:

- Global - anywhere in the profile:
  - ${AWS::AccountId}: Replaced with your AWS account ID
  - ${AWS::Partition}: Replaced with the partition of the AWS Region the resource is in (this is 'aws' for most Regions); for more information, see the entry for partition in the ARN reference.
  - ${AWS::Region}: Replaced with the Region name of the Region that your resource is deployed to (for example us-east-1)
- In an **AWS::EC2::Instance** resource type block:
  - ${EC2::InstanceId}: (**identifier**) replaced by the instance ID of your Amazon EC2 instance.
- In an **AWS::EC2::Instance::Disk** resource type block:
  - ${EC2::InstanceId}: (**identifier**) Replaced by the instance ID of your Amazon EC2 instance.
  - ${EC2::Disk::Device}: Replaced by the name of the disk. (Linux only, on instances managed by the CloudWatch Agent).
  - ${EC2::Disk::FSType}: Replaced by the file system type of the disk. (Linux only, on instances managed by the CloudWatch Agent).
  - ${EC2::Disk::Path}: Replaced by the disk path. On Linux, this is the mount point of the disk (for example, /), while in Windows this is the drive label (for example, c:/) (only on instance managed by the CloudWatch Agent).
  - ${EC2::Disk::UUID}: Replaced by a generated UUID that uniquely identifies the disk, this must be specified in the name of the alarm, as an alarm under AWS::EC2::Instance::Disk resource type will create one alarm per volume. Specifying ${EC2::Disk::UUID} will maintain uniqueness of alarm names.

Note
All parameters marked with **identifier** are used as a prefix for the name of created alarms, unless you specify that identifier in the alarm name.

Configurations example

In the following example, the system creates an alarm for each disk attached to the matching Linux instance.

```
{
    "AWS::EC2::Instance::Disk": {
        "LinuxDiskAlarm": {
            "Tag": {
                "Key": "ams:rt:mylinuxinstance",
                "Value": "true"
            },
            "AlarmDefinition": {
                "MetricName": "disk_used_percent",
                "Namespace": "CWAgent",
                "Dimensions": [
                    {
                        "Name": "InstanceId",
                        "Value": "${EC2::InstanceId}"
                    },
                    {
                        "Name": "device",
                        "Value": "${EC2::Disk::Device}"
                    },
                    {
                        "Name": "fstype",
                        "Value": "${EC2::Disk::FSType}"
                    },
                    {
                        "Name": "path",
                        "Value": "${EC2::Disk::Path}"
                    }
                ],
                "AlarmName": "${EC2::InstanceId}: Disk Usage Too High - ${EC2::Disk::UUID}"
                ...
            }
        }
    }
}
```

In the following example, the system creates an alarm for each disk attached to the matching Windows instance.

```
{
     "AWS::EC2::Instance::Disk": {
        "WindowsDiskAlarm": {
            "Tag": {
                "Key": "ams:rt:mywindowsinstance",
                "Value": "true"
            },
            "AlarmDefinition": {
                "MetricName": "LogicalDisk % Free Space",
                "Namespace": "CWAgent",
                "Dimensions": [
                    {
                        "Name": "InstanceId",
                        "Value": "${EC2::InstanceId}"
                    },
                    {
                        "Name": "objectname",
                        "Value": "LogicalDisk"
                    },
                    {
                        "Name": "instance",
                        "Value": "${EC2::Disk::Path}"
                    }
                ],
                "AlarmName": "${EC2::InstanceId}: Disk Usage Too High - ${EC2::Disk::UUID}"
                ...
            }
        }
    }
}
```

Viewing Alarm Manager configuration

Both the **AMSManagedAlarms** and **CustomerManagedAlarms** can be reviewed in AppConfig with GetConfiguration.

The following is an example of the `GetConfiguration` call:

```
aws appconfig get-configuration --application AMSAlarmManager --environment
 AMSInfrastructure --configuration AMSManagedAlarms --client-id any-string outfile.json
```

- **Application**: this is AppConfig's logical unit to provide capabilities; for the Alarm Manager, this is `AMSAlarmManager`
- **Environment**: this is the AMSInfrastructure environment
- **Configuration**: to view AMS Accelerate baseline alarms, the value is `AMSManagedAlarms`; to view customer alarm definitions, the configuration is `CustomerManagedAlarms`
- **Client ID**: this is a unique application instance identifier, which can be any string
- The alarm definitions can be viewed in the specified output file, which in this case is `outfile.json`

You can see which version of configuration is deployed to your account by viewing the past deployments in the AMSInfrastructure environment.

Changing the configuration

To add or update new alarm definitions, invoke the CreateHostedConfigurationVersion API.

This is a Linux command line command that generates the parameter value in base64, which is what the AppConfig CLI command expects. For information, see the AWS CLI documentation, Binary/Blob (binary large object).

As an example:

```
aws appconfig create-hosted-configuration-version --application-id application-id --
configuration-profile-id configuration-profile-id --content base64-string
  --content-type application/json
```

- **Application ID:** ID of the application AMSAlarmManager; you can find this out with the ListApplications API call.
- **Configuration Profile ID**: ID of the configuration CustomerManagedAlarms; you can find this out with the ListConfigurationProfiles API call.
- **Content**: Base64 string of the content, to be created by creating a document and encoding it in base64: cat alarms-v2.json | base64 (see Binary/Blob (binary large object)).

  Content Type: MIME type, `application/json` because alarm definitions are written in JSON.

> **Important**
> Restrict access to the StartDeployment and StopDeployment API actions to trusted users who understand the responsibilities and consequences of deploying a new configuration to your targets.

To learn more about how to use AWS AppConfig features to create and deploy a configuration, see Working with AWS AppConfig.

Modifying the default configuration

While you can't modify the default configuration profile, you can provide overrides to the defaults by specifying a configuration block in your customization profile with the same **ConfigurationID** as the default configuration block. If you do this, your whole configuration block overwrites the default configuration block for which tagging configuration to apply.

For example, consider the following default configuration profile:

```
{
    "AWS::EC2::Instance": {
        "AMSManagedBlock1": {
            "Enabled": true,
            "Tag": {
                "Key": "ams:rt:ams-monitoring-policy",
                "Value": "ams-monitored"
            },
            "AlarmDefinition": {
                "AlarmName": "${EC2::InstanceId}: AMS Default Alarm",
                "Namespace": "AWS/EC2",
                "MetricName": "CPUUtilization",
                "Dimensions": [
                    {
                        "Name": "InstanceId",
                        "Value": "${EC2::InstanceId}"
                    }
                ],
                "Threshold": 5,
                ...
            }
        }
    }
}
```

In order to change the threshold of this alarm to 10, **you must provide the entire alarm definition**, not only the parts you want to change. For example, you might provide the following customization profile:

```
{
    "AWS::EC2::Instance": {
        "AMSManagedBlock1": {
            "Enabled": true,
            "Tag": {
                "Key": "ams:rt:ams-monitoring-policy",
                "Value": "ams-monitored"
            },
            "AlarmDefinition": {
                "AlarmName": "${EC2::InstanceId}: AMS Default Alarm",
                "Namespace": "AWS/EC2",
                "MetricName": "CPUUtilization",
                "Dimensions": [
                    {
                        "Name": "InstanceId",
                        "Value": "${EC2::InstanceId}"
                    }
                ],
                "Threshold": 10,
                ...
            }
        }
    }
}
```

> **Important**
> Remember to deploy your configuration changes after you have made them. In SSM AppConfig, you must deploy a new version of the configuration after creating it.

Deploying configuration changes

Once the customization is completed, these changes must be deployed through StartDeployment.

```
  aws appconfig start-deployment --application-id application_id
  --environment-id environment_id Vdeployment-strategy-id
  deployment_strategy_id --configuration-profile-id configuration_profile_id --
configuration-version 1
```

- **Application ID**: ID of the application `AMSAlarmManager`, you can find this with the ListApplications API call.
- **Environment ID**: You can find this with the ListEnvironments API call.
- **Deployment Strategy ID**: You can find this with the ListDeploymentStrategies API call.
- **Configuration Profile ID**: ID of `CustomerManagedAlarms`; you can find this with the ListConfigurationProfiles API call.
- **Configuration Version**: The version of the configuration profile to be deployed.

> **Important**
> Alarm Manager applies the alarm definitions as specified in the configuration profiles. Any manual modifications you make with the AWS Management Console or CloudWatch CLI/SDK to the CloudWatch alarms is automatically reverted back, so make sure your changes are defined through Alarm Manager. To understand which alarms are created by the Alarm Manager, you can look for the `ams:alarm-manager:managed` tag with value `true`.
> Restrict access to the StartDeployment and StopDeployment API actions to trusted users who understand the responsibilities and consequences of deploying a new configuration to your targets.

To learn more about how to use AWS AppConfig features to create and deploy a configuration, see the documentation.

Rolling back changes

You can roll back alarm definitions through the same deployment mechanism by specifying a previous configuration profile version and running StartDeployment.

Disabling the default configuration

AMS Accelerate provides the default configuration profile in your account based on the baseline alarms. However, this default configuration can be disabled by overriding any of the alarm definitions. You can disable a default configuration rule by overriding the **ConfigurationID** of the rule in your customization configuration profile and specifying the enabled field with a value of false.

