Attribute Reference Guide

V1.01

ActionName

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Type** | **No.** | **Name** | **Description** | **Relevant Words** |
| **DNS** | 000 | send dns query | Specifies the defined Action of sending a DNS query. | [DNS][query][send] |
| 001 | send reverse dns lookup | Specifies the defined Action of sending a reverse DNS lookup. | [DNS][reverse lookup][send] |
|  |  |  |  |  |
| **Debugging** | 002 | check for kernel debugger | Specifies the defined Action of checking for the presence of a kernel debugger. | [debug][kernel debugger] |
| 003 | check for remote debugger | Specifies the defined Action of checking for the presence of a remote debugger. | [debug][remote debugger] |
|  |  |  |  |  |
| **DeviceDriver** | 004 | emulate driver | Specifies the defined Action of emulating an existing driver on a system. | [driver][emulate] |
| 005 | load and call driver | Specifies the defined Action of loading a driver into a system and then calling the loaded driver. | [driver][load][call] |
| 006 | load driver | Specifies the defined Action of loading a driver into a system. | [driver][load] |
| 007 | unload driver | Specifies the defined Action of unloading a driver from a system. | [driver][unload] |
|  |  |  |  |  |
| **Directory** | 008 | create directory | Specifies the defined Action of creating a new directory on the filesystem. | [directory][create] |
| 009 | delete directory | Specifies the defined Action of deleting an existing directory on the filesystem. | [directory][delete] |
| 010 | hide directory | Specifies the defined Action of hiding an existing directory. | [directory][hide] |
| 011 | monitor directory | Specifies the defined Action of monitoring an existing directory on the filesystem for changes. | [directory][monitor][check] |
|  |  |  |  |  |
| **Disk** | 012 | emulate disk | Specifies the defined Action of emulating an existing disk. | [disk][emulate] |
| 013 | get disk attributes | Specifies the defined Action of querying the attributes of a disk such as the amount of available free space. | [disk][query attribute] |
| 014 | get disk type | Specifies the defined Action of getting the disk type. | [disk][type] |
| 015 | list disks | Specifies the defined Action of listing all disks available on a system. | [disk][list] |
| 016 | monitor disks | Specifies the defined Action of monitoring an existing disk for changes. | [disk][monitor] |
| 017 | mount disk | Specifies the defined Action of mounting an existing file system to a mounting point. | [disk][mount] |
| 018 | unmount disk | Specifies the defined Action of unmounting an existing file system from a mounting point. | [disk][unmount] |
|  |  |  |  |  |
| **FTP** | 019 | connect to ftp server | Specifies the defined Action of connecting to an existing FTP server. | [FTP][connect] |
| 020 | disconnect from ftp server | Specifies the defined Action of disconnecting from an existing FTP server. | [FTP][disconnect] |
| 021 | send ftp command | Specifies the defined Action of sending a command on an FTP server connection. | [FTP][send command] |
|  |  |  |  |  |
| **File** | 022 | close file | Specifies the defined Action of closing an existing file that previously opened for reading or writing. | [file][close] |
| 023 | copy file | Specifies the defined Action of copying an existing file from one location to another. | [file][copy] |
| 024 | create file | Specifies the defined Action of creating a new file. | [file][create] |
| 025 | create file alternate data stream | Specifies the defined Action of creating an alternate data stream in an existing file. | [file][alternate data stream][create] |
| 026 | create file mapping | Specifies the defined Action of creating a new file mapping object. | [file][mapping][create] |
| 027 | create file symbolic link | Specifies the defined Action of creating a symbolic link to an existing file. | [file][symbolic link] |
| 028 | delete file | Specifies the defined Action of deleting an existing file. | [file][delete][wipe] |
| 029 | execute file | Specifies the defined Action of executing an existing file. | [file][execute] |
| 030 | find file | Specifies the defined Action of searching for an existing file. | [file][find][search][check] |
| 031 | get file attributes | Specifies the defined Action of getting the attributes of an existing file. | [file][get attribute] |
| 032 | hide file | Specifies the defined Action of hiding an existing file. | [file][hide] |
| 033 | lock file | Specifies the defined Action of locking an existing file. | [file][lock] |
| 034 | modify file | Specifies the defined Action of modifying an existing file in some manner. | [file][modify][append][overwrite] |
| 035 | move file | Specifies the defined Action of moving an existing file from one location to another. | [file][move] |
| 036 | open file | Specifies the defined Action of opening an existing file for reading or writing. | [file][open] |
| 037 | open file mapping | Specifies the defined Action of opening an existing file mapping object. | [file][mapping][open] |
| 038 | read from file | Specifies the defined Action of reading from an existing file. | [file][read] |
| 039 | rename file | Specifies the defined Action of renaming an existing file. | [file][rename] |
| 040 | send control code to file | Specifies the defined Action of sending a control code to a file. | [file][control code][send] |
| 041 | set file attributes | Specifies the defined Action of setting the file attributes for an existing file. | [file][set attribute] |
| 042 | unlock file | Specifies the defined Action of unlocking an existing file. | [file][unlock] |
| 043 | write to file | Specifies the defined Action of writing to an existing file. | [file][write] |
|  |  |  |  |  |
| **GUI** | 044 | create dialog box | Specifies the defined Action of creating a new dialog box. | [GUI][dialog box][create] |
| 045 | create window | Specifies the defined Action of creating a new window. | [GUI][window][create] |
| 046 | enumerate windows | Specifies the defined Action of enumerating all open windows | [GUI][window][enumerate] |
| 047 | find window | Specifies the defined Action of search for a particular window. | [GUI][window][find][search] |
| 048 | hide window | Specifies the defined Action of hiding an existing window. | [GUI][window][hide] |
| 049 | kill window | Specifies the defined Action of killing an existing window. | [GUI][window][kill] |
| 050 | show window | Specifies the defined Action of showing an existing window. | [GUI][window][show] |
|  |  |  |  |  |
| **HTTP** | 051 | receive http response | Specifies the defined Action of receiving an HTTP server response for a prior HTTP request. | [HTTP][receive][server response] |
| 052 | send http connect request | Specifies the defined Action of sending an HTTP CONNECT client request to an existing server. | [HTTP][CONNECT] |
| 053 | send http delete request | Specifies the defined Action of sending an HTTP DELETE client request to an existing server. | [HTTP][DELETE] |
| 054 | send http get request | Specifies the defined Action of sending an HTTP GET client request to an existing server. | [HTTP][GET] |
| 055 | send http head request | Specifies the defined Action of sending an HTTP HEAD client request to an existing server. | [HTTP][HEAD] |
| 056 | send http options request | Specifies the defined Action of sending an HTTP OPTIONS client request to an existing server. | [HTTP][OPTIONS] |
| 057 | send http patch request | Specifies the defined Action of sending an HTTP PATCH client request to an existing server. | [HTTP][PATCH] |
| 058 | send http post request | Specifies the defined Action of sending an HTTP POST client request to an existing server. | [HTTP][POST] |
| 059 | send http put request | Specifies the defined Action of sending an HTTP PUT client request to an existing server. | [HTTP][PUT] |
| 060 | send http trace request | Specifies the defined Action of sending an HTTP TRACE client request to an existing server. | [HTTP][TRACE] |
|  |  |  |  |  |
| **Hooking** | 061 | add system call hook | Specifies the defined Action of adding a new system call hook. | [hook][system call][add] |
| 062 | add windows hook | Specifies the defined Action of adding a new Windows application-defined hook procedure. | [hook][Windows][add] |
| 063 | hide hook | Specifies the defined action of hiding an existing hook. | [hook][hide] |
|  |  |  |  |  |
| **IPC** | 064 | connect to named pipe | Specifies the defined Action of connecting to an existing named pipe. | [IPC][pipe][connect] |
| 065 | create mailslot | Specifies the defined Action of creating a new named mailslot. | [IPC][mailslot][create] |
| 066 | create named pipe | Specifies the defined Action of creating a new named pipe. | [IPC][pipe][create] |
| 067 | delete named pipe | Specifies the defined Action of deleting an existing named pipe. | [IPC][pipe][delete] |
| 068 | disconnect from named pipe | Specifies the defined Action of disconnecting from an existing named pipe. | [IPC][pipe][disconnect] |
| 069 | read from mailslot | Specifies the defined Action of reading some data from an existing named mailslot. | [IPC][mailslot][read] |
| 070 | read from named pipe | Specifies the defined Action of reading some data from an existing named pipe. | [IPC][pipe][read] |
| 071 | write to mailslot | Specifies the defined Action of writing some data to an existing named mailslot. | [IPC][mailslot][write] |
| 072 | write to named pipe | Specifies the defined Action of writing some data to an existing named pipe. | [IPC][pipe][write] |

