Credential Access

This lesson is going to cover the fourth stage in the MITRE ATT&CK framework, Credential Access. These techniques are used to describe ways that adversaries will work to steal credentials such as passwords and usernames from compromised systems using methods such as credential dumping (retrieving credentials that are stored in memory while the system is powered on) or deploying a key logger to monitor what keyboard buttons are pressed. At the time of writing this category currently includes 14 top-level techniques. **We will be looking at two of the big ones in this category:**

- OS Credential Dumping
- Brute Force

MITRE Technique T1003

Adversaries with local access to a system can work to retrieve passwords from the operating system and running applications. There are 8 sub-techniques in this technique, and we're going to look at the following two:

- LSASS Memory
- /etc/passwd and /etc/shadow

LSASS Memory – T1003.1

Malicious actors may attempt to retrieve credentials stored in the memory of the process for **Local Security Authority Subsystem Service (LSASS)**. When a user logs onto a Windows system their credentials are stored in LSASS process memory which can be accessed by an Administrator or a SYSTEM-level user. If the adversary has control of an admin account they are able to dump everything from the LSASS memory and then brute force the password hashes offline to retrieve the plaintext versions and then use the list of valid credentials to log into other accounts and systems within the network!

/etc/passwd /etc/shadow – T1003.8

We've actually covered exactly how this works in the Digital Forensics domain of BTL1 within the Linux Investigations section! (If you haven't already done the exercise where you crack the passwd and shadow files using John The Ripper, go back and do it now so you can fully understand how this works). Adversaries may attempt to dump the contents of `/etc/passwd` and `/etc/shadow` to enable offline password cracking, but note that /etc/shadow can only be accessed by a root-level user because it holds all user accounts passwords while /etc/passwd holds all usernames (confusing that 'passwd' isn't password, right?!). Tools such as John The Ripper can brute force and crack the password hashes and reveal the plaintext versions which can be used to log into the system.

Moving back to the main technique of Credential Access we can see that APT28 have been observed using Mimikatz to dump credentials from LSASS memory allowing them to crack the password hashes offline and then use the access to legitimate accounts to access other users and systems. APT32 have used a different tool called GetPassword_x64 to retrieve credentials, and APT39 have also used Mimikatz to collect valid usernames and passwords.

Procedure Examples

Name	Description
APT28	APT28 regularly deploys both publicly available (ex: Mimikatz) and custom password retrieval tools on victims.[20][21]
APT32	APT32 used GetPassword_x64 to harvest credentials.[22][23]
APT39	APT39 has used different versions of Mimikatz to obtain credentials.[24]

For Mitigations we have picked out a few to focus our attention on. Local administrator accounts should have unique and complex passwords for every system that has that account enabled on. It is not enough to have one strong password used on all local admins across an organisation, because if the attacker can dump OS credentials from a Windows system then they can log into every other system using that username and password combination! Privileged Account Management (PAM) is a huge area of cybersecurity that requires a lot of time and attention to ensure that accounts with higher privileges than standard users are properly secured and managed to prevent

compromise and allow for lateral movement and privilege escalation. And finally user training should be employed to ensure users are familiar with why passwords should be different and that one password shouldn't be used for multiple systems or services to prevent it being used in a password reuse attack.

Password Policies	Ensure that local administrator accounts have complex, unique passwords across all systems on the network.
Privileged Account Management	Windows:Do not put user or admin domain accounts in the local administrator groups across systems unless they are tightly controlled, as this is often equivalent to having a local administrator account with the same password on all systems. Follow best practices for design and administration of an enterprise network to limit privileged account use across administrative tiers.[1] Linux:Scraping the passwords from memory requires root privileges. Follow best practices in restricting access to privileged accounts to avoid hostile programs from accessing such sensitive regions of memory.
User Training	Limit credential overlap across accounts and systems by training users and administrators not to use the same password for multiple accounts.

MITRE offers a **lot** of recommendations for detecting activity related to OS credential dumping so we'll cover the main points for Windows and Linux systems. For Windows we should be monitoring for activity related to lsass.exe as this could represent malicious activity such as credential dumping from memory.

Windows

Monitor for unexpected processes interacting with lsass.exe.[26] Common credential dumpers such as Mimikatz access the LSA Subsystem Service (LSASS) process by opening the process, locating the LSA secrets key, and decrypting the sections in memory where credential details are stored. Credential dumpers may also use methods for reflective Process Injection to reduce potential indicators of malicious activity.

And for Linux systems we can make use of the monitoring tool AuditD to detect malicious processes opening a maps file which can generate an alert which a security analyst could investigate to determine exactly what is happening.

