Brought to you by:

ManageEngine

Defending Against Active Directory Attacks

for dummies®
A Wiley Brand

Employ the best defense techniques

Know five of the most common AD attacks

Defend against AD's vulnerabilities

ManageEngine Special Edition

Ram Vaidyanathan, B.Eng.Mgmt, MBA

About ManageEngine

As the IT management division of Zoho Corporation, ManageEngine prioritizes flexible solutions that work for all businesses, regardless of size or budget. ManageEngine has 90+ products and free tools that comprehensively cover your IT needs, at prices you can afford. More than 180,000 companies worldwide rely on ManageEngine products to ensure the optimal performance of their critical IT infrastructure, including networks, servers, applications, desktops, Active Directory, and more. With over 4,000 employees working round the clock to make product requests a reality, ManageEngine focuses on simplifying IT for everyone.

Defending Against Active Directory Attacks

ManageEngine Special Edition

by Ram Vaidyanathan, B.Eng.Mgmt, MBA

for dummies®
A Wiley Brand

Defending Against Active Directory Attacks For Dummies®, ManageEngine Special Edition

Published by
John Wiley & Sons, Inc.
111 River St.
Hoboken, NJ 07030-5774
www.wiley.com

ISBN 978-1-394-20795-4 (pbk); ISBN 978-1-394-20796-1 (ebk)

Publisher's Acknowledgments

Development Editor: Rachael Chilvers

Project Editor: Saikarthick Kumarasamy

Acquisitions Editor: Traci Martin

Editorial Manager: Rev Mengle

Business Development Representative: Matt Cox

Table of Contents

Introduction

Active Directory (AD) is used by millions of organizations around the world to authenticate and authorize users within their network. You can think of it as a database that holds all user and computer information, including their different attributes and their passwords. It is, then, no surprise that cyber attackers target AD to get a foothold into a network and ultimately exfiltrate data.

If you're a security analyst who works within the security operations center (SOC) of your organization, you need to play a major part in keeping attacks at bay. You need to use a security information and events management (SIEM) solution to constantly look for and detect threats, investigate the probable root cause, and respond. Since AD is a low-hanging fruit for attackers, you need to especially focus on it. You need to know about AD's inherent vulnerabilities and the popular ways in which attackers leverage them. You also need to know how you can detect and respond to these attacks with a SIEM solution.

About This Book

This book explains five popular attacks that plague Active Directory. You'll see how these attacks are done, where the gaps lie, and what you can do as an effective defender of your fort. At the end of the day, to be an effective defender, you also need to know what motivates attackers. In a way, you have to think like an attacker so that you can pre-empt their moves.

Please note that this book is not exhaustive. After all, I can't cover all the attacks AD is exposed to in just a few pages. My sincere hope is that you'll read about the five popular attacks, get more interested in the topic, and go on to learn more. In the field of cyber security, when you learn about one attack, your knowledge builds, and learning about a second attack should become considerably simpler. By the time you finish reading the book, I hope that you'll feel adept at defending against AD attacks.

Foolish Assumptions

While writing this book, I made some assumptions about who will read it. You may find yourself in one of these general profiles:

- You're a security analyst who's responsible for protecting your company against cyber attacks. You use a SIEM or a security analytics solution to detect, investigate, and respond to threats.
- You're an Active Directory administrator who's tasked with creating user accounts, computer accounts, security groups, password policies, Group Policy objects, and so on. You want to know about how to secure AD against popular attacks.
- You're a security manager or a SOC manager who wants to understand how the most popular AD attacks work. You may want to use this knowledge to get hands-on yourself or to train your team.
- You're someone who's interested in cyber security in general, and Active Directory security in particular. Perhaps you want to evaluate if a career in cyber security is a right fit for you.

Icons Used in This Book

This book uses icons in the margin to draw your attention to certain kinds of information. Here's a guide to the icons:


TIP

The Tip icon highlights anything that'll save you time or money or just make your life a little easier.


REMEMBER

When I tell you something so important that you should commit it to memory, I mark it with the Remember icon.


TECHNICAL STUFF

Sometimes I get into the weeds, providing some information that's a bit more technical in nature. When I do, I mark it with the Technical Stuff icon.

Beyond the Book

This book is packed with useful information about AD attacks, but if you want resources beyond what this book offers, I have some insight for you:

- Expert Talks is a cyber security blog by ManageEngine. In it, you can find numerous articles about the latest attacks and techniques of defense. You can access it here: `https://www.manageengine.com/log-management/cyber-security/expert-talk.html`.
- Would you like to know how much money a SIEM or security analytics solution can save for your company? Go here to calculate: `https://www.manageengine.com/log-management/cyber-security/calculating-the-cost-savings-of-a-siem-solution.html`.
- Ask for a personalized demo of ManageEngine Log360, a comprehensive SIEM solution. With Log360, you can protect against AD attacks, DDoS, ransomware, and more. Visit: `https://www.manageengine.com/log-management/ad-custom-demo-request-page.html`.

IN THIS CHAPTER

- Learning how Active Directory is vulnerable to attack
- Identifying the top five Active Directory attacks

Chapter 1
Understanding Active Directory's Vulnerabilities

Every day, on networks all over the world, billions of users are signing in and doing their daily duties, using appropriate access permissions. And the system that makes that process work — at least on Microsoft networks — is Active Directory.

Active Directory (AD) is a database that stores details about users, their passwords, and their access privileges. Whenever a user supplies their credentials to log on to a domain-joined machine, AD verifies their identity, authenticates them, and grants their assigned permissions to access network resources.

It's no surprise, then, that cyber attackers continuously target AD systems to gain illegitimate entry into networks and unauthorized access to resources. At the end of a successful AD attack, they can exfiltrate, encrypt, or even destroy sensitive data. Attacks on AD are popular because it's *especially* vulnerable.

In this chapter, you'll discover some of the vulnerabilities in AD and five important AD exploits that security analysts must be wary about.

Why is Active Directory so Vulnerable?

REMEMBER

AD is vulnerable to attacks for five major reasons:

- **Complex configurations.** AD's flexibility enables administrators to set policies and permissions in numerous ways. That's good, but it also introduces complexity. Unless administrators take utmost care, they can easily miss something — and that "something" can give an attacker an entry point into the network. From here, they can burrow deeper and get access to sensitive data.
- **Availability of AD-exploitation tools.** Attackers have a variety of free tools at their disposal, such as Mimikatz and BloodHound. They can also enumerate an AD domain using PowerShell, a Microsoft command line interface shell.
- **Auditing challenges.** To log security-related events, administrators first have to configure the right audit policies and advanced audit policies. Then they must use the not-so-friendly Event Viewer interface to find out what happened. If they're unable to identify incidents quickly, an attacker can burrow deeper into the network.
- **The propensity to bloat.** Over time, AD has the propensity to bloat. There can be inactive user accounts, privilege creep, and complex nested permissions. Each of these is a vulnerability that an attacker can exploit.
- **The popularity of AD.** More than 90 percent of organizations around the world use AD. Attackers know that targeting AD gives them many potential victims to go after.

