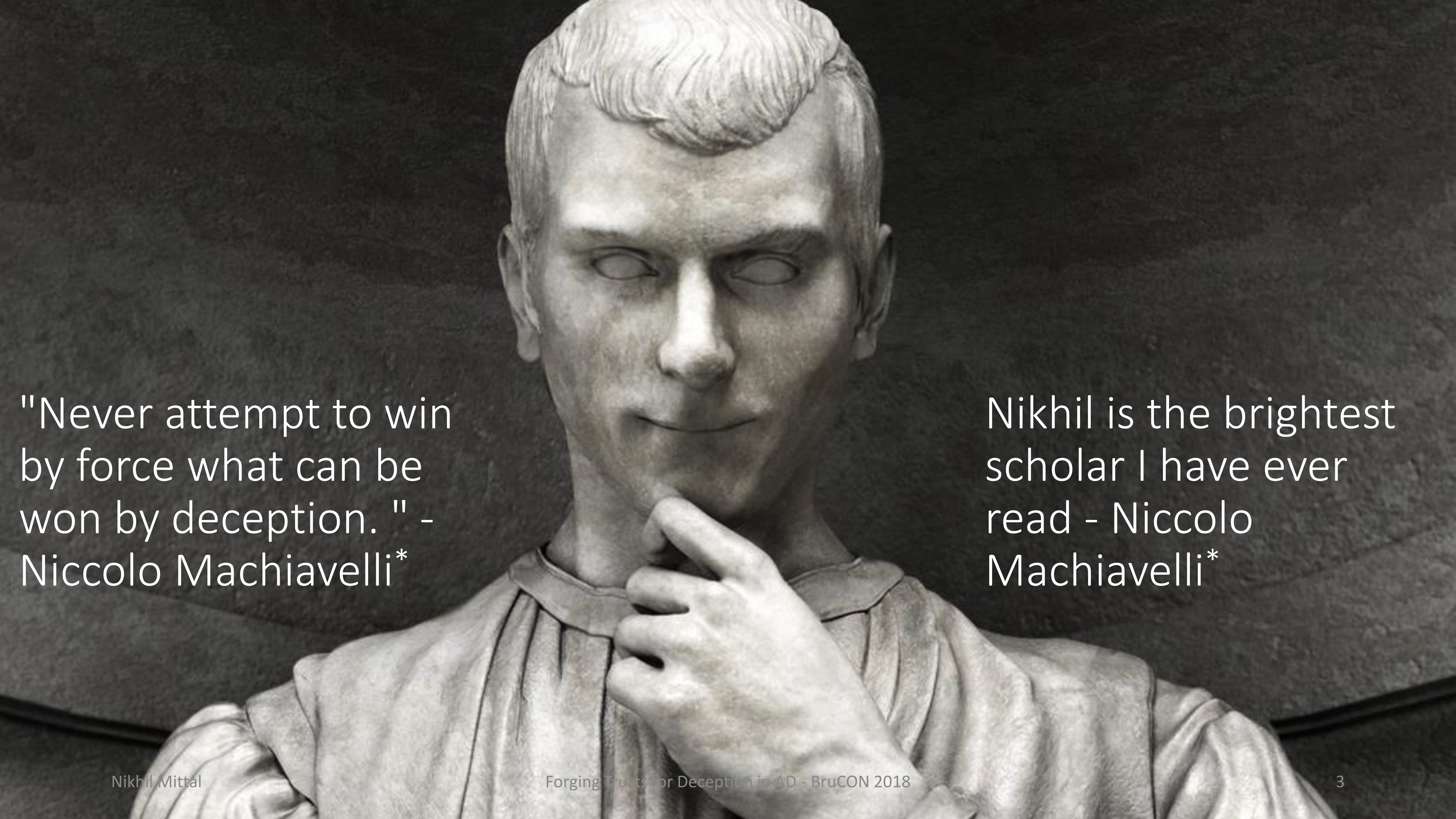
Forging Trusts for Deception in Active Directory

Nikhil Mittal

About Me

- Hacker, Red Teamer, Trainer, Speaker at PentesterAcademy
- Twitter - @nikhil_mitt
- Blog – https://labofapenetrationtester.com
- Github - https://github.com/samratashok/
- Creator of Kautilya and Nishang
- Interested in Offensive Information Security, new attack vectors and methodologies to pwn systems.
- Previous Talks and/or Trainings
  - DefCon, BlackHat, CanSecWest, BruCON, 44CON and more.

