AWS re:Invent

NOV. 29 – DEC. 3, 2021 | LAS VEGAS, NV

SEC325-R2

Manage secrets like a pro

Ritesh Desai
Sr. SDM
AWS Secrets Manager

Avni Rambhia
Sr. Product Manager, Technical
AWS Secrets Manager

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Customers want to . . .

- Centrally manage secrets
- Secure storage and retrieval of secrets
- Control access to secrets
- Centrally audit and monitor secrets usage
- Ensure compliance with organizational compliance regulations and standards
- Return to life before COVID

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

What to expect in today's session

- Learn about the right AWS service to secure a given type of secret
- Secrets management: Shared Responsibility Model
- Getting the most out of AWS Secrets Manager

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Secrets of many types

IAM credentials

Database credentials

Oauth tokens

Encryption keys

SECRETS

Application credentials

SSH keys

Certificates (and private keys)

Application programming interface (API) keys

. . .

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Secrets in AWS

AWS credentials

Encryption keys

SSH keys

Private keys + Certificates

Database credentials

Application credentials

OAuth tokens

Application programming interface (API) keys

AWS Identity & Access Management (IAM)

AWS Key Management Service (KMS)

Amazon EC2 Instance Connect

AWS Certificate Manager

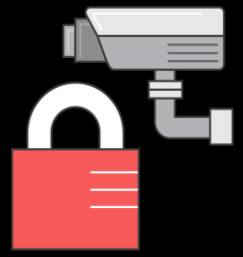
AWS Secrets Manager

aws

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Manage secrets through their lifecycle

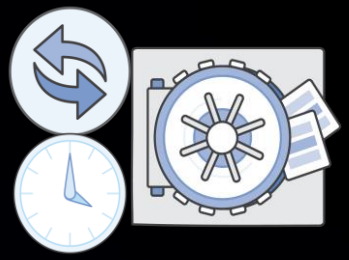
AWS Secrets Manager enables customers to create, rotate, manage, and retrieve database credentials, API keys, and other secrets while ensuring compliance



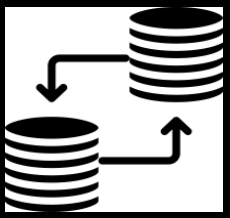
Secure by default



Fine-grained access control



Rotate



Replicate



Retrieve at scale



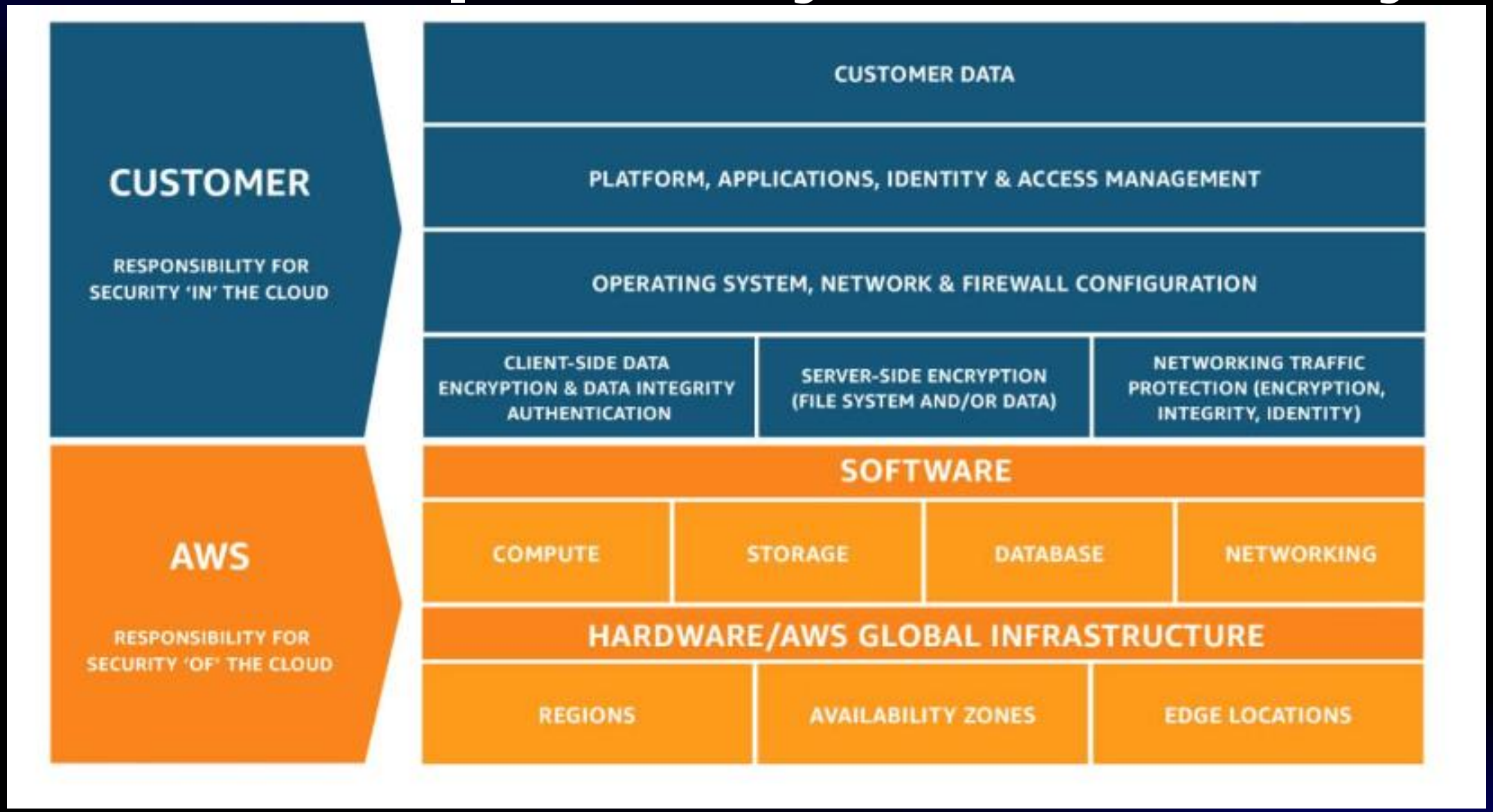
Monitor and audit



Meet compliance

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS shared responsibility – Cloud security



© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Create and use your secrets securely

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Why does it matter?

