factum.es

Laboratorio de explotación de vulnerabilidades

Control de versiones (Git, SVN)

FACTUM

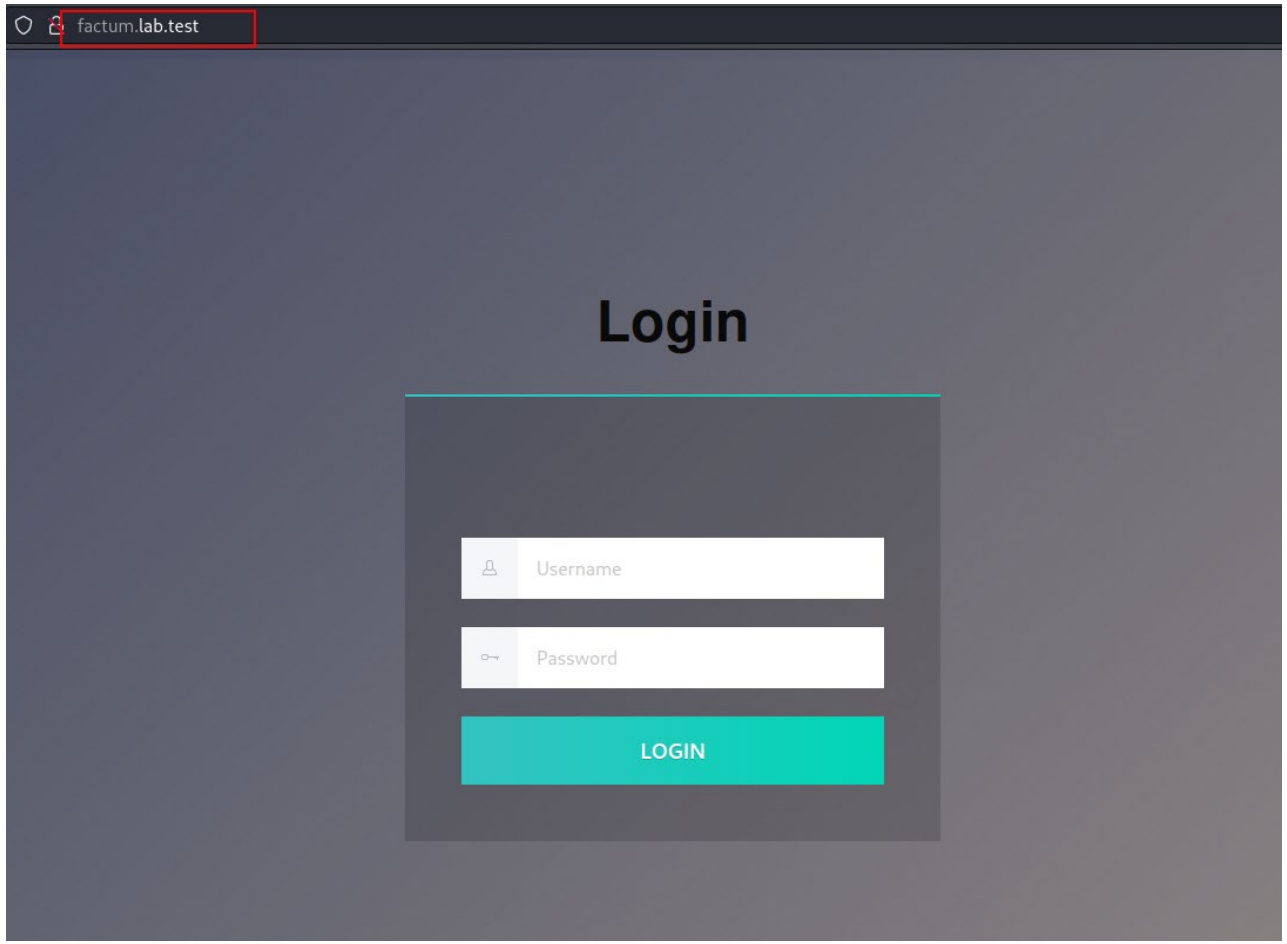
La exposición inadvertida de repositorios de control de versiones (Git, SVN) por parte de una compañía, supone una gran brecha de seguridad que permite a los ciberdelincuentes acceder a sus aplicaciones web.

A continuación, te mostramos paso a paso, cómo puedes comprobar el estado de los repositorios de tu compañía, para verificar si han sido expuestos de manera inadvertida.

