

### Installation

1) Install Visual Studio C++ build tools (both 64 and 32 bit)

2) Clone the latest Volatility version  
`git clone https://github.com/volatilityfoundation/volatility3.git`

3) As of 02.2024 the plugin yara-python is not yet updated so make sure to delete it from requirements.txt before installing.  
`py -m pip install -r requirements.txt`

4) Download symbol tables and put and extract inside "volatility3\symbols":

Windows

Mac

Linux

5) Start the installation by entering the following commands in this order.

`py setup.py build`

`py setup.py install`

Once the last commands finishes work Volatility will be ready for use.

### OS Information

#Show OS & kernel details of the memory sample being analyzed.

`py vol.py -f "filename" windows.info`

### Hashes

#Dumps user hashes from memory

`py vol.py -f "filename" windows.hashdump.Hashdump`

### Cache

#Dumps lsa secrets from memory

`py vol.py -f "filename" windows.cachedump.Cachedump`

### Environment Variables

#Display process environment variables

`py vol.py -f "filename" windows.envvars.EnvironmentVars`

### Symlinks

#Scans for links present in a particular windows memory image.

`py vol.py -f "filename" windows.symlinks.Scan.SymlinkScan`

### Network

#Scans for network objects present in a particular windows memory image.

`py vol.py -f "filename" windows.netscan`

#Traverses network tracking structures present in a particular windows memory image.

`py vol.py -f "filename" windows.netstat`

### Registry

#Lists the registry hives present in a particular memory image.

`py vol.py -f "filename" windows.registry.HiveList`

#Scans for registry hives present in a particular windows memory image.

`py vol.py -f "filename" windows.registry.HiveScan`

#Lists the registry keys under a hive or specific key value.

`py vol.py -f "filename" windows.registry.PrintKey --key <KEY>`

### Command line arguments

#Lists process command line arguments.

`py vol.py -f "filename" windows.cmdline.CmdLine`

### Services

#Lists process token sids.

`py vol.py -f "filename" windows.getservice-sids.GetServiceSIDs`

### Drivers

#List IRPs for drivers in a particular windows memory image.

`py vol.py -f "filename" windows.driverirp.DriverIrp`

#Scans for drivers present in a particular windows memory image.

`py vol.py -f "filename" windows.driverscan.DriverScan`

### Processes

#Get process list (EPROCESS)

`py vol.py -f "filename" windows.pslist`

#Get hidden process list(malware)

`py vol.py -f "filename" windows.psscanner`

#Get processes tree (not hidden)

`py vol.py -f "filename" windows.pstree`

#Dumps cached file contents from memory samples

`py vol.py -f "filename" -o "output/dir" windows.dumpfiles --pid <PID>`

#Prints the memory map

`py vol.py -f "filename" -o "output/dir" windows.memmap --dump --pid <PID>`

#Lists process open handles.

`py vol.py -f "filename" windows.handles --pid <PID>`

#Lists the loaded modules in a particular windows memory image.

`py vol.py -f "filename" windows.dlllist --pid <PID>`

#Lists process token privileges

`py vol.py -f "filename" windows.privileges.Privs`



By BpDZone  
[cheatography.com/bpdzone/](https://cheatography.com/bpdzone/)

Not published yet.  
 Last updated 7th February, 2024.  
 Page 1 of 2.

Sponsored by [ApolloPad.com](https://apollopad.com)  
 Everyone has a novel in them. Finish Yours!  
<https://apollopad.com>

### Files

#Scans for file objects present in a particular windows memory image.

```
py vol.py -f "filename" windows.filescan
```

#Dumps cached file contents from Windows memory samples.

```
py vol.py -f -o "output/dir" "filename" windows.dumpfiles
```

### Malware General

#Lists process memory ranges that potentially contain injected code.

```
py vol.py -f "filename" windows.malfind.Malfind
```

#Lists the system call table.

```
py vol.py -f "filename" windows.ssdt.SSDT
```



By **BpDZone**  
[cheatography.com/bpdzone/](https://cheatography.com/bpdzone/)

Not published yet.

Last updated 7th February, 2024.

Page 2 of 2.

Sponsored by **ApolloPad.com**

Everyone has a novel in them. Finish

Yours!

<https://apollopad.com>