**Funnel**

**What I was trying to do**

My goal was to see whether internal services (like a database bound to localhost) could be accessed from my machine after getting a shell on a host inside the target network. I focused on quick, repeatable commands so you can follow along.

**Quick reconnaissance — what I ran and found**

I started with simple scans and directory checks to see what was visible.

Commands I used (examples):

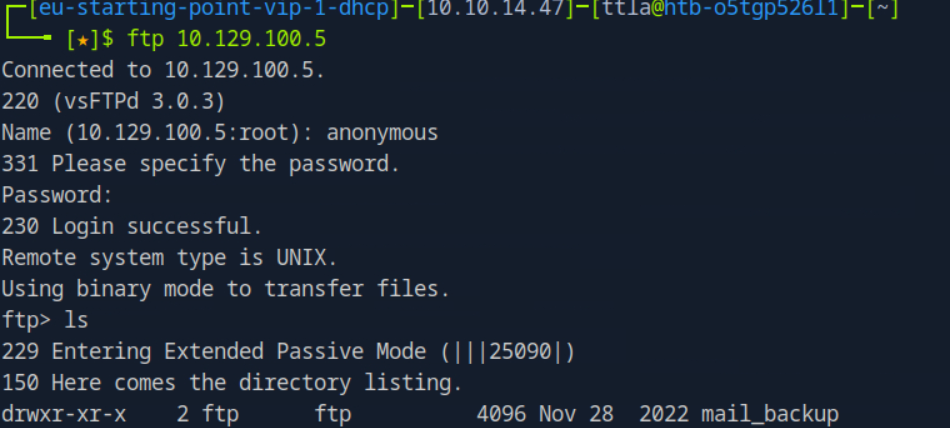
nmap -sV -sC 10.129.100.5

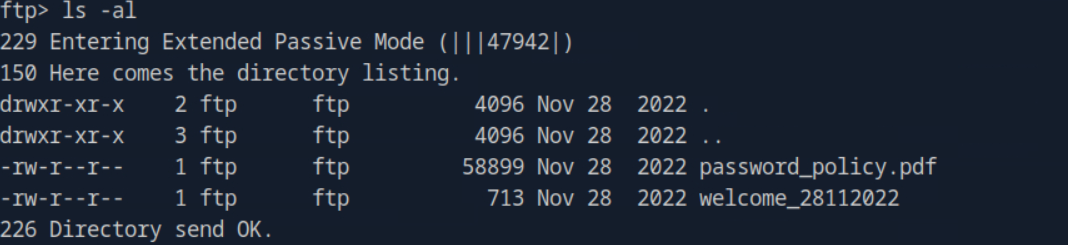
What I found:

* FTP: anonymous access with a mail\_backup folder
* SSH: standard login available
* Postgres: listening on localhost only (127.0.0.1:5432)

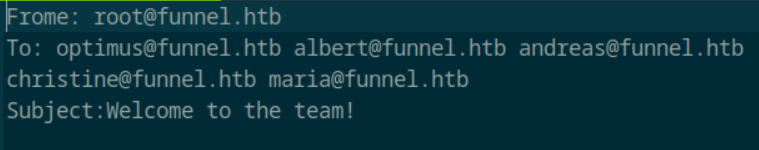
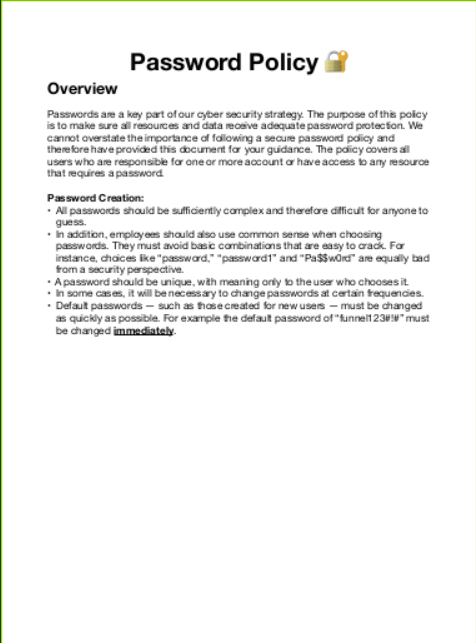


The FTP backup included a welcome email listing usernames and a password\_policy.pdf that contained a default password (funnel123#!#). That gave me usernames + a likely password — a great place to try a focused login attempt.