- Secrets enable access to sensitive data
- Tightly control access to secrets
  - At creation
  - During use
  - During updates

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Pro tip #1: Automate provisioning of secrets through AWS CloudFormation

- Secret: Specify a value, or let Secrets Manger generate one for you
- Other supported resource types: ResourcePolicy, RotationSchedule, SecretTargetAttachment

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Pro tip #2: Rotate secrets immediately after manual creation

- Ensures any access to current secret value is controlled
- Via console: Rotation resources can be automatically provisioned, while creating a secret
- Via API/CLI: RotateSecret
- Ad-hoc: PutSecretValue + Resource update

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Pro tip #3: Scope down to least privilege access

- Can be liberal with ListSecrets and DescribeSecret
- Applications need read-only permissions to GetSecretValue
- Use conditions to allow access to “Current” version of the secret

Additional tip: You can use the IAM Policy Simulator (https://policysim.aws.amazon.com/) to help you learn which principals have access to a secret through an IAM policy

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Pro tip #4: Using AWS managed policies

- SecretsManagerReadWrite
- IAMFullAccess**

** Trusted administrators only and enforce multi-factor authentication (MFA)

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Pro tip #5: ABAC

For convenient bulk-editing of permissions:

- Tag groups of related secrets
- Use Attribute-Based Access Control (ABAC) to simplify and automate permissions

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Pro tip #6: Leverage KMS CMKs for added control

- Admins can bulk-revoke application's access to secrets by revoking CMK permissions
- Encrypt secrets with CMK to share them across accounts

Note: You must use resource-based policies and CMKs to share secrets across accounts

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Pro tip #7: Use AWS Secrets Manager to secure secrets across AWS services

Containers



Amazon ECS



Amazon EKS



AWS Fargate

Analytics



Amazon Redshift



Amazon EMR



AWS Glue



Amazon MSK

Internet of Things



AWS IoT Greengrass

Database



Amazon RDS



AWS DMS



Amazon DocumentDB

Developer Tools



AWS CodeBuild

Machine Learning



Amazon SageMaker



Amazon Kendra

Application Integration



Amazon AppFlow



Amazon EventBridge

Management & Governance



AWS OpsWorks

aws

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Retrieve secrets securely

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Why does it matter

- Eliminate the need for developers to view or handle secrets
- Easily revoke access to secrets

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

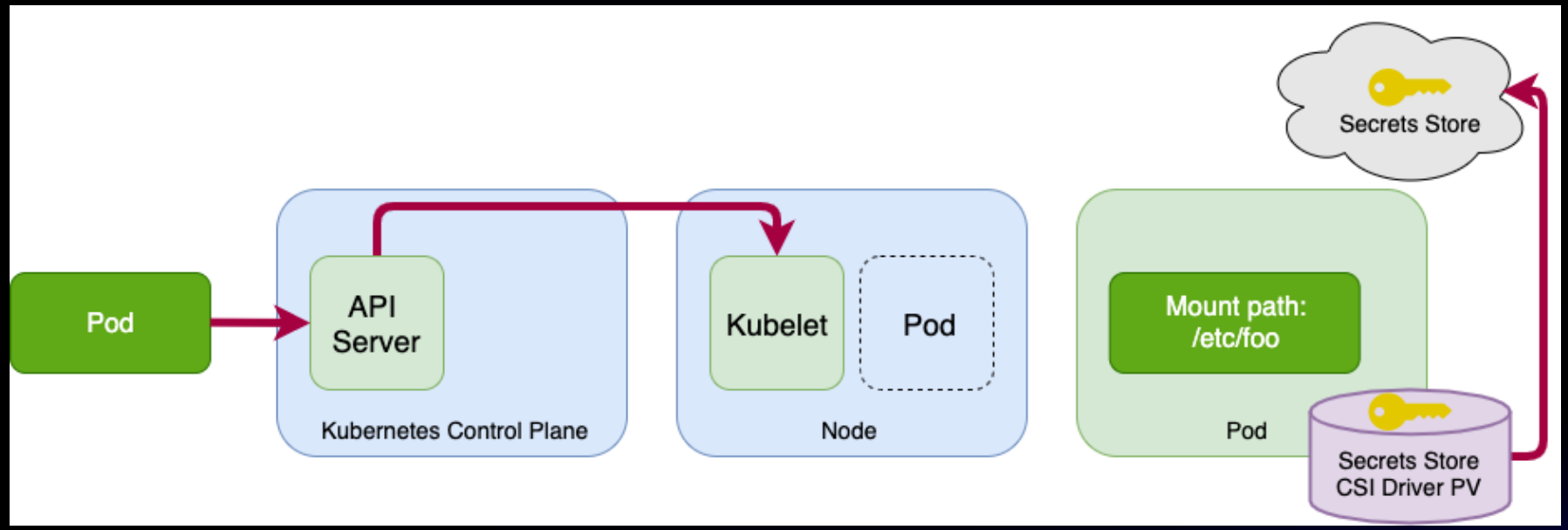
Pro tip #1: Use client-side caching libraries instead of directly calling GetSecretValue

