AWS Systems Manager Parameter Store

1.1 **Overview**

- AWS Systems Manager Parameter Store is a secure place to store configuration values like:
  - Database passwords
  - API keys
  - Application environment variables
  - Amazon Machine Image (AMI) IDs
- Instead of hardcoding values inside code, applications fetch them securely from Parameter Store.
- We can store values as plain text or encrypted data.
- A Parameter Store parameter is any piece of data that is saved in Parameter Store, such as a block of text, a list of names, a password, and so on. You can centrally and securely reference this data in your scripts, commands, and SSM documents. It provides three types of parameters to store and manage configuration data.
  - String: Stores plain text values, such as names, IDs, or file paths. Example: abc123, Whoami.
  - StringList: Stores a comma-separated list of values. Useful for handling multiple values in a single parameter. Example: Now, where, when, how
  - SecureString: Used for storing sensitive data securely, such as passwords or API keys. Encrypted with AWS Key Management Service (KMS) to prevent exposure.
  - Never store sensitive data in String or StringList parameters. Always use SecureString for encrypted storage.

1.2 **Use-cases and Features**

- Store sensitive data securely (passwords, tokens). Supports AWS Key Management Service (KMS) encryption for enhanced security.
- Centralize application configuration. All configuration values are managed from one central location.
- Update config without redeploying code. Configuration changes can be made without touching or redeploying the application.

AWS Systems Manager Parameter Store

- Versioning & History. Automatically track parameter changes, allowing us to roll back to previous versions if needed.
- Control access to parameters using IAM permissions.
- Integration with AWS Services. Works with AWS Lambda, EC2, ECS, and other services for seamless configuration management.
- Automation & Notifications. Integrates with AWS CloudWatch and AWS Lambda to automate responses when parameters change.
- Two Parameter Types:
  - Standard Parameters: Support up to 4 KB of data per parameter. Total parameters per account and region are 10,000.
  - Advanced Parameters: Support up to 8 KB of data and offer additional capabilities like version history. Total parameters per account and region are 10,000.
- Parameter Store uses folder-like names (for example, /dev/app/db-password) to organize settings, making it easy to group and manage configurations by environment or application.
  Env ----> Service ----> Resource ----> Configuration Name
- A standard parameter can be upgraded to an advanced parameter at any time, but an advanced parameter cannot be downgraded. This is because downgrading could cause data loss, remove attached policies, and change the encryption used.

1.3 **Architecture Flow**

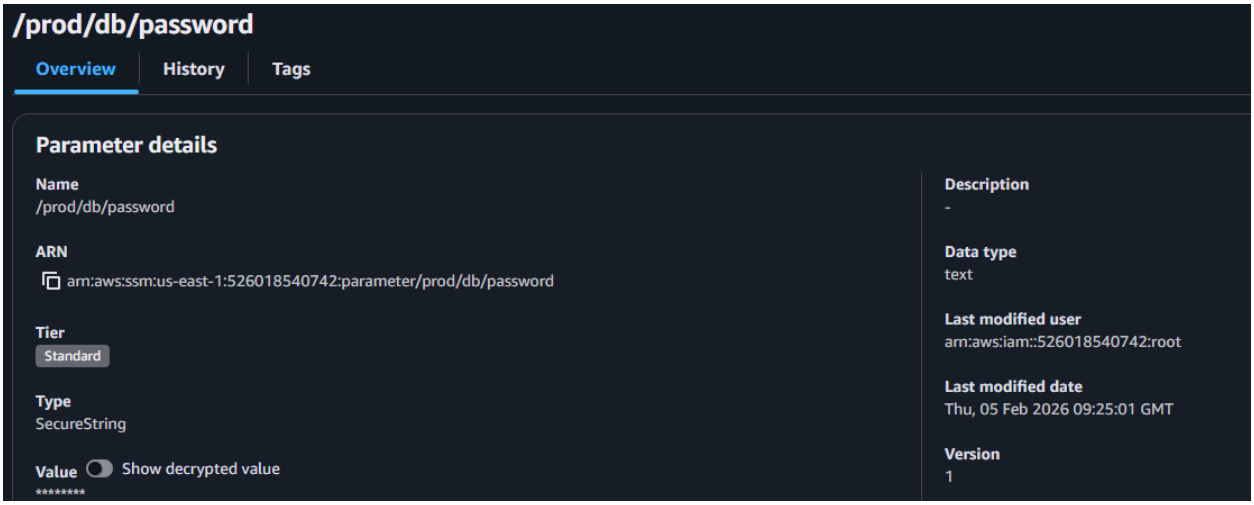
Developer
↓
Stores Config / Secret
↓
AWS Parameter Store (Encrypted with KMS)
↓
IAM Permission Check
↓
EC2 / Lambda / ECS
↓
Application Uses Value at Runtime

AWS Systems Manager Parameter Store

- User stores a parameter (e.g., DB password) in Parameter Store
- Parameter is encrypted using AWS KMS
- EC2 / Lambda / ECS fetches the parameter at runtime
- IAM controls who can read or write the parameter

1.4 **Execution Steps**

- Store a database password securely and retrieve it from an EC2 instance.