EJERCICIO

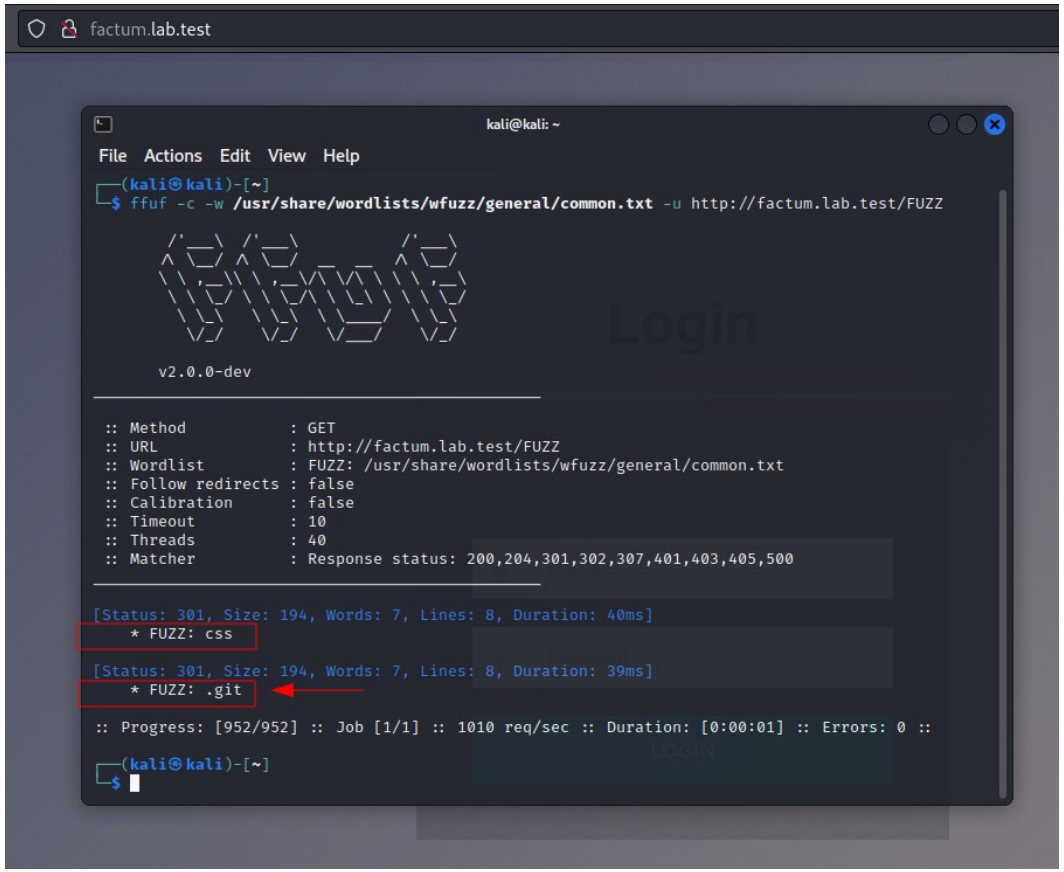
PASO 1

Identificar el sitio web que queremos auditar, y realizar una enumeración que se puede lograr tanto de manera manual, como con herramientas que nos permitan automatizar esta tarea para identificar si el sitio o la aplicación web cuenta con un directorio .git.

