ISO 31000:2018 Risk Management on AWS

Compliance Guide

April 29, 2026

Notices

Customers are responsible for making their own independent assessment of the information in this document. This document: (a) is for informational purposes only, (b) represents current AWS product offerings and practices, which are subject to change without notice, and (c) does not create any commitments or assurances from AWS and its affiliates, suppliers or licensors. AWS products or services are provided “as is” without warranties, representations, or conditions of any kind, whether express or implied. The responsibilities and liabilities of AWS to its customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.

The information within this guide is presented as informational and is for reference only. Customers must manage their own ISO 31000 risk management program implementation. Customers may use AWS services to support their risk identification, assessment, treatment, and monitoring activities, AWS does not directly manage customer risk management programs or frameworks. This compliance guide is provided to support customers' consideration and review of their risk management scope and implementation of controls when using AWS services as they develop or mature their ISO 31000 aligned risk management practices.

This guide is provided by AWS Security Assurance Services, LLC (AWS SAS), a wholly owned subsidiary of AWS. AWS SAS, a Payment Card Industry-Qualified Security Assessor company (PCI-QSAC) and HITRUST External Assessor Firm, is a team of industry certified assessors, helping customers to achieve, maintain, and automate compliance in the cloud by connecting audit standards to AWS service-specific features and functionality.

Contents

Introduction

This guide shows you how to identify, assess, and treat security and compliance risks in AWS environments using ISO 31000:2018 principles with native AWS services that automate risk detection, enforce controls, and maintain continuous compliance. You'll implement risk management across four phases, from establishing core security architecture to achieving continuous, automated monitoring, while understanding your responsibilities under the AWS Shared Responsibility Model.

ISO 31000 risk management overview

ISO 31000:2018 outlines how to identify, analyze, evaluate, treat, and monitor risks without mandating specific controls. You will choose which controls fit your environment.

Periodic assessments miss changes between review cycles. Continuous monitoring detects configuration changes and threats in real time, reducing response time from weeks to minutes. You can adapt to operational changes and include input from stakeholders across your organization.

Context and criteria

Define scope, context, and criteria to align risk management with your objectives. External factors like market shifts and regulations shape risk exposure. Internal factors like culture and resources determine risk management capacity.

Scope defines boundaries. Context includes environmental factors like regulatory requirements and organizational culture. Criteria establish thresholds for measuring risk likelihood, impact, and acceptability.

External context

External context includes the regulatory environment that influences your risk management approach. You must navigate compliance requirements across your industry and jurisdictions. Requirements vary by region and sector, and regulatory changes can affect your risk criteria, thresholds, and control measures. Build regulatory monitoring into your operations: track changes, assess impacts, and adjust your risk management approach accordingly. Subscribe to

regulatory update services for your industry (such as Federal Register notifications for US healthcare, or FCA policy statements for UK financial services). Assign responsibility for quarterly regulatory change reviews and document impact assessments.

Internal context

Internal context includes your governance structure, roles, policies, and processes. Culture determines whether teams report security findings immediately or wait for scheduled reviews. It affects whether developers view security controls as obstacles or safeguards. Resources like budget, people, and technology determine what's possible when a risk materializes. Risk treatments must account for available budget, staff expertise, and technology constraints. A treatment requiring 24/7 security operations center staffing won't work for organizations without those resources.

Defining your risk criteria

Risk criteria define how you measure and prioritize risks. Customize these three elements for your organization:

- **Impact categories**: Define the specific consequences that matter most to you such as financial losses, operational disruptions, reputational damage, regulatory noncompliance, or customer impact.
- **Likelihood assessment**: Determine the probability of risk events occurring, incorporating historical data analysis, threat intelligence, vulnerability assessments, and expert judgment.
- **Risk appetite thresholds**: Establish acceptable risk levels for pursuing your objectives. These thresholds typically vary across different risk categories, business units, and criticality levels, reflecting your strategic priorities and regulatory constraints.

Conducting risk assessments

Risk assessment includes three components: identification, analysis, and evaluation. Conduct assessments with stakeholder input and current threat intelligence, repeating as your environment changes.

Risk identification

Risk identification catalogs threats, vulnerabilities, and potential impacts that could prevent you from achieving objectives.

Use structured threat modeling approaches:

- **STRIDE methodology:** Identifies threats related to Spoofing, Tampering, Repudiation, Information disclosure, Denial of service, and Elevation of privilege
- **Attack trees**: Models possible attack paths through you AWS workloads to understand potential exploit scenarios
- **Data flow analysis**: Examines data movement through your AWS workloads to uncover security gaps and vulnerabilities

Analyzing risks

Risk analysis assesses likelihood, potential impact, and existing control effectiveness through two common approaches.

Quantitative risk analysis

Quantitative analysis uses numerical and statistical methods to assess risk probability and impact. It offers more precision but requires sufficient reliable data.

- **Security metrics collection:** Gather quantifiable data about security standing
- **Vulnerability scoring**: Assess the severity of identified vulnerabilities
- **Exposure analysis**: Determine the potential impact of identified risks

Qualitative risk analysis

Use qualitative approaches for risks that are difficult to quantify. Qualitative analysis is faster than quantitative methods but provides less precision.

- **Scenario analysis**: Evaluate potential risk scenarios and their outcomes
- **Control effectiveness assessment**: Evaluate how well existing controls mitigate identified risks
- **Expert judgment**: Use specialized knowledge to assess complex risks

Evaluating risks

Risk evaluation compares your analysis results against the criteria you've defined. Treat risks that exceed your tolerance thresholds; Risks within tolerance may need only monitoring.

