

INGENIERÍA EN SISTEMAS COMPUTACIONALES.

SEGURIDAD Y VIRTUALIZACION.



ACTIVIDAD: REPORTE DE PRACTICA 2.

INTEGRANTES DEL EQUIPO:

JEANETTE ARLET SALAZAR NICOLÁS.	21620202
NELSY ORTIZ LÓPEZ.	21620165
MARIBEL LUCERO ZUÑIGA.	21620139

SEMESTRE: SEPTIMO

GRUPO: 7 US

ASESOR: EDWARD OSORIO SALINAS.

TLAXIACO, OAX, A 12 DE SEPTIEMBRE DE 2024.

CONTENIDO.

1. LDAP (PROTOCOLO LIGERO DE ACCESO A DIRECTORIOS):	10
2. RADIUS (SERVICIO DE USUARIO DE AUTENTICACIÓN REMOTA POR MARCACIÓN). 11	
3. TACACS+ (SISTEMA DE CONTROL DE ACCESO CONTROLADOR DE ACCESO TERMINAL PLUS).....	12
4. KERBEROS	13
5. LCA (LISTA DE CONTROL DE ACCESO):.....	13
6. RBAC (CONTROL DE ACCESO BASADO EN ROLES):	14
7. ABAC (CONTROL DE ACCESO BASADO EN ATRIBUTOS):	14
8. PBAC (CONTROL DE ACCESO BASADO EN POLÍTICAS):	14
9. CONCLUSION.	15
10. BIBLIOGRFIA.	15

INDICE DE TABLAS.

Tabla 1: Protocolo Ligero De Acceso A Directorios.....	11
Tabla 3: Servicio De Usuario De Autenticación Remota Por Marcación.	12
Tabla 4: Servicio De Usuario De Autenticación Remota Por Marcación.	13
Tabla 5: Servicio De Usuario De Autenticación Remota Por Marcación.	13

INDICE DE ILUSTRACIONES.

Ilustración 1: Aplicación que permita loguearse con un usuario y contraseña.....	4
Ilustración 2: Formulario de registro con los campos de usuario, contraseña y confirmación de contraseña.	4
Ilustración 3: Aplicación que afirma si la contraseña es segura o no.	5
Ilustración 4: Página de inicio accesible para cualquier usuario.	5
Ilustración 5: Página de perfil accesible si el usuario ha iniciado sesión.	6
Ilustración 6: Página de administración.	6
Ilustración 7: Mecanismo de autorización que permita o deniegue el acceso a ciertas rutas de la aplicación.....	7

Ilustración 8: Mecanismo de autenticación que permita a los usuarios registrarse, iniciar sesión y cerrar sesión.	8
Ilustración 9: Mecanismo para cerrar sesión después de 5 mins.....	9
Ilustración 10: Protocolo Ligero De Acceso A Directorios.....	10
Ilustración 11: Servicio De Usuario De Autenticación Remota Por Marcación.	11

PRÁCTICA 2 - AUTORIZACIÓN Y AUTENTICACIÓN

Objetivo: El objetivo de esta práctica es que el alumno conozca y aplique los conceptos de autorización y autenticación en la seguridad de la información.

Instrucciones

1. **Crea una aplicación [web|móvil|escritorio] que permita loguearse con un usuario y contraseña.**
 - La aplicación debe tener un formulario de inicio de sesión con los campos de usuario y contraseña.

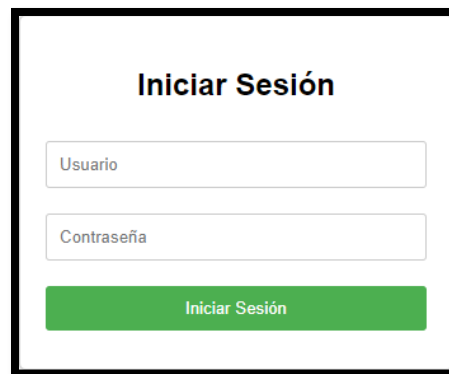


Diagrama de un formulario de inicio de sesión. El formulario tiene un título "Iniciar Sesión" en el centro. Debajo del título hay dos campos de entrada de texto: "Usuario" y "Contraseña". Debajo de estos campos hay un botón verde con el texto "Iniciar Sesión".