1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| **IRC** | 073 | connect to irc server | Specifies the defined Action of connecting to an existing IRC server. | [IRC][connect] |
| 074 | disconnect from irc server | Specifies the defined Action of disconnecting from an existing IRC server. | [IRC][disconnect] |
| 075 | join irc channel | Specifies the defined Action of joining a channel on an IRC server. | [IRC][join] |
| 076 | leave irc channel | Specifies the defined Action of leaving a channel on an IRC server. | [IRC][leave] |
| 077 | receive irc private message | Specifies the defined Action of receiving a private message from another user on an IRC server. | [IRC][private message][receive] |
| 078 | send irc private message | Specifies the defined Action of sending a private message to another user on an IRC server. | [IRC][private message][send] |
| 079 | set irc nickname | Specifies the defined Action of setting an IRC nickname on an IRC server. | [IRC][nickname] |
|  |  |  |  |  |
| **Library** | 080 | call library function | Specifies the defined action of calling a function exported by a library. | [library][call function] |
| 081 | enumerate libraries | Specifies the defined Action of enumerating the libraries used by a process. | [library][enumerate] |
| 082 | free library | Specifies the defined Action of freeing a library previously loaded into the address space of the calling process. | [library][free] |
| 083 | get function address | Specifies the defined Action of getting the address of an exported function or variable from a library. | [library][get function address] |
| 084 | load library | Specifies the defined Action of loading a library into the address space of the calling process. | [library][load] |
|  |  |  |  |  |
| **Network** | 085 | close port | Specifies the defined Action of closing a network port. | [network][port][close] |
| 086 | connect to ip | Specifies the defined Action of connecting to an IP address. | [network][IP][connect] |
| 087 | connect to socket address | Specifies the defined Action of connecting to a socket address consisting of an IP address and port number. | [network][socket address][connect] |
| 088 | connect to url | Specifies the defined Action of connecting to a URL. | [network][URL][connect] |
| 089 | disconnect from ip | Specifies the defined Action of disconnecting from a previously established connection with an IP address. | [network][IP][disconnect] |
| 090 | download file | Specifies the defined Action of downloading a file from a remote location. | [network][file][download] |
| 091 | listen on port | Specifies the defined Action of listening on a specific port. | [network][port][listen][wait] |
| 092 | open port | Specifies the defined Action of opening a network port. | [network][port][open] |
| 093 | receive network packet | Specifies the defined action of receiving a packet on a network. | [network][packet][receive] |
| 094 | send email message | Specifies the defined Action of sending an email message. | [network][email][send] |
| 095 | send icmp request | Specifies the defined Action of sending an ICMP request. | [network][ICMP][send] |
| 096 | send network packet | Specifies the defined action of sending a packet on a network. | [network][packet][send] |
| 097 | upload file | Specifies the defined Action of uploading a file to a remote location. | [network][file][upload] |
|  |  |  |  |  |
| **NetworkShare** | 098 | add connection to network share | Specifies the defined Action of adding a connection to an existing network share. | [network share][add connection] |
| 099 | add network share | Specifies the defined Action of adding a new network share on a server. | [network share][add] |
| 100 | connect to network share | Specifies the defined Action of connecting to an existing network share. | [network share][connect] |
| 101 | delete network share | Specifies the defined Action of deleting an existing network share on a server. | [network share][delete] |
| 102 | disconnect from network share | Specifies the defined Action of disconnecting from an existing network share. | [network share][disconnect] |
| 103 | enumerate network shares | Specifies the defined Action of enumerating the available shared resources on a server. | [network share][enumerate] |
|  |  |  |  |  |
| **Process** | 104 | create process | Specifies the defined Action of creating a new process. | [process][create] |
| 105 | create process as user | Specifies the defined Action of creating a new process in the security context of a specified user. | [process][create as user] |
| 106 | enumerate processes | Specifies the defined Action of enumerating all of the running processes on a system. | [process][enumerate] |
| 107 | flush process instruction cache | Specifies the defined Action of flushing the instruction cache of an existing process. | [process][flush instruction cache] |
| 108 | get process current directory | Specifies the defined Action of getting the current directory of an existing process. | [process][get current directory] |
| 109 | get process environment variable | Specifies the defined Action of getting an environment variable used by an existing process. | [process][get environment variable] |
| 110 | get process startupinfo | Specifies the defined Action of getting the STARTUPINFO struct associated with an existing process. | [process][get STARTUPINFO] |
| 111 | kill process | Specifies the defined Action of killing an existing process. | [process][kill] |
| 112 | open process | Specifies the defined Action of opening an existing process. | [process][open] |
| 113 | set process current directory | Specifies the defined Action of setting the current directory of an existing process. | [process][set current directory] |
| 114 | set process environment variable | Specifies the defined Action of setting an environment variable used by an existing process. | [process][set environment variable] |
| 115 | sleep process | Specifies the defined Action of sleeping an existing process for some period of time. | [process][sleep] |
|  |  |  |  |  |
| **ProcessMemory** | 116 | allocate process virtual memory | Specifies the defined Action of allocating some virtual memory region in an existing process. | [memory][allocate] |
| 117 | free process virtual memory | Specifies the defined Action of freeing some virtual memory region from an existing process. | [memory][free] |
| 118 | map file into process | Specifies the defined Action of mapping an existing file into the address space of the calling process. | [address space][map] |
| 119 | map library into process | Specifies the defined Action of mapping a library into the address space of the calling process. | [address space][map library] |
| 120 | modify process virtual memory protection | Specifies the defined Action of modifying the protection on a memory region in the virtual address space of an existing process. | [memory][modify protection] |
| 121 | read from process memory | Specifies the defined Action of reading from a memory region of an existing process | [memory][read] |
| 122 | unmap file from process | Specifies the defined Action of unmapping an existing file from the address space of the calling process. | [address space][unmap] |
| 123 | write to process memory | Specifies the defined Action of writing to a memory region of an existing process. | [memory][write] |
|  |  |  |  |  |
| **ProcessThread** | 124 | create remote thread in process | Specifies the defined Action of creating a thread that runs in the virtual address space of another existing process. | [thread][remote][create] |
| 125 | create thread | Specifies the defined Action of creating a new thread in the virtual address space of the calling process. | [thread][create] |
| 126 | enumerate threads | Specifies the defined Action of enumerating all threads in the calling process. | [thread][enumerate] |
| 127 | get thread context | Specifies the defined Action of getting the context structure (containing processor-specific register data) of an existing thread. | [thread][get context] |
| 128 | get thread username | Specifies the defined Action of getting the name or ID of the user associated with an existing thread. | [thread][get username] |
| 129 | impersonate process | Specifies the defined Action of a thread in the calling process impersonating the security context of another existing process. | [thread][impersonate] |
| 130 | kill thread | Specifies the defined Action of killing a thread existing in the virtual address space of the calling process. | [thread][kill][terminate] |
| 131 | queue apc in thread | Specifies the defined Action of queuing a new Asynchronized Procedure Call (APC) in the context of an existing thread. | [thread][queue APC] |
| 132 | revert thread to self | Specifies the defined Action of reverting an existing thread to its own security context. | [thread][revert] |
| 133 | set thread context | Specifies the defined Action of setting the context structure (containing processor-specific register data) for an existing thread. | [thread][set context] |
|  |  |  |  |  |
| **Registry** | 134 | close registry key | Specifies the defined Action of closing a handle to an existing registry key. | [registry key][close] |
| 135 | create registry key | Specifies the defined Action of creating a new registry key. | [registry key][create] |
| 136 | create registry key value | Specifies the defined Action of creating a new named value under an existing registry key. | [registry key][key value][create] |
| 137 | delete registry key | Specifies the defined Action of deleting an existing registry key. | [registry key][delete] |
| 138 | delete registry key value | Specifies the defined Action of deleting an existing named value under an existing registry key. | [registry key][key value][delete] |
| 139 | enumerate registry key subkeys | Specifies the defined Action of enumerating the registry key subkeys under an existing registry key. | [registry key][subkey][enumerate] |
| 140 | enumerate registry key values | Specifies the defined Action of enumerating the named values under an existing registry key. | [registry key][key value][enumerate] |
| 141 | get registry key attributes | Specifies the defined Action of getting the attributes of an existing registry key. | [registry key][get attribute] |
| 142 | modify registry key | Specifies the defined Action of modifying an existing registry key. | [registry key][modify] |
| 143 | modify registry key value | Specifies the defined Action of modifying an existing named value of an existing registry key. | [registry key][key value][modify] |
| 144 | monitor registry key | Specifies the defined Action of monitoring an existing registry key for changes. | [registry key][monitor] |
| 145 | open registry key | Specifies the defined Action of opening an existing registry key. | [registry key][open] |
| 146 | read registry key value | Specifies the defined Action of reading an existing named value of an existing registry key. | [registry key][key value][read] |
|  |  |  |  |  |
|  | 147 | create service | Specifies the defined Action of creating a new service. | [service][create] |
| 148 | delete service | Specifies the defined Action of deleting an existing service. | [service][delete] |

