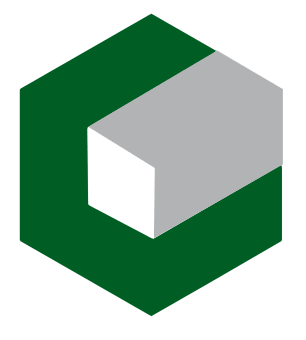
Universidad Tecnológica de La Habana “José Antonio Echeverría”

Facultad de Ingeniería Informática.



Aplicación web de código abierto para la gestión y autenticación de usuarios basada en Directorio Activo.

**Autor**: Carlos Daniel Vilaseca Illnait

**Tutores**: Dra. C. Raisa Socorro Llanes

Dra. C. Lisandra Bravo Ilisastigui

La Habana, Cuba

Julio,2024

Resumen

El documento presenta una solución propuesta para la gestión y autenticación de usuarios basada en Directorio Activo a través de LDAP. Se abordan temas como la historia detrás de esta propuesta, la selección del cliente LDAP que permita la comunicación con el directorio desde la aplicación, cómo se logra que la solución sea configurable y personalizable, las pruebas realizadas y los desafíos encontrados para lograr el despliegue de la solución.

**Palabras clave**: gestión de usuarios, código abierto, Directorio Activo, LDAP, personalización.

Abstract

The document presents a proposed solution for user management and authentication based on Active Directory through LDAP. It addresses topics such as the history behind this proposal, the selection of the LDAP client that allows communication with the directory from the application, how to make the solution configurable and customizable, the tests performed and the challenges encountered to achieve the deployment of the solution.

**Keywords**: user management, open source, Active Directory , LDAP, personalization.

Índice

[Introducción 1](#_Toc1)

[Capítulo 1 Fundamentación teórica 6](#_Toc2)

[1.1 Gestión de usuarios y Directorio Activo 6](#_Toc3)

[1.2 Protocolo Ligero de Acceso a Directorios (LDAP) 8](#_Toc4)

[1.3 Tecnologías y herramientas existentes 9](#_Toc5)

[1.3.1 Herramientas de gestion de AD 10](#_Toc6)

[1.3.2 Frameworks para el desarrollo web 11](#_Toc7)

[1.3.3 Clientes LDAP 12](#_Toc8)

[1.4 Archivos de configuración 12](#_Toc9)

[1.5.1 Variables de entorno (.env) 12](#_Toc10)

[1.5.1 Esquema JSON (JSON Schema) 12](#_Toc11)

[1.5.1 YAML 12](#_Toc12)

[1.5.1 TOML 12](#_Toc13)

[1.5 Requisitos de la aplicación y personalización del sistema 12](#_Toc14)

[Capítulo 2 Solución propuesta 13](#_Toc15)

[2.1 Arquitectura de la aplicación web 13](#_Toc16)

[2.2 Implementación de funciones básicas 13](#_Toc17)

[Capítulo 3 Validación de la solución 14](#_Toc18)

[3.1 Pruebas de integración 14](#_Toc19)

[Conclusiones 15](#_Toc20)

[Recomendaciones 16](#_Toc21)

[Bibliografía 16](#_Toc22)

Índice de tablas

[Tabla 1 Algunos atributos del esquema de usuario en el Directorio Activo 7](#_Toc1)

[Tabla 2 Comparación de tecnologías existentes en cuanto a capacidad de personalizacion y simplicidad de despliegue 10](#_Toc2)

[Tabla 3 Comparación entre frameworks de desarrollo wrb 11](#_Toc3)

# Introducción

En el mundo digital actual, la gestión de usuarios es un aspecto crucial para cualquier organización o negocio que opere en línea. Con el aumento del uso de plataformas en línea y servicios digitales, gestionar el acceso, los permisos y la seguridad de los usuarios se ha convertido en un desafío complejo y fundamental [1], [2], [3].

La gestión de usuarios implica la creación y mantenimiento de cuentas de usuario, la asignación de roles y permisos, y la garantía de que solo las personas autorizadas tengan acceso a la información y los sistemas adecuados. Esto es especialmente importante en entornos donde múltiples usuarios, con diferentes niveles de acceso, interactúan con sistemas sensibles o información confidencial. Una gestión de usuarios bien implementada puede mejorar la seguridad, proteger los datos y garantizar que los usuarios tengan una experiencia personalizada [1], [2], [3].