For example, if the following configuration was present in the default configuration profile:

```
{
    "AWS::EC2::Instance": {
        "AMSManagedBlock1": {
            "Enabled": true,
            "Tag": {
                "Key": "ams:rt:ams-monitoring-policy",
                "Value": "ams-monitored"
            },
            "AlarmDefinition": {
                ...
            }
        }
    }
```

You could disable this tagging rule by including the following in your customization configuration profile:

```
{
    "AWS::EC2::Instance": {
        "AMSManagedBlock1": {
            "Enabled": false
        }
    }
}
```

To make these changes, the CreateHostedConfigurationVersion API must be called with the JSON profile document (see Changing the configuration (p. 137)) and subsequently must be deployed (see Deploying configuration changes (p. 138)). Note that when you create the new configuration version, you must also include any previously created custom alarms that you want in the JSON profile document.

> **Important**
> When AMS Accelerate updates the default configuration profile, it's not calibrated against your configured custom alarms, so review changes to the default alarms when you're overriding them in your customization configuration profile.

AMS automatic remediation of alerts

Some alerts are automatically remediated by AMS. This section describes how this remediation works and the conditions that must be met for the remediation to take place.

Alert name	Description	Remediation
Status Check Failed	This alarm indicates that the instance is running on degraded hardware or entered a fault state.	Our remediation first validates instance accessibility. If confirmed that accessibility is impacted, it stops the instance and starts it again so it can be migrated to new underlying hardware.
Root Volume Usage	This alarm indicates that the root volume (C: Drive in Windows) of your EC2 instance is filling up.	The remediation first deletes temporary files. If this does not free up required space, it extends the volume to prevent downtime if the volume were to get full.
Non-Root Volume Usage	This alarm indicates that an attached volume (not root or C:) is filling up.	The remediation first deletes temporary files. If this does not free up required space, it extends the volume to prevent downtime if the volume were to get full.
RDS-EVENT-0089	This alarm indicates that the DB instance has consumed more than 90% of its allocated storage.	The remediation first validates the DB is in a modifiable and available/storage-full state. It will attempt to increase the allocated storage via cloudformation changeset, if stack drift is already detected it will fall back to RDS API to prevent downtime.
RDS-EVENT-0007	This alarm indicates that the allocated storage for the DB instance has been exhausted.	The remediation first validates the DB is in a modifiable and available/storage-full state. It will attempt to increase the allocated storage via cloudformation changeset, if stack drift is already

Alert name	Description	Remediation
		detected it will fall back to RDS API to prevent downtime.

EC2 status check failure remediation automation

These are some notes about how AMS auto-remediation works with EC2 status check failure issues.

- Your EC2 instance has become unreachable. In order to recover it, it must be stopped and started again so it's migrated to new hardware.
- The automation is not able to recover your instance if the root of the problem is within the OS.; for example, missing devices in fstab, kernel corruption, and so on.
- If your instance belongs to an Auto Scaling group, the automation takes no action. The autoscaling replaces the instance.
- The remediation doesn't take action if EC2 Auto Recovery is enabled for this instance.

EC2 volume usage remediation automation

How AMS auto-remediation works with EC2 volume usage issues.

- Before trying to extend the volume, the automation performs cleanup tasks (Windows: Disk Cleaner Linux: Logrotate + Simple Service Manager Agent Log removal) on the instance to try to free up space.
- This cleanup step will not be run on EC2 "T" family instances due to its reliance on CPU credits for continued functionality.
- The automation doesn't take action if the affected volume is already bigger than 2 TiB.
- The automation doesn't extend volumes that are part of Logical Volume Manager (LVM) or RAID.
- On Linux, the automation only supports extending file systems of type EXT2, EXT3, EXT4 and XFS.
- On Windows, the automation only supports New Technology File System (NTFS) and Resilient File System (ReFS).
- The automation doesn't extend instance stored backed volumes.
- The capacity expansion portion of the automation only occurs once every 6 hours with a 3-time volume expansion lifetime limit.

Under these EC2 volume usage issues, AMS reaches out to you through an outbound service request to determine the next actions to take.

Amazon RDS low storage event remediation automation

How AMS auto-remediation works with Amazon RDS low storage event issues.

- Before trying to extend the Amazon RDS instance storage, the automation performs several checks to ensure the Amazon RDS instance is in a modifiable and available, or storage-full, state.
- Where CloudFormation stack drift is detected, remediation occurs through Amazon RDS API.
- The remediation action does not run in the following scenarios:
  - The Amazon RDS instance status is not "available" or "storage-full".
  - The Amazon RDS instance storage is not currently modifiable (such as when the storage has been modified in the last 6 hours).
  - The Amazon RDS instance has auto-scaling storage enabled.
  - The Amazon RDS instance is not a resource within a CloudFormation stack.

- Remediation is limited to 1 expansion per 6 hours and no more than 3 expansions within a rolling fourteen day period.
- Where the above states are met, AMS reaches out to you with an outbound incident to determine next actions.

Creating additional CloudWatch alarms

You can create additional CloudWatch alarms for AMS Accelerate using custom CloudWatch metrics and alarms for Amazon EC2 instances.

Produce your application monitoring script and custom metric. For more information and access to example scripts, see Monitoring Memory and Disk Metrics for Amazon EC2 Linux Instances.

The CloudWatch monitoring scripts for Linux Amazon EC2 instances demonstrate how to produce and consume custom CloudWatch metrics. These sample Perl scripts comprise a fully functional example that reports memory, swap, and disk space utilization metrics for a Linux instance.

> **Important**
> AMS Accelerate does not monitor CloudWatch alarms created by you.

Backup management in AMS Accelerate

Topics

You use AWS Backup for continuity management in your AWS and AMS Accelerate accounts. AWS Backup is a native AWS service.

To learn more, see What Is AWS Backup? and AWS Backup: How It Works.

With AWS Backup you can create different backup policies called backup plans, which define what AWS resources to protect, how frequently they need to be backed up, and the backup retention. Each time AWS Backup backs up a resource, it creates a recovery point, and stores it into a backup vault. In AWS Backup, a backup vault is a container that you organize your backups in.

AMS Accelerate builds on top of native AWS Backup to provide the following for new or existing AWS Backup users:

- AMS Accelerate monitors and attempts to remediate failed AWS Backup creation and restore jobs, irrespective of what backup plan is used (custom or AMS Accelerate default).
- AMS Accelerate provides a backup coverage report which includes both protected and unprotected resources, irrespective of what backup plan is used (custom or AMS Accelerate default). Note that only you can confirm that the backup we create is fit for purpose.

For information on backup reports, see Backup reporting (p. 70).

Using your existing AWS Backup configuration in AMS Accelerate

After creating a backup plan, you can edit the plan; for example, you can add tags, or you can add, edit, or delete backup rules. Any changes that you make to a backup plan have no effect on existing backups created by the backup plan. The changes apply only to backups that are created in the future.

Common AWS Backup operations

If you have the AWS Management Console or API access, then you can use AWS Backup directly to create, manage and restore your backups.

To configure AWS Backup, see the following documentation:

- Backup plan: Managing Backup Plans with AWS Backup
- Backup plan: Configuring Cross Region Backups
- Backup plan: Delete a Backup Plan
- Backup plan: Starting a Backup Job
- Backup job: Stopping a Backup Job
- Backup: Restoring a Backup

IAM permissions to perform common AWS Backup operations

AMS Accelerate creates an IAM role in your account called **ams-backup-iam-role**. Different personas can use this role as follows to perform common AWS Backup operations:

- **Backup administrator**: Create, modify and delete AWS Backup plans and AWS Backup vaults. Use the AWS-managed policy **arn:aws:iam::aws:policy/AWSBackupFullAccess**, along with the following policy, to be able to associate the AMS Accelerate-managed AWS Backup role **ams-backup-iam-role** to your AWS Backup plans.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::ACCOUNT_ID:role/ams-backup-iam-role",
            "Effect": "Allow"
        }
    ]
}
```

- **Backup operator**: Assign resources to existing AWS Backup plans, create on-demand backups and restore backups as needed. Use the AWS Managed policy **arn:aws:iam::aws:policy/AWSBackupOperatorAccess**, along with the following policy to be able to use the AMS Accelerate-managed IAM role **ams-backup-iam-role** for AWS Backup operations such as create, copy, or restore jobs.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::ACCOUNT_ID:role/ams-backup-iam-role",
            "Effect": "Allow"
        }
    ]
}
```

Adding AWS Backup to your AMS Accelerate accounts

If you do not use AWS Backup in your AWS accounts, you can use the AMS Accelerate configuration of AWS Backup.

AMS Accelerate backup configuration

AMS Accelerate, during onboarding, deploys AMS Accelerate-managed backup vaults and backup plans to your account, collectively referred to as the AMS Accelerate backup configuration. Use the AMS Accelerate backup configuration if you do not have an existing AWS Backup configuration and want to ensure that you're following AWS Backup best practices.

> **Important**
> The AMS Accelerate backup configuration is overwritten when updated by AMS Accelerate, so changes made to it are removed. If you need customizations, create a new plan and modify it.

AMS Accelerate-managed AWS Backup plans

The AMS Accelerate backup plan provides a quick start way for you to protect your AWS resources with a daily, weekly, monthly and yearly schedule, follow AWS Backup best practices, and benefit from the additional backup job monitoring AMS Accelerate offers. You can use the default AMS Accelerate backup plan, or create your own backup vaults and plans.

Default backup plans, AMS Accelerate multi-account landing zone

During the account onboarding, AMS ensures that there is an overarching default backup plan at the account level to safeguard your workloads. The values for the following mandatory fields are set up by default.

