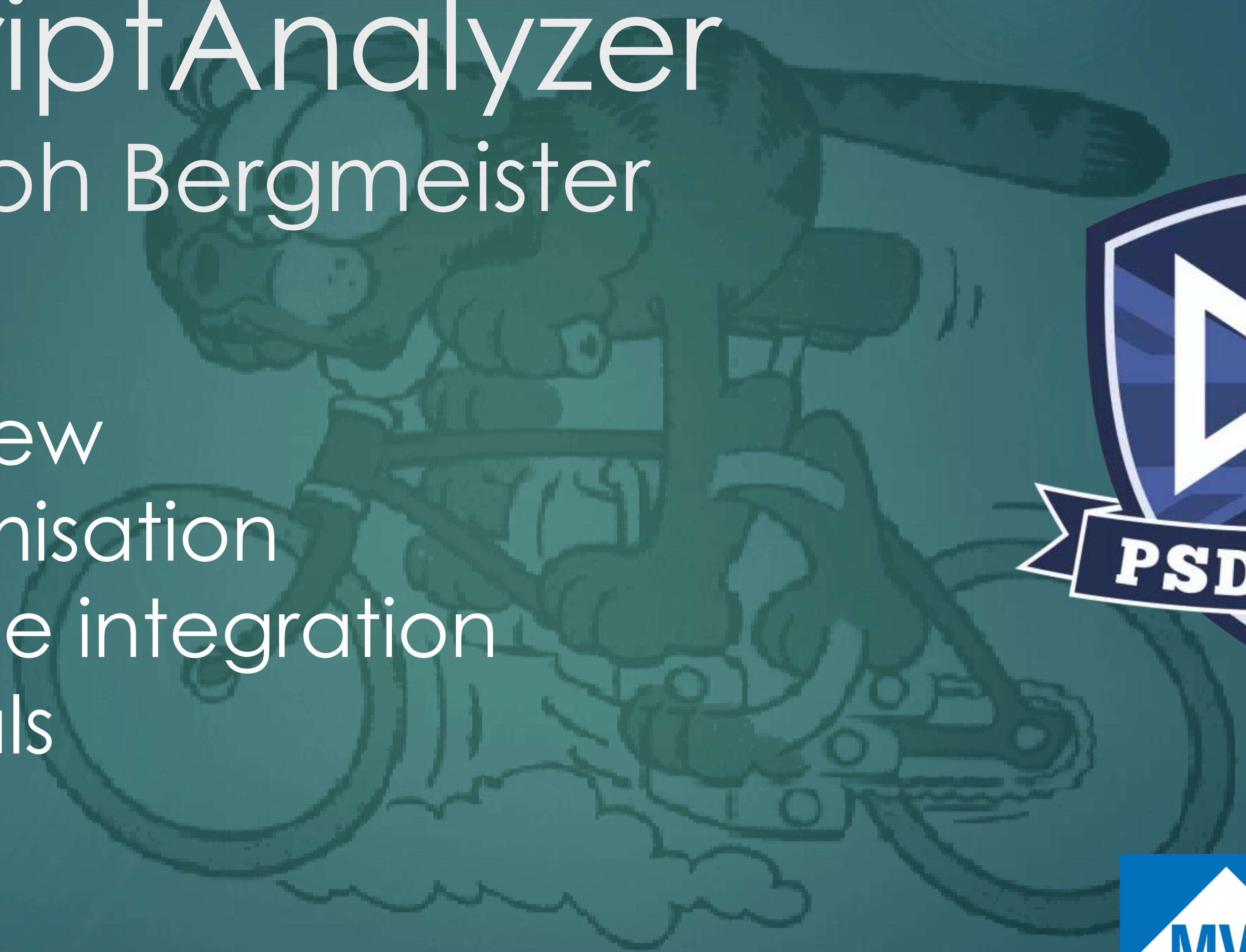
PSScriptAnalyzer

Christoph Bergmeister

- Overview
- Customisation
- VSCode integration
- Internals



PSDAY.UK

MVP Microsoft® Most Valuable Professional

Thank you

DevOpsGroup

System Frontier

eSynergySolutions

TECHSNIPS

PSDAY.UK

> whoami

- .Net Developer by day for Willis Towers Watson in Reigate insurance tech hub
- CI/CD/Automation/Testing
- PSScriptAnalyzer community maintainer
- PowerShell contributor
- Microsoft MVP

