Desafío 17 – Miriam Romitelli.

Desarrollo de un Sistema de Gestión de Seguridad de la Información (SGSI) para una organización que aloja aplicaciones en la nube, y aplicar el ciclo PDCA con enfoque en dos aspectos de seguridad (Identificación y Autenticación, y Autorización) en un entorno Linux (sistema operativo + servidor web + base de datos).

Programa completo:

Objeto de estudio: Sistema Operativo Linux (Ubuntu) + Servidor Web (Apache) + Base de Datos (MySQL).

Aspecto de Seguridad: ***Identificación y Autenticación***

Objetivo 1: Mejorar la Gestión de Identidades y Autenticación

Planificar:

1. Identificar todos los sistemas y recursos críticos que requieren autenticación.

Servidor Web (Apache):

- Identificar los sitios web y aplicaciones alojadas en el servidor Apache.
- Determinar si se requiere autenticación para el acceso a la administración del servidor o a las aplicaciones específicas.

Base de Datos (MySQL):

- Listar las bases de datos críticas que almacenan información sensible.
- Definir los usuarios y privilegios necesarios para cada base de datos según los roles de usuario (por ejemplo, administrador, desarrollador, usuario final).

Sistema Operativo (Linux):

- Identificar los usuarios con acceso administrativo (root) y asegurarse de que la autenticación para estos usuarios sea robusta (por ejemplo, mediante MFA si es posible).

2. Definir políticas claras de creación, gestión y eliminación de cuentas de usuario.

Requisitos de Creación:

Establecer criterios claros para la creación de nuevas cuentas de usuario. Por ejemplo, podría requerir que todas las solicitudes de creación de cuentas sean aprobadas por un administrador de sistemas o por un supervisor.

Información: Definir qué información debe proporcionar un usuario al solicitar una cuenta (nombre completo, departamento, motivo de acceso, etc.).

Políticas de Contraseñas: Especificar los requisitos.

Ejemplo Práctico:

Las contraseñas deben tener al menos 10 caracteres de longitud.

Deben incluir al menos un carácter especial y un número.

Las contraseñas deben cambiarse cada 90 días.

No se pueden reutilizar las últimas cinco contraseñas.

Auditorías y Revisiones: Establecer procedimientos regulares para revisar y auditar las cuentas de usuario para garantizar que estén activas y sean necesarias.

Ejemplo Práctico - Suspensión y Reactivación:

Las cuentas de usuario serán suspendidas automáticamente después de 7 días de inactividad.

Las cuentas suspendidas se pueden reactivar dentro de los 30 días posteriores a la suspensión, previa solicitud del supervisor del usuario y aprobación del administrador de sistemas.

Políticas de Eliminación de Cuentas de Usuario

El administrador de sistemas eliminará la cuenta de usuario y revocará todos los accesos y privilegios asociados.

Implementación

Una vez que se haya definido estas políticas, es fundamental comunicarlas claramente a todos los empleados relevantes y asegurarse de que se implementen de manera efectiva y se mantengan actualizadas a medida que evolucionan las necesidades y los riesgos de seguridad de la organización. Regularmente se deberá revisar y actualizar estas políticas para asegurar que sigan siendo adecuadas y efectivas.

3. Evaluar la implementación de autenticación multifactor (MFA) para accesos críticos.

Hacer:

- Implementar un servicio centralizado de autenticación como LDAP o Active Directory.
- Configurar y documentar la integración del servidor web y la base de datos con el servicio de autenticación centralizado.
- Establecer MFA para todas las cuentas de administradores y cuentas con acceso privilegiado.

Check:

- Verificar que todos los usuarios tengan acceso apropiado según sus roles y responsabilidades.
- Revisar la configuración de MFA para asegurar su efectividad y cumplimiento con las políticas establecidas.

Actuar:

- Corregir cualquier desviación encontrada durante las revisiones.
- Mejorar las políticas de gestión de identidades y MFA en base a las lecciones aprendidas y mejores prácticas.

Aspecto de Seguridad: ***Autorización***

Objetivo 2: Reforzar el Control de Acceso y la Autorización

Planificar:

- Definir roles y permisos necesarios para acceder a recursos críticos.
- Implementar un sistema de control de acceso basado en roles (RBAC).

Hacer:

- Configurar listas de control de acceso (ACLs) en el sistema operativo y la base de datos para restringir el acceso a datos sensibles.
- Documentar y automatizar la revisión periódica de los permisos asignados.

Check:

- Realizar auditorías regulares para asegurar que los permisos asignados sigan siendo adecuados.
- Verificar que las ACLs estén correctamente configuradas y que no existan excepciones no autorizadas.

Actuar:

- Ajustar los permisos según cambios organizacionales o en los requisitos de seguridad.
- Implementar mejoras en los procedimientos de revisión y auditoría basados en los resultados obtenidos.