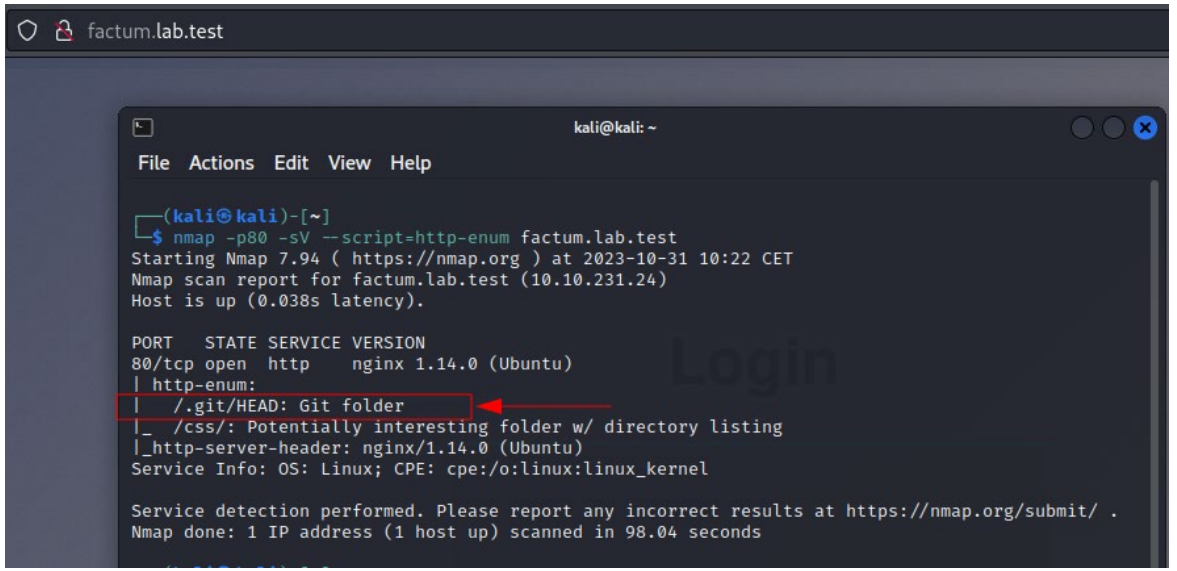


Identificación del sitio web

EJERCICIO



Enumeración



Enumeración con nmap usando script http-enum.

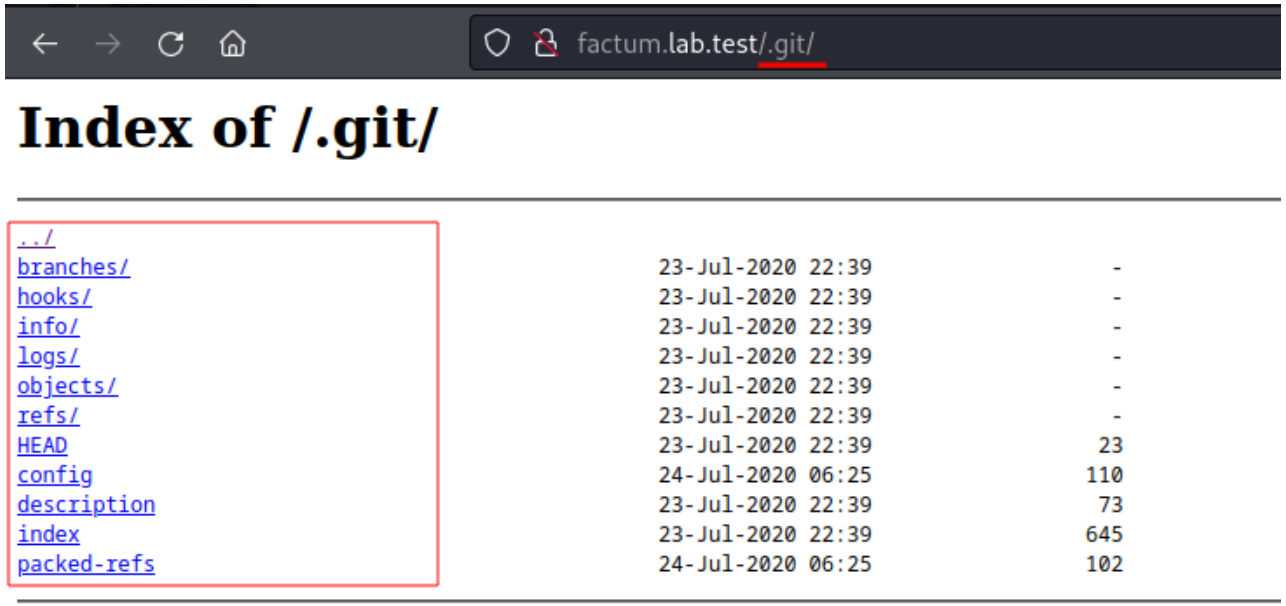
Code: nmap -p80 -sV --script=http-enum factum.lab.test

EJERCICIO

PASO 2

Una vez terminada la enumeración, se encuentra que el script de nmap ha detectado que existe un directorio git expuesto.

Algunos sitios web alojan el control de versiones dentro de su propio servidor, pero limitan el Directory Listing, y al tratar de enumerarlos obtendremos un mensaje de error (forbidden). Sin embargo, algunos atacantes pueden llegar a evadir este tipo de control.



Ejemplo: Exposición de un directorio git.

