netwrix

Netwrix Attack Catalog

Attack-to-Product Mapping

netwrix

Table of contents

Introduction

The Netwrix Attack catalog is designed to educate IT and security professionals to be a useful, educational asset for those looking to understand the specific tactic, techniques, and procedures (TTPs) attackers are leveraging to compromise credentials and data.

Additionally, this document is designed to give an understanding of how Netwrix products help to detect, mitigate, and prevent these attacks.

Pass the Hash

Threat (Lateral Movement)

Pass the Hash is a technique that enables an attacker (typically using Mimikatz) to leverage the LanMan or NTLM hashes of a user's password – instead of the user's plaintext password – to authenticate to a directory or resource.

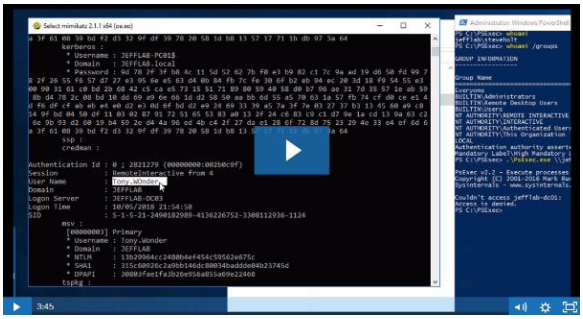
How Pass the Hash Works

- **Step 1** – An attacker obtains the password hashes of one or more users on a computer network. This is typically accomplished by extracting password hashes that exist in memory on a compromised machine or via more advanced attacks that have already taken place, such as the successful compromise of Active Directory's Ntds.dit file.
- **Step 2** – Using Mimikatz or a similar tool, the attacker leverages the compromised user's username and password hash to authenticate to other systems or resources that account has access to.

NOTE: Successful execution of a Pass the Hash attack does not necessarily grant the attacker elevated or privileged access rights to network resources. The attack, while serious, only provides a way for the attacker to obtain the equivalent of a user's password without having to actually know the plaintext password. Ultimately, the attacker only has as much access as the account they've compromised.

That said, savvy attackers can effectively utilize lower profile accounts leverage Pass the Hash to elude detection, making it much more difficult to spot signs of compromise.

Video Tutorial - https:// youtube.com/681324174



Solutions with Netwrix

Detection

Detection of pass-the-hash attacks is challenging. While the attacker is bypassing the password validation step of the authentication process by using stolen NTLM hashes, the actual network authentication that is performed is valid. Here are some of the techniques to best detect pass-the-hash attacks:

Detection	Netwrix Product	Details
Honey Tokens	Netwrix StealthDEFEND	Leverage Honey Tokens to inject fake credentials into LSASS memory on target machines and monitor for the usage of those credentials. If you see these credentials in use, it is conclusive that they were retrieved from memory on one of the honeypot machines and used for lateral movement.

Detection	Netwrix Product	Details
Abnormal Behavior	Netwrix StealthDEFEND	By baselining normal user behavior and looking for anom-alous usage of accounts it is possible to detect pass-the-hash and other lateral movement attacks. Typical behavior to look for includes: ▪ Account being used from host(s) it has never authenticated from before ▪ Account being used to access host(s) it has never before accessed ▪ Accessing a large number of hosts across the network that contradicts normal access patterns

Mitigation

There are several things that can be done to mitigate against Pass the Hash. At a high level, you want to accomplish two things:

- Prevent the password artifacts (E.g. NTLM hashes) of privileged accounts from being stored on unprivileged systems (e.g. Domain Admin shouldn't log onto a workstation)
- Restrict users from obtaining administrative privileges on their workstations where, if compromised, their accounts can be used to retrieve password artifacts from disk/memory

Approach	Netwrix Product	Details
Reduce Administrator Rights	Netwrix StealthAUDIT	One of the most impactful ways to reduce the risk of privileged access is to minimize the administrative rights on servers and desktops. Users should not log into their workstations with administrative rights. ▪ Report on what users have administrative rights on workstations through direct and nested membership in the Administrators group ▪ Perform regular reviews of Administrator group membership within the Access Information Center and remove unnecessary members ▪ Report on Administrative equivalent rights on desktops and workstations through user rights such as Act as Part of the Operating System (SeTcbPrivilege)
PowerShell Monitoring	Netwrix StealthAUDIT	PowerShell is a popular technique for performing credential extraction and pass-the-hash. Monitoring for suspicious PowerShell commands can detect pass-the hash and the use of credential extraction tools such as Mimikatz.
Logon Rights	Netwrix StealthAUDIT	As a best practice you should restrict highly privileged accounts from logging onto lower privilege systems. For example, domain administrators should not log onto workstations, because their password artifacts will be left in memory and can be vulnerable if that workstation is compromised. Netwrix StealthAUDIT can help by reporting on the log-on restrictions enforced through user rights assignments (e.g. Allow Log On Through Remote Desktop Services). Netwrix StealthAUDIT can also be used to review log-on policies that can restrict local accounts such as the Administrator account from being used for network access which is a common approach for pass-the-hash.