Ilustración 1: Aplicación que permita loguearse con un usuario y contraseña.

- La aplicación debe tener un formulario de registro con los campos de usuario, contraseña y confirmación de contraseña.



Diagrama de un formulario de registro. El formulario tiene un título "Formulario de Registro" en el centro. Debajo del título hay tres campos de entrada de texto: "Usuario:", "Contraseña:" y "Confirmar Contraseña:". Debajo de estos campos hay dos botones: "Registrar" (azul) y "Cancelar" (rojo).

Ilustración 2: Formulario de registro con los campos de usuario, contraseña y confirmación de contraseña.

- La aplicación debe decirme si la contraseña es segura o no (extra).

Formulario de Registro

Usuario:

Contraseña:

Confirmar Contraseña:

La contraseña no es segura

[Registrar](#) [Cancelar](#)

Formulario de Registro

Usuario:

Contraseña:

Confirmar Contraseña:

La contraseña es segura

[Registrar](#) [Cancelar](#)

Ilustración 3: Aplicación que afirma si la contraseña es segura o no.

- La aplicación debe tener una página de inicio que sea accesible para cualquier usuario.

Bienvenido



Punto de Venta SOL

[Registrar](#) [Acceder](#)

Ilustración 4: Página de inicio accesible para cualquier usuario.

- La aplicación debe tener una página de perfil que solo sea accesible si el usuario ha iniciado sesión.

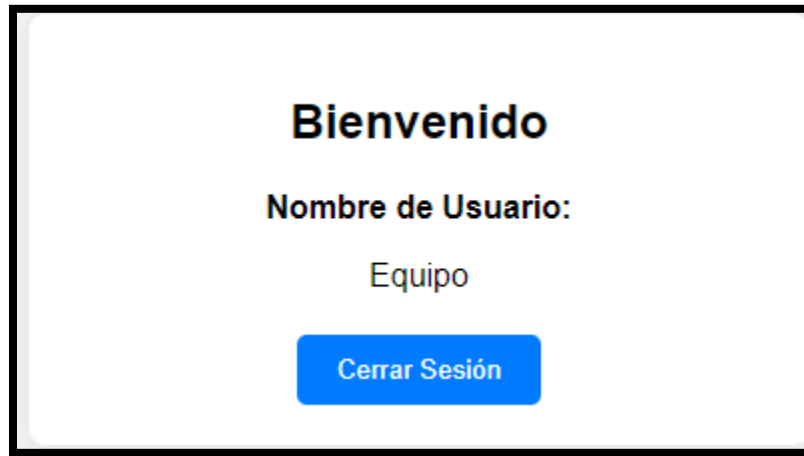


Ilustración 5: Página de perfil accesible si el usuario ha iniciado sesión.

- La aplicación debe tener una página de administración que solo sea accesible si el usuario ha iniciado sesión y tiene un rol de administrador.

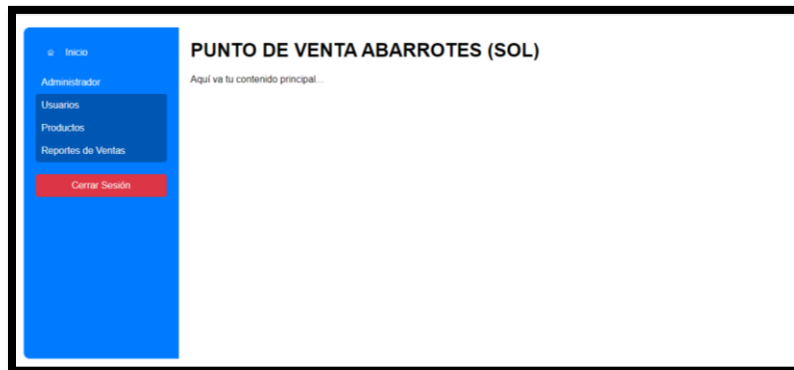


Ilustración 6: Página de administración.

