Python 3 in Splunk

How PyDen will put the full power of Python in your stack

Jason Rauen | Booz Allen Hamilton

October 2019

.conf19

splunk>

Jason Rauen

Senior Lead Technologist | Booz Allen Hamilton

Badarsebard

Splunk Usergroup Slack

Forward-Looking Statements

During the course of this presentation, we may make forward-looking statements regarding future events or the expected performance of the company. We caution you that such statements reflect our current expectations and estimates based on factors currently known to us and that actual events or results could differ materially. For important factors that may cause actual results to differ from those contained in our forward-looking statements, please review our filings with the SEC.

The forward-looking statements made in this presentation are being made as of the time and date of its live presentation. If reviewed after its live presentation, this presentation may not contain current or accurate information. We do not assume any obligation to update any forward-looking statements we may make. In addition, any information about our roadmap outlines our general product direction and is subject to change at any time without notice. It is for informational purposes only and shall not be incorporated into any contract or other commitment. Splunk undertakes no obligation either to develop the features or functionality described or to include any such feature or functionality in a future release.

Extending Splunk with Python

Be the Splunk developer you wish to see in the world

.conf19

splunk>

Python Primer

A brief history of Python

- Created in 1991 by Guido van Rossum
- Language properties:
  - Dynamically-typed
  - Garbage-collected
  - Multi-paradigm
    - Object-oriented
    - Functional
    - Procedural
- Interpreted at runtime

Python in Splunk

What, Where, and Why

- Splunk contains a built-in Python distribution compiled specifically for its use
- Splunk uses Python for a number of its web framework pieces as well as a way to extend SPL
- Currently runs the latest 2.7 version
- Splunk 8.0 will contain both 2.7 and 3.7
- Splunk 8.1 will remove 2.7

Python in Splunk

What, Where, and Why

- Located in $SPLUNK_HOME/lib/python2.7
- Running $SPLUNK_HOME/bin/splunk cmd python will execute the Python terminal using the environment variables Splunk needs
- This is how Splunk runs Python scripts without needing it installed to the system

The Problem with Python and Splunk

Relearning the history of Python

.conf19
splunk>

You get what you get

And that's all that you get

- The main value proposition of using Python inside of Splunk is the ability to *extend* the system
- Developers can create scripts and commands to make Splunk do things that haven't been thought of by Splunk, Inc. or aren't worth their time because it's too specific to an environment or not enough demand

You get what you get

And that's all that you get

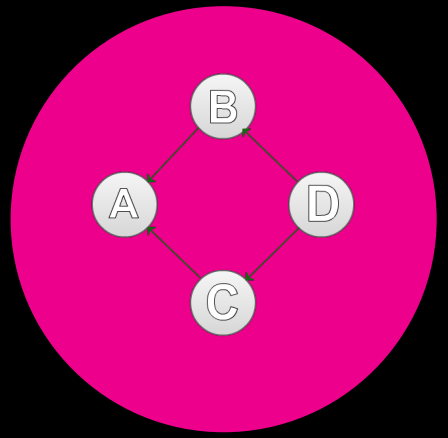
- However, the current implementation within Splunk means that the Python version is fixed to the version of Splunk and the packages included are only the ones Splunk deigns
- This causes three specific issues for developers looking to extend Splunk with Python