Approach	Netwrix Product	Details
LSA Protection	Netwrix StealthAUDIT	Netwrix StealthAUDIT can help ensure LSA Protection is enabled on all systems Windows 8.1 / Server 2012 R2 and higher. This makes it more difficult to extract credentials from LSASS.

AdminSDHolder Modification

Threat (Persistence)

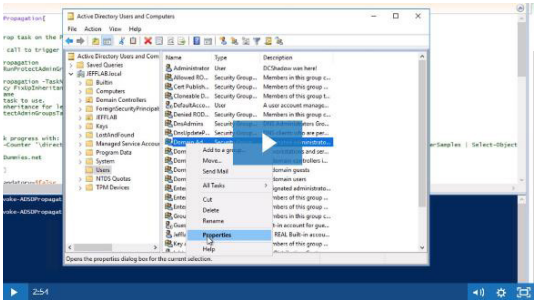
Modifying the Access Control List (ACL) of the AdminSDHolder container in Active Directory (AD) enables an attacker to achieve and maintain persistence in an already compromised domain, even if an administrator finds and removes the attacker's permission on a protected object the AdminSDHolder controls.

How the AdminSDHolder Modification Attack Works

- **Step 1** – An attacker compromises privileged credentials (e.g. via phishing or social engineering, is already a privileged insider, by exploiting weak permissions, etc.).
- **Step 2** – The attacker modifies AdminSDHolder by adding a new user to its ACL.
- **Step 3** – Via Security Descriptor propagator (SDProp), the AdminSDHolder permissions are pushed down to all protected objects every 60 minutes by default, including privileged groups such as Domain Admins, Administrators, Enterprise Admins, and Schema Admins. Even if an administrator sees an inappropriate permission on a protected object and removes it, within an hour those permissions will be put back in place by SDProp.

NOTE: The ACL of the AdminSDHolder object is used as a template to copy permissions to all "protected groups" and their members in AD. Protected groups and their members are flagged in AD using an attribute called "admin-Count", which will be set to 1 for protected users and groups. Once a user is removed from a privileged group, they still maintain the adminCount value of 1, but are no longer considered a protected object in AD. That means the AdminSDHolder permissions will not be applied to them. However, they will likely have a version of the AdminS Holder permissions still set, because the inheritance of their permissions will still be disabled as a remnant of when they were protected by the AdminSDHolder permissions. Therefore, it is still useful to look at these objects and, in most cases, to turn on inheritance of permissions.

Video Tutorial - https:// youtube.com/678104936



Solutions with Netwrix

Detection

Detection of AdminSDHolder is straightforward and involves monitoring for changes to the Access Control List for this container.

Approach	Netwrix Product	Details
Permissions Change Detection	Netwrix StealthDEFEND / Netwrix StealthINTERCEPT	Monitor for changes to the ACL of the AdminSD-Holder container ("CN=AdminSDHolder, CN=System,DC=domain,DC =com,") in all domains.

Prevention

Using blocking policies can prevent even administrative accounts from modifying the ACL of the AdminSDHolder container, ensuring this cannot be used for a persistence technique by an attacker.

Approach	Netwrix Product	Details
Permissions Change Blocking	Netwrix StealthINTERCEPT	Block all changes to the ACL of the AdminSDHolder container ("CN=AdminSDHolder,CN=System,DC=domain, DC=com,") in all domains.

Mitigation

In addition to monitoring for changes and blocking them going forward, it is best to perform an initial review and cleanup of the AdminSDHolder rights to ensure no inappropriate Access Control Entries exist.

Approach	Netwrix Product	Details
Permissions Clean Up	Netwrix StealthAUDIT	Report on the AdminSDHolder permissions and ensure that only trusted accounts have access to change permissions (WriteDACL) as well as cleaning up any inappropriate permissions where lower level accounts may have access to change the objects protected by AdminSDHolder.

DCShadow

Threat (Persistence)

DCShadow enables an attacker (using Mimikatz) to create a fake Active Directory Domain Controller (DC) that can replicate malicious changes to legitimate DCs.

