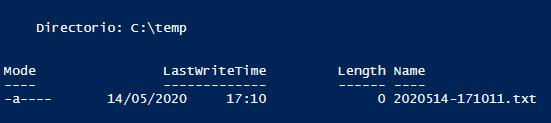
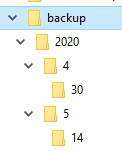
|  |
| --- |
| **Sistemas informáticos.**  **UD 10 – Introducción a scripting.**  **Práctica 10.1 – Ejercicios Powershell.** |

1. Deseamos crear archivos únicos, para ellos escogemos la fecha y hora del sistema y creamos una carpeta en C:\temp con fecha y hora actual. La ejecución del programa el 14 de Abril del 2024(Ejemplo), a las 10 horas, 10 minutos y 11 segundos, daría el siguiente resultado:



Tratar que el mes, el día, la hora, los minutos y los segundos tengan 2 caracteres. Sino puede el archivo 2020514-17428.txt, podría ser a los 4 minutos y 28 segundos o a los 42 minutos y 8 segundos.

1. Programar una tarea programada que todos los días a las 23 horas realice una copia de una carpeta C:\datos a C:\backup. En la carpeta c:\backup se guardan con la siguiente estructura, por año, mes y día.



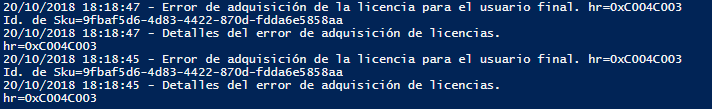
Los meses en lugar de guardarlos en forma numérica, guardarlo en texto, es decir el mes 5 es mayo.

1. Contar cuantos servicios están arrancados y cuantos están parados. La salida sería algo así:

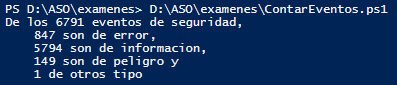


Ayúdate del comando Get-Service.

1. El comando Get-EventLog seguido del tipo, muestra los eventos de esa categoría. Así Get-EventLog Application, muestra los eventos de aplicación. Mostrar los eventos de error de las aplicaciones. Mostrar la fecha y el mensaje del evento. La salida sería algo así.



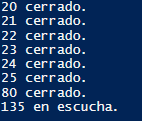
1. De los eventos de sistema, deseamos conocer cuántos hay en total, y cuantos son de cada clase (Error, Information,. Warning y otros). La salida será algo así:



1. Realizar un archivo de Powershell, que muestre información del sistema como tipo y cantidad de ram, procesador, versión del sistema operativo y cualquier tipo de información del sistema que creas importante.

Ayúdate de Get-CimInstance –ClassName

1. Crear un archivo Powershell que comprueba el estado de los puertos 20,21,22,23,24,25,80 y 135, en el equipo local (127.0.0.1). Para ello ayúdate del objeto net.sockets.tcpClient y de las excepciones en Powershell. La salida será algo así:



1. Realice un archivo Powershell, que solicite al usuario un número, y se genere una clave aleatoria, de la longitud introducida. Ejemplo de ejecución:



1. Realice un archivo Powershell, que solicite al usuario un directorio, comprueba que existe ese directorio y si existe mostrar su contenido ordenado por tamaño. Ejemplo de ejecución:

