CYBER JUDO: OFFENSIVE CYBER DEFENSE

Tal Be'ery, Sr. Security Research Manager, Microsoft ATA, @TalBeerySec
Itai Grady, Security Researcher, Microsoft ATA, @ItaiGrady

Microsoft

Speaker Info – Tal Be'ery

- Sr. Security Research Manager @Microsoft
- Developing MicrosoftATA (Advanced Threat Analytics)
- Former VP for Research @Aorato (Acquired by Microsoft)
- 15 years of security research experience
- Author of the TIME attack on SSL
- Regular speaker in top international security conferences
- Named a "Facebook Whitehat"
- Twitter: @TalBeerySec

Speaker Info – Itai Grady

- Security Researcher @Microsoft
- Developing MicrosoftATA (Advanced Threat Analytics)
- Twitter : @ItaiGrady

Agenda

- Intro
  - The Cyber Boxer vs. the Cyber Judoka
  - Targeted attacks: TTPs, Kill-chain, MicrosoftATA
- Lateral Movement Reconnaissance
  - Attacker's TTPs + BloodHound + Automated Lateral Movement 

  - Cyber Boxer Defense: Net Cease and Friends 

  - Cyber Judo Defense
    - NetSess to Detect PtH 

    - SAMR to Detect Local Users attacks 

- Kerberos Error Injection:
  - Attacks, Cyber Boxer Defense, Cyber Judo Defense 

- Outro

Intro

Level 0: The “Fighting in the Dark” Model

“If you know the enemy and know yourself, you need not fear the result of a hundred battles.

If you know yourself but not the enemy, for every victory gained you will also suffer a defeat.

If you know neither the enemy nor yourself, you will succumb in every battle.”

https://en.wikipedia.org/wiki/Sun_Tzu#/media/File:Enchoen27n3200.jpg

Level 1 : The “Boxer” Model

- Defenders Learn attackers' TTPs:
  - Tactics
  - Techniques
  - Procedures
- Defenders build their own TTPs
- Defenders practice the attackers’ TTPs
  - Red Team
- Defenders practice their TTPs
  - Blue Team
- Some time same team for both
  - Purple Team

Level 2 : The Judoka Model

- Judoka, a Judo warrior, uses the opponent's strength and movement against him
- Defenders Learns attackers' TTPs:
- Defenders **adopts** some attackers' TTPs in order to defeat them!
- That's our topic for today's talk

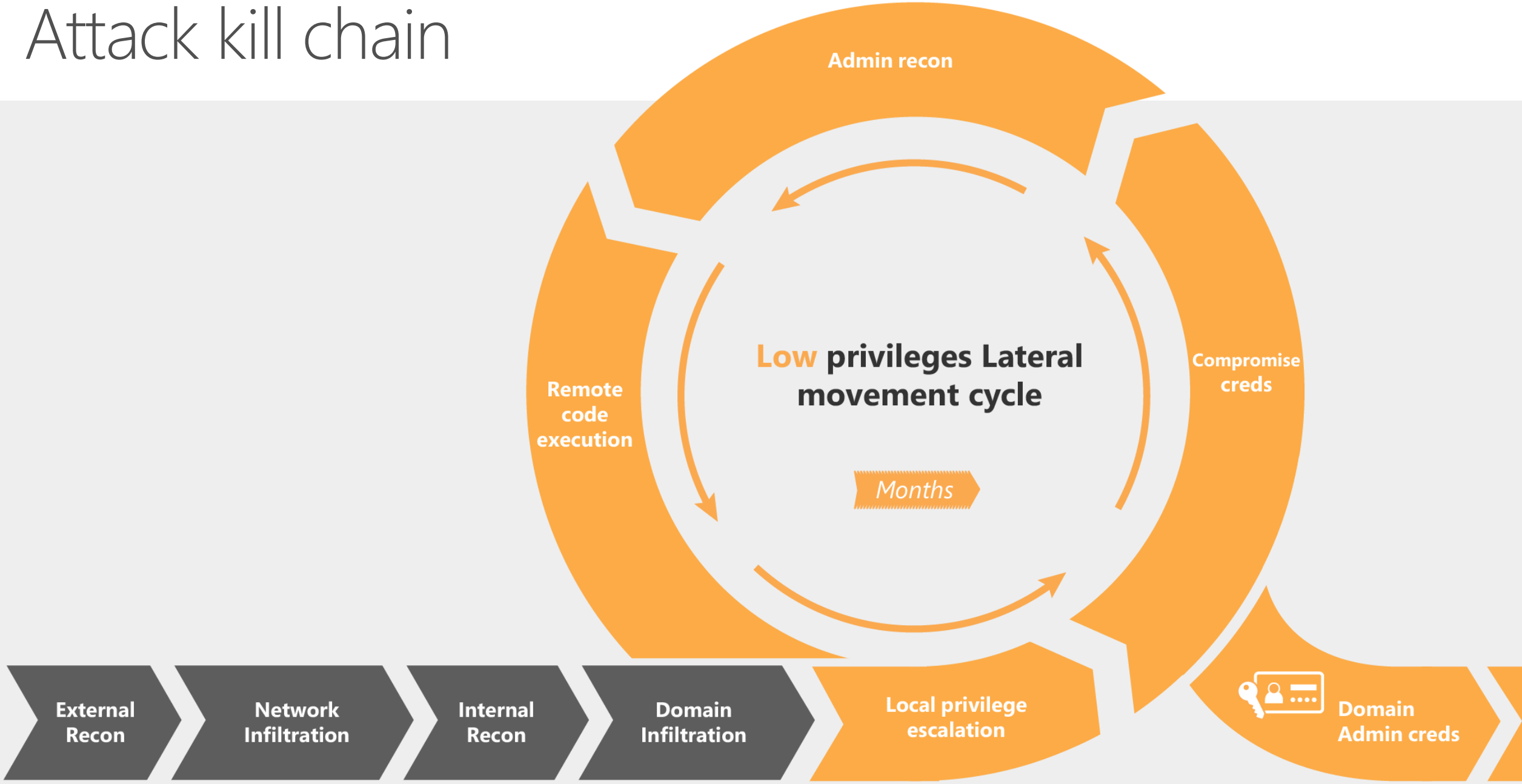
Defenders and Attackers: Not so Apart

- Similar tools, challenges and scenarios

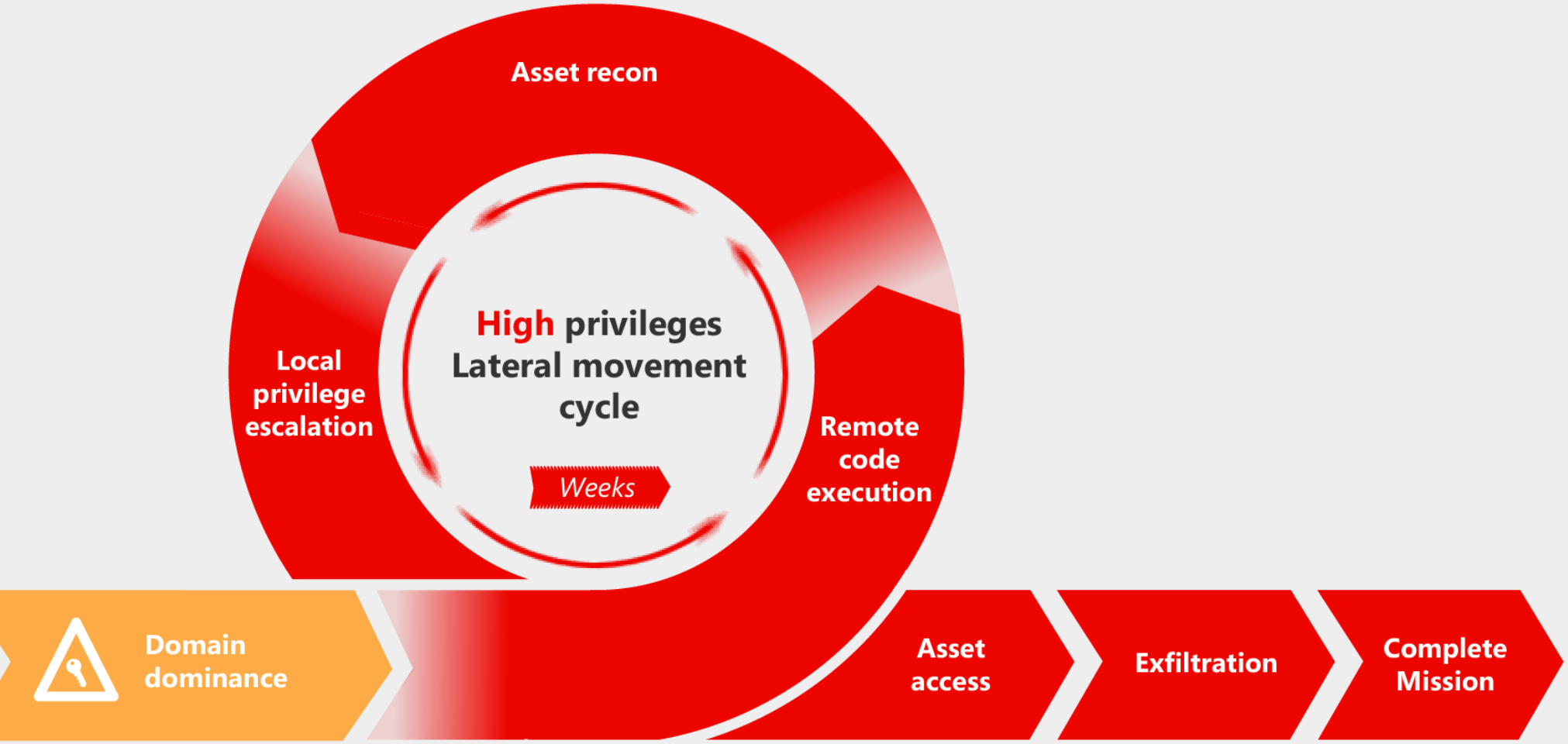
	Defenders	Attackers
Network Deployment	Proxy / Network Monitoring	MITM / Eavesdropper
Host Deployment	Agent (but the prefer to refrain: compatibility, performance)	Malware (but the prefer to refrain: compatibility, performance, detection)
Privileges	Least, o.w. part of the problem (see: @taviso)	Least, privileged user are more monitored
Integrations	“living off the land”. Core functionality must be delivered independently, opportunistic integrations	“living off the land”. Core functionality must be delivered independently, opportunistic existing non-default capabilities abuse
Expertise	OS internals, networking	OS internals, networking