- **ams-default-backup-plan**: This plan is a blueprint for AWS Backup best practices. It implements a daily, weekly, monthly, and yearly backup strategy.

  Resource tag key: **ams:rt:backup-orchestrator**

  Resource tag value: **true**

  RuleForDailyBackups schedule expression: `cron(0 0 4 ? * * )` (a daily backup for 04:00 UTC time)

  RuleForDailyBackups delete after days: `7 days`

  RuleForWeeklyBackups schedule expression: `cron(0 0 2 ? * 7)` (a weekly backup for 02:00 UTC time only on Saturday)

  RuleForWeeklyBackups delete after weeks: `4 weeks`

  RuleForMonthlyBackups schedule expression: `cron(0 2 1 * ? *)` (a monthly backup for 02:00 UTC time on day 1 of the month)

  RuleForMonthlyBackups delete after weeks: `26 weeks`

  RuleForYearlyBackups schedule expression: `cron(0 2 1 1 ? *)` (a yearly backup for 02:00 UTC time on day 1 of the month, only in January)

  RuleForYearlyBackups delete after years: `2 years`
- **ams-onboarding-backup-plan**: This plan is used only during account onboarding by AMS Accelerate to protect all supported resources and automatically create frequent, short retention recovery points inside the **ams-onboarding-backups** plan:

  Resource tag key: **ams:rt:backup-orchestrator-onboarding**

  Resource tag value: **true**

  RuleForHourlyBackups schedule expression: `cron(0 * * * *)` (an hourly backup)

  RuleForHourlyBackups delete after weeks: `2 weeks`

AMS Accelerate-managed AWS Backup vaults

AMS Accelerate uses the following backup vaults:

- **ams-automated-backups**: This vault receives all recovery points taken by the AMS Accelerate default AWS Backup plan **ams-default-backup-plan**.
- **ams-manual-backups**: This vault is purpose-built for all backups taken using Start Backup Job automation (AWSManagedServices-StartBackupJob SSM Automation document).
- **ams-onboarding-backups**: This vault receives recovery points taken during account onboarding using the temporary account onboarding backup plan **ams-onboarding-backup-plan**.
- **ams-patch-backups**: This vault stores snapshots taken by the default AWS Backup stack Patch Manager stack.
- **ams-custom-backups**: This vault is purpose-built for all backup plans created outside the default AWS Backup stack.

Associating resources to the backup plan using Resource Tagger

The AMS Accelerate default backup plan uses tag-based management to identify the resources to protect. To use the Resource Tagger to associate your backup plan with specific resources, follow these steps.

1. Browse to the AppConfig console within your account.
2. Select the ResourceTagger application.
3. Select the **Configuration profiles** tab, and then select **CustomerManagedTags**.
4. Click **Create** to create a new profile.
5. Select **JSON** and define your configuration. The following example associates the backup plan with instances across all platforms. For more examples of filter and platform definition, see Resource Tagger (p. 37).

```
{
    "AWS::EC2::Instance": {
        "AccelerateBackupPlan": {
            "Enabled": true,
            "Filter": {
                "Fn::AND": [
                    {
                        "Platform": "*"
                    }
                ]
            },
            "Tags": [
                {
                    "Key": "ams:rt:backup-orchestrator",
                    "Value": "true"
                }
            ]
        }
    }
}
```

6. Click **Create hosted configuration version**.
7. Click **Start deployment**.
8. Define the following deployment details:

```
Environment: AMSInfrastructure
          Hosted configuration version: <Select the version that you have just created>
            Deployment Strategy: AMSNoBakeDeployment
```

9. Click **Start deployment**.

Your instances become tagged with `"ams:rt:backup-orchestrator": "true"`, which ensures that the backup plan is applied.

AWS Backup monitoring and job failure remediation in AMS Accelerate

Managing AWS Backup in AMS Accelerate includes remediating failed backup jobs and reporting backup jobs.

When backup jobs fail, AMS Accelerate Operations is alerted and follows AWS best practices to complete your backup job. After AMS Accelerate Operations has triaged the issue, they alert your account point-of-contact about the failed backup job. The following steps occur when a backup job failure is detected in AMS Accelerate:

1. **AWS Backup job failure**:
   a. AMS Operations is notified of backup job failure, if the failure occurs in an AMS Accelerate-managed AWS Region.
   b. An AMS Operations engineer accesses the account and investigates the cause of the failure. If the backup job failed due to an anomaly, then the engineer escalates the ticket to the AWS Backup service team.
   c. After identifying the root cause, the engineer notifies your account's primary contact. The notification includes the root cause, backup job ID, and resource ID.
2. **AWS Backup job expiration**:
   a. If a backup job fails due to expiration, then the engineer retries the backup job using an on-demand backup job.
   b. The engineer provides a notification, including the root cause, resource ID, backup job ID, and suggestion for mitigating the issue.
3. **Amazon RDS backup job failures**:
   a. The AMS Operations engineer conducts a root cause analysis on the backup failure.
   b. The AMS Operations engineer then notifies your account's point-of-contact of the root cause, backup job ID, resource ID, and suggested mitigation measures.

Note
Support tickets for any AWS Backup-related issues can be submitted through the AWS Support console. To report backup incidents or issues to AMS Accelerate, submit an incident in the AWS Support console and choose **'Service':'AWS Managed Services'** and **'Category': 'AWS Backup'**.

AWS Backup job reporting in AMS Accelerate

Backup status reports, including resource coverage and backup job status, can be sourced from your AMS Accelerate CSDM.

Patch management in AMS Accelerate

Topics

You can use the AMS Accelerate patching system to patch your instances with security related and other types of updates. You can schedule patching an instance, or a group of instances, through the AWS Systems Manager (Systems Manager) maintenance window. You configure the window by using the **AWSManagedServices-PatchInstance** SSM Automation document, which is available through the Systems Manager console.

AMS Accelerate patch management uses the Systems Manager patch baseline functionality to control the definition of the patches that are applied on an instance. The patch baseline contains the list of patches that are pre-approved; for example, all security patches. The compliance of the instance is measured against the patch baseline associated to it. AMS Accelerate, by default, installs all patches available to keep the instance up to date.

> **Note**
> AMS Accelerate applies only operating system (OS) patches. For example, for Windows, only Windows updates are applied, not Microsoft updates.

For information on reports, see Patch reporting (p. 65).

Enable patching access for users

After your account is onboarded to AMS Accelerate patching, AMS Accelerate deploys a managed policy, **amspatchmanagedpolicy**, which contains the required permissions for patching using SSM services. For you to access patching services, have the administrator for your account follow these steps:

Create a role using the AWS Management Console:

1. Sign in to the AWS Management Console and open the IAM console.
2. In the navigation pane of the console, choose **Roles**, then **Create role**.
3. Choose the **Another AWS account** role type.
4. For **Account ID**, enter the AWS account ID to which you want to grant access to your resources.

   The administrator of the specified account can grant permission to assume this role to any IAM user in that account. To do this, the administrator attaches a policy to the user, or a group, that grants permission for the **sts:AssumeRole** action. That policy must specify the role's Amazon Resource Name (ARN) as the resource. Note the following:

   - If you are granting permissions to users from an account that you do not control, and the users will assume this role programmatically, then choose **Require external ID**. The external ID can be any word or number that is agreed upon between you and the administrator of the third-party account.

This option automatically adds a condition to the trust policy that enables the user to assume the role only if the request includes the correct **sts:ExternalID**. For more information, see How to use an external ID when granting access to your AWS resources to a third party.

- If you want to restrict the role to users who sign in with multi-factor authentication (MFA), choose **Require MFA**. This adds a condition to the role's trust policy that checks for an MFA sign-in. A user who wants to assume the role must sign in with a temporary one-time password from a configured MFA device. Users without MFA authentication can't assume the role. For more information about MFA, see Using multi-factor authentication (MFA) in AWS.

5. Choose **Next: Permissions**.

   IAM includes a list of the policies managed by and by customers in your account. Choose the policies **amspatchmanagedpolicy**, **customer_ssm_automation_policy**, and **customer_ssm_automation_policy2** for the permissions policy. After you create the policy, close that tab and return to your original tab. Select the check box next to the permissions policies that you want anyone who assumes the role to have. If you prefer, you can select no policies at this time, and then attach policies to the role later. By default, a role has no permissions.

(Optional) Set a permissions boundary. To do this, follow these steps:

1. Open the **Set permissions boundary** section and choose **Use a permissions boundary to control the maximum role permissions**. Choose the policy to use for the permissions boundary.

   Choose **Next: Tags**.
2. (Optional) Add metadata to the role by attaching tags as key–value pairs. For more information about using tags in IAM, see Tagging IAM users and roles.
3. Choose **Next: Review**.
4. For **Role name**, enter a name for your role (for example, PatchRole). Role names must be unique within your AWS account; they're not case sensitive (so you can't create roles named both **PRODROLE** and **prodrole**). Because other AWS resources might reference the role, roles names can't be edited after they've been created.
5. (Optional) For **Role description**, enter a description for the new role.
6. Review the role and then choose **Create role**.

Create an SSM maintenance window for patching

In AMS Accelerate patching, you have access to your SSM console where you configure your patching maintenance window.

Using the SSM Maintenance Window console, you configure the schedule for patching the instances. During patching, the system takes a snapshot of the root volume, which AMS Operations engineers use to restore the instance's root volume, if required. Additionally, an SSM OpsItem is created to track the failure.