Risk evaluation requires structured approaches to prioritization including:

- **Risk matrices**: Visualize risks based on likelihood and impact
- **Risk appetite alignment**: Compare analyzed risks against defined risk tolerance
- **Cost-benefit analysis**: Evaluate the economics of risk treatment options

Risk treatment

Cloud environments introduce distinct characteristics that influence risk treatment approaches like cloud-focused controls, automation, and region-specific compliance requirements. Cloud services can remediate non-compliance within minutes, compared to days or weeks for manual processes. In Amazon Web Services (AWS) environments, risk treatment relies on integrated services and automation to monitor, enforce, and optimize controls. Automated controls enforce security baselines without slowing operations.

Figure 1: Risk treatment components

The components of risk treatment—shown in figure 1—include:

- **Risk avoidance:** The decision to reduce exposure to a risk by not undertaking activities that trigger it—such as avoiding certain technologies or services that pose unacceptable challenges. If you have data residency requirements for a particular workload, you can deploy that workload in an AWS Region that meets those requirements. Not all AWS Regions offer the same services, so the choice of Region should align with both data residency obligations and the availability of required services.
- **Risk mitigation:** Steps to decrease the probability or impact of a risk. Does not eliminate all risks, but manages them to acceptable levels. AWS provides tools and services to support mitigation efforts, including identity and access management, encryption, network segmentation, logging and monitoring, automated compliance verification, and real-time responses to compliance violations. These controls minimize exposure to security, operational, and compliance risk while maintaining the scalability cloud environments provide.
- **Risk transfer:** Moving responsibility for certain risks to an external entity. In AWS, this is reflected in the AWS Shared Responsibility Model. AWS secures the underlying cloud infrastructure (security of the cloud), while customers secure their own data, applications, and workloads deployed in the cloud (security in the cloud). When risks are transferred, you can focus your efforts on the risks you control while relying on others for their areas of responsibility.
- **Risk acceptance**: Not every risk can be mitigated cost-effectively. Risk treatment decisions balance multiple factors: business impact, technical feasibility, likelihood, consequences, and mitigation cost. Not every risk justifies expensive controls. In these cases, you may decide to formally accept the residual risk, if it's within acceptable thresholds. For instance, accepting the risk of downtime for non-critical development environments as opposed to spending money on multi-Region replication.

Document each identified risk and its treatment in your risk register, including owner, deadline, and success criteria. ISO 31000 requires this accountability to verify treatments are feasible and tracked to completion.

Risk monitoring and review

Monitoring and review verify that treatments work as designed, detect control failures, identify changes affecting risk profiles, and capture lessons for improvement.

Implement three monitoring layers:

- **Operational monitoring:** Monitor your AWS resources using Amazon CloudWatch, AWS CloudTrail, and AWS Config daily to gauge performance and security. It includes tracking key risk indicators such as uptime, speed, resource usage, and access patterns, while also being alert for unusual behavior, misconfigurations, or unauthorized activities. Operational monitoring helps teams identify issues early, respond to incidents quickly, and maintain compliance with policies and regulations. AWS tools such as Amazon CloudWatch, AWS CloudTrail, and AWS Config offer real-time insights into changes, events, and alerts so you can quickly detect any deviations from established baselines.
- **Control effectiveness reviews:** Regular assessments verify that security and compliance controls function correctly. Test both technical measures like access permissions, encryption, and monitoring, and process-driven safeguards such as incident response and audit procedures. For instance, confirming that encryption is enforced or that policies are correctly applied.
- **Strategic reviews:** Evaluations that help ensure your risk management strategy aligns with changing business objectives, regulatory demands, and the adoption of new technologies. Strategic reviews examine how emerging threats, new regulations, or changes in business operations affect the risk environment. Reviews equip leadership to make decisions, prioritize risk mitigation efforts, and adjust governance, policies, and controls to foster both growth and compliance. For instance, when introducing AI and machine learning (AI/ML) workloads, new concerns about privacy or model integrity can arise, necessitating updated treatment plans.

The ISO 31000 framework highlights the importance of feedback loops, which work well AWS environments. Use insights from incidents, audits, and configuration changes to revise risk assessments and treatment strategies.

Recording and reporting risk activities

ISO 31000 requires documentation of risk decisions, ownership assignments, and treatment outcomes. In the AWS environment, this is accomplished by documenting and reporting risk management activities, as shown in Figure 2.