EJERCICIO

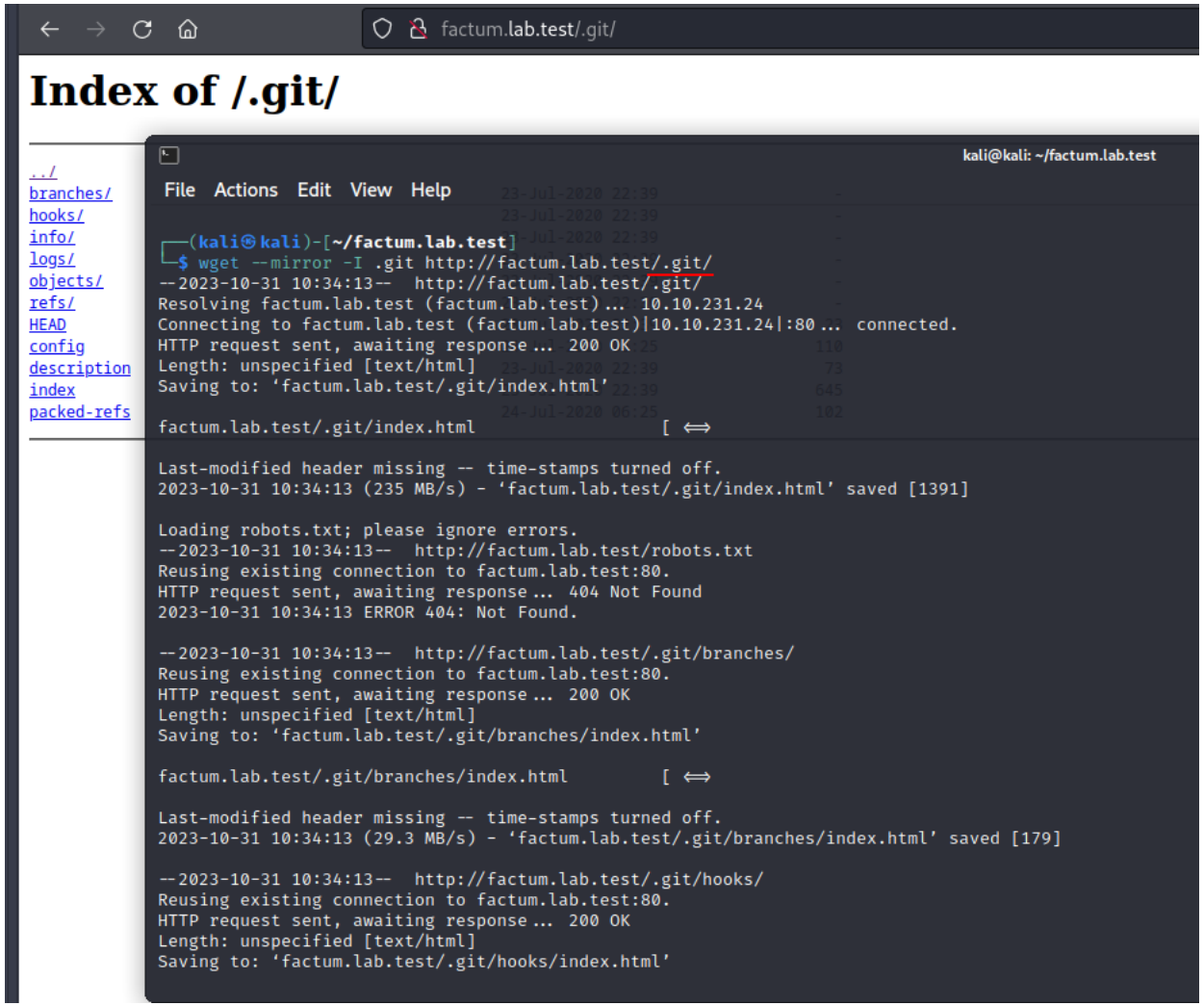
PASO 3

A continuación, podríamos intentar descargar el directorio con wget o usar herramientas que nos permiten reconstruir el git e incluso bypassear si tiene un filtro que no permita hacer abuso de Directory Listing.