GitHub: https://github.com/bergmeister
Twitter: https://twitter.com/CBergmeister
Blog: https://bracketsandbraces.blogspot.com

Microsoft® MVP Most Valuable Professional

PSScriptAnalyzer

- Binary PowerShell module available from PSGallery
- Cross Platform
  - Windows
  - Linux
  - Mac
- Supported on PowerShell 3 – 6.1 and Docker
- Open Source (MIT)
- > 250,000 installations (current rate: 500 per day)

PSScriptAnalyzer Overview

Features:

- Code Analysis
- Formatter and Linter

Integrations:

- PowerShellEditorServices -> PowerShell VSCode extension
- PSCodeHealth

Extensions:

- Custom Rule support
  - InjectionHunter by Lee Holmes
  - Thomas Rayner's (ex MVP, now @MSFT) custom rules

Why use PSScriptAnalyser?

- Best Practices (from PowerShell team and community)
  - Cmdlet Design Rules
  - Security
  - Script Function usage
  - Script Analysis
  - Script Style
  - Consistent Formatting
  - Auto Format/Fix

- → Prevent and Find bugs early
- Better UX

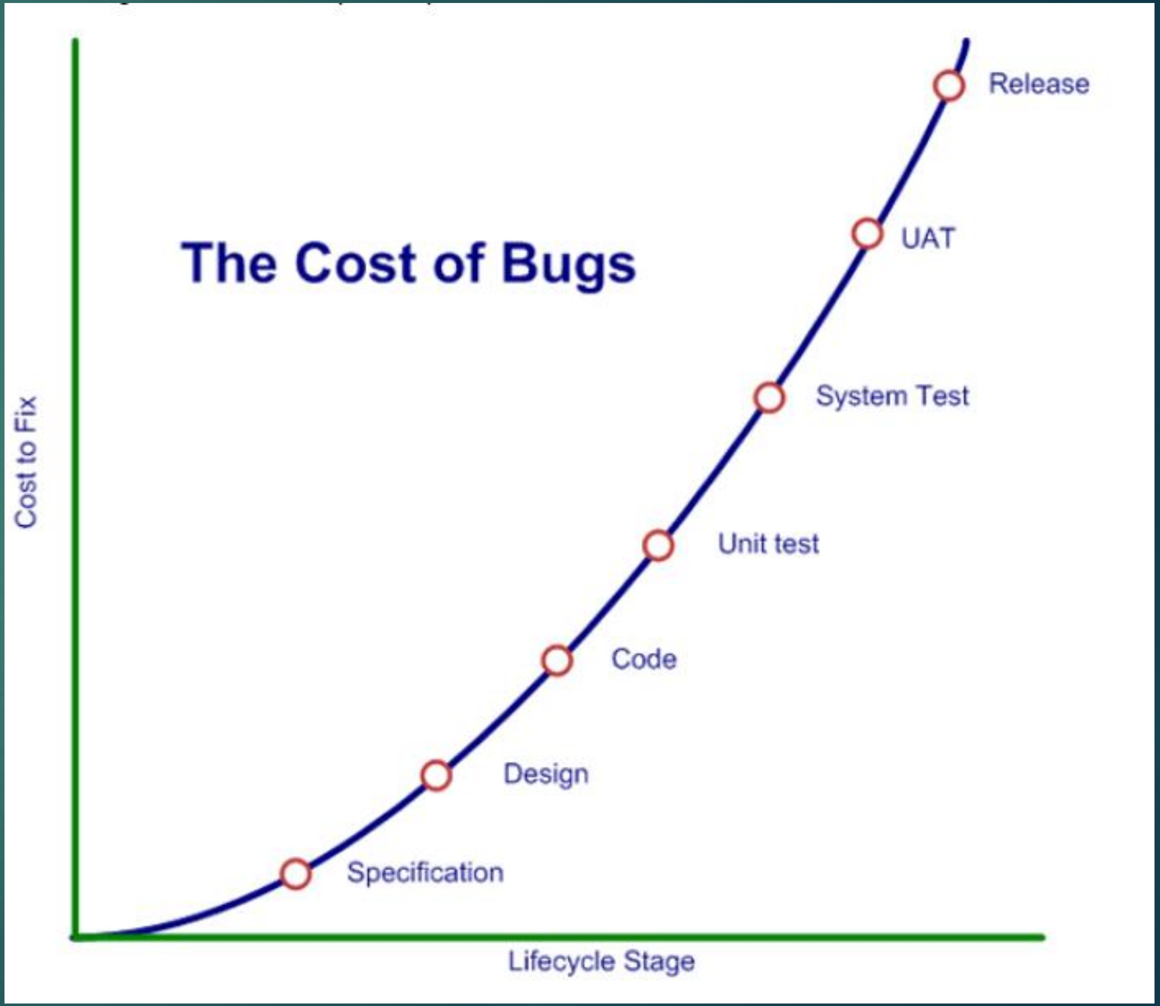


Image Source: https://www.experimentus.com/itm/W_06_00055_The_Cost_of_Defects.htm

> Get-Command –Module PSScriptAnalyzer -Syntax