- Reduce API bill, increase availability, reduce undifferentiated heavy lifting
- Java Database Connectivity (JDBC) driver automatically refreshes secret after it rotates
- Other client-side libraries (Java, Python, Go, .NET, Kubernetes) can refresh periodically

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Pro tip #2: Use the AWS Secrets and config provider to securely mount secrets to Kubernetes workloads

- Scope secrets to pods within clusters
- Use rotation reconciler to mount rotated secret values



© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Pro tip #3: For AWS Lambda users, use global variables instead of retrieving secrets inside Lambda handler

- Reduces API cost, prevents throttling
- Prevents GetSecretValue API calls every time the Lambda executes

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Pro tip #4: Schedule deletion of unused secrets

- Identify unused secrets using AWS Config



○	secretsmanager-secret-unused	SecretsManager, Secret, Unused	Checks if AWS Secrets Manager secrets have been accessed in the past 90 days. This rule is NON_COMPLIANT if a secret has not been accessed in 90 days.

- Schedule deletion after 7–30 days to delete safely using

  --recovery-window-in-days

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Pro tip #5: Account for rotation strategy when handling errors

- Two secrets valid at a time (recommended):
  - Refresh secrets at half-life
- Only one secret valid at a time:
  - Refresh aggressively, cache connections, or proactively broadcast update via rotation Lambda
  - Retries with exponential back-off and jitter
  - Set higher lockout limits on associated resources

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Rotate secrets

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Why does it matter

- Short-term secrets have lower blast radius than long-term secrets
- Enhance security posture
- Simplify compliance

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

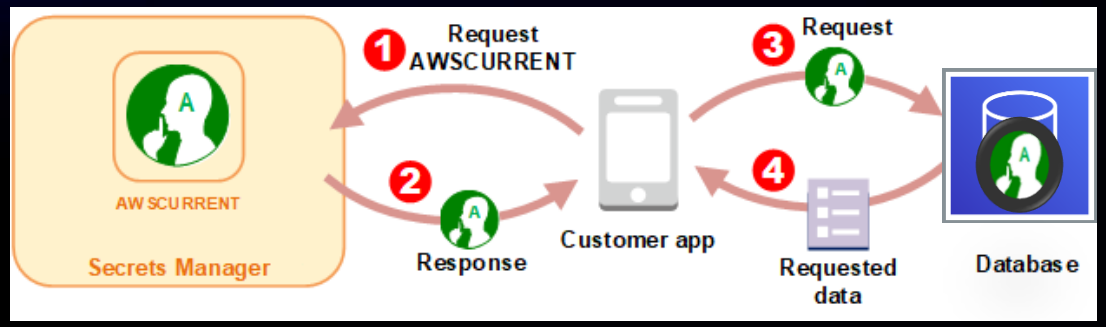
Pro tip #1: Alternate between two credentials for high-availability applications

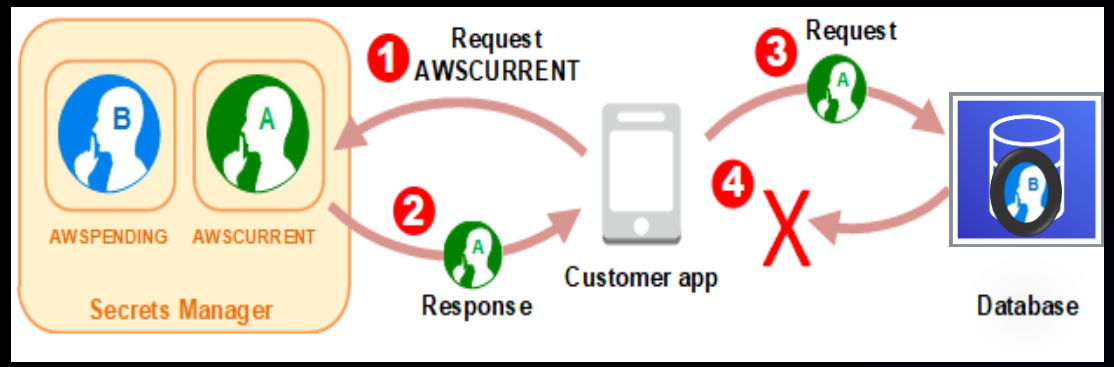
Account for rotation strategy when handling errors

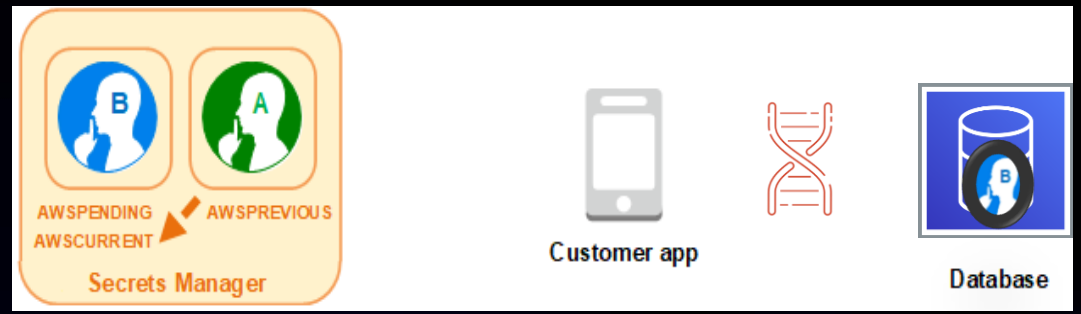
- Two secrets valid at a time (recommended):
  - Refresh secrets at half-life
- Only one secret valid at a time:
  - Refresh aggressively, cache connections, or proactively broadcast update via rotation Lambda
  - Retries with exponential back-off and jitter
  - Set higher lockout limits on associated resources