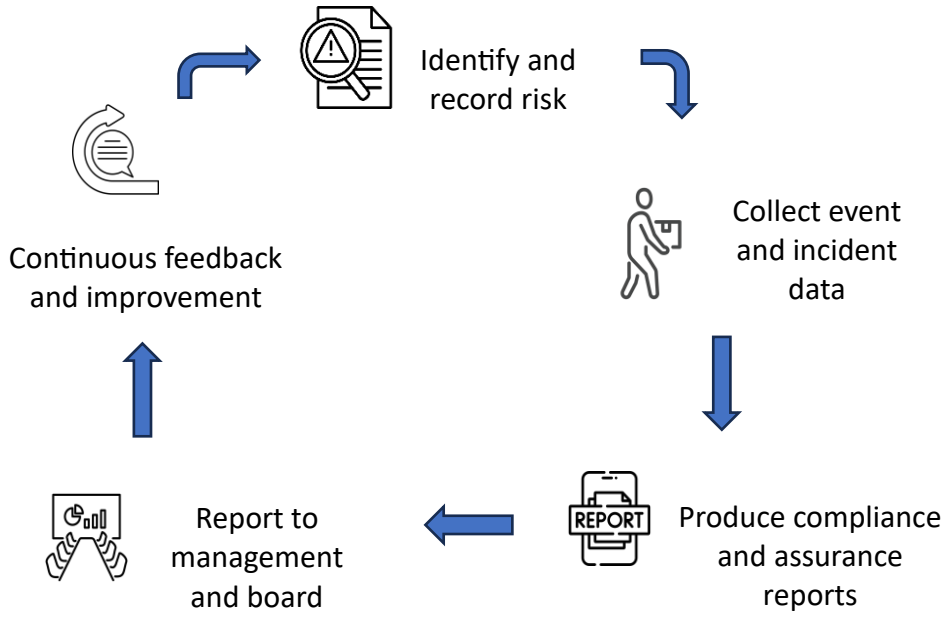

Figure 2: Risk management cycle

Implement these practices:

1. **Maintain a risk register:** Record, evaluate, and monitor each identified risk, including treatment decisions, responsible parties, and residual ratings. Extend your Governance, Risk, and Compliance (GRC) tools to track AWS-specific risks.
2. **Collecting event and incident data**: Tools like AWS CloudTrail and Amazon GuardDuty generate detailed logs that offer traceability for investigations and reporting. Centralized SIEM platforms aggregate CloudTrail and GuardDuty logs. This centralization improves audit trail completeness and reduces incident investigation time
3. **Producing compliance and assurance reports:** AWS Artifact gives access to compliance reports and certifications, while AWS Security Hub and AWS Config dashboards and scorecards support ongoing assurance against standards such as ISO 27001, SOC 2, or HIPAA.
4. **Reporting for management and the board**: Translate risk data into business impact metrics. Executives need risk metrics aligned to business objectives, risk appetite, regulatory compliance, and availability targets, not technical alert details.

Recording and reporting create an audit trail that documents risk decisions and treatment outcomes for future assessments and compliance verification.

Managing risk in AWS Cloud environments

AWS risk management requires continuous identification, analysis, evaluation, and treatment of risks through AWS services, policies, and automation. ISO 31000 alignment maps your risk tolerances to specific AWS security and compliance controls.

Connecting business-level risk management to AWS implementations aligns risk appetite to specific controls. Map identified risks to industry standards based on your risk appetite and regulatory requirements.

The AWS Shared Responsibility Model

The AWS Shared Responsibility Model divides security and compliance responsibilities between AWS and customers.

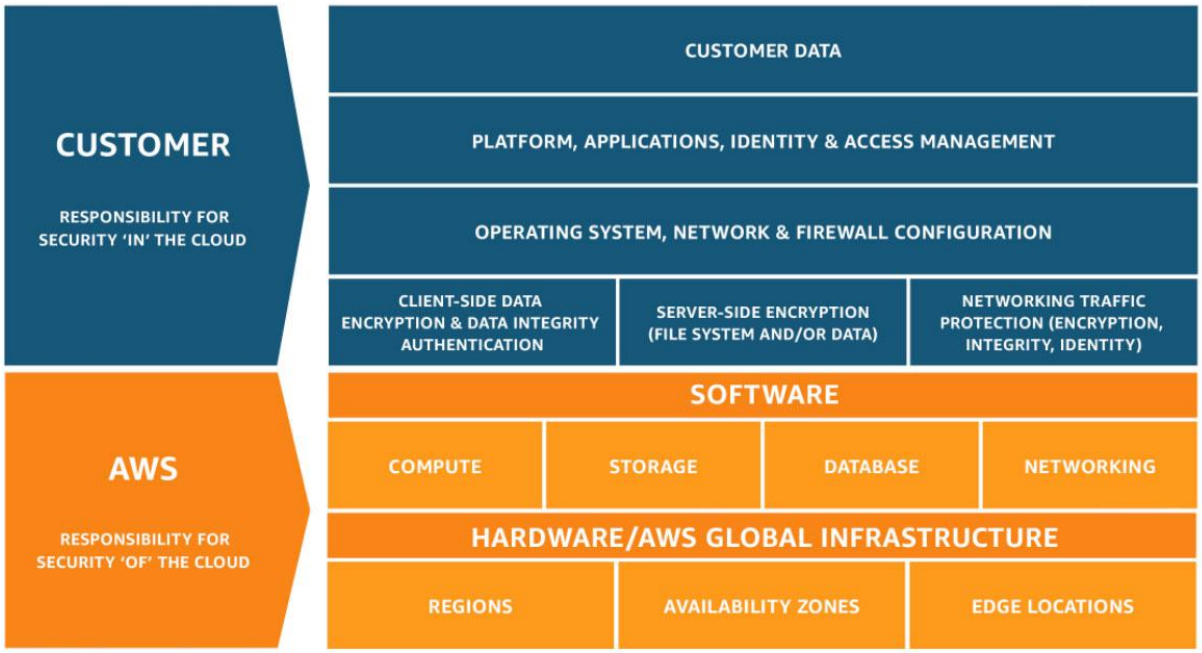

Figure 3: AWS Shared Responsibility Model

AWS is responsible for security of the cloud, which encompasses the infrastructure that runs the services offered in the AWS Cloud and can include hardware, software, networking, and facilities. Customers are responsible for security in the cloud, which can include customer data, platform configuration, identity and access management, and network controls.

