Assignment 4

Report for breaking users programs

Conducted the following tests and attempted to gain access to the secret files of various users.

1. Fake Config File

In this attack, the program is run from a directory different to the one in which it is meant to run. This allows the attacker to specify their own config file and trick the pogram into thinking it is the original config file. This is because the program looks for the files in the directory from which it is being run.

This attack only works if the programmer does not implement the full path name of the file.

2. Writing to the secret file

This can be done by hijacking the part that writes to the log file and tricking it to write to the secret text file. We create a symbolic link to the secret file in a new directory but we label it as the log file. We run the program from that directory. The program opens the symbolic link thinking it is the log file but actually it opens the secret file and writes into it.

3. Large input Crash

We try to Crash the Program by providing large inputs to the program hoping the buffer used to handle the input overflows.

4. System calls / Popen calls

None of the programs had this, so i didnt run this attack. But the concept is to create functions/scripts that do whatever we want and replace the functionality intended by the programmer.

Results:

	Users --->	339257	4BF11F	D06E08
Attacks |/				
Fake Config Files		N	N	N
Writing to secret File		Y	N	N
Large Input Crash		N	N	N
System/Popen Calls		N	N	N

	Users --->	76B55C	D115BE	B801C5	382D0D
Attacks |/					
Fake Config Files		Y	Y	Y	Y
Writing to secret File		Y	Y	Y	Y
Large Input Crash		N	N	N	N
System/Popen Calls		N	N	N	N