2. **Implementa un mecanismo de autorización que permite o deniegue el acceso a ciertas rutas de la aplicación, en este caso la página de perfil y la página de administración si en dado caso el usuario no ha iniciado sesión o no tiene el rol de administrador.**
- Archivo de Autenticación (LoginC.php):
Este archivo es el encargado de validar las credenciales y establecer las sesiones.

```

C: > xampp > htdocs > solpuntoventa > LoginC.php
1  <?php
2  session_start();
3
4  // Datos de ejemplo para autenticación
5  $usuarios = [
6      'admin' => ['password' => 'admin', 'role' => 'admin'],
7      'user' => ['password' => 'user', 'role' => 'user']
8  ];
9
10 // Capturar datos del formulario
11 $usuario = $_POST['usuario'] ?? '';
12 $password = $_POST['password'] ?? '';
13
14 if (isset($usuarios[$usuario]) && $usuarios[$usuario]['password'] === $password) {
15     // Guardar información del usuario en la sesión
16     $_SESSION['usuario'] = $usuario;
17     $_SESSION['role'] = $usuarios[$usuario]['role'];
18     $_SESSION['last_activity'] = time(); // Establecer la última actividad
19
20     // Redirigir según el rol
21     if ($_SESSION['role'] === 'admin') {
22         header('Location: principal.html');
23     } else {
24         header('Location: perfil.html');
25     }
26     exit();
27 } else {
28     // Redirigir a login con mensaje de error
29     header('Location: login.php?error=1');
30     exit();
31 }
32 ?>

```

Ilustración 7: Mecanismo de autorización que permita o deniegue el acceso a ciertas rutas de la aplicación.

3. Implemente un mecanismo de autenticación que permita a los usuarios registrarse, iniciar sesión y cerrar sesión.

- Registro de Usuarios (registrar.php)

Primero, necesitas un formulario para que los usuarios se registren. Aquí es donde se almacenan las credenciales de los usuarios en la base de datos.

```

C: > xampp > htdocs > solpuntoventa > registrar.php
1  <?php
2  // register.php
3  session_start();
4  if ($_SERVER['REQUEST_METHOD'] === 'POST') {
5      // Datos de ejemplo para registrar
6      $usuario = $_POST['usuario'] ?? '';
7      $password = $_POST['password'] ?? '';
8
9      // Validar y almacenar datos en la base de datos (esto es solo un ejemplo)
10     // Aquí deberías conectar con tu base de datos y almacenar las credenciales de manera segura (hash).
11
12     // Ejemplo simple (sin base de datos):
13     $usuarios = [
14         'admin' => ['password' => 'admin', 'role' => 'admin'],
15         'user' => ['password' => 'user', 'role' => 'user']
16     ];
17
18     if (!isset($usuarios[$usuario])) {
19         $usuarios[$usuario] = ['password' => $password, 'role' => 'user'];
20         // Redirige a la página de inicio de sesión después del registro
21         header('Location: login.php');
22         exit();
23     } else {
24         echo "Usuario ya existe";
25     }
26 }
27 ?>
28
29 <!DOCTYPE html>
30 <html lang="es">
31 <head>
32     <meta charset="UTF-8">

```

Ilustración 8: Mecanismo de autenticación que permita a los usuarios registrarse, iniciar sesión y cerrar sesión.