Targeted Attacks

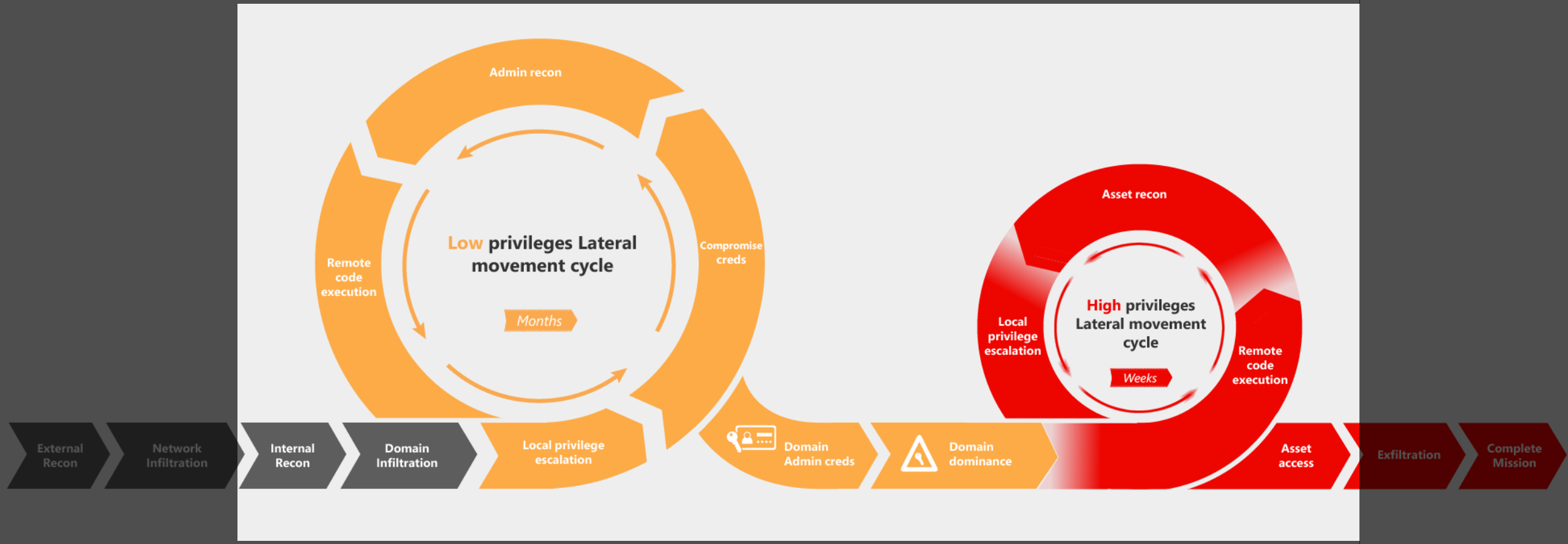
Attack kill chain



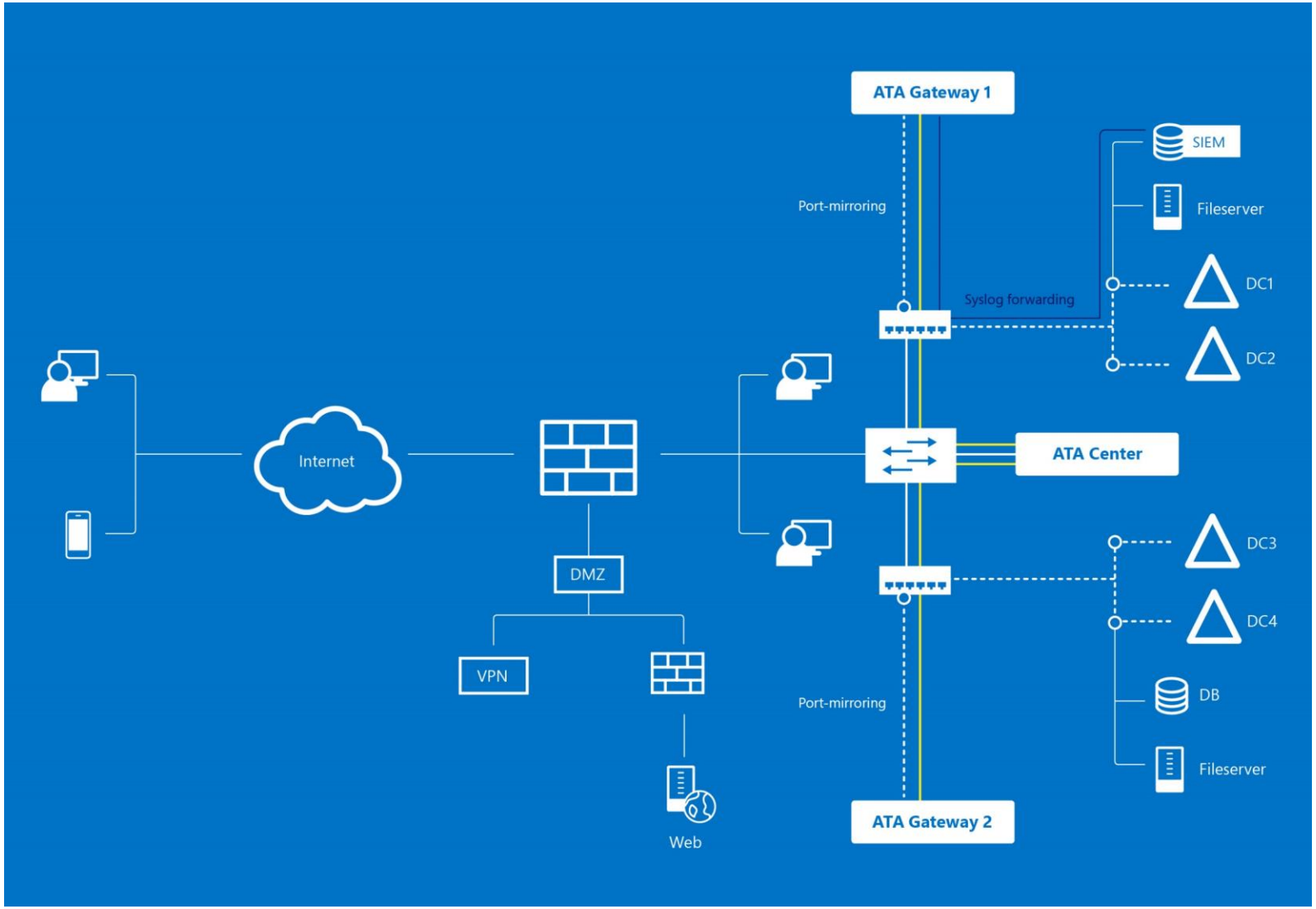
Attack kill chain



Attack kill chain and ATA

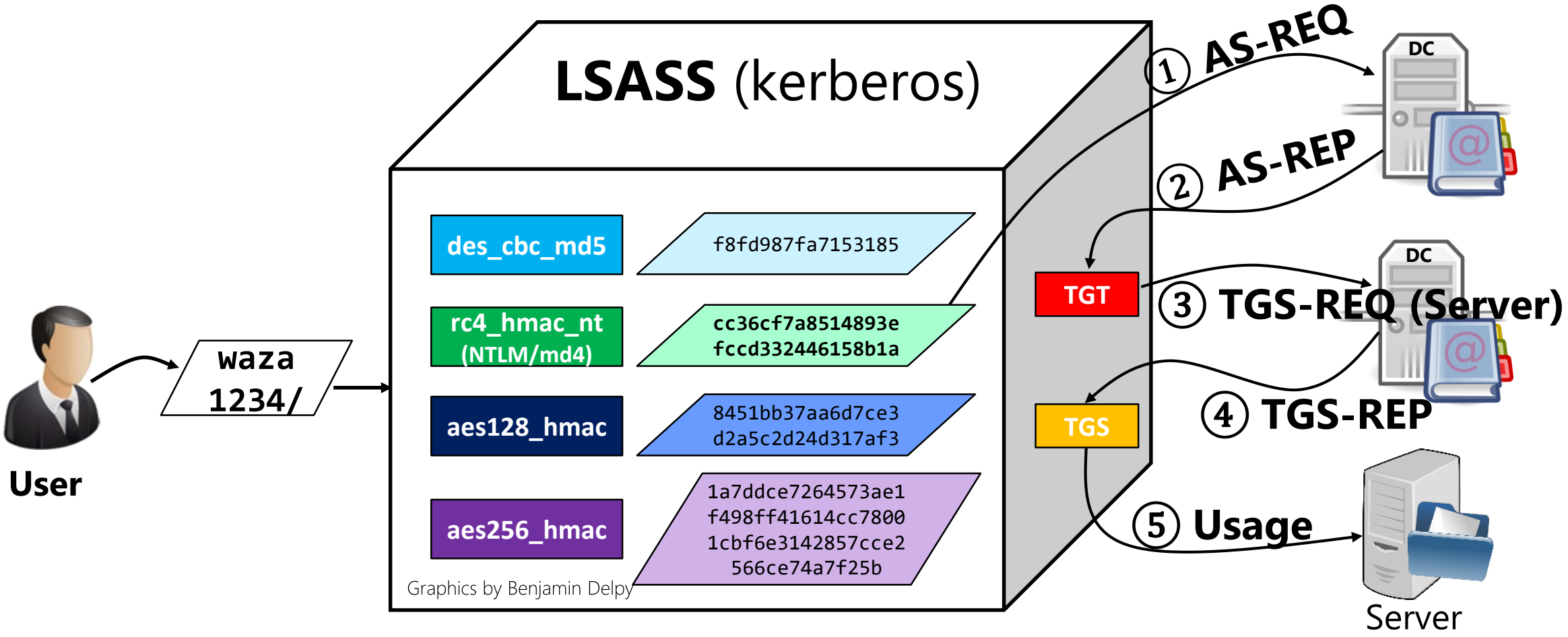


Microsoft ATA topology



Kerberos Authentication

- Windows default authentication (+ authorization) protocol



NTLM Authentication

- Authentication
- Authorization

Lateral Movement Reconnaissance

Lateral Movement

- Attackers' mission
- Get from HERE:
  - The gullible user who clicked the phishing email
  - Not always the manager ☺
  - Probably low privileged

- To THERE:
  - Full domain dominance
  - Control the Domain Controller

How to get from HERE to THERE?

- A map! (CS: Graph)
- Cities (CS: Nodes)
  - **Computers and logon user**
- Roads (CS: Edges)
  - **Computer and users permissions**
- Computer A → Computer B
  - **Logged on users in computer A have code execution permissions on Computer B**

Building the map

- To build the map attackers need to know:
  - **Cities: Where users are logged-on**
  - **Roads: User's permissions to computers**
  - **Destination: Who are the Domain Admins**
- Knowledge must be gained with just a low privileged user account
- "Reconnaissance": Gaining knowledge on adversary in a hostile environment

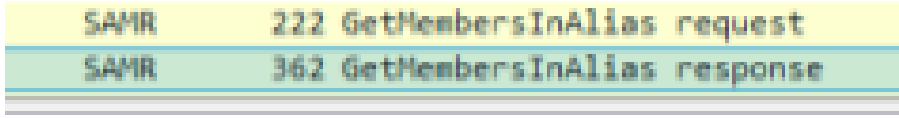
Logged-on User Recon

- SMB service can be remotely queried for active sessions
  - Returns: IP address, User
  - Implemented in NetSess tool (others)
  - On the wire SRVSVC protocol
  - Required Permissions: Any domain user
- Every logged-on user fetches Group Policy (GP) from DC
  - On logon and periodically (every ~1.5 hour)
  - GP is a bunch of files, sent over SMB
- 1.5 hour of periodic sampling of DC is enough to find all currently logged-on users' computers


```
C:\Research>NetSess.exe win-2008r2 /full

NetSess V02.00.00cpp Joe Richards <joe@joeware.net> January 2004

Enumerating Host: win-2008r2
Client                User Name            Client Type            Opens  Time
     Idle Time Transport
-------------------------------------------------------------------------------
-----------------------------------------------
\\10.0.0.2            theadmin                                        1  000:0
0:00 000:00:00

Total of 1 entries enumerated
```

Computer's Local Admin Recon

- PS cmdlet
  - 'Get-NetLocalGroup'
  - PowerSploit by @mattifestation
  - https://github.com/PowerShellMafia/PowerSploit
- Can list local groups & users
- Queries the Computer over SAMR protocol
- Required Permissions: Any domain user




```
PS C:\Users\DomainUser2> Get-NetLocalGroup -ComputerName Client1 -ListGroups

Server                  Group                       SID          Description
------                  -----                       ---          -----------
Client1                 Access Control Assistance ... S-1-5-32-579 Members of this group can ...
Client1                 Administrators              S-1-5-32-544 Administrators have comple...
Client1                 Backup Operators            S-1-5-32-551 Backup Operators can overr...
Client1                 Cryptographic Operators     S-1-5-32-569 Members are authorized to ...
Client1                 Distributed COM Users       S-1-5-32-562 Members are allowed to lau...
Client1                 Event Log Readers           S-1-5-32-573 Members of this group can ...
Client1                 Guests                      S-1-5-32-546 Guests have the same acces...
Client1                 Hyper-V Administrators      S-1-5-32-578 Members of this group have...
Client1                 IIS_IUSRS                   S-1-5-32-568 Built-in group used by Int...
```

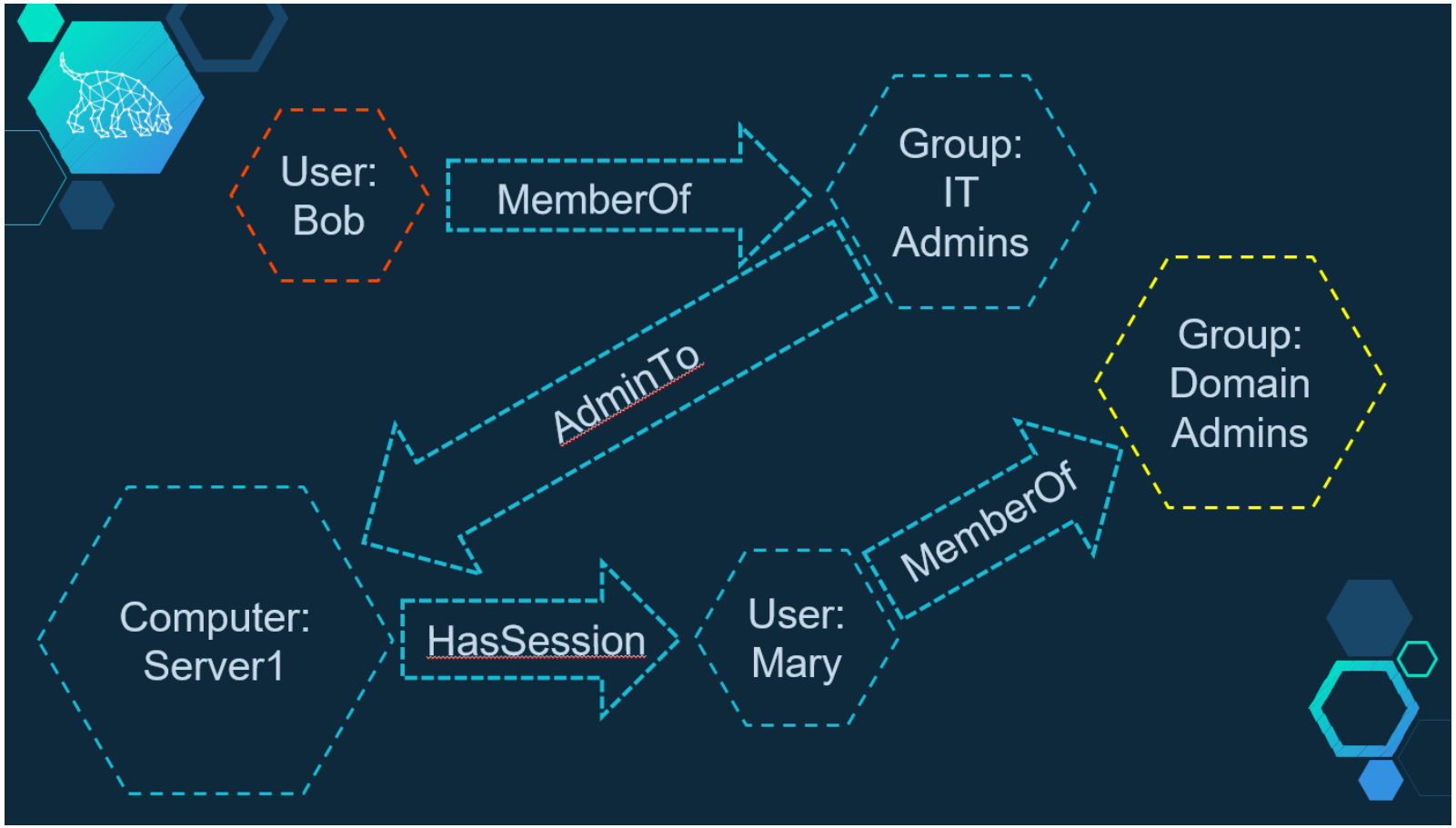
Users + Group Membership Recon

- Simple command line
- Full enumeration
  - Net User /domain
  - Net Group /domain
- Specific entity info
  - Net user /domain <user name>
  - Net group /domain <group name>
- Queries the DC over SAMR protocol
- Required Permissions: Any domain user