Out of all these reasons, the first is the most critical. Threat actors mainly exploit the fact that AD environments can be complex to configure. Here are some common misconfigurations that they can leverage to launch or propagate an attack:

- Weak passwords on user accounts and service accounts, usually because the administrator has not configured strong password policies.

- A user account is configured as a local administrator in multiple machines.
- Server Message Block (SMB) signing isn't enabled on user workstations.
- A service account has domain administrator privileges.

Identifying Five Common AD Attacks

This isn't a comprehensive list, but these misconfigurations are the root cause of five prevalent AD attacks that you may face:

- LLMNR/NBT-NS poisoning
- SMB relay
- Kerberoasting
- Domain enumeration
- Password spray

In the rest of this book, you'll learn how each of these attacks work, and what you can do to defend against them.

IN THIS CHAPTER

- Introducing the LLMNR and NBT-NS protocols
- Learning how attacking LLMNR/NBT-NS protocols compromises accounts
- Understanding how an adversary might carry out an attack
- Identifying ways to detect and mitigate attacks

Chapter 2
Understanding an LLMNR/NBT-NS Poisoning Attack

The *Link Local Multicast Name Resolution (LLMNR)* and *NetBIOS Name Service (NBT-NS)* protocols enable name resolution for local network hosts. Normally a Domain Name System (DNS) performs this job, but LLMNR and NBT-NS will kick in when DNS fails. They serve as a failover so a network doesn't face any downtime if DNS fails.

These are useful services to have at the ready, but they can also pose a security risk. This chapter explains how attackers target LLMNR and NBT-NS and what you can do about it.

What is an LLMNR /NBT-NS Poisoning Attack?

To understand the LLMNR/NBT-NS poisoning attack, you first need to know what a password hash is. Think of it as a representation of a password by another more complex string. A password hash is obtained when a password is put through a hashing algorithm. The password hash of a password will be different depending upon the algorithm used.

In an LLMNR/NBT-NS poisoning attack, an attacker intercepts a user's password hash and cracks it offline. Figure 2-1 shows how this attack plays out.

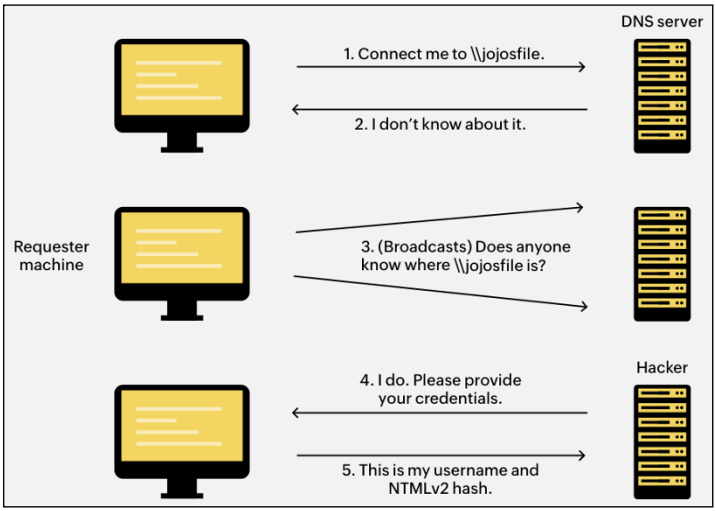


FIGURE 2-1: An overview of an LLMNR poisoning attack.

TECHNICAL STUFF

Here are the steps of this attack:

1. A user in your network wants to access the file server \\jojofiles but requests \\jojosfile (note the typo) by mistake.
2. As expected, the DNS server cannot recognize the server name and is unable to resolve it to its IP address.
3. The LLMNR/NBT-NS protocol kicks in. The user sends a broadcast message to all the devices on the network to check if any of them can direct them to \\jojosfile.

4. An attacker intercepts this broadcast message and responds to the user, "I know the location of \\jojosfile. Send me your hash, and I will point you to it."
5. The user believes the attacker and provides their NTLMv2 hash. The attacker receives the hash by using a tool such as Responder.
6. The attacker receives the hash and cracks it using tools such as Hashcat.

Figure 2-2 shows how an attacker can view a user's hash using Responder. I obtained this screenshot by running Responder on Kali Linux. In this example, the attacker has obtained a user's NTLMv2 hash.

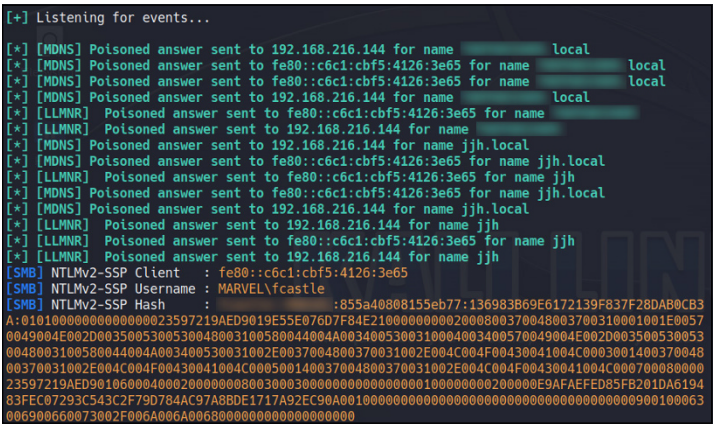


FIGURE 2-2: Obtaining a password hash using Responder.

Figure 2-3 shows the cracked password after running Hashcat. As you can see, the user's password is password@123, which is a very weak password. It took only a few seconds to crack using Hashcat.

FIGURE 2-3: Cracking a password hash with Hashcat.

Defending Against LLMNR/ NBT-NS Poisoning Attacks

TIP

Here are some ways to defend against LLMNR and NBT-NS poisoning attacks:

- Ask the AD administrator to disable LLMNR and NBT-NS.
- If the AD administrator can't disable LLMNR and NBT-NS due to organizational policies, ask them to work around it by ensuring that Network Access Control is enabled.
- Use an effective identity and access management solution that requires strong password policies and multi-factor authentication.
- Use an effective SIEM solution like ManageEngine Log360 to detect unusual user logons.

IN THIS CHAPTER

- Discovering the importance of the SMB protocol
- Understanding how attackers use the SMB protocol
- Learning how to defend against this attack

Chapter 3
Defending Against SMB Relay Attacks

This chapter explains the service that the *Server Message Block (SMB)* protocol provides on a network, how attackers target systems via SMB, and what you can do about it.

Understanding SMB's Purpose

When users want to share files and other resources within a network, they use the SMB protocol. Their computer first sends an SMB request to another computer or server, which checks their access permissions. If permission is granted, the other computer or server sends an SMB response, establishing a two-way connection. Once this connection is established, the user can start sharing files. The shared resource is called an SMB Share. Figure 3-1 shows how an SMB connection is usually established.

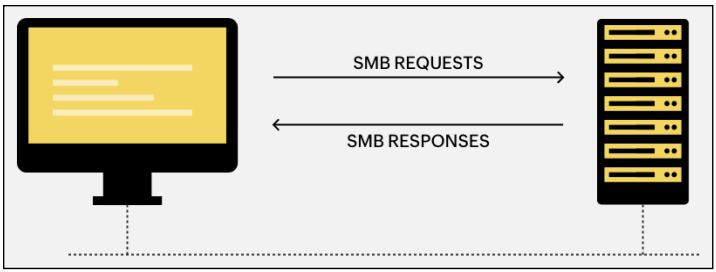


FIGURE 3-1: SMB request and response in normal conditions.