- **Get-ScriptAnalyzerRule** [[-Name] <string[]>] [-CustomRulePath <string[]>] [-RecurseCustomRulePath] [-Severity <string[]>] [<CommonParameters>]

- **Invoke-Formatter** [-ScriptDefinition] <string> [[-Settings] <Object>] [[-Range] <int[]>] [<CommonParameters>]

- **Invoke-ScriptAnalyzer** [-Path] <string> [-CustomRulePath <string[]>] [-RecurseCustomRulePath] [-IncludeDefaultRules] [-ExcludeRule <string[]>] [-IncludeRule <string[]>] [-Severity <string[]>] [-Recurse] [-SuppressedOnly] [-Fix] [-EnableExit] [-Settings <Object>] [-SaveDscDependency] [-ReportSummary] [-WhatIf] [-Confirm] [<CommonParameters>]
- % for –ScriptDefinition parameter set

Demo Time :)

VS Code PSSA Settings

Ctrl+K+F to auto-format PowerShell Code in VSCode (uses Invoke-Formatter)

VS Code settings powershell.codeFormatting.*

- alignPropertyValuePairs
- ignoreOneLineBlock
- NewLineAfterCloseBrace, newLineAfterOpenBrace
- openBraceOnSameLine
- whitespaceAfterSeparator, whitespaceAroundOperator, whitespaceBeforeOpenBrace, whitespaceBeforeOpenParen
- **Preset**: Custom, Allman, OTBS, Stroustrup

  (can also be used from -Settings param; files are shipped)

