Procedures for Automated Enterprise Architecture

Patrick Stingley

[covertchannel@yahoo.com](mailto:Pstingley@blm.gov)

(240) 475-8400

I just rewrote the Python programs that process the data again this month and renamed the program ‘The\_Mill.py” . This was done for a number of reasons:

1. The program was rewritten to be used by other organizations outside of BLM, so it was simplified.
2. OMB M-21-30 has a requirement to be able to identify the operating systems in use by the organization. Restructuring the python program allowed the same program to be used to process the applications for the TRM and for the data from it for M-21-30, but it also allowed the operating system data to be processed by the same program.
3. The previous programs created interim files with the same name, so if the python program was used more than once it overwrote the previous interim files.

Since the product/version are not in the same field with this version, it was necessary to create a completely new taxonomy tab that correlated the product name with the appropriate FEAF category. The taxonomy tab in the worksheet has all of the most current edits, not the separate taxonomy spreadsheet, so remember to use the sheet in the template file.

Getting started

The first thing I do is create a new directory, such as /TRM/January 2021 and put the following files in it:

TRM Template June 2021.xlsx

The\_Mill.py

The Python program has been written with simplicity in mind. It has been designed such that a person who is not good at Python can maintain it. There are a lot of if statements that replace various things found in the inventory. These could be simplified, but I found that the program actually took longer to run when the preceding ifs were removed, so I left them.

Next, I gather the data. To do this, you will need to be on the BLM network (or VPNed in). The data is not Internet accessible. All of these reports work better in Internet Explorer.

Getting the BFI Data. Go to <https://bfi.doi.net>:9081/ . (It was moved this month to this port address). It uses SAML to allow you in. Once you’re in you will see a dashboard. Under the Reports pull-down, select Package Data Report. Click on the “(All Data)” link on the right. Uncheck all of the boxes except (under Package Data) Name and Version. BigFix will save the data into the directory you want, so I put it into this working directory. BigFix will provide a unique name to the file, such as “package-data-20210623184811.csv”

I also collect a copy of everything, including the PC the software was loaded on and the software manufacturer. I saved the file in the working directory using the default name, which is something like: package-data-20210114155447.csv I find that these full inventories with hostnames and software manufacturers, are useful tools to use in analyzing things, but it is always over 200MB and much too big for Excel.

The BFI-Inventory.csv file was 69M this month, which seems to be about typical. I find that the numbers of things BigFix finds fluctuate from month to month. Sometimes it is because the buffers have been purged and haven’t completely recovered, but often there is no apparent reason for the fluctuation.

It takes quite a while to download these files completely.

Typically, the inventory dump is 1.4 – 1.6 million lines deep (today it was 1,540,766), which is too much for Excel, so you’ll have to pre-process it with the Python program that sorts it and counts it.

Process the data

Open a command window and go to the current month’s directory.

Issue the command: The\_Mill.py package-data-20210623184811.csv

You’re literally running the data through The Mill. The\_MIll will process the data file and create a number of interim files. It will create one once the data has been cleaned up. It will create one with the data sorted, and it will create a final one once the data has been counted. Each interim file is available for your use. I find them helpful in diagnosing problems.

This month it took 2 minute and 51 seconds to process the data.

Open the resulting counted-sorted-scrubbed-<XXXX.csv>.txt file in notepad. Today, BigFix found 1,489,768 unique products on the network representing 37,515 unique types of products.

Open the file named TRM Template 40-20-2020.xlsx and save it with the current date.

Go to the Products tab, select all and clear contents. This will clean out the old data so it is ready for new data. Open the Counted-Sorted-Scrubbed file in Notepad. Select All. Copy, then go to cell A1 and paste the contents into the Products sheet in the Template file. This is the quickest and generally best way to insert the data into the TRM worksheet.

The macros in the TRM-Index spreadsheet will automatically align everything in the Products tab with the FEAF.

Next, clear the contents of the TRM tab then copy the entire contents of the TRM-Index sheet and paste it over the TRM tab (As LITERAL VALUES!)

Save!

Posting the TRM data as literal values allows the users to sort the data.

Sort the data based on the FEAF Code field to find the new products that have yet to be catalogued. This month there are 2,362 new products that will require cataloguing.

You will find many files under the categories of virus, policy violations, gaming and etc. This is because there are no categories in the FEAF for these kinds of programs, but a complete inventory of the products on our network will bring them to light.

TBM

In the template worksheet, there is a tab named TBM Taxonomy. As might be surmised from its name, this contains the TBM taxonomy.

While all of the software falls under software in the Cost Pools and by definition is somewhere within the IT Towers, many products on our network have yet to be mapped to the IT Sub Towers. We are not alone. I have been working with the government leads at GSA and with the TBM council and the services these products perform simply have not been aligned yet.

There is a tab titled TRM – TBM Crosswalk that shows the correspondences between the FEAF and the TBM taxonomies.

Macros in the TBM Enabled TRM tab will take the products posted in the TRM-Index tab and align them with the TBM based on the correspondences found in the TRM – TBM Crosswalk table.

That’s it! That’s as far as it is possible to go. Not everything that exists in the FEAF framework has mappings to the TBM framework.

SCCM Data

To get a report of all software on each computer, go to the following link:

<http://ilmocop3sm1/Reports/report/ConfigMgr_OC1/Software%20-%20Companies%20and%20Products/Count%20of%20all%20instances%20of%20software%20registered%20with%20Add%20or%20Remove%20Programs>

In the collection text box, choose the entry titled “All Systems”. This gives a list of all of the registered software on all of the computers. Choose the Excel spreadsheet format. Click the floppy disk icon near the top of the screen to download the .xlsx file with this data. Copy it into the month’s TRM directory.

Belarc

Open <https://belmanage.blm.doi.net/BelManage> in Internet Explorer.

If you have problems opening it (as I often do) send an e-mail to Mike Gascon (mgascon@blm.gov) to let you in.

Under Software, choose the option titled “All Software Versions” Then in the upper right, choose Excel and download the spreadsheet with all of the software versions found on BLM computers.

Open the spreadsheet in Excel.

Copy the spreadsheet to one to edit.

Delete the top 10 lines

Delete column B

Copy the entire spreadsheet using Ctrl+C

Paste into Notepad

Copy the entire document from Notepad

Paste into a new, blank, spreadsheet

Sort by column A

Go to bottom and delete all of the lines that are company names, starting from the botto

Delete Columns C & D

What remains is the count, product and version.

Go to Column D and put in the following macro: =concatenate(b3," ",c3)

Copy this down the entire column

Copy the resultant column over column B as literal values

Delete columns C and D

Now you are ready to compare against the other data sources.

IEMS/Belarc/SCCM

Last month, I took the inventory of products that IEMS (BigFix) found and compared it to the inventory of products from Belarc. What I found was alarming. Both inventory systems missed large numbers of products on the network. I was more disturbed that IEMS is missing things than that Belarc is, because while Belarc is merely a convenient tool for the support staff, IEMS is supposed to be sending the entire inventory of applications to the Continuous Diagnostics and Monitoring program at DHS, where these inventories are compared against known good products, looking for products that have been infected by viruses. If IEMS can’t see these products, that means that DHS isn’t seeing them either, which further means that the security the Government is counting on isn’t working.

As a result of this finding, I expanded my research to evaluate how Microsoft’s SCCM tool compares to these other two tools. What I found was that all three missed huge amounts of software. I created a spreadsheet comparing the data from these three tools earlier in the month. I will use the data from this TRM to validate my results by seeing if I find the same inconsistencies (especially the gaps in IEMS). Towards that end I am writing a Python program to show which products each inventory system collects. (Last month I did this by hand and sampled 1500 of the 10600 products on BLM networks. It was a very time consuming process.) This month I hope to compare the inventories in their entireties.