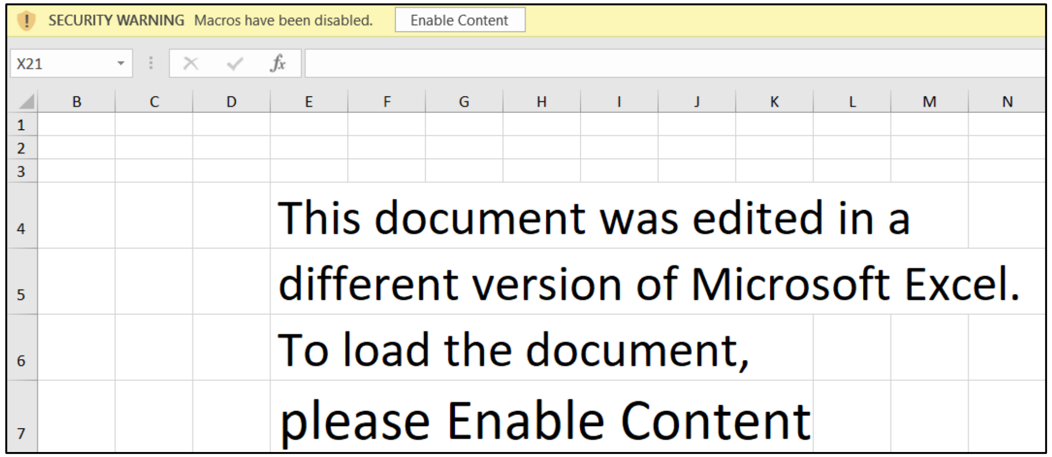


"Never attempt to win by force what can be won by deception. " - Niccolo Machiavelli*

Nikhil is the brightest scholar I have ever read - Niccolo Machiavelli*

Deception

- Deception targets with psychology of an adversary.
- Adversaries have been using deception for a long time:
  - Crafted emails tricking users to open attachments and ignoring warnings.
  - Phishing pages mimicking actual login pages.
  - Using credentials of legit users to access critical information.
  - Human Interface Devices disguised as thumb drives or other USB devices.
- Defenders and Blue Teams have used:
  - Honeypots/HoneyTokens/HoneyCreds


SECURITY WARNING Macros have been disabled.
Enable Content
X21
B C D E F G H I J K L M N
1 2 3 4 5 6 7

This document was edited in a different version of Microsoft Excel. To load the document, please Enable Content

Deception - Attacker Psychology

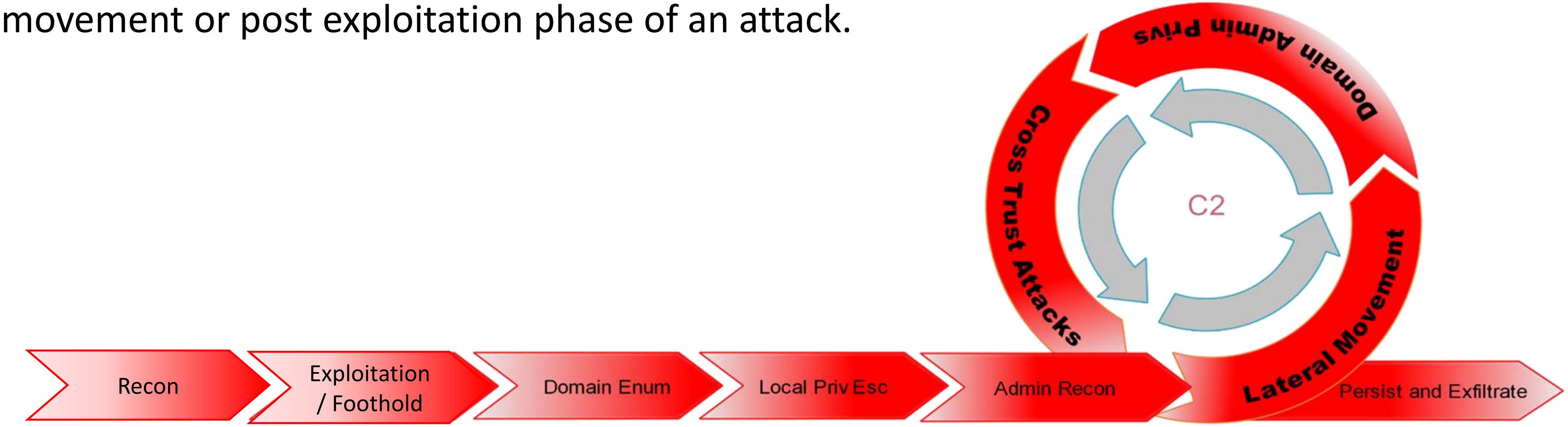
- Attackers have illusory superiority over defenders - a mental state where they think of defenders as idiots who cannot take care of their own backyard.
- Couple this with the long preached technique of "go for the lowest hanging fruit" and the urge to get Domain Admin privileges as quickly as possible and the defenders have it all set for deploying deception.