This situation presents a unique opportunity for attackers to listen in during a connection attempt, intercept the requester's New Technology LAN Manager (NTLM) password hash, and relay it to the server to get privileged access. This is because, by default, SMB signing isn't required in client machines, and many organizations don't change this default setting in AD. On top of that, a user may have local administrator privileges on another machine, apart from their own. And these are exactly the loopholes attackers want to exploit.

Breaking Down an SMB Relay Attack

Figure 3-2 shows what happens during an SMB relay attack.

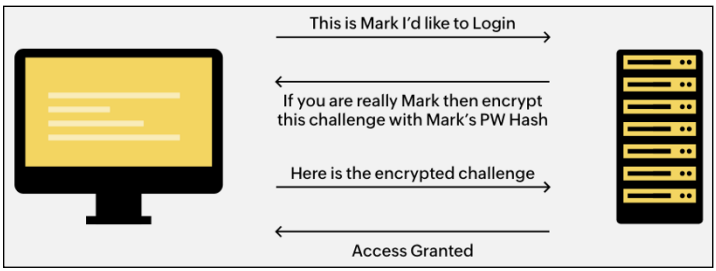


FIGURE 3-2: Overview of a SMB relay attack.

Here are the steps an attacker would likely take to perform a SMB relay attack:

1. They perform domain enumeration to find servers and computers likely to hold sensitive data in file shares.

(Chapter 5 explains domain enumeration in greater detail.) Let's assume that the attacker finds a user's machine to be a particularly attractive target, after performing a domain enumeration. For our discussion, let's also assume that the IP address of this machine is 192.168.216.142.

2. The attacker checks to see if a SMB relay attack will work against the target's machine. To do that, they run a Network Mapper (Nmap) scan on the network (as in Figure 3-3) to ensure that SMB signing is enabled but not required on this machine. This is a default setting for all domain-joined computers in AD.

```
Nmap scan report for 192.168.216.142
Host is up (0.00058s latency).

PORT    STATE SERVICE
445/tcp open  microsoft-ds
MAC Address: 00:0C:29:E0:F7:C3 (VMware)

Host script results:
| smb2-security-mode:
|   311:
|_    Message signing enabled but not required

Nmap scan report for 192.168.216.144
Host is up (0.00045s latency).

PORT    STATE SERVICE
445/tcp open  microsoft-ds
MAC Address: 00:0C:29:16:70:A7 (VMware)

Host script results:
| smb2-security-mode:
|   311:
|_    Message signing enabled but not required
```

FIGURE 3-3: Result of an Nmap scan that shows SMB signing isn't needed on workstations.

You can block Nmap scanning by setting effective firewall rules, but attackers can use several techniques to bypass those rules. All they need is one overlooked point of vulnerability. And blocking Nmap isn't a sure-fire protection method, because it's far from the only port scanners available.

3. The attacker learns from the Nmap scan that the organization hasn't made SMB signing mandatory on any of the other workstations in the network. For example, Figure 3-3 shows that SMB signing isn't mandatory in the machine with the IP address 192.168.216.144 (another user's machine).

4. The attacker makes a mental note that there's a good opportunity for them in case there's a user who's a local administrator on both their own machine and the target's machine. The attacker waits patiently.
5. When a user who has local administrator privileges on both their own machine and the target's machine attempts to reach a file share, the attacker intercepts their NTLM password hash with a tool such as Responder.
6. The attacker relays this hash to the target's machine without even cracking it, using a tool like ntlrelay.py. Then they gain the access they wanted! (This step is exactly what's shown in Figure 3-2.)
7. The attacker opens an interactive SMB client shell on the target's machine via TCP on 127.0.0.1 via the Port 11000, as shown in Figure 3-4.
8. They do all sorts of malicious activity via this shell on the target's machine!

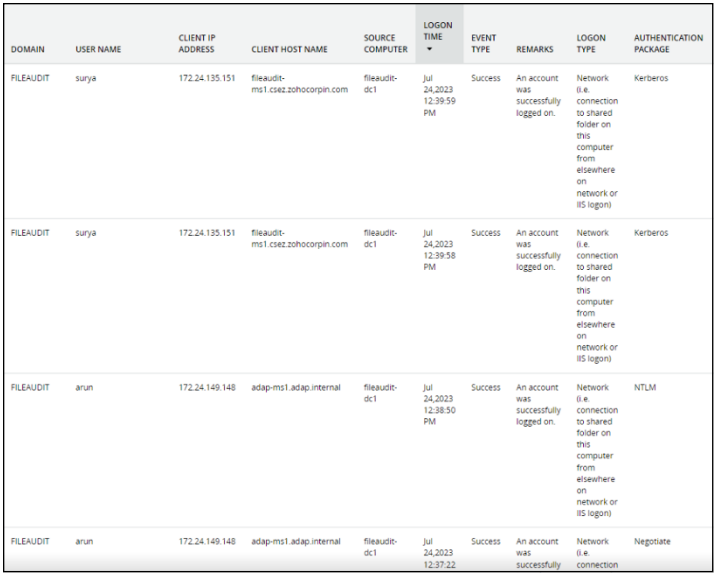
FIGURE 3-4: Results of a NTLM relay that shows a successful connection.

Defending Against SMB Relay Attacks

TIP

The SMB relay attack is extremely dangerous, and you must leave no stone unturned to protect against it. Here are some ways to do this:

- **Enable SMB signing for all workstations.** By default, SMB signing is only required for domain controllers, and isn't required for workstations. Your AD administrator must enable this setting in AD for your workstations.
- **Use Kerberos authentication instead of NTLM.** The SMB relay attack is dependent on users authenticating with the NTLM protocol. Your AD administrator must ensure that your users authenticate with Kerberos instead. As a proactive security analyst, you can check which users authenticate with which protocol using a solution like Log360. Figure 3-5 shows a sample report from Log360 that gives you this information. Look for the relevant details in the column *Authentication Package*.



DOMAIN	USER NAME	CLIENT IP ADDRESS	CLIENT HOST NAME	SOURCE COMPUTER	LOGON TIME	EVENT TYPE	REMARKS	LOGON TYPE	AUTHENTICATION PACKAGE
FILEAUDIT	surya	172.24.135.151	fileaudit-ms1.csez.zohocorpin.com	fileaudit-dc1	Jul 24,2023 12:39:59 PM	Success	An account was successfully logged on.	Network (i.e. connection to shared folder on this computer from elsewhere on network or IIS logon)	Kerberos
FILEAUDIT	surya	172.24.135.151	fileaudit-ms1.csez.zohocorpin.com	fileaudit-dc1	Jul 24,2023 12:39:58 PM	Success	An account was successfully logged on.	Network (i.e. connection to shared folder on this computer from elsewhere on network or IIS logon)	Kerberos
FILEAUDIT	arun	172.24.149.148	adap-ms1.adap.internal	fileaudit-dc1	Jul 24,2023 12:38:50 PM	Success	An account was successfully logged on.	Network (i.e. connection to shared folder on this computer from elsewhere on network or IIS logon)	NTLM
FILEAUDIT	arun	172.24.149.148	adap-ms1.adap.internal	fileaudit-dc1	Jul 24,2023 12:37:22	Success	An account was successfully	Network (i.e. connection	Negotiate

FIGURE 3-5: A security information and event management (SIEM) solution like Log360 can show you the authentication protocols that are used.