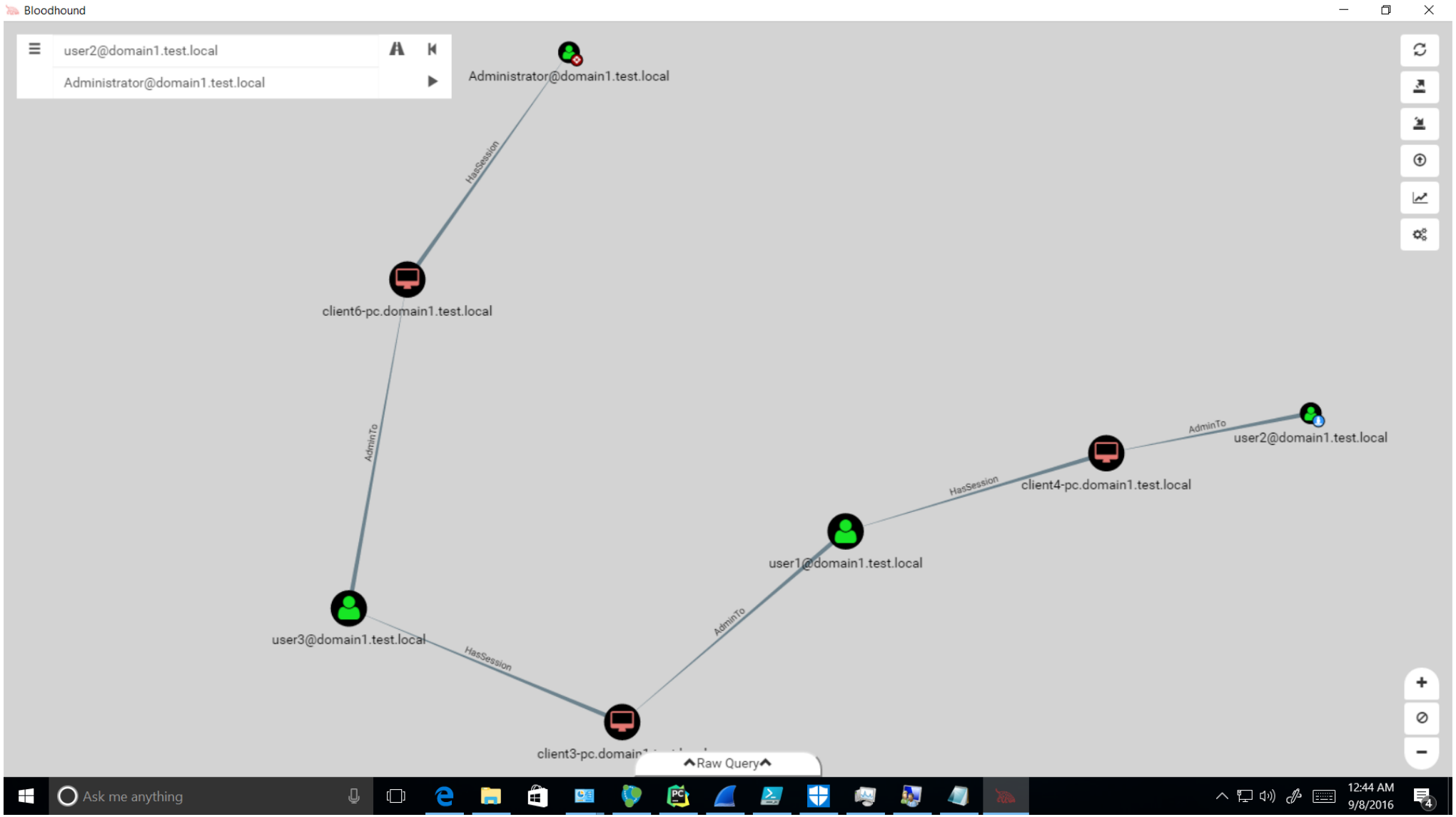
Info
Connect5 request
Connect5 response
EnumDomains request
EnumDomains response
LookupDomain request,
LookupDomain response
OpenDomain request
OpenDomain response
EnumDomainUsers request
EnumDomainUsers response
Close request
Close response
Close request
Close response

Putting it All Together: BloodHound

- BloodHound: developed by @_wald0, @CptJesus, and @harmj0y.
- https://github.com/adaptivethreat/BloodHound

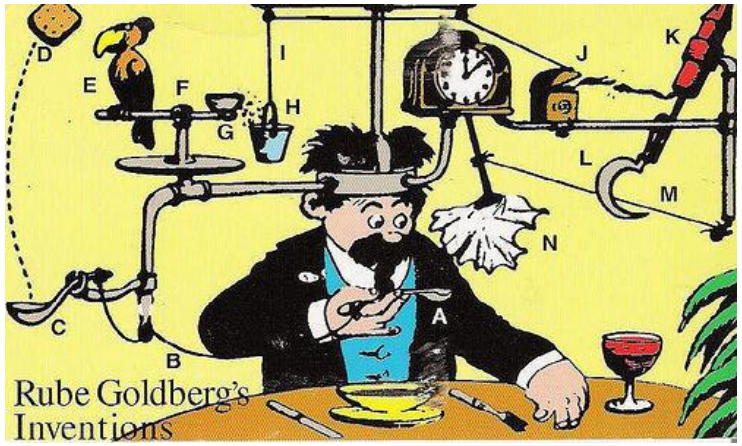


BloodHound's Shortest Path Functionality



Imagine: Automatic Internal Network Campaign

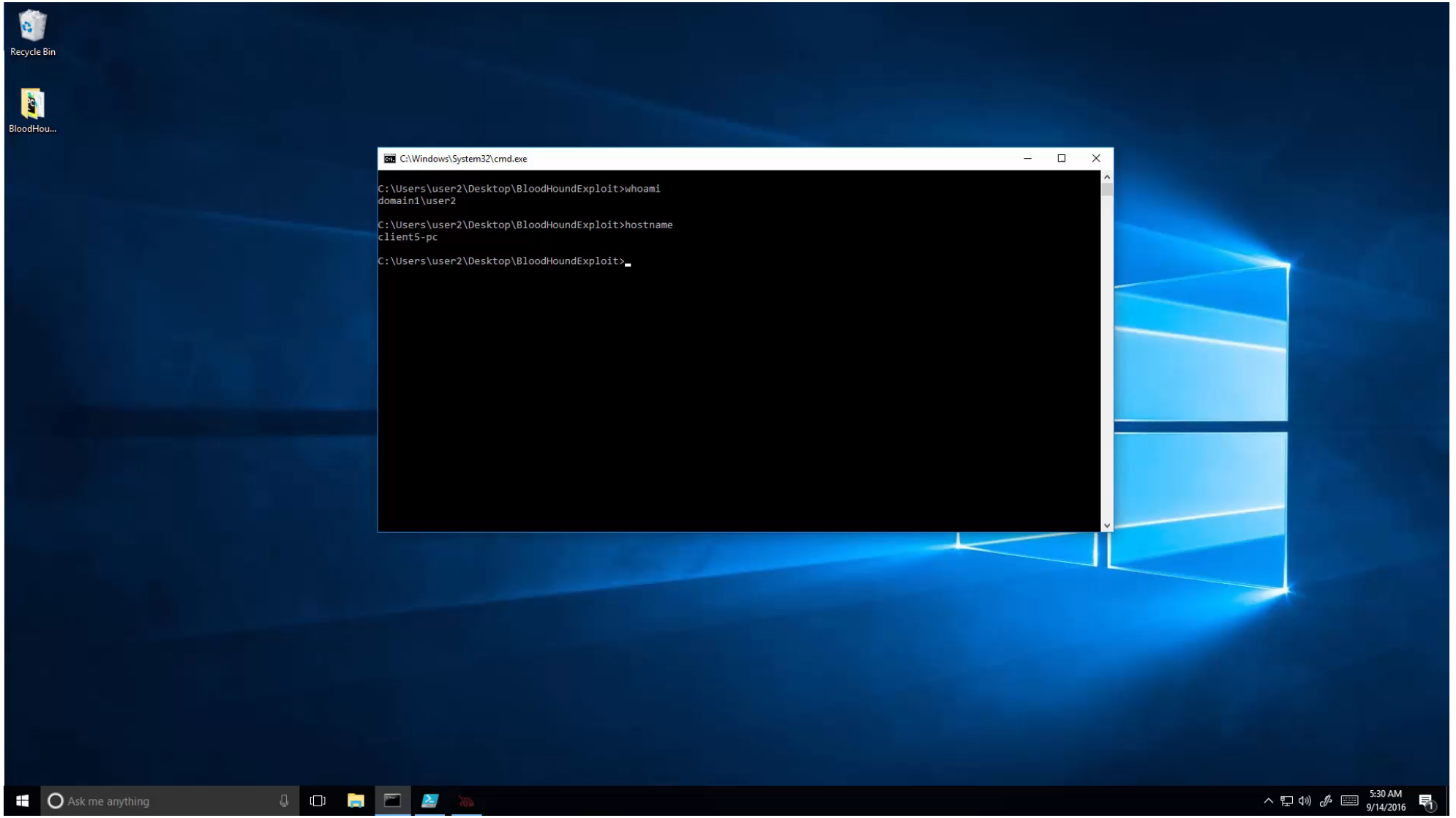
- Attackers launch a massive phishing email campaign
  - **A single low privileged domain user Clicks phishing email**
  - **gets infected with a malware**
- The malware *automatically* executes a "bloodhound" style recon
  - **Computes shortest path to Domain Admin**
- The malware *automatically* do Lateral Movement all the way to domain admin
- The malware *automatically* downloads all users' domain keys using DC-SYNC
  - **All domain keys are sent to attackers**



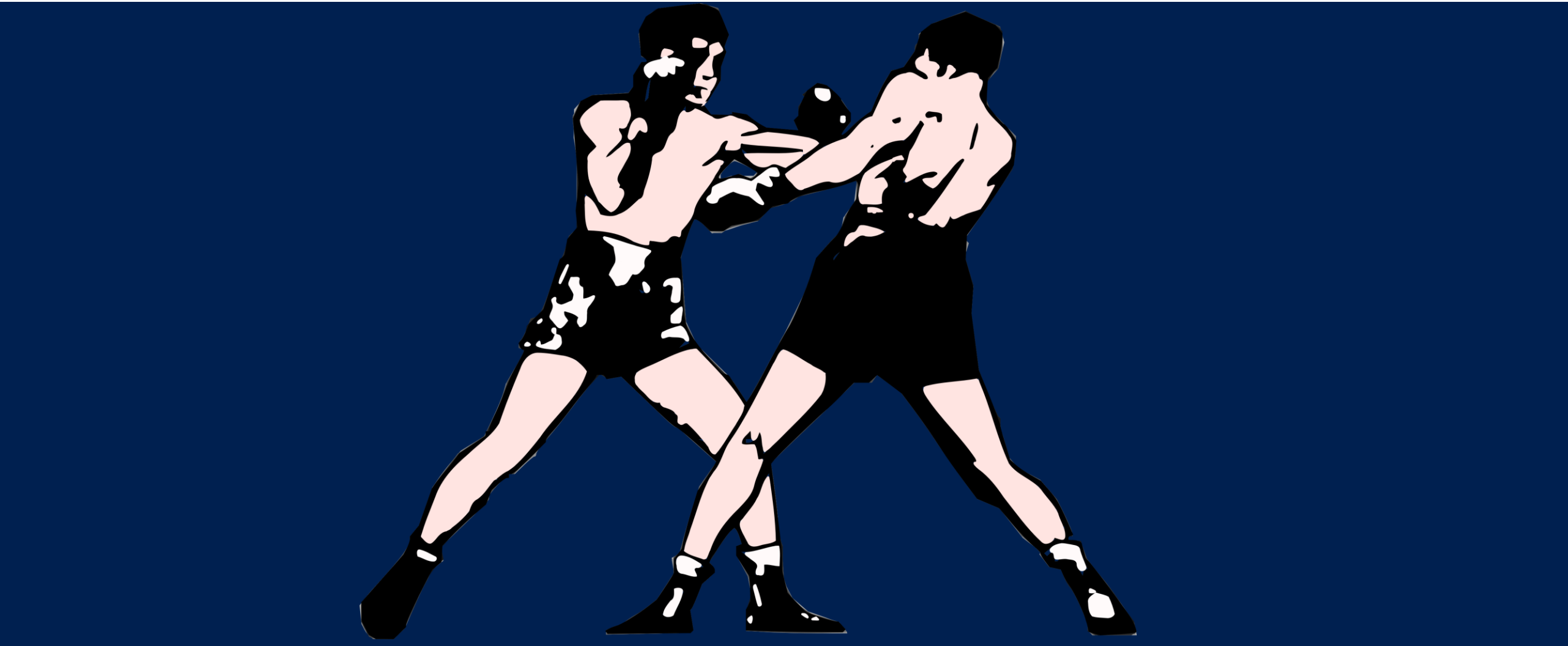
Now Open Your Eyes.. ☺

- Automatic Lateral Movement tool
- Developed by MicrosoftATA researcher @talthemaor
- Gets a bloodhound exported graph (JSON) as “attack plan” input
- Executes the attack plan
  - Remote Code Execution (RCE) via Remote PS
  - Credential harvesting via Mimikatz

DEMO!


Recycle Bin
BloodHou...
C:\Windows\System32\cmd.exe
C:\Users\user2\Desktop\BloodHoundExploit>whoami
domain1\user2
C:\Users\user2\Desktop\BloodHoundExploit>hostname
client5-pc
C:\Users\user2\Desktop\BloodHoundExploit>
Ask me anything
5:30 AM
9/14/2016

Lateral Movement Reconnaissance: Defense



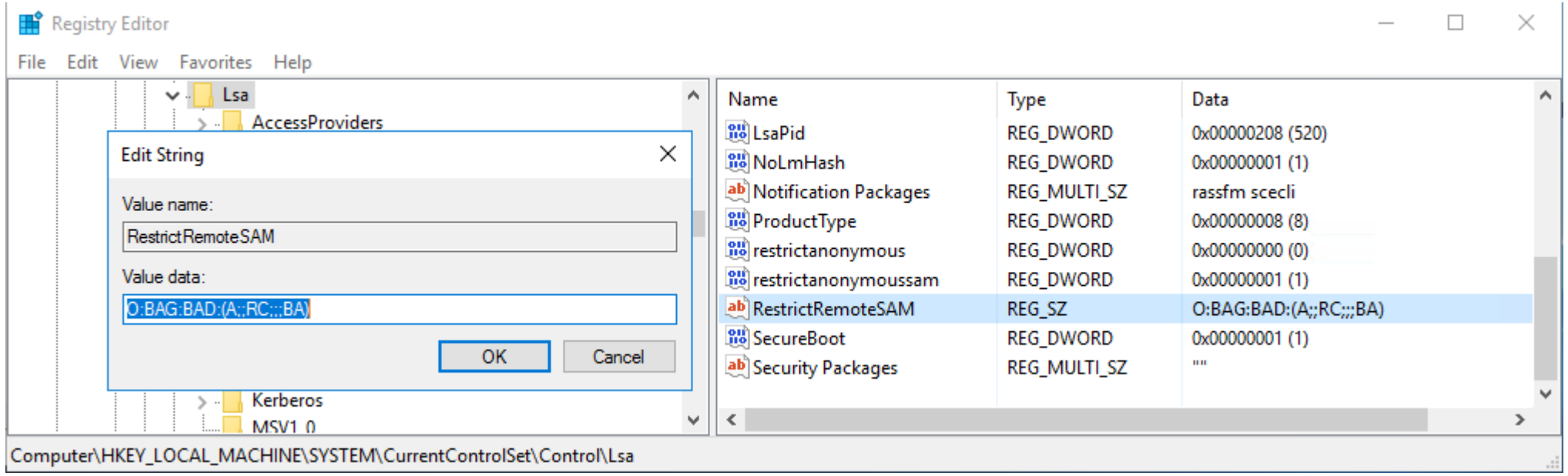
“SAMR Moved On” Hardening #1

- Win10 allows admins to control SAMR Recon
  - Registry: HKLM/System/CurrentControlSet/Control/Lsa/RestrictRemoteSAM
  - GP: “Network Access: Restrict clients allowed to make remote calls to SAM”