Shared controls require both AWS and customers to address security and compliance, with each party handling different aspects of the same control area. In the shared responsibility model, AWS manages the infrastructure components while customers implement corresponding controls for their applications and data, helping provide substantial security coverage across the cloud stack. An example can be patch management, where AWS takes responsibility for maintaining, patching, and securing the underlying infrastructure, hypervisors, and host operating systems, while customers must apply patches and updates to their guest operating systems, applications, and any software they deploy on AWS services.

Applying principles for cloud risk management

ISO 31000 is built on eight core principles that guide risk management. Apply these principles to your AWS environment to support risk-informed decision making.

- **Integrated:** Risk management is an integral part of your activities. Embed risk considerations into cloud architecture, deployment, and operations. Integrate automated compliance checks into your CI/CD pipeline using AWS CodePipeline and AWS Config to evaluate each deployment against security and compliance requirements before production.
- **Structured and comprehensive**: A structured approach contributes to consistent and comparable results. Use AWS Control Tower and Security Hub to establish baseline controls across all accounts.
- **Customized**: The risk management framework must be tailored to your context (internal or external). You may configure AWS Config rules to align with specific PCI DSS requirements, creating custom rules that reflect your unique payment processing architecture.
- **Inclusive:** Include stakeholders in risk assessments to incorporate their knowledge and perspectives. A typical implementation is to use Amazon Simple Notification Service (Amazon SNS) to integrate security alerts from GuardDuty into your Slack channels, helping ensure that operational teams are aware of potential security issues.

- **Dynamic:** Vulnerabilities emerge daily, regulations change quarterly, and business priorities shift with market conditions. You may implement Security Hub to monitor for new vulnerabilities and configuration drift, adapting security controls as new threats emerge.
- **Best available information**: Risk management accounts for limitations and uncertainties in available information. You may activate Amazon Detective to gather and analyze detailed security data for security investigations.
- **Human and cultural factors:** Human behavior and culture influence risk management at every level. Organizations often enable AWS IAM Access Analyzer to monitor for overly permissive access policies that can result from human error or cultural tendencies toward convenience over security.
- **Continual improvement**: Risk management improves through learning and experience. Configure AWS Systems Manager to collect operational metrics and security findings, then feed this data into regular risk reviews to refine security controls.

Automating risk identification and detection

Detect misconfigurations and threats before they cause incidents. Address misconfigurations quickly to maintain security and operational stability. The AWS Well-Architected Framework emphasizes automatic remediations that can respond to items such as misconfigurations faster than manual processes, which helps minimize potential business impacts and reduce the window of opportunity for unintended uses. Unaddressed misconfigurations create growing risks that become harder to resolve.

Amazon Macie supports identification of risks related to sensitive data management and compliance. By classifying and monitoring sensitive data, you can better assess privacy risks and verify controls addressing data protection and regulatory compliance.

AWS Config detects configuration deviations and provides compliance evidence and change history for audit purposes.

Amazon CloudWatch is the core monitoring service for automated risk detection and identification across AWS environments. Amazon CloudWatch delivers data and insights to monitor workloads, respond to performance changes, optimize resource use, and maintain a unified view of operational health across both AWS and on-premises environments. Through

the collection of metrics across your resources in your architecture, you can collect and publish custom metrics and visualize them through dashboards. AWS best practices involve having a dedicated and centralized logging account. You can set your centralized alerts in this account, providing isolation and better visibility, while being able to define your key metrics, thresholds, and conditions.

By using Amazon CloudWatch you can collect metrics across your resources in your architecture, establish baseline monitoring capabilities and alarms that evolve with workload maturity, and support both standard AWS service metrics and custom metrics. You can then use the results to surface business-specific or derived metrics that reflect your unique operational requirements. Alarms must be able to alert teams when there is an indicator of performance degradation, malicious activity, or failure of a component. With proper alerting, responses can be quicker, more efficient, and potentially avoid further security or performance threats. Without automatic alerting in place, teams can't respond quickly enough to prevent extended downtime or minimize security vulnerabilities.

Choosing alarm types for CloudWatch

Amazon CloudWatch offers two alarm types: metric alarms and composite alarms.

Metric alarms: Monitor individual metrics or math expressions against defined thresholds. When breached, they can trigger actions such as SNS notifications, Amazon Elastic Compute Cloud (Amazon EC2) auto scaling actions, CloudWatch investigations, and Systems Manager OpsItems (incidents). Metric alarms serve as early warning indicators by monitoring leading indicators rather than lagging failure metrics. You can configure thresholds at 70–80% of capacity limits to identify resource exhaustion risks before service impact occurs.

Composite alarms: Combine multiple alarm states using rule expressions. They trigger only when the specified thresholds are met, reducing alert noise by requiring multiple related issues before alerting. Composite alarms support multi-service risk detection, monitoring risk indicators spanning multiple AWS services to identify failure patterns before they happen through dependencies.

Risk thresholds balance sensitivity with practicality. Calibrate sensitivity levels to minimize false positives while ensuring genuine incidents trigger responses.

Understanding your environment's normal operational levels is important in deciding on alert thresholds. Consider the following:

1. Collect baseline performance data across multiple days.
2. Analyze detailed metrics at the minute level.
3. Establish initial alert boundaries just above typical values.

Adopting ISO 31000 for automated risk identification positions you to alert the right personnel and trigger the right remediations in real time.

Automating controls

Maintaining consistent governance across growing AWS environments requires automation. Services such as AWS Control Tower and accelerators like the Landing Zone Accelerator (LZA) provide baseline frameworks and built-in controls that can be deployed across accounts and organizational units.

