High-level Analysis of Sandboxes

Assignment 1, Part 2

Application Security CS-GY 4753

Prof. Justin Cappos

Author: Casey McGinley

Prof. Cappos – easytocode.py

Prof. Cappos' easytocode.py sandbox is written in Python. First, the entire program (written in a limited version of Python) is read in as a string, and then this string is run against a blacklist of "dangerous" words and characters; if a blacklist item is found, the program exits, otherwise it clears the namespace (apart from some constants) and then executes the code in this cleared namespace.

Prof. Cappos – potentiallyhackablesandbox.py

The potentiallyhackablesandbox.py sandbox is written in Python and the input programs are written in a limited Python dialect. Programs are read entirely into memory, the program string is checked for blacklisted words, and if none are found, the namespace is cleared and the program is extended in the cleared namespace.

Prof. Cappos – a-sandbox.py

The a-sandbox.py is written in Python, with input programs written in limited Python. The program is read into memory, compared against a whitelist of chars, executed in a minimal namespace, and finally the contents of the variable a are printed.

ceinfo – TuringTest

ceinfo's TuringTest sandbox is written in Java and accepts programs written using an entirely new instruction set. The Java sandbox only accepts this extremely minimal set of instructions and nothing else; in fact, I think it is so restricted as to not be Turing complete.

mramdass – sandbox.py

mramdass' sandbox.py is written in Python (v3.3+) and accepts programs written in a limited version of Python. The sandbox places a lock on its own source code, and then reads the program, checking it against a blacklist; if it passes, its executed in a partially cleared namespace.

fjm266 – sandbox.py

fjm266's sandbox.py is written in Python, with input programs written in Python. The sandbox just executes the input program in a partially cleared namespace.

kellender – turingComplete.c

kellender's turingComplete.c sandbox is written in C, with input programs written in C. This sandbox forks itself so that the child runs the program code; additionally, limits are placed on the address space size and the max file size the child can create is set to 0 to prevent file creation.

crimsonBeard – sandbox.py

crimsonBeard's sandbox.py is written in Python and takes programs written in a unique instruction set as input. The instruction set accepted only allows for the essentials (e.g. arithmetic, looping, assignment, etc.).

PankajMoolrajani – sandbox.py

PankajMoolrajani's sandbox.py is written in Python, with input programs written in limited Python. It blacklists certain words, sets resource limits (file size), and removes specific modules (e.g. subprocess, os, sys) before execution.