To set up patching using a maintenance window, follow these steps:

1. In the SSM console, select **Maintenance Windows** under the **Change Management vs Actions & Change** pane on the left side and then choose **Create Maintenance Window** on the top right of the screen. Configure the window:

   - **Name**: Provide a meaningful name for the maintenance window.
   - **Description**: Optional
   - **Schedule**:
     - **Specify with:** (Choose a method.)

        - **Cron schedule builder**: Radio buttons for **Default**, **Hourly**, or **Daily**.
        - **Rate schedule builder**: Specify number and units, for example every **90 minutes**.
        - **CRON/Rate expression**: Specify a Unix CRON expression, for example, `cron(0 30 23 ? * TUE#2 *)`.
    - **Duration**: The duration of a maintenance window in hours. (AMS Accelerate recommends a minumum of two hours, plus an additional hour per every 50 instances.)
    - **Stop initiating tasks**: Don't start new maintenance tasks in the last `x` hour(s) of the maintenance window. This buffer gives maintenance tasks time to complete. (Must be less than the total **Duration** value.)
    - **Window start date**: (Optional) Start date/time of maintenance schedule.
    - **Window end date**: (Optional) End date/time of maintenance schedule.
    - **Schedule timezone**: (Optional) Choose a time zone.
    - **Schedule offset**: (Optional) Delay all windows by a fixed number of days. For example, if you specified a CRON expression that is always a Tuesday, an offset of **1** will shift maintenance by one day, to Wednesday.
2. Click **Create maintenance window**. This takes you back to the maintenance window home page. Select the newly created maintenance window.
3. Go to the **Targets** tab, choose **Register target**.

    - (Optional) Provide a meaningful target name. This helps you identify this target. For example: application1-qa.
    - Patching windows can support either targeting from tags or choosing the instances manually
        - For a tag target under the **Targets** section, choose **Specify instance tags**.
        - Provide the tag key and value for the instance the patching maintenance window will target then choose **Add** (for example, ApplicationId (tag key), App1 (tag value)).
        - For instance target, under the **Targets** section, select **Choose instances manually**.
        - Select the instances you want to target.
    - Choose **Register target**.
4. Go to the **Tasks** tab of the maintenance window and choose **Register Task**, then choose **Register Automation Task**.

    1. (Optional) Provide a meaningful task name. For example: AmsPatch.
    2. Under the **Automation document**, for the search box, choose **Owner**, then **Shared documents**.
    3. Choose in the search box, choose **Document name prefix, Equals** and type: **AWSManagedServices-PatchInstance**.
    4. Choose the document with the name identical to **AWSManagedServices-PatchInstance**.

        Important
        Do not choose **AWSManagedServices-PatchInstanceFromMaintenanceWindow**.
        AMS Accelerate does not support this.
    5. Under document version, choose **Default version at runtime**.
    6. Under the **Targets** section, select the target matching the name or the target registered previously.
    7. In the **Rate control** section, choose percentages:
        - **MaxConcurrency**: AMS Accelerate recommends **50%**. (How many instances can be patched simultaneously.)
        - **MaxErrors**: AMS Accelerate recommends **50%**. (Stop maintenance if errors exceed this threshold.)
    8. In the **IAM service role** section, choose **Use a custom service role**, then choose the **customer_ssm_automation_role**. For the **Input** parameters:
        - InstanceId: {{TARGET_ID}}

- StartInactiveInstance: True to start the instances if they are stopped

9. Choose **Register Automation task**.

The patching maintenance window is created. Under the **Description** tab, you can see the **Next execution time**.

Default patch cycle

AMS Accelerate provides you with a default patch cycle when you onboard to patching. A default resource state manager is deployed into your account with the name **AmsDefaultPatchCycle**. It targets all the instances in your account with the tag**AmsDefaultPatchKey** and the value **True**.

You can update the cron on this state manager to enable the default patch cycle. To do so, follow these steps:

1. In the SSM console, go to **State Manager** under **Instances and Nodes** on the left pane.
2. Choose **AmsDefaultPatchCycle**, then **Edit** on the top right, and then scroll down to **Specify schedule** and update the **cron**. Choose **Save Changes**.

   Note
   Choosing **Save Changes** immediately triggers patching on the targeted instances. To enable the patch to run only during the cron schedule, and not immediately, select the checkbox **Apply association only at the next specified cron interval** under **Specify schedule** and then choose **Save Changes**.
3. You can also update the default patch tag key value for this default patch cycle:

   - In the SSM console, on the left navigation, under **Instances and Nodes**, go to **State Manager**.
   - Choose the **AmsDefaultPatchCycle**, then **Edit** on top right. Scroll down to **Targets** and update the tag key value, then choose **Save Changes**

Important
Do not delete the default patch cycle state manager.

AMS Accelerate patch baseline

A patch baseline defines which patches are approved for installation on your instances. You can specify approved or rejected patches one by one. You can also create auto-approval rules to specify that certain types of updates (for example, critical updates) should be automatically approved. The rejected list overrides both the rules and the approve list.

Default patch baseline

When you onboard to AMS Accelerate patching, the default patch baselines are overridden by the AMS Accelerate default patch baselines for the following operating systems:

- **Windows**
- **Amazon Linux 1**
- **Amazon Linux 2**
- **CentOS**
- **Suse**
- **Rhel**

The AMS Accelerate patch baselines defined as **product = *** mean that all patches are applied to the instance of all security and classifications.

Custom patch baseline

You can create a custom patch baseline by going to Patch Manager and, under **Instances & Nodes**, choosing **PatchManager**.

To learn more:

- See Creating a custom patch baseline (Windows)
- See Creating a custom patch baseline (Linux)
- See Updating or deleting a custom patch baseline (console)

Patching recommendations

If you are involved in application or infrastructure operations, you understand the importance of an operating system (OS) patching solution that is flexible and scalable enough to meet the varied requirements from your application teams. In a typical organization, some application teams use an architecture that involves immutable instances whereas others deploy their applications on non-immutable instances.

For more information on AWS Prescriptive Guidance for patching, see Automated patching for non-immutable instances in the hybrid cloud using AWS Systems Manager.

Patch responsibility recommendations

The patching process for persistent instances should involve the following teams and actions:

- **The application (DevOps) teams** define the patch groups for their servers based on application environment, OS type, or other criteria. They also define the maintenance windows specific to each patch group. This information should be stored on tags attached to the instances. Recommended tag names are 'Patch Group' and 'Maintenance Window'. During each patch cycle, the application teams prepare for patching, test the application after patching, and troubleshoot any issues with their applications and OS during patching.
- **The security operations team** defines the patch baselines for various OS types that are used by the application teams, and make the patches available through Systems Manager Patch Manager.
- **The automated patching solution** runs on a regular basis and deploys the patches defined in the patch baselines, based on the user-defined patch groups and maintenance windows.
- **The governance and compliance teams** define patching guidelines and exception processes & mechanisms.

For more information, see Patching solution design for non-immutable EC2 instances.

Guidance for application teams

- Review and become familiar with creating and managing maintenance windows; see AWS Systems Manager Maintenance Windows and Create an SSM Maintenance window for patching to learn more. Understanding the general structure and use of maintenance windows helps you understand what information to provide if you are not the person creating them.
- For High Availability (HA) setups, plan to have one maintenance window per availability zone and per environment (Dev/Test/Prod). This will ensure continued availability during patching.

- Recommended Maintenance Window duration is 4 hours with a 1-hour cutoff, plus 1 additional hour per 50 instances
- Patch Dev and Test versions with enough time between each to allow you to identify any potential issues prior to Production patching.
- Automate common pre- and post-patching tasks via SSM automation and run them as maintenance window tasks. Note that for post-patching tasks you must ensure that there is sufficient time allotted, as tasks will not launch once the cutoff is reached.
- Become familiar with Patch Baselines and their features—particularly around auto-approval delays for patch severity types that can be used to ensure that only the patches that were applied in Dev/Test get applied in Production at a later date. See About patch baselines for details.

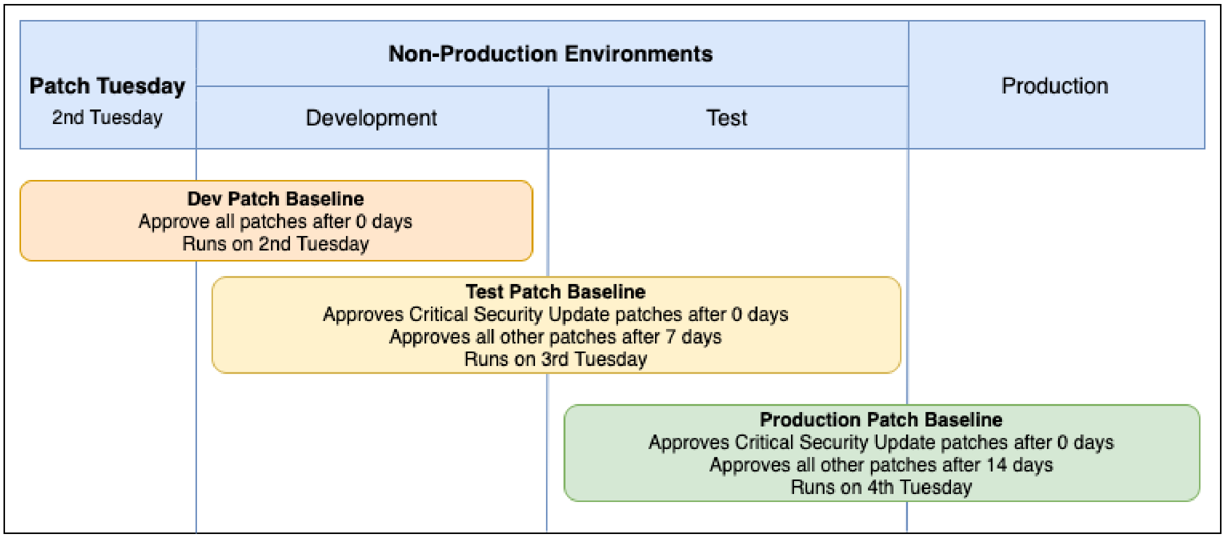
Guidance for security operations teams

- Review and become familiar with patch baselines. Patch approval is handled in an automated fashion and has different rule options. See About patch baselines for more information.
- Discuss needs around patching Dev/Test/Prod with application teams and develop multiple baselines to accommodate these needs.