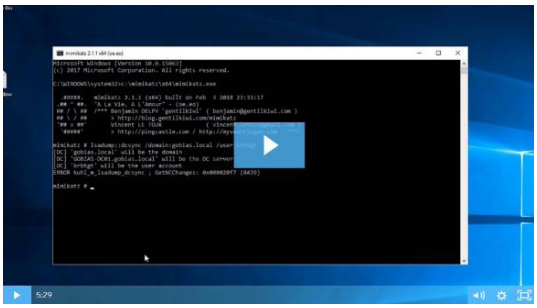
DCShadow attacks are difficult to detect. Because the changes are committed through replication, these changes are not logged to the event log the way other changes would be. The DC is where changes normally originate, but in this case there is no actual DC.

DCShadow attacks are difficult to prevent. The DCShadow attack uses native features of Active Directory (AD), so it is not a vulnerability and cannot be patched.

How the DCShadow Attack Works

- **Step 1** – An attacker obtains Domain Admin rights and wants to make changes that will not be detected to create persistence.
- **Step 2** – Using DCShadow, a feature of Mimikatz, the attacker will register the computer it is run from as DC in Active Directory by making changes to the AD's Configuration schema and the workstation's Service Principal Name (SPN) values. Now AD thinks this workstation is a DC, and it is trusted to replicate changes.
- **Step 3** – A change is crafted by the attacker. The workstation makes this change available to a legitimate DC through replication.
- **Step 4** – Replication is triggered by DCShadow, and the change is replicated and then committed by a legitimate DC.

Video Tutorial - https:// youtube.com/678105528



Solutions with Netwrix

Detection

Detection of DCShadow is possible by looking for the process of registering any system other than a Domain Controller with the required SPNs to perform the attack.

Approach	Netwrix Product	Details
Domain Controller Impersonation	Netwrix StealthDEFEND / Netwrix StealthINTERCEPT	Monitor for modification to the SPN values for any computers not in the Domain Controllers Group or OU with values including: ▪ Any value starting with GC/ ▪ The well-known GUID of the DRS service class E3514235–4B06–11D1-AB04–00C04FC2DCD2

Mitigation

The ability to perform the DCShadow attack requires elevated rights within Active Directory, typically those of a Domain Administrator. The best mitigation is to protect and closely monitor your Domain Admins and other privileged groups. However, it is also possible to perform DCShadow using a least privilege model and therefore permissions on Active Directory should be inspected to ensure no unnecessary users have these elevated rights.

Approach	Netwrix Product	Details
Protect Privileged Groups	All	"Administrators", "Power Users", "Account Operators", "Server Operators", "Print Operators", "Backup Operators", "Replicators", "Network Configuration Operators", "Incoming Forest Trust Builders", "Domain Admins", "Domain Controllers", "Group Policy Creator Owners", "read-only Domain Controllers", "Enterprise Read-only Domain Controllers", "Schema Admins", "Enterprise Admins", "DnsAdmins", "DHCP Administrators"
Active Directory Domain Permissions	Netwrix StealthAUDIT Netwrix StealthINTERCEPT Netwrix SbPAM Netwrix StealthDEFEND	Review the following domain permissions to make sure you approve all authorized users/groups: ▪ Add/Remove Replica in Domain (DS-Install-Replica) ▪ Manage Replication Topology (DS-Replication-Manage-Topology) ▪ Replication Synchronization (DS-Replication-Synchronize) As well as these permissions on the Sites object (CN=Sites,CN=Configuration,DC=domain,DC=com): ▪ Create all child objects ▪ Delete all child objects Remove any unnecessary permissions.

Golden Ticket

Threat (Persistence)

By obtaining the password hash for the KRBTGT account, the most powerful service account in Active Directory (AD), an attacker is able to get unlimited and virtually undetectable access to any system connected to AD.

Golden Tickets are very difficult to detect. The parameters the attacker can use to generate a Golden Ticket do not have to be real. The User account name and the Relative ID (RID) of the account can be real or fake, depending on what the attacker is looking to accomplish. When configuring the groups the impersonated account will belong to, Mimikatz includes the Domain Admin group by default. As a result, the ticket will be created with maximum privileges.