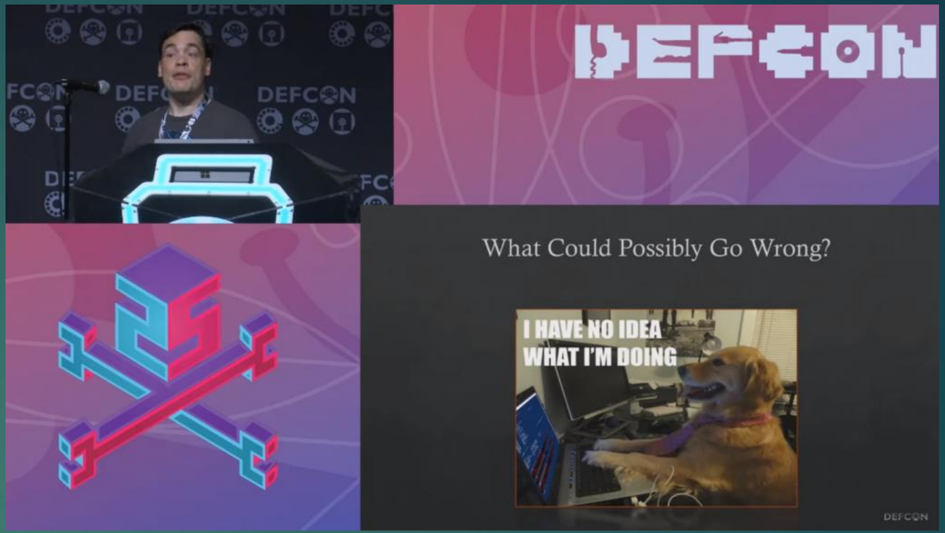
https://github.com/PoshCode/PowerShellPracticeAndStyle/issues/81

Custom Rules

Simple example:

https://github.com/bergmeister/PSScriptAnalyzer-VSCodeIntegration

Injection Hunter



Blog post from Lee Holmes:
https://blogs.msdn.microsoft.com/powershell/2018/08/03/powershell-injection-hunter-security-auditing-for-powershell-scripts/

Setup:

1. Save-Module InjectionHunter -Path .
2. Configure PSScriptAnalyzer Settings either via command line or VSCode

CustomRulePath = Path to Psd1 of InjectionHunter.psd1

PSScriptAnalyzer in CI

DIY example:

https://github.com/bergmeister/posh-profile/blob/master/test/psscriptanalyzer.tests.ps1

PSCodeHealth

By MathieuBuisson

Awesome docs: https://pscodehealth.readthedocs.io/en/latest/

PowerShell module on PSGallery

- https://www.powershellgallery.com/packages/PSCodeHealth
- https://github.com/MathieuBuisson/PSCodeHealth

Azure DevOps (VSTS) extension for build/release

- https://marketplace.visualstudio.com/items?itemName=MathieuBuisson.MB-PSCodeHealth-task
- https://github.com/MathieuBuisson/vsts-PSCodeHealth

Demo: https://christophbergmeister.visualstudio.com/PSCodeHealth%20Demo/PSCodeHealth%20Demo%20Team/_build/results?buildId=271

Internals: PSAvoidUsingCmdletsAliases Rule

- Checks for Explicit and Implicit aliases
- Why?
  - Readability
  - Platform availability of aliases in PSCore
  - Avoid 'Works on my machine'
    - -> Retrieves aliases from local session
      - The machine running PSSA matters

Internals: PSAvoidUsingCmdletsAliases Rule

Demo

- Define local Alias:

New-Alias foo gps; Invoke-ScriptAnalyzer -ScriptDefinition foo

- Explicit and Implicit aliases on different environments:
  - curl
  - service

Internals: Known issues

- PSUseDeclaredVarsMoreThanAssignment
  - SSA (static single assignment form) -> needed for +=
- Possibly outdated rules such as PSUseBOMForUnicodeEncodedFile
- Multi-Threading:
  - Locking issue
  - Not fully understood race condition can lead to error (has been observed on slow machines)

Internals:
How does it work?

- Show-PSAst module:

```
Show-Ast { if($a -eq "$b.wth()"){ }  }
```

Outlook PSScriptAnayzer 1.18.0

- Better tab completion (merged)
- Better performance for big scripts (merged)
- Bug fixes
- New command data files for 6.1 (PR open)
- Enhanced version/platform analysis (from MSFT)
- 2 new options for PSUseConsistentWhitespace (brace+pipe)

Thank you for your attention!

Questions?