Win version	Who can query SAMR by default	Can default be changed
< Win10	Any domain user	No
Win10	Any domain user	Yes (only via registry)
> Win10 (e.g. anniversary)	Only local administrators	Yes (registry or GPO)

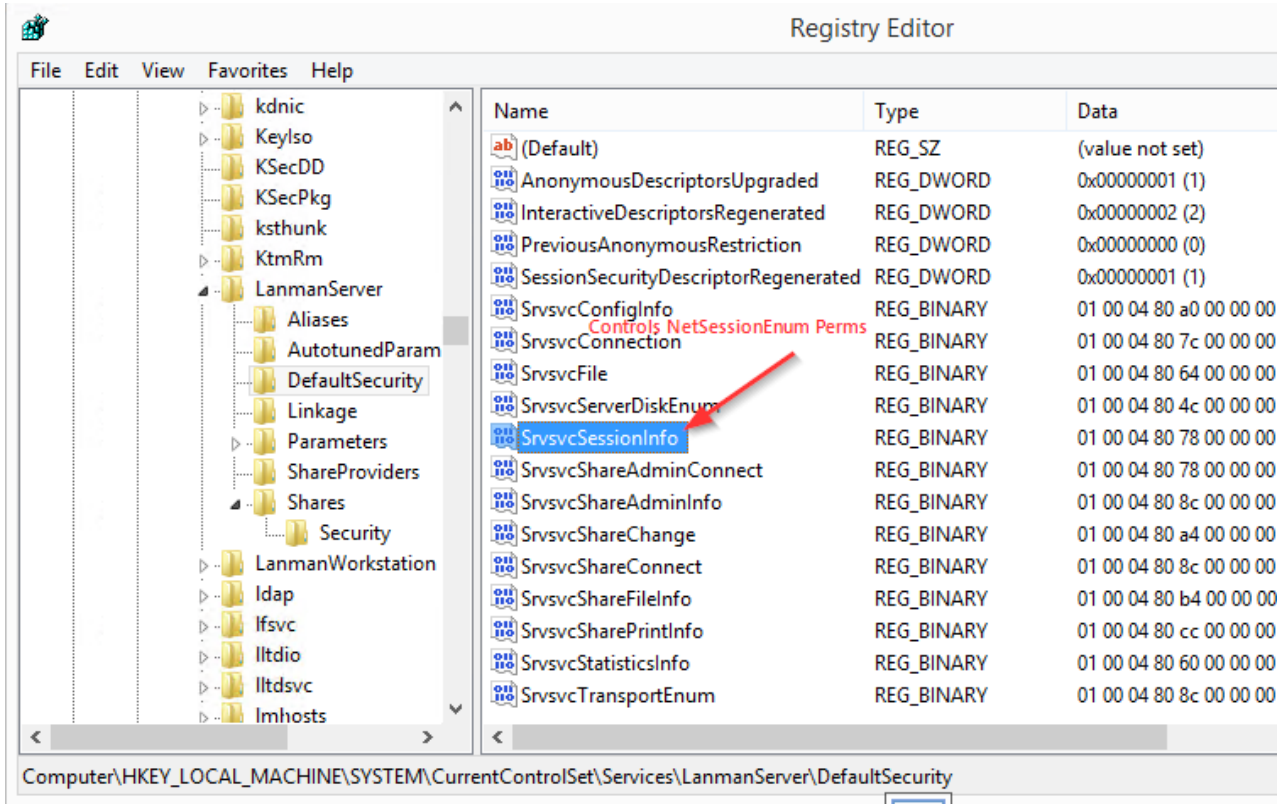
“SAMR Moved On” Hardening #2

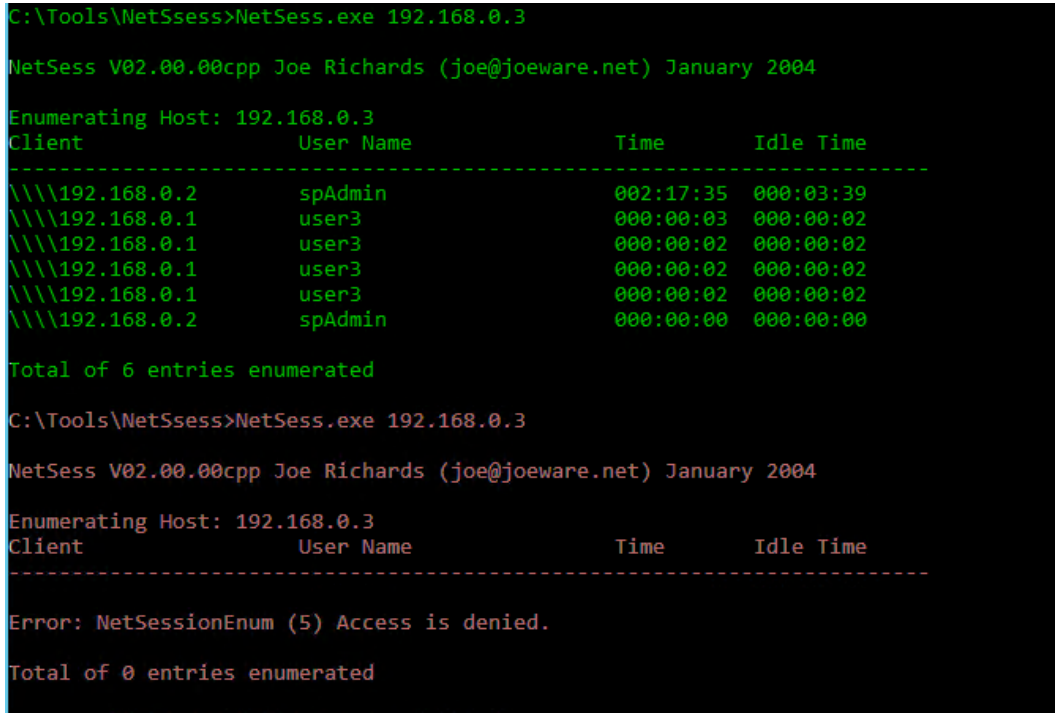
- Windows Server 2016 allows admins to control SAMR Recon on DC
- Net User/Group queries on Domain can be limited!



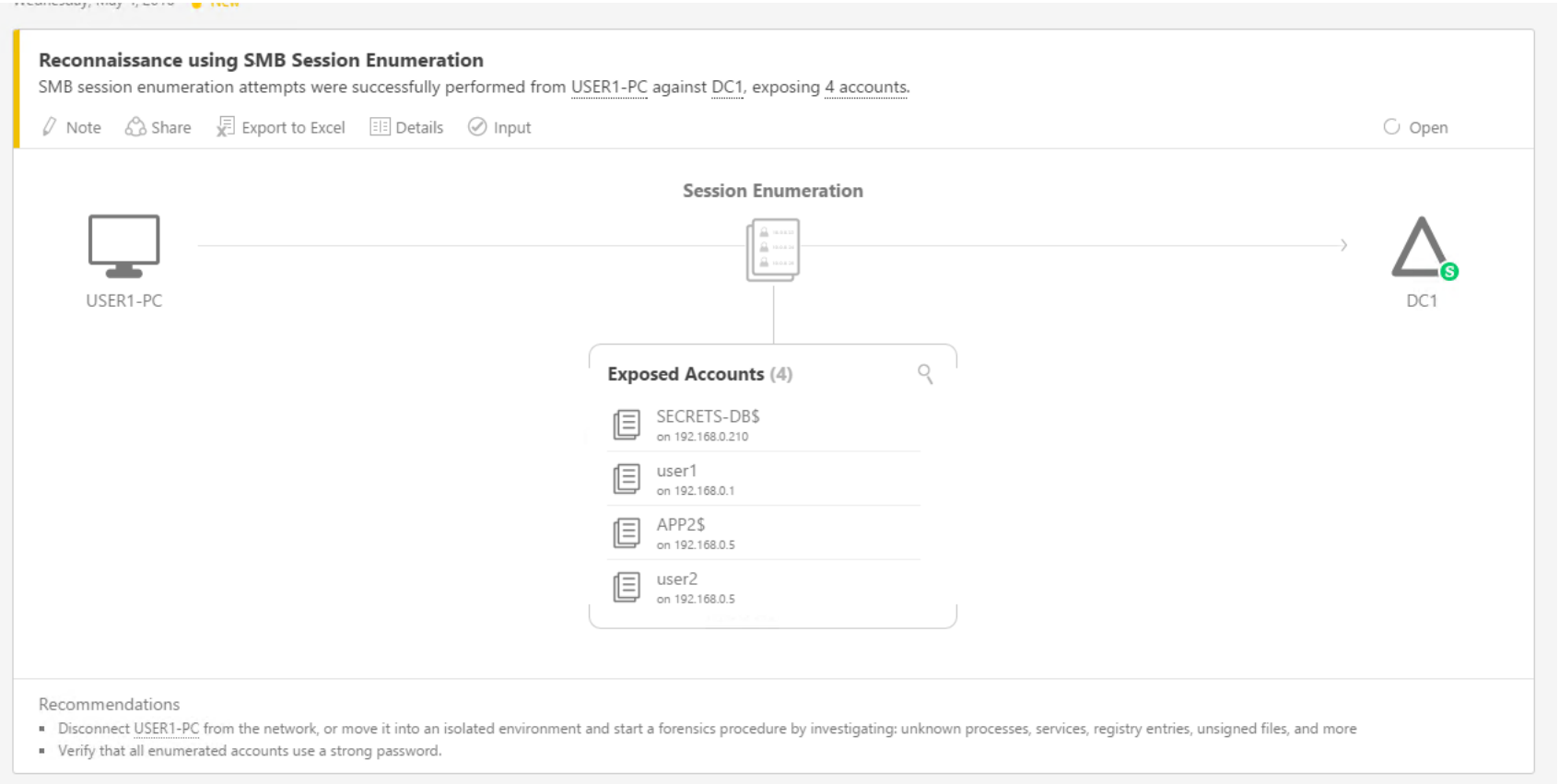
“Net Cease” Hardening

- NetSess API access can be controlled by ACL in the Registry!
- Remove “Authenticated Users” group manually, or use our tool
  - https://gallery.technet.microsoft.com/Net-Cease-Blocking-Net-1e8dcb5b