4. Implementa un mecanismo para cerrar sesión de un usuario si ha pasado un tiempo determinado sin actividad de 5 mins.

- Implementar Tiempo de Inactividad (session_timeout.php)

Puedes implementar una verificación de tiempo de inactividad para cerrar sesión automáticamente después de 5 minutos (300 segundos).


```

C: > xampp > htdocs > solpuntoventa > ⚙️ Tiempo.php
1  <?php
2  session_start();
3
4  // Verificar si la sesión ha expirado
5  $inactividad = 300; // Tiempo en segundos (5 minutos)
6  if (isset($_SESSION['last_activity']) && (time() - $_SESSION['last_activity']) > $inactividad) {
7      // Si la sesión ha expirado, cerrar sesión
8      session_unset();
9      session_destroy();
10     header('Location: login.php');
11     exit();
12 }
13
14 // Actualizar el tiempo de última actividad
15 $_SESSION['last_activity'] = time();
16 ?>
17
18 <!DOCTYPE html>
19 <html lang="es">
20 <head>
21     <meta charset="UTF-8">
22     <meta name="viewport" content="width=device-width, initial-scale=1.0">
23     <title>Área de Usuario</title>
24     <style>
25         /* Agrega tu estilo aquí */
26     </style>
27 </head>
28 <body>
29     <h2>Bienvenido, <?php echo htmlspecialchars($_SESSION['usuario']); ?></h2>
30     <a href="logout.php">Cerrar Sesión</a>
31 </body>
32 </html>

```

Ilustración 9: Mecanismo para cerrar sesión después de 5 mins.

5. Investiga y describe los siguientes servicios de autenticación:

1. LDAP (PROTOCOLO LIGERO DE ACCESO A DIRECTORIOS):

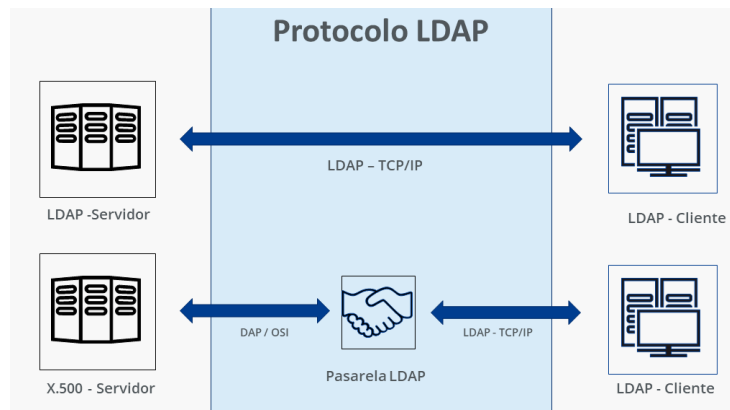


Ilustración 10: Protocolo Ligero De Acceso A Directorios.

Protocolo estándar utilizado para acceder y gestionar servicios de directorio distribuidos en una red. Un directorio es una base de datos jerárquica que almacena información sobre los usuarios, grupos y recursos de una organización.

PROTOCOLO LIGERO DE ACCESO A DIRECTORIOS.	
VENTAJAS.	<ul style="list-style-type: none">• Alta escalabilidad y capacidad de soportar un gran número de usuarios.• Funciona bien con otros protocolos de autenticación• Ampliamente adoptado en sistemas empresariales.• Permite centralizar la administración de usuarios en un repositorio único, accesible desde cualquier parte de la red.• Seguridad: LDAP soporta SSL y TLS, lo que garantiza la protección de datos confidenciales.• Flexibilidad en bases de datos: LDAP soporta diferentes tipos de bases de datos back-end, lo que permite a los administradores elegir la más adecuada para su entorno.

CARACTERISTICAS.	<ul style="list-style-type: none"> • Soporte para LDAPv3: Incluye mejoras como SASL, TLS y SSL, que fortalecen la seguridad. • Soporte para IPv6: Es compatible con la próxima generación del protocolo de Internet. • LDAP sobre IPC: Permite la comunicación interna dentro de un sistema sin necesidad de red, mejorando la seguridad. • API de C actualizada: Facilita a los programadores conectarse y usar servidores LDAP. • Soporte completo para LDIFv1: Garantiza la compatibilidad con el formato de intercambio de datos LDIF. • Servidor Stand-Alone mejorado: Incorpora un mejor control de acceso, un conjunto de hilos más eficiente y herramientas optimizadas.
USOS COMUNES.	<ul style="list-style-type: none"> • Autenticación de usuarios en redes corporativas. • Servicios como Active Directory. • Manejo de identidades.