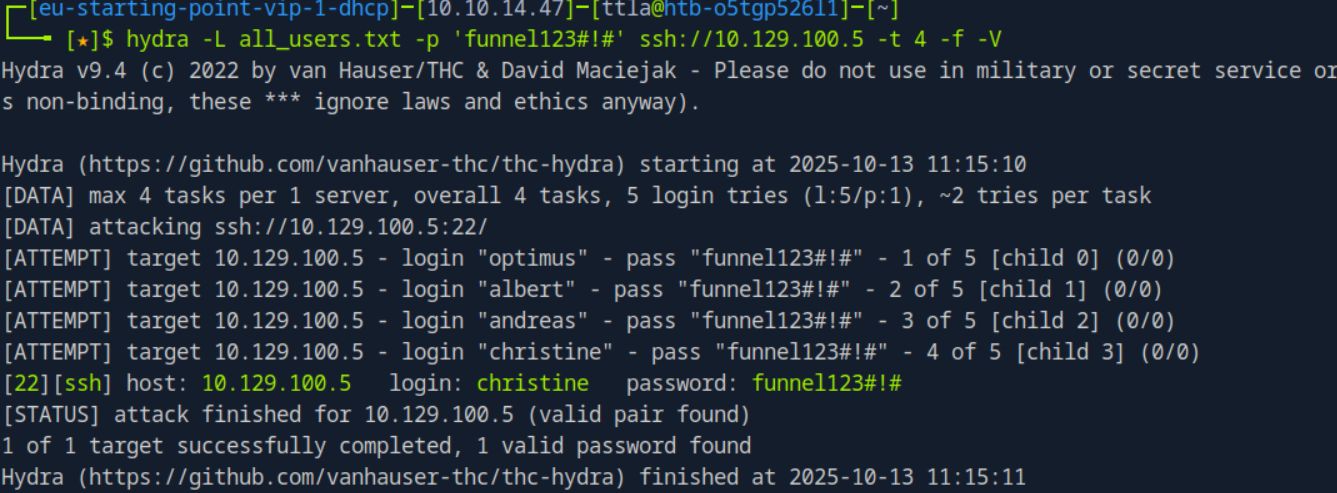


I then use GET to install these files



In the policy it said that the default password is funnel|123#!#

So I want to check if anyone hasnot changed yet

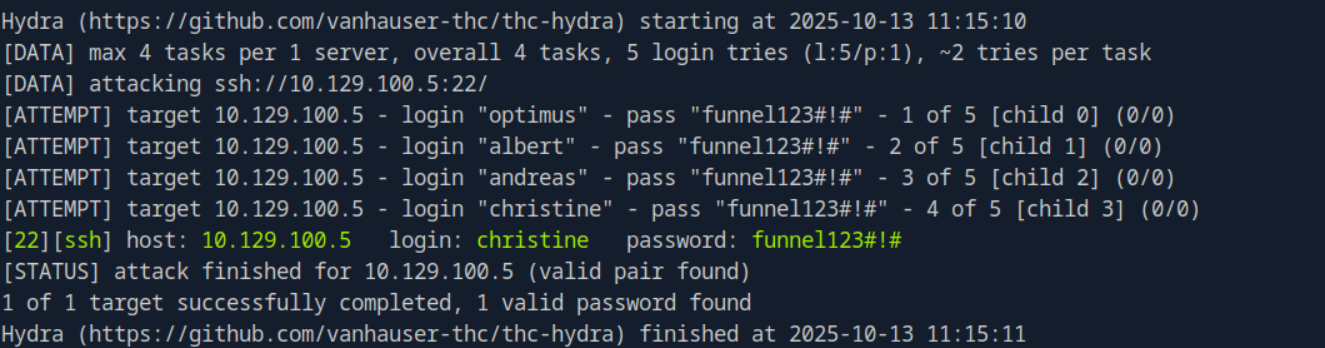


I create a file all\_users and add all the name to that file txt

I tried a password-spray approach: one password across multiple usernames. Hydra does this quickly.

Example command I ran:

hydra -L usernames.txt -p 'funnel123#!#' ssh://10.129.228.195 -t 4 -f -V



Result: I discovered christine:funnel123#!# worked and I was able to SSH in as christine.

Always validate credentials manually after an automated hit (e.g., ssh christine@10.129.228.195).

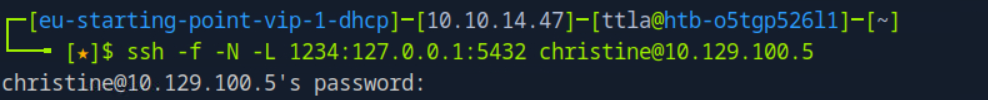
ss -tlnp

I confirmed Postgres was listening on 127.0.0.1:5432 — meaning it’s not exposed to the network, but it was reachable from the host.

Because Postgres was only listening locally on the remote, I created an SSH tunnel from *my* machine so localhost:1234 on my laptop forwarded to 127.0.0.1:5432 on the remote host.