When is he going to realize that he's the mediocrity we've been talking about?

Source: https://80000hours.org/2012/11/do-you-think-you-re-better-than-average/

Deception in Active Directory

- In AD, adversaries mostly implement deception by pivoting and replaying credentials so as to appear legit in logs etc.
- Defenders, try to counter this by injecting fake or decoy credentials which when used, result in alerts.
- But, this is where deception in AD is mostly limited to. What if we want to catch an adversary before they use credentials? - We must target the enumeration phase of an attack!
- We will also look at some techniques which can be used in the both the enumeration and lateral movement or post exploitation phase of an attack.



Deception in Active Directory

- Desired properties for a decoy:
  1. Should be desirable enough so that an attacker enumerates the object.
  2. Should be easily configurable.
  3. No configuration changes required on endpoints.
  4. Should not be triggered for normal admin activity.

- Point 4 above is the hardest part. We must look for such an attribute for each type of objet which is not read by-default in system activity and possibly not by normal administrator enumeration like listing objects.

Deception in Active Directory - Enumeration

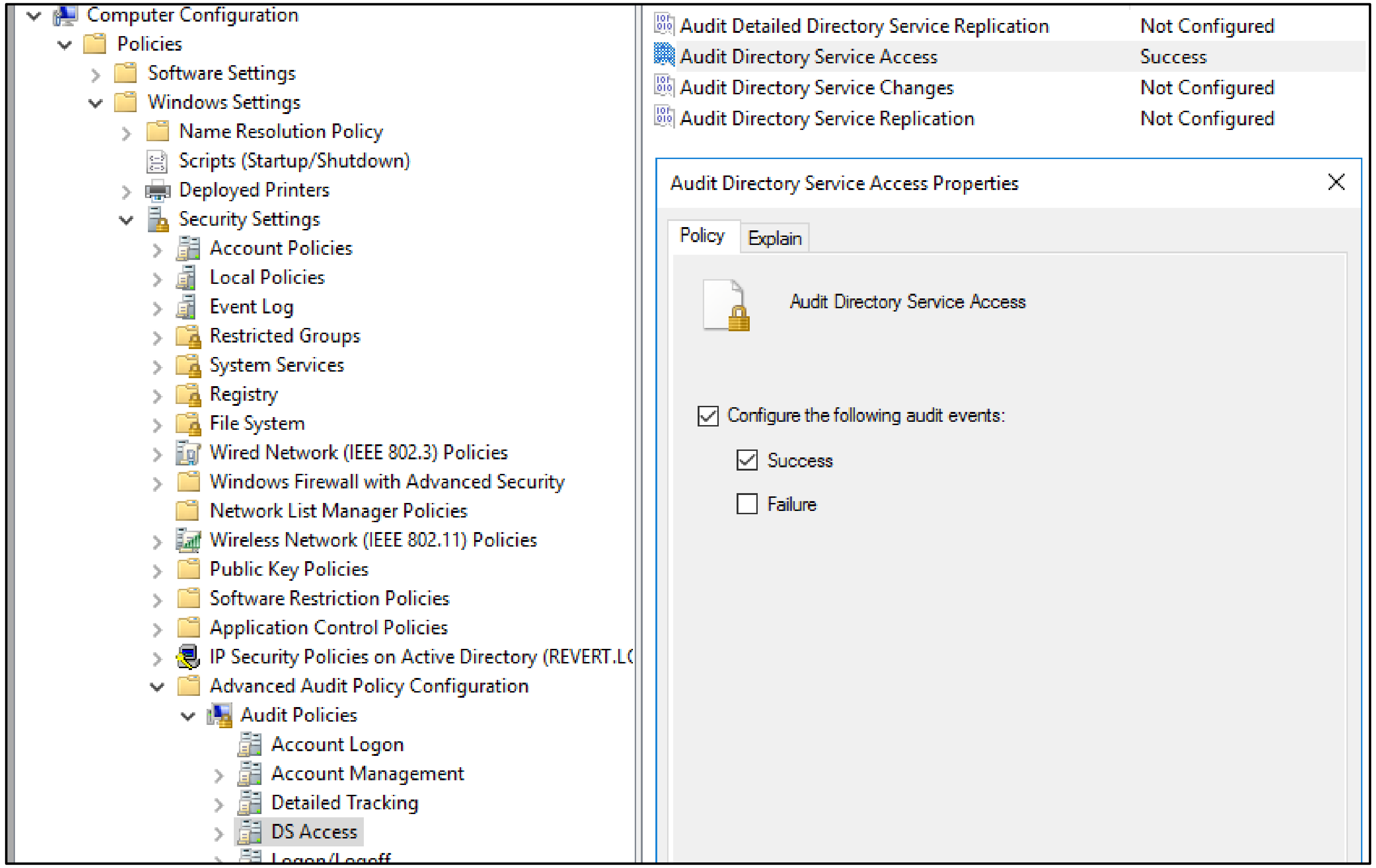
- We can trick an adversary to enumerate decoy AD objects like users, groups, OUs and computers which have aggressive logging turned on.
- This not only increases the costs to an adversary in terms of time, it also increases detection rates.
- We can use built-in tools like Group Policy, Windows Event Logging and PowerShell AD module to set it up.

Deception in Active Directory - Enumeration - User Objects

- Some of the attributes of a user object that are interesting for an adversary:
  - Password does not expire