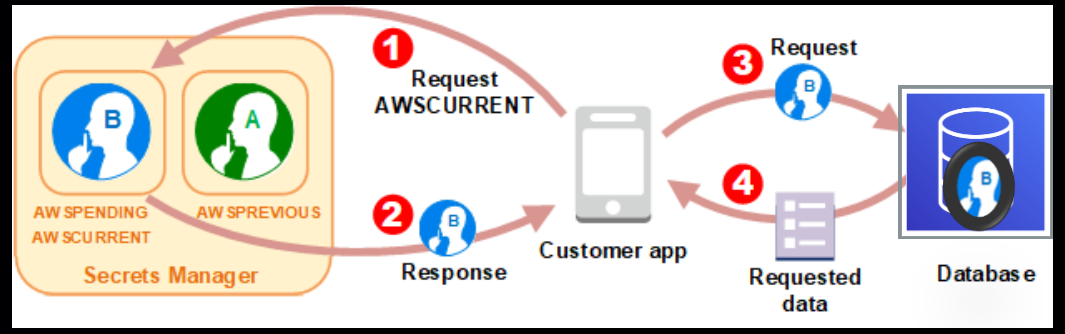
© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Detour: Single user rotation



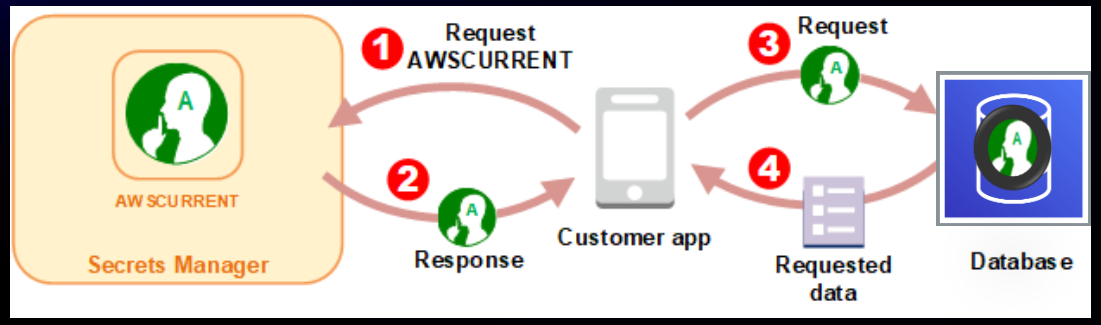


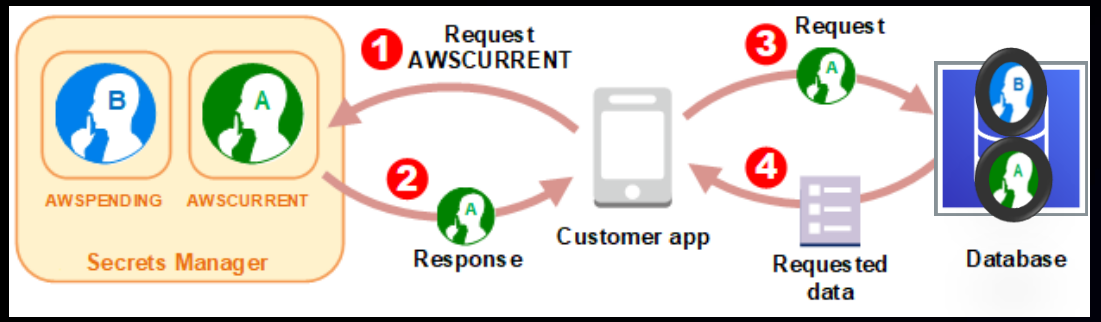


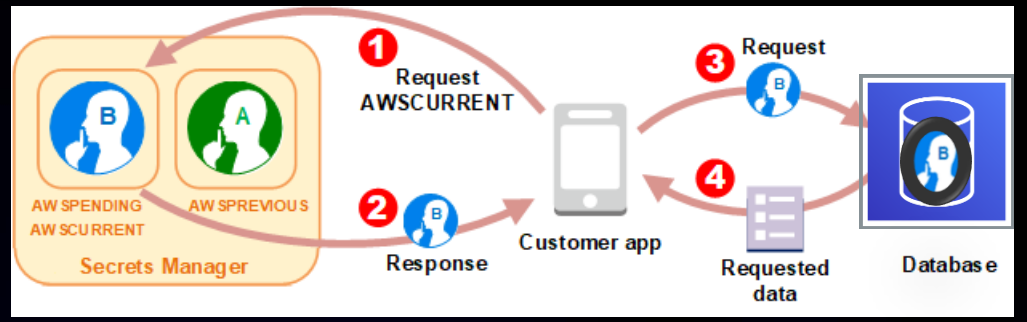


© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Two user rotation







© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Pro tip #2: Reuse rotation Lambdas for multiple secrets

- Easier to separate IAM permissions from Secrets Manager permissions
- Easier to manage a small number of rotation Lambda functions
- Customize password generation to your downstream application requirements