Guidance for governance and compliance teams

- Patching should be an Opt Out function. A default maintenance window and automated tagging should exist to ensure nothing goes unpatched. AMS Resource Tagger can help with this—please discuss this option with your CA/CSDM for guidance on implementation.
- Requests for exemption from patching should require documentation justifying the exemption. A Cheif Information Security Officer (CISO) or other approval officer should approve or deny the request.
- Patching compliance should be reviewed on a regular schedule via the Patch Manager console, Security Hub, or a vulnerability scanner.

Example design for high availability Windows application


Patch Tuesday
2nd Tuesday
Non-Production Environments
Development
Test
Production
Dev Patch Baseline
Approve all patches after 0 days
Runs on 2nd Tuesday
Test Patch Baseline
Approves Critical Security Update patches after 0 days
Approves all other patches after 7 days
Runs on 3rd Tuesday
Production Patch Baseline
Approves Critical Security Update patches after 0 days
Approves all other patches after 14 days
Runs on 4th Tuesday

Overview:

- One Maintenance Window per AZ.
- One Set of Maintenance Windows per Environment.
- One Patch Baseline per Environment:
  - Dev: Approve all severity and classification after 0 days.
  - Test: Approve critical security update patches after 0 days and all other severity and classifications after 7 days.
  - Prod: Approve critical security update patches after 0 days and all other severity and classifications after 14 days.

CloudFormation Scripts:

These scripts are setup to build out the maintenance windows, baselines, and patching tasks for a two availability zone Windows HA EC2 application using the baseline approval settings described above.

- Windows Dev CFN Stack Example: HA-Patching-Dev-Stack.json
- Windows Test CFN Stack Example: HA-Patching-Test-Stack.json
- Windows Prod CFN Stack Example: HA-Patching-Prod-Stack.json

Patch recommendations FAQs

Q: How do I handle unscheduled patching for "0" day exploits?

A: SSM supports a **Patch Now** feature that uses the current default baseline for the instance's OS. AMS deploys a default set of Patch Baselines that approves all patches after 0 days. However, when using the **Patch Now** feature, a pre-patch snapshot is not taken, as this command runs the AWS-RunPatchBaseline SSM document. We recommend that you take a manual backup prior to patching.

Q: Are there any limitations for Maintenance Windows to keep in mind?

A: Yes, there are a few limitations you should be aware of.

- Maintenance Windows per Account: 50
- Tasks per Maintenance Window: 20
- Maximum number of concurrent automations per Maintenance Window: 20
- Maximum number of concurrent Maintenance Windows: 5

For a full list of default SSM limits, see AWS Systems Manager endpoints and quotas.

Patch monitoring and failure remediation

AMS Accelerate Patch add-on monitors patching and remediates failures.

Patch notification

AMS Accelerate sends notifications for the patching maintenance windows configured, see Create an SSM maintenance window for patching (p. 149). The system sends CloudWatch Events as advance notice of each upcoming maintenance window and a CloudWatch Event at the end of each maintenance window. The advance notices are sent four days and one hour before the maintenance window. The advance notice CloudWatch Events are sent with the following schema:

```
{
    "version": "0",
    "id": "37004d81-458d-2cef-fe1c-8afa8af30406",
    "detail-type": "AMS Patch Window Execution State Change",
    "source": "aws.managedservices",
    "account": "145917996532",
    "time": "2021-05-20T02:00:00Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:ssm:us-east-1:123456789012:maintenancewindow/mw-00000001235",
        "arn:aws:ec2:us-east-1:123456789012:instance/i-0000000aaaaaaaaaa",
        "arn:aws:ec2:us-east-1:123456789012:instance/i-0000000aaaaaaaaab"
    ],
    "detail": {
        "State": "PREEMPTIVE",
        "StartTime": "2021-05-24T02:00:00.000000",
        "WindowArn": "arn:aws:ssm:us-east-1:123456789012:maintenancewindow/mw-00000001235",
        "Results": "[{\"instanceId\": \"i-0000000aaaaaaaaaa\"}, {\"instanceId\":
 \"i-0000000aaaaaaaaab\"}]"
    }
}
```

This table describes the advance notice event schema.

Patch notification details

Property name	Description	Sample values
State	The state of the patching maintenance window	PREEMPTIVE – The patching window scheduled to begin soon
Status	The status of the patching maintenance window	SUCCESS – All instances were patch without failure FAILED – At least one instance has failed to patch
StartTime	The start time, in ISO format, of the patching maintenance window	2021-02-03T22:14:05.814308
WindowArn	The unique identifier of the patching maintenance window.	arn:aws:ssm:us-east-1: 123456789012:maintenancewindow/ mw-00000001235
Results	The list of instances that will be targeted by the patch window	InstanceId – the instance ID targeted

The CloudWatch Events can be used to trigger a CloudWatch rule to notify you whenever a patching maintenance window advance notice is sent. In this case, you would configure the CloudWatch rule with the following:

```
{"source": [
    "aws.managedservices"
  ],
  "detail-type: ["AMS Patch Window Execution State Change"],
  "detail": {
      "State": ["PREEMPTIVE"]
  }
 }
```

The window end CloudWatch Event is sent with the following schema:

```
{"version": "0",
    "id": "0f25add5-44a9-0702-d2bc-bd2102affefe",
    "detail-type": "AMS Patch Window Execution State Change",
    "source": "aws.managedservices",
    "account": "123456789012",
    "time": "2021-02-03T22:14:06Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:ssm:us-east-1:123456789012:maintenancewindow/mw-00000001235",
        "arn:aws:ec2:us-east-1:123456789012:instance/i-0000000aaaaaaaaaa",
        "arn:aws:ec2:us-east-1:123456789012:instance/i-0000000aaaaaaaaab"
    ],
    "detail": {"State": "[COMPLETED]",
        "Status": "SUCCESS",
        "StartTime": "2021-02-03T22:12:00.814308",
        "EndTime": "2021-02-03T22:14:05.814309",
        "WindowArn": "arn:aws:ssm:us-east-1:123456789012:maintenancewindow/mw-00000001235",
        "WindowExecutionId": "e32088eb-c05f-4c63-b766-6866e163c818",
        "Results": "[{\"instanceId\": \"i-0000000aaaaaaaaaa\", \"status\": \"Success\",
 \"missing_critical_patch_count\": 0, \"missing_total_patch_count\": 0} }, {\"instanceId
\": \"i-0000000aaaaaaaaab\", \"status\": Success}, \"missing_critical_patch_count\": 0,
 \"missing_total_patch_count\": 0}]"
    }
}
```

Patch window end details

Property name	Description	Sample values
State	The state of the patching maintenance window	COMPLETED – The patching window is finished
Status	The status of the patching maintenance window	SUCCESS – All instances were patch without failure FAILED – At least one instance has failed to patch
StartTime	The start time, in ISO format, of the patching maintenance window	2021-02-03T22:14:05.814308
EndTime	The end time, in ISO format, of the patching maintenance window	2021-02-03T23:14:05.814308
WindowArn	The unique identifier of the patching maintenance window.	arn:aws:ssm:us-east-1:123456789012:maintenancewindow/mw-00000001235
WindowExecutionId	The window execution ID, which can be seen from the SSM Maintenance Window Console	e32088eb-c05f-4c63-b766-6866e163c818

Property name	Description	Sample values
Results	The list of instances that will be targeted by the patch window	InstanceId – the instance ID targeted status – the instance patch status missing_critical_patch_count - the count of critical patches missing on the instance missing_total_patch_count - the count of total patches missing on the instance

Patch remediation

AWS Managed Services (AMS) manages patching and includes patch failure remediation. When patch fails, AMS Operations is alerted and they proceed to remediate by following AWS and AMS best practices to address the issue.

AMS creates an SSM OpsItem in the account with the following title: **AWS Managed Services – Patch Instance failure for instance <instance-id>**. AMS Operation will resolve the OpsItem if the situation can be corrected without your intervention. If they need your intervention, a Service Request is sent to the account owner to collaborate and address the issue. Upon remediation, the OpsItem is resolved.

If no action is taken, the patching maintenance window runs on the next cycle and attempts to patch the instance again.

Log management in AMS Accelerate

Topics

AMS Accelerate configures supported AWS services to collect logs. These logs are used by AMS Accelerate to ensure compliance and auditing of resources within your account.

Log management — AWS CloudTrail

AWS CloudTrail is a service that is used for account governance: compliance, operational auditing, and risk auditing. With CloudTrail, you can log, continuously monitor, and retain account activity related to actions across your AWS infrastructure.