Balancing governance and security requires risk-based enforcement through AWS services that Integrate ISO 31000 principles so you can deploy automated controls at scale while meeting efficiency and compliance objectives.

AWS Control Tower is an approach for automated governance across multi-account environments that includes built-in guardrails that establish a baseline set of security controls. These guardrails include best practices and requirements such as activating CloudTrail logging and AWS Config across accounts. Additional controls can be included such as Amazon Simple Storage Service (Amazon S3) bucket encryption enforcement and root account access management.

The LZA extends control towers capabilities through infrastructure as code (IaC) deployments of customized control frameworks. The LZA provides sophisticated control architectures that help support compliance efforts through code-based configuration rather than manual deployment processes. These pre-built configurations include major frameworks including NIST 800-53, PCI-DSS, SOC 2 and more. The LZA accelerates control implementation but doesn't achieve compliance. You must still:

- Validate control effectiveness in your environment
- Customize configurations for your risk profile and business requirements
- Deploy additional controls beyond LZA baselines
- Maintain ongoing monitoring and assessment

- Undergo formal compliance audits

The LZA accelerates the implementation of common security controls, reducing the time and effort required to establish a compliant baseline. Recognize that compliance is an ongoing process requiring continuous validation and improvement. See Table A.

Control implementation level	Score	ISO 31000 risk treatment status	Description
No controls	5	Risk acceptance (unmanaged)	Control doesn't exist. Risk exposure exceeds your risk appetite and requires immediate risk treatment consideration.
Weak controls	4	Risk treatment required	Control is poorly implemented. Risk treatment activities must be prioritized to bring exposure within acceptable tolerance levels.
Adequate controls	3	Risk treatment ongoing	Control meets minimum requirements, but risk exposure remains above optimal levels. Continued risk treatment recommended to enhance effectiveness.
Strong controls	2	Risk treatment effective	Control implementation effectively manages risk exposure within your tolerance. Periodic risk monitoring and review recommended.
Fully controlled	1	Risk treatment optimized	Control implementation fully manages risk exposure below tolerance thresholds. Maintain current risk treatment with routine monitoring.

Table A: Control implementation ISO 31000 risk treatment alignment

Implementing continuous monitoring and reporting

When infrastructure changes frequently, periodic assessments create blind spots. Continuous monitoring tracks configuration changes, security events, and compliance status in real time.

AWS has architected its services to follow this continuous approach, including integration with

monitoring analysis and response. Instead of periodic review, security events and configuration changes can trigger immediate analysis and response workflows.

- **Amazon CloudWatch:** Frequently paired with CloudTrail as part of continuous monitoring. Amazon CloudWatch gives you a clear overview of performance, identifies any issues with error rates, and shows you how your resources are being used.
- **AWS CloudTrail:** Captures and records authenticated API activities within your AWS account in its supported Regions. Table B describes the structure of an AWS CloudTrail log.
- **Amazon GuardDuty:** One of the primary services used to detect unauthorized behavior, malicious activity, and threats is Amazon GuardDuty, a service that uses AI/ML tools such as predictive analysis, anomaly detection and monitoring to enhance its detections. GuardDuty can process billions of events from multiple data sources, such as CloudTrail, Flow Logs, and CloudWatch events.

Element	Description
Who	The principal that initiated the API call
Where	Source IP address where the request originated
When	Timestamp of when the action occurred
What	Specific action and service that was accessed or modified
Which	Resources that were affected by the API call

Table B: CloudTrail log entry structure

ISO 31000 implementation in AWS

The following is optional guidance that can be used to set up a risk management program based on ISO 30001:2018.

Required access: AWS Organizations administrator permissions, IAM permissions to create roles and policies, Access to AWS Control Tower, CloudTrail, Config, and Security Hub

Recommended knowledge: Familiarity with AWS IAM, CloudTrail, and Config. Understanding of ISO 31000:2018 risk management principles. Experience with multi-account AWS architectures

Estimated implementation time:
Phase 1: 2-4 weeks
Phase 2: 3-6 weeks
Phase 3: 4-8 weeks
Phase 4: Ongoing

Cost considerations: CloudTrail, Config, and GuardDuty incur per-event/resource charges. Security Hub charges per finding per month. Review AWS pricing pages for current rates

Phase 1: Foundation – Establishing core security architecture

The foundation phase establishes essential AWS security services that form the base of your risk management program. Establish core capabilities for thorough audit logging, identity management, compliance monitoring, and multi-account governance.

To establish AWS Control Tower for multi-account governance:

AWS Control Tower provides a ready-made, multi-account environment with built-in governance controls. Set up a landing zone that enforces security baselines and standardizes account provisioning to prevent configuration drift across accounts by enforcing guardrails. This reduces the risk of security gaps caused by inconsistent account setup.

Implementation steps:

1. Set up an AWS Control Tower landing zone to establish the core multi-account architecture.
2. Apply pre-built and custom SCPs as preventative guardrails to prevent high-risk actions.
3. Deploy AWS Config rules as detective guardrails to detect policy violations.
4. Standardize new account provisioning using Account Factory.

5. Establish centralized logging for your accounts.

Example implementation: You have more than 50 AWS accounts and implements AWS Control Tower to manage them. You organize accounts into OUs based on environment (production, development, and testing) and business function. You apply strict guardrails to production accounts, preventing public S3 buckets and requiring encryption for storage services. You customize Account Factory deploy standard networking configurations and security tools when new accounts are provisioned. Connect AWS Control Tower with AWS Service Catalog to provide self-service infrastructure that remains compliant with their policies.

