Deploying and Managing SQL Server with dbatools

Anthony E. Nocentino (he/him)
Enterprise Architect
Centino Systems

Explore

Everything PASS Has To Offer

Free Resources Online

PASS.org

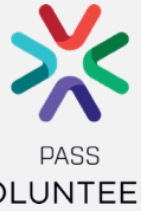
Unlock exclusive training & networking

Local user groups around the world

Free 1-day local training events

Back-to-back live webinar events

Online special interest user groups



Get involved

Anthony E. Nocentino

He/him

Enterprise Architect

Centino Systems

/nocentino

@nocentino

Specialize in system architecture and performance

Microsoft® Most Valuable Professional

PLURALSIGHT

Apress®

Agenda

- Deployment challenges
- Benefits of automation
- Automation solutions
- Using dbatools for automated deployment
  - Installing SQL Server
  - Configuring SQL Server
- Pester for managing configuration

Survey…

- How many of you
  - Have a SQL Installation checklist?
  - How many of you have logged into a server and found deviations from that standard'?

Anthony E. Nocentino
@nocentino

In your environment, do you have automated SQL Server installations?

If so, what are you using?

#ScientificTwitterPolling

Yea - automated	42.6%
No - Next, Next, Finish	**57.4%**

115 votes · Final results

Deployment Challenges…

- Consistency
- Fast
- Configuration skew

Benefits of automation

- Repeatable and consistent processes
- Speed
- Infrastructure as code
- Reduces human error (or increases it :)
- Scale out installations

Lesser known benefits of automation

- Measure configuration skew
- High availability
  - Restores can be simpler and automated
- Troubleshooting
  - If all the systems are the same…

Possible Solutions

- Configuration.ini
- PowerShell Desired State Configuration (DSC)
- Chef/Puppet/Ansible/Chocolatey
- Containers and Kubernetes
- dbatools - PowerShell Module

Using dbatools for automation

What is dbatools?

- Community driven PowerShell module
- Manage, configure and deploy SQL Server
- Command line SQL Server Management Studio

Getting dbatools

- PowerShell Gallery
- GitHub - https://github.com/sqlcollaborative/dbatools
- Chocolatey
- Offline install - https://dbatools.io/getting-started/