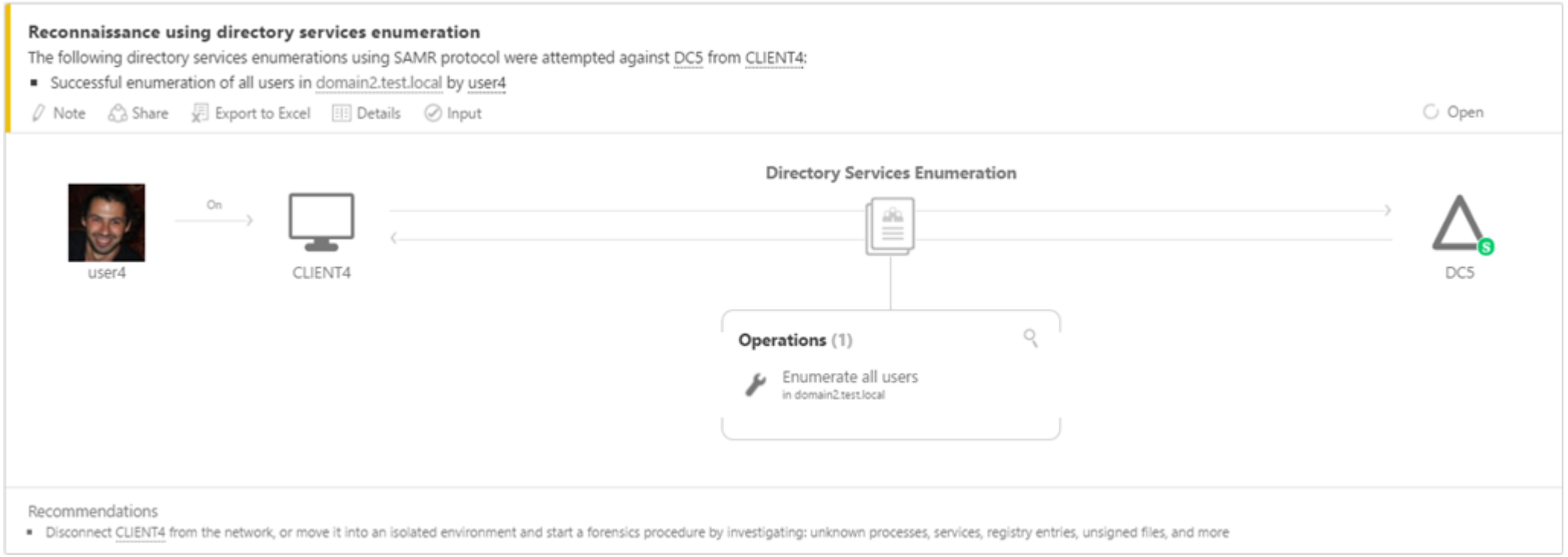




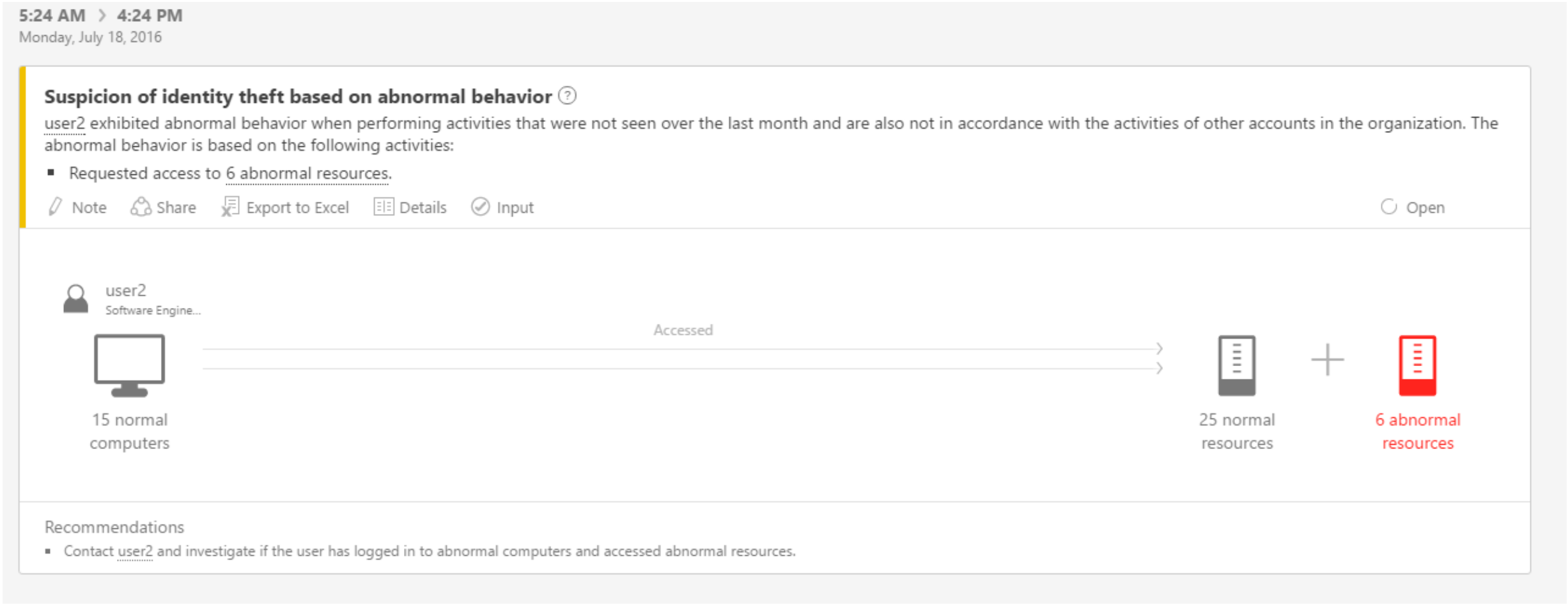
Detection (Microsoft ATA): NetSess Recon



Detection (MicrosoftATA): SAMR Recon



Detection (MicrosoftATA): Abnormal Access



Cyber Judo with NetSess

NTLM Authentication

- Authentication
- Authorization

NTLM in attacks

- NTLM is the essence
  - NTLM authentication is very relevant to Pass-the-Hash attacks
  - NTLM relay attack
- NTLM by default:
  - Many attack tools did not “upgrade” to Kerberos
  - Access by IP (e.g. “\\10.0.0.1\C$”) defaults to NTLM (no SPN for Kerberos)
  - Kerberos failures (e.g. port 88 is blocked in FireWall)

NTLM Visibility Problem

- Netlogon messages are encrypted ☹
  - The opcode is visible, but parameters are encrypted
- A security device eavesdropping to DC traffic only learns
  - "Someone performed NTLM logon to a computer"
- Missing info
  - Which user?
  - From which computer?

```
RPC NETLOGON                NetrLogonSamLogonWithFlags request
> Frame 28: 1090 bytes on wire (8720 bits), 1090 bytes captured (8720 bits) on interface 0
> Ethernet II, Src: Microsof_c5:46:b0 (00:15:5d:c5:46:b0), Dst: Microsof_c5:46:9e (00:15:5d:
> Internet Protocol Version 6, Src: daf::5, Dst: daf::200
> Transmission Control Protocol, Src Port: 49754 (49754), Dst Port: 49158 (49158), Seq: 222,
> Distributed Computing Environment / Remote Procedure Call (DCE/RPC) Request, Fragment: Sin
v Microsoft Network Logon, NetrLogonSamLogonWithFlags
    Operation: NetrLogonSamLogonWithFlags (45)
    [Response in frame: 29]
    Encrypted stub data: 0cc2bb6864ebd473e2a7a883b73abe9964612ac424cf6f37...
```

NTLM Visibility Solution

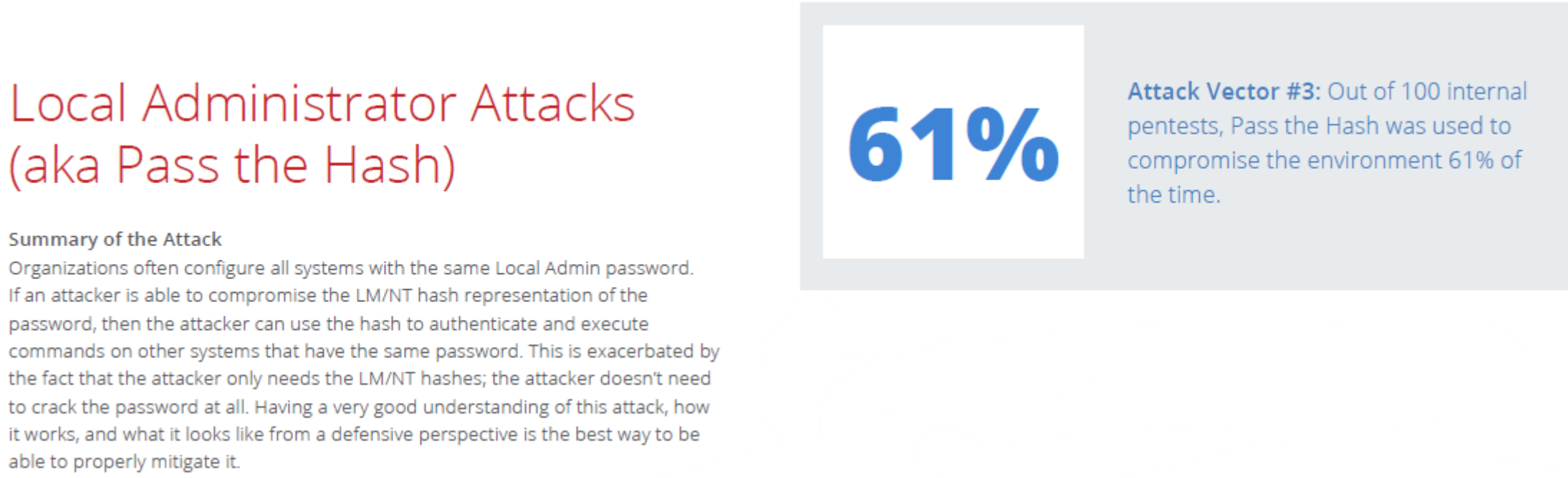
- When observing a relevant successful NETLOGON message
  - **Extract source IP**
- Perform NetSess Recon on that IP
- Extract original user information from NetSess results
- Win! (at least for SMB)

Demo!

Cyber Judo with SAMR

Local User in Attacks

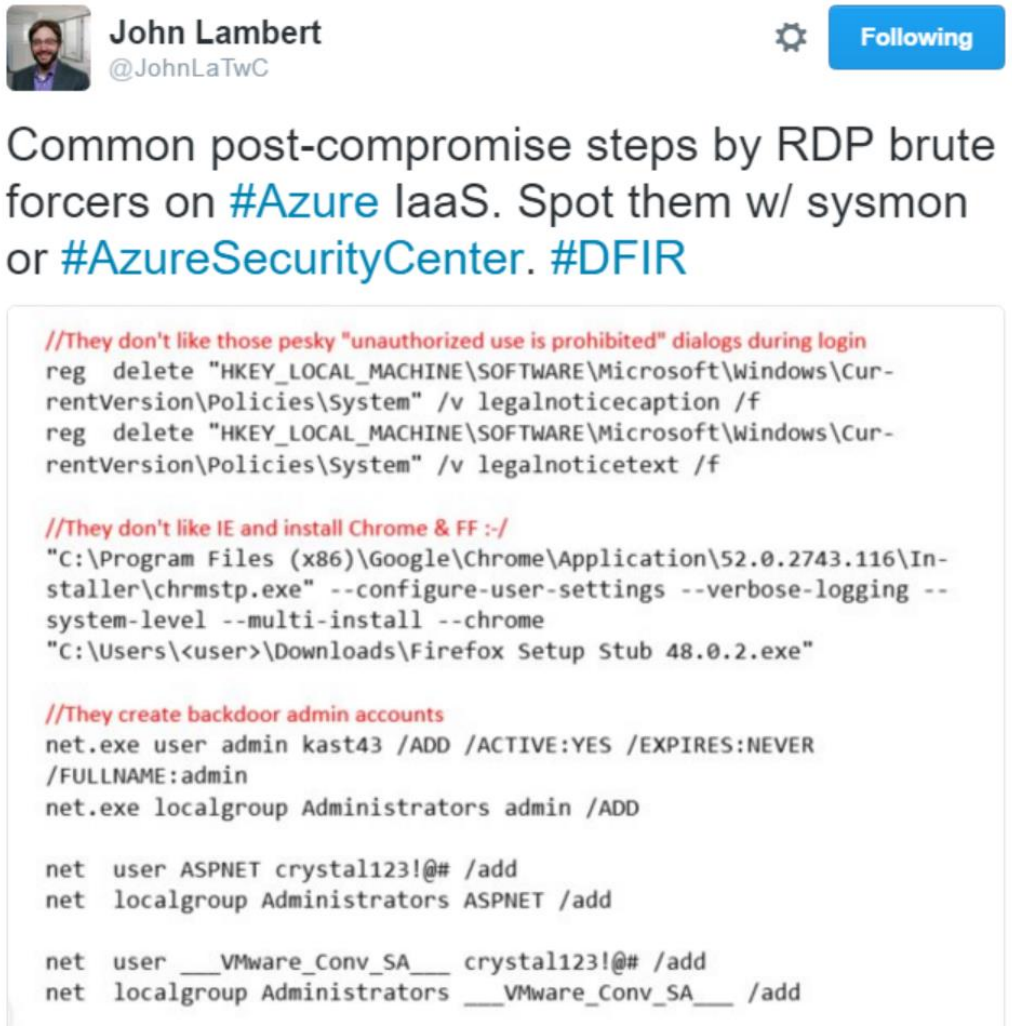
- Very relevant
  - Identical passwords problem
  - PtH against local admins is successfully Used in 60% of Praetorian's pen-tests



Source: Praetorian

Local Users Attacks in the Wild

https://twitter.com/JohnLaTwC/status/777569424156921856



Local User Visibility Problem

- Local users authentication is… well… Local
- A network security device cannot see it ☹

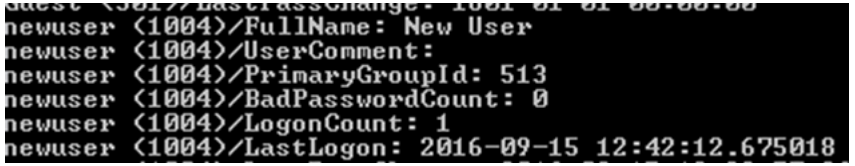
Local Users Visibility Solution

- Periodically query Local Users over SAMR
  - Users Info
  - Group membership
- To discover:
  - Abnormal login patterns
  - BruteForce attempts
  - Privileged group modifications
  - Password configuration issues
- And much more…

```
USER1 (1001)/FullName:
USER1 (1001)/UserComment:
USER1 (1001)/PrimaryGroupId: 513
USER1 (1001)/BadPasswordCount: 0
USER1 (1001)/LogonCount: 3
USER1 (1001)/LastLogon: 2016-08-29 09:54:53.853097
USER1 (1001)/LastPassChange: 2016-08-29 08:42:47.516191
```

Local User Logon Anomalies Detection

- Dormant Local User Logon:
  - Compare current last logon time to previous last logon time
  - If user never logged-on, time is 0 (“1.1.1601 00:00:00”)



- Local User Brute Force
  - Compare current BadPwdCount with previous

User Added to Privileged Local Group Detection

```
PS C:\Windows\system32> Get-NetLocalGroup -ComputerName CLIENT2

Server      : CLIENT2
AccountName : VLAB1/CLIENT2/Administrator
SID         : S-1-5-21-2855241813-3116034789-286929080-500
Disabled    : True
IsGroup     : False
IsDomain    : False
LastLogin   : 4/13/2014 5:37:53 PM

Server      : CLIENT2
AccountName : VLAB1/CLIENT2/testadmin
SID         : S-1-5-21-2855241813-3116034789-286929080-1001
Disabled    : False
IsGroup     : False
IsDomain    : False
LastLogin   : 8/5/2015 2:51:48 PM

Server      : CLIENT2
AccountName : VLAB1/CLIENT2/USER2
SID         : S-1-5-21-2855241813-3116034789-286929080-1002
Disabled    : False
IsGroup     : False
IsDomain    : False
LastLogin   : 8/31/2016 11:25:41 AM

Server      : CLIENT2
AccountName : VLAB1.com/Domain Admins
SID         : S-1-5-21-3383964581-1309953776-2693364552-512
Disabled    : False
IsGroup     : True
IsDomain    : True
LastLogin   :

Server      : CLIENT2
AccountName : VLAB1.com/USER2
SID         : S-1-5-21-3383964581-1309953776-2693364552-1106
Disabled    : False
IsGroup     : False
IsDomain    : True
LastLogin   : 9/15/2016 5:19:55 PM
```

```
Server      : CLIENT2
AccountName : VLAB1/CLIENT2/Administrator
SID         : S-1-5-21-2855241813-3116034789-286929080-500
Disabled    : True
IsGroup     : False
IsDomain    : False
LastLogin   : 4/13/2014 5:37:53 PM

Server      : CLIENT2
AccountName : VLAB1/CLIENT2/testadmin
SID         : S-1-5-21-2855241813-3116034789-286929080-1001
Disabled    : False
IsGroup     : False
IsDomain    : False
LastLogin   : 8/5/2015 2:51:48 PM

Server      : CLIENT2
AccountName : VLAB1/CLIENT2/USER2
SID         : S-1-5-21-2855241813-3116034789-286929080-1002
Disabled    : False
IsGroup     : False
IsDomain    : False
LastLogin   : 8/31/2016 11:25:41 AM

Server      : CLIENT2
AccountName : VLAB1.com/Domain Admins
SID         : S-1-5-21-3383964581-1309953776-2693364552-512
Disabled    : False
IsGroup     : True
IsDomain    : True
LastLogin   :

Server      : CLIENT2
AccountName : VLAB1.com/USER2
SID         : S-1-5-21-3383964581-1309953776-2693364552-1106
Disabled    : False
IsGroup     : False
IsDomain    : True
LastLogin   : 9/15/2016 5:19:55 PM

Server      : CLIENT2
AccountName : VLAB1/CLIENT2/MaliciousUser
SID         : S-1-5-21-2855241813-3116034789-286929080-1005
Disabled    : False
IsGroup     : False
IsDomain    : False
LastLogin   :
```

