

CrackMapExec Module Library

 infosecmatter.com/crackmapexec-module-library

July 6, 2021



On this page you will find a comprehensive list of all **CrackMapExec modules** that are currently available in the latest public version (5.1.7dev) of [CrackMapExec](https://github.com/Alc0n/CrackMapExec), one of the most capable tools for pentesting internal networks.

CrackMapExec (or CME) contains a number of modules which makes this tool so useful. I'm hoping that this list will help you navigate through all the modules more easily and gives you information on how to use them.

Introduction

In the latest version of CME, there are 68 modules in total. These modules are all in the post exploitation category, complementing the CME's powerful login brute force capabilities and password spraying attack features. On this page, however, you will find information only related to the modules.

If you are looking for how to use CrackMapExec in general, please check these excellent resources:

Below you can find the list of CME modules as shown while using the tool. CME currently supports the following network protocols:

- **LDAP** (port 389 or 636) – 5 modules

- **MSSQL** (port 1433) – 23 modules
- **SMB** (port 135, 139 or 445) – 39 modules
- **SSH** (port 22) – 1 module
- **WinRM** (port 5985 or 5986) – 0 modules

Alright, let's get to the actual lists. By clicking on the module links you will find detailed information about each module with examples on how to use it.

CME LDAP modules

Here's a list of all CrackMapExec modules that can be used with LDAP protocol:

```
# cme ldap -L
[*] MAQ attribute Retrieves the MachineAccountQuota domain-level
[*] adcs Find PKI Enrollment Services in Active Directory
[*] get-desc-users Get description of the users. May contained password
[*] laps Retrieves the LAPS passwords
[*] user-desc Get user descriptions stored in Active Directory
```

CME MSSQL modules

Here's a list of all CrackMapExec modules that can be used with MSSQL protocol:

cme mssql -L

[*] <u>Get-ComputerDetails</u>	Enumerates sysinfo
[*] <u>empire_exec</u>	Uses Empire's RESTful API to generate a launcher for the specified listener and executes it
[*] <u>enum_chrome</u>	Decrypts saved Chrome passwords using Get-ChromeDump
[*] <u>get_keystrokes</u>	Logs keys pressed, time and the active window
[*] <u>get_netdomaincontroller</u>	Enumerates all domain controllers
[*] <u>get_netrdpsession</u>	Enumerates all active RDP sessions
[*] <u>get_timscreenshots</u>	Takes screenshots at a regular interval
[*] <u>invoke_sessiongopher</u>	Digs up saved session information for PuTTY, WinSCP, FileZilla, SuperPuTTY, and RDP using SessionGopher
[*] <u>invoke_vnc</u>	Injects a VNC client in memory
[*] <u>met_inject</u>	Downloads the Meterpreter stager and injects it into memory
[*] <u>mimikatz</u>	Dumps all logon credentials from memory
[*] <u>mimikatz_enum_chrome</u>	Decrypts saved Chrome passwords using Mimikatz
[*] <u>mimikatz_enum_vault_creds</u>	Decrypts saved credentials in Windows Vault/Credential Manager
[*] <u>mimikittenz</u>	Executes Mimikittenz
[*] <u>mssql_priv</u>	Enumerate and exploit MSSQL privileges
[*] <u>multirdp</u>	Patches terminal services in memory to allow multiple RDP users
[*] <u>netripper</u>	Capture's credentials by using API hooking
[*] <u>pe_inject</u>	Downloads the specified DLL/EXE and injects it into memory
[*] <u>rid_hijack</u>	Executes the RID hijacking persistence hook.
[*] <u>shellcode_inject</u>	Downloads the specified raw shellcode and injects it into memory
[*] <u>test_connection</u>	Pings a host
[*] <u>tokens</u>	Enumerates available tokens
[*] <u>web_delivery</u>	Kicks off a Metasploit Payload using the exploit/multi/script/web_delivery module