  - Trusted for Delegation
  - Users with SPN
  - Password in description
  - Users who are members of high privilege groups
  - Users with ACL rights over other users, groups or containers

Deception in Active Directory - Enumeration - User Objects

- We can turn on Audit for Directory Service Access using Group Policy: Windows Settings -> Security Settings -> Advanced Audit Policy Configuration -> DS Access - Audit Directory Service Access.
- This setting generates a 4662 for every access to the object which has auditing turned on whenever the object is accessed.



- Computer Configuration
  - Policies
    - Software Settings
    - Windows Settings
      - Name Resolution Policy
      - Scripts (Startup/Shutdown)
      - Deployed Printers
      - Security Settings
        - Account Policies
        - Local Policies
        - Event Log
        - Restricted Groups
        - System Services
        - Registry
        - File System
        - Wired Network (IEEE 802.3) Policies
        - Windows Firewall with Advanced Security
        - Network List Manager Policies
        - Wireless Network (IEEE 802.11) Policies
        - Public Key Policies
        - Software Restriction Policies
        - Application Control Policies
        - IP Security Policies on Active Directory (REVERT.L(
        - Advanced Audit Policy Configuration
          - Audit Policies
            - Account Logon
            - Account Management
            - Detailed Tracking
            - DS Access
            - Logon/Logoff

Audit Detailed Directory Service Replication	Not Configured
Audit Directory Service Access	Success
Audit Directory Service Changes	Not Configured
Audit Directory Service Replication	Not Configured

Audit Directory Service Access Properties

Audit Directory Service Access

☑ Configure the following audit events:

☑ Success

☐ Failure

Deception in Active Directory - Enumeration - User Objects

- Auditing needs be set on a decoy user object. A quick look at the structure of SYSTEM_AUDIT_ACE and AddAuditAccessObjectAce function is useful:

```
BOOL AddAuditAccessObjectAce(
  PACL  pAcl,
  DWORD dwAceRevision,
  DWORD AceFlags,
  DWORD AccessMask, // Rights for which Auditing is done
  GUID  *ObjectTypeGuid,
  GUID  *InheritedObjectTypeGuid,
  PSID  pSid, // SID of Trustees for whom auditing is enabled
  BOOL  bAuditSuccess, // Audit successful use of a right
  BOOL  bAuditFailure // Audit failure to use a right
);
```

Deception in Active Directory - Enumeration - User Objects

- A user is created with some "interesting" flags and auditing is set on it - For what rights? For auditing Enumeration these are useful ones:
  - GENERIC_READ
  - READ_CONTROL (Read DACL)
  - Specific rights like Read only a particular - a bit obscure - property. For example, read x500uniqueIdentifier.

Deception in Active Directory - Enumeration - User Objects

- How to automate this?
- Introducing **Deploy-Deception**! A simple PowerShell module (uses the ActiveDirectory module cmdlets) which can be used to create decoy objects, set interesting attributes, turn on auditing for different Active Directory objects.

  https://github.com/samratashok/
- Deploy-Deception uses AI and ML to learn adversary behaviour and then uses next-gen Blockchain to compare with our resilient database of adversary behaviour from 10 million+ endpoints :)

Deception in Active Directory - Enumeration - User Objects

- Let's create a decoy user "usermanager" whose password never expires and turn on GenericRead for "Everyone":

  ```
  Create-DecoyUser -UserFirstName user -UserLastName
  manager -Password Pass@123 | Deploy-UserDeception -
  UserFlag PasswordNeverExpires -Verbose
  ```

- Please remember that an actual user is created on the DC where the above command is executed. Please remember to document this user's creation.

Deception in Active Directory - Enumeration - User Objects

- A GenericRead/ReadProperty triggers a 4662 in all cases even when the decoy user is not specifically enumerated. For example, following commands trigger a 4662 for decoy user called usermanager.

  `net user /domain`

  `Get-WmiObject -Class win32_UserAccount`

  `Get-ADUser -Filter *` (ActiveDirectory module)

  `Get-NetUser` (PowerView and other LDAP based tools)

  Find Users, Contacts, and Groups GUI

Deception in Active Directory - Enumeration - User Objects

- A better use case is to get a log entry only when an obscure/uncommon property is read.
- Let's create a decoy user "usermanager-uncommon" whose password never expires and turn on auditing x500uniqueIdentifier is read for "Everyone":

```
Create-DecoyUser -UserFirstName user -UserLastName
manager-uncommon -Password Pass@123 | Deploy-
UserDeception -UserFlag PasswordNeverExpires -GUID
d07da11f-8a3d-42b6-b0aa-76c962be719a -Verbose
```

Deception in Active Directory - Enumeration - User Objects

- For the previous decoy user, only LDAP based tools like PowerView, ADExplorer etc. trigger 4662.
- Tools which use LDAP or other offensive tools fetch all the information in a single attempt which makes them stand out.
- Since we are targeting very basic enumeration which means there is a lot of noise, this is useful for filtering out some of the noise.
- Results are quite similar for user SPN (You may like to use a Kerberoast-able password when targeting lateral movement):

```
Create-DecoyUser -UserFirstName user -UserLastName
manager-spn -Password Pass@123 | Deploy-UserDeception
-SPN 'dc/MSSQLSvc' -GUID f3a64788-5306-11d1-a9c5-
0000f80367c1 -Verbose
```

Deception in Active Directory - Enumeration - User Objects

- Let's filter the noise a bit more by auditing for ReadControl (read DACL) property:

  ```
  Create-DecoyUser -UserFirstName user -UserLastName
  manager-control -Password Pass@123 | Deploy-
  UserDeception -UserFlag
  AllowReversiblePasswordEncryption -Right ReadControl -
  Verbose
  ```

- The above is triggered only when either DACL or all attributes are read for the user.