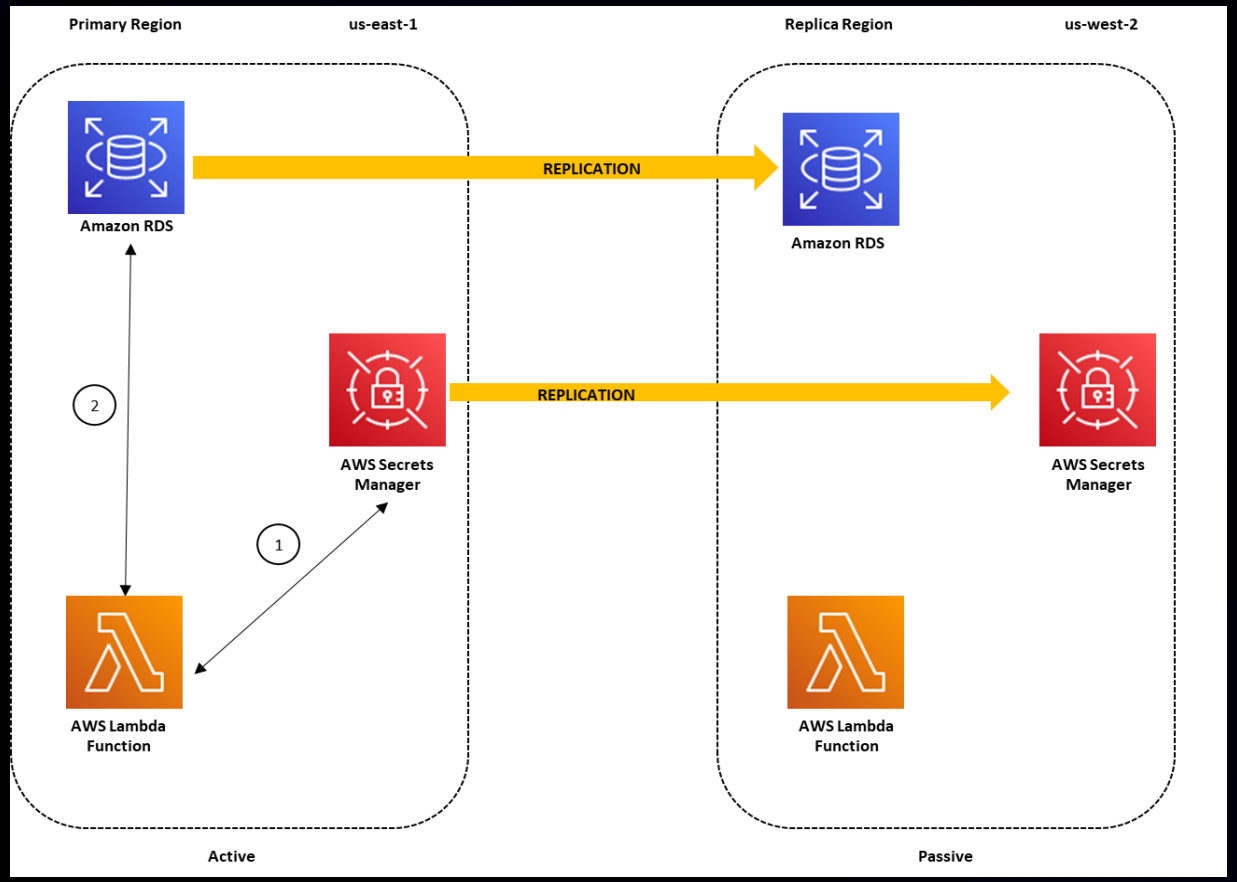
© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

