

Trabajo práctico evaluado N° 01

Tema: Script para análisis de sitios web

Cada grupo debe elegir a un representante, quién deberá subir las evidencias en la entrega de esta actividad, disponible en la plataforma web. Se deben superar los desafíos de esta actividad sin ningún tipo de error y siguiendo todas las instrucciones para obtener el puntaje máximo.

Instrucciones:

1. Cree un directorio exclusivamente para esta actividad que contenga todos los elementos tales como script, archivos de salida, entre otros.
2. **CARACTERÍSTICAS DEL SCRIPT:**
 - a. Debe utilizar como base todo el código entregado junto con esta actividad.
 - b. Nombre del archivo debe ser: **"análisis_web.sh"**
 - c. Permisos del archivo script: "755" o "rwxr-xr-x"
 - d. Separar todos los datos sensibles (como credenciales, tokens, contraseñas, etc) en un archivo anexo al script:
 - i. Nombre de archivo: **"credenciales.dat"**
 - ii. Permisos del archivo: "600" o "rw-----"
3. **EN EL SCRIPT "ANALISIS_WEB.SH" DEBE:**
 - a. Implementar una función llamada **"validar_comandos"** que verifique si "curl", "jq", "nmap", y "wafw00f" están instalados; si no, pida confirmación para instalarlos y si ya existen, informe al usuario que están instalados.
 - i. Use un arreglo indexado para definir la lista de comandos.
 - ii. Esta función se debe ejecutar siempre al iniciar la ejecución misma del script.
 - iii. Evalúe los comandos utilizando **"if else"** dentro de un ciclo **"for"**.
 - b. Utilice **"case"** para crear un **"menú de opciones"** que muestre un listado ordenado con las opciones.
 - i. Se deben usar números para elegir manualmente la opción deseada.
 - ii. Cada opción será una función del script.
 - iii. Se debe usar **"Enter"** para volver al menú principal.
 - c. Utilizando **"while"**, haga que el menú se vuelva a ejecutar luego de cada operación, hasta que el usuario ingrese la opción para salir.
 - i. Utilice **"break"** para la opción de salida del ciclo **"while"** que utilizó en el menú de opciones.
 - ii. Se debe detectar si se ingresó una opción diferente a las que creó en el menú.
 - d. Redirigir todos los **STDOUT** a **"STDOUT.log"** y **STDERR** a **"STDERR.log"** a nivel de función, utilizando la redirección dentro de la definición de cada función.

- i. También aplicar a todos los mensajes de usuario que estén o haya agregado.
- e. Cree una función llamada **“leer_log_errores”** que lea el registro de STDERR.
 - i. Debe mostrar el registro solo si existe el log, si no existe debe mostrar un mensaje.

4. SUBIR EVIDENCIA:

- a. Captura de la terminal que **muestre la ruta del directorio de trabajo** mediante el uso del comando correspondiente.
- b. Captura de la terminal que **muestre la lista de los archivos de utilizados**, mediante el comando correspondiente que muestre nombre, permisos, y usuario y grupo propietario.
- c. Suba una copia de **todos los archivos utilizados y generados**: analisis_web.sh, credenciales.dat, STDOUT.log, STDERR.log.
 - i. **Comprima directamente en Linux** todos los archivos y luego copie desde Linux a su sistema local el archivo comprimido.
- d. Elimine cualquier credencial, contraseña o token utilizado en el código y reemplácelo por: **“CONFIDENTIAL”**.

5. PRECAUCIONES:

- a. Debe utilizar todas las buenas prácticas para escribir el código del script, como por ejemplo el manejo de arreglos para variables con varios datos, uso de ciclos, uso de condiciones if else, manejo de STDOUT y STDERR, entre otros.
- b. No modifique la variable “sitios”.
- c. Un solo alumno de cada grupo debe subir las evidencias a la plataforma.
- d. Se debe eliminar cualquier código residual (código que no tenga ningún uso).
- e. Se deben escribir comentarios sobre cada función, línea o bloque de código indicando su funcionamiento.

6. PLAZO DE ENTREGA: 1 de julio de 2024 a las 23:59.

- a. Hasta la fecha limite se pueden realizar máximo 2 entregas y solo se evaluará la más reciente.

7. RÚBRICA:

Criterio		Descripción	Puntos
Ejecución Bash	Ejecución del script del script	El script se ejecuta correctamente y sin errores.	5
Directorio y Permisos (3 puntos)	Creación del directorio	El directorio para la actividad fue creado correctamente.	1
	Permisos del script y archivo de credenciales	El script tiene permisos 755 y el archivo de credenciales tiene permisos 600.	2
Estructura del Script (3 puntos)	Nombre del script	El archivo se llama analisis_web.sh.	1
	Separación de credenciales	Credenciales están separadas en credenciales.dat y se leen adecuadamente.	2
Función validar_comandos (4 puntos)	Uso de arreglo indexado	La lista de comandos está definida en un arreglo indexado.	1
	Verificación e instalación	La función verifica y solicita	3

	de comandos	instalación de comandos no encontrados, informando adecuadamente.	
Menú de Opciones (4 puntos)	Uso de case para el menú	El menú está implementado con case y muestra las opciones numeradas.	3
	Retorno al menú principal	Se usa Enter para volver al menú principal después de cada operación.	1
Ciclo while (4 puntos)	Ejecución continua del menú	El menú se ejecuta continuamente hasta que se elige la opción de salida.	3
	Manejo de opciones inválidas	Se detectan y manejan opciones inválidas.	1
Redirección de Salida (4 puntos)	Redirección en funciones	Todas las funciones redirigen STDOUT a STDOUT.log y STDERR a STDERR.log.	3
	Redirección de mensajes de usuario	Mensajes de usuario también están redirigidos a los archivos de log.	1
Función leer_log_errores (3 puntos)	Lectura de STDERR.log	La función lee y muestra el contenido de STDERR.log si existe.	2
	Manejo de ausencia del log	La función muestra un mensaje adecuado si STDERR.log no existe.	1