code:

wget --mirror -I .git http://factum.lab.test/.git/

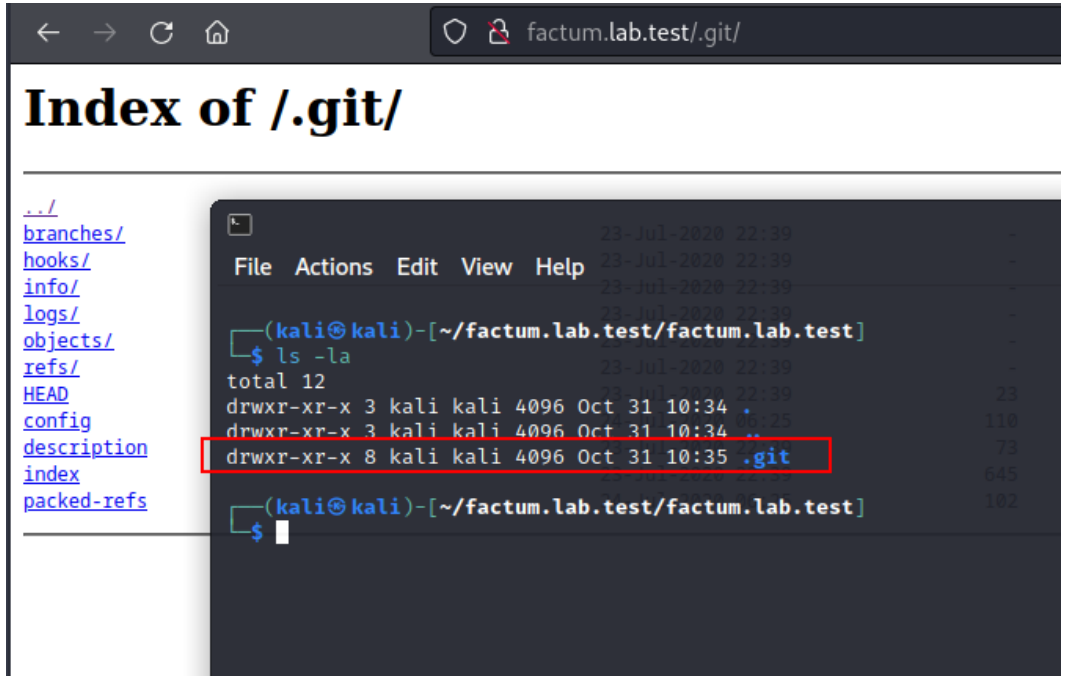
--mirror: Descargará la ruta completa y mantendrá una estructura similar a un espejo en tu sistema local con la rama completa de rutas desde la raíz hasta la hoja junto con todos los datos.

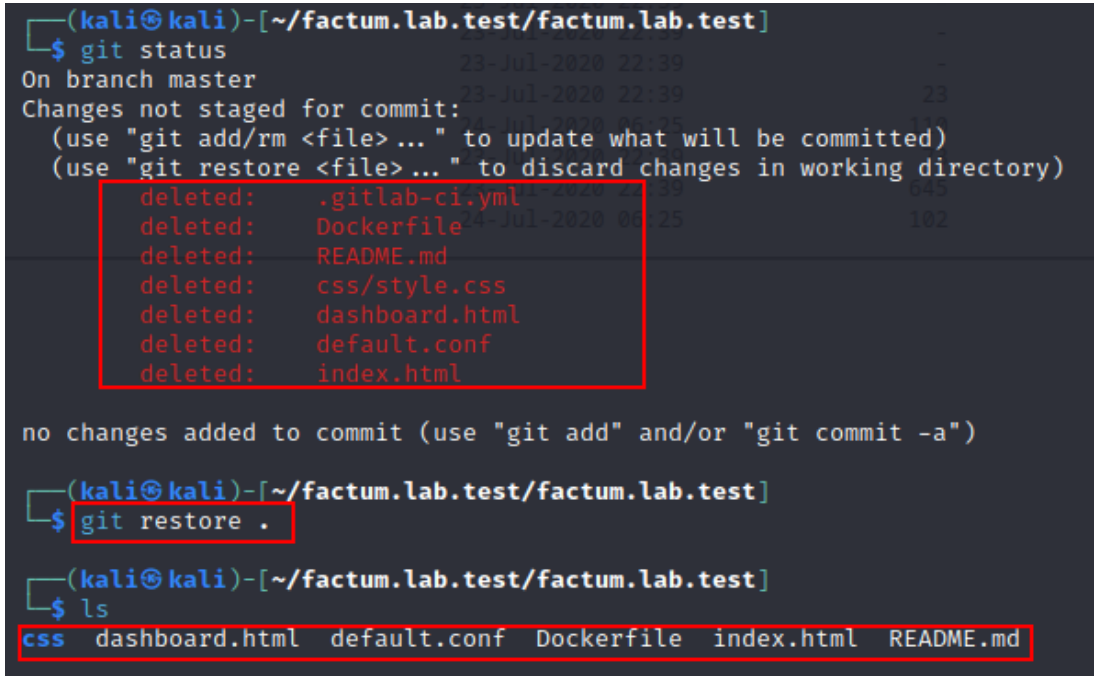


EJERCICIO

PASO 4

Al obtener el git en nuestra carpeta local, ya podríamos intentar analizar la información.