Deception in Active Directory - Enumeration - Computers

- Computer objects can be created or modified to work as decoys.
- It is better to use actual computers as decoys to avoid easy identification. Decoy computers should either be VMs or turned off after joining the domain unless they are used as honeypots.
- What computers the attackers are interested in?
  - Older Operating Systems
  - Interesting SPN
  - Delegation Settings
  - Membership of privileged groups

Deception in Active Directory - Enumeration - Computers

- Crete a computer object for auditing whenever x500uniqueIdentifier is read:

```
Create-DecoyComputer -ComputerName revert-web -Verbose
| Deploy-ComputerDeception -PropertyFlag
TrustedForDelegation -GUID d07da11f-8a3d-42b6-b0aa-
76c962be719a  -Verbose
```

- Modify a computer object for auditing whenever its DACL is read:

```
Deploy-ComputerDeception -DecoyComputerName comp1 -
PropertyFlag TrustedForDelegation -Right ReadControl -
Verbose
```

- We can also use - with limited success- DCShadow (thanks to Benjamin and Vincent) to mimic a Domain Controller. See: https://www.labofapenetrationtester.com/2018/04/dcshadow.html

Deception in Active Directory - Enumeration - Groups

- Groups are, of course, interesting to attackers. We can have decoy groups with logging enabled for interesting activity like when Group Membership is read or Group members are read or an obscure property like x500uniqueIdentifier or the DACL is read.
- We can make a Group, a member of other interesting groups.
- We can also create decoy users and make them member of the decoy group we are creating.
- An example for auditing when the DecoyGroup DACL is read:

```
Create-DecoyGroup -GroupName "Forest Admins" -Verbose
| Deploy-GroupDeception -AddMembers -Members slaveuser
-AddToGroup -AddToGroupName dnsadmins -Right
ReadControl -Verbose
```

Deception in Active Directory - Enumeration - Groups

- Here is an example which logs 4662 when Group membership property set is read:

  ```
  Create-DecoyGroup -GroupName "Forest Admins" -Verbose
  | Deploy-GroupDeception -AddMembers -Members slaveuser
  -AddToGroup -AddToGroupName dnsadmins -GUID bc0ac240-
  79a9-11d0-9020-00c04fc2d4cf -Verbose
  ```

- Similar to Groups for User objects we can create decoy OUs for Users and Computer objects.

Deception in Active Directory - Lateral Movement - Users

- Couple of very interesting techniques which are also usable with the popular honey-credentials method. Make the decoy user a part of the domain admins or other privileged group or rights like DCSync :
  - Set the Logon Workstation to a non-existent machine
  - Deny logon to the user.
- In both the above cases, even with valid credentials, an adversary cannot abuse the credentials.
- With Audit Kerberos Authentication Service with Audit Failure enabled, a 4768 is logged every time someone tries to use that user.
- Such a decoy user will also be very interesting for enumeration!

Deception in Active Directory - Lateral Movement - Users

- Create a decoy user who is member of the domain admins group and is denied logon:

```
Create-DecoyUser  -UserFirstName dec -UserLastName da
-Password Pass@123 | Deploy-PrivilegedUserDeception -
Technique DomainAdminsMemebership -Protection
DenyLogon -Verbose
```

- To enable Directory Access (4662) auditing on the above user:

```
Deploy-UserDeception -DecoySamAccountName decda -GUID
d07da11f-8a3d-42b6-b0aa-76c962be719a -Verbose
```

Deception in Active Directory - Lateral Movement - Users

- Another interesting technique is to provide a "master" user FullControl over a "slave" user. This makes both the master and slave users interesting for an adversary looking at ACLs.
- Like the previous one, this technique is also useful in both the enumeration and lateral movement phase.
- For targeting lateral movement, we can make either slave or master or both privileged users, set SPN or any other flag we saw in Deploy-UserDeception.