Linux

To obtain the passwords and hashes stored in memory, processes must open a maps file in the /proc filesystem for the process being analyzed. This file is stored under the path `/proc/*/maps`, where the directory is the unique pid of the program being interrogated for such authentication data. The AuditD monitoring tool, which ships stock in many Linux distributions, can be used to watch for hostile processes opening this file in the proc file system, alerting on the pid, process name, and arguments of such programs.

MITRE Technique T1110

There are two main paths that an attacker can go down that would involve brute force. The first would be when they do not have access to any valid credentials and they need to guess a correct username and password combination. Obviously doing this manually would likely completely ineffective, so they could utilise a username list and a password list and automatically provide these credentials to a system, hoping they find a valid combination. Alternatively instead of using a password list (as the password needs to be in the list for a successful authentication to occur) the attacker can instead iterate through every possible combination of a password using alphanumeric and special character. This method will always get the password, but depending on the complexity it could take millions of years (chuck a random password into How Secure Is My Password? to see how long it could take to guess!). The second scenario would be where an attacker has performed OS credential dumping and has obtained usernames and hashed passwords which need to be cracked to reveal the plaintext passwords. Attackers can utilise offline tools such as Hashcat to attack the encrypted passwords by brute forcing them until the plaintext password match has been found.

Looking at the Procedure Examples table we can see that APT39 have previously used Ncrack to discover credentials. Ncrack is a tool developed by the same team as Nmap and allows for network brute forcing to detect accounts and services using weak credentials, and is legitimately used for security and auditing purposes but can just as easily be used for malicious actions. We can also see that Chaos has historically launched brute force attacks against the SSH service to identify a valid account and log in so they can execute commands. DarkVishnya have also conducted brute force attacks to obtain credentials.

Procedure Examples

Name	Description
APT39	APT39 has used Ncrack to reveal credentials.[8]
Chaos	Chaos conducts brute force attacks against SSH services to gain initial access.[3]
DarkVishnya	DarkVishnya used brute-force attack to obtain login data.[9]

The Mitigations section for this technique offers some extremely useful and effective advice. Account lockout policies can help to lock accounts after a threshold of failed login attempts have occurred to prevent continued brute force attacks that may eventually guess the correct username and password combination. It is important to set an appropriate lockout threshold for your environment as you don't want users getting locked out after getting their password wrong a few times! Multi-factor authentication is absolutely key to reducing the effectiveness of this kind of attack. If an attacker successfully guesses your password but you have two-factor authentication set up using SMS messages to your phone then they won't be able to login (but be aware that there are mechanisms and tricks to bypass 2FA/MFA, but it creates another barrier that attackers need to get through). NIST offers some good guidelines on creating password policies to increase the time it would take to brute force credentials, you can read their suggestions here. Finally, if an organisation has a threat intelligence function they should be keeping an eye out on the latest data breaches, obtain breach lists (through dark web monitoring or specialist threat intelligence vendors) and identify if any company accounts are included in the leaked data – if they are, immediate issue a

password reset for that account so that malicious actors can't successfully login.

Mitigations

Mitigation	Description
Account Use Policies	Set account lockout policies after a certain number of failed login attempts to prevent passwords from being guessed. Too strict a policy may create a denial of service condition and render environments un-usable, with all accounts used in the brute force being locked-out.
Multi-factor Authentication	Use multi-factor authentication. Where possible, also enable multi-factor authentication on externally facing services.
Password Policies	Refer to NIST guidelines when creating password policies.[1]
User Account Management	Proactively reset accounts that are known to be part of breached credentials either immediately, or after detecting bruteforce attempts.

MITRE suggests that organisations monitor for logs generated when an account failures to successfully authenticate, as a high number of these in a short period of time (or even over a long period of time) could represent a brute force attack against that account. The golden log to monitor for in Windows environments is Windows Security Log Event ID 4625 "An account failed to logon". This log will also very helpfully include an error code which will tell us exactly why the account failed to login, from an expired password to the account currently being locked out. Take a look at that link and find the error code table, what is the failure reason associated with **0xC000006A**?

Detection

Monitor authentication logs for system and application login failures of Valid Accounts. If authentication failures are high, then there may be a brute force attempt to gain access to a system using legitimate credentials. Also monitor for many failed authentication attempts across various accounts that may result from password spraying attempts. It is difficult to detect when hashes are cracked, since this is generally done outside the scope of the target network.

Previous Topic

Mark Complete

Next Topic

Back to Lesson