DSC at Scale with AWS Systems Manager

Andrew Pearce
Senior Systems Development Engineer
@austoonz

© 2019, Amazon Web Services, Inc. or its Affiliates.

Session Topics

- Desired State Configuration at scale on AWS
    - What is DSC on AWS?
    - Using DSC at scale: growing from 1 to N instances
- DSC Enhancements with AWS
    - Token substitution
    - Multiple configurations
    - Credential handling
    - Reboot behaviors
    - Compliance reporting
    - PowerShell Module dependencies

© 2019, Amazon Web Services, Inc. or its Affiliates.

aws

What is DSC on AWS?

- AWS Systems Manager document: AWS-ApplyDSCMofs
- Launched November 2018
- LCM like experience, without using LCM
- PowerShell script that wraps all DSC logic
- Provides extensions to enhance DSC capabilities
- Provides rich compliance and execution reporting

© 2019, Amazon Web Services, Inc. or its Affiliates.

aws

Using DSC at scale: growing from 1 to N instances

- Use Run Command or State Manager building blocks for execution
    - Supports Rate Controls:
      Concurrency / Error thresholds
- Target nodes using tags
- Target EC2 Instances or Windows nodes running anywhere

© 2019, Amazon Web Services, Inc. or its Affiliates.

aws

High level workflow

1. If “on-premises”, configure Managed Instances
2. Compile DSC Configurations to MOFs as normal
3. Upload MOFs to S3 Bucket
4. If private repository, upload PowerShell Modules to S3 Bucket
5. Execute the “AWS-ApplyDSCMof” document against target nodes
6. Validate using AWS Systems Manager Compliance or JSON reports in S3

© 2019, Amazon Web Services, Inc. or its Affiliates.

aws

Demo: DSC Hello World

© 2019, Amazon Web Services, Inc. or its Affiliates.

aws

DSC Enhancements with AWS

© 2019, Amazon Web Services, Inc. or its Affiliates.

aws

DSC Enhancements with AWS

- Token substitution
- Multiple configurations
- Credential handling
- Reboot behaviors
- Compliance reporting
- PowerShell Module dependencies

© 2019, Amazon Web Services, Inc. or its Affiliates.

aws

Token substitution

- Provides “at-runtime” configuration data
- Standard DSC:
    - MOFs are “static”
    - MOFs are compiled “per node”
    - MOFs are not re-usable
    - Build machine must have access to all configuration data
- DSC on AWS:
    - MOFs are re-usable “artifacts”
    - MOFs support dynamic configuration data “at runtime”
    - Use the same MOFs in test and production

© 2019, Amazon Web Services, Inc. or its Affiliates.

aws

Tokens: Environment Variables

- Replace a string using Environment Variables: ‘{env:EnvironmentVariableName}’

```
Configuration WorkgroupJoin
{
    Import-DscResource -ModuleName ComputerManagementDsc

    Computer Computer
    {
        Name = '{env:ComputerName}'
        WorkGroupName = 'MyWorkgroup'
    }
}
```

© 2019, Amazon Web Services, Inc. or its Affiliates.

aws

Tokens: EC2 / Managed Instance Tags

- Replace a string using EC2 / Managed Instance Tags:
  ‘{tag:TagName}’

```
Configuration RenameComputer
{
    Import-DscResource -ModuleName ComputerManagementDsc

    Computer Computer
    {
        Name = '{tag:ComputerName}'
    }
}
```

© 2019, Amazon Web Services, Inc. or its Affiliates.

aws

Tokens: Parameter Store

- Replace a string using Parameter Store ‘{ssm:ParameterStoreName}’

```
Configuration WorkgroupJoin
{
    Import-DscResource -ModuleName ComputerManagementDsc

    Computer Computer
    {
        Name = '{env:ComputerName}'
        WorkGroupName = '{ssm:WorkgroupName}'
    }
}
```

© 2019, Amazon Web Services, Inc. or its Affiliates.

aws

Tokens: One-Box testing

- Replace a string using EC2 / Managed Instance Tags
  ‘{tagssm:TagParameterStoreName}’
- If the tag does not exist, replace a string using Parameter Store
- Allows “override” values to support one-box testing

```
Configuration WorkgroupJoin
{
    Import-DscResource -ModuleName ComputerManagementDsc

    Computer Computer
    {
        Name = '{env:ComputerName}'
        WorkGroupName = '{tagssm:WorkgroupName}'
    }
}
```

© 2019, Amazon Web Services, Inc. or its Affiliates.

aws

DSC Enhancements with AWS

- Token substitution
- Multiple configurations
- Credential handling
- Reboot behaviors
- Compliance reporting
- PowerShell Module dependencies

© 2019, Amazon Web Services, Inc. or its Affiliates.

aws

Multiple configurations with dependency ordering

1. Separate DSC Configurations into focused MOFs
2. Allows configuration re-use for common configurations
    1. “SecurityStandards.mof”
    2. “FileServerBase.mof”
3. MOFs are handled individually
4. MOFs order is the dependency chain