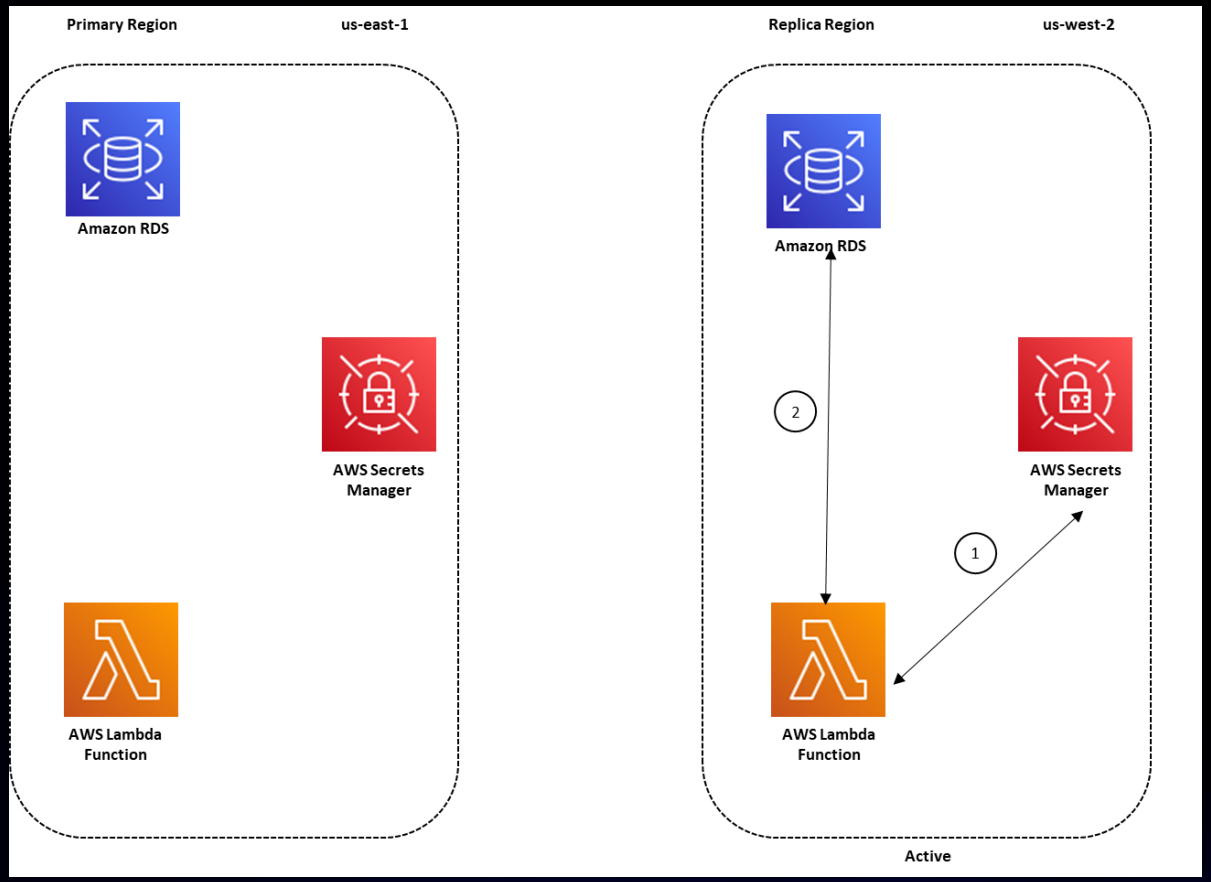
Pro tip #3: Manage and monitor rotation

- By default, secrets rotate every 30 days;
  set your rotation based on your compliance needs
- Configure Config rules to watch for secrets
  that have not rotated as expected
- Leverage AWS CloudTrail events for rotation status

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Pro tip #4: Use multi-region secrets for multi-region resources





© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Audit and monitor

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Why does it matter

- Identify unexpected usage or changes, so you can investigate
- Identify unwanted changes, so they can be rolled back
- Compliance

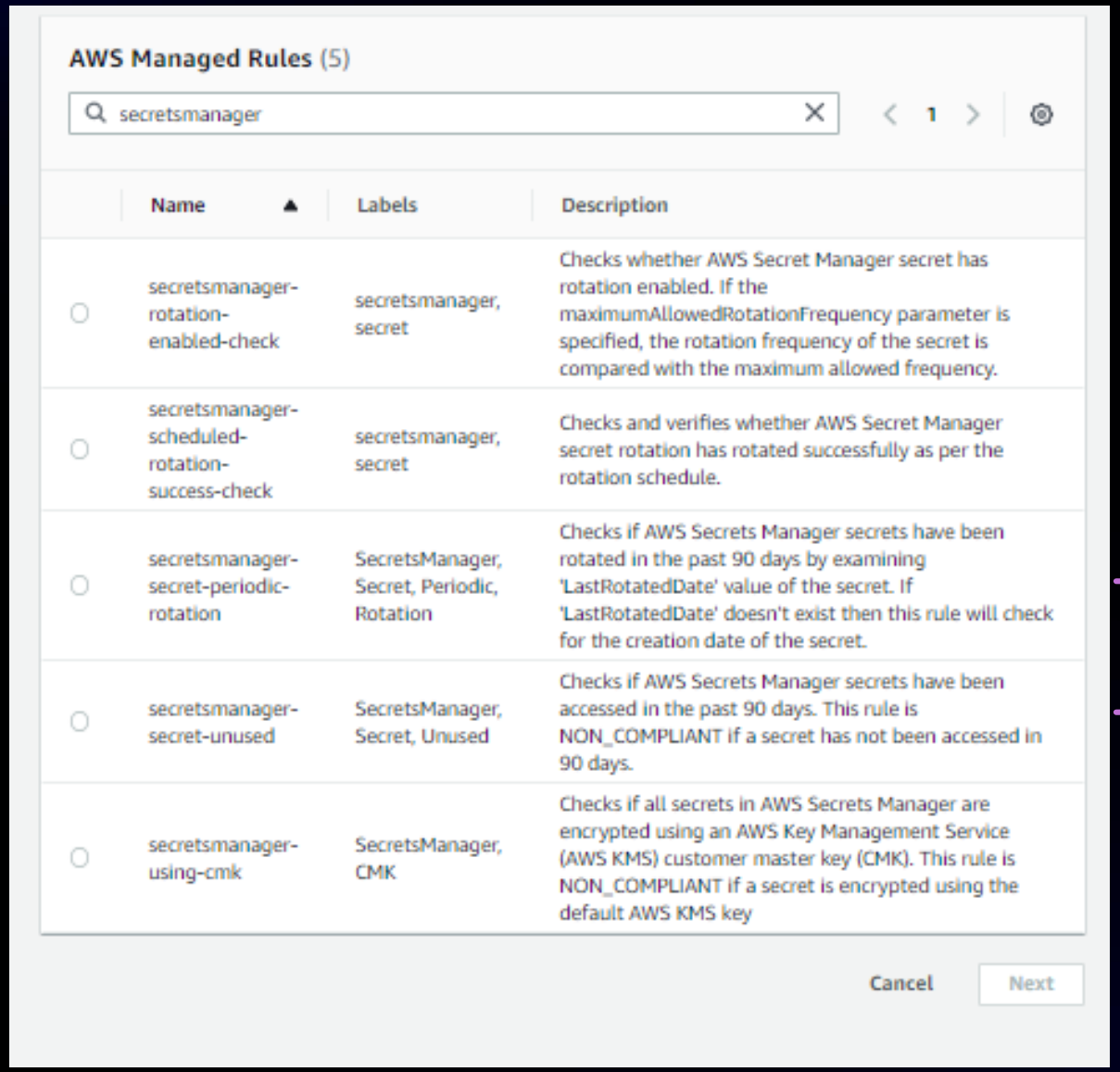
© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Pro tip #1: Leverage AWS services to monitor secrets

- Use IAM Access Analyzer to analyze permissions to secrets
- Use CloudTrail logs to monitor use
- Use Amazon CloudWatch events to audit secrets
- Use AWS Config to evaluate compliance rules (e.g., for schedule rotation, etc.)