Los Directorios Activos (AD por sus siglas en inglés) son una solución robusta y ampliamente adoptada para la gestión de usuarios en diversidad de entornos. Proporcionan una forma centralizada y segura de gestionar las identidades, los permisos y la autenticación. Al utilizar un AD, las organizaciones pueden controlar y monitorear el acceso a sus recursos de red y aplicaciones, garantizando que solo las personas autorizadas puedan acceder a la información confidencial [4], [5], [6].

Un AD funciona como una base de datos central que almacena información sobre las identidades, como nombres de cuenta, contraseñas, direcciones de correo electrónico, roles y pertenencia a grupos. Esta base de datos se puede integrar con múltiples sistemas y aplicaciones, lo que proporciona un único punto de autenticación y gestión de acceso. Esto significa que los administradores de sistema pueden crear y administrar cuentas de forma eficiente, y los usuarios finales pueden acceder a los recursos que necesitan con un único conjunto de credenciales [4], [5], [6].

Una de las principales ventajas de los directorios activos es su capacidad para escalar y adaptarse a las necesidades de una organización en crecimiento. Pueden manejar fácilmente un gran número de usuarios y proporcionar una forma flexible de gestionar los cambios en los roles y permisos del personal. Los AD también ofrecen una seguridad mejorada, con funciones de cifrado y la capacidad de aplicar políticas de contraseñas seguras [8], [16].

Los AD usan un protocolo de comunicación llamado Protocolo Ligero de Acceso a Directorios (LDAP por sus siglas en inglés); un protocolo de red abierto y estándar para acceder y mantener información en un servicio de directorio. LDAP juega un papel crucial en la gestión de usuarios y la autenticación en muchos sistemas y aplicaciones [5], [7], [8], [9], [10], [11].

Para la gestión de directorios, existen herramientas como *Remote Server Administration Tools* [14], Webmin [15], samba4-manager [12] y ADwebmanager [13], que usan LDAP para la conexión y gestión del directorio. Sin embargo estas pueden resultar insuficientes en términos de personalización y facilidad de despliegue a necesidades organizacionales o de la comunidad.

La gestión de AD a menudo se encuentra con obstáculos en términos de personalización y simplicidad debido a su enfoque en satisfacer necesidades específicas, manejar complejidades técnicas, priorizar seguridad y estabilidad, enfrentar restricciones arquitectónicas, y evolucionar a partir de diseños que no consideraron la personalización y la facilidad de despliegue.

A partir de esta **situación problemática** se formula el siguiente **problema**: ¿Cómo puede desarrollarse una herramienta de gestión para AD que ofrezca una mayor personalización y facilidad de despliegue, sin sacrificar la seguridad, estabilidad y capacidad para manejar las complejidades técnicas inherentes?. Para solucionar el problema se tienen como objeto de estudio los AD y la comunicación a través de LDAP; el campo de acción serían las herramientas de gestión de directorio activo.

Como hipótesis se plantea que desarrollar una aplicación web de código abierto para la gestión de AD, con archivos de configuración que permitan la personalización y un proceso de despliegue documentado, ofrecerá ventajas significativas en términos de flexibilidad y facilidad de uso.

Para demostrar esta hipótesis se plantea como objetivo general crear una consola de administración de código abierto adaptable, y fácil de desplegar para la gestión de AD, ofreciendo una solución práctica y flexible para diferentes contextos.

A partir de este objetivo general, se derivan los siguientes objetivos específicos y tareas:

1. Analizar los requisitos de la aplicación y la personalización del sistema:
   1. Documentar los requisitos funcionales y no funcionales que debe cumplir la aplicación.
   2. Analizar diferentes casos de uso para identificar las opciones de personalización.
   3. Documentar los requisitos de personalización, incluyendo la interfaz de usuario y ajustes de seguridad.
2. Seleccionar tecnologías adecuadas:
   1. Evaluar diferentes clientes LDAP disponibles en el mercado, considerando factores como compatibilidad, rendimiento y facilidad de integración.
   2. Elegir el cliente LDAP que mejor se alinee con los requisitos funcionales y no funcionales previamente definidos.
3. Implementar la arquitectura y funciones básicas de la aplicación:
   1. Establecer la arquitectura base del proyecto y configurar el ambiente de desarrollo necesario.
   2. Configurar el cliente LDAP seleccionado y las herramientas asociadas para iniciar el desarrollo.
   3. Desarrollar mecanismos de autenticación que interactúen con el AD utilizando el cliente LDAP seleccionado.
   4. Implementar funcionalidades críticas para la gestión de usuarios y grupos utilizando el cliente LDAP, incluyendo operaciones de lectura, eliminación y actualización.
   5. Realizar pruebas unitarias y funcionales para garantizar la confiabilidad y corrección de estas funciones básicas.
4. Realizar pruebas para asegurar el correcto funcionamiento del sistema:
   1. Diseñar y ejecutar pruebas de integración que verifiquen el correcto funcionamiento del sistema en su conjunto, desde la autenticación hasta la gestión de recursos.
   2. Documentar los resultados de las pruebas y realizar los ajustes necesarios basados en los hallazgos.
   3. Extender el conjunto de pruebas de integración para abarcar nuevas funcionalidades y garantizar la estabilidad y compatibilidad del sistema ante cambios y actualizaciones futuras.

El alcance de las prácticas profesionales 2 es del objetivo 1 al 4; de cara a la tesis se proponen los siguientes objetivos y tareas:

1. Simplificar y documentar el proceso de despliegue:
   1. Identificar y documentar estrategias y herramientas que simplifiquen el proceso de instalación y configuración inicial de la aplicación.
   2. Utilizar contenedores Docker para simplificar el proceso de puesta en marcha de la aplicación.
   3. Elaborar documentación detallada del proceso de despliegue.
2. Desplegar documentación:
   1. Crear y estructurar la documentación técnica que incluya la descripción del sistema, la arquitectura, y las guías de desarrollo.
   2. Documentar referencias de API y/o archivos de configuracion que sean claras y accesibles.
   3. Publicar la documentación en un sitio web accesible.
3. Desplegar demo:
   1. Configurar el servidor donde se va a desplegar.
   2. Configurar Docker para simplificar el despliegue.
   3. Configurar CI/CD para despliegues automáticos.
   4. Ejecutar el primer despliegue exitoso.

Como valor práctico con la realización de este trabajo se espera un diseño de software de una herramienta para la gestión de directorio activo que sea personalizable y fácil de desplegar.

La estructura del informe se organiza de la siguiente manera:

Capítulo 1 Fundamentación teórica. En este capítulo se presenta una explicación teórica sobre los conceptos y tecnologías fundamentales para la gestión de usuarios y AD. Se analizarán en profundidad temas como la importancia de la gestión de usuarios en entornos digitales, los principios de seguridad y autenticación, y el funcionamiento de los directorios activos y LDAP. También se abordarán las diferentes herramientas existentes para la gestión de AD, sus ventajas y limitaciones, y se establecerá el marco teórico que sustenta la propuesta de solución planteada en este trabajo.

Capítulo 2 Descripción de la propuesta de solución. Este capítulo aporta una explicación detallada sobre la propuesta de solución a los problemas identificados en la gestión de AD. Se describirá la arquitectura de la aplicación web de código abierto propuesta, sus funcionalidades principales, y cómo se abordarán los requisitos de personalización y simplicidad de despliegue. Además, se explicarán las decisiones de diseño y las tecnologías seleccionadas para el desarrollo de la herramienta, así como los beneficios esperados en términos de seguridad, estabilidad y facilidad de uso.

Capítulo 3 Validación de la propuesta de solución. Este capítulo está dedicado a la validación de la propuesta de solución mediante la realización de pruebas exhaustivas. Se diseñarán y ejecutarán pruebas de integración para verificar el correcto funcionamiento del sistema en su conjunto, desde la autenticación hasta la gestión de recursos. Los resultados de estas pruebas se documentarán y se realizarán los ajustes necesarios basados en los hallazgos.