2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Service** | 149 | enumerate services | Specifies the defined Action of enumerating a specific set of services on a system. | [service][enumerate] |
| 150 | modify service configuration | Specifies the defined Action of modifying the configuration parameters of an existing service. | [service][configuration][modify] |
| 151 | open service | Specifies the defined Action of opening an existing service. | [service][open] |
| 152 | send control code to service | Specifies the defined Action of sending a control code to an existing service. | [service][control code][send] |
| 153 | start service | Specifies the defined Action of starting an existing service. | [service][start] |
| 154 | stop service | Specifies the defined Action of stopping an existing service. | [service][stop] |
|  |  |  |  |  |
| **Socket** | 155 | accept socket connection | Specifies the defined Action of accepting a socket connection. | [socket connection][accept] |
| 156 | bind address to socket | Specifies the defined Action of binding a socket address to a socket. | [socket][bind address] |
| 157 | close socket | Specifies the defined Action of closing an existing socket. | [socket][close] |
| 158 | connect to socket | Specifies the defined Action of connecting to an existing socket. | [socket][connect] |
| 159 | create socket | Specifies the defined Action of creating a new socket. | [socket][create] |
| 160 | disconnect from socket | Specifies the defined Action of disconnecting from an existing socket. | [socket][disconnect] |
| 161 | get host by address | Specifies the defined Action of getting information on a host from a local or remote host database by its IP address. | [socket][get host][IP address] |
| 162 | get host by name | Specifies the defined Action of getting information on a host from a local or remote host database by its name. | [socket][get host][name] |
| 163 | listen on socket | Specifies the defined Action of listening on an existing socket. | [socket][listen] |
| 164 | receive data on socket | Specifies the defined Action of receiving data on an existing socket. | [socket][data][receive] |
| 165 | send data on socket | Specifies the defined Action of sending data on an existing connected socket. | [socket][data][send] |
| 166 | send data to address on socket | Specifies the defined Action of sending data to a specified IP address on an existing unconnected socket. | [socket][IP address][data][send] |
|  |  |  |  |  |
| **Synchronization** | 167 | create critical section | Specifies the defined Action of creating a new critical section. | [critical section][create] |
| 168 | create event | Specifies the defined Action of creating a new named event object. | [event][create] |
| 169 | create mutex | Specifies the defined Action of creating a new named mutex. | [mutex][create] |
| 170 | create semaphore | Specifies the defined Action of creating a new named semaphore. | [semaphore][create] |
| 171 | delete critical section | Specifies the defined Action of deleting an existing critical section object. | [critical section][delete] |
| 172 | delete event | Specifies the defined Action of deleting an existing named event object. | [event][delete] |
| 173 | delete mutex | Specifies the defined Action of deleting an existing named mutex. | [mutex][delete] |
| 174 | delete semaphore | Specifies the defined Action of deleting an existing named semaphore. | [semaphore][delete] |
| 175 | open critical section | Specifies the defined Action of opening an existing critical section object. | [critical section][open] |
| 176 | open event | Specifies the defined Action of opening an existing named event object. | [event][open] |
| 177 | open mutex | Specifies the defined Action of opening an existing named mutex. | [mutex][open] |
| 178 | open semaphore | Specifies the defined Action of opening an existing named semaphore. | [semaphore][open] |
| 179 | release critical section | Specifies the defined Action of releasing an existing critical section object. | [critical section][release] |
| 180 | release mutex | Specifies the defined Action of releasing ownership of an existing named mutex. | [mutex][release] |
| 181 | release semaphore | Specifies the defined Action of releasing ownership of an existing named semaphore. | [semaphore][release] |
| 182 | reset event | Specifies the defined Action of resetting an existing named event object to the non-signaled state. | [event][reset] |
|  |  |  |  |  |
| **System** | 183 | add scheduled task | Specifies the defined Action of adding a scheduled task to a system. | [system][scheduled task][add] |
| 184 | enumerate system handles | Specifies the defined Action of enumerating all open handles on a system. | [system][enumerate handles] |
| 185 | get elapsed system up time | Specifies the defined Action of getting the elapsed up-time for a system. | [system][get elapsed up-time] |
| 186 | get netbios name | Specifies the defined Action of getting the NetBIOS name of a system. | [system][get NetBIOS name] |
| 187 | get system global flags | Specifies the defined Action of getting the enabled global flags on a system. | [system][get global flags] |
| 188 | get system host name | Specifies the defined Action of getting the host name of a system. | [system][get host name] |
| 189 | get system local time | Specifies the defined Action of getting the local time of a system. | [system][get local time] |
| 190 | get system time | Specifies the defined Action of getting the system time of a system represented in Coordinated Universal Time (UTC). | [system][get system time (UTC)] |
| 191 | get username | Specifies the defined Action of getting the username of the currently logged in user of a system. | [system][get username] |
| 192 | get windows directory | Specifies the defined Action of getting the path to the Windows installation directory on a system. | [system][get Windows installation directory] |
| 193 | get windows system directory | Specifies the defined Action of getting the path to the Windows \System directory on a system. | [system][get Windows System directory] |
| 194 | get windows temporary files directory | Specifies the defined Action of getting the path to the Windows Temporary Files Directory on a system. | [system][get Windows Temporary Files directory] |
| 195 | set netbios name | Specifies the defined Action of setting the NetBIOS name of a system. | [system][set NetBIOS name] |
| 196 | set system global flags | Specifies the defined Action of setting system global flags on a system. | [system][set global flags] |
| 197 | set system host name | Specifies the defined Action of setting the system host name of a system. | [system][set host name] |
| 198 | set system local time | Specifies the defined Action of setting the local time of a system. | [system][set local time] |
| 199 | set system time | Specifies the defined Action of setting the system time for a system represented in Coordinated Universal Time (UTC). | [system][set system time (UTC)] |
| 200 | shutdown system | Specifies the defined Action of shutting down a system. | [system][shut down][restart][reboot] |
| 201 | sleep system | Specifies the defined Action of sleeping a system for some period of time. | [system][sleep] |
|  |  |  |  |  |
| **User** | 202 | add user | Specifies the defined Action of adding a new user. | [user][add] |
| 203 | add user to group | Specifies the defined Action of adding an existing user to an existing group. | [user][add to group] |
| 204 | change password | Specifies the defined Action of changing an existing user's password. | [user][change password] |
| 205 | delete user | Specifies the defined Action of deleting an existing user. | [user][delete] |
| 206 | enumerate users | Specifies the defined Action of enumerating all users. | [user][enumerate] |
| 207 | get user attributes | Specifies the defined Action of getting the attributes of an existing user. | [user][get attribute] |
| 208 | invoke user privilege | Specifies the defined Action of invoking a privilege given to an existing user. | [user][invoke privilege][Administrator access needed] |
| 209 | logon as user | Specifies the defined Action of logging on as a specific user. | [user][log on] |
| 210 | remove user from group | Specifies the defined Action of removing an existing user from existing group. | [user][remove from group] |