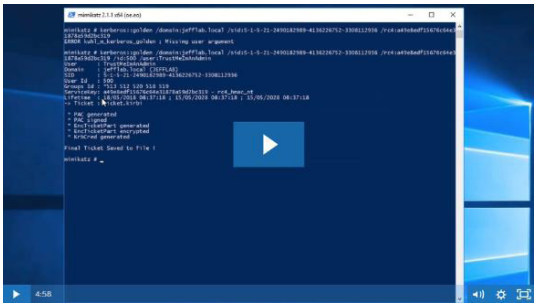
How the Golden Ticket Attack Works

- **Step 1** – Once an attacker has obtained privileged access to an Active Directory Domain Controller and can log on interactively or remotely, they can use Mimikatz to extract the KRBTGT account's password hash, in addition to the name and Security Identifier (SID) of the domain to which the KRBTGT account belongs.

- **Step 2** – Again using Mimikatz, the attacker generates a ticket (a "Golden Ticket") leveraging available commands and parameters such as:
  - the User account the ticket will be created for
  - the RID of the account being impersonated
  - the Groups to which the account in the ticket will belong
  - a SID to be injected into the SIDHistory attribute of the account in the ticket if cross-domain authentication is desired

- **Step 3** – Once the Golden Ticket has been generated, the attacker will perform a Pass-the-Ticket (PtT) attack by loading the ticket into the current session, providing them access to any resource connected to Active Directory.

Video Tutorial - https:// youtube.com/681319062



Solutions with Netwrix

Detection

Detection of Golden Ticket is possible by inspecting Kerberos ticket requests where the TGT lifespan values are above the allowed ranges.

Approach	Netwrix Product	Details
Golden Ticket Forged Lifetime Detection	Netwrix StealthDEFEND / Netwrix StealthINTERCEPT	Monitor for Kerberos tickets issued with values for the Maximum Lifetime for User Ticket and Maximum Lifetime for User Ticket Renewal values are above the values allowed in the domain policy. This will detect the majority of golden tickets, but if any users create golden tickets that are within the allowed lifespan those will not be detected. However, that largely defeats the purpose of the golden ticket to have nonexpiring administrative access to the domain.

Mitigation

Creating a golden ticket requires information such as the krbtgt account hash which is only accessible to privileged accounts. The best mitigations to golden tickets involve restricting administrative rights to Active Directory as much as possible.

Approach	Netwrix Product	Details
Reduce Domain Administrative Rights	Netwrix StealthAUDIT	Review membership of privileged domain groups (e.g. Domain Admins, Enterprise Admins, Server Operators) and remove unnecessary members. These groups provide rights to access domain controllers.
Secure Active Directory Permissions	Netwrix StealthAUDIT	Review the following Active Directory permission applied at the domain level: ▪ Replicating Directory Changes ▪ Replicating Directory Changes All These rights provide attackers the ability to obtain the krbtgt hash using the DCSync technique. Remove any unnecessary permissions.

Ntds.dit Password Extraction

Threat (Persistence)

By stealing the Ntds.dit file – Active Directory (AD)'s database – an attacker can extract a copy of every user's password hash and subsequently act as any user in the domain.

Once the hashes have been extracted or cracked, there is no limitation to what the attacker can do with them.

How Ntds.dit Password Extraction Works

Because the Ntds.dit file is constantly in use by AD, it cannot simply be copied and pasted to another drive as access will be denied. However, there are capabilities built into Windows, or with Powershell libraries, that provide ways around this.

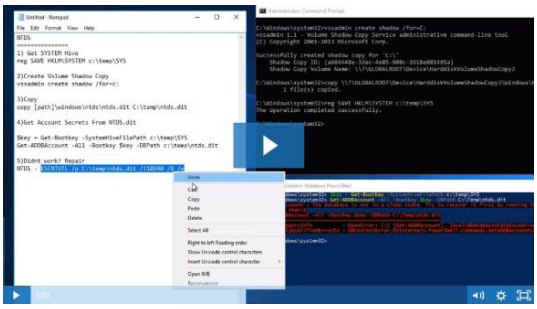
This is how the attack works when using VSSAdmin to steal the Ntds.dit file through **the Domain Controller's Volume Shadow Copy:**

- **Step 1** – An attacker obtains access to an Active Directory Domain Controller.
- **Step 2** – The attacker creates a Volume Shadow Copy from the system command prompt.
- **Step 3** – The attacker retrieves the Ntds.dit file from the Volume Shadow Copy.
- **Step 4** – The attacker copies the SYSTEM file from the Registry or Volume Shadow Copy, as it contains the Boot Key needed to decrypt the Ntds.dit file at a later time.

This is how the attack works when using tools like **PowerSploit** – a PowerShell penetration testing framework – giving the attacker the ability to copy a file from a raw NTFS-partitioned volume:

- **Step 1** – Offline, thus undetectable, the attacker extracts password hashes from the Ntds.dit file.
- **Step 2** – Once extracted, the attacker can now use tools like Mimikatz to perform Pass the Hash (PtH) attacks, or password cracking tools like Hashcat, to obtain their clear text values.

