GitHub Link:
https://github.com/CallMeSteve/AppSec/tree/master/Assignment%201%20Sandbox

The candidate sandbox enforces restrictions on builtins. It whitelist all safe builtins to run untrusted code with limited functionality. Moreover, resource assigned to sandbox are also limited like number of file objects, number of processes, memory space etc.

Vulnerabilities:

The memory management is poor. Using attack-1.py program crashes and causes segmentation fault.

The candidate sandbox limits number of file objects, but this can be overcome. The trick is to close existing file object and open a new file. So total number of file objects will remain same.

The candidate sandbox allows sandbox users to view system related files also like "/etc/passwd". It should chroot jail the execution of untrusted code to sandbox directory

The candidate sandbox removes unsafe builtins but still keyword lambda is left out. Using lambda function, an attacker can import 'os' module and use it's functions. OS module can delete system files and do system level calls also.