3

Capability

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Type** | **No.** | **Name** | **Description** | **Keywords** |
| **MalwareCapability** | 000 | anti-behavioral analysis | Indicates that the malware instance is able to prevent behavioral analysis or make it more difficult. | [anti-behavioral analysis][prevent] |
| 001 | anti-code analysis | Indicates that the malware instance is able to prevent code analysis or make it more difficult. | [anti-code analysis][prevent] |
| 002 | anti-detection | Indicates that the malware instance is able to prevent itself and its components from being detected on a system. | [anti-detection][prevent][evasion][hide][mask] |
| 003 | anti-removal | Indicates that the malware instance is able to prevent itself and its components from being removed from a system. | [anti-removal][prevent] |
| 004 | availability violation | Indicates that the malware instance is able to compromise the availability of a system or some aspect of the system. | [availability violation][compromise] |
| 005 | command and control | Indicates that the malware instance is able to receive and execute remotely submitted commands. | [command and control][C2][C&C][communicate] [contact][connect] |
| 006 | data exfiltration | Indicates that the malware instance is able to exfiltrate stolen data or perform tasks related to the exfiltration of stolen data. | [data exfiltration][extract][pack][upload] |
| 007 | data theft | Indicates that the malware instance is able to steal data from the system on which it executes. This includes data stored in some form e.g. in a file as well as data that may be entered into some application such as a web- browser. | [data theft][steal][harvest] |
| 008 | destruction | Indicates that the malware instance is able to destroy some aspect of a system. | [destruction][wipe][overwrite] |
| 009 | fraud | Indicates that the malware instance is able to defraud a user or a system. | [fraud] |
| 010 | infection/propagation | Indicates that the malware instance is able to propagate through the infection of a machine or is able to infect a file after executing on a system. The malware instance may infect actively (e.g. gain access to a machine directly) or passively (e.g. send malicious email). This Capability does not encompass any aspects of the initial infection that is done independently of the malware instance itself. | [infection][propagation][self-replicate][spread][copy] |
| 011 | integrity violation | Indicates that the malware instance is able to compromise the integrity of a system. | [integrity violation][compromise] |
| 012 | machine access/control | Indicates that the malware instance is able to provide the means to access or control the machine on which it is resident. | [machine access][control][execute][terminate][create] |
| 013 | persistence | Indicates that the malware instance is able to persist and remain on a system regardless of system events. | [persistence][remain] |
| 014 | privilege escalation | Indicates that the malware instance is able to elevate the privileges under which it executes. | [privilege escalation][elevate] |
| 015 | probing | Indicates that the malware instance is able to probe its host system or network environment; most often this is done to support other Capabilities and their Objectives. | [probe][check] |
| 016 | remote machine manipulation | Indicates that the malware instance is able to manipulate or access other remote machines. | [remote machine manipulation][man-in-the-middle] |
| 017 | secondary operation | Indicates that the malware instance is able to achieve secondary objectives in conjunction with or after achieving its primary objectives. | [secondary operation][objective] |
| 018 | security degradation | Indicates that the malware instance is able to bypass or disable security features and/or controls. | [security degradation][bypass][disable][reduce integrity] |
| 019 | spying | Indicates that the malware instance is able to capture information from a system related to user or system activity (e.g. from a system's peripheral devices). | [spy][capture information] |