Checklist para Futuros Despliegues

Planificación:

¿Se ha documentado y comunicado claramente la política de gestión de identidades y autenticación?

¿Están definidos los roles y permisos requeridos para todos los recursos críticos?

Implementación (Hacer)

¿Se ha implementado correctamente el sistema de autenticación centralizado y MFA?

¿Están configuradas las ACLs y RBAC según las políticas establecidas?

Verificación (Check):

¿Se han realizado auditorías regulares de identidades y permisos?

¿Se han verificado y corregido las desviaciones encontradas durante las auditorías?

Acción (Actuar):

¿Se han ajustado los permisos y políticas según los cambios organizacionales o de seguridad?

¿Se han implementado mejoras en los procedimientos de revisión y auditoría?

Este programa PDCA asegura que el SGSI en Linux para las aplicaciones en la nube esté robustamente protegido, cumpla con las normativas y estándares de seguridad, y esté preparado para mantener la continuidad del negocio ante cualquier eventualidad.

Desarrollo practico OpenLDAP y MFA en Linux Ubuntu (TEST):

OpenLDAP

sudo apt install slapd ldap-utils

Comprobar el estado del servicio OpenLDAP:

```
ubuntu@ip-10-0-14-133:~$ sudo systemctl status slapd
● slapd.service - LSB: OpenLDAP standalone server (Lightweight Directory Access Protocol)
     Loaded: loaded (/etc/init.d/slapd; generated)
    Drop-In: /usr/lib/systemd/system/slapd.service.d
             └─slapd-remain-after-exit.conf
     Active: active (running) since Sun 2024-07-07 18:04:37 UTC; 30s ago
       Docs: man:systemd-sysv-generator(8)
    Process: 1796 ExecStart=/etc/init.d/slapd start (code=exited, status=0/SUCCESS)
      Tasks: 3 (limit: 1130)
     Memory: 3.3M (peak: 4.2M)
        CPU: 28ms
     CGroup: /system.slice/slapd.service
             └─1808 /usr/sbin/slapd -h "ldap:/// ldapi:///" -g openldap -u openldap -F /etc/ldap/slapd.d

Jul 07 18:04:37 ip-10-0-14-133 systemd[1]: Starting slapd.service - LSB: OpenLDAP standalone server (Lightweight Directory Access Protocol)...
Jul 07 18:04:37 ip-10-0-14-133 slapd[1796]:  * Starting OpenLDAP slapd
Jul 07 18:04:37 ip-10-0-14-133 slapd[1807]: @(#) $OpenLDAP: slapd 2.6.7+dfsg-1~exp1ubuntu8 (Apr  3 2024 18:47:41) $
                                                    Ubuntu Developers <ubuntu-devel-discuss@lists.ubuntu.com>
Jul 07 18:04:37 ip-10-0-14-133 slapd[1808]: slapd starting
Jul 07 18:04:37 ip-10-0-14-133 slapd[1796]:    ...done.
Jul 07 18:04:37 ip-10-0-14-133 systemd[1]: Started slapd.service - LSB: OpenLDAP standalone server (Lightweight Directory Access Protocol).
```

Configuración y creación de roles para usuarios con OpenLDAP:

```
ubuntu@ip-10-0-14-133:~$ sudo slapcat
dn: dc=ec2,dc=internal
objectClass: top
objectClass: dcObject
objectClass: organization
o: ec2.internal
dc: ec2
structuralObjectClass: organization
entryUUID: 4cfc518a-d0da-103e-8697-97da9ff08061
creatorsName: cn=admin,dc=ec2,dc=internal
createTimestamp: 20240707182724Z
entryCSN: 20240707182724.411261Z#000000#000#000000
modifiersName: cn=admin,dc=ec2,dc=internal
modifyTimestamp: 20240707182724Z

dn: ou=admin,dc=ec2,dc=internal
objectClass: top
objectClass: organizationalUnit
ou: unidad
ou: admin
structuralObjectClass: organizationalUnit
entryUUID: a9a68e8c-d0da-103e-9646-65ff00948170
creatorsName: cn=admin,dc=ec2,dc=internal
createTimestamp: 20240707182959Z
entryCSN: 20240707182959.877335Z#000000#000#000000
modifiersName: cn=admin,dc=ec2,dc=internal
modifyTimestamp: 20240707182959Z

dn: cn=grupo,ou=admin,dc=ec2,dc=internal
objectClass: top
objectClass: posixGroup
gidNumber: 10000
cn: grup
cn: grupo
structuralObjectClass: posixGroup
entryUUID: 3ce7b1d0-d0db-103e-9647-65ff00948170
creatorsName: cn=admin,dc=ec2,dc=internal
createTimestamp: 20240707183406Z
entryCSN: 20240707183406.929313Z#000000#000#000000
modifiersName: cn=admin,dc=ec2,dc=internal
modifyTimestamp: 20240707183406Z
```

```
dn: cn=administrador,ou=admin,dc=ec2,dc=internal
objectClass: top
objectClass: posixGroup
gidNumber: 10000
cn: administrador
structuralObjectClass: posixGroup
entryUUID: 20696e2a-d0dd-103e-9648-65ff00948170
creatorsName: cn=admin,dc=ec2,dc=internal
createTimestamp: 20240707184738Z
entryCSN: 20240707184738.119082Z#000000#000#000000
modifiersName: cn=admin,dc=ec2,dc=internal
modifyTimestamp: 20240707184738Z

dn: cn=desarrollador,ou=admin,dc=ec2,dc=internal
objectClass: top
objectClass: posixGroup
gidNumber: 10000
cn: desarrollador
structuralObjectClass: posixGroup
entryUUID: 881d9136-d0dd-103e-9649-65ff00948170
creatorsName: cn=admin,dc=ec2,dc=internal
createTimestamp: 20240707185032Z
entryCSN: 20240707185032.104953Z#000000#000#000000
modifiersName: cn=admin,dc=ec2,dc=internal
modifyTimestamp: 20240707185032Z

dn: cn=usuariofinal,ou=admin,dc=ec2,dc=internal
objectClass: top
objectClass: posixGroup
gidNumber: 10000
cn: usuariofinal
structuralObjectClass: posixGroup
entryUUID: 0f03f2f8-d0de-103e-964a-65ff00948170
creatorsName: cn=admin,dc=ec2,dc=internal
createTimestamp: 20240707185418Z
entryCSN: 20240707185418.429478Z#000000#000#000000
modifiersName: cn=admin,dc=ec2,dc=internal
modifyTimestamp: 20240707185418Z
```

```
dn: uid=ramon,ou=admin,dc=ec2,dc=internal
objectClass: top
objectClass: posixAccount
objectClass: inetOrgPerson
objectClass: person
cn: ramon
uid: jlopez
uid: ramon
ou: admnistrador
uidNumber: 2000
gidNumber: 10000
homeDirectory: /home/ramon
loginShell: /bin/bash
userPassword:: e1NTSEF9eDZVK1JaekwxRXJQUk9MYlRZb3NjUFJNUnpUeHBwT2E=
sn: Lopez
mail: ramon@ec2.com
givenName: ramon
structuralObjectClass: inetOrgPerson
entryUUID: 2be47720-d0df-103e-964b-65ff00948170
creatorsName: cn=admin,dc=ec2,dc=internal
createTimestamp: 20240707190216Z
entryCSN: 20240707190216.373801Z#000000#000#000000
modifiersName: cn=admin,dc=ec2,dc=internal
modifyTimestamp: 20240707190216Z

ubuntu@ip-10-0-14-133:~$ sudo su ramon
$
$
$ ls -l
ls: cannot open directory '.': Permission denied
$
```

Comando usado para agregar grupos.

```
sudo ldapadd -x -D cn=admin,dc=somebooks,dc=local -W -f grp.ldif
```

Comando usado para agregar usuarios

Para evitar que la contraseña del usuario se almacene en texto plano dentro del archivo *ldif*. usaremos el comando **slappasswd** que produce, a partir de la contraseña original, un *hash* utilizando el algoritmo *SHA-1* (aunque podríamos cambiar el algoritmo que se aplique usando el argumento **-h**).

```
sudo slappasswd
```

y luego agregamos el usuario

```
sudo ldapadd -x -D cn=admin,dc=somebooks,dc=local -W -f usr.ldif
```

MFA Google authenticator:

apt-get install libpam-google-authenticator

```
ubuntu@ip-10-0-14-133:~$ sudo adduser test1
info: Adding user `test1' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group `test1' (1002) ...
info: Adding new user `test1' (1002) with group `test1 (1002)' ...
info: Creating home directory `/home/test1' ...
info: Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for test1
Enter the new value, or press ENTER for the default
        Full Name []:
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n] Y
info: Adding new user `test1' to supplemental / extra groups `users' ...
info: Adding user `test1' to group `users' ...
```

```
ubuntu@ip-10-0-14-133:~$ su test1
Password:
test1@ip-10-0-14-133:/home/ubuntu$ google-authenticator
```

```
test1@ip-10-0-14-133:/home/ubuntu$ google-authenticator

Do you want authentication tokens to be time-based (y/n) y
Warning: pasting the following URL into your browser exposes the OTP secret to Google:
  https://www.google.com/chart?chs=200x200&chld=M|0&cht=qr&chl=otpauth://totp/test1@ip-10-0-14-133%3Fsecret%3DQ4PKBTT2TDQKNA2W6FA6JEUKHI%26issuer%3Dip-10-0-14-133
```