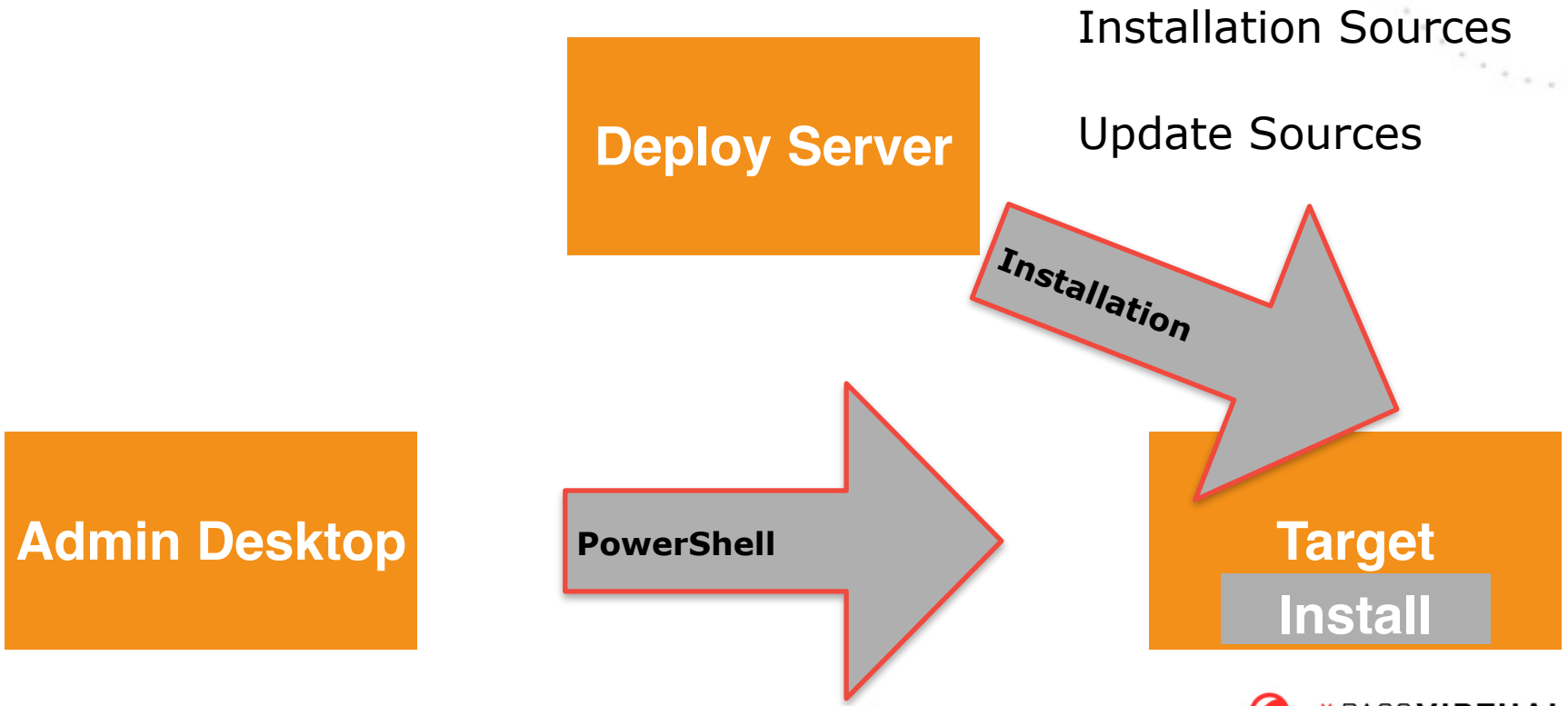
Core dbatools functionality

Availability Groups
Backup and Restore
Community Tools
Connection Strings
Databases
Data Masking
dbatools Computer Management
dbatools Configuration
dbatools Support tools
dbatools update watcher
DBCC
Detach and Attach
Diagnostics and Performance
Endpoints
Export
File System and Storage
FileStream
Finders
General
Log Shipping
Login and User Management
Mail and logging
Max Memory
Migration
Mirroring
Network and connectivity
Policy-Based Management
Registered Servers
Replication
Resource Governor
Security and Encryption
Server Management
Service Principal Names (SPNs)
Services
Snapshots
sp_configure
SQL Agent
SQL Client Configuration
SQL Management Objects
SSIS
System startup
tempdb
Data Masking
Traces, Profiler and Extended Events
Utilities
Windows Server Failover Cluster
Writing to SQL Tables