1

StrategicObjectives

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Type** | **No.** | **Name** | **Description** | **Keywords** |
| **AntiBehavioralAnalysis** | 000 | anti-sandbox | Indicates that the malware instance is able to prevent sandbox-based behavioral analysis or make it more difficult. | [prevent sandbox analysis] |
| 001 | anti-vm | Indicates that the malware instance is able to prevent virtual machine (VM) based behavioral analysis or make it more difficult. | [prevent virtual machine analysis][VM] |
|  |  |  |  |  |
| **AntiCodeAnalysis** | 002 | anti-debugging | Indicates that the malware instance is able to prevent itself from being debugged and/or from being run in a debugger or is able to make debugging more difficult. | [prevent debug][check presence][tracer] |
| 003 | anti-disassembly | Indicates that the malware instance is able to prevent itself from being disassembled or make disassembly more difficult. | [prevent disassemble] |
| 004 | code obfuscation | Indicates that the malware instance is able to obfuscate its code. | [obfuscate code] |
|  |  |  |  |  |
| **AntiDetection** | 005 | anti-memory forensics | Indicates that the malware instance is able to prevent or make memory forensics more difficult | [prevent memory forensics] |
| 006 | hide executing code | Indicates that the malware instance is able to hide its executing code. | [executing code][hide][decoy] |
| 007 | hide malware artifacts | Indicates that the malware instance is able to hide its artifacts. | [artifacts][hide][disguise] |
| 008 | hide non-executing code | Indicates that the malware instance is able to hide its non-executing code. | [non-executing code][hide] |
| 009 | security software evasion | Indicates that the malware instance is able to evade security software (e.g. anti-virus tools). | [security software][evasion] |
| 010 | self-modification | Indicates that the malware instance is able to modify itself. | [modify itself][wipe itself] |
|  |  |  |  |  |
| **AntiRemoval** | 011 | prevent malware artifact access | Indicates that the malware instance is able to prevent its artifacts from being accessed. | [artifacts][access] |
| 012 | prevent malware artifact deletion | Indicates that the malware instance is able to prevent its artifacts from being deleted from a system. | [artifacts][delete] |
|  |  |  |  |  |
| **AvailabilityViolation** | 013 | compromise data availability | Indicates that the malware instance is able to compromise the availability of data on a system. | [data availability][compromise] |
| 014 | compromise system availability | Indicates that the malware instance compromises the availability of the system. | [system availability][compromise] |
| 015 | consume system resources | Indicates that the malware instance is able to consume system resources for its own purposes. | [system resources[consume] |
|  |  |  |  |  |
| **CommandandControl** | 016 | determine c2 server | Indicates that the malware instance is able to identify one or more command and control (C2) servers with which to communicate. | [command and control][C&C][C2 server][determine] [identify] |
| 017 | receive data from c2 server | Indicates that the malware instance is able to control its behavior through some external stimulus (e.g. a remotely submitted command). | [command and control][C&C][C2 server][data] [receive][communicate] |
| 018 | send data to c2 server | Indicates that the malware instance is able to send some data to a command and control server. | [command and control][C&C][C2 server][data][send] [communicate] |
|  |  |  |  |  |
| **DataExfiltration** | 019 | obfuscate data for exfiltration | Indicates that the malware is able to obfuscate data that will be exfiltrated. | [obfuscate][data][encode][encrypt][XORed] |
| 020 | perform data exfiltration | Indicates that the malware instance is able to perform data exfiltration via some physical or virtual means. | [data exfiltration][send] |
| 021 | stage data for exfiltration | Indicates that the malware instance is able to gather and prepare data for exfiltration. | [data][stage][gather][prepare] |
|  |  |  |  |  |
| **DataTheft** | 022 | steal authentication credentials | Indicates that the malware instance is able to steal authentication credentials. | [steal][authentication credentials] |
| 023 | steal stored information | Indicates that the malware instance is able to steal information stored on a system (e.g. files). | [steal][stored information][file] |
| 024 | steal system information | Indicates that the malware instance is able to steal information about a system (e.g. network address data). | [steal][system information] |
| 025 | steal user data | Indicates that the malware instance is able to steal user data (e.g. email). | [steal][user data][email] |
|  |  |  |  |  |
| **Destruction** | 026 | destroy physical entity | Indicates that the malware instance is able to destroy a physical entity. | [destroy][physical entity] |
| 027 | destroy virtual entity | Indicates that the malware instance is able to destroy a virtual entity. | [destroy][virtual entity][wipe] |
|  |  |  |  |  |
| **Fraud** | 028 | perform click fraud | Indicates that the malware instance is able to simulate clicks on website advertisements for the purpose of revenue generation. | [fraud][click][advertisement] |
| 029 | perform premium rate fraud | Indicates that the malware instance is able to send text messages or dial phone numbers that are charged at premium rates. | [fraud][premium rate][text message][phone number] |
|  |  |  |  |  |
| **InfectionPropagation** | 030 | infect file | Indicates that the malware instance is able to infect a file. | [infect file][propagate][spread] |
| 031 | infect remote machine | Indicates that the malware instance is able to self-propagate or infect a machine with malware that is different than itself. | [infect remote machine][propagate][spread][drop] |
| 032 | prevent duplicate infection | Indicates that the malware instance is able to prevent itself from infecting a machine multiple times. | [prevent duplicate infection] |
|  |  |  |  |  |
| **IntegrityViolation** | 033 | annoy user | Indicates that the malware instance is able to annoy the users of a system. | [annoy user] |
| 034 | compromise network operational integrity | Indicate that the malware instance is able to compromise the operational integrity of a network. | [network operational integrity][compromise] |
| 035 | compromise system data integrity | Indicates that the malware instance is able to compromise the integrity of a system's data. | [system data][compromise] |
| 036 | compromise system operational integrity | Indicates that the malware instance is able to compromise the operational integrity of a system. | [system operational integrity][compromise] |
| 037 | compromise user data integrity | Indicates that the malware instance is able to compromise a system's user data. | [user data integrity][compromise] |
|  |  |  |  |  |
| **MachineAccessControl** | 038 | control local machine | Indicates that the malware instance is able to control the machine on which it is resident. Examples of malware with this capability include bots/backdoors/RATs. | [control local machine][backdoor][RAT] |
| 039 | install backdoor | Indicates that the malware instance is able to install a backdoor capable of providing covert remote access to the machine on which it is resident. | [install backdoor][RAT] |
|  |  |  |  |  |
| **Persistence** | 040 | ensure compatibility | Indicates that the malware instance is able to manipulate or modify the system on which it executes to ensure that it is able to continue executing. | [ensure compatibility] |
| 041 | gather information for improvement | Indicates that the malware instance is able to gather information from its environment to make itself less likely to be detected. | [gather information] |
| 042 | persist to continuously execute on system | Indicates that the malware instance is able to continue to execute on a system after significant system events (e.g. after a reboot). | [continue to execute] |
| 043 | persist to re-infect system | Indicates that the malware instance is able to re-infect a system after some of its components have been removed. | [re-infect] |
|  |  |  |  |  |
| **PrivilegeEscalation** | 044 | escalate user privilege | Indicates that the malware instance is able to obtain a higher level of access than intended by the system (also known as vertical privilege escalation). | [escalate user privilege] |
| 045 | impersonate user | Indicates that the malware instance is able to impersonate another user to operate within a different security context (also known as horizontal privilege escalation). | [impersonate user] |
|  |  |  |  |  |
| **Probing** | 046 | probe host configuration | Indicates that the malware instance is able to probe the configuration of the host system on which it executes. | [probe][host configuration] |
| 047 | probe network configuration | Indicates that the malware instance is able to probe the properties of its network environment e.g. to determine whether it funnels traffic through a proxy. | [probe][network configuration][internet connection] |
|  |  |  |  |  |
| **RemoteMachineManipulation** | 048 | access remote machine | Indicates that the malware instance is able to access a remote machine. | [remote machine][access][backdoor] |
| 049 | search for remote machine | Indicates that the malware instance is able to search for remote machines to target. | [remote machine][search] |
|  |  |  |  |  |
| **SecondaryOperation** | 050 | install other components | Indicates that the malware instance is able to install additional components. This encompasses the dropping/downloading of other malicious components such as libraries/other malware/tools. | [install] |
| 051 | lay dormant | Indicates that the malware instance is able to lay dormant on a system for some period of time. | [dormant] |
| 052 | log activity | Indicates that the malware instance is able to log its own activity. | [log activity] |
| 053 | patch operating system file(s) | Indicates that the malware instance is able to patch or modify the critical system files of the operating system under which it executes. | [patch system file][modify] |
| 054 | remove traces of infection | Indicates that the malware instance is able to remove traces of its infection of a system. | [remove traces] |
| 055 | suicide exit | Indicates that the malware instance is able to terminate itself based on some condition or value. | [suicide] |
|  |  |  |  |  |
| **SecurityDegradation** | 056 | degrade security programs | Indicates that the malware instance is able to degrade security programs running on a system either by stopping them from executing or by making changes to their code or configuration parameters. | [degrade security] |
| 057 | disable [host-based or os] access controls | Indicates that the malware instance is able to bypass access control mechanisms designed to prevent unauthorized or unprivileged use or execution of applications or files. | [disable access controls] |
| 058 | disable os security features | Indicates that the malware instance is able to bypass inherent operating system security mechanisms that typically involve elevated privileges. | [disable security][operating system] |