Deception in Active Directory - Lateral Movement - Users

- Create a slave user and set FullControl over it for a master user for targeting enumeration.

```
Create-DecoyUser -UserFirstName master -UserLastName
user -Password Pass@123
```

```
Create-DecoyUser -UserFirstName slave -UserLastName
user -Password Pass@123 | Deploy-SlaveDeception -
DecoySamAccountName masteruser -Verbose
```

- To target lateral movement, as an example, we can set auditing whenever master user changes the ACL of slave:

```
Deploy-UserDeception -DecoySamAccountName slaveuser -
Principal masteruser -Right WriteDacl -Verbose
```

Deception in Active Directory - Lateral Movement - Users

- To target lateral movement, for any existing "honeyuser", set auditing whenever honeyuser is used to interact with the slaveuser :

```
Deploy-UserDeception -DecoySamAccountName slaveuser -
Principal honeyuser -Right ReadProperty -Verbose
```

Deception in Active Directory - Lateral Movement - Forest Trusts

- For forest trusts, I could not find a way yet, to automate setting up a decoy one.
- There are many interesting approaches to this. We can have a complete forest (blue forest?) full of decoy users and computers.
- A bi-directional trust can be established with that decoy forest with Selective Authentication.
- By configuring selective authentication, we can allow enumeration of all users and computers in the decoy forest but no access to any other resource!

Red Team Revenge - Identifying Deception

- I have seen multiple Enterprise solutions which do not use actual objects and can be spotted by looking at object properties like:
  - objectSID
  - lastLogon, lastlogontimestamp
  - Logoncount
  - whenCreated
  - Badpwdcount
  - Compare with known actual objects
- No screenshots because NDA!

© Photoshot

Red Team Revenge - Identifying Deception

- Some Enterprise solutions also fill up ALL possible attributes for an object which can be easily spotted by comparing attributes with a real computer, say, the domain controller.
- In an assume breach scenario or from a foothold box, you can always get the actual DC by looking at logonserver env variable. Use the DC or your own computer object's properties to compare properties of other computers.
- For multiple solutions, using WMI for retrieving information lists only the actual objects and not the fake ones.
- No screenshots because NDA!

Red Team Revenge - Avoiding Deception

- Red teams need to change their approach to avoid detection by deception.
- Please stop going for the lowest hanging fruit. Enterprise networks are mess but if something looks to good to be true, investigate carefully!
- Avoid automated enumeration tools unless you absolutely know what they are doing in the background.
- I have been urging this in my talks (on ATA) and trainings - Avoid the urge to go for the DA privileges so that you can brag about it in the reports! Focus on goals of your operation.

Blue Teams - Avoiding identification

- Using actual AD objects helps in avoiding detection. Deploy-Deception uses actual AD objects which means they may not stand out in very first attempt. At least, that is the idea :)
- Specifically for user decoys, it makes sense to create a logon to avoid funny entries in lastlogon, lastlogontimestamp and logoncount. Even more for Domain Admin decoys!

Blue Teams - Avoiding identification

- Deploy-Deception addresses this to a limited extent by starting (and stopping) a process as the decoy DA when LogonWorkstation is set to one of the DCs. This fills up the "suspicious" properties:

```
Create-DecoyUser -UserFirstName test -UserLastName da
-Password Pass@123 | Deploy-PrivilegedUserDeception -
Technique DomainAdminsMemebership -Protection
LogonWorkStation -LogonWorkStation revert-dc -Create
Logon -Verbose
```

- Please be warned that the CreateLogon option in above command will also create a profile for the decoy DA on the DC.

Future Work

- Fine tuning the logging - Want to get involved but does not know/want to code. Test the deployment and let me know how many false positives to you get (I think a lot initially)
- Deploying decoy domain and forest objects by modifying active directory schema.
- Utilizing virtualization to deploy decoy computers and forests in real time.

Conclusion

- By targeting the adversary mind-set we can trick them to interact with decoys which increases their cost in form of time and chances of detection.
- Deception can be utilized in an AD environment to detect multiple phases of an attack cycle.
- By cleverly using existing tools like ActiveDirectory module we can deploy deception which is - if not betters - matches the detection by paid solutions.

Thank you

- Please leave feedback.

- Follow me @nikhil_mitt
- For questions, training, assessments - nikhil.uitrgpv@gmail.com nikhil@pentesteracademy.com

- Slides and blog posts will be posted on my blog and Github http://labofapenetrationtester.com/ https://github.com/samratashok