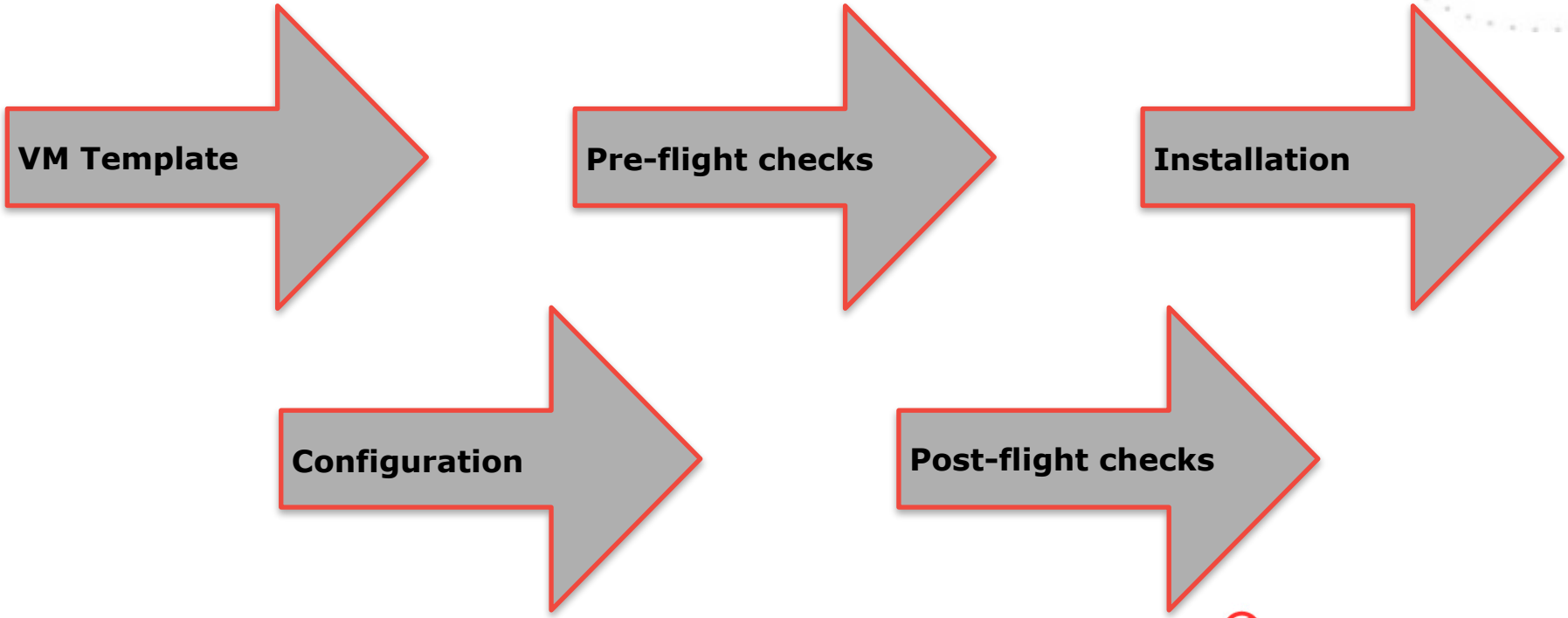
Installation cmdlet

- `Install-DbaInstance`
- Install SQL Server
  - Need an installation source files

Solution Architecture



Solution Workflow



Virtual Machine Template

- Configuration best practices
  - Disable vCPU Hot Plug, PVSCSI Network and storage adapters…
- Standardize drive topology
  - Volumes and folders
    - D:\DATA, T:\LOGS, S:\SYSTEM
    - NTFS Allocation Units
- Base OS settings best practices
  - Swap configuration

VMware on SQL Best Practices
https://bit.ly/31u0ntr

Pre-flight checks

- WinRM
- Service Accounts
- Installation Account
- Test drive topology
- Access to Installation share
- Access to Updates share

What's Pester

- Testing framework
- We'll use it to validate pre and post configuration
- Measure configuration skew
- Assert desired state
- What about dbachecks?
  - https://dbachecks.readthedocs.io/en/latest/

Pre-flight checks with Pester

```
Context "Server accessible via WinRM" {
    $result = Test-NetConnection -ComputerName $SqlInstance -InformationLevel Quiet -CommonTCPPort WINRM
    It "The target server should be accessible via WinRM" {
       $result | Should -BeTrue -Because "We need to do stuff with WinRM during the installation."
    }
}

Context "Service Account Validation" {
    $CredentialTestResult = Test-AdCredential -Credential $EngineCredential
    It "Testing to see if the Engine Service account credential is valid $($EngineCredential.Username): " {
        $CredentialTestResult | Should -BeTrue -Because "SQL Server requires a valid service account."
    }
}
```

Pre-flight checks with Pester

```
Executing script .\Test-PreInstallationChecks.ps1

  Describing Pre-Installation Checks

    Context Server accessible via WinRM
      [+] The target server should be accessible via WinRM 2ms

    Context Service Account Validation
      [+] Testing to see if the Engine Service account credential is valid LAB\SA-DBASQL1:  2ms

    Context Installation Account Validation
      [+] Testing to see if the installation account credential is valid LAB\aen:  2ms

    Context Testing for the existence of required drives on target
      [+] Should have a drive C 4ms
      [+] Should have a drive D 2ms
      [+] Should have a drive F 2ms
      [+] Should have a drive L 2ms
      [+] Should have a drive S 3ms
      [+] Should have a drive T 2ms
```

Install-DbaInstance

Installation Sources

Version

Features

Instance Path

File locations

Install-DbaInstance

Admin accounts

Feature Credentials

Advanced Configuration

Perform Volume Maintenance

Restart

Install-DbaInstance

```
$InstallationParameters = @{
    SqlInstance = $SqlInstance
    Path = $InstallationSources[$Version]
    Version = $Version
    Feature = $Features
    InstancePath = $InstancePath
    DataPath = $DataPath
    LogPath = $LogPath
    TempPath = $TempPath
    BackupPath = $BackupPath
    AdminAccount = $AdminAccount
    EngineCredential = $EngineCredential
    AgentCredential = $AgentCredential
    Credential = $InstallationCredential
    Configuration = $Configuration
    PerformVolumeMaintenanceTasks = $true
    Restart = $true
    Confirm = $false
    Verbose = $true
}
```

```
Install-DbaInstance @InstallationParameters
```

What about parameters that aren't exposed by the cmdlet?