Kerberos is more secure than NTLM or NTLMv2, but there are ways to break Kerberos as well. (I cover the vulnerabilities of Kerberos authentication in Chapter 4.)

- **Detect anomalous user and machine associations.** If a user performs an activity on a machine that they've never been associated with before, you can track and get alerted about it. Again, a SIEM solution such as ManageEngine Log360 can help with this.
- **Perform file integrity and file activity monitoring.** File integrity monitoring detects changes that happen in your volume and system files and directories. It helps detect malicious file activity such as file modifications, deletions, renames, moves, downloads, and so on.
- **Look for users added to local administrators.** If a new user is added as a local administrator on a machine assigned to them, administrators need to know about it right away. This could be a case of privilege escalation.

IN THIS CHAPTER

- Learning about the Kerberos authentication protocol
- Identifying Kerberos' weaknesses
- Understanding golden ticket and silver ticket attacks
- Tracing the steps of a Kerberoasting attack
- Defending against Kerberoasting

Chapter 4
Protecting Against a Kerberoasting Attack

The *Kerberos authentication protocol* is named after Cerberus, the three-headed dog that guards the gates of the underworld according to ancient Greek legend. Kerberos works because of an interplay between three elements: the user or client, the key distribution center, and the service.

In Chapter 4, I explained that Kerberos is a big improvement over its predecessors, and most organizations use it. However, it has its own security weaknesses, like any protocol does, and it can be exploited by attackers.

In this chapter you'll learn some basic facts about Kerberos and how a common attack against it — *Kerberoasting* — works. In a Kerberoasting attack, an attacker leverages an inherent weakness present in the Kerberos authentication protocol. You'll also find out how to defend against this type of attack.

Identifying the Three Key Elements of Kerberos

As mentioned, Kerberos has three key components: the user, the key distribution center, and the service to be accessed. Figure 4-1 summarizes these elements.

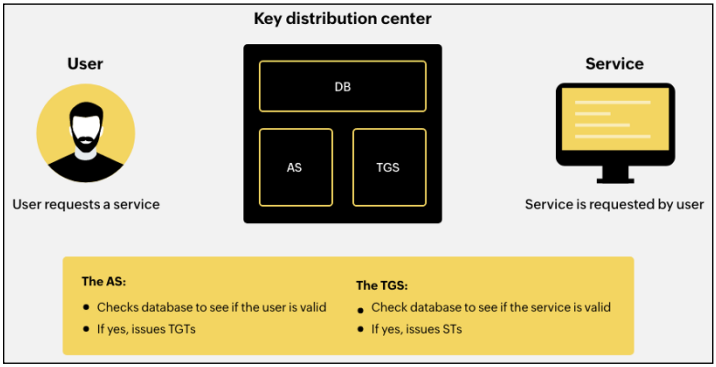


FIGURE 4-1: The three elements of Kerberos authentication.

REMEMBER

The *user* is someone who wants access to a resource or a service. They could be any legitimate employee within your organization who has a user account in AD.

The *key distribution center (KDC)*, as the name suggests, is a server that distributes keys. The KDC is composed of two other servers: The Authentication Server (AS) and Ticket Granting Server (TGS).

The Authentication Server (AS) performs two jobs:

- Checks if a user requesting a service is legitimate.
- Issues ticket granting tickets (TGTs), which the user can use to communicate with the Ticket Granting Server (TGS).

The TGS also performs two jobs:

- Checks if your requested service is legitimate.
- Issues service tickets (STs) that the user can use to communicate with a service.

In addition to the AS and the TGS, the KDC also has a database that contains all user names and the corresponding client secret keys, and all service names and the corresponding service secret keys. These keys are a function of the user's and service's password, respectively.

The *service* is the resource that a user wants to access. It could be an application, file server, SQL database, or other resource.

Deciphering Kerberos Authentication

Here are the six steps of Kerberos authentication:

1. The user sends a message to the AS, in plain text, containing the username and the service name.

   The AS receives this message and checks its database to see if the user is valid. If yes, it gets the corresponding client secret key and then creates two messages:

   - Message #1 contains the TGS name and a TGS Session Key. The client's secret key encrypts that.
   - Message #2, the TGT, contains the username, the TGS name, and the TGS Session Key. The TGS Secret Key encrypts that. This TGS Secret Key is a hash of the Kerberos ticket granting ticket (KRBTGT) account that exists in every Active Directory.

2. The AS sends the two messages to the user. The user decrypts Message #1 easily since it's encrypted by its own secret key. It uses the data in Message #1 to obtain the TGS name and the TGS Session Key. It sets the encrypted TGT (Message #2) aside for a moment and creates two new messages:

   - Message #3 contains the requested service's name.
   - Message #4 is a user authenticator message that contains the username encrypted by the TGS Session Key.

3. The user sends the encrypted TGT (Message #2), Message #3, and Message #4 to the TGS.

   The TGS looks at Message #3, gets the service name, and checks if it exists in its database. If yes, it grabs the corresponding service secret key.

The TGS also decrypts the encrypted TGT (Message #2) because it has access to its own secret key. By doing this, it obtains the username, the TGS name, and the TGS Session Key. With the TGS Session Key, it then decrypts the user authenticator (Message #4). It then checks if the username in the user authenticator is the same as the username in the TGT.

After doing all this work, the TGS creates two new messages:

- Message #5 contains the service name and a Service Session Key. It's encrypted by the TGS Session Key.
- Message #6 This message, which is the ST, contains service name, the username, and the Service Session Key. It's encrypted by the Service Secret Key.

4. The TGS sends messages 5 and 6 to the user. The user decrypts Message #5 with the TGS Session Key and obtains the service name and the Service Session Key. It then creates a new user authenticator, encrypted by the Service Session Key (Message #7).
5. The user sends the encrypted ST (Message #6) and the user authenticator (Message #7) to the service. The service decrypts the encrypted ST (since it has access to its own secret key), obtaining the username, service name, and the Service Session Key. It now uses the Service Session Key to decrypt the user authenticator (Message #7). Then it checks if the username in the ST is the same as the username in the user authenticator. Finally, it creates a service authenticator encrypted by the Service Session Key (Message #8).
6. The service sends the service authenticator (Message #8) to the user, who then decrypts it. The user checks if the service name in the service authenticator is actually the service it requested. If yes, the authentication is successfully completed.

How Kerberoasting Happens

You might have already noticed that Step 4 of the Kerberos authentication protocol has an inherent weakness. It requires the TGS to send a service ticket, encrypted by the service secret key, to the user. At this point the TGS doesn't even know if the user to whom it's sending the ST has access to the application. It just receives the TGT from the user and creates a ST.