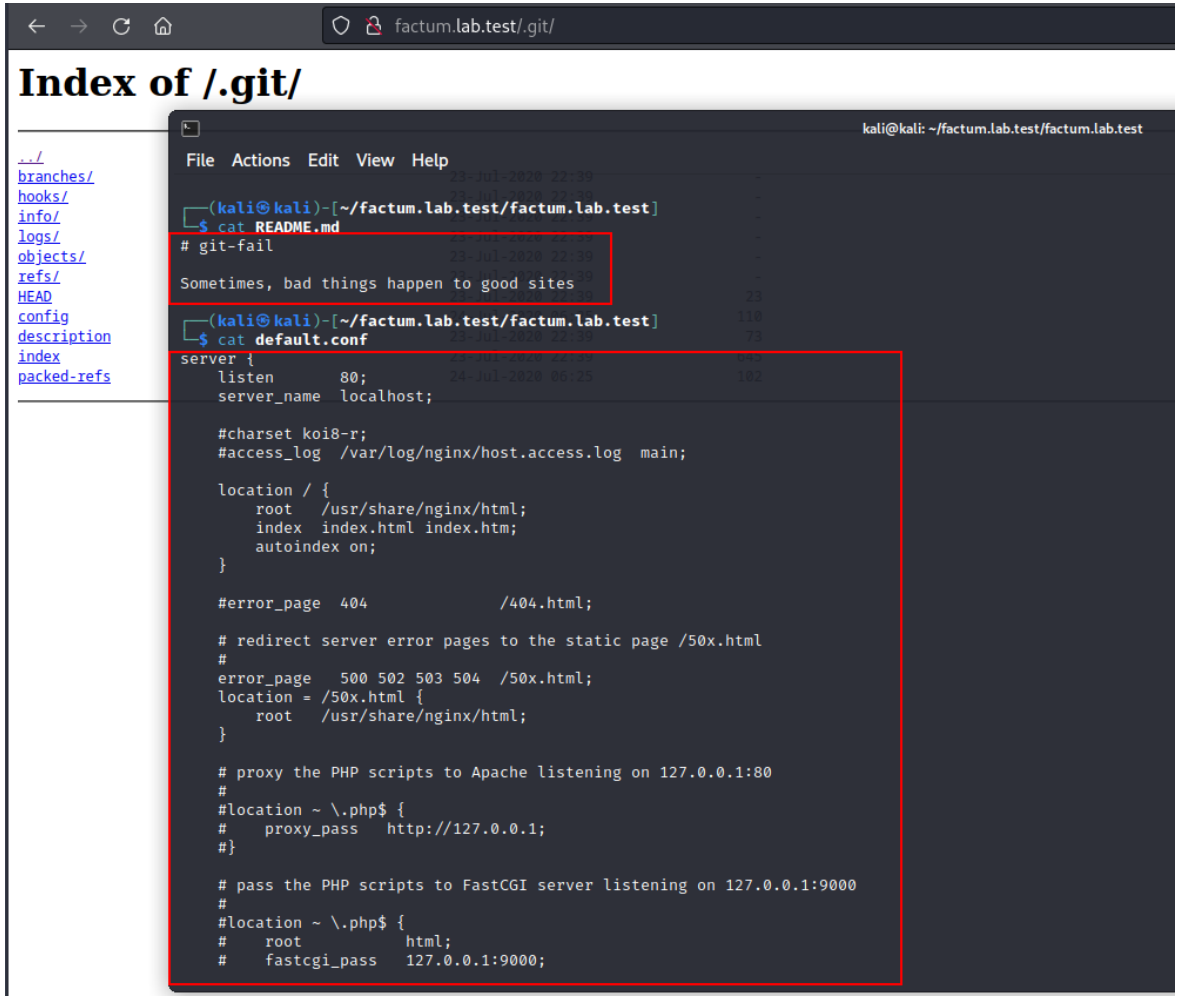
El comando git status muestra el estado del directorio de trabajo.

EJERCICIO

PASO 5

En este punto realizaremos dos acciones:

- restaurar el ultimo commit que se ve en pantalla
- analizar el código fuente restaurado para buscar credenciales, vulnerabilidades de código o extraer propiedad intelectual.