Splunk Python

The Unpythonic Python

Version Conflicts

Package Isolation

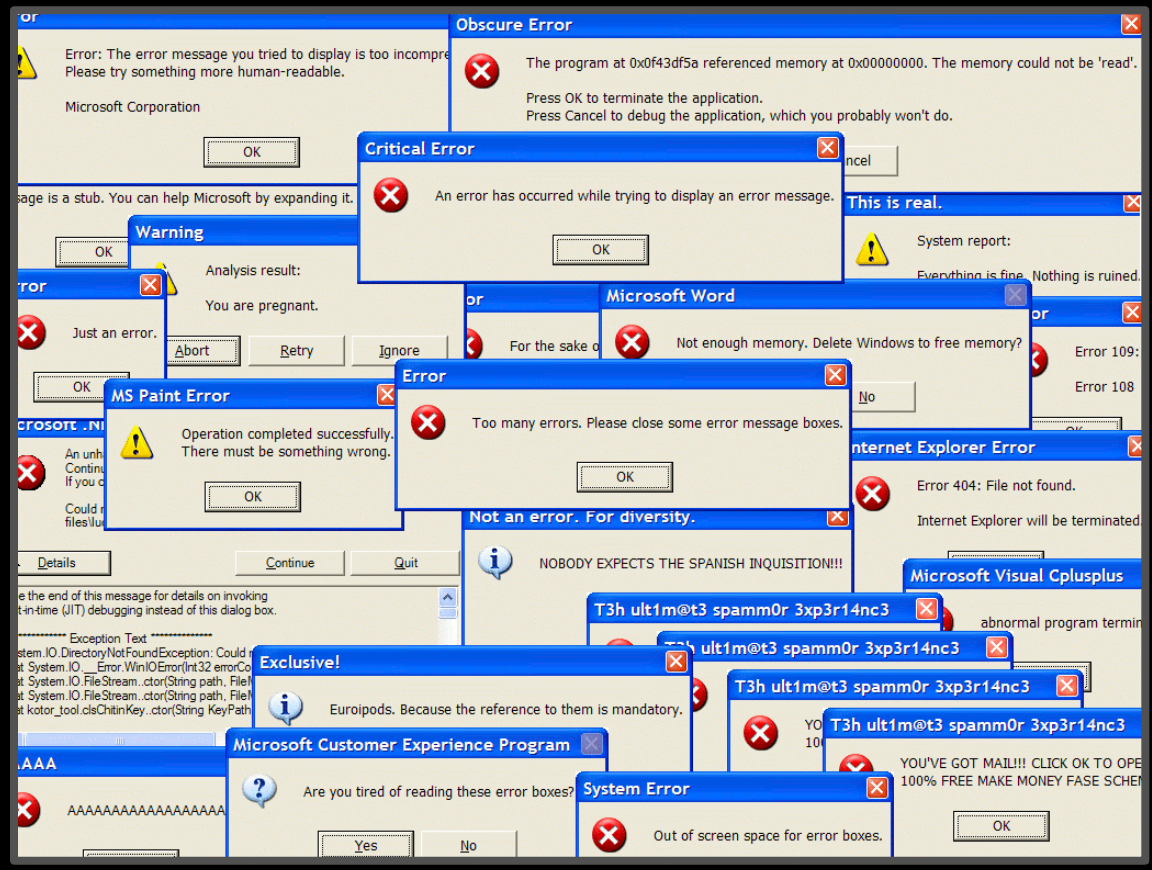


Dependency Management

Version Conflicts

The hidden pain coming soon

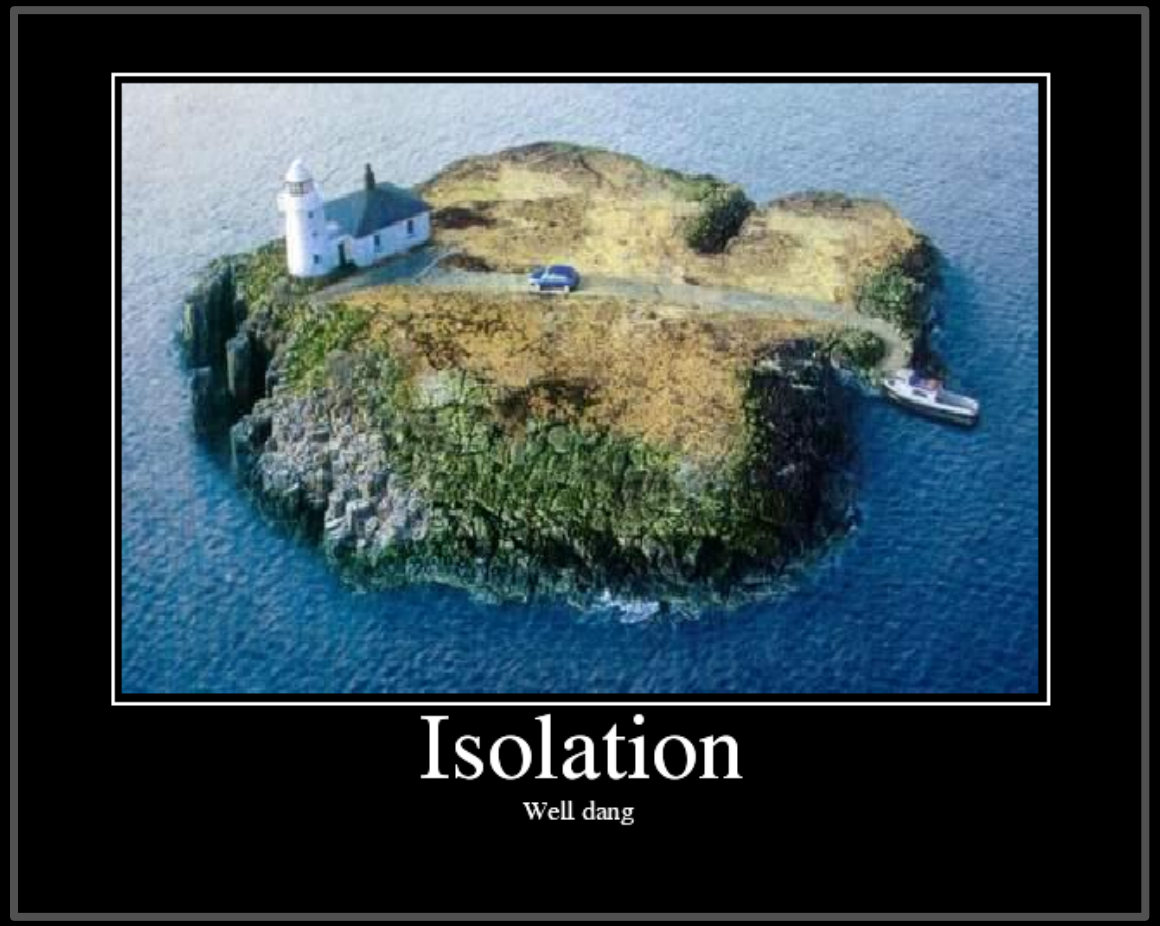
- The CPython interpreter for Python is under active development
- New versions are pushed regularly; micro versions every 6 months, minor versions every 18
- For years Splunk has run Python 2.7 and received only updates to the micro version
- Now with Splunk 8, we will need to be concerned with changes introduced in minor versions
- For developers, this means versioning apps to support multiple versions of Splunk may require refactoring your Python
- This means maintenance branches if you want new features of your apps to be supported on older versions of Splunk



Package Isolation

Why virtual environments exist

- When Splunk executes Python, a developer can include additional packages to support their script
- These are called through Python's import system
- However, conflicts can arise when two different apps include the same packages in different versions
- In pure Python development, this issue is resolved by virtual environments



Dependency Management

Pip-in is easy

- It was mentioned earlier that developers can include packages that aren't part of the Splunk library for use with their Python scripts
- The issue is, the supported way of doing this is to bundle the package into your app
- If the package you're trying to install is very complex, this is a literal nightmare
- One that was solved long ago with pip