Technical documentation:

- AWS Control Tower User Guide
- AWS Control Tower guardrails
- Customizing your landing zone

To configure AWS CloudTrail for thorough audit logging:

AWS CloudTrail provides visibility into user, role, or AWS service activity by recording actions taken in the AWS Management Console, AWS Command Line Interface (CLI), AWS SDKs, and API calls across your AWS environment. Establish a centralized, tamper-resistant audit trail across your accounts by capturing activity broadly, protecting log integrity for regulatory trustworthiness, and retaining records to meet framework-specific retention requirements. Activating CloudTrail Insights adds automated anomaly detection, enabling faster identification of unusual behavior without manual log review, which strengthens both incident response and overall compliance posture.

Implementation steps:

1. Configure CloudTrail at the AWS Organizations level to capture actions across your AWS environment.
2. Turn on CloudTrail log file integrity validation to detect changes to log files.
3. Set lifecycle rules for your AWS CloudTrail Amazon S3 bucket to meet compliance requirements.
4. Activate CloudTrail Insights to detect unusual API activity patterns.

Example implementation: You deploy organization-wide CloudTrail with both management and data events enabled. Direct CloudTrail logs to be delivered to a dedicated logging account's S3 bucket with server-side encryption and access policies. Feed these logs into Amazon Athena for SQL-based analysis and set up automated alerts for suspicious activities such as unauthorized IAM policy changes.

Technical documentation:

- Creating a trail for Your AWS account
- Organization-wide trails
- AWS CloudTrail Lake for SQL-based queries

To configure IAM access controls:

AWS Identity and Access Management (IAM) controls who can access your AWS resources and under what conditions. Enforce least-privilege access through role-based policies and strong authentication, including MFA and temporary credentials, which reduces the risk of unauthorized access from compromised or overly permissive accounts. Activating IAM Access Analyzer adds continuous monitoring for unintended resource exposure to external entities, helping identify access risks before they become security incidents or audit findings.

Implementation steps:

1. Establish IAM role-based access control policies based on job functions using AWS managed policies and custom policies.
2. Configure strong password policies for your AWS accounts.
3. Activate and require MFA for human users.
4. Turn on IAM Access Analyzer to identify resources shared with external entities.
5. Replace long-term credentials with temporary credentials using IAM roles for applications.

Example implementation: You establish a detailed IAM strategy where developers receive access only to development environments, with production access requiring just-in-time elevation through AWS IAM Identity Center. Apply SCPs that prevent the creation of IAM users with console access, enforcing federation through their identity provider. Privileged actions require MFA, and you use IAM Access Analyzer to monitor for unintended resource exposure.

Technical documentation:

- IAM best practices
- IAM Access Analyzer
- AWS IAM Identity Center

To configure AWS Config for resource compliance monitoring:

AWS Config monitors and records your AWS resource configurations, and assesses compliance against internal policies and regulatory requirements. Configure AWS Config across your accounts and Regions with conformance packs and custom to provide continuous, automated compliance monitoring, reduce configuration blind spots and reliance manual reviews. Assigning automatic remediation actions closes the loop by correcting non-compliant resources, minimizing the window of exposure and providing auditors with evidence that detective controls are paired with corrective responses.

Implementation steps:

1. Enable AWS Config across your accounts and Regions.
2. Use pre-built Conformance Packs.
3. Develop custom AWS Config rules specific to your requirements.
4. Assign automatic remediation actions for non-compliant resources.

Example implementation: You deploy AWS Config across your accounts with the Security Hub Operational Best Practices for PCI DSS conformance pack. Define custom rules to enforce tagging policies and encryption requirements specific to their business. Set up automated remediation actions for critical findings, such as activating S3 bucket encryption when unencrypted buckets are detected. Connect AWS Config with EventBridge to trigger notifications for compliance drift.

Technical documentation:

- AWS Config Conformance Packs
- AWS Config Managed Rules
- Remediating Noncompliant Resources

Phase 2: Integration – Aligning risk management with AWS tools

During integration, you connect ISO 31000 risk management processes with AWS tools to create a unified approach that supports risk identification, assessment, treatment, and monitoring.

To establish a risk register using AWS Systems Manager:

A risk register provides a centralized record of identified risks, their severity, and assigned treatment actions. AWS Systems Manager is a unified operations management service that provides visibility and control over your AWS infrastructure. Configuring OpsCenter with risk-tagged resources and Parameter Store thresholds creates a centralized, auditable risk register that links identified risks to the AWS resources they affect and provides assessors with traceable evidence of active risk management. Define automated runbooks for risk treatment for consistent, repeatable responses to identified risks, reducing human error and demonstrating to auditors that the organization has moved beyond documentation into operational risk remediation.

Implementation steps:

1. Use AWS Systems Manager OpsCenter to create operational items to track identified risks.
2. Tag resources with risk-related metadata.
3. Define risk assessment criteria and thresholds for the Parameter Store.
4. Create runbooks for automated risk treatment.

Example implementation: You use Systems Manager OpsCenter as their centralized risk register. Each identified risk is created as an OpsItem with standardized operational data fields for risk rating, risk owner, treatment strategy, and residual risk assessment. You link related AWS Config findings and Security Hub alerts to these OpsItems, creating a detailed view of each risk. You store risk acceptance criteria in Parameter Store as secure strings and develop automation documents that serve as standardized remediation runbooks for common risks. Apply the tagging capabilities available through Systems Manager to categorize risks according to their ISO 31000 framework.