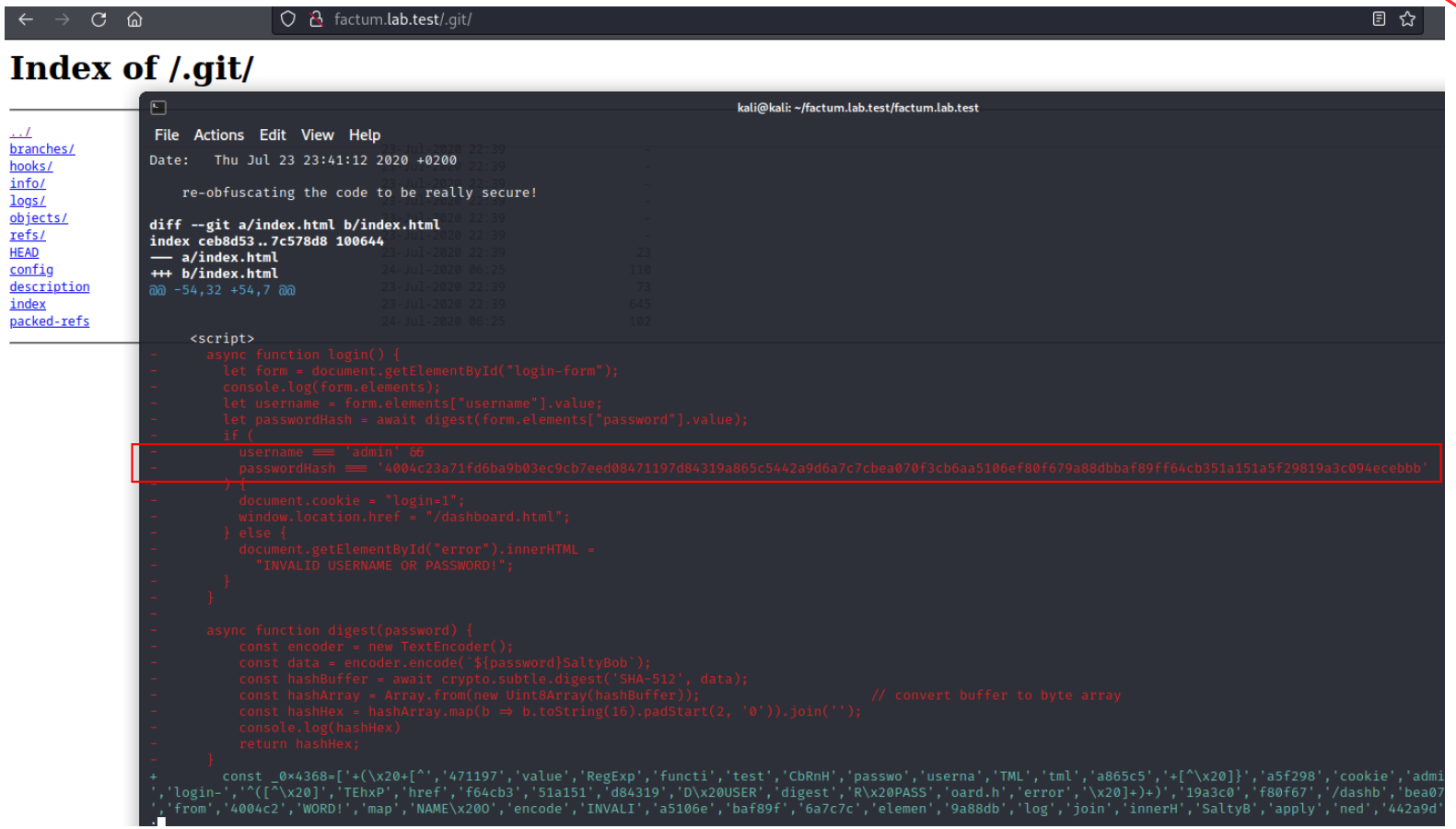


Análisis de la configuración de la aplicación a nivel interno mediante el código fuente exfiltrado.