Introducing PyDen

The Pythonic Python for Splunk

.conf19
splunk>

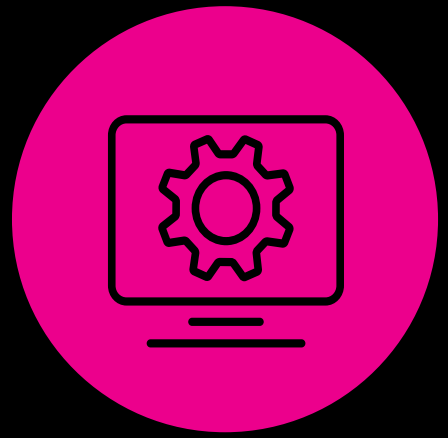
PyDen

And PyDen accessories

- The PyDen Suite is a set of Splunk apps used to provide three benefits to developers:
  - Version Choice
  - Environment Isolation
  - Simplified Package Management
- It consists of two apps:
  - PyDen
  - PyDen Manager

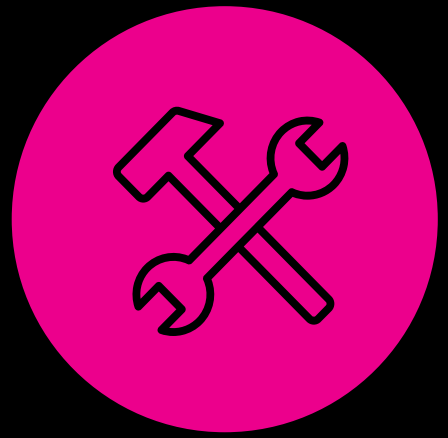
PyDen Workflow

From start to finish



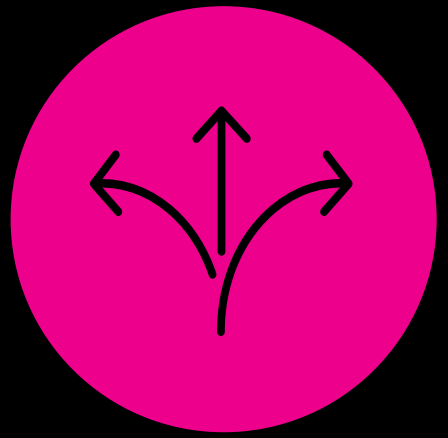
Develop

Specify requirements and write scripts that use them



Build

Compile Python and build the environments for the target deployment



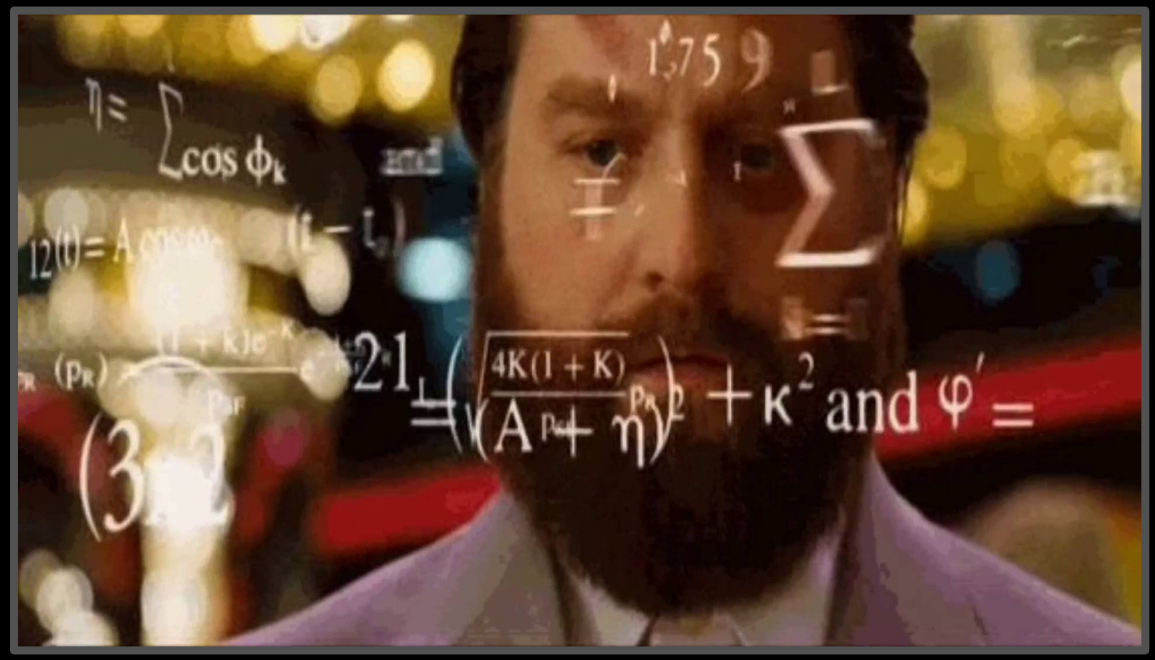
Deploy

Push the PyDen app to the Splunk servers that need it for execution

PyDen Development

A developer workflow

- Splunk app developers who want to use PyDen to leverage their choice of version and packages need to do two things:
  - Create a list of requirements:
    - Python Version
    - Virtual Environment Name
    - Required Packages
  - Activate the specified environment in their script



Activating Environments

The not so secret sauce

- PyDen contains example code for activating a virtual environment
- This code should be included in the developer's app
- Any script needing to utilize a PyDen virtual environment simply needs to include an import statement at the top of the script that references the name of the virtual environment

PyDen Manager

Build baby, build

- PyDen Manager is the main interface for the Splunk administrator