- ❖ Go to AWS Console.
  - ➢ Systems Manager
  - ➢ Parameter Store
  - ➢
- ❖ Click Create parameter
- ❖ Enter:
  - ➢ Name: /prod/db/password
  - ➢ Type: SecureString
  - ➢ Value: your password
- ❖ Choose KMS key (default is fine)
- ❖ Save parameter
  - ➢ 

- ❖ Attach IAM permission to EC2/Lambda to allow ssm:GetParameter

AWS Systems Manager Parameter Store

- Policies
- Create policy
- JSON
- {

  "Version": "2012-10-17",

  "Statement": [

  {

  "Effect": "Allow",

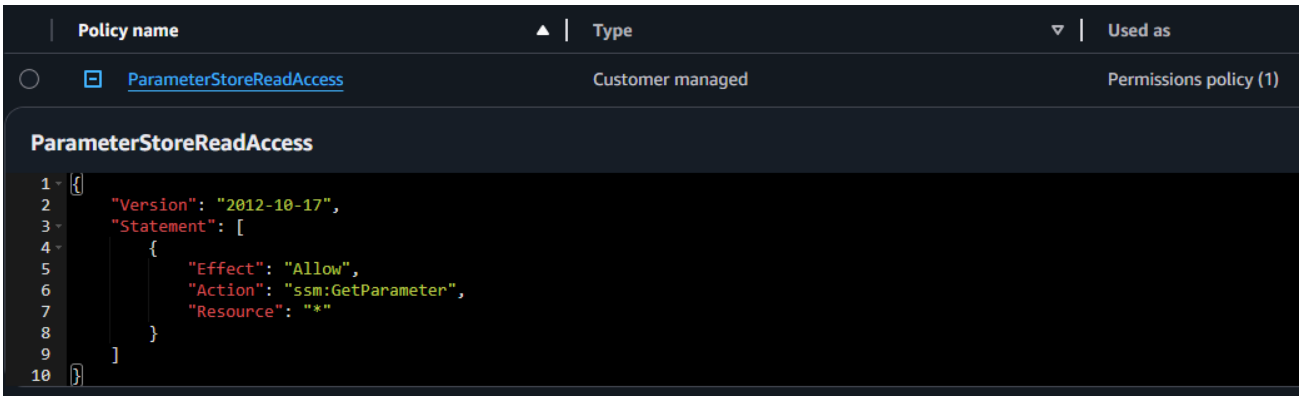
  "Action": "ssm:GetParameter",

  "Resource": "*"

  }

  ]

  }
- Name: ParameterStoreReadAccess
- This controls who can read the secret.

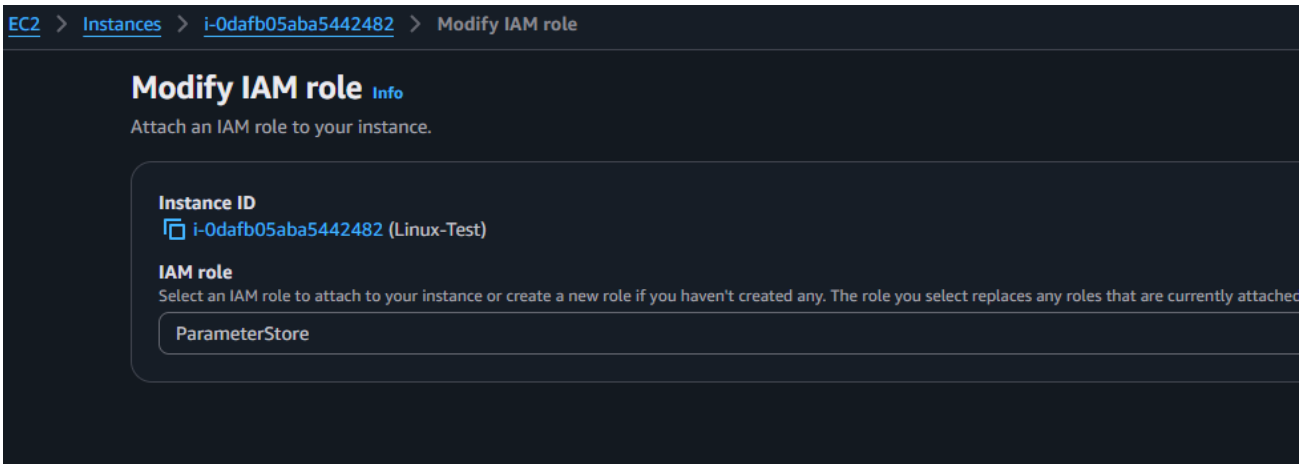


❖ Attach Policy to EC2 Role

- IAM
- Roles
- Select EC2 Roles
- Attach policies

AWS Systems Manager Parameter Store

- ParameterStoreReadAccess



- Application fetches the value at runtime.
  - Connect to EC2
  - Run command:

    aws ssm get-parameter \

    --name "/dev/app/db-password" \
    --with-decryption

```
[ec2-user@ip-172-31-72-11 ~]$ aws ssm describe-parameters \
> --query "Parameters[].Name"
[
    "/prod/db/password"
]
[ec2-user@ip-172-31-72-11 ~]$ aws ssm get-parameter \
> --name "/prod/db/password" \
> --with-decryption
{
    "Parameter": {
        "Name": "/prod/db/password",
        "Type": "SecureString",
        "Value": "Deadzone001",
        "Version": 1,
        "LastModifiedDate": "2026-02-05T09:25:01.025000+00:00",
        "ARN": "arn:aws:ssm:us-east-1:526018540742:parameter/prod/db/password",
        "DataType": "text"
    }
}
```

- Flow: User stores secret → Securely Encrypted → EC2 Fetches → App Uses It