You can therefore have a situation where any legitimate network user can request the AS for a TGT, and then present this TGT to the TGS, and get back a ST encrypted with the service secret key. What if an attacker has already compromised a legitimate user's account? The attacker can then get the ST along with the service secret key. Then they could crack the service's password using a tool such as Hashcat. Figure 4-2 shows the steps of Kerberos authentication and where Kerberoasting takes place (marked with the dotted arrow).

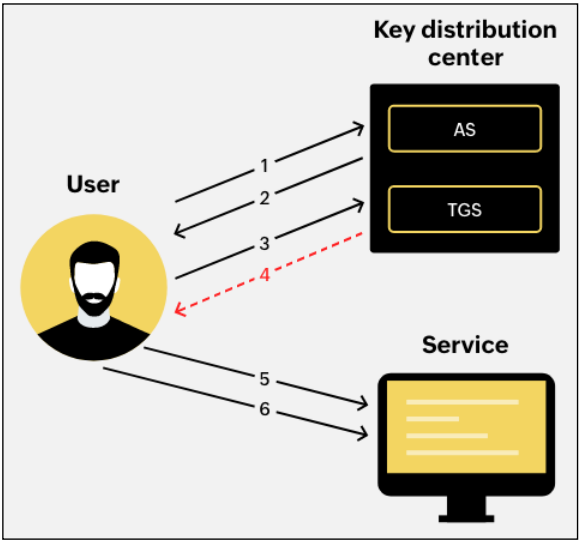


FIGURE 4-2: Kerberoasting takes place at step #4 when attacker cracks the password of the service.

If the service account they cracked has domain administrator privileges, then there's even more trouble! The attacker can then gain more privileged access.

Figure 4-3 shows where I obtained a service account's password hash using the GetUserSPNs.py command. All I needed to provide was a valid user account and its password.

From here, it would be a simple matter of cracking it with Hashcat.

With this Kerberoast, an attacker can go on to perform something called a *silver ticket attack* as well. In a silver ticket attack, after cracking a service's password, the attacker can easily create service tickets on demand with any user account.

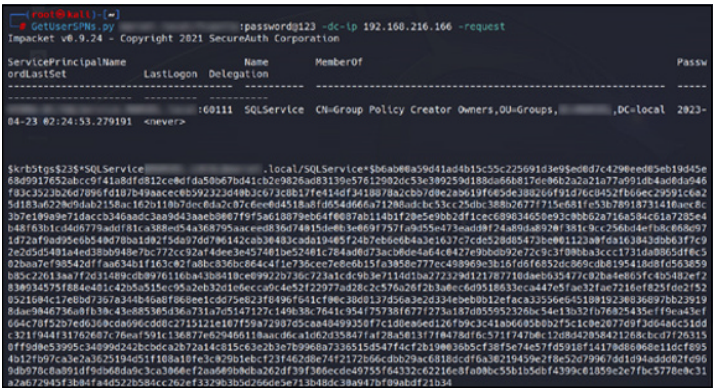


FIGURE 4-3: Obtaining the service account's password hash with the GetUserSPNs.py command.

You just saw that it's possible for the attacker to crack the service account's password, but what if the attacker goes after something even more lucrative? In Step 2 of Kerberos authentication, the AS sends the TGT to the user, encrypted by the TGS Secret Key. Recall that the TGS Secret Key is the hash of the KRBTGT account of Active Directory. If an attacker cracks this, they've hit gold! After all, the TGS Secret Key is used to encrypt every TGT sent to every user. This is why an attack at this juncture is called a *golden ticket attack*. The attacker can literally create TGTs as they fancy.

Protecting Against Kerberoasting

TIP

You need to do four things to protect against Kerberoasting attacks:

- **Have a strong password policy.** Ensure that all user accounts require strong passwords, and even stronger ones for service accounts. Attackers can use automated tools to crack weak passwords within seconds, so users must set up passwords with sufficient length and complexity. User account passwords should be at least 12 characters in length, and service account passwords at least 20 characters. Using a randomly generated password that's stored and managed through a password vault is even more effective. Administrators must also change the service account

passwords frequently. There are solutions available that can enable you to do this easily.

- **Enforce a strong Kerberos encryption algorithm.** Make sure that your AD administrator enforces a strong Kerberos encryption algorithm. They can do this by editing their domain policy. Encryption methods that use AES128 and higher are more difficult to crack.
- **Never make a service account a domain administrator.** In many organizations, service accounts are set up as domain administrators because they run many automated services and processes, and lack of privilege may lead to service downtime. This isn't a safe practice, though. Administrators must limit the resources the service account can access to what's necessary. One way to accomplish this is to create service accounts to perform specific tasks, and to always follow the principle of least privilege. As a security analyst, you must oversee if this is done properly, and highlight any issues.
- **Detect Kerberoasting attacks with an effective security information and event management (SIEM) solution.** Windows Event ID 4769 is generated every time the KDC receives a ST request from a user account. Of course, not every such request is malicious. In modern Windows versions, non-malicious STs are mostly AES-encrypted, but a malicious ST will be RC4-encrypted. An administrator can detect this by configuring the correct filters and alerts in their SIEM. They must also look for any abnormally high numbers of ST requests from a single user account, especially if these occur in a short span of time. Figure 4-4 shows how a SIEM like Log360 captures ST requests from different users in AD.

USER NAME	CLIENT HOST NAME	DOMAIN CONTROLLER	LOGON TIME	EVENT TYPE	FAILURE REASON	REMARKS	MESSAGE
harry	admandemo.admanagerplus.com	admandemo.admanagerplus.com	Sep 01,2023 02:25:05 AM	Failure	Account disabled, expired, or locked out	A Kerberos authentication ticket (TGT) was requested.	A Kerberos authentication ticket (TGT) was requested for harry from admandemo.admanagerplus.com. Status : Failure. Error : Account disabled, expired, or locked out.
adap	admandemo.admanagerplus.com	admandemo.admanagerplus.com	Sep 01,2023 02:15:37 AM	Success	-	A Kerberos authentication ticket (TGT) was requested.	A Kerberos authentication ticket (TGT) was requested for adap from admandemo.admanagerplus.com. Status : Success.
adap	admandemo.admanagerplus.com	admandemo.admanagerplus.com	Sep 01,2023 02:15:36 AM	Success	-	A Kerberos authentication ticket (TGT) was requested.	A Kerberos authentication ticket (TGT) was requested for adap from admandemo.admanagerplus.com. Status : Success.
adap	admandemo.admanagerplus.com	admandemo.admanagerplus.com	Sep 01,2023 02:15:35 AM	Success	-	A Kerberos authentication ticket (TGT) was requested.	A Kerberos authentication ticket (TGT) was requested for adap from admandemo.admanagerplus.com. Status : Success.

FIGURE 4-4: Capturing ST requests in a SIEM like ManageEngine Log360.

IN THIS CHAPTER

- Enumerating domains with some top PowerShell commands
- Detecting domain enumeration attempts immediately with a SIEM

Chapter 5
Diving into Domain Enumeration

If attackers gain a foothold into your network, the first thing they may want to do is domain enumeration. *Domain enumeration* is the process of digging deep and finding intricate details about an AD domain. It's an integral part of an attacker's playbook and helps them design their attack strategy. As a defender, you need to understand exactly what an attacker may look for during this phase of reconnaissance so you can be better placed at detecting this activity quickly.