AMS Accelerate relies on AWS CloudTrail logging in order to manage audits and compliance for all the resources created in the account. During onboarding, AMS Accelerate deploys a global CloudTrail trail in your primary AWS Region, which sends logs to Amazon S3.

The Amazon S3 bucket created for this trail uses the AWS Key Management Service (AWS KMS) encryption, and is accessed by the AMS Accelerate operations team for investigation and diagnosis purposes. If the account already has an existing CloudTrail trail enabled, this trail is in addition to that.

Also, AMS Accelerate deploys AWS Config rules to ensure that the CloudTrail is set up and encrypted in the state you want. To learn more, see AWS Config. These are the rules used, presented as links to the AWS documentation describing them:

- multi-region-cloudtrail-enabled. Checks that AMS Accelerate CloudTrail is properly set up with the correct configurations.
- cloud-trail-encryption-enabled. Checks that AWS CloudTrail is configured to use the server-side encryption (SSE) with AWS KMS customer master key (CMK) encryption.
- cloud-trail-log-file-validation-enabled. When enabled, checks that AWS CloudTrail creates a signed digest file with logs. We strongly recommend that you enable file validation on all trails.
- s3-bucket-default-lock-enabled. When enabled, checks that the Amazon S3 bucket has lock enabled.
- s3-bucket-logging-enabled. When enabled, checks whether logging is enabled for Amazon S3 buckets.

AMS Accelerate also relies on AWS KMS to encrypt the logged events. This key is controlled by, and is accessible to, the account administrators, AMS Accelerate operators, and CloudTrail. For more information about AWS KMS, see AWS Key Management Service features product documentation.

Accessing and auditing CloudTrail logs

CloudTrail logs are stored in an Amazon S3 bucket and also in a CloudWatch log group named **/aws/ams/cloudtrail**, within your account. All the events are encrypted using the AWS KMS key created at the same time as the CloudTrail resources.

Amazon S3 buckets leverage a naming pattern of **ams-a*aws account id*-cloudtrail-*AWS Region***, (example: **ams-a123456789-cloudtrail-us-east-1a**) and all the events are stored with the **AWS/**

CloudTrail prefix. All access to the primary bucket is logged and the log objects are encrypted and versioned for auditing purposes.

For more information about tracking changes and querying the logs, see Tracking changes in your AMS Accelerate accounts (p. 161).

Protecting and retaining CloudTrail logs

During account onboarding, AMS Accelerate enables Amazon S3 object locking with Governance Mode to ensure that users can't overwrite or delete an object version or alter its lock settings without special permissions. For more information, see Amazon S3 object locking.

By default, all logs in this bucket are kept indefinitely. If you want to change the retention period, you can submit a service request through the AWS Support Center to set up a different retention policy.

Accessing Amazon EC2 logs

You can access Amazon EC2 instance logs by using the AWS Management Console. Logs produced by instances and AWS services are available in CloudWatch Logs, which is available in each account managed by AMS Accelerate. For information about accessing your logs, see the CloudWatch Logs documentation.

Retaining Amazon EC2 logs

Amazon EC2 instance logs are kept indefinitely, by default. If you want to change the retention period, you can submit a service request through the AWS Support Center to set up a different retention policy.

Log management — Amazon EC2

AMS Accelerate installs the CloudWatch agent on all Amazon EC2 instances that you have identified as AMS Accelerate-managed. This agent sends system-level logs to Amazon CloudWatch Logs. For information, see What are Amazon CloudWatch Logs?

The following log files are sent to CloudWatch Logs, into a log group of the same name as the log. Within each log group, a log stream is created for each Amazon EC2 instance, named according to the Amazon EC2 instance ID.

Linux

- /var/log/amazon/ssm/amazon-ssm-agent.log
- /var/log/amazon/ssm/errors.log
- /var/log/audit/audit.log
- /var/log/cloud-init-output.log
- /var/log/cloud-init.log
- /var/log/cron
- /var/log/maillog
- /var/log/messages
- /var/log/secure
- /var/log/spooler
- /var/log/yum.log
- /var/log/zypper.log

For more information, see Manually Create or Edit the CloudWatch Agent Configuration File.

Windows

- C:\\ProgramData\\Amazon\\SSM\\Logs\\amazon-ssm-agent.log
- C:\\ProgramData\\Amazon\\SSM\\Logs\\amazon-cloudwatch-agent.log
- C:\\ProgramData\\Amazon\\SSM\\Logs\\errors.log
- C:\\cfn\\log\\cfn-init.log

For more information, see Quick Start: Enable Your Amazon EC2 Instances Running Windows Server 2016 to Send Logs to CloudWatch Logs Using the CloudWatch Logs Agent.

Tracking changes in your AMS Accelerate accounts

Topics

AWS Managed Services helps you track changes made by the AMS Accelerate Operations team and AMS Accelerate automation by providing a queryable interface using the Amazon Athena (Athena) console and AMS Accelerate log management.

Athena is an interactive query service you can use to analyze data in Amazon S3 by using standard Structured Query Language (SQL) (see SQL Reference for Amazon Athena). Athena is serverless, so there is no infrastructure to manage, and you pay only for the queries that you run. AMS Accelerate creates Athena tables with daily partitions over CloudTrail logs, and provides queries on your primary AWS Region and within the **ams-change-record** workgroup. You can choose any of the default queries and run them as needed. To learn more about Athena workgroups, see How Workgroups Work.

Using change record, you can easily answer questions like:

- Who (AMS Accelerate Systems or AMS Accelerate Operators) has accessed your account
- What changes have been made by AMS Accelerate in your account
- When did AMS Accelerate perform changes in your account
- Where to go to view changes made in your account
- Why AMS Accelerate needed to make the changes in your account
- How to modify queries to get answers to all those questions for any non-AMS changes too

Viewing your change records

To use Athena queries, sign in to the AWS Management console and navigate to the Athena console in your primary AWS Region.

> **Note**
> If you see the **Amazon Athena Get Started** page while performing any of the steps, click **Get Started**. This might appear for you even if your Change Record infrastructure is already in place.

1. Choose **Workgroup** from the upper navigation panel in the Athena console.
2. Choose the **ams-change-record** workgroup, and then click **Switch Workgroup**.
3. Choose **ams-change-record-database** from the **Database Combo** box. The **ams-change-record-database** includes the **ams-change-record-table** table.
4. Choose **Saved Queries** from the upper navigation panel.
5. The **Saved Queries** window shows a list of queries that AMS Accelerate provides, which you can run. Choose the query you want to run from the **Saved Queries** list. For example, **ams_session_accesses_v1 query**.

   For the full list of preset AMS Accelerate queries, see Default queries (p. 162).
6. Adjust the **datetime** filter in the query editor box as needed; by default, the query only checks changes from the last day.

7. Choose **Run query**.

Default queries

AMS Accelerate provides several default queries you can use within the Athena console; they are listed in the following table.

Note

- All queries accept **datetime range** as an optional filter; all the queries run over the last 24 hours, by default. For expected input, see the following subsection, Modifying the **datetime** filter in queries (p. 166).
- Parameter inputs that you can or need to change are shown in the query as *<PARAMETER_NAME>* with angular braces. Replace the placeholder **and** the angular braces with your parameter value.
- All filters are optional. In the queries, some optional filters are commented out with a double dash (--) at the start of the line. All queries will run without them, with default parameters. If you want to specify parameter values for these optional filters, remove the double dash (--) at the start of the line and replace the parameter as you want.
- All queries return `IAM PincipalId` and `IAM SessionId` in the outputs
- The calculated cost for running a query depends on how many CloudTrail logs are generated for the account. To calculate the cost, use the AWS Athena Pricing Calculator.

Canned queries

Purpose/Description	Inputs	Outputs
Query name: `ams_access_session_query_v1`		
Tracking AMS Accelerate access sessions Provides information about a specific AMS Accelerate access session. The query accepts the IAM Principal ID as an optional filter and returns event time, business need for accessing the account, requester, and so on. You can filter on a specific IAM Principal ID by uncommenting the line and replacing the placeholder *`IAM PrincipalId`* with a specific ID in the query editor. You can also list non-AMS access sessions by removing the **useragent** filter line in the WHERE clause of the query.	(Optional) `IAM PrincipalId`: The IAM Principal identifier of the resource that is trying to access. The format is *UNIQUE_IDENTIFIER*:*RESOURCE_NAME*. For details see unique identifiers. You can run the query without this filter to determine the exact IAM PrincipalId the you want to filter with.	• EventTime: Time of gaining the access • EventName: AWS Event name (AssumeRole) • EventRegion: AWS Region that gets the request • EventId: CloudTrail Event ID • BusinessNeed Type: Business reason type to access the account. Allowed values are: SupportCase, OpsItem, Issue, Text. • BusinessNeed: Business need to access the account. For example, Support Case ID, Ops Item ID, and so forth. • Requester: Operator ID that accesses the account, or Automation system that access the account. • RequestAccessType: Requester type (System, OpsConsole, OpsAPI, Unset)