Tabla 1: Protocolo Ligero De Acceso A Directorios.

2. RADIUS (SERVICIO DE USUARIO DE AUTENTICACIÓN REMOTA POR MARCACIÓN).

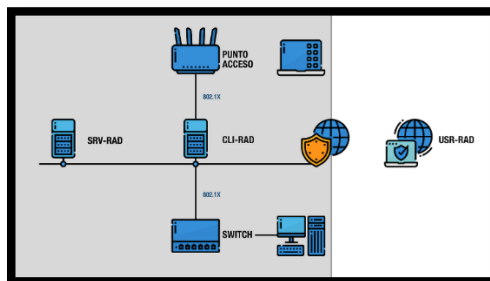


Ilustración 11: Servicio De Usuario De Autenticación Remota Por Marcación.

RADIUS es un protocolo cliente-servidor que facilita la autenticación, autorización y contabilidad (AAA) para acceder a redes. RADIUS se utiliza comúnmente en redes corporativas para autenticar usuarios que acceden de forma remota a la red mediante VPNs, Wi-Fi o servidores dial-in.

SERVICIO DE USUARIO DE AUTENTICACIÓN REMOTA POR MARCACIÓN.	
VENTAJAS.	<ul style="list-style-type: none"> • Maneja la autenticación y la autorización. • Puede registrar el uso de la red para contabilidad. • Es ligero. • Se puede integrar fácilmente en una variedad de redes.
USOS COMUNES.	<ul style="list-style-type: none"> • Redes inalámbricas. • VPN. • Acceso remoto. • ISP. • Acceso a servicios de red corporativa.

Tabla 2: Servicio De Usuario De Autenticación Remota Por Marcación.

3. TACACS+ (SISTEMA DE CONTROL DE ACCESO CONTROLADOR DE ACCESO TERMINAL PLUS).

Protocolo desarrollado por Cisco para proporcionar un enfoque más avanzado que RADIUS en la autenticación y autorización de usuarios. Aunque TACACS+ también es un protocolo AAA (Autenticación, Autorización, y Auditoría), ofrece una mayor granularidad en el control de acceso y más opciones de cifrado que RADIUS.

SERVICIO DE USUARIO DE AUTENTICACIÓN REMOTA POR MARCACIÓN.	
VENTAJAS.	<ul style="list-style-type: none"> • Mejora la seguridad con la separación de los procesos de autenticación y autorización. • Ofrece una mayor flexibilidad y capacidad de personalización.

USOS COMUNES.	<ul style="list-style-type: none"> • Control de acceso a dispositivos de red (routers, switches, etc.). • Entornos de red empresariales más seguros.
----------------------	--

Tabla 3: Servicio De Usuario De Autenticación Remota Por Marcación.

4. KERBEROS

Protocolo de autenticación basado en tickets que permite a los nodos en una red comunicarse de forma segura. Fue desarrollado en el MIT y es utilizado principalmente en redes de gran escala para la autenticación de usuarios en un entorno seguro. Kerberos usa criptografía de clave simétrica y un tercero de confianza, el KDC (Key Distribution Center), para autenticar a los usuarios y servicios.

SERVICIO DE USUARIO DE AUTENTICACIÓN REMOTA POR MARCACIÓN.	
VENTAJAS.	<ul style="list-style-type: none"> • Alta seguridad mediante autenticación mutua y cifrado de tickets. • Robusto para entornos empresariales y escalables.
USOS COMUNES.	<ul style="list-style-type: none"> • Autenticación en grandes redes corporativas. • Sistemas como Microsoft Active Directory. • Autenticación en servicios distribuidos como aplicaciones web y bases de datos.