Technical documentation:

- AWS Systems Manager OpsCenter
- Working with Parameter Store
- Creating runbooks

To enable Amazon GuardDuty for threat detection:

Amazon GuardDuty uses machine learning and threat intelligence to detect malicious activity and unauthorized behavior in your AWS accounts. Activate Amazon GuardDuty across your accounts and configure automated remediation. GuardDuty detects threats across accounts using machine learning, threat intelligence, and trusted IP lists to reduce false positives. Lambda-based remediation and SNS notifications turn detections into immediate action and awareness, reducing the time between threat identification and response.

Implementation steps:

1. Enable GuardDuty across your accounts.
2. Configure lists of trusted IP addresses.
3. Create Lambda functions for automated remediation.
4. Create SNS topics for finding notifications.

Example implementation: You enable GuardDuty across your accounts managed by AWS Organizations with the delegated administrator pattern. Build custom threat lists for known malicious IPs in their industry and trusted IP lists for their corporate networks. Configure automated response workflows using EventBridge rules that trigger Lambda functions to quarantine compromised EC2 instances by moving them to an isolated security group. Feed GuardDuty findings into their SIEM solution and establish different notification pathways based on finding severity. Critical findings trigger PagerDuty alerts, while medium-severity findings are routed to their ticketing system.

Technical documentation:

- Amazon GuardDuty User Guide
- Managing GuardDuty accounts with AWS Organizations
- Threat intelligence sets
- Automating response to GuardDuty findings

To implement compliance reporting with AWS Security Hub:

AWS Security Hub is a cloud security posture management service (CSPM) that aggregates findings from across your AWS environment and maps them to compliance frameworks. Enable Security Hub and configure custom views, automated responses, and GRC platform integration so findings reach the right people quickly and feed into existing governance workflows.

Implementation steps:

1. Activate Security Hub standards for compliance frameworks.
2. Create custom views for risk categories.
3. Set up automated responses to critical findings.
4. Connect Security Hub with third-party GRC platforms.

Example implementation: You Enable Security Hub with CIS AWS Foundations, AWS Foundational Security Best Practices, and PCI DSS standards enabled. Create custom insights that group findings by business unit, application, and risk category. Automate actions that create JIRA tickets for critical findings and Slack notifications for high-severity issues. Build a custom dashboard using Amazon QuickSight that provides executive-level reporting on security status and compliance status across their AWS environment. Connect Security Hub with their enterprise GRC platform using the Security Hub API to maintain a single source of truth for compliance reporting.

Technical documentation:

- AWS Security Hub User Guide
- Working with security standards
- Custom insights
- Custom actions

Phase 3: Optimization – Enhancing risk management through automation

Optimization enhances risk management capabilities through advanced automation, supporting more efficient risk responses.

To configure AWS Config rules for automated remediation:

Automated remediation reduces the time between detecting a non-compliant resource and returning it to a compliant state. Define custom Config rules and configure Systems Manager Automation to respond to violations. Custom Config rules paired with Systems Manager Automation remediate non-compliant resources, closing the gap between detection and correction without waiting for manual intervention. Retry logic handles transient failures, and deploying rules through AWS Organizations enforces the same standards across every account.

Implementation steps:

1. Develop custom AWS Config rules for your specific requirements.
2. Set up Systems Manager Automation for remediation.
3. Specify retry behavior for failed remediations.
4. Apply consistent AWS Config rules using AWS Organizations.

Example implementation: You set up automated remediation by implementing a thorough set of AWS Config rules aligned with their risk management framework. Define custom rules that enforce data sovereignty requirements specific to their jurisdiction. Automate remediation for common compliance issues, such as enabling default encryption for S3 buckets, removing overly permissive security group rules, and enforcing resource tagging policies. Deploy a multi-account strategy where remediation actions are executed by a central security account with cross-account roles. Customize EventBridge rules to notify security teams when automatic remediation fails, requiring manual intervention.

Technical documentation:

- AWS Config rules
- Remediating non-compliant resources
- AWS Config Managed Rules list

- Organization AWS Config rules

To configure AWS Lambda for automated risk response:

AWS Lambda functions support event-driven, automated responses to security findings without requiring dedicated infrastructure. Configure Lambda functions triggered by EventBridge rules automated response to security findings, removing the delay that comes with manual triage and intervention. Deploy functions across accounts with tiered severity to isolate a compromised resource for high-severity findings while logging lower-severity ones for review.

Implementation steps:

1. Develop AWS Lambda functions for specific risk scenarios.
2. Set up EventBridge rules to trigger functions.
3. Deploy AWS Lambda function to respond across accounts.
4. Arrange tiered responses based on risk severity.

Example implementation: You develop a suite of Lambda functions that respond to different risk scenarios. Deploy functions that revoke exposed IAM credentials, quarantine compromised EC2 instances, and restore modified security group rules to their baseline state. Establish a risk-based escalation workflow where different Lambda functions are triggered based on the severity of the detected risk. High-severity findings trigger immediate containment actions, while medium-severity findings initiate investigation workflows. Orchestrate complex multi-step response processes that include both automated actions and human approval steps for critical workflows.