Video Tutorial - https:// youtube.com/681323313



Solutions with Netwrix

Detection

The best detection is to look for unexpected access events on the Ntds.dit file.

Approach	Netwrix Product	Details
Ntds.dit File Access	Netwrix StealthDEFEND / Netwrix StealthINTERCEPT	Monitor for access to the Ntds.dit file in the following ways: ▪ Direct access to the file on the file system. This file is locked by Active Directory while in use so typically an attacker cannot obtain the file without stopping the Active Directory service. Monitoring for access events as well as access denied events by user accounts can provide meaningful insight

Approach	Netwrix Product	Details
Ntds.dit File Access	Netwrix StealthDEFEND / Netwrix StealthINTERCEPT	into unwanted access at-tempts, because the AD service runs as Local System. ▪ Access to the Ntds.dit file through Volume Shadow Copies. While the file is locked attackers are able to create a shadow copy of the entire drive and extract the Ntds.dit file from the shadow copy.

Prevention

In order to prevent malicious access to the Ntds.dit file you can implement blocking rules.

Approach	Netwrix Product	Details
NTDS.dit File Access Blocking	Netwrix StealthINTERCEPT	Block access to the Ntds.dit file through Volume Shadow Copies and direct access on the file system. This will ensure even if an attacker stops Active Directory to unlock the file and has full admin rights, they will not be able to gain access to it directly.

Mitigation

The best way to protect against attacks leveraging the Ntds.dit file is to tightly control the administrative groups that provide access to your domain controllers.

Approach	Netwrix Product	Details
DC Logon Groups	Netwrix StealthAUDIT	Perform reviews of all domain groups which provide logon rights to domain controllers (e.g. Domain Admins, Server Operators) as the members of these groups can gain access to the Ntds.dit file which resides on the file system of the domain controller. Perform regular reviews and remove unnecessary members.

DCSync

Threat (Persistence)

DCSync is a command within a Mimikatz that an attacker can leverage to simulate the behavior of Domain Controller (DC). More simply, it allows the attacker to pretend to be a DC and ask other DC's for user password data.

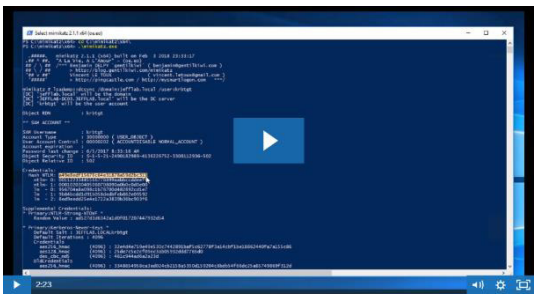
DCSync attacks are difficult to prevent. The DCSync attack asks other domain controllers to replicate information using the Directory Replication Service Remote Protocol (MS-DRSR). Because MS-DRSR is a valid and neces-sary function of Active Directory (AD), it cannot be turned off or disabled.

Additionally, while Domain Replication capabilities are controlled by the Replicating Changes permissions set on the domain and are limited to the Domain Admins, Enterprise Admins, Administrators, and DC groups by default, it is possible for any account or group to be granted these rights.

How the DCSync Attack Works

- **Step 1** – An attacker compromises an account with the rights to perform domain replication (e.g. Domain Admins, Enterprise Admins, Administrators, and Domain Controllers groups by default).
- **Step 2** – Once the proper privileges are obtained, the attacker leverages the Mimikatz DCSync command to retrieve account password hashes from AD.
- **Step 3** – Once obtained, the attacker can create forged Kerberos tickets to access any resource connected to AD.

Video Tutorial - https:// youtube.com/681318975



Solutions with Netwrix

Detection

Detection of DCSync is possible by looking for replication requests against domain controllers that are not originating from other domain controllers.

Approach	Netwrix Product	Details
Domain Controller Impersonation	Netwrix StealthDEFEND / Netwrix StealthINTERCEPT	Monitor for Active Directory replication traffic coming from a machine that is not a domain controller.

Prevention

Prevention of DCSync is possible by blocking replication requests against domain controllers that are not originating from other domain controllers.

Approach	Netwrix Product	Details
Block Domain Controller Impersonation	Netwrix StealthDEFEND / Netwrix StealthINTERCEPT	Block Active Directory replication traffic coming from a machine that is not a domain controller.
Restrict Domain Permissions Changes	Netwrix StealthDEFEND / Netwrix StealthINTERCEPT	Monitor and optionally block the ability to change permissions to the Domain. By restricting users adding permissions for replication, it will reduce the ability to create persistence where non-administrator accounts can perform the DCSync attack.

