ADITYA DASGUPTA

NONOHUP DOCUMENTATION

CS450 FALL ‘20

FRANCIS LEUNG

DESCRIPTION -

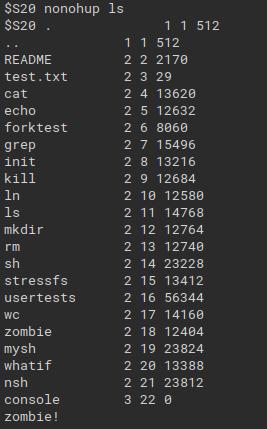
Calling a command or a program after nonohup causes the command or program to run in the background while returning the shell prompt back to the user.

USAGE -

nonohup prog arg

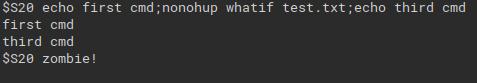
TEST CASES -

1) nonohup ls

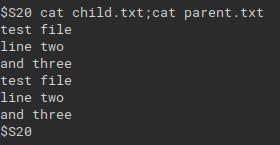


As expected, we get the shell prompt back while the command ‘ls’ is being run in the background.

2) echo first cmd;nonohup whatif test.txt;echo third cmd



On using the cat function, we can test if child.txt and parent.txt exist.



3) nonohup whatif

We get the prompt back immediately. Child.txt and parent.txt but have no values in them because the program was run in the background. CTL-D does not affect the whatif program anymore. Instead, it quits the custom shell.

KNOWN BUGS

Since nonohup does not reap the background child process, we encounter the ‘zombie!’ error processed by the zombie.c program.

DIFFERENCE FROM THE POSIX NOHUP FUNCTION

The nonohup function merely puts a process in the background. Furthermore, it does not even reap the zombie process the child process leaves behind. However, the nohup function runs a process in the background even when the parent shell process is stopped.