Laboratorio 26: Role Based Acess Control (RBAC)

Investiguen y describan 2 sistemas, uno que aplique RBAC y uno que no. Realicen un análisis de las ventajas y desventajas de cada uno con respecto al control de acceso

PREGUNTA

¿En qué consiste el control de acceso basado en roles?

El control de acceso basado en roles (RBAC) es una función de seguridad para controlar el acceso de usuarios a tareas que normalmente están restringidas al superusuario. En otras palabras, consiste en asignar derechos de acceso a los usuarios de su organización en función de sus roles y las tareas que ejecutan. Esto garantiza que los usuarios y equipos solo puedan tener acceso a los niveles que pertenezcan.

SISTEMAS

Sistema con RBAC

Un sistema conocido por muchos que cuenta con RBAC es GitHub. Los usuarios de GitHub cuentan con diferentes privilegios dependiendo de su membresía (prémium o free) así como si es autor de un repositorio, colaborador, usuario externo a un repositorio ya sea autenticado o sin autenticar.

Otro sistema de ejemplo puede ser una página de WordPress donde se tienen diferentes roles, por ejemplo:

- Super-administrador: Cuenta con todos los permisos de acceso y gestión. Puede editar contenido de la web y también gestionar los permisos del resto de usuarios
- Autor: Permiso para editar y publicar artículos propios y acceso a los recursos multimedia de la web, sin embargo, no puede llevar a cabo otras acciones como editar artículos de otros colaboradores.

Ventajas:

- Flexibilidad
- Menor esfuerzo administrativo
- Baja susceptibilidad a errores
- Aumento de la eficiencia
- Seguridad
- Transparencia
- Agregar, eliminar o cambiar roles, e impleméntarlos en todas las llamadas a la API
- Auditar los privilegios de los usuarios y corregir los problemas identificados
- Disminuir el riesgo de filtraciones de datos y la fuga de datos mediante la restricción de acceso a información sensible.

Desventajas:

- Gran esfuerzo en la implementación
- Asignaciones temporales
- Creación y mantenimiento de los roles es más costosa
- Útil a partir de cierto número de funciones y empleados

Sistema sin RBAC

Un sistema bien conocido que no utiliza RBAC es el de las Wikis, ya que las mismas le dan a todos los usuarios la posibilidad de editar el contenido de las mismas sin la necesidad de tener un rol de editor o administrador.

Ventajas:

- Más fácil de programar que un sistema con RBAC.
- Agregar usuarios nuevos es más simple y directo que en sistemas donde se cuenta con roles, ya que los usuarios con algún rol superior al básico no podrán adquirir su rol por sí mismos pues será necesario que alguien haga su alta en el sistema o que se le indique al sistema que los usuarios con ciertos correos recibirán ciertos beneficios.

Desventajas:

- Menos seguridad.
- Pérdida de credibilidad en la información, ya que la misma podría ser agregada o modificada por cualquier persona, incluyendo quienes no tienen conocimientos reales.
- Vulnerable a que personas las editen maliciosamente.
- Dependiendo de cómo estén programadas, podrían ser vulnerables a distintos tipos de ataque que afecten o destruyan la integridad de la base de datos.

Conclusiones:

Michell: El control de acceso basado en roles, si bien es un modelo no siempre conveniente para por ejemplo, empresas pequeñas, me parece que en general se podría considerar como un modelo de mejores prácticas, ya que tiene muchas ventajas, ya que bajamos la susceptibilidad a errores de que numerosos usuarios tengan demasiadas autorizaciones y manipulen funciones que no deberían manipular, por lo tanto, aumenta la eficiencia, la seguridad, etc. Por último, el RBAC nos facilita hacer cambios en la estructura administrativa, ya que los privilegios se transfieren rápidamente a todos los empleados y su rol correspondiente cambia automáticamente sin tener la necesidad de hacer la compleja tarea de asignar privilegios individuales.

David: Si bien es cierto que utilizar RBAC hace que programar la aplicación web requiera trabajo adicional, también es verdad que esto puede ser extremadamente conveniente y, en muchas ocasiones, indispensable en aras de mantener la integridad de la base de datos y la seguridad del sitio web en cuestión, después de todo, hay elementos en ciertas aplicaciones

que podrían ser demasiados sensibles como para permitir que cualquier usuario pueda verlos o manipularlos. RBAC también es útil cuando tenemos usuarios que han de cumplir con tareas diferentes en una misma aplicación web, pues esta tecnología nos permite distribuir funciones entre ellos, lo que resulta conveniente para aplicaciones más grandes. En pocas palabras, aumenta la seguridad y segmenta a los usuarios.

Joseph: El control de acceso basado en roles (RBAC) a pesar de ser una implementación que requiere esfuerzo y costo, considero que beneficia al sistema para evitar modificaciones (creación, modificación y eliminación) de archivos o de cualquier dato dentro de el. Además, reduce los riesgos de robo o espionaje de datos, entre otros. Implementar el control de acceso basado en roles en un sistema principalmente dependerá de las necesidades del mismo y la cantidad de usuarios dentro de este. Por ejemplo, en el caso de nuestro proyecto únicamente se tiene dos roles, ya que la aplicación solo será usada por una cantidad pequeña de usuarios, de los cuales solo pueden tener dos roles: administrador o empleado normal. Por ello, las funcionalidades de nuestra aplicación dependen de los roles. Para implementar el control de acceso basado en roles realizamos una interfaz dinámica que verifica que tipo de rol (administrador o empleado normal) tiene el usuario que se ha autenticado, una vez verificado su rol el sistema ocultará o mostrará las funcionalidades a las que no tiene acceso. Asimismo, se desarrolló un middleware que revisa si el usuario autentificado es administrador y luego se aplicó a las rutas que únicamente debe acceder el administrador, esto para evitar que un empleado normal pueda acceder a las funcionalidades por medio de la url.

REFERENCIAS

Control de acceso basado en roles (descripción general) - Guía de administración del sistema: servicios de seguridad. (2011). Retrieved November 15, 2021, from Oracle.com website: https://docs.oracle.com/cd/E24842_01/html/E23286/rbac-1.html

ManageEngine. (2021). Control de accesos basado en roles | RBAC | Device Control Plus. Retrieved November 15, 2021, from Manageengine.com website: https://www.manageengine.com/latam/device-control/role-based-access-control.html

ciberseg1922. (2021, March 12). Control de acceso basado en roles, ¿qué es y cómo implementarlo? Retrieved November 15, 2021, from Ciberseguridad website: https://ciberseguridad.com/servicios/control-acceso-basado-roles/}

IONOS Digital Guide. (2020, October 14). Role based access control (RBAC): ¿cómo funciona el control de acceso basado en roles? Retrieved November 15, 2021, from IONOS Digitalguide website: https://www.ionos.mx/digitalguide/servidores/seguridad/que-es-el-role-based-access-control-rbac/