Purpose/Description	Inputs	Outputs
Query name: `ams_events_query_v1`		
Track all mutating actions done by AMS Accelerate Returns all write actions done on the account using that AMS Accelerate role filter. You can also track mutating actions done by non-AMS roles by removing the **useridentity.arn** filter lines from the WHERE clause of the query.	(Optional) Only **datetime range**. See Modifying the **datetime** filter in queries (p. 166).	• AccountId: AWS Account ID • RoleArn: RoleArn for the requester • EventTime: Time of gaining the access • EventName: AWS Event name (AssumeRole) • EventRegion: AWS Region that gets the request • EventId: CloudTrail Event ID • RequestParameters : Request parameters for the request • ResponseElements: Response elements for the response. • UserAgent: AWS CloudTrail User Agent
Query name: `ams_instance_access_sessions_query_v1`		
Track instance accesses by AMS Accelerate Returns a list of AMS Accelerate instance accesses; every record includes event time, event Region, instance ID, IAM Principal ID, IAM Session ID, SSM Session ID. You can use the IAM Principal ID to get more details on the business need for accessing the instance by using the `ams_access_sessions_query_v1` Athena query. You can use the SSM Session ID to get more details on the instance access session, including the start and end time of the session, log details, and using the AWS Session Manager console in the instance's AWS Region. Users can also list non-AMS instance accesses by removing the **useridentity** filter line in the WHERE clause of the query.	Only `datetime range`. See Modifying the **datetime** filter in queries (p. 166).	• InstanceId: Instance ID • SSMSession Id: SSM Session ID • RoleArn: RoleArn for the requester • EventTime: Time of gaining the access • EventName: AWS Event name (AssumeRole) • EventRegion: AWS Region that gets the request • EventId: CloudTrail Event ID
Query name: `ams_privilege_escalation_events_query_v1`		

Purpose/Description	Inputs	Outputs
Track permission (escalation) events for AMS and non-AMS users Provides a list of events that can directly or potentially lead to a privilege escalation. The query accepts ActionedBy as an optional filter and returns EventName, EventId, EventTime, and so forth. All fields associated with the event are also returned. Fields are blank if not applicable for that event. The ActionedBy filter is disabled, by default; to enable it, remove "-- " from that line. By default, the ActionedBy filter is disabled (it will show privilege escalation events from all users). To show events for a particular user or role, remove the double dash (--) from the **useridentity** filter line in the WHERE clause and replace the placeholder *`ACTIONEDBY_PUT_USER_NAME_HERE`* with an IAM user or role name. You can run the query without the filter to determine the exact user you want to filter with.	(Optional) `ACTIONEDBY_PUT_USER_NAME`: Username for the actionedBy user. This can be an IAM user or role. For example, ams-access-admin. (Optional) `datetime range`. See Modifying the **datetime** filter in queries (p. 166).	• AccountId: Account Id • ActionedBy: ActionedBy Username • EventTime: Time of gaining the access • EventName: AWS Event name (AssumeRole). • EventRegion: AWS Region that gets the request • EventId: CloudTrail Event ID
Query name: `ams_resource_events_query_v1`		

Purpose/Description	Inputs	Outputs
Track write events for specific resources AMS or non-AMS Provides a list of events done on a specific resource. The query accepts resource ID as part of the filters (replace placeholder *`RESOURCE_INFO`* in the WHERE clause of the query), and returns all write actions done on that resource.	(Required) `RESOURCE_INFO`: The resource identifier, can be an ID for any AWS resource in the account. Do not confuse this with resource ARNs. For example, an instance ID for an EC2 instance, table name for a DynamoDB table, logGroupName for a CloudWatch Log, etc. (Optional) `datetime range`. See Modifying the **datetime** filter in queries (p. 166).	- AccountId: Account Id - ActionedBy: ActionedBy Username - EventTime: Time of gaining the access - EventName: AWS Event name (AssumeRole). - EventRegion: AWS Region that gets the request - EventId: CloudTrail Event ID
Query name: `ams_session_events_query_v1`		
Track write actions performed by AMS Accelerate during specific session Provides a list of events done on a specific session. The query accepts IAM Principal ID as part of the filters (replace the placeholder *`PRINCIPAL_ID`* in the WHERE clause of the query), and returns all write actions done on that resource.	(Required) `PRINCIPAL_ID`: Principal ID for the session. The format is *`UNIQUE_IDENTIFIER`*:*`RESOURCE_NAME`*. For details see unique identifiers. You can run the query "ams_session_ids_by_requester_v1" to get list of IAM Principal IDs for a requester. You can also run the query without this filter to determine the exact IAM PrincipalId you want to filter with. (Optional) `datetime range`. See Modifying the **datetime** filter in queries (p. 166).	- AccountId: Account Id - ActionedBy: ActionedBy Username - EventTime: Time of gaining the access - EventName: AWS Event name (AssumeRole) - EventRegion: AWS Region that gets the request - EventId: CloudTrail Event ID
Query name: `ams_session_ids_by_requester_v1`		
Track IAM Principal/Session IDs for a specific requester. The query accepts "requester" (replace the placeholder *`Requester`* in the WHERE clause of the query), and returns all IAM Principal Ids by that requester during the specified time range.	(Required) `Requester`: Operator ID that accesses the account (for example: alias of an operator), or Automation system that access the account (for example: OsConfiguration, AlarmManager, etc.). (Optional) `datetime range`. See Modifying the **datetime** filter in queries (p. 166).	- IAM PrincipalId - IAM Principal Id of the session. The format is *`UNIQUE_IDENTIFIER`*:*`RESOURCE_NAME`*. For details see unique identifiers. You can run the query without this filter to determine the exact IAM PrincipalId you want to filter with. - IAM SessionId - IAM Session Id for the access session - EventTime: Time of gaining the access

Modifying the **datetime** filter in queries

All queries accept **datetime** range as an optional filter. All the queries run over the last one day by default.

The format used for the **datetime** field is yyyy/MM/dd (for example: 2021/01/01). Remember that it only stores the date and not the entire timestamp. For the entire timestamp, use the field **eventime**, which stores the timestamp in the ISO 8601 format yyyy-MM-dd**T**HH:mm:ss**Z** (for example: 2021-01-01T23:59:59Z). However, since the table is partitioned on the datetime field, you'll need to pass in both the datetime and eventtime filter to the query. See the following examples.

> **Note**
> To see all the accepted ways you can modify the range, see the latest Presto function documentation based on the Athena engine version currently used for the **Date and Time Functions and Operators** to see all the accepted ways you can modify the range.

Date Level: Last 1 day or last 24 hours (Default) example: If the CURRENT_DATE='2021/01/01' , the filter will subtract one day from the current date and format it as datetime > '2020/12/31'

```
datetime > date_format(date_add('day', - 1, CURRENT_DATE), '%Y/%m/%d')
```

Date Level: Last 2 months example:

```
datetime > date_format(date_add('month', - 2, CURRENT_DATE), '%Y/%m/%d')
```

Date Level: Between 2 dates example:

```
datetime > '2021/01/01'
      AND
      datetime < '2021/01/10'
```

Timestamp Level: Last 12 hours example:

Partition data scanned to last 1 day and then filter all events within the last 12 hours

```
datetime > date_format(date_add('day', - 1, CURRENT_DATE), '%Y/%m/%d')
      AND
      eventtime > date_format(date_add('hour', - 12, CURRENT_TIMESTAMP), '%Y-%m-%dT%H:%i:
%sZ')
```

Timestamp Level: Between 2 timestamps example:

Get events between Jan 1, 2021 12:00PM and Jan 10, 2021 3:00PM.

```
datetime > '2021/01/01' AND datetime < '2021/01/10'
      AND
      eventtime > '2021-01-01T12:00:00Z' AND eventtime < '2021-01-10T15:00:00Z'
```

Change record permissions

The following permissions are needed to run change record queries:

- **Athena**
  - athena:GetWorkGroup

  - athena:StartQueryExecution
  - athena:ListDataCatalogs
  - athena:GetQueryExecution
  - athena:GetQueryResults
  - athena:BatchGetNamedQuery
  - athena:ListWorkGroups
  - athena:UpdateWorkGroup
  - athena:GetNamedQuery
  - athena:ListQueryExecutions
  - athena:ListNamedQueries
- **AWS KMS**
  - kms:Encrypt
  - kms:Decrypt
  - Resource: Key ID of AMSCloudTrailLogManagement
- **AWS Glue**
  - glue:GetDatabase
  - glue:GetTables
  - glue:GetDatabases
  - glue:GetTable
- Amazon S3 read access
  - Resource: ams-a*AccountId*-cloudtrail-*primary region*
- **Amazon S3 write access**
  - Resource: workgroup Athena results bucket

AWS Systems Manager in AMS Accelerate

Topics

An AWS Systems Manager document (SSM document) defines the actions that Systems Manager performs on your AWS resources. Systems Manager includes more than a dozen pre-configured documents that you can use by specifying parameters at runtime. Documents use JavaScript Object Notation (JSON) or YAML, and they include steps and parameters that you specify.

AWS Managed Services (AMS) is a trusted publisher for SSM documents. SSM documents owned by AMS are shared only with onboarded AMS accounts, always begin with a reserved prefix (AWSManagedServices-*), and show up in the Systems Manager console, as owned by Amazon. The AMS process for SSM document development and publishing follows AWS best practices and requires multiple peer reviews throughout the document life cycle. For more information on AWS best practices for sharing SSM Documents, please visit Best practices for shared SSM documents.

Available AMS Accelerate SSM documents

AMS Accelerate SSM documents are available exclusively to AMS Accelerate customers, and are used to automate operational workflow to operate your account.

To see the available AMS Accelerate SSM documents from the AWS Management Console:

1. Open the Systems Managerconsole at AWS Systems Manager console.
2. Choose **Shared with me**.
3. In the search bar, filter by **Document name prefix**, then **Equals**, and set the value to **AWSManagedServices-**.