© 2019, Amazon Web Services, Inc. or its Affiliates.

aws

DSC Enhancements with AWS

- Token substitution
- Multiple configurations
- Credential handling
- Reboot behaviors
- Compliance reporting
- PowerShell Module dependencies

© 2019, Amazon Web Services, Inc. or its Affiliates.

aws

Credential Handling Overview

- Supports all PSCredential objects in DSC Configurations
- Supports two credential sources:
    - Parameter Store
    - Secrets Manager
- Credentials retrieved using API calls at run-time

- Benefits:
    - Centralized credential management
    - Certificate management no longer required
    - Credentials are never stored in on disk or in MOFs

© 2019, Amazon Web Services, Inc. or its Affiliates.

aws

Credential Handling – Saving to AWS

- Credential value stored is a string in JSON format

```
New-SECSecret -Name 'SecretName' -SecretString (ConvertTo-Json -InputObject @{
    Username = $Credential.UserName
    Password = $Credential.GetNetworkCredential().Password
} -Compress)
```

```
{"Username":"domain\\MyUser","Password":"MySuperSecretPassword1"}
```

© 2019, Amazon Web Services, Inc. or its Affiliates.

aws

Credential Handling – DSC Configuration

- Username is the “name” or “ARN” to the secret
    - Parameter Store or Secrets Manager
- Password is completely ignored.

```
# The "Password" is completely ignored by the implementation
$ss = ConvertTo-SecureString -String 'This is ignored!' -AsPlaintext -Force

# Credential referenced by name
$credential = [PSCredential]::New('DSCDemoUserAccount', $ss)

# Credential referenced by ARN
$user = 'arn:aws:secretsmanager:us-west-2:123456789012::secret:DSCDemoUserAccount'
$credentialArn = [PSCredential]::New($user, $ss)
```

© 2019, Amazon Web Services, Inc. or its Affiliates.

aws

Credential Handling - PSDscAllowPlainTextPassword

- Must specify “PSDscAllowPlainTextPassword” in ConfigurationData

```
$configData = @{
    AllNodes = @(
        @{
            NodeName = 'localhost'
            PSDscAllowPlainTextPassword = $true
        }
    )
}
MyConfiguration -ConfigurationData $configData
```

© 2019, Amazon Web Services, Inc. or its Affiliates.

aws

Credential Handling – MOF Content

- Compiled MOF does not include credential details

```
instance of MSFT_Credential as $MSFT_Credential1ref
{
UserName = "DSCDemoUserAccount";
Password = "This is ignored!";
};
```

© 2019, Amazon Web Services, Inc. or its Affiliates.

aws

DSC Enhancements with AWS

- Token substitution
- Multiple configurations
- Credential handling
- Reboot behaviors
- Compliance reporting
- PowerShell Module dependencies

© 2019, Amazon Web Services, Inc. or its Affiliates.

aws

Reboot Behavior

- Reboot options:
  - AfterMof – aggregates reboots after MOF execution
  - Immediately – as soon as the reboot flag is set
  - Never

- Pre-reboot Script
  - Allows taking actions prior to system restarts, such as…
    - Remove instances from Load Balancer
    - Perform clean service shutdowns

© 2019, Amazon Web Services, Inc. or its Affiliates.

aws

DSC Enhancements with AWS

- Token substitution
- Multiple configurations
- Credential handling
- Reboot behaviors
- Compliance reporting
- PowerShell Module dependencies

© 2019, Amazon Web Services, Inc. or its Affiliates.

