Application Security

Assignment 1 - Part 1 (Turing Complete Sandbox)

Pankaj Moolrajani - pm2306@nyu.edu

1. File Formats

a. Programming language used for implementation of sandbox - python 2.7.6

b. File Details:
   i. sandbox.py - main file to load sandbox
   ii. operations.py - contains example programs of computation
   iii. security.py - checks if exploit code is secure or not to execute in sandbox

2. Thinking

a. Following restrictions and policies are required to implemented turing complete sandbox:

   i. Limit resource usage to avoid memory leak and segmentation faults. For this python module 'resource' is used to limit address space, cpu time, number of file objects handled and number of process created by untrusted code.
      1. Source File: sandbox.py
      2. Class: Restrictions
      3. Method: restrictResources()

   ii. Blacklist system related python modules to prevent execution of system commands. On importing these modules in sandbox interpreter or exploit-code.py, exception will be raised. Use sys module and pip module to set all external libraries and blacklisted modules to None.
      1. Source File: sandbox.py
      2. Class: Restrictions
      3. Method: restrictImport()

   iii. To avoid system unavailability due to large file size, check filesize of exploit code.
      1. Source File: security.py
      2. Class: Security
      3. Method: fileSize()

   iv. To avoid exploit code from jumping outside working directory while giving exploit code filepath, check file path if it's in same working directory or not. Only file names are allowed, absolute paths are denied.

        1. Source File: security.py
        2. Class: Security
        3. Method: filepath()

    v. To avoid execution of malicious code, check for untrusted keywords and functions in exploit code provided.
        1. Source File: security.py
        2. Class: Security
        3. Method: code()

3. Turing Complete

- a. Sandbox implemented in python is turing complete because it allows:
    - i. to read and write to address space
    - ii. mathematical calculations
    - iii. navigation in memory space

- b. Extra Credit
    - i. Sandbox can be loaded in interactive mode using *python -i sandbox* command, and provides sandboxed python interpreter with required resource limitations and security policies to ensure harmless execution of untrusted code.

4. Execution

- a. All execution related commands of sandbox are described in readme file