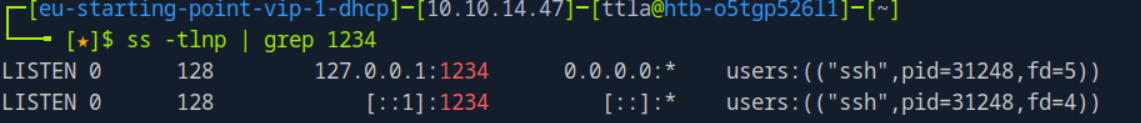
Command I used (backgrounded):

ssh -f -N -L 1234:127.0.0.1:5432 [christine@10.129.228.195](mailto:christine@10.129.228.195)



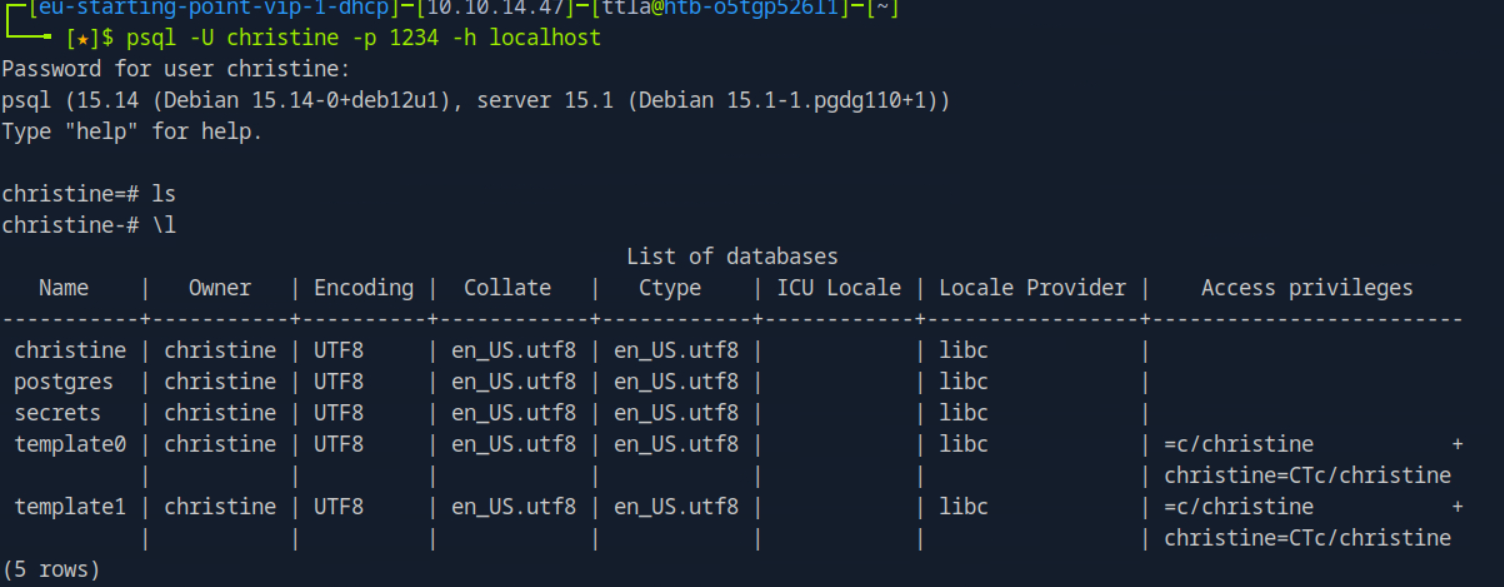
Quick checks I ran locally:

ss -tlnp | grep 1234



There we go

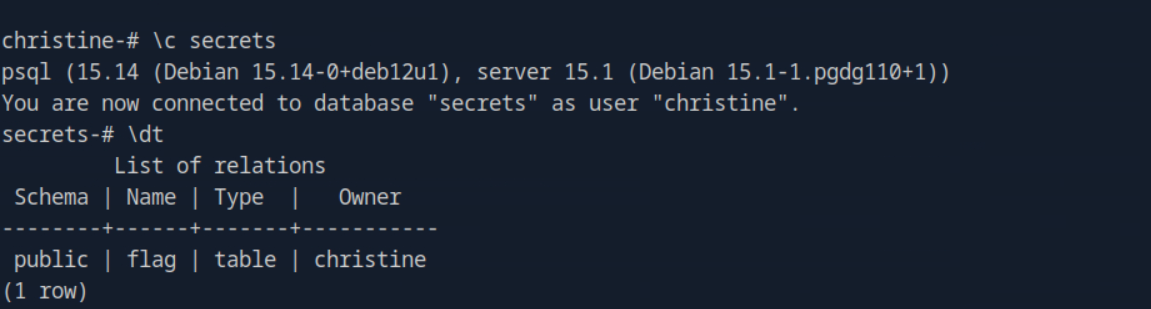
Then since I know port 5432 run on Psql



psql -h 127.0.0.1 -p 1234 -U christine -W

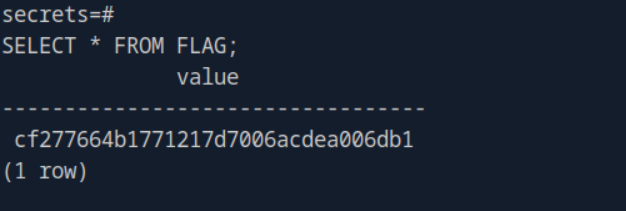
I connected with the same password and listed the secrets database contents to confirm access.

Then I move to secrets by using \c



Then normal SQL command:

SELECT \* FROM FLAG;



I found it