# AutoPkg for Windows Software

James Stewart
Systems Administrator @ UCB
https://twitter.com/jgstew

Matt Hansen
Systems Administrator @ PSU
https://twitter.com/hansen_m

## Shared Notes Here

**Session Feedback:** http://j.mp/psumac2015-140

# Overview

- BigFix / IBM Endpoint Manager
- How to run AutoPkg on Windows?
- What is AutoPkg?
- What is needed?

- Windows Related Processors
  - **WinInstallerExtractor**
  - **HachoirMetaDataProvider**
  - **MSIVersionProvider**
  - Questions?
- More Details, Demos, etc...

# BigFix

# Endpoint Manager

**Works on Windows, Mac, and Linux**

- Deploy Software
- Update Software
- Configuration Management

# AutoPkg Stats - Penn State

15 Windows Recipes (Since November 2014)

70 Windows Deployment Tasks

~1000 Mac Deployment Tasks

# How to run AutoPkg on Windows?

**AutoPkg DOES NOT run on Windows!**

AutoPkg is written in Python.

AutoPkg has some OS X specific code.

AutoPkg could be cross platform in the future.

**https://github.com/autopkg/autopkg/pull/169**

# What is AutoPkg?

The goal of AutoPkg is to automate the process of readying new software for deployment.

AutoPkg automatically:

- detects new software releases
- downloads the new software
- prepares software for deployment (if needed)
  - get metadata
  - transform
  - repackage
- puts software into your deployment solution

# What is needed?

Primarily:

- a Deployment System with an API
  - Munki, Casper, Absolute, SCCM, BigFix, …
- a Download URL for the software
  - technically, things other than URLs could be used

Secondarily:

- The Product Name (supplied)
- The Product Version (the hard part)
- Any required command line parameters

# Finding Windows Product Versions on OS X

If the version can be determined from the download URL, then no further work is needed.

**http://download.royalapplications.com/RoyalTS/RoyalTSInstaller_3.00.01.60226.msi**

**Challenges:**

- OS X cannot natively read MSI file properties
- OS X cannot natively read EXE file properties
- OS X cannot extract most Windows archives

# WinInstallerExtractor.py

Unpack the installer using **7zip** and read the version number from the resulting files.

```
7z e -oExeExtract iTunes.exe
```

# WinInstallerExtractor.py

Input Variables:

- **exe_path** - Path to exe or msi, defaults to %pathname%"
- **preserve_paths** - eXtract archive with full paths, defaults to 'True'
- **extract_dir** - Output path for the extracted archive

- Running with -vv displays full 7z output

# Get Version info from EXE files

**Issue:** OS X cannot read metadata from EXEs

**Solution:** hachoir-metadata python library

```
python file_meta_data.py BoxSyncSetup.exe
```

**Metadata:**

- Title: Box Sync
- Author: Box Inc.
- **Version: 4.0.6442.0**
- Creation date: 2012-12-24 21:43:11
- Copyright: Copyright (c) Box Inc.. All rights reserved.
- Comment: CPU: Intel 80386

# HachoirMetaDataProvider.py

AutoPkg Processor that makes use of the Hachoir library to get version info from EXEs.

```
ufilepath = hachoir_core.cmd_line.unicodeFilename(str(filepath))
parser = hachoir_parser.createParser(ufilepath, filepath)
metadata = hachoir_metadata.extractMetadata(parser)
```

# HachoirMetaDataProvider.py

Input Variables:

- **file_path** - Path to the file, defaults to % pathname%"
- **metadata_key** - key to return the value of, defaults to ‘Version’
- **metadata_index** - Value index, if multiple values exist, defaults to 0
- **output_var_name** - Output variable, defaults to %metadata_key% or 'version'

# MSIVersionProvider.py

MSIs are Microsoft's proprietary package format

Most Windows Software installs using MSIs

Reading MSI's on OS X turns out to be especially difficult

**Took a lot of research to find a cross platform solution:**

- Python Library? Windows Only
- Cross platform programs?  404 : Not Found
- WINE? Can only install MSIs, but not export values.
  - There might be a way to read MSI values using WINE code, but not that could be easily determined

# MSIVersionProvider.py

**What is the solution?** (a combination)

- Find a windows program that:
  - has a command line interface
  - will export the ProductVersion from the MSI
  - runs on OS X under WINE

**msi2xml -** first attempt

- worked, but complicated
- could be useful if other msi details are needed

# MSIVersionProvider.py

**lessmsi** - the chosen option

wine lessmsi.exe v msi2xml-2.2.1.957.msi

# MSIVersionProvider.py

Input Variables:

- **msi_path -** Path to the .msi, defaults to %pathname%

Output Variables:

- **version -** Version of %msi_path%

# Any questions so far?

Session Feedback: http://j.mp/psumac2015-140

# Why use AutoPkg anyway?

Community of shared recipes

- Windows Download recipes are similar to existing Mac Download recipes for the same product

Community of shared processors

Why recreate AutoPkg when it already exists?

# Why not use Chocolatey?

# Making AutoPkg work on Windows

AutoPkg will now run with warnings

The only functionality that works is git search

It will take some effort to do this.

Please help if you can.

**https://github.com/autopkg/autopkg/pull/169**

# Demos

**Questions?**

Session Feedback:
http://j.mp/psumac2015-140