CME SMB modules

Here's a list of all CrackMapExec modules that can be used with SMB protocol:

cme smb -L

[*] <u>Get-ComputerDetails</u>	Enumerates sysinfo
[*] <u>bh owned</u>	Set pwned computer as owned in Bloodhound
[*] <u>bloodhound</u>	Executes the BloodHound recon script on the target and retrieves the results to the attackers' machine
[*] <u>empire_exec</u>	Uses Empire's RESTful API to generate a launcher for the specified listener and executes it
[*] <u>enum_avproducts</u>	Gathers information on all endpoint protection solutions installed on the the remote host(s) via WMI
[*] <u>enum_chrome</u>	Decrypts saved Chrome passwords using Get-ChromeDump
[*] <u>enum_dns</u>	Uses WMI to dump DNS from an AD DNS Server
[*] <u>get_keystrokes</u>	Logs keys pressed, time and the active window
[*] <u>get_netdomaincontroller</u>	Enumerates all domain controllers
[*] <u>get_netrdpsession</u>	Enumerates all active RDP sessions
[*] <u>get_timedsscreenshot</u>	Takes screenshots at a regular interval
[*] <u>gpp_autologin</u>	Searches the domain controller for registry.xml to find autologon information and returns the username and password.
[*] <u>gpp_password</u>	Retrieves the plaintext password and other information for accounts pushed through Group Policy Preferences.
[*] <u>invoke_sessiongopher</u>	Digs up saved session information for PuTTY, WinSCP, FileZilla, SuperPuTTY, and RDP using SessionGopher
[*] <u>invoke_vnc</u>	Injects a VNC client in memory
[*] <u>lsassy</u>	Dump lsass and parse the result remotely with lsassy
[*] <u>met_inject</u>	Downloads the Meterpreter stager and injects it into memory
[*] <u>mimikatz</u>	Dumps all logon credentials from memory
[*] <u>mimikatz_enum_chrome</u>	Decrypts saved Chrome passwords using Mimikatz
[*] <u>mimikatz_enum_vault_creds</u>	Decrypts saved credentials in Windows Vault/Credential Manager
[*] <u>mimikittenz</u>	Executes Mimikittenz
[*] <u>multirdp</u>	Patches terminal services in memory to allow multiple RDP users
[*] <u>netripper</u>	Capture's credentials by using API hooking
[*] <u>pe_inject</u>	Downloads the specified DLL/EXE and injects it into memory
[*] <u>rdp</u>	Enables/Disables RDP
[*] <u>rid_hijack</u>	Executes the RID hijacking persistence hook.
[*] <u>runasppl</u>	Check if the registry value RunAsPPL is set or not
[*] <u>scuffy</u>	Creates and dumps an arbitrary .scf file with the icon property containing a UNC path to the declared SMB server against all writeable shares
[*] <u>shellcode_inject</u>	Downloads the specified raw shellcode and injects it into memory
[*] <u>slinky</u>	Creates windows shortcuts with the icon attribute containing a UNC path to the specified SMB server in all shares with write permissions
[*] <u>spider_plus</u>	List files on the target server (excluding `DIR` directories and `EXT` extensions) and save them to the `OUTPUT` directory if they are smaller than `SIZE`
[*] <u>spooler</u>	Detect if print spooler is enabled or not
[*] <u>test_connection</u>	Pings a host
[*] <u>tokens</u>	Enumerates available tokens
[*] <u>uac</u>	Checks UAC status
[*] <u>wdigest</u>	Creates/Deletes the 'UseLogonCredential' registry key enabling WDigest cred dumping on Windows >= 8.1
[*] <u>web_delivery</u>	Kicks off a Metasploit Payload using the

exploit/multi/script/web_delivery module

[*] [webdav](#) Checks whether the WebClient service is running on the target

[*] [wireless](#) Get key of all wireless interfaces

CME SSH modules

Here's a list of all CrackMapExec modules that can be used with SSH protocol:

cme ssh -L

[*] [mimipenguin](#) Dumps cleartext credentials in memory

CME WinRM modules

Here's a list of all CrackMapExec modules that can be used with WinRM protocol:

cme winrm -L

As you can see, there are currently no modules at this point.

Conclusion

CrackMapExec is still an actively maintained project with new features and more modules potentially coming in the future. I will do my best to keep this page updated, but if you find something is missing, please don't hesitate to [contact me](#).

If you find this list useful, please consider [subscribing](#) and following InfosecMatter on [Twitter](#), [Facebook](#) or [Github](#) to keep up with the latest developments. You can also [buy me a coffee](#) to support this website.