Tabla 4: Servicio De Usuario De Autenticación Remota Por Marcación.

6. Investiga y describe los siguientes servicios de autorización:

5. LCA (LISTA DE CONTROL DE ACCESO):

Las Listas de Control de Acceso son un modelo de autorización en el cual se define una lista para cada recurso del sistema, especificando qué usuarios o grupos de usuarios tienen permiso para realizar operaciones específicas (como lectura, escritura, ejecución, etc.) sobre ese recurso. Cada objeto tiene su propia lista de acceso, donde se indican los permisos individuales para cada sujeto. Es una forma estática de control de acceso y es relativamente simple de implementar.

6. RBAC (CONTROL DE ACCESO BASADO EN ROLES):

Role-Based Access Control o Control de Acceso Basado en Roles es un sistema de autorización donde los permisos están asociados a roles en lugar de directamente a los usuarios. Los usuarios son asignados a uno o más roles, y los roles definen los permisos que los usuarios tienen. Este enfoque facilita la administración de permisos en sistemas grandes, ya que los administradores solo necesitan gestionar los roles y las asignaciones de los usuarios a estos roles.

7. ABAC (CONTROL DE ACCESO BASADO EN ATRIBUTOS):

Attribute-Based Access Control o Control de Acceso Basado en Atributos es un modelo en el que los permisos se otorgan en función de una combinación de atributos de los sujetos (usuarios), objetos (recursos) y del entorno. Estos atributos pueden ser cualquier información como el rol del usuario, la ubicación, la hora, o el tipo de dispositivo utilizado para acceder al sistema. ABAC ofrece una mayor flexibilidad y granularidad comparado con RBAC.

8. PBAC (CONTROL DE ACCESO BASADO EN POLÍTICAS):

Policy-Based Access Control o Control de Acceso Basado en Políticas es un modelo que define las decisiones de acceso utilizando políticas definidas previamente. Estas políticas son reglas que especifican las condiciones bajo las cuales los usuarios pueden acceder a recursos específicos. A menudo, se utilizan lenguajes de políticas como XACML para implementar este tipo de control de acceso. PBAC permite definir reglas complejas basadas en varios factores (roles, atributos, contexto, etc.) y es adecuado para entornos con múltiples requisitos de acceso.

9. CONCLUSION.

La implementación y comprensión de los protocolos de autenticación y autorización, como LDAP, RADIUS, TACACS+, Kerberos, así como los modelos de control de acceso LCA, RBAC, ABAC y PBAC, son fundamentales para un ingeniero en sistemas en el ámbito de la seguridad de la información. Estos protocolos y mecanismos permiten gestionar de manera eficiente la identidad y los accesos de usuarios en redes empresariales de igual manera proporcionan seguridad.

Un ingeniero en sistemas debe ser capaz de implementar soluciones de autenticación robustas que protejan los recursos de una organización, gestionando adecuadamente los roles, permisos y políticas de acceso. Esto asegura la integridad, confidencialidad y disponibilidad de los sistemas.

Con el desarrollo de esta práctica podemos concluir la importancia de dichos protocolos y mecanismos que garantizan que un ingeniero en sistemas pueda diseñar, implementar y mantener infraestructuras seguras, con adecuada protección y acceso eficiente a la información.

10. BIBLIOGRAFIA.

- (s.f.). Retrieved 09 de Septiembre de 2024, from <https://hopelchen.tecnm.mx/principal/sylabus/fpdb/recursos/r130657.PDF>
- IBM. (s.f.). Retrieved 09 de Septiembre de 2024, from <https://www.ibm.com/docs/es/aix/7.3?topic=network-remote-authentication-dial-in-user-service-server>
- *INTRODUCCION A LA VIRTUALIZACION*. (s.f.). Retrieved 09 de septiembre de 2024, from <http://148.202.167.116:8080/xmlui/bitstream/handle/123456789/2273/Introducci%C3%B3n%20a%20la%20virtualizaci%C3%B3n.pdf?sequence=1&isAllowed=y>