aws

Compliance Reporting

- Integrated with Systems Manager Compliance

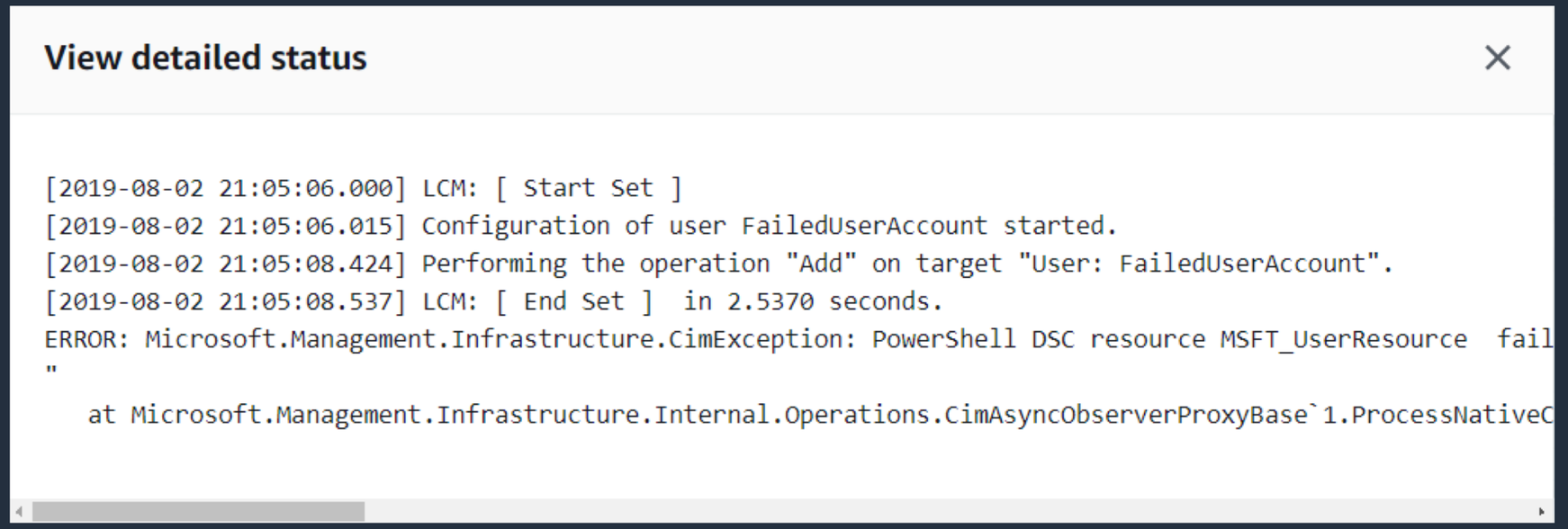
Compliance type	Compliant resources	Non-Compliant resources	Critical resources	High resources	Medium resources	Low resources	Informational resources	Unspecified resources
Association	2	0	0	0	0	0	0	0
Custom:DSC	2	0	0	0	0	0	0	0
Custom:DSCBadCompliance	0	1	1	0	0	0	0	0

© 2019, Amazon Web Services, Inc. or its Affiliates.

aws

Compliance Reporting

- Compliance console access to failed compliance output



© 2019, Amazon Web Services, Inc. or its Affiliates.

aws

JSON -> Status Overview to S3


```
{
  "ServicePath": "potr",
  "InstanceId": "i-0711a8b069ecc63f3",
  "ComputerName": "EC2AMAZ-G4BS711",
  "AppliedMofs": [
    "SecurityStandards",
    "FileServer"
  ],
  "Status": "Compliant",
  "Compliant": "2",
  "NotCompliant": "0",
  "SetCount": 1,
  "LastRunId": "56933368131036146-20190802-213306-Compliant-2-0",
  "LastRunUTC": "2019-33-02T21:33:06Z"
}
```


© 2019, Amazon Web Services, Inc. or its Affiliates.

aws

JSON -> Detailed Report to S3


```
"Name": "SecurityStandards",
"ResourceResults": [
  {
    "ResourceId": "[WindowsFeature]RemoveSMB1",
    "TestResult": {
      "VerboseMessages": [
        {↔},
        {
          "Timestamp": "2019-08-02 21:32:40.402",
          "Message": "The operation 'Get-WindowsFeature' started: FS-SMB1"
        },
        {↔},
        {↔}
      ],
      "Result": true
    },
    "SetRun": false,
    "Status": "Compliant"
  }
],
"SetCount": 0,
"Skipped": 0,
"Compliant": 1,
"NotCompliant": 0,
"Status": "Compliant"
```


© 2019, Amazon Web Services, Inc. or its Affiliates.

aws

DSC Enhancements with AWS

- Token substitution
- Multiple configurations
- Credential handling
- Reboot behaviors
- Compliance reporting
- PowerShell Module dependencies

© 2019, Amazon Web Services, Inc. or its Affiliates.

aws

PowerShell Module Dependencies

- Automatically detect required PowerShell Modules from the MOF
- Automatically installed from either:
    - PowerShell Gallery
    - S3 Bucket
        - S3 Object using the same zip format as on-premises Pull Server:

          ModuleName_ModuleVersion.zip

© 2019, Amazon Web Services, Inc. or its Affiliates.

aws

Demo: DSC Enhancements with AWS

© 2019, Amazon Web Services, Inc. or its Affiliates.

aws

Recap

- AWS Systems Manager has supported Desired State Configuration since November 2018
- DSC usability enhancements
- Supports EC2 Instances or on-premises machines
- Rich compliance in Systems Manager Compliance or JSON reports

© 2019, Amazon Web Services, Inc. or its Affiliates.

aws

Useful links

- AWS Tools for PowerShell:
  - https://aws.amazon.com/powershell/

- DSC on AWS: AWS-ApplyDSCMofs:
  - https://docs.aws.amazon.com/systems-manager/latest/userguide/systems-manager-state-manager-using-mof-file.html
  - https://aws.amazon.com/blogs/mt/run-compliance-enforcement-and-view-compliant-and-non-compliant-instances-using-aws-systems-manager-and-powershell-dsc/

© 2019, Amazon Web Services, Inc. or its Affiliates.

aws

Thank you!

Andrew Pearce
@austoonz

© 2019, Amazon Web Services, Inc. or its Affiliates.

aws