Mitigation

To mitigate the DCSync attack it is necessary to restrict domain replication permissions. By default, Domain Admins and other privileged users will have these rights but they can access account information several other ways. It is important to limit other users from having these sensitive permissions.

Approach	Netwrix Product	Details
Secure Active Directory Permissions	Netwrix StealthAUDIT	Review the following Active Directory permission applied at the domain level: ▪ Replicating Directory Changes ▪ Replicating Directory Changes All These rights provide attackers the ability to obtain the password hashes using the DCSync technique. Regularly review and remove unnecessary permissions.

Kerberoasting

Threat (Persistence)

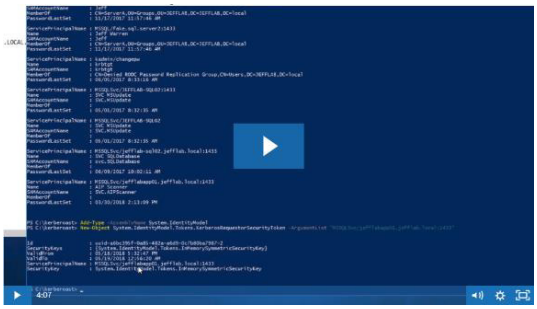
Kerberoasting is an attack method that allows an attacker to crack the passwords of service accounts in Active Directory (AD) offline and without fear of detection.

Kerberoasting is difficult to detect. Cracking service accounts is a particularly successful approach because their passwords very rarely change. Additionally, cracking tickets offline will not cause any domain traffic or account lockouts, so it is undetectable.

How Kerberoasting Works

- **Step 1** – An attacker scans AD for user accounts with Service Principal Name (SPN) values set using any number of methods, including PowerShell and LDAP queries, scripts provided by the Kerberoast toolkit, or tools like PowerSploit.
- **Step 2** – Once a list of target accounts is obtained, the attacker requests service tickets from AD using SPN values.
- **Step 3** – Using Mimikatz, the attacker then extracts the service tickets to memory and saves the information to a file.
- **Step 4** – Once the tickets are saved to disk, the attacker passes them into a password cracking script that will run a dictionary of passwords as NTLM hashes against the service tickets they have extracted until it can successfully open the ticket. When the ticket is finally opened, it will be presented to the attacker in clear text.

Video Tutorial - https:// youtube.com/681320535



Solutions with Netwrix

Detection

Detection of Kerberoasting is possible by looking for Kerberos ticket requests with weak encryption for accounts with SPN values.

Approach	Netwrix Product	Details
Service Ticket Request with Weak Encryption	Netwrix StealthDEFEND / Netwrix StealthINTERCEPT	Monitor for Kerberos ticket requests using weak encryption (RC4_HMAC_MD5). These tickets are obtained when requesting Kerberos tickets for a particular service principal name (SPN), and are returned encrypted with the password of the service account tied to that SPN.
Adding SPN Values	Netwrix StealthDEFEND / Netwrix StealthINTERCEPT	Monitor for addition of new SPN values to accounts. These can be added maliciously by attackers so they can later Kerberoast the account.
Service Account Recon	Netwrix StealthDEFEND / Netwrix StealthINTERCEPT	Monitor for LDAP activity that is explicitly performing reconnaissance on service accounts (accounts with service principal names)

Mitigation

Mitigation of Kerberoasting is possible by ensuring a proper inventory is taken of all accounts with SPN values and enforcing best practices for password security.

Approach	Netwrix Product	Details
Enforce Strong Passwords	Netwrix Enterprise Password Enforce	The best way to mitigate Kerberoasting is to enforce long, complex and regularly changing passwords for service accounts. Also, reduce sharing of passwords across accounts and using easily guessed passwords that may appear in hacker dictionaries.
Service Account Inventory	Netwrix StealthAUDIT	Inventory all service accounts in Active Directory with SPN values registered. Review and remove/disable any unnecessary accounts. Identify any accounts with old passwords and force password updates.

Password Spraying

Threat (Persistence)

Password Spraying is a technique that attackers leverage to guess the password of an account. By trying a small number of highly common passwords against large numbers of accounts while also staying below an organization's defined lockout threshold, the attacker can compromise accounts without any elevated privileges and likely without detection.