Users Removed from Privileged Group Detection

```
PS C:\Windows\system32> Get-NetLocalGroup -ComputerName CLIENT2

Server      : CLIENT2
AccountName : VLAB1/CLIENT2/Administrator
SID         : S-1-5-21-2855241813-3116034789-286929080-500
Disabled    : True
IsGroup     : False
IsDomain    : False
LastLogin   : 4/13/2014 5:37:53 PM

Server      : CLIENT2
AccountName : VLAB1/CLIENT2/testadmin
SID         : S-1-5-21-2855241813-3116034789-286929080-1001
Disabled    : False
IsGroup     : False
IsDomain    : False
LastLogin   : 8/5/2015 2:51:48 PM

Server      : CLIENT2
AccountName : VLAB1/CLIENT2/USER2
SID         : S-1-5-21-2855241813-3116034789-286929080-1002
Disabled    : False
IsGroup     : False
IsDomain    : False
LastLogin   : 8/31/2016 11:25:41 AM

Server      : CLIENT2
AccountName : VLAB1.com/Domain Admins
SID         : S-1-5-21-3383964581-1309953776-2693364552-512
Disabled    : False
IsGroup     : True
IsDomain    : True
LastLogin   :

Server      : CLIENT2
AccountName : VLAB1.com/USER2
SID         : S-1-5-21-3383964581-1309953776-2693364552-1106
Disabled    : False
IsGroup     : False
IsDomain    : True
LastLogin   : 9/15/2016 5:19:55 PM
```

```
Server      : CLIENT2
AccountName : VLAB1/CLIENT2/Administrator
SID         : S-1-5-21-2855241813-3116034789-286929080-500
Disabled    : True
IsGroup     : False
IsDomain    : False
LastLogin   : 4/13/2014 5:37:53 PM

Server      : CLIENT2
AccountName : VLAB1/CLIENT2/testadmin
SID         : S-1-5-21-2855241813-3116034789-286929080-1001
Disabled    : False
IsGroup     : False
IsDomain    : False
LastLogin   : 8/5/2015 2:51:48 PM

Server      : CLIENT2
AccountName : VLAB1/CLIENT2/USER2
SID         : S-1-5-21-2855241813-3116034789-286929080-1002
Disabled    : False
IsGroup     : False
IsDomain    : False
LastLogin   : 8/31/2016 11:25:41 AM

Server      : CLIENT2
AccountName : VLAB1/CLIENT2/MaliciousUser
SID         : S-1-5-21-2855241813-3116034789-286929080-1005
Disabled    : False
IsGroup     : False
IsDomain    : False
LastLogin   :
```

Local User Password Configuration Issues

- "Shallow copied" Local users:
  - E.g. Created via HD copy, VM export, etc.
  - Shares the same password
  - Can be identified via same username, password change time

```
PS C:\Users\user1.TESTDOMAIN.000> Get-NetLocalGroup -ComputerName "Client3Prep" -GroupName "Guests"


ComputerName : Client3Prep
AccountName  : TESTDOMAIN/Client3Prep/Guest
SID          : S-1-5-21-2937651619-4167467795-303580696-501
Description  : Built-in account for guest access to the computer/domain
Disabled     : True
IsGroup      : False
IsDomain     : False
LastLogin    :
PwdLastSet   : 9/19/2016 11:24:21 AM
PwdExpired   : False
UserFlags    : 66147

ComputerName : Client3Prep
AccountName  : TESTDOMAIN/Client3Prep/test_user
SID          : S-1-5-21-2937651619-4167467795-303580696-1006
Description  :
Disabled     : False
IsGroup      : False
IsDomain     : False
LastLogin    :
PwdLastSet   : 9/13/2016 3:30:29 PM
PwdExpired   : False
UserFlags    : 66049

ComputerName : Client3Prep
AccountName  : TESTDOMAIN/Client3Prep/new_user
SID          : S-1-5-21-2937651619-4167467795-303580696-1008
Description  :
Disabled     : False
IsGroup      : False
IsDomain     : False
LastLogin    :
PwdLastSet   : 9/18/2016 11:46:17 AM
PwdExpired   : False
UserFlags    : 66049
```

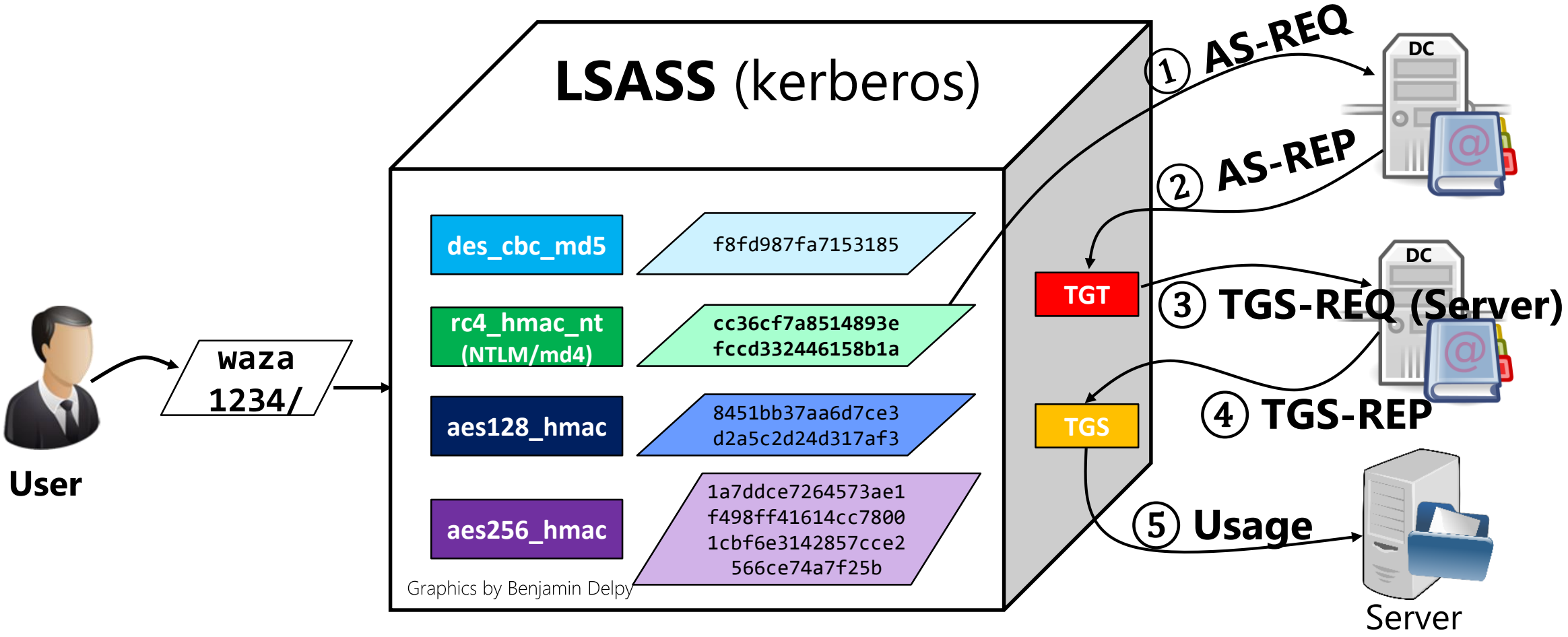
=

```
PS C:\Users\user1.TESTDOMAIN.000> Get-NetLocalGroup -ComputerName "client3" -GroupName "Guests"


ComputerName : client3
AccountName  : TESTDOMAIN/client3/Guest
SID          : S-1-5-21-473845342-2774817237-2297341088-501
Description  : Built-in account for guest access to the computer/domain
Disabled     : True
IsGroup      : False
IsDomain     : False
LastLogin    :
PwdLastSet   : 9/18/2016 11:46:47 AM
PwdExpired   : False
UserFlags    : 66147

ComputerName : client3
AccountName  : TESTDOMAIN/client3/test_user
SID          : S-1-5-21-473845342-2774817237-2297341088-1006
Description  :
Disabled     : False
IsGroup      : False
IsDomain     : False
LastLogin    :
PwdLastSet   : 9/13/2016 3:30:29 PM
PwdExpired   : False
UserFlags    : 66049

ComputerName : client3
AccountName  : TESTDOMAIN/client3/new_user
SID          : S-1-5-21-473845342-2774817237-2297341088-1008
Description  :
Disabled     : False
IsGroup      : False
IsDomain     : False
LastLogin    :
PwdLastSet   : 9/18/2016 11:46:17 AM
PwdExpired   : False
UserFlags    : 66049
```

Kerberos Error Message Injection

Kerberos Authentication

- Windows default authentication (+ authorization) protocol



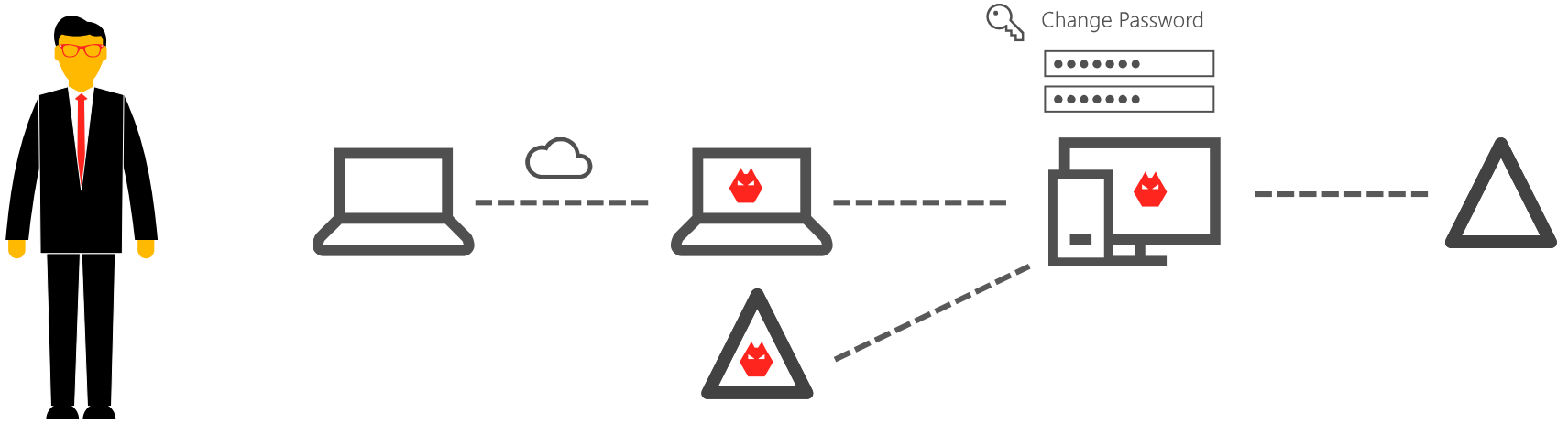
Malicious Kerberos Error Injection

- A MITM Attacker can cause the following effects by sending the following error messages:
  - Downgrade encryption: KDC_ERR_PREAUTH_REQUIRED
  - Change password: KDC_ERR_KEY_EXPIRED
  - Re-enter password: KDC_ERR_TGT_REVOKED
  - Block users: KDC_ERR_CLIENT_REVOKED

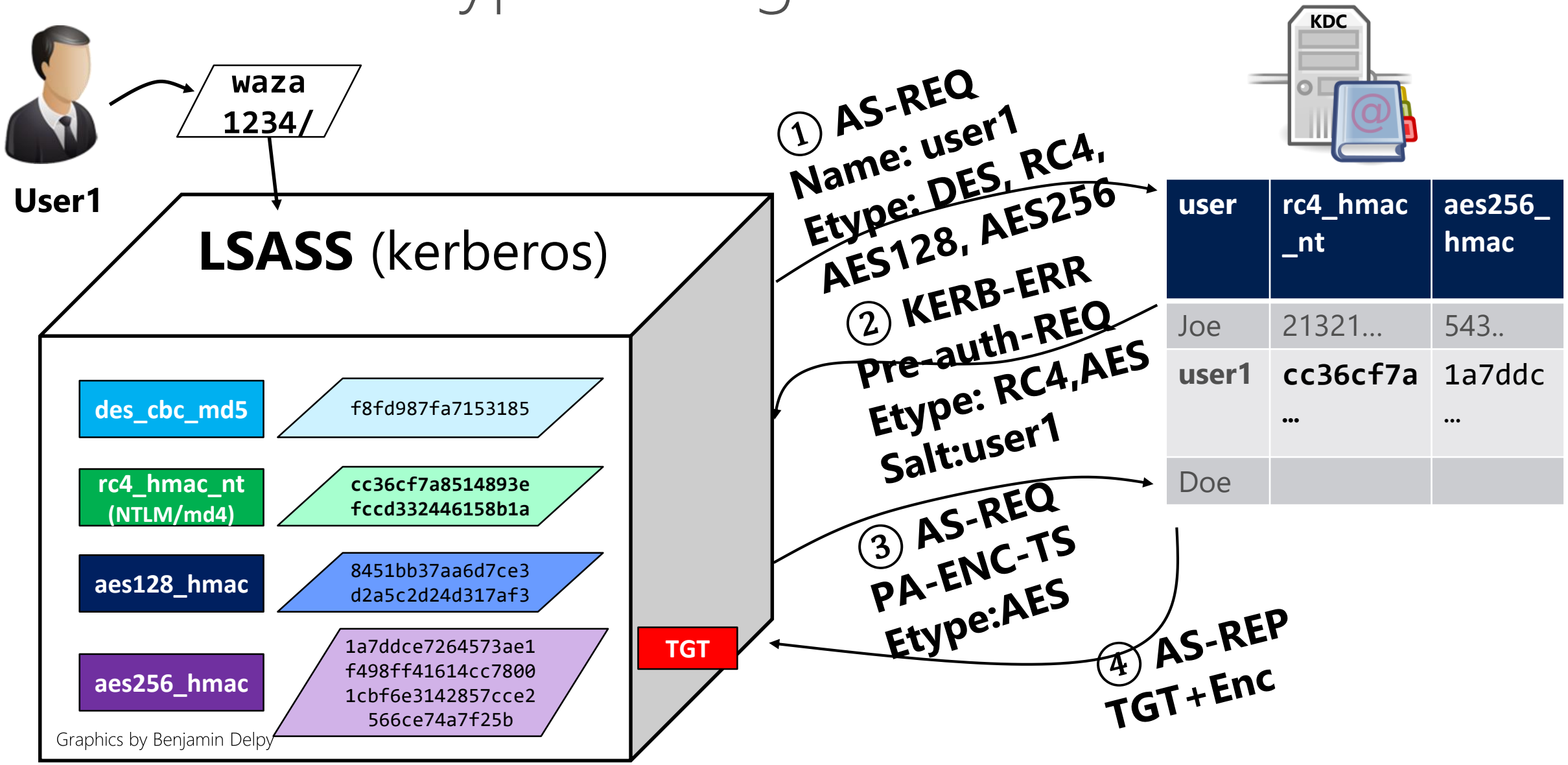
Password Expired Injection: Remote Butler