Enumerating Domains with PowerView

Attackers can use many of the free tools available online to perform a domain enumeration against your organization. One of the most popular tools is *PowerView*, which is available on GitHub.

Written by Will Shroeder, a well-known security researcher and pen-tester, PowerView contains many functions that can help enumerate a domain. PowerView is written in PowerShell and utilizes AD hooks and Win32 API functions to do its job. With PowerView, an attacker can get to know your AD domain like it's their own!

The following sections outline just six of the many commands attackers can use to discover sensitive things about your domain. By knowing what attackers care about, you can take measures to detect their activity.

Get-NetDomain

This command (see Figure 5-1) gives information about the user's domain. If the attacker has already compromised a user account, they can find out details about this user's domain.

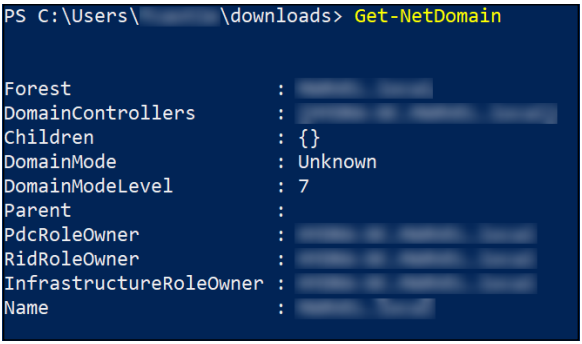


FIGURE 5-1: Get-NetDomain gives attackers useful information about the AD domain.

Get-NetDomainController

This gives the attacker details about the domain controller to which the compromised user belongs, as shown in Figure 5-2. This information can then be used to greater effect as the attacker burrows deeper into the network.

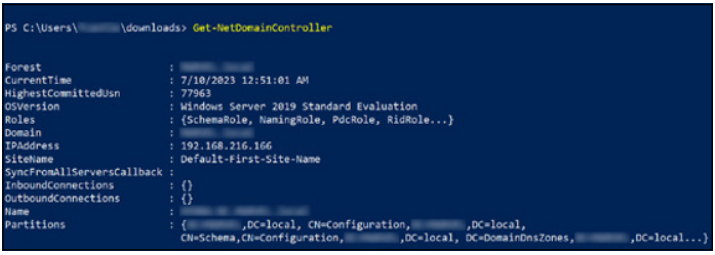


FIGURE 5-2: Get-NetDomainController gives attackers useful information about the domain controller.

Get-DomainPolicy

This command, shown in Figure 5-3, provides details about the domain policy. The attacker can clearly see sensitive information, including policies about system access.

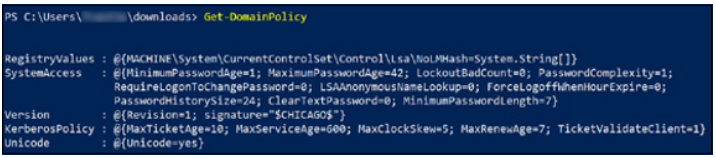


FIGURE 5-3: Get-DomainPolicy shows attackers policies applicable for the domain.

(Get-DomainPolicy)."SystemAccess"

With this command, the attacker can go find out intricate details about your system access policy, as shown in Figure 5-4. For example, they can see the number of failed password attempts allowed before the account is locked out. They can also see the minimum password requirements you've set for the domain, and can use this information when designing a brute force attack.

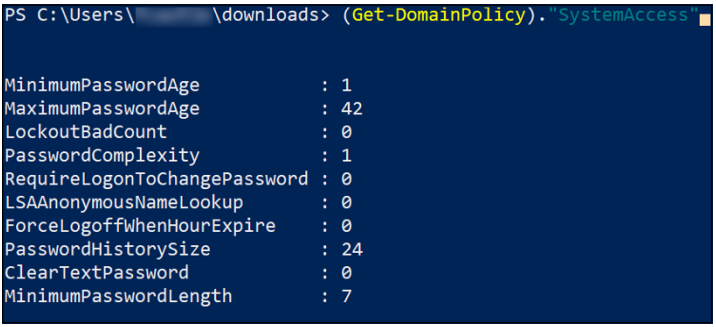


FIGURE 5-4: This command lets attackers discover policies on account lockouts, password complexity, and much more.

Get-NetGroupMember -GroupName "Domain Admins"

With this command, shown in Figure 5-5, an attacker can get all the members of the security group called Domain Admins. Domain Admins is a privileged group and only domain administrators are

supposed to be a part of this group. An attacker can use this information to find out which accounts they need to ultimately compromise to get to domain admin privileges. They can also discover any service account which is part of this group.

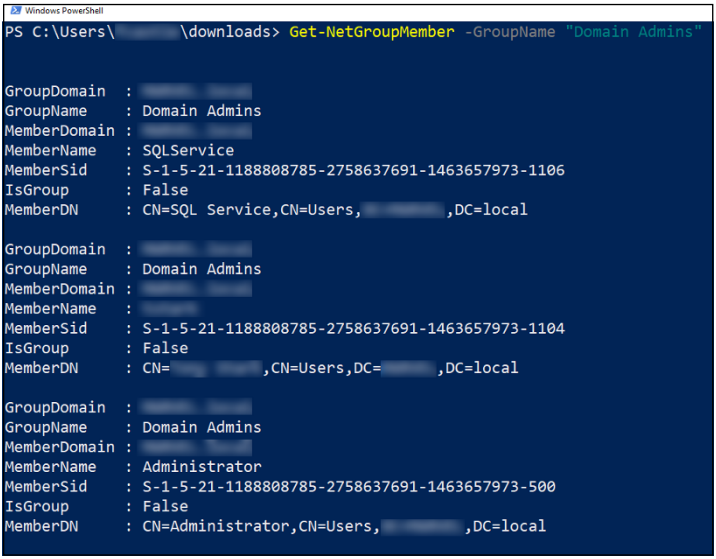


FIGURE 5-5: Attackers can find out all members of a privileged group with this command.

Invoke-ShareFinder

An attacker can discover all the network shares you've configured with this command. There are numerous default shares in any AD domain plus there'll always be some administrator-created shares. If the attacker finds network shares with names that sound important (such as Classified), they'll likely make that their target. You can see the result of running this command in Figure 5-6.

```
PS C:\Users\[redacted]\downloads> Invoke-ShareFinder
[redacted]local\ADMIN$      - Remote Admin
[redacted]local\C$  - Default share
[redacted]local\IPC$        - Remote IPC
[redacted]local\Share       -
[redacted]local\Users       -
[redacted]l\ADMIN$  - Remote Admin
[redacted]local\C$      - Default share
[redacted]local\hackme  -
[redacted]local\IPC$    - Remote IPC
[redacted]local\NETLOGON        - Logon server share
[redacted]local\SYSVOL  - Logon server share
[redacted]local\ADMIN$      - Remote Admin
[redacted]local\C$  - Default share
[redacted]local\IPC$        - Remote IPC
[redacted]local\Share       -
[redacted]local\Users       -
PS C:\Users\[redacted]\downloads>
```

FIGURE 5-6: Invoke-ShareFinder lets attackers find out network shares in the domain.