Technical documentation:

- AWS Lambda Developer Guide
- Using Lambda with EventBridge
- AWS Step Functions for orchestration
- Cross-account access with Lambda

To set up Amazon EventBridge for automated risk response:

Amazon EventBridge routes security events from AWS services to response targets in real time. Configure event buses with pattern-matching rules and route security events to the right response targets as they occur, and replace manual monitoring with precise event-driven automation. Cross-account routing extends this to every account in the organization, so security events trigger the same response actions regardless of where they originate.

Implementation steps:

1. Set up dedicated buses for security events.
2. Configure event rules using patterns to match security-related events.
3. Assign targets to define response actions for matched events.
4. Set up cross-account event routing.

Example implementation: You build a centralized security event bus in your security account that receives events from accounts in your organization. Write EventBridge rules that detect security-relevant events such as root account usage, security group changes, IAM policy modifications, and GuardDuty findings. Different event patterns trigger response workflows, such as sending high-severity events to PagerDuty, medium-severity events to a Slack channel, and events to their SIEM solution. Add custom event transformations using Lambda functions to normalize event formats before sending them downstream. Apply EventBridge Pipes to filter, enrich, and transform security events before triggering automated remediation workflows.

Technical documentation:

- Amazon EventBridge User Guide
- Creating EventBridge Rules
- Event Patterns
- Cross-Account Event Delivery

To configure Amazon CloudWatch for advanced risk monitoring:

Amazon CloudWatch provides the metrics and alerting capabilities needed to detect operational risk conditions as you develop. Define custom risk metrics, configure composite alarms, and

build dashboards for ongoing monitoring. Custom metrics and composite alarms let you detect risk conditions that single-metric thresholds miss, such as a combination of elevated error rates and unusual access patterns occurring together. Metric math expressions and dashboards turn raw data into derived risk indicators that you can monitor in real time.

Implementation steps:

1. Define custom risk-related metrics.
2. Set up composite alarms for complex alerting conditions.
3. Create derived risk indicators using CloudWatch metric math expressions.
4. Create custom dashboards for risk monitoring.

Example implementation: You establish a detailed CloudWatch monitoring strategy for risk management. Define custom metrics that track key risk indicators such as the number of public-facing resources, IAM principals with admin privileges, and resources missing security controls. Construct composite alarms that correlate multiple risk signals, such as unusual API activity combined with network traffic to known malicious IP addresses. Deploy CloudWatch Synthetics canaries that regularly test security controls, such as verifying that public S3 buckets cannot be created. Build custom dashboards for different stakeholders, including technical dashboards for security analysts and executive dashboards that translate technical metrics into business risk indicators.

Technical documentation:

- Amazon CloudWatch User Guide
- Using Amazon CloudWatch metrics
- Composite alarms
- CloudWatch dashboards

Phase 4: Maturity – Achieving continuous risk management

At maturity, you establish continuous risk management and compliance processes that provide real-time visibility and support proactive risk management. Dashboards provide stakeholders with a consolidated view of risk status, compliance status, and operational health. Create

reporting views in Amazon QuickSight, Security Hub, and CloudWatch tailored to both technical and executive audiences.

Implementation steps:

1. Create Amazon QuickSight dashboards for risk reporting.
2. Create custom AWS Security Hub dashboards.
3. Create AWS CloudWatch dashboards to view operational risks.
4. Create business-focused risk views for executive reporting.

Example implementation: You create a multi-layered dashboard strategy for risk management. Configure technical dashboards in Security Hub that provide detailed visibility into security findings, compliance status, and vulnerability trends. Assemble operational dashboards in Amazon CloudWatch that monitor resource health, performance anomalies, and availability risks. Build business-oriented dashboards in Amazon QuickSight that translate technical metrics into business impact, showing risk exposure.

Technical documentation:

- Amazon QuickSight User Guide
- Creating Amazon QuickSight dashboards
- Security Hub custom insights
- CloudWatch dashboard JSON reference
- Amazon QuickSight dashboard embedding

Conclusion

ISO 31000 provides a framework for risk identification, analysis, evaluation, treatment, monitoring, and reporting that adapts to any organizational context. Integrating these principles into AWS bridges business-level risk management with practical cloud controls.

By establishing clear scope, context, and criteria, you can align risk processes to your unique environments and goals, enhancing focus and efficiency. Systematic risk assessment, using both qualitative and quantitative methods, supports informed prioritization and treatment of risks.

The treatment phase uses cloud-focused controls, automation, and compliance tools to balance agility with security and resilience.

Continuous monitoring and review with AWS services creates a feedback loop that adapts risk management to evolving threats and business changes. Thorough documentation and transparent reporting you may foster accountability and support ongoing improvement.

The AWS Shared Responsibility Model demonstrates cooperative risk governance by defining duties between AWS and customers. Embedding ISO 31000's eight core principles into cloud risk management enables you to manage your complex environment and strengthen resilience.

Start by assessing your risk management maturity against ISO 31000, identifying gaps in your AWS environment, and prioritizing where automation can deliver immediate value. Engage stakeholders across your organization and use AWS Security Assurance Services for advisory guidance tailored to your objectives.

Contributors

The following individuals and organizations contributed to this document:

- Jesse McMahan, Sr. Assurance Consultant, AWS Security Assurance Services
- Juan Rodriguez, Associate Assurance Consultant, AWS Security Assurance Services
- Sana Rahman, Sr. Assurance Consultant, AWS Security Assurance Services
- Akanksha Chaturvedi, Sr. Assurance Consultant, AWS Security Assurance Services
- Mayur Jadhav, Sr. Assurance Consultant, AWS Security Assurance Services

Document revisions

Date	Description
April 29, 2026	First publication