- This app contains interfaces for:
  - Downloading and compiling specific versions of Python
  - Creating virtual environments based on those versions
  - Installing packages to the environments through pip

PyDen

Run baby, run

- After the PyDen Manager has completed the build and installation process, the PyDen app will contain everything Splunk needs to use them
- The PyDen app is then deployed onto the search heads and indexers to be used by the scripts that require it using standard app deployment methods like cluster master and/or deployer

Demo

PyDen Roadmap

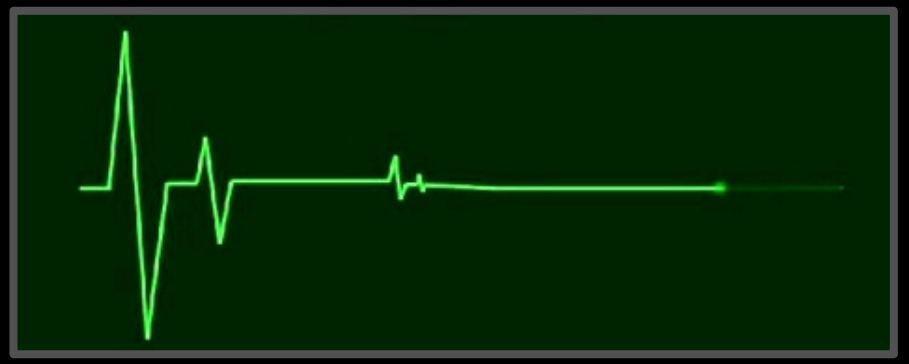
Splunk 8 and Beyond the Infinite

.conf19
splunk>

Splunk 8 and Python 2.7 EOL

Not a legacy workaround

- PyDen is not intended to provide support for Python 2 in Splunk after EOL
- PyDen will maintain the ability to compile and use Python 2 for versions of Splunk that utilize it
- Once Splunk removes Python 2, so will PyDen
- This will not help you keep from refactoring



Upcoming Features

Spoilers

- Requirements file capability
- Virtual environment activation through configuration, instead of hard coding
- Modular input to automate build step

Long Range Goals

Spoilers

- Splunk Cloud approval
- Windows support
- Integration with other Splunk solutions
  - MLTK
  - Add-On Builder
- Remove the need for two apps

Q&A

Jason Rauen | Senior Lead Technologist

.conf19
splunk>

splunk>

Thank You!

Go to the .conf19 mobile app to

RATE THIS SESSION