How Password Spraying Works

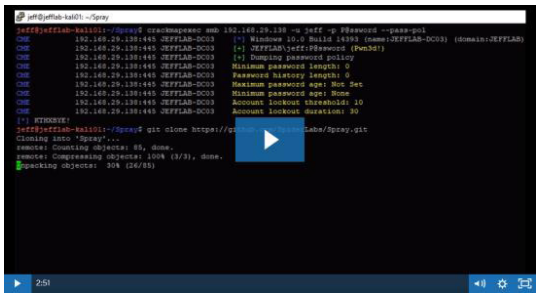
- **Step 1** – An attacker compromises a standard domain user (e.g. phishing, social engineering, etc.).
- **Step 2** – Using tools like CrackMapExec (CME), the attacker enumerates Active Directory (AD)'s password and lockout policies
- **Step 3** – Using an LDAP query, the attacker compiles a list of users to attack
- **Step 4** – Again using CME, the attacker runs commands against the Domain Controller and cycles through the list of passwords for every user account until a hit is found

NOTES: The lockout policy information collected by CME is used to determine how many bad passwords the attacker can guess per account to avoid a lockout.

The password policy information collected by CME is used to help the attacker craft a custom dictionary of potential passwords to guess against all accounts based on what's actually possible in the target Active Directory environment.

There are existing password lists from real-world and well-known data breaches that can be obtained from sites like GitHub, making this even easier for the attacker.

Video Tutorial - https://youtube.com/681325017



Solutions with Netwrix

Detection

Detection of password spraying is possible by looking for patterns that indicate password guessing is taking place across numerous accounts.

Approach	Netwrix Product	Details
Bad User ID Attacks	Netwrix StealthDEFEND / Netwrix StealthINTERCEPT	Monitor for attempts to authenticate using non-existent user accounts. Many times password spraying tools will attempt to guess account names rather than attacking a list of known accounts.
Adding SPN Values	Netwrix StealthDEFEND / Netwrix StealthINTERCEPT	Monitor for addition of new SPN values to accounts. These can be added maliciously by attackers so they can later Kerberoast the account.

Mitigation

Mitigation of password spraying is possible by enforcing strong password standards and reducing password sharing across accounts.

Approach	Netwrix Product	Details
Enforce Strong Passwords	Netwrix Enterprise Password Enforce	The best way to mitigate password spraying is to enforce long, complex and regularly changing passwords. Also, ensure accounts cannot use easily guessed passwords that may be found in hacker dictionaries.
Reduce Password Sharing	Netwrix StealthAUDIT	Identify when multiple user accounts are sharing the same password and force them to change their password.

Plaintext Password Extraction – Group Policy Preferences

Threat (Privilege Escalation)

Group Policy Preferences allow administrators to create and manage local accounts on servers and workstations in an Active Directory (AD) domain. Attackers can easily find and obtain the encrypted passwords of administrative account credentials managed by Group Policy Preferences and decrypt them using the Microsoft-published AES key.

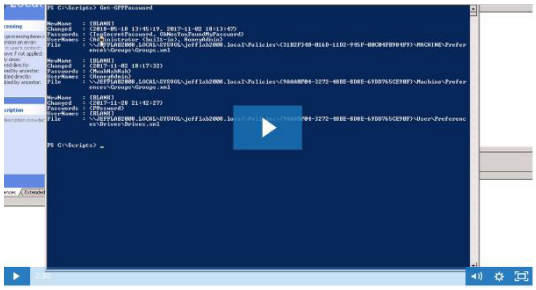
How Plaintext Password Extraction through Group Policy Preferences Works

An attacker locates group policy XML files containing AES encrypted local account passwords on a Domain Controller's SYSVOL share, leveraging PowerShell or other tools like PowerSploit's Get-GPPPassword command.

In conjunction with the Microsoft-published AES key, the attacker decrypts the passwords, exposing Administrative account passwords in clear text.

NOTES: Because the SYSVOL share is open to Authenticated Users, anybody within the organization can read the files stored here. Therefore, any user account can find and decrypt these files and gain access to plain text pass-words for Administrator accounts.

Video Tutorial - https:// youtube.com/681327780



Solutions with Netwrix

Mitigation

The best protection from Group Policy Preference abuse is to remove any passwords from GPPs.

Approach	Netwrix Product	Details
Identify & Remove GPP Passwords	Netwrix StealthAUDIT	Report on any group policy preferences which leverage cPassword fields that contain password data which can be decrypted. Remove the dependency on these and migrate to a more secure way to accomplish the task of that GPP setting.

