1. Conceptos básicos

Wednesday, August 12, 2020 7:19 PM

Los script de shell permiten a los profesionales de la Seguridad Informática realizar actividades tales como:

- Encadenar comandos complejos
- Desarrollar herramientas
- Automatizar procesos
- Manipular múltiples archivos

Todo esto solo con utilizar un conjunto de recursos de desarrollo.
Recordemos que como profesionales de seguridad, no siempre tenemos la oportunidad de elegir nuestras herramientas, ya que en ocasiones no podemos instalar aplicaciones o utilerías en un sistema. Para estos casos, el ser capaces de desarrollar herramientas a partir de scripts nativos nos permite realizar nuestras actividades sin inconvenientes.

Según Wikipedia:
shell o **intérprete de órdenes** o **intérprete de comandos**

> Es el programa informático que provee una interfaz de usuario para acceder a los servicios del sistema operativo.

From <https://es.wikipedia.org/wiki/Shell_(inform%C3%A1tica)>

Según la RAE:
interfaz

> Del ingl. *interface* 'superficie de contacto'.
> **1.** f. Conexión o frontera común entre dos aparatos o sistemas independientes.
> **2.** f. *Inform.* Conexión, física o lógica, entre una computadora y el usuario, un dispositivo periférico o un enlace de comunicaciones.

From <https://dle.rae.es/interfaz>

Formalmente, podemos definir el concepto de shell como se muestra a continuación.

☆ **shell**

- También es llamado intérprete de órdenes o comandos
- Espacio de contacto entre el usuario y el sistema operativo
- El mecanismo para gestionar la interacción entre el usuario y el sistema operativo a través de un mecanismo de entrada de datos, interpretación de la entrada y gestión del resultado de salida
- La capa más externa del sistema operativo
- Interfaz a través de la cual, el usuario podrá ejecutar programas, manipular archivos, brindar indicaciones al SO, entre otras acciones

Todo sistema operativo utiliza al menos un tipo de shell, el cual puede ser de diversos tipos:

- **Interfaz gráfica**. Utilizan menús gráficos e íconos para representar archivos y programas, nos permiten interactuar con los elementos a través de clics y se caracterizan por su conveniencia y facilidad (en general) de uso. Algunos de ejemplos de estas shell son el escritorio de Windows, Gnome o KDE.
- **Shells basadas en texto**. Nos permiten comunicarnos con el sistema operativo a través de una variedad de comandos y características integradas en la shell, así como ejecutando otros programas y utilidades. Son "interfaz de usuario ancestral" ancestral y, aún hoy en día, son ampliamente utilizados por la comunidad técnica.

Además de las mencionadas anteriormente, existen las **shell de tipo interfaz natural de usuario** las cuales consisten en formas de interactuar con el sistema sin el uso de comandos. Las interfaces actuales de los sistemas touch se pueden considerar una combinación de interfaz gráfica y de lenguaje natural, ya que es a través de la pantalla capacitiva realizamos la interacción con movimientos naturales como mover, acercar o dar "tap". También se incluyen dentro de esta categoría los sistemas que permiten el uso de lápices ópticos, joysticks y otros aditamentos. Además, el control de sistemas operativos a través de la voz humana es otro claro

ejemplo de shell de interfaz natural de usuario donde Siri, Cortana o Google Now nos facilitan la interacción con el SO.

☆ **línea de mandatos o comandos**
Es la línea en la que escribe.
Contiene el indicador del shell.

El formato básico de cada línea es el siguiente:

```
$ Mandato Argumento(s)
```

El shell considera que la primera palabra de un línea de comandos (hasta el primer espacio en blanco) es el primer comandos y que todas las palabras posteriores son argumentos (como los datos que enviamos a las funciones).

Recordemos que

☆ Un **lenguaje de programación de scripting** es aquel que es interpretado en lugar de ser compilado, lo que implica que la rutina de código resultante si puede ser leída por *humanos* a diferencia de los programas resultantes del proceso de compilación. Entonces, el interprete del lenguaje de programación de scripting lee los comandos del código y hace la conversión a lenguaje máquina mientras el programa es ejecutado, en tiempo real.

☆ **script de shell**
- Es una secuencia de comandos del shell y del sistema operativo que se almacena en un archivo.
- Es un programa, escrito en un lenguaje de scripting, el cual usa la interfaz en algún modo con la shell del sistema operativo que estamos usando.

Los lenguajes de scripting requieren de la instalación del interprete (como Python, Ruby o Perl) para poder ser ejecutados, pero los script de shell no, ya que son interpretados usando la shell misma, la cual ya contiene al interprete presente como parta misma de la shell.
Es posible hacer un script para nuestras interacciones con una shell gráfica gracias a que existen muchos programa y aplicaciones para ello, pero el término *shell script* o *script de shell* se usa comúnmente para referirnos a programas que interactúan con shell basadas en texto.
Es importante destacar que, de manera general, no existen diferencias las herramientas usadas para procesar scripts de shell y las usadas para manejar comandos individuales.
Aunque los lenguajes usado para escribir scripts de shell pudieran no ser clasificados como lenguajes de programación *reales*, estos poseen características muy similares a los lenguajes de programación tradicionales:

- Almacenamiento de datos en variables y otras estructuras de datos
- Crear subrutinas
- Controlar el flujo del programa
- Incluir comentarios

Entre otros. Lo anterior nos permite escribir extensos y complejos programas como scripts de shell.

Ventajas y utilidades del shell scripting

- Rapidez en el desarrollo e implementación. Solo requerimos codificar lo necesario para nuestra tarea, sin necesidad de desarrollar una aplicación completa.
- Permite "aglutinar" diversas herramientas para hacer tareas más complejas. Puede ser un método no muy elegante, pero nos permite trabajar de forma rápida en este encadenamiento de scripts y comandos.
- Todo sistema operativo tiene, al menos una shell basada en texto. Cuando "securizamos" un sistema, en ocasiones removemos todas las aplicaciones no necesarias, razón por la cual podríamos no siempre encontrar un intérprete de Perl o Python en un equipo o servidor, pero la shell del SO siempre estará ahí.
- No requiere compilación. Los scripts de shell son archivos de texto plano, por lo que usar un script puede ser tan simple como crear un archivo y pegar o escribir ahí nuestro código.

Referencias:

- Andress, Jasson and Linn, Ryan (20120) . *Chapter 1 - Introduction to command shell scripting. Coding for Penetration Testers.* "Syngress" https://doi.org/10.1016/B978-1-59749-729-9.00001-1.
- https://www.ibm.com/support/knowledgecenter/es/ssw_aix_71/osmanagement/shells.html?view=embed