Vulnerabilidades de control de acceso

Las directivas de control de acceso tienen como objetivo garantizar que los usuarios solo puedan acceder a funcionalidades en las que tengan autorización. Las consecuencia de este tipo de vulnerabilidades son:

- Acceso a cuentas de usuario sin pasar por autenticación
- Acceso a funcionalidades de administrador por parte de usuarios sin permisos
- Acceso a recursos no permitidos.

El control de acceso debe seguir el principio de privilegio menor POLP (Principle Of Least Privilege) que consiste en limitar los permisos de acceso de un usuario a los mínimos imprescindibles para que este pueda realizar su trabajo.

Problemas en el control de acceso

- Modificaciones en las URL cuando referencian las claves primarias de los recursos
- Manipulación de metadatos
- Acceso al sistema de ficheros
- Redirecciones no controladas

Cuando no existe control de acceso o este no es bueno, conociendo la URL, el atacante puede acceder a zonas de la aplicación a las que no debería tener acceso. Por ejemplo, si el usuario sabe la URL del panel de administración, puede acceder a esta solo poniéndola en el navegador. En la actualidad se recomienda separar la zona de administración de la zona de usuario.

Prevención

- El control de acceso debe ser realizado en el backend.
- Por defecto se debe denegar el acceso como política
- Se debe imponer la propiedad “dueño” para que un usuario solo pueda acceder a sus datos.

Atravesar Directorios (Path Traversal)

Es una vulnerabilidad que permite al atacante acceder a ficheros y directorios a los que no debería tener acceso. Se suele explotar mediante el uso de la concatenación de cadenas y el uso del string “../” para conseguir acceder fuera del directorio raiz de la web. Para ello se pueden modificar cookies que apunten a una localización fuera del directorio de aplicación.

```
GET /home.php HTTP/1.0
Cookie:TEMPLATE=../../../../etc/passwd
```

Redirecciones

Cuando el servidor lanza un código 300 el navegador se va a redireccionar. esto se puede usar como un ataque con una redirección incontrolada, que puede mandarnos a una dirección diferente. Por ejemplo, en la página de login podemos hacer que al terminar el logueo nos lleve a otra parte de la web con:

```
http://patatas.es/login?redit=/seccion
```

Siendo sección el sitio al que se puede hacer. Si cambiamos manualmente la cabecera podemos hacer que nos lleve a una parte de la web que queramos, esto también se puede usar como ataque de phising, metiendo una dirección web diferente de forma que cuando el usuario se loguee, le mande a un dominio malicioso. Para prevenir eso se debe verificar que la URL que haya en el redirect sea legítima. Esto se puede combinar con CSRF e inyección de javascript para comprometer al usuario.

Esto mismo se puede aplicar a los fowards, que envían parámetros a una URL de destino.

Recomendaciones par aprevención:

- Evitar redirecciones
- No son evitables, evitar que sean usadas por el usuario
- Para validar las URL usar una lista blanca
- Se recomienda usar una clave almacenada internamente en un mapa
- Utilizar una web intermedia que requiera la confirmación del usuario antes de llevarlo a una URL externa.