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

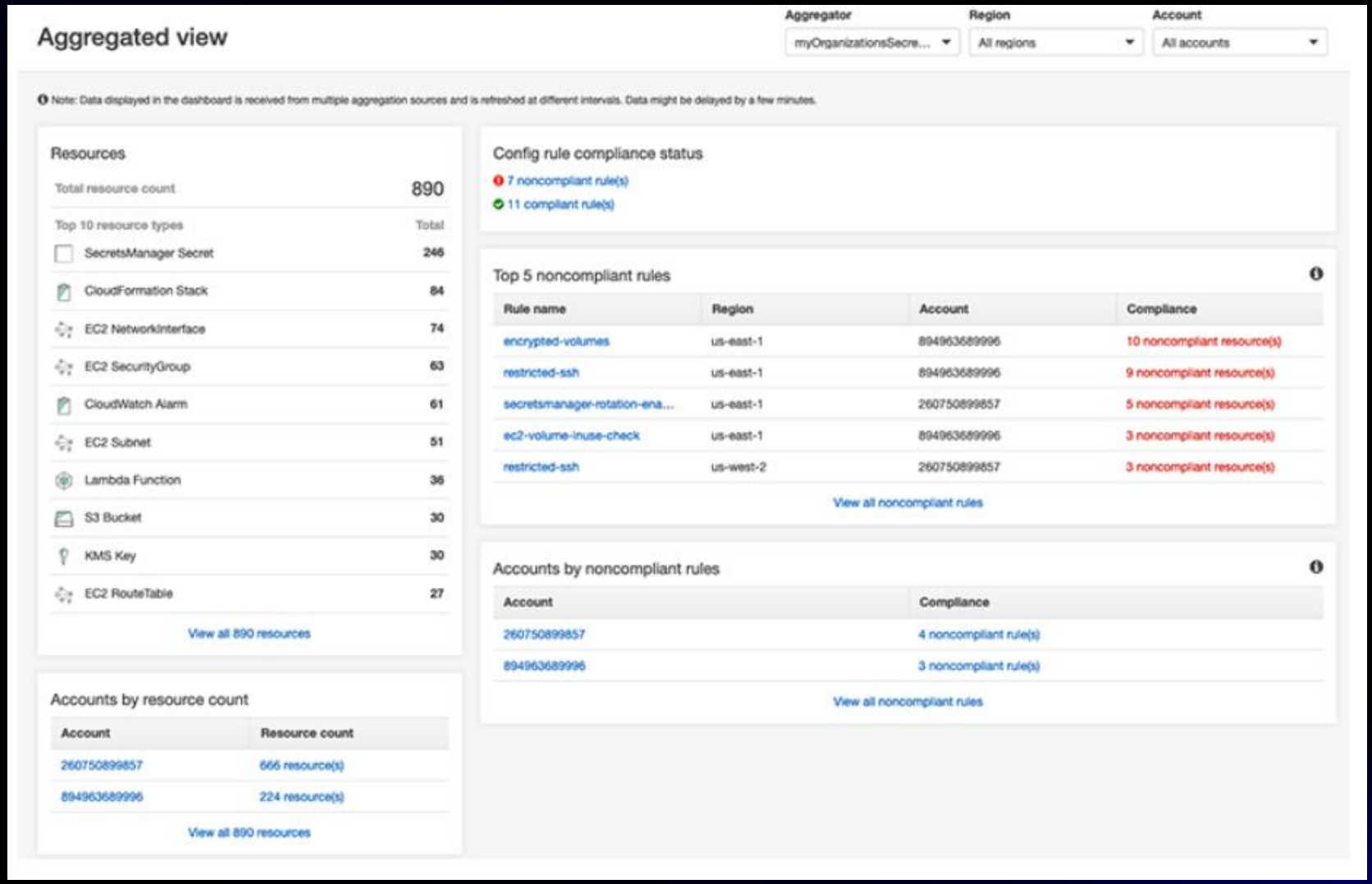
Pro tip #2: Evaluate compliance in an account using AWS Config