EJERCICIO

PASO 6

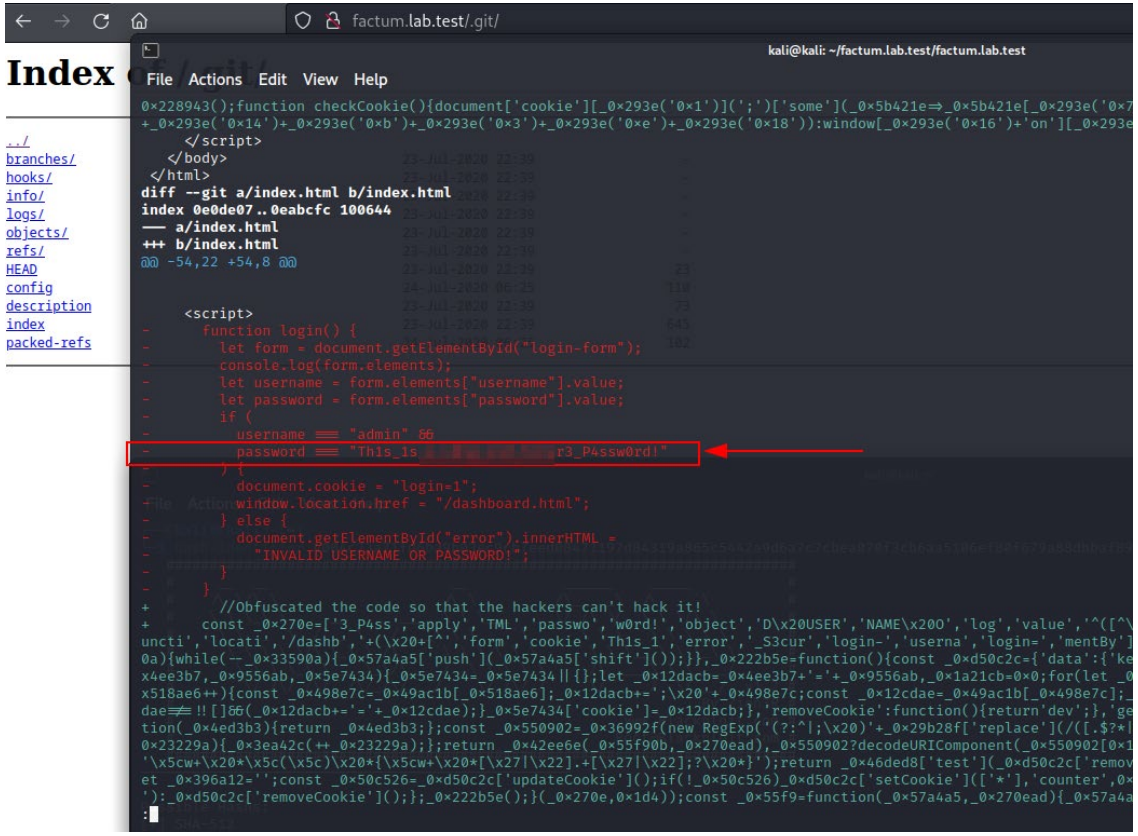
Una vez analizado el código actual nos damos cuenta de que de que no se encuentran secretos, pero podríamos hacer uso de git log que nos permitiría enumerar, filtrar, buscar commits según cambios específicos y obtener diferentes versiones del código que han sido eliminadas.



code: git log –p

EJERCICIO

Se obtiene una versión eliminada del código en el cual se expone un usuario y un hash de una contraseña que podría intentar romper el hash y encontrar la contraseña dependiendo de su complejidad. Sin embargo, si seguimos buscando en los commits, encontraríamos el commit de donde fue borrada la contraseña en texto plano.



De esta manera, aunque las contraseñas e información sensible no estén directamente en el código, podríamos analizar y conseguir exfiltrar la información de las versiones de cambios que se han realizado en el proyecto.

CONCLUSIONES

En el ejemplo anterior se ha realizado de forma manual, pero esto son tareas que en ejercicios de pentesting o en ataques, están automatizadas.

Algunas herramientas que se usan para reconstruir las carpetas .git son:

- https://github.com/captain-noob/GitHacker
- https://github.com/arthaud/git-dumper

Además, algunas herramientas también permiten buscar de manera mucho más optima con expresiones regulares especificas en la versión de cambios del proyecto.

Ejemplo:

code: git log -p | grep -b3 "password"

Incluso el uso de herramientas como **trufflehog** o **gitleaks** disponen de listas específicas de expresiones regulares para buscar todo tipo de información confidencial.

GitHub Dorks

Algunas empresas exponen inadvertidamente sus repositorios de manera pública, no solo en sus propios servidores, si no en servicios de gestión de control de versiones: GitHub, BitBucked, Gitlab, ...

Esto podría hacer posible buscar repositorios git indexados mediante Google Dorks, Git dorks, con el fin de extraer información sensible filtrada mediante estos repositorios o en su historial de versiones.

+34 91 352 44 79
info@factum.es

¿Hablamos?

FACTUM