- Presented in BlackHat USA 2016 by Tal Be'ery & Chaim Hoch
- Demo: https://www.youtube.com/watch?v=CSdJ_-PhaUI

Evil Butler



Kerberos Encryption Negotiation



Kerberos Authentication: Over the Wire

```
as-req
    pvno: 5
    msg-type: krb-as-req (10)
  padata: 1 item
    PA-DATA PA-PAC-REQUEST
  req-body
      Padding: 0
    kdc-options: 40810010 (forwardable, renewable, canonicalize, renewable-ok)
    cname
      realm: aorato.research
    sname
      till: 2037-09-13 02:48:05 (UTC)
      rtime: 2037-09-13 02:48:05 (UTC)
      nonce: 160211996
    etype: 6 items
        ENCTYPE: eTYPE-AES256-CTS-HMAC-SHA1-96 (18)
        ENCTYPE: eTYPE-AES128-CTS-HMAC-SHA1-96 (17)
        ENCTYPE: eTYPE-ARCFOUR-HMAC-MD5 (23)
        ENCTYPE: eTYPE-ARCFOUR-HMAC-MD5-56 (24)
        ENCTYPE: eTYPE-ARCFOUR-HMAC-OLD-EXP (-135)
        ENCTYPE: eTYPE-DES-CBC-MD5 (3)
```

```
krb-error
    pvno: 5
    msg-type: krb-error (30)
    stime: 2014-03-10 20:05:07 (UTC)
    susec: 165032
    error-code: eRR-PREAUTH-REQUIRED (25)
    realm: aorato.research
  sname
  e-data: 30543031a103020113a22a04283026301da003020112a116..
    PA-DATA PA-ENCTYPE-INFO2
      padata-type: kRB5-PADATA-ETYPE-INFO2 (19)
        padata-value: 3026301da003020112a1161b14414f5241544f
      ETYPE-INFO2-ENTRY
          etype: eTYPE-AES256-CTS-HMAC-SHA1-96 (18)
          salt: AORATO.RESEARCHbugsb
      ETYPE-INFO2-ENTRY
          etype: eTYPE-ARCFOUR-HMAC-MD5 (23)
```

```
as-req
    pvno: 5
    msg-type: krb-as-req (10)
  padata: 2 items
    PA-DATA PA-ENC-TIMESTAMP
      padata-type: kRB5-PADATA-ENC-TIMESTAMP (2)
        padata-value: 3041a003020112a23a0438c871bc029b90195c7d2981b0cd...
          etype: eTYPE-AES256-CTS-HMAC-SHA1-96 (18)
          cipher: c871bc029b90195c7d2981b0cd8e4c98fa5fa747689f86e1...
```

AES vs. RC4: Key Derivation

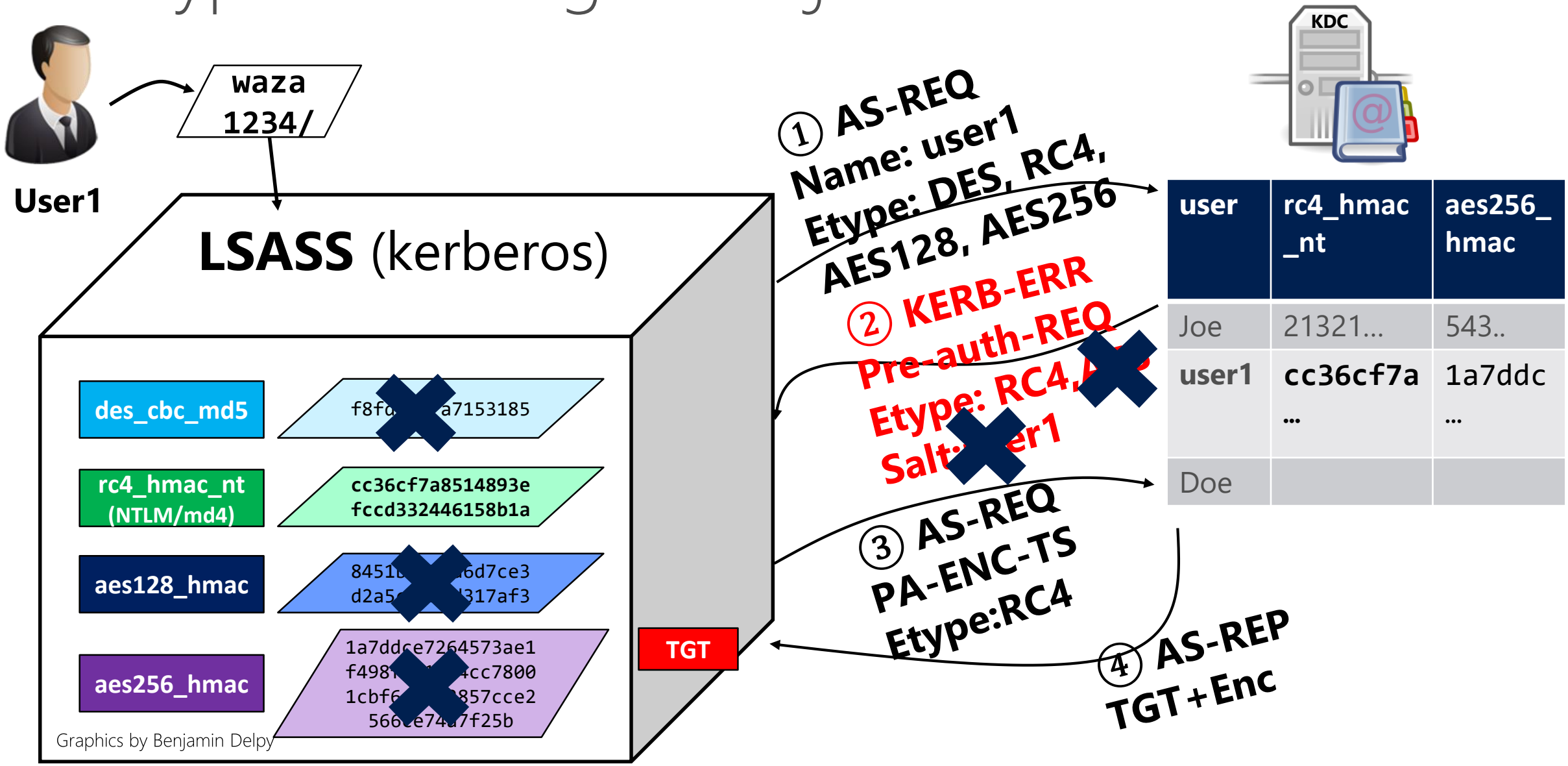
- Salting
  - Goal: Same passwords, different users = different keys
  - Create-Key(password + salt)
  - AES uses the username for salt
  - RC4-HMAC does not have any!
- "Key Stretching"
  - Goal: increase CPU load per password
  - AES uses PBKDF2= Thousands of SHA rounds
  - RC4-HMAC does not have any!

https://commons.wikimedia.org/wiki/File:Jodsalz_mit_Fluor_und_Folsaeure.jpg

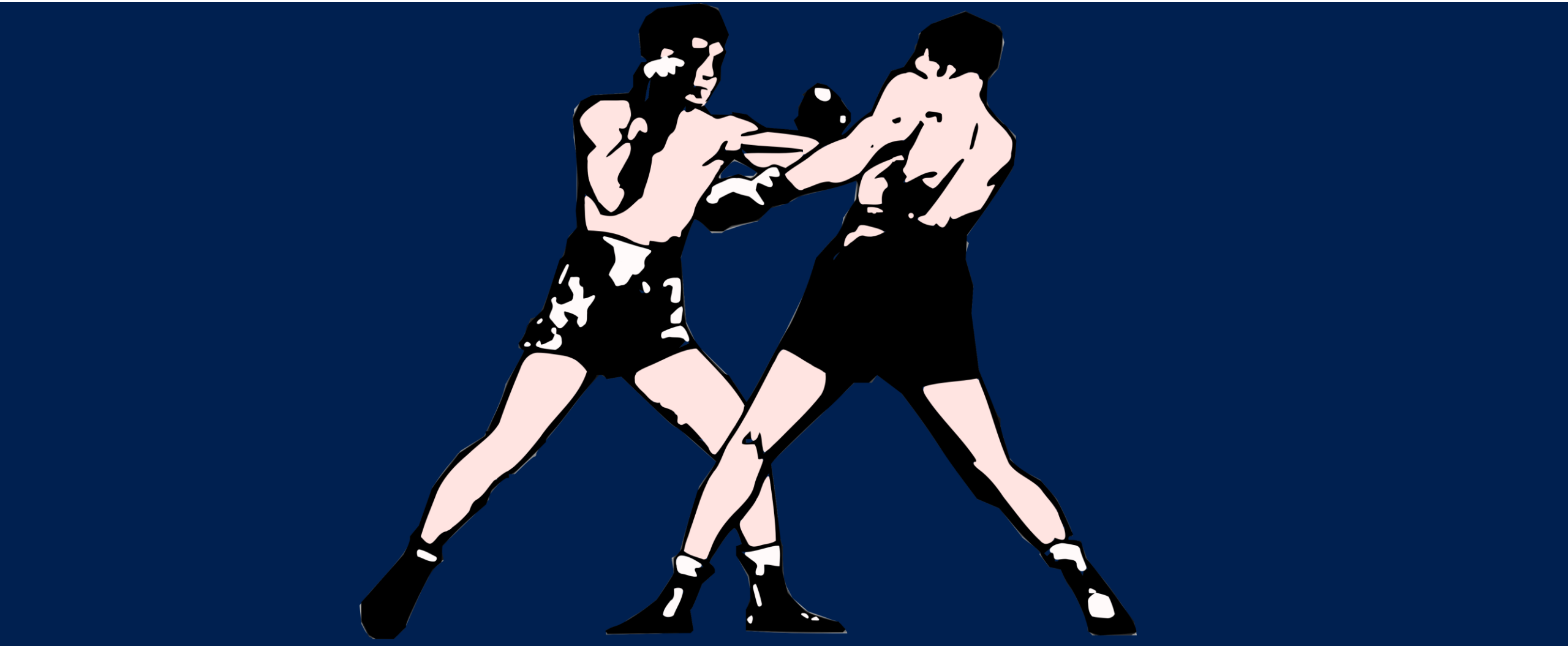
Attacker + RC4 = 

- Due to salting, identical passwords create different AES keys for different users
  - Attacker cannot prepare a rainbow table in advance
- Due to “Key Stretching” Brute Force is much more CPU intensive
- Attacker’s Solution: Downgrade to RC4

Encryption Downgrade Injection Attack



Kerberos Error Injection: Defense



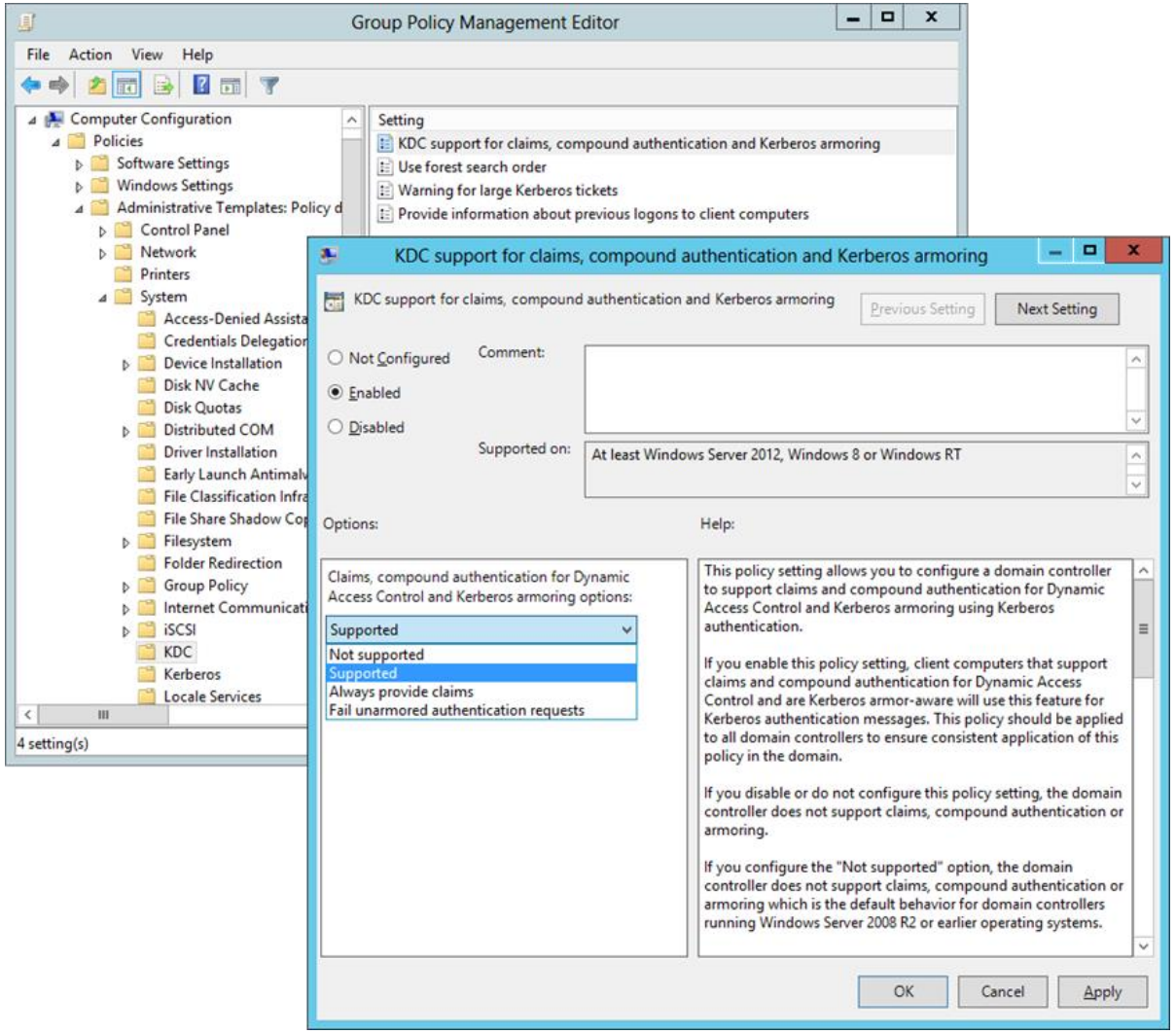
Hardening: Kerberos Armoring

- The computer's Kerberos session key protects the user's Kerberos messages
  - Kerberos errors gets signed with the computer's session key
- Prevents MITM Kerberoserror injections as attackers don't have the computer's credentials / keys