1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 059 | disable server provider security features | Indicates that the malware instance is able to bypass or disable third-party security features that would otherwise identify or notify users of its presence. | [disable security][server provider] |
| 060 | disable system updates | Indicates that the malware instance is able to disable the downloading and installation of system updates. | [disable system updates] |
|  |  |  |  |  |
| **Spying** | 061 | capture system input peripheral data | Indicates that the malware instance is able to capture data from a system's input peripheral devices. | [capture data][input peripheral device] |
| 062 | capture system interface data | Indicates that the malware instance is able to capture data from a system's interfaces. | [capture data][interface] |
| 063 | capture system output peripheral data | Indicates that the malware instance is able to capture data sent to a system's output peripheral devices | [capture data][output peripheral device] |
| 064 | capture system state data | Indicates that the malware instance is able to capture information about a system's state (e.g. from its RAM). | [capture data][state][RAM] |

2

TacticalObjectives

**TacticalObjectives**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Type** | **No.** | **Name** | **Description** | **Keywords** |
| **AntiBehavioralAnalysis** | 000 | detect sandbox environment | Indicates that the malware instance is able to detect whether it is being executed in a sandbox environment. | [detect sandbox] |
| 001 | detect vm environment | Indicates that the malware instance is able to detect whether it is being executed in a virtual machine (VM). | [detect virtual machine][VM] |
| 002 | overload sandbox | Indicates that the malware instance is able to overload a sandbox (e.g. by generating a flood of meaningless behavioral data). | [overload sandbox] |
| 003 | prevent execution in sandbox | Indicates that the malware instance is able to prevent its execution in a sandbox. | [prevent execute][in virtual machine][VM] |
| 004 | prevent execution in vm | Indicates that the malware instance is able to prevent its execution in a virtual machine (VM). | [prevent execute][in sandbox] |
|  |  |  |  |  |
| **AntiCodeAnalysis** | 005 | defeat call graph generation | Indicates that the malware instance is able to defeat accurate call graph generation during disassembly. | [defeat call graph generation] |
| 006 | defeat flow-oriented (recursive traversal)  disassemblers | Indicates that the malware instance is able to defeat its disassembly in a flow-oriented (recursive traversal) disassembler. | [defeat disassemble][flow-oriented] |
| 007 | defeat linear disassemblers | Indicates that the malware instance is able to prevent its disassembly in a linear disassembler. | [defeat disassemble][linear] |
| 008 | detect debugging | Indicates that the malware instance is able to detect its execution in a debugger. | [detect debug] |
| 009 | obfuscate imports | Indicates that the malware instance is able to obfuscate its import table making disassembly more difficult. | [obfuscate][imports] |
| 010 | obfuscate instructions | Indicates that the malware instance obfuscates its instructions. | [obfuscate][instructions] |
| 011 | obfuscate runtime code | Indicates that the malware instance is able to obfuscate its runtime code. | [obfuscate][runtime code] |
| 012 | prevent debugging | Indicates that the malware instance is able to prevent its execution in a debugger. | [prevent execute][in debug] |
| 013 | restructure arrays | Indicates that the malware instance is able to restructure its arrays making disassembly more difficult. | [restructure array] |
| 014 | transform control flow | Indicates that the malware instance is able to transform its control flow. | [transform control flow] |
|  |  |  |  |  |
| **AntiDetection** | 015 | change/add content | Indicates that the malware instance is able to change or add to its content. | [change its content] |
| 016 | encrypt self | Indicates that the malware is able to encrypt itself. | [encrypt][self] |
| 017 | execute before/external to kernel/  hypervisor | Indicates that the malware instance is able to execute some or all of its code before or external to the system's kernel or hypervisor (e.g. through the BIOS). | [execute code before kernel][hypervisor] |
| 018 | execute non-main cpu code | Indicates that the malware instance is able to execute some or all of its code on a secondary non CPU processor (e.g. a GPU). | [execute non-main CPU code] |
| 019 | execute stealthy code | Indicates that the malware instance is able to execute some or all of its code in a hidden manner (e.g. by injecting it into a benign process). | [execute stealthy code] |
| 020 | feed misinformation during physical memory acquisition | Indicates that the malware instance is able to report inaccurate data when the content of physical memory is retrieved. | [feed misinformation][report inaccurate data][during physical memory acquisition] |
| 021 | hide arbitrary virtual memory | Indicates that the malware instance is able to hide arbitrary virtual memory to prevent retrieval. | [hide][arbitrary library] |
| 022 | hide code in file | Indicates that the malware instance is able to hide its code in a file. | [hide][code][decoy] |
| 023 | hide file system artifacts | Indicates that the malware instance is able to hide its file system artifacts. | [hide][file system artifact] |
| 024 | hide kernel modules | Indicates that the malware instance is able to hide its usage of kernel modules. | [hide][kernel] |
| 025 | hide network traffic | Indicates that the malware instance is able to hide its network traffic. | [hide][network traffic] |
| 026 | hide open network ports | Indicates that the malware instance is able to hide its open network ports. | [hide][open network ports] |
| 027 | hide processes | Indicates that the malware instance is able to hide its processes. | [hide][processes] |
| 028 | hide registry artifacts | Indicates that the malware instance is able to hide its Windows registry artifacts. | [hide][registry artifacts] |
| 029 | hide services | Indicates that the malware instance is able to hide any system services it creates or injects itself into. | [hide][service] |
| 030 | hide threads | Indicates that the malware instance is able to hide its threads. | [hide][thread] |
| 031 | hide userspace libraries | Indicates that the malware instance is able to hide its usage of userspace libraries. | [hide][userspace library] |
| 032 | obfuscate artifact properties | Indicates that the malware instance is able to hide the properties of its artifacts (e.g. by altering timestamps). | [obfuscate][artifact properties] |
| 033 | prevent native api hooking | Indicates that the malware instance is able to prevent other software from hooking native APIs. | [prevent native API hooking] |
| 034 | prevent physical memory acquisition | Indicates that the malware instance is able to prevent the contents of a system's physical memory from being retrieved. | [prevent physical memory acquisition] |
|  |  |  |  |  |
| **AntiRemoval** | 035 | prevent api unhooking | Indicates that the malware instance is able to prevent its API hooks from being removed. | [prevent API unhook] |
| 036 | prevent file access | Indicates that the malware instance is able to prevent access to the file system. | [prevent access][file] |
| 037 | prevent file deletion | Indicates that the malware instance is able to prevent its files from being deleted from a system. | [prevent delete][file] |
| 038 | prevent memory access | Indicates that the malware instance is able to prevent access to system memory where it may be storing code or data. | [prevent access][memory] |
| 039 | prevent registry access | Indicates that the malware instance is able to prevent access to the Windows registry. | [prevent access][registry] |
| 040 | prevent registry deletion | Indicates that the malware instance is able to prevent its Windows registry entries from being deleted from a system. | [prevent delete][registry entry] |
|  |  |  |  |  |
| **AvailabilityViolation** | 041 | compromise access to information assets | Indicates that the malware instance is able to prevent data from being accessed (e.g. by encrypting user data on a compromised system). | [prevent access][information assets] |
| 042 | compromise local system availability | Indicates that the malware instance is able to cause the local system to be unavailable. | [compromise local system availability] |
| 043 | crack passwords | Indicates that the malware instance is able to consume system resources for password cracking. | [crack password][guess] |
| 044 | denial of service | Indicates that the malware instance is able to cause a server to be unavailable otherwise known as a denial of service (DOS). | [denial of service][DOS][DDOS] |
| 045 | mine for cryptocurrency | Indicates that the malware instance is able to consume system resources for cryptocurrency mining. | [mine cryptocurrency] |
|  |  |  |  |  |
| **CommandandControl** | 046 | check for payload | Indicates that the malware instance is able to query a command and control server to check whether a new malicious payload is available for download | [check payload] |
| 047 | control malware via remote command | Indicates that the malware instance is able to execute commands issued to it from a remote source such as a command and control server for the purpose of controlling its behavior. | [remote command][command and control][C2][C&C] [communicate][receive][send] |
| 048 | generate c2 domain name(s) | Indicates that the malware instance is able to generate the domain name of the command and control server to which it connects. | [generate C2 domain name][command and control] [C&C] |
| 049 | send heartbeat data | Indicates that the malware instance is able to send heartbeat data to a command and control server indicating that it is still active on the host system and able to communicate. | [heartbeat data][send] |
| 050 | send system information | Indicates that the malware instance is able to send data regarding the system on which it is executing to a command and control server. | [system information][send] |
| 051 | update configuration | Indicates that the malware instance is able to update its configuration using data received from a command and control server. | [update configuration] |
| 052 | validate data | Indicates that the malware instance is able to validate the integrity of the data it receives from a command and control server. | [validate data] |
|  |  |  |  |  |
| **DataExfiltration** | 053 | encrypt data | Indicates that the malware instance is able to encrypt data that will be exfiltrated. | [encrypt][data][encode][XORed] |
| 054 | exfiltrate via covert channel | Indicates that that the malware instance is able to exfiltrate data using a covert channel. | [exfiltrate][via covert channel] |
| 055 | exfiltrate via dumpster dive | Indicates that the malware instance is able to exfiltrate data via dumpster dive (i.e. encoded data printed by malware is viewed as garbage and thrown away to then be physically picked up). | [exfiltrate][via dumpster dive][thrown away][garbage] |
| 056 | exfiltrate via fax | Indicates that the malware instance is able to exfiltrate data using a fax system. | [exfiltrate][via fax] |
| 057 | exfiltrate via network | Indicates that the malware instance is able to exfiltrate data across the network. | [exfiltrate][via network] |
| 058 | exfiltrate via physical media | Indicates that the malware instance is able to exfiltrate data using physical media (e.g. a USB drive). | [exfiltrate][via physical media][USB] |
| 059 | exfiltrate via voip/phone | Indicates that the malware instance is able to exfiltrate data (encoded as audio) using a phone system. | [exfiltrate][via VOIP][phone] |
| 060 | hide data | Indicates that the malware instance is able to hide data that will be exfiltrated in other formats (also known as steganography). | [hide][data] |
| 061 | move data to staging server | Indicates that the malware instance is able to move data to be exfiltrated to a particular server to prepare for exfiltration. | [move data][staging server] |
| 062 | package data | Indicates that the malware instance is able to package data for exfiltration. | [package data] |
|  |  |  |  |  |
| **DataTheft** | 063 | steal browser cache | Indicates that the malware instance is able to steal a user's browser cache | [steal][browser cache] |
| 064 | steal browser history | Indicates that the malware instance is able to steal a user's browser history. | [steal][browser history] |
| 065 | steal contact list data | Indicates cates that the malware instance is able to steal a user's contact list. | [steal][contact list] |
| 066 | steal cookie | Indicates that the malware instance is able to steal cookies. | [steal][cookie] |
| 067 | steal cryptocurrency data | Indicates that the malware instance is able to steal cryptocurrency data (e.g. Bitcoin wallets). | [steal][cryptocurrency data] |
| 068 | steal database content | Indicates that the malware instance is able to steal database content. | [steal][database content] |
| 069 | steal dialed phone numbers | Indicates that the malware instance is able to steal the list of phone numbers that a user has dialed. | [steal][dialed phone number] |
| 070 | steal documents | Indicates that the malware instance is able to steal document files stored on a system. | [steal][documents][file] |
| 071 | steal email data | Indicates that the malware instance is able to steal a user's email data | [steal][email data] |
| 072 | steal images | Indicates that the malware instance is able to steal image files stored on a system. | [steal][image] |
| 073 | steal make/model | Indicates that the malware instance is able to steal the information on the make and/or model of a system. | [steal][system make/model information] |