   For AWS CLI instructions, see Using shared SSM documents.

AMS Accelerate SSM document versions

SSM documents support versioning. AMS Accelerate SSM documents can't be modified from the customer's account and can't be re-shared. They're centrally managed and maintained by AMS Accelerate in order to operate the account.

Version numbers are incremented with each document update in a specific AWS Region. As new Regions become available, the same document content in two Regions can have different version numbers; this is typical and doesn't mean their behavior will be different. If you want to compare two AMS Accelerate SSM documents, we recommend comparing their hashes with the AWS CLI:

```
    aws ssm describe-document \
    --name AWSManagedServices-DOCUMENTNAME \
    --output text --query "Document.Hash"
```

Two SSM documents are identical if their hashes match.

Systems Manager pricing

There is no cost associated with AMS Accelerate SSM document access. Runtime cost varies based on the type of SSM document, its steps, and runtime duration. For more information, refer to AWS Systems Manager pricing.

Document history

The following table describes the important changes to the documentation since the last release of the AMS Accelerate guide.

- **Latest documentation update:** September 30, 2021

Change	Description	Date
Onboarding Process	Shorter description of Onboarding phases. See Getting Started with AWS Managed Services (p. 15).	September 30, 2021
Config Rules and Config Reports	Added or updated descriptions of AWS Config Rules and Config Reports to facilitate compliance with industry standards. See Infrastructure security (p. 99) and Compliance and conformance (p. 106).	September 30, 2021
Resource Tagger	Updated JSON samples (bug fixes), and added examples for using ReadOnly to prevent tagger from modifying resources. See Configuration profile document format (p. 39).	September 30, 2021
Maintenance Windows	Updated instructions for creating and updating Maintenance Windows from the AWS console. See Create an SSM maintenance window for patching (p. 149).	September 30, 2021
Updated table of monitoring alarms	Added Site-to-Ste VPN to list of available monitoring alarms. See Supported services (p. 8).	September 30, 2021
New feature! Operations on Demand	Operations on Demand provides you with a curated catalog of operations activities. See AWS Managed Services Operations On Demand (p. 60).	September 16, 2021
Config Aggregator	As a part of the AMS Accelerate setup, the AMS team configures a config aggregator that aggregates the config compliance information in your account to an AMS-owned account where it is compiled into a report for you. See Infrastructure security (p. 99).	September 16, 2021
Patch Scheduling	When following step two in the procedure to set up a patching schedule, a note was added	September 16, 2021

Change	Description	Date
	warning about initiating an immediate patch versus following the set schedule. See Default patch cycle (p. 151).	
New monitoring alarm	Alert name and trigger condition: VPNTunnelDown, TunnelState > 0 for 1 min, 20 consecutive times. TunnelState is 0 when both tunnels are down, .5 when one tunnel is up and 1.0 when both tunnels are up. See Alerts from baseline monitoring in AMS (p. 122).	September 16, 2021
Bad JSON, backup	An example JSON block was missing a closing curly bracket, that was fixed. See Associating resources to the backup plan using Resource Tagger (p. 146).	September 16, 2021
Bad JSON, patching	An example JSON block was not formatted correctly, that was fixed. See Patch monitoring and failure remediation (p. 154).	September 16, 2021
Bad JSON	An example JSON block was not formatted correctly, that was fixed. See Patch monitoring and failure remediation (p. 154).	September 16, 2021
Root account	New section on using the root credentials. See How and when to use the root user account (p. 97).	August 26, 2021
Change record	New and updated information on using the AMS Accelerate change tracking tool. See Tracking changes in your AMS Accelerate accounts (p. 161).	August 26, 2021
Compliance and conformance	Removed the ams-nist-cis-s3-bucket-policy-grantee-check Config rule. See Compliance and conformance (p. 106).	August 26, 2021
Getting started	Created a prerequisites section with the list of external dependencies for network configuration. See Accelerate onboarding prerequisites (p. 15).	August 12, 2021
Backup	Clarified wording for adding backup to an account. See Adding AWS Backup to your AMS Accelerate accounts (p. 144).	August 12, 2021

Change	Description	Date
Account discovery tool	The Changelog zip has been updated. See Account discovery (p. 16).	August 12, 2021
Service description	Windows 2008 R2 is removed from the list of supported OSes. See Supported configurations (p. 7).	August 12, 2021
Service description	New RDS alert remediation feature. See AMS automatic remediation of alerts (p. 140).	July 29, 2021
Backup	Details on default backup plans. See Backup management in AMS Accelerate (p. 143).	July 29, 2021
Service description	Added NAT Gateway to the list of service supported by monitoring. See Service description (p. 6).	July 15, 2021
Support experience	Added missing AWS Support service codes • Service requests: service-ams-operations-service-request and for • Incidents: service-ams-operations-report-incident See What is incident management? (p. 52).	July 15, 2021
Backups	Clarified function of backup vaults. See Backup management in AMS Accelerate (p. 143).	July 15, 2021
Account dicovery	Clarified CloudSearch example: Don't change the "--domain-owner 354220221581" and " --region us-west-2", copy as-is. Also, added access to the account discovery CLI CHANGELOG. See Account discovery (p. 16).	July 15, 2021
June 2021		
New feature: Self-service reporting	AMS now offers self-service reporting through a new **Reporting** page in the AMS Console. See Self-service reporting (p. 74).	June 17, 2021
Patching: Added patching recommendations and additional failure and remediation information to the Patching section.	See Patching recommendations (p. 152) and Patch monitoring and failure remediation (p. 154).	June 17, 2021

Change	Description	Date
Account discovery: Added an IAM policy to the Account Discovery section and updated the AWS account discovery with AWS CloudShell section with information about using AWS Regions. Also added notes recommending using the AWS CloudShell method for account discovery to all other methods.	See Account discovery (p. 16).	June 17, 2021
Updates to Patch remediation in Accelerate: Support SSM maintenance window for patching with instance target, advance notice patch notification information, added missing patch counts in end patch notification.	See Patch monitoring and failure remediation (p. 154)	June 17, 2021
Updated the Alerts from baseline monitoring section with information on how to set up direct notifications.	See Alerts from baseline monitoring in AMS (p. 122).	June 17, 2021
Added information abouit our SSM document policy.	AWS Systems Manager in AMS Accelerate (p. 168).	June 17, 2021
Updated the AMS responsibility matrix (RACI).	See Roles and responsibilities (p. 8)	June 17, 2021
Updated the AWS Config reporting table.	See Reporting in AMS (p. 65).	June 17, 2021
Renamed Remediation Category: Auto Remediation to Remediation Category: Automatic Remediation nad removed duplicate rule (ams-nist-cis-cloudtrail-enabled).	See Compliance and conformance (p. 106).	June 17, 2021
Updated Macie links from 'findings' page to 'alerts' page; removed 'Average SwapUsage' from Alerts from baseline monitoring table.	See Alerts from baseline monitoring in AMS (p. 122).	June 17, 2021
Added GuardDuty suppression rules.	See Data protection (p. 100).	June 17, 2021
Removed the SALZ diagram from the How monitoring works page.	See How monitoring works (p. 121).	May 13, 2021

Change	Description	Date
Updated note to explain that the Instance Configuration Service is triggered through events and those events, as of now, are Instance tag events and Instance start events.	See Automated instance configuration setup (p. 49).	May 13, 2021
Several minor updates, resulting from a full document review, to improve clarity and accuracy.	See What is AWS Managed Services? (p. 1).	May 13, 2021
Improved accuracy in the Roles and responsibilities table by replacing relevant instances of "detective controls" with "Config Rules".	See Roles and responsibilities (p. 8).	May 13, 2021
Added a new section: Granting permissions for AMS Accelerate administration. This new section provides a CloudFormation template that contains policies related to customer access within your Accelerate account.	See Granting permissions for AMS Accelerate administration (p. 86).	May 13, 2021
Updated the "Remediation Category: Auto Remediation" in the "AMS Accelerate AWS Config Rules Inventory" table to include the three rules ams-nist-cis-cloudtrail-enabled, ams-nist-cis-guardduty-enabled-centralized, and ams-nist-cis-vpc-flow-logs-enabled	See Compliance and conformance (p. 106).	May 13, 2021
Updated the description of how to use 'AwsAccountDiscoveryCli' for discovery on the Account discovery page.	See Account discovery (p. 16).	May 13, 2021
Updated the Backup management section to remove outdated wording pertaining to existing backup plans and clarified language to distinguish between backup vaults and Accelerate accounts; formatted code blocks.	See Backup management in AMS Accelerate (p. 143).	May 13, 2021
Clarified and elaborated on how AMS processes incidents and included a flowchart for added visual clarity.	See What is incident management? (p. 52).	May 13, 2021

Change	Description	Date
Many updates and new content for Monitoring tags and Tagging.	See Monitoring tags (p. 34), Resource Tagger (p. 37), and Tag-based Alarm Manager (p. 128).	April 15, 2021
Added a link to monitoring information in the Supported Services section.	See Supported services (p. 8).	April 15, 2021
Updated the AMS Accelerate service description page.	See Service description (p. 6).	March 19, 2021
The Service Description was updated and the 'Getting Started' section of the 'What is Accelerate?' chapter was made into a separate chapter.	See Getting Started with AWS Managed Services (p. 15).	March 19, 2021
This is the first release of this document.	See What is AWS Managed Services? (p. 1).	March 16, 2021

AWS glossary

For the latest AWS terminology, see the AWS glossary in the *AWS General Reference*.