PCI compliance for secrets

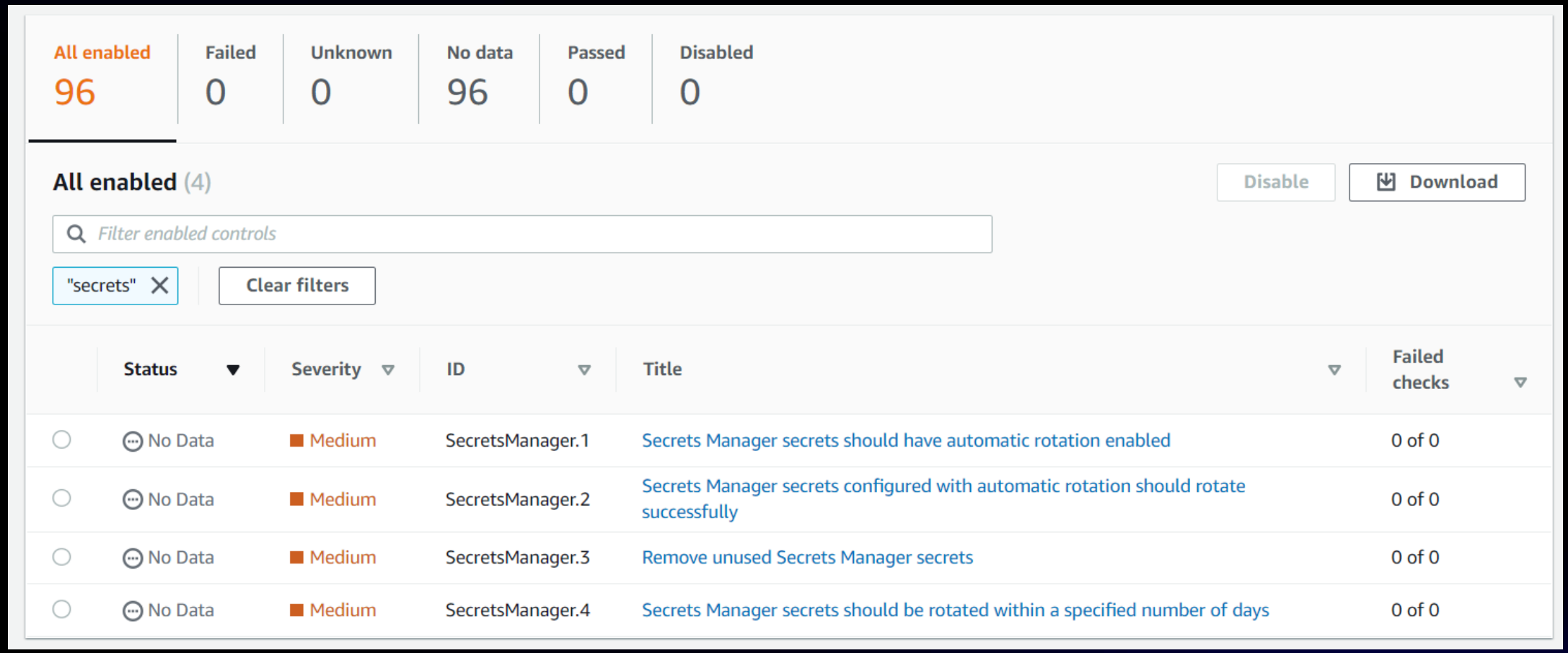
© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Pro tip #3: Evaluate compliance across accounts with AWS Config aggregator



© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Pro tip #4: Monitor organizational compliance using AWS Security Hub



© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Decentralize secrets

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Pro tips

Create and store secrets in the accounts where they will be used

- Preserves velocity of development teams
- Reduces blast radius from compromised account credentials
- Prevents cross-application operational impact

Monitor secrets centrally, using best practices and pro tips in previous section

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

SRM for Secrets Manager (examples)

Customer

- Properly configure policies on secret and key that encrypts it
- Use HA strategies and caching libraries to keep applications running while secrets rotate
- Set monitors for non-compliant secrets
- Be aware of service limits, and load test

AWS

- Encrypt secrets by default
- Enforce IAM and resource policies
- Best practices and code samples for safe rotation
- Client-caching libraries
- Integrations with AWS monitoring services
- SLA

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Final thoughts

- AWS manages our secrets with Secrets Manager – When you use Secrets Manager, you benefit from our mindset of “security is job zero”
- Minimize opportunities for secrets and humans to mix; automation is key
- Continue to raise the bar on security **and** availability; build a strong security culture, make it important, and include everyone in your organization

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

References – Getting started

- Tutorials:
  https://docs.aws.amazon.com/secretsmanager/latest/userguide/tutorials.html
- Understanding rotation:
  https://docs.aws.amazon.com/secretsmanager/latest/userguide/rotating-secrets-lambda-function-overview.html
- Manage database secrets:
  https://aws.amazon.com/blogs/security/how-to-use-aws-secrets-manager-rotate-credentials-amazon-rds-database-types-oracle/
- Manage private keys and certificates:
  https://aws.amazon.com/blogs/security/use-aws-secrets-manager-to-simplify-the-management-of-private-certificates/
- Tag and organize secrets:
  https://aws.amazon.com/blogs/security/identify-arrange-manage-secrets-easily-using-enhanced-search-in-aws-secrets-manager/
- The BBC's adoption journey:
  https://d1.awsstatic.com/events/reinvent/2019/REPEAT_1_How_the_BBC_uses_AWS_Secrets_Manager_to_manage_secrets_SEC354-R1.pdf

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

References – Centralized management

- Track secret status with Config: https://aws.amazon.com/blogs/security/how-to-track-changes-to-secrets-stored-in-aws-secrets-manager-using-aws-config-and-aws-config-rules/
- Check compliance status with Security Hub: https://docs.aws.amazon.com/securityhub/latest/userguide/securityhub-standards-fsbp-controls.html
- Resource-based policies: https://aws.amazon.com/blogs/security/how-to-use-resource-based-policies-aws-secrets-manager-console-to-securely-access-secrets-aws-accounts/

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

References – Caching library resources

- Overview:
  https://docs.aws.amazon.com/secretsmanager/latest/userguide/use-client-side-caching.html
- .NET Tutorial:
  https://www.youtube.com/watch?v=D-YOcfeZPCk
- Access secrets in Kubernetes:
  https://aws.amazon.com/blogs/security/how-to-use-aws-secrets-configuration-provider-with-kubernetes-secrets-store-csi-driver/
- Efficient access in Lambda:
  https://aws.amazon.com/blogs/networking-and-content-delivery/securing-and-accessing-secrets-from-lambdaedge-using-aws-secrets-manager/

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Thank you!

Ritesh Desai
ritedesa@amazon

Avni Rambhia
arambhia@amazon

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.