1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 074 | steal network address | Indicates that the malware instance is able to steal information about the network addresses used by a system. | [steal][network address] |
| 075 | steal open port | Indicates that the malware instance is able to steal information about the open ports on a system. | [steal][open port] |
| 076 | steal password hash | Indicates that the malware instance is able to steal password hashes. | [steal][password hash] |
| 077 | steal pki key | Indicates icates that the malware instance is able to steal one or more public key infrastructure (PKI) keys. | [steal][PKI key][public key infrastructure] |
| 078 | steal pki software certificate | Indicates that the malware instance is able to steal one or more public key infrastructure (PKI) software certificates. | [steal][PKI software certificate][public key infrastructure] |
| 079 | steal referrer urls | Indicates that the malware instance is able to steal HTTP referrer information (URL of the Web page that linked to the resource being requested). | [steal][referrer urls] |
| 080 | steal serial numbers | Indicates that the malware instance is able to steal serial numbers stored on a system. | [steal][serial number] |
| 081 | steal sms database | Indicates that the malware instance is able to steal a user's short message service (SMS) (text messaging) database. | [steal][SMS database] |
| 082 | steal web/network credential | Indicates that the malware instance is able to steal usernames/passwords/other forms of network credentials. | [steal][network credential][username][password] |
|  |  |  |  |  |
| **Destruction** | 083 | destroy firmware | Indicates that the malware instance is able to destroy a system's firmware. | [destroy][firmware] |
| 084 | destroy hardware | Indicates that the malware instance is able to destroy a system's hardware. | [destroy][hardware] |
| 085 | erase data | Indicates that the malware instance is able to destroy data by erasure. | [erase data][wipe] |
|  |  |  |  |  |
| **Fraud** | 086 | access premium service | Indicates that the malware instance is able to access a premium service. | [access premium service] |
|  |  |  |  |  |
| **InfectionPropagation** | 087 | identify file | Indicates that the malware instance is able to identify a file or files on a local removable and/or network drive for infection. | [identify file] |
| 088 | identify target machine(s) | Indicates that the malware instance is able to identify one or more machines to be targeted for infection via some remote means (e.g. via email or the network). | [identify target machine] |
| 089 | inventory victims | Indicates that the malware instance is able to keep an inventory of the victims that it remotely infects. | [victim inventory] |
| 090 | modify file | Indicates that the malware instance is able to modify a file in some other manner than writing code to it such as packing it (in terms of binary executable packing). | [modify file] |
| 091 | perform autonomous remote infection | Indicates that the malware instance is able to infect a remote machine autonomously without the involvement of any end user  (e.g. through the exploitation of a remote procedure call vulnerability). | [autonomous remote infect][drop] |
| 092 | perform social-engineering based remote infection | Indicates that the malware instance is able to infect remote machines via some method that involves social engineering (e.g. sending an email with a malicious attachment). | [social engineering] |
| 093 | write code into file | Indicates that the malware instance is able to write code into a file. | [write code] |
|  |  |  |  |  |
| **IntegrityViolation** | 094 | annoy local system user | Indicates that the malware instance is able to annoy local system users. | [annoy user][local system] |
| 095 | annoy remote user | Indicates that the malware instance is able to annoy a remote user. | [annoy user][remote] |
| 096 | corrupt system data | Indicates that the malware instance is able to corrupt a system's data. | [corrupt][system data][infect][wipe] |
| 097 | corrupt user data | Indicates that the malware instance is able to corrupt a system's user data. | [corrupt][user data][wipe] |
| 098 | intercept/manipulate network traffic | Indicates that the malware is able to intercept and/or manipulate traffic on a network. | [intercept][manipulate][network traffic] |
| 099 | subvert system | Indicates that the malware instance is able to subvert a system to perform beyond its operational boundaries or to perform tasks for which it was not originally intended. | [subvert system] |
|  |  |  |  |  |
| **MachineAccessControl** | 100 | control machine via remote command | Indicates that the malware instance is able to execute commands issued to it from a remote source for the purpose of controlling the machine on which it is resident. | [control machine][via remote command][command and control][C2][C&C] |
|  |  |  |  |  |
| **Persistence** | 101 | drop/retrieve debug log file | Indicates that the malware instance is able to generate and retrieve a log file of errors associated with the malware. | [debug log file][drop][generate][retrieve] |
| 102 | limit application type/version | Indicates that the malware instance is able to limit the type or version of an application that runs on a system in order to ensure that it is able to continue executing. | [limit application type] |
| 103 | persist after os install/reinstall | Indicates that the malware instance is able to continue to execute after the operating system is installed or reinstalled. | [persist after][OS install][reinstall] |
| 104 | persist after system reboot | Indicates that the malware instance is able to continue to execute after a system reboot. | [persist after][system reboot] |
| 105 | persist independent of hard disk/os changes | Indicates that the malware instance is able to continue to execute after changes to the hard disk or the operating system have been made. | [persist][independent of hard disk/OS change] |
| 106 | reinstantiate self after initial detection | Indicates that the malware instance s able to re-establish itself on the system after it is initially detected. | [re-establish][self][after initial detection] |
|  |  |  |  |  |
| **PrivilegeEscalation** | 107 | elevate cpu mode | Indicates that the malware instance is able to elevate the CPU (processor) mode under which it executes. | [elevate CPU mode] |
|  |  |  |  |  |
| **Probing** | 108 | check for firewall | Indicates that the malware instance is able to check whether the network environment in which it executes contains a hardware or software firewall. | [check firewall] |
| 109 | check for internet connectivity | Indicates that the malware instance is able to check whether the network environment in which it executes is connected to the internet. | [check internet connectivity] |
| 110 | check for network drives | Indicates that the malware instance is able to check for network drives that may be present in the network environment. | [check network drive] |
| 111 | check for proxy | Indicates that the malware instance is able to check whether the network environment in which it executes contains a hardware or software proxy. | [check proxy] |
| 112 | check language | Indicates that the malware instance is able to check the language of the host system on which it executes. | [check language] |
| 113 | identify os | Indicates that the malware instance is able to identify the operating system under which it executes. | [identify OS] |
| 114 | inventory system applications | Indicates that the malware instance is able to inventory the applications installed on the system on which it executes. | [list system applications] |
| 115 | map local network | Indicates that the malware instance is able to map the layout of the local network environment in which it executes. | [map local network] |
|  |  |  |  |  |
| **RemoteMachineManipulation** | 116 | compromise remote machine | Indicates that the malware instance is able to gain control of a remote machine through compromise. | [remote machine][compromise][infiltrate network] |
|  |  |  |  |  |
| **SecondaryOperation** | 117 | install legitimate software | Indicates that the malware instance is able to install legitimate software. | [install][legitimate software] |
| 118 | install secondary malware | Indicates that the malware instance is able to install another malware instance. | [install][secondary malware] |
| 119 | install secondary module | Indicates that the malware instance is able to install a secondary module (typically related to itself). | [install][secondary module] |
| 120 | remove self | Indicates that the malware instance is able to remove itself from the system. | [remove][self][wipe] |
| 121 | remove system artifacts | Indicates that the malware instance is able to remove its artifacts from a system. | [remove][system artifact] |
|  |  |  |  |  |
| **SecurityDegradation** | 122 | disable access right checking | Indicates that the malware instance is able to bypass/disable/modify the access tokens or access control lists thereby enabling the malware to read/write/or execute a file with one or more of these controls set. | [disable][access right checking] |
| 123 | disable firewall | Indicates that the malware instance is able to evade or disable the host-based firewall or otherwise prevent the blocking of network communications. | [disable][firewall] |
| 124 | disable kernel patching protection | Indicates that the malware instance is able to bypass or disable PatchGuard; thus it is capable of operating at the same level as the kernel and kernel mode drivers (KMD). | [disable][kernel patching protection][PatchGuard] |
| 125 | disable os security alerts | Indicates that the malware instance is able to evade or disable identification and/or notification of its presence by inherent features of the operating system. | [disable][OS security alert] |
| 126 | disable privilege limiting | Indicates that the malware instance is able to bypass controls that limit the privileges that can be granted to a user or entity. | [disable][privilege limit] |
| 127 | disable system file overwrite protection | Indicates that the malware instance is able to bypass or disable the Windows file protection feature; thus enabling system files to be modified or replaced. | [disable][system file overwrite protection] |
| 128 | disable system service pack/patch installation | Indicates that the malware instance is able to disable the system's ability to install service packs or patches. | [disable][install][service pack][patch] |
| 129 | disable system update services/  daemons | Indicates that the malware instance is able to disable system update services or daemons that may be running on a system. | [disable][system update][daemon] |
| 130 | disable user account control | Indicates that the malware instance is able to bypass or disable user account control (UAC); thus enabling a user to run an application with elevated privileges. | [disable][user account control] |
| 131 | gather security product info | Indicates that the malware instance is able to gather information about the security products installed or running on a system. | [gather security product information] |
| 132 | modify security program configuration | Indicates that the malware instance is able to modify the configuration of one or more security programs running on a system in order to hamper their usefulness and ability to detect the malware instance. | [modify security program configuration] |
| 133 | prevent access to security websites | Indicates that the malware instance is able to prevent access from a system to one or more security vendor or security-related websites. | [prevent access][security website] |
| 134 | prevent security program from running | Indicates that the malware instance is able to prevent one or more security programs from running on a system. | [prevent security run][disable] |
| 135 | remove sms warning messages | Indicates that the malware instance is able to capture the message body of incoming SMS messages and abort the broadcasting of a message that meets a certain criteria. | [remove SMS warning] |
| 136 | stop execution of security program | Indicates that the malware instance is able to stop one or more security programs that may already be executing on a system. | [stop security execute][disable] |
|  |  |  |  |  |
|  | 137 | capture camera input | Indicates that the malware instance is able to capture data from a system's camera. | [capture][camera input] |