- Custom installation options
- `-Configuration`
- UpdateSource - Enables patching during the installation process
- You can still use configuration.ini

https://docs.microsoft.com/en-us/sql/database-engine/install-windows/install-sql-server-from-the-command-prompt?#Install

DEMO

Pre-flight Checks

Installing SQL Server with `Install-DbaInstance`

Invoke-SqlConfigure

- Custom function
- Post installation configuration tasks
- Idempotent

Invoke-SqlConfigure

```
# Configure SQL instance
Invoke-SqlConfigure -SqlInstance $SqlInstance

function DisableSaLogin {
    Param(
        [Parameter(Mandatory = $True)]  [String] $SqlInstance,
        [String] $InstanceName = "MSSQLSERVER"
    )

    #Disable the sa login.
    Get-DbaLogin -SqlInstance "$SqlInstance\$InstanceName" |
            Where-Object { $_.Name -eq 'sa' } |
            Set-DbaLogin -Disable
}
```

DEMO

Configuring SQL Server with dbatools

Post-flight checks

- Services Started
- Accounts added or disabled
- SPNs Configured
- Instance Settings
- Agent Settings and Jobs
- Database Settings and Stored Procs

Post-flight checks

```
Executing script .\Test-PostInstallationChecks.ps1

  Describing SQL Agent Configuration

    Context DBASQL1: Testing to see if the SQL Server Agent Service is running
      [+] Testing to see if the SQL Server Agent Service is running 1.09s

    Context DBASQL1: SqlAgent Operator

    Context DBASQL1: Agent History Retention
      [+] DBASQL1: Should have a job history length set to 1000 per job 61ms
      [+] DBASQL1: Should have a job history length set to 10000 total 5ms

  Describing Ola Hallengren SP and Job Configuration

    Context DBASQL1: Test to see if Ola Hallengrens Maintenance Solution and if sp_whoisactive is installed
      [+] Testing for DatabaseBackup 21ms
      [+] Testing for DatabaseIntegrityCheck 4ms
      [+] Testing for IndexOptimize 3ms
      [+] Testing for CommandExecute 3ms
      [+] Testing for sp_WhoIsActive 3ms
```

DEMO

Managing configuration with Pester

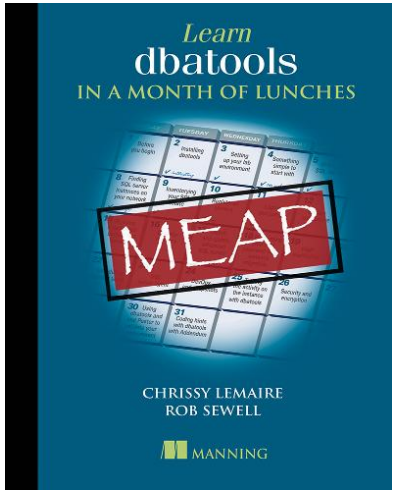
Agenda

- Deployment challenges
- Benefits of automation
- Automation solutions
- Using dbatools for automated deployment
  - Operating system templating
  - Installing SQL Server
  - Configuring SQL Server
- Pester for managing configuration

References and Resources

- dbatools.io
- dbatools.io/slack/

Books



The Dbatools Team and my Friends!

- **Chrissy** @cl
- **Kirill** @nvarscar
- **Rob** @sqldbawithbeard
- **Jess** @jpomfret
- **Claudio** @ClaudioESSilva
- **Sander** @SQLStad
- **Stuart** @napalmgram
- …and so many more

Thank you

Anthony Nocentino
he/him

@nocentino

aen@centinosystems.com

PASS VIRTUAL SUMMIT 2020