HTB — DarkZero

I scanned the network and found an MSSQL instance on the target. I tested credentials for john.w and they worked. From there I discovered a linked-server setup from DC01 to DC02, used that link to run commands as the SQL service account, staged a reverse payload, and got a foothold. From the foothold I ran local escalation modules to reach Administrator, dumped hashes and creds, grabbed Kerberos tickets with Rubeus to pivot to the domain controllers, performed a DCSync for the Administrator account, and pulled the root flag from the Administrator's desktop.

Recon

I started with a full port scan and found Microsoft SQL Server on the host — the instance showed up as DC01 in the darkzero.htb domain. I tried MS SQL auth with john.w and the password worked, so I connected with an MSSQL client and began enumerating.

Observed (sanitized):

MSSQL 10.129.174.88:1433 DC01

[*] 10.0 Build 26100 (name:DC01) (domain:darkzero.htb)

[+] darkzero.htb\john.w:RFulUtONCOL!

SQL enumeration

Inside the database I listed linked servers and found DC02.darkzero.ext linked from DC01. The mapping showed my login (darkzero\john.w) mapped to the remote login dc01_sql_svc, which is exactly what you don't want to see when it has high privileges.

Key output (sanitized):

SRV_NAME SRV_PROVIDERNAME SRV_PRODUCT SRV_DATASOURCE

DC01 SQLNCLI SQL Server DC01

DC02.darkzero.ext SQLNCLI SQL Server DC02.darkzero.ext

Linked Server Local Login Remote Login

DC02.darkzero.ext darkzero\john.w dc01_sql_svc

I also checked role membership:

SELECT IS_SRVROLEMEMBER('sysadmin');

The account had the rights needed to enable xp_cmdshell on the linked server.

Command execution via linked server

I enabled xp_cmdshell on the linked server and confirmed the commands ran as the SQL service account:

whoami -> darkzero-ext\svc_sql

That gave me command-level access on DC02 as svc_sql.

Foothold

I generated a reverse executable locally, hosted it on a simple HTTP server, and used the SQL command execution to download and run it from DC02. That opened a Meterpreter session back to my listener — my initial foothold on the network.

Privilege escalation to Administrator

With a session in hand I backgrounded it and ran a local exploit suggester to find privilege escalation options. One of the recommended local exploits worked for the host's patch level and kernel, and that got me Administrator.

Post-exploitation and credential collection

Once I had Administrator, I dumped credential artifacts and hashes. Notable entries I pulled:

Administrator:500:...:5917507bdf2ef2c2b0a869a1cba40726

svc_sql:1103:...:816ccb849956b531db139346751db65f

DC02$:1000:...:663a13eb19800202721db4225eadc38e

I also ran SharpHound to collect AD relationships and imported the results into BloodHound to map escalation paths:

.\SharpHound.exe -c All --zipfilename zero

.\SharpHound.exe -c All --zipfilename zero2 -d darkzero.htb

That helped me identify where to pivot next.

Kerberos ticket capture and domain pivot

To move to the DC, I ran Rubeus on the compromised host to monitor for Kerberos tickets and triggered an authentication event from DC01 to DC02 to capture a TGT for the machine account. Once I had the base64 ticket, I injected it (PTT) and gained a Kerberos context that let me interact with the domain controller as the machine account.

Commands I used (sanitized):

Rubeus.exe monitor /interval:5 /nowrap

trigger authentication (e.g., provoke a network auth)

Rubeus.exe renew /ticket:<base64-ticket> /ptt

The monitor showed a captured TGT for DC01$, which I used for the next step.

DCSync and domain compromise

With the Kerberos context in place I performed DCSync using mimikatz and extracted the Administrator NTLM hash:

mimikatz # lsadump::dcsync /domain:darkzero.htb /user:DARKZERO\Administrator

NTLM: 5917507bdf2ef2c2b0a869a1cba40726

With that hash I authenticated to the DC (pass-the-hash or DRSUAPI methods) and confirmed full domain control.

Flags

After achieving Administrator privileges, I retrieved the final flag from:

C:\Users\Administrator\Desktop\root.txt

Lessons learned

- Linked servers are a major lateral-movement risk when service accounts are over-privileged.
- xp_cmdshell is an immediate execution vector if misused or left enabled.
- Service account credentials and build artifacts are high-value targets — treat them like secrets.
- Kerberos ticket capture and replay is powerful for domain pivoting but can be noisy and detectable.
- DCSync remains the fastest way to extract domain credentials once you have sufficient access.

— MalwareMusashi