2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Spying** | 138 | capture file system | Indicates that the malware instance is able to capture data from a system's file system. | [capture][file system][list file] |
| 139 | capture gps data | Indicates that the malware instance is able to capture system GPS data. | [capture][GPS data] |
| 140 | capture keyboard input | Indicates that the malware instance is able to capture data from a system's keyboard. | [capture][keyboard input][keylogger] |
| 141 | capture microphone input | Indicates that the malware instance is able to capture data from a system's microphone. | [capture][microphone input] |
| 142 | capture mouse input | Indicates that the malware instance is able to capture data from a system's mouse. | [capture][mouse input] |
| 143 | capture printer output | Indicates that the malware instance is able to capture data sent to a system's printer. | [capture][printer output] |
| 144 | capture system memory | Indicates that the malware instance is able to capture data from a system's RAM. | [capture][system memory][RAM] |
| 145 | capture system network traffic | Indicates that the malware instance is able to capture system network traffic. | [capture][network traffic] |
| 146 | capture system screenshot | Indicates that the malware instance is able to capture images of what is currently being displayed on a system's screen either locally or remotely via a remote desktop protocol. | [capture][screenshot] |
| 147 | capture touchscreen input | Indicates that the malware instance is able to capture data from a system's touchscreen. | [capture][touchscreen input] |

3