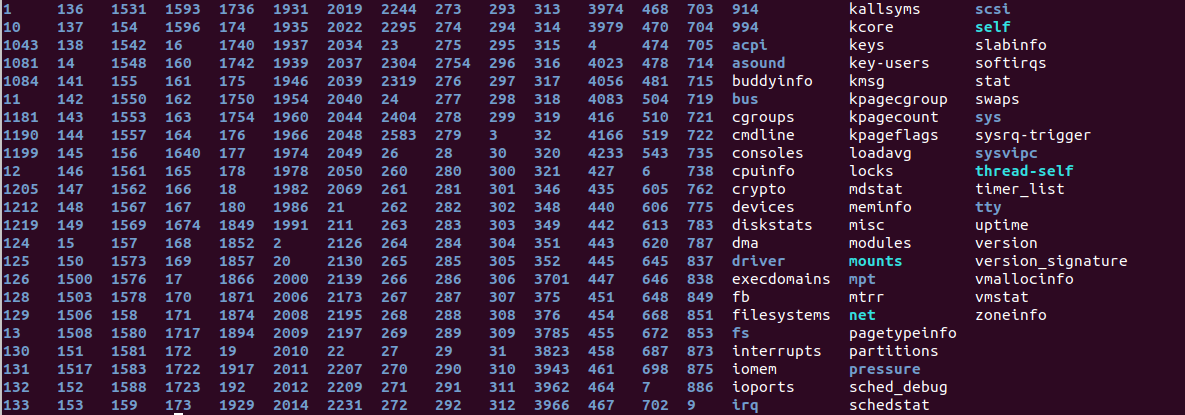
Understanding the Linux Process Folder

The /proc filesystem is a pseudo-filesystem that provides access to the data structures within the kernel.

Pseudo-filesystem in this case means that the folder only exists while the computer is running and is removed once the system is powered off.

A typical view of the folder appears as follows:



Each folder and file can be explained by viewing the following information: ***https://www.tldp.org/LDP/sag/html/proc-fs.html***

The numbers represent the PID of a processes on the system. Those numbers are in fact the folder name related to that process.

Other files that are of interest are the **meminfo**, **cpuinfo**. Take some time to also peruse the other subfolders of the /proc filesystem.

**What is the use of meminfo?**

**What is the cpuinfo used for?**

**Another interesting file is the /proc/sys/kernel/randomize\_va\_space**

**What is the randomize\_va\_space used for?**

Investigate each of the other folders to find other interesting treasures.

The /proc folder has been replaced by the /sys filesystem in newer versions of linux as the de facto directory for system kernel data. For compatibility reasons though the /proc system still exists and provides a wealth of information to the user.

The focus of this lesson are the many numbered folders within this directory. They allow the analyst access, albeit temporarily, to the processes currently executing on the system. For example, by taking a look at the following folder /proc/<PID> you determine a lot about a particular process. PID, is the acronym for **Process Identifier** (PID).

For a detailed explanation of all the features of this folder take a look here:

[**https://www.tldp.org/LDP/Linux-Filesystem-Hierarchy/html/proc.html**](https://www.tldp.org/LDP/Linux-Filesystem-Hierarchy/html/proc.html)

Don’t try to read the whole thing focus on the areas applicable to the assignment.

Other valuable resources that will help to distill the information required for the lab is located at:

[**https://www.kernel.org/doc/Documentation/filesystems/proc.txt**](https://www.kernel.org/doc/Documentation/filesystems/proc.txt)

**https://www.kernel.org/doc/gorman/html/understand/understand007.html**

The information is a bit older (up kernel 2.4.\*) but is still valuable as it relates to this particular lab.

Using the resources provided, find what the terms stands for and write 1 sentence for each of the following:

1. rss (resident set size) How much of RAM memory is currently being occupied by this process.

**https://en.wikipedia.org/wiki/Resident\_set\_size**

1. start\_code (starting address of the code segment)

This is the address of the beginning code segment

1. start\_stack
2. esp
3. eip
4. start\_data
5. start\_brk
6. arg\_start
7. env\_start

/proc/<PID/status table is Table 1-2

/proc/<PID>/stat table is Table 1-4

**Can the tables above be used to determine the size of stack or text?**

From the Red or Blue Team perspective, are the previously mentioned folders of any interest?

Think about how this information can be used by either of those teams.

Can the files or folders be modified be it by a user or root? What does it mean if the files and or folders can be modified?