LDAP Reconnaissance

Threat (Discovery)

When an attacker initially compromises a system on a network, they will have little to no privileges within the domain. However, once an attacker has infiltrated any domain-joined computer, they are able to query Active Directory (AD) and its objects using Lightweight Directory Access Protocol (LDAP), allowing them to locate sensitive accounts and assets to target in their attack.

LDAP Reconnaissance is difficult to detect. Due the architecture of AD, searching AD for privileged information rarely requires privileged access rights.

How LDAP Reconnaissance Works

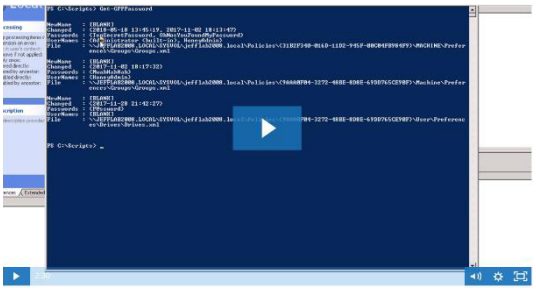
An attacker obtains access to any domain-joined system (e.g. via phishing, social engineering, etc.)

Using PowerShell, the attacker crafts and executes queries against AD objects, searching for various conditions including:

- User objects containing Service Principal Names (SPNs), indicating the accounts are used to run services to support applications like Microsoft SQL Server and SharePoint
- The membership of Sensitive Security Groups like Domain, Enterprise, and Schema Admins, listing the user accounts containing the highest level of privilege in the domain
- The location of high-profile assets, such as file servers, SQL databases, and AD Domain Controllers

NOTE: While SPNs make it easy to locate Service Accounts (which are prime targets because they often contain privileged access rights and have loose password expiration restrictions), there are other variables an attacker can query to identify accounts that are likely Service Accounts such as the “Password Expiration” setting on each user’s account configured via User Account Control.

Video Tutorial - https:// youtube.com/681325918



Solutions with Netwrix

Detection

Detection of LDAP reconnaissance is possible by looking for abnormal LDAP query activity against Active Directory.

Approach	Netwrix Product	Details
Admin Account Recon	Netwrix StealthDEFEND / Netwrix StealthINTERCEPT	Monitor for LDAP activity that is explicitly performing reconnaissance on administrative groups and users within AD.
Service Account Recon	Netwrix StealthDEFEND / Netwrix StealthINTERCEPT	Monitor for LDAP activity that is explicitly performing reconnaissance on service accounts (accounts with service principal names)
BloodHound Detection	Netwrix StealthDEFEND / Netwrix StealthINTERCEPT	Monitor for LDAP activity that is used by the attack path mapping tool BloodHound to show attackers how to move laterally across the network towards higher value targets.

Mitigation

LDAP reconnaissance is impossible to stop entirely, due to the design of Active Directory. However, it is important to make sure secure data is protected and safe from LDAP queries.

Approach	Netwrix Product	Details
Sensitive Object & Attribute Permissions	Netwrix StealthAUDIT	Ensure objects and attributes that should be protected (e.g. the ms-Mcs-AdmPwd attribute) are secured and can-not be exported through LDAP.

About Netwrix

Netwrix® makes data security easy by simplifying how professionals can control sensitive, regulated and business-critical data, regardless of where it resides. More than 11,500 organizations worldwide rely on Netwrix solutions to secure sensitive data, realize the full business value of enterprise content, pass compliance audits with less effort and expense, and increase the productivity of IT teams and knowledge workers.

Founded in 2006, Netwrix has earned more than 150 industry awards and been named to both the Inc. 5000 and Deloitte Technology Fast 500 lists of the fastest growing companies in the U.S.

For more information, visit www.netwrix.com

Next Steps

See Netwrix products — Check out the full portfolio of Netwrix products: netwrix.com/products

Get a live demo — Take a personalized product tour with a Netwrix expert: netwrix.com/livedemo

Request a quote — Receive pricing information: netwrix.com/buy

CORPORATE HEADQUARTER:

300 Spectrum Center Drive
Suite 200 Irvine, CA 92618

565 Metro Place S, Suite 400
Dublin, OH 43017

5 New Street Square
London EC4A 3TW

PHONES:

1-949-407-5125
Toll-free (USA): 888-638-9749

1-201-490-8840

+44 (0) 203 588 3023

OTHER LOCATIONS:

Spain:	+34 911 982608
Netherlands:	+31 858 887 804
Sweden:	+46 8 525 03487
Switzerland:	+41 43 508 3472
France:	+33 9 75 18 11 19
Germany:	+49 711 899 89 187
Hong Kong:	+852 5808 1306
Italy:	+39 02 947 53539

SOCIAL:

netwrix.com/social