# Capítulo 1 Fundamentación teórica

El propósito de este capítulo es proporcionar una base sólida sobre los conceptos y tecnologías fundamentales para la gestión de usuarios y AD, que sustentan la propuesta de solución planteada. Se explorará la importancia de una gestión eficaz de usuarios en entornos digitales, abordando los principios clave de seguridad y autenticación que son esenciales para proteger la integridad y confidencialidad de los datos. Asimismo, se detallarán los conceptos de AD y LDAP, explicando su funcionamiento y relevancia en la administración de identidades y accesos. Además, se realizará un análisis exhaustivo de las herramientas existentes para la gestión de AD, evaluando sus ventajas y limitaciones, con el objetivo de establecer un marco teórico robusto que guíe el desarrollo de una solución personalizable y fácil de desplegar.

## 1.1 Gestión de usuarios y Directorio Activo

En la administración de sistemas informáticos, la gestión de usuarios es un componente esencial para asegurar el correcto funcionamiento y la seguridad de la infraestructura tecnológica de una organización. El Directorio Activo (AD) es una herramienta crucial en este ámbito, ya que permite la centralización y automatización de la gestión de usuarios, dispositivos y recursos. Este epígrafe aborda los conceptos fundamentales de la gestión de usuarios, las ventajas y características del AD, así como su implementación y administración.

Un AD es una base de datos central que se utiliza para administrar y organizar los recursos de una red de computadoras. La gestión de usuarios es una función importante en un AD, ya que permite crear, eliminar y editar los perfiles de los usuarios en un dominio. Además, también implica la asignación de permisos y roles, que determinan el nivel de acceso y las acciones que cada usuario puede realizar en la red [8], [17], [18].

El esquema de AD contiene definiciones formales de cada clase de objeto que se puede crear en un directorio. Por ejemplo, en el caso de los usuarios, el esquema define atributos como el nombre, apellido, nombre para mostrar, nombre de inicio de sesión, dirección de correo electrónico, número de teléfono [5], [9], [16]. En la **Tabla 1** se muestran algunos atributos del esquema de usuario de un AD, así como ejemplos de valores que pueden tomar estos atributos.

Tabla 1 Algunos atributos del esquema de usuario en el Directorio Activo

|  |  |  |
| --- | --- | --- |
| **Atributo** | **Descripción** | **Ejemplo** |
| displayName | Nombre para mostrar del usuario | Juan Pérez |
| distinguishedName | Nombre único del objeto en el directorio | CN=jperez,OU=Dirección,DC=ejemplo,DC=com |
| givenName | Nombre del usuario | Juan |
| mail | Dirección de correo electrónico del usuario | juan.perez@ejemplo.com |
| objectSid | Identificador de seguridad (SID) del usuario | S-1-5-21-1234567890-1234567890-1234567890-1234 |
| sAMAccountName | Nombre de inicio de sesión compatible con versiones anteriores de Windows | jperez |
| sn | Apellido del usuario | Pérez |
| telephoneNumber | Número de teléfono del usuario | +34 123 456 789 |
| userPrincipalName | Nombre de inicio de sesión del usuario | jperez@ejemplo.com |

## 1.2 Protocolo Ligero de Acceso a Directorios (LDAP)

LDAP es un estándar de protocolo utilizado para acceder y gestionar información almacenada en directorios de manera eficiente y segura. En el contexto de la administración de sistemas informáticos, LDAP desempeña un papel fundamental al facilitar la búsqueda, autenticación y gestión de usuarios, dispositivos y otros recursos.

Este epígrafe explora los principios fundamentales de LDAP, incluyendo su arquitectura, funcionamiento y principales características. Se examinará cómo LDAP permite la estructuración jerárquica de la información mediante la utilización de entradas y atributos, lo cual facilita la organización y el acceso a datos.

**Arquitectura de LDAP**

LDAP se basa en una arquitectura cliente-servidor, donde el cliente LDAP envía solicitudes al servidor LDAP para realizar diversas operaciones, como búsquedas, actualizaciones y autenticaciones de información almacenada en el directorio. Esta arquitectura facilita la gestión centralizada y eficiente de los datos.

* **Cliente LDAP**: Es el software que realiza peticiones de búsqueda, modificación o consulta de información almacenada en el servidor LDAP.
* **Servidor LDAP**: Es el software que almacena la base de datos de directorio y responde a las peticiones de los clientes LDAP. El servidor LDAP gestiona y organiza la información en forma de entradas almacenadas en un árbol de directorio.
* **Protocolo de Comunicación**: LDAP define cómo se comunican el cliente y el servidor a través de un protocolo eficiente y ligero, diseñado principalmente para la lectