



# Windows API Exploitation Recipes for Red – Blue Teams

Vivek Ramachandran  
Chief Trainer, Pentester Academy  
<http://PentesterAcademy.com>

# Process Listing APIs: CreateToolhelp32Snapshot

# Tool Help Library: CreateToolhelp32Snapshot

## CreateToolhelp32Snapshot function

Takes a snapshot of the specified processes, as well as the heaps, modules, and threads used by these processes.

### Syntax

C++

```
HANDLE WINAPI CreateToolhelp32Snapshot(  
    _In_ DWORD dwFlags,  
    _In_ DWORD th32ProcessID  
);
```

[https://msdn.microsoft.com/en-us/library/windows/desktop/ms682489\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/ms682489(v=vs.85).aspx)

# TH32CS\_SNAPPROCESS

<b>TH32CS_SNAPPROCESS</b> 0x00000000 2	Includes all processes in the system in the snapshot. To enumerate the processes, see <a href="#">Process32First</a> .
--	--

# Let's get some process data!

## PROCESSENTRY32 structure

Describes an entry from a list of the processes residing in the system address space when a snapshot was taken.

### Syntax

C++

```
typedef struct tagPROCESSENTRY32 {
    DWORD      dwSize;
    DWORD      cntUsage;
    DWORD      th32ProcessID;
    ULONG_PTR  th32DefaultHeapID;
    DWORD      th32ModuleID;
    DWORD      cntThreads;
    DWORD      th32ParentProcessID;
    LONG       pcPriClassBase;
    DWORD      dwFlags;
    TCHAR      szExeFile[MAX_PATH];
} PROCESSENTRY32, *PPROCESSENTRY32;
```

[https://msdn.microsoft.com/en-us/library/windows/desktop/ms684839\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/ms684839(v=vs.85).aspx)

# Note: Can do a LOT More!

Value	Meaning
<b>TH32CS_INHERIT</b> 0x80000000	Indicates that the snapshot handle is to be inheritable.
<b>TH32CS_SNAPALL</b>	Includes all processes and threads in the system, plus the heaps and modules of the process specified in <i>th32ProcessID</i> . Equivalent to specifying the <b>TH32CS_SNAPHEAPLIST</b> , <b>TH32CS_SNAPMODULE</b> , <b>TH32CS_SNAPPROCESS</b> , and <b>TH32CS_SNAPTHREAD</b> values combined using an OR operation (' ').
<b>TH32CS_SNAPHEAPLIST</b> 0x00000001	Includes all heaps of the process specified in <i>th32ProcessID</i> in the snapshot. To enumerate the heaps, see <a href="#">Heap32ListFirst</a> .
<b>TH32CS_SNAPMODULE</b> 0x00000008	Includes all modules of the process specified in <i>th32ProcessID</i> in the snapshot. To enumerate the modules, see <a href="#">Module32First</a> . If the function fails with <b>ERROR_BAD_LENGTH</b> , retry the function until it succeeds.  <b>64-bit Windows:</b> Using this flag in a 32-bit process includes the 32-bit modules of the process specified in <i>th32ProcessID</i> , while using it in a 64-bit process includes the 64-bit modules. To include the 32-bit modules of the process specified in <i>th32ProcessID</i> from a 64-bit process, use the <b>TH32CS_SNAPMODULE32</b> flag.
<b>TH32CS_SNAPMODULE32</b> 0x00000010	Includes all 32-bit modules of the process specified in <i>th32ProcessID</i> in the snapshot when called from a 64-bit process. This flag can be combined with <b>TH32CS_SNAPMODULE</b> or <b>TH32CS_SNAPALL</b> . If the function fails with <b>ERROR_BAD_LENGTH</b> , retry the function until it succeeds.
<b>TH32CS_SNAPPROCESS</b> 0x00000002	Includes all processes in the system in the snapshot. To enumerate the processes, see <a href="#">Process32First</a> .
<b>TH32CS_SNAPTHREAD</b> 0x00000004	Includes all threads in the system in the snapshot. To enumerate the threads, see <a href="#">Thread32First</a> .  To identify the threads that belong to a specific process, compare its process identifier to the <b>th32OwnerProcessID</b> member of the <a href="#">THREADENTRY32</a> structure when enumerating the threads.

# Do we need Privileges?

- For PID, EXE name etc. basic information we do not need elevation or SeDebugPrivilege 😊
- Keeps our call under the radar 😊
  - Check what we can find for all process enumeration APIs we discuss without requiring elevation or SeDebugPrivilege

# Who else uses it? 😊

- Zeus

<http://www.ioactive.com/pdfs/ZeusSpyEyeBankingTrojanAnalysis.pdf>

- PoS Malware

<http://www.trendmicro.co.nz/cloud-content/nz/pdfs/security-intelligence/white-papers/wp-pos-ram-scrapers-malware.pdf>

- ... many many more!

- Used by a lot of system monitoring and performance gathering utilities

- Not considered malicious by AVs

# Advantages

- Quickly figure out what is running
- AV, HIDS etc. security solutions have predictable EXE names
- Can decide if:
  - Run further
  - Which evasion to use?
  - Or maybe even exit?

# Downsides?

- It creates a snapshot!
- While you are analyzing the data:
  - New processes might be created
  - Older processes might be destroyed
  - Parent-Child relationships might change
  - ...
- From a Red Teaming perspective this is generally ok
- Need Kernel Mode to have “live” visibility

# Exercises!

The screenshot shows a web browser window with the URL [https://msdn.microsoft.com/en-us/library/windows/desktop/ms686701\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/ms686701(v=vs.85).aspx). The page header includes the Microsoft logo and navigation links for Technologies, Documentation, and Resources. A search bar for the Dev Center and a sign-in link are also present. The main navigation bar features 'Windows Dev Center' and various categories like 'Windows desktop', 'Get started', 'Design', 'Develop', 'Test & deploy', 'Resources', and 'Dashboards'. The breadcrumb trail indicates the path: 'Diagnostics > Tool Help Library > Using the Tool Help Functions'. The left sidebar contains a list of topics, with 'Taking a Snapshot and Viewing Processes' selected. The main content area has a large heading for the selected topic, followed by an introductory paragraph and a paragraph about error-reporting functions.

Secure [https://msdn.microsoft.com/en-us/library/windows/desktop/ms686701\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/ms686701(v=vs.85).aspx)

Microsoft Technologies Documentation Resources Search Dev Center Sign in

Windows Dev Center Windows desktop Get started Design Develop Test & deploy Resources Dashboard

... > Diagnostics > Tool Help Library > Using the Tool Help Functions

Taking a Snapshot and Viewing Processes

- Traversing the Thread List
- Traversing the Module List
- Traversing the Heap List

## Taking a Snapshot and Viewing Processes

The following simple console application obtains a list of running processes. First, the `GetProcessList` function takes a snapshot of currently executing processes in the system using `CreateToolhelp32Snapshot`, and then it walks through the list recorded in the snapshot using `Process32First` and `Process32Next`. For each process in turn, `GetProcessList` calls the `ListProcessModules` function which is described in [Traversing the Module List](#), and the `ListProcessModules` function which is described in [Traversing the Thread List](#).

A simple error-reporting function, `printError`, displays the reason for any failures, which usually result from security restrictions. For example, `OpenProcess` fails for the Idle and CSRSS processes because their access restrictions prevent user-level code from opening them.

[https://msdn.microsoft.com/en-us/library/windows/desktop/ms686701\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/ms686701(v=vs.85).aspx)

# Thank You!

PentesterAcademy

a SecurityTube.net initiative

Follow @SecurityTube 89.6K followers

Recommend Share 311K