Detecting Domain Enumeration

You need to know about any PowerShell-based domain enumeration attempt, so ask your AD administrator to enable both script block logging and module logging. *Script block logging* provides all the PowerShell commands in their raw form, and *module logging* enables you to apply filters and see specific types of command invocations.

Once these services are enabled, you can detect possible malicious use of PowerShell within your organization using your SIEM solution. For example, in ManageEngine Log360, you can view all the users who've run PowerShell during a period of your choice, as shown in Figure 5-7. You can go on to view the actual script details of the commands run as well. This will give you ready analytics about user accounts which may be misusing PowerShell within your network.

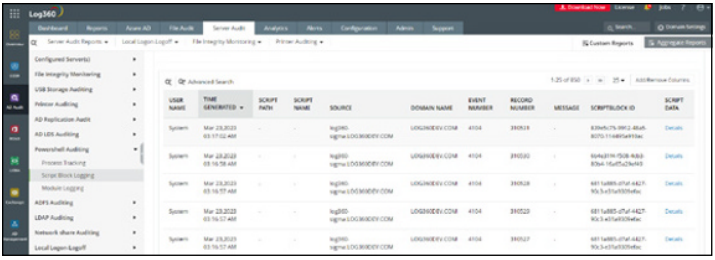


FIGURE 5-7: Script Block Logging in Log360 gives you details about PowerShell use within your organization.

Of course, not all PowerShell scripts are malicious. After all, administrators use PowerShell all the time to get things done quickly. It's the *misuse* of PowerShell that you're after. You can use the alerting functionality of your SIEM solution to look for such misuse. With alerts, you can configure rules, and get notified when particular types of PowerShell usage occur. For example, you can look for instances when a regular domain account uses Invoke-ShareFinder or Get-DomainPolicy. You can also look for times when a particular user is running multiple PowerShell commands in short succession (indicating the use of scripting), or when PowerShell is being used outside of normal working hours.

Attackers can use PowerShell not just for domain enumeration but for other nefarious designs as well. PowerShell can also be used to launch malware deep into a network, or perform brute force and password spray attacks, among other things. You can detect all of this if you monitor PowerShell use and get alerts using your SIEM solution.

IN THIS CHAPTER

- Introducing brute force attacks
- Understanding password spray attacks
- Launching a password spray attack with PowerShell
- Detecting brute force and password spray attacks with your SIEM

Chapter 6
Breaking Down Brute Force and Password Spray Attacks

This chapter explains how brute force attacks happen, as well as password spray attacks. You'll learn how attackers launch these attacks and how to detect them with your security information and event management (SIEM).

Punching into Brute Force Attacks

In a brute force attack, cybercriminals try to guess the password of a target user account, analyze the result, and try again until they succeed. This is a simple, non-sophisticated attack akin to a burglar trying to break into a locked house, but most attacks include it in some form. During the COVID-19 pandemic, when many people first started working from home, many attackers tried brute forcing into networks by going after weak remote desktop protocol (RDP) accounts.

Today, an attacker almost always automates their brute force attack. You can get the gist of this in Figure 6-1.

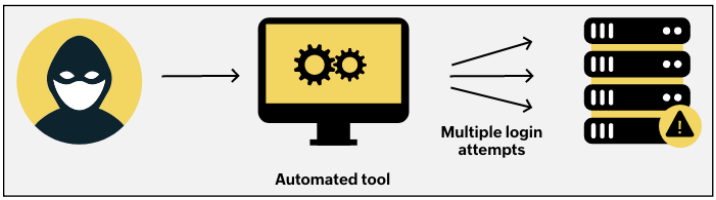


FIGURE 6-1: The modus operandi of a brute force attack.

Here are the four steps in this attack:

1. The attacker obtains a list of valid usernames of the victim organization, which they can put into a file. This is simple to obtain for someone who's done their initial due diligence.
2. The attacker assembles a list of passwords that they want to try with each of the usernames. These passwords can be commonly used passwords such as password@123, admin, welcome, summer@2020, and so on.
3. The attacker inputs the username and password lists into an automated tool.
4. The tool attempt the brute force and lets the attacker know whenever there's success.

Explaining Password Spray Attacks

You can think of a *password spray attack* as just the opposite of a brute force attack. That's why it's also called a *reverse brute force attack*. In brute force, an attacker goes after a small set of user accounts by trying different passwords; in a password spray, the attacker goes after multiple user accounts with a smaller list of passwords. The logic is that at least some of the users will have a commonly used password. You can see how this works in Figure 6-2.

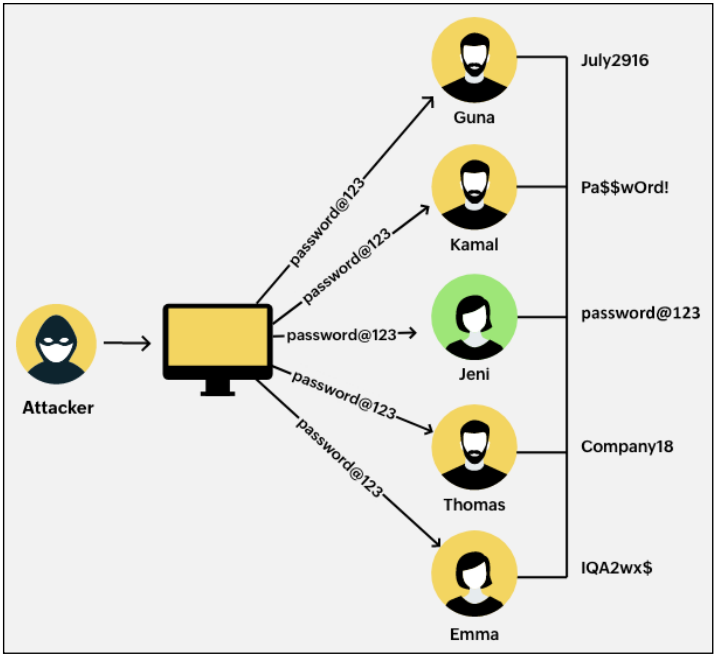


FIGURE 6-2: Overview of a password spray attack.

In Figure 6-2, the attacker is almost certain that at least one user in the network uses the password password@123, so they try this password against multiple usernames. The user account Jeni is then compromised.

Launching Password Spray with PowerShell

There are many ways an attacker can launch a password spray attack. One popular method is to leverage PowerShell. Scripts are easily available online for password spray attacks. The attacker simply downloads a script and launches the attack. Figure 6-3 shows the result of a typical password spray attack. This example uses the password *password@123* against all user accounts in the AD domain.

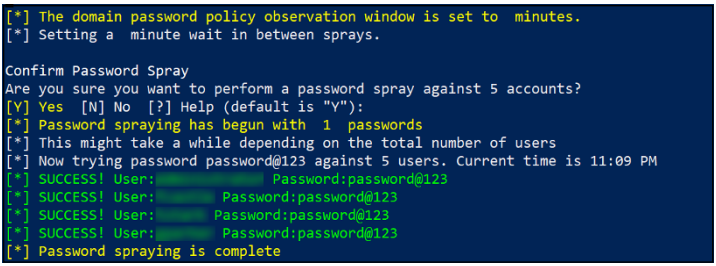


FIGURE 6-3: The results of a simple password spray attack.

