CVE Binary Tool: Add Windows Support for CVE-Bin-Tool

Note: Make sure to include the sub-org name in the title both in Google's system and in your document.

About me

1. Ziao Wang (github name: wzao1515)
2. Columbia University / first-year Master in computer science / graduate in 12/2019.
3. email: zw2498@columbia.edu phone: 347-556-6872
4. Time zone: EDT
5. resume: https://ziaowang.files.wordpress.com/2019/03/ziao*wang*resume.pdf

Code contribution

1. Create new test case: https://github.com/intel/cve-bin-tool/pull/70
2. Enhance NVD data for curl: https://github.com/intel/cve-bin-tool/pull/74
3. Add a new checker: https://github.com/intel/cve-bin-tool/pull/77
4. refactor exist code: https://github.com/intel/cve-bin-tool/pull/78
5. implement linux bash command by Python: https://github.com/intel/cve-bin-tool/pull/86, https://github.com/intel/cve-bin-tool/pull/94
6. Create new rpm test case: https://github.com/intel/cve-bin-tool/pull/108, https://github.com/intel/cve-bin-tool/pull/113

Project information

1. Sub-org name: CVE Binary Tool
2. Project Abstract

CVE Binary Tool is running on Linux systems now by taking advantages of bash commands like `file` and `string`. Since `file` and `string` have already been naively implemented, it is ideal to extend the tool to other operating systems like Windows. In addition, in order to achieve better performance,

another goal is to rewrite those implementations in C using Python/C API and implement multi-thread (multi-process) scanning.

1. Detailed description

The CVE Binary Tool was designed for use on Linux, and thus makes assumptions about the availability of command line utilities, but it doesn't have to be that way. The two utilities it uses for parsing files are file (gives you file type information) and strings (gives you a list of strings found in a given binary). These can be written in pure python, allowing the CVE Binary Tool to be architecture independent.

The CVE Binary Tool also uses a number of system utilties for extracting files from various archive formats like apk and zip. These utilities may also have different names on different platforms. Investigate how to deal with those more smoothly. It's possible this could also be done in pure python, we could use utilities that are platform specific and do appropriate checks to make sure they're installed or suggest them to the user.

To extend this tool to multi platforms, we cannot use `subprocess.call` because Python/C API provides an interface for users to write extension modules in C/C++. API functions have one or more arguments as well as a return value of type `PyBbject*`. This is a pointer represents a Python object. Since error checking in C always has to be explicit, handling exceptions and debugging would be a potential challenge in this project.

1. Weekly timeline

- **Community Bonding** (May 7-26): Create make-up test cases (binary files) on Windows.
- **Week 1** (May 27-31): Create make-up test cases on Windows.
- **Week 2** (June 3): Rewrite the code in `file` and `string` in C. the goal is to make it more efficient.
- **Week 3** (June 10): There are five types of extractions, so I would spend one week for each. This week I will implement extracting `tar` files.
- **Week 4** (June 17): Implement extracting `rpm` files in Python.
- **Week 5** (June 24): Implement extracting `deb` files in Python.
- **Week 6** (July 1): Implement extracting `zip` files in Python.
- **Week 7** (July 8): Implement extracting `cab` files in Python.
- **Week 8** (July 15): Implement multi-thread modules for scanning in Python.
- **Week 9** (July 22): Refactor the code to make it more readable.
- **Week 10** (July 29): Evaluate running time overhead.

- **Week 11** (August 5): Start writing documents for what I have done as well as using instructions on Windows.
- **Week 12** (August 12): Finish documentation and make sure everything is all set.
- **Final week** (August 19): Submitting project.

Other commitments

I don't have other things that could affect my ability to work this summer, and Python CVE Binary Tool is the only organization I am applying for.