Name	Value	Bit Offset	Bit Length	Type
Length	Length: 430	0	32	Kerbero…
Message	KRB_ERROR	32	3440	Kerbero…
Pvno	5 (0x0000000000000005)	96	40	Int64
MsgType	KRB_ERROR(30) (0x000000000000001E)	136	40	MsgType
Stime	2014-03-10T22:10:56.0000000	176	152	DateTime
Susec	927784 (0x00000000000E2828)	328	56	Int64
ErrorCode	KDC_ERR_PREAUTH_REQUIRED(25) (0x0000000000000019)	384	40	ErrorCo…
Realm	aorato.research	424	152	String
Sname	krbtgt/aorato.research	576	304	Kerbero…
EData	MethodData{MethodData=[PA-FX-FAST (136)]}	880	2592	Kerbero…
MethodData	[PA-FX-FAST (136)]	0	2528	ArrayVa…
[0]	PA-FX-FAST (136)			Kerbero…
PADataType	PA-FX-FAST (136) (0x0000000000000088)	64	48	Int64
PadataValue	PA-FX-FAST-REPLY	112	2416	Kerbero…
PADataValue	KrbFastArmoredRep{EncFastRep=EncryptedData{Etype=18,Kvno=not…	0	2352	Kerbero…

Hardening: Kerberos Armoring (Cont.)

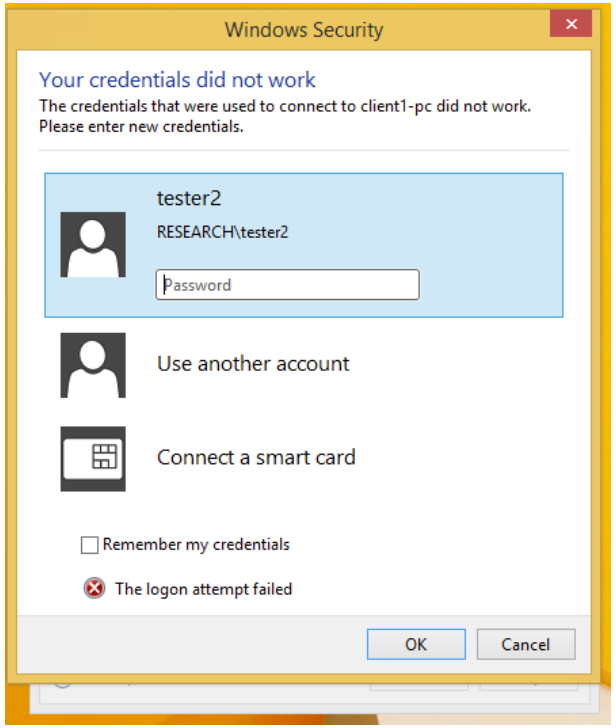
- Available since Windows 8, Server 2012

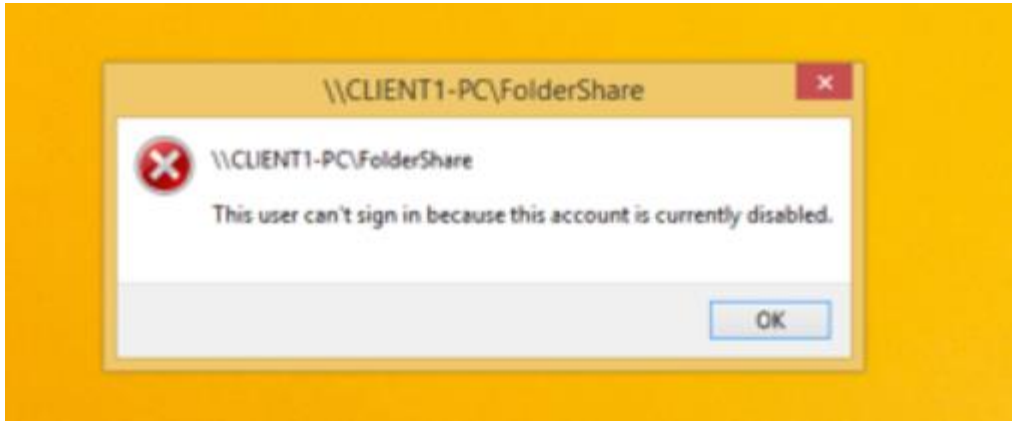


Cyber Judo with Kerberos Error Injection

Injecting to Client: Kerberos Error Injection

- Security Devices can use this method too!
  - Aorato (acquired by Microsoft) patent
  - "System, method and process for mitigating advanced and targeted attacks with authentication error injection"
- A "followed action" to detection
  - Force password change: KDC_ERR_KEY_EXPIRED
  - Force re-enter password: KDC_ERR_TGT_REVOKED
  - Elegantly block users: KDC_ERR_CLIENT_REVOKED

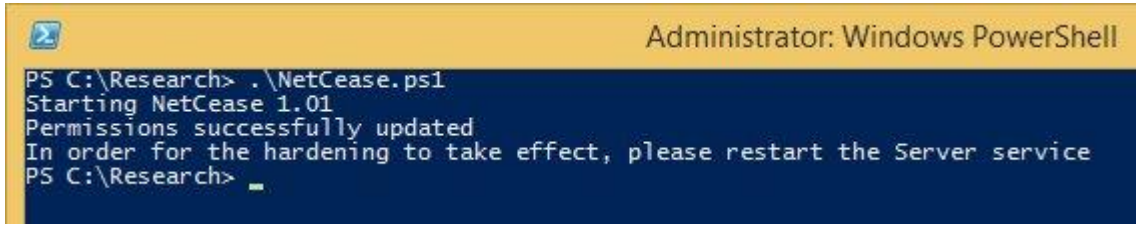


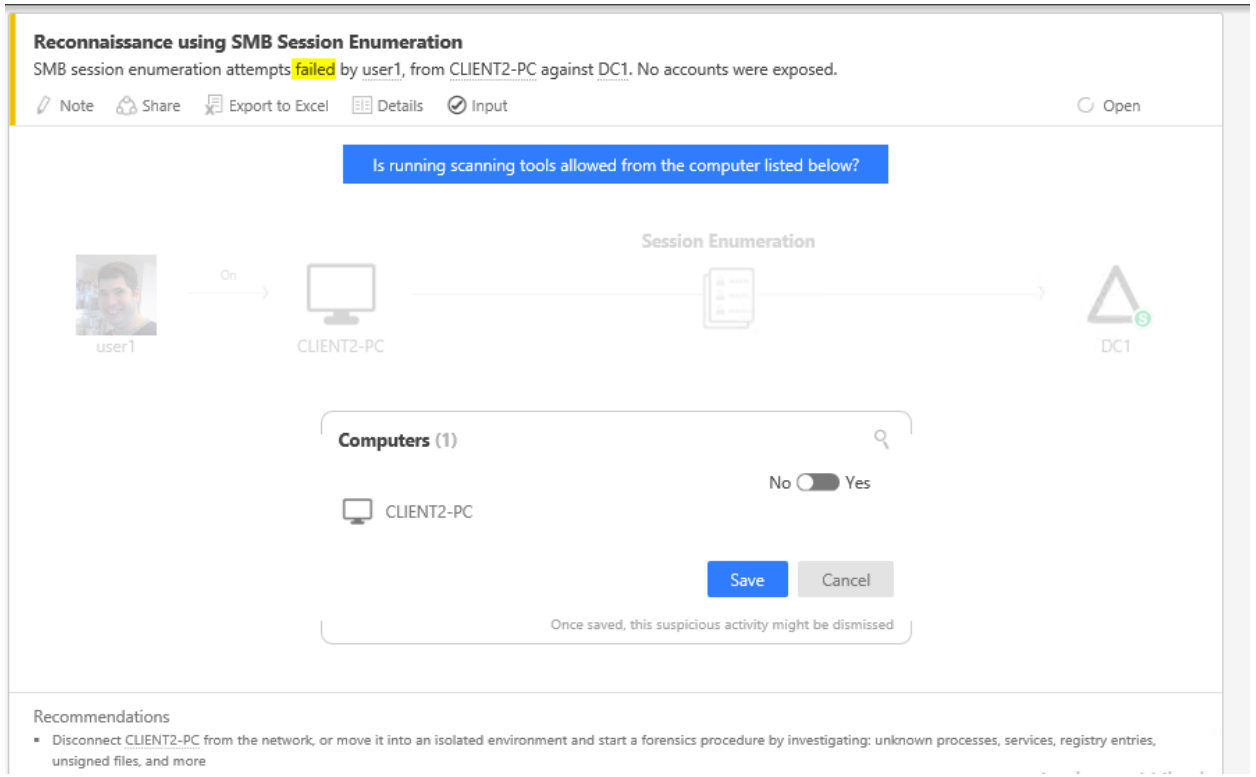


Parting Thoughts

Judo or Boxing?

- Should defenders
  - be boxers and block?
  - Be Judokas and use?
- Why not both?
- For reconnaissance APIs
  - Harden generic excessive access
    - But allow defenders' user
  - Detect failed attempts



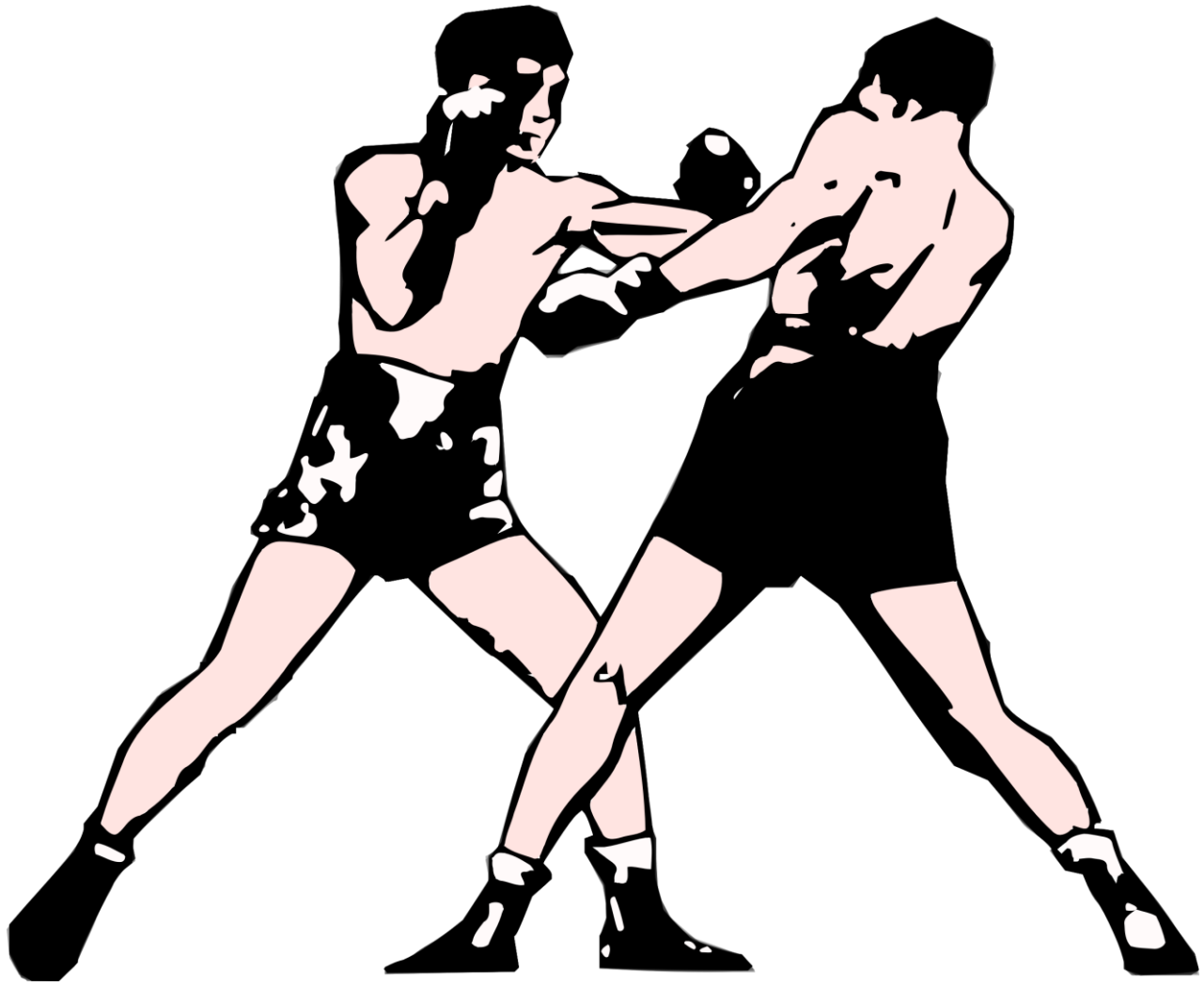


New Contributions

- Lateral Movement automation Module
- "Boxer" Defenses:
  - Detections
    - NetSess recon
    - SAMR recon
  - Hardening
    - Kerberos Armoring
    - SAMR access
- Judo defenses
  - Kerberos defensive error injection
  - NTLM authentication visibility via NetSess Recon
  - Local Users visibility via SAMR Recon

Conclusions

- To defeat the enemy
  - Learn the enemy
  - Know the enemy
  - Be the enemy



Credits and Thanks

- Reviewers
  - BloodHound's Andrew Robbins @_wald0
- Microsoft ATA Research team (other members)
  - Tal Maor @TaltheMaor
  - Marina Simakov
  - Chaim Hoch @chaimh90
- Microsoft ATA Designer
  - Dan Mor @danmor84

Questions?