Mitigating Brute Force and Password Spray Attacks

TIP

Here's how you can mitigate a brute force or password spray attack:

- **Identify the attack immediately.** Continuous login failures are the sign of brute force attacks. Auditing users' login attempts is one way to track login attempts and act on suspicious repetitive failures.
- **Use effective correlation rules in your SIEM.** You can create correlation rules to track and identify brute force attacks that happen in sensitive areas of your network.
- **Prevent the attack in the first place.** Preventing a hacker with brute-forced credentials from logging into the system is another way to mitigate attacks. Multifactor authentication (MFA), conditional access, and CAPTCHA are useful in preventing brute-force attacks.

IN THIS CHAPTER

- Using strong passwords and password policies
- Tracking the use of PowerShell in real time
- Leveraging the MITRE ATT&CK framework to understand adversary behavior
- Using a SIEM for effective detection

Chapter 7
(Not Quite) Ten Effective AD Defense Techniques

Having the right defense techniques in place to protect your Active Directory can potentially save thousands of dollars for your organization. Security analysts can make a big impact for the company's bottom line and reputation by addressing these suggestions.

Discover Devices Automatically

An AD domain can be a complex environment with numerous member servers, file servers, printers, workstations, and domain controllers. In addition, this environment can also frequently change with new devices being added. In such an environment, you should be able to detect new devices automatically and analyze the events happening on them. A SIEM solution can give you this required visibility to protect your company.

Use Strong User Account Passwords

You need to ensure that all accounts use strong passwords. The best way to do this is to make users aware of the importance of strong passwords. A user is usually the weakest link in the network, and most attackers get in by compromising a single user account. An attacker who gets in by compromising a user account can move laterally, escalate privileges, and get to domain administrator within 48 hours.

TIP

Ask your AD administrator to set strong password policies to make sure:

- Passwords are complex with a mix of different types of required characters.
- Passwords have adequate length. User accounts should have passwords that are at least 12 characters, and service accounts should have passwords that are at least 20 characters. Even better, recommend that users employ passphrases instead; they're easier to remember but more difficult to crack.
- Multi-factor authentication is enabled.
- A decision is taken about the frequency of password change. There was a time when it was recommended to change passwords frequently with secure password-history requirements. As a general rule of thumb, passwords needed to be changed at least once every 90 days. Nowadays, many experts don't recommend the changing of passwords, as this leads to users setting easy-to-remember passwords. Instead, they recommend the use of hard-to-crack passwords.
- Passwords that are in use in your organization aren't found in the database of the website `HaveIbeenPwned.com`, where you can check if any passwords have been compromised. Leverage HaveIbeenPwned and ensure your users don't use compromised passwords.
- You use an effective solution that can help you implement strong password policies, and monitor them.

Adhere to the Principle of Least Privilege

Ensure that non-privileged users don't get heightened privileges. When this restriction isn't enforced properly, attackers can easily move laterally and steal the crown jewels of your organization. Some examples of non-privileged users being added to privileged groups include a user account that is a local administrator on more than one machine, a service account that has been added to the domain administrators group, and a user from the Sales team being added to a particularly sensitive finance security group. Another example is an AD environment where SMB signing is enabled but not required for user workstations.

As a security analyst, you must be able to view the current privileges for every user and service account, and you need to be alerted about instances where permissions have been elevated.

Make Use of the MITRE ATT&CK Framework

The MITRE ATT&CK framework breaks down adversary behavior in tactics, techniques, and sub-techniques, and enables you to think in a logical manner about defense. The framework highlights numerous AD-based attacks. Make a list of the attacks that you're most concerned about, then search the MITRE ATT&CK framework to look for prevention, detection, and mitigation strategies for them.

Monitor PowerShell Use

PowerShell is a great tool for managing and administering AD. However, in the hands of an attacker, PowerShell can be a dangerous weapon. It can be used to perform brute force and password spray attacks, domain enumeration, and even malware and ransomware attacks. Other applications can also use PowerShell to start a malicious process.

To continuously monitor PowerShell use across all users and service accounts, you must first enable script block and module logging. After this, you can use a SIEM solution like Log360

that gives you real time analytics on PowerShell use. You must also create relevant alerts to stay ahead of attackers who may use PowerShell.

Detect Anomalous User and Entity Behavior

Use your SIEM solution to detect anomalous user and entity behavior in your network. For many attackers, it's easy to compromise a legitimate user's credentials; however, it's hard for them to mimic their usual behavior. A SIEM solution with anomaly detection capabilities will first create behavior baselines for every user and entity in the network, using machine learning algorithms. It will then actively look for events where an observed behavior deviates from the baseline behavior. When this happens, the corresponding user's or entity's risk score is also increased.

As a security analyst, you can get alerted when an anomalous behavior is detected or when a risk score increases beyond the defined threshold for an acceptable value.

Anomaly detection powered by machine learning can also help you spot the use of a compromised credential. It can alert you whenever a strange logon happens from a different host, different geography, different IP, different time zone, and so on. Spotting anomalies is essential for tracking lateral movement into a cloud infrastructure.

Track Kerberoasting, Silver Ticket, and Golden Ticket Attacks

Kerberoasting is a popular attack wherein an attacker can crack the password of an application or service. To protect against Kerberoasting, you should enable AES Kerberos encryption instead of RC4. You also should focus on your password policies and limit service account privileges.

You must also investigate irregular patterns of activity such as accounts making numerous Event ID 4769 requests within a short time, especially if they also request RC4 encryption.

ManageEngine
Log360

Empower
YOUR SECURITY TEAM
with Log360 SIEM

Software that's simple to deploy and easy to use. Spur your threat detection and response, and create value for your business.

Get a demo

3 STEPS TO SECURITY SUCCESS

STEP 1

Enhance your AD security

STEP 2

Protect against the latest cyberattacks

STEP 3

Save money for your business

Protect your data from cyber attackers

Active Directory (AD) is used by millions of organizations around the world to authenticate and authorize users within their network. AD stores details about users, their passwords, and their access privileges. It's no surprise that cyber attackers target AD to get a foothold into a network, and ultimately exfiltrate data.

If you're a security analyst who works within the security operations center of your organization, you need to play a major part in keeping attacks at bay. You need to use a security information and events management (SIEM) solution to constantly look for and detect threats, investigate the probable root cause, and respond. *Defending Against Active Directory Attacks For Dummies* explains how to detect and respond to these attacks with a SIEM solution. Attackers beware!

Inside…

- Understand Active Directory's vulnerabilities
- Know the threat of an LLMNR/NBT-NS poisoning attack
- Defend against SMB relay attacks
- Protect against a Kerberoasting attack
- Detect domain enumeration
- Guard against password spray attacks

ManageEngine

Go to Dummies.com™
for videos, step-by-step photos, how-to articles, or to shop!

ISBN: 978-1-394-20795-4
Not For Resale

for dummies®
A Wiley Brand

WILEY END USER LICENSE AGREEMENT

Go to www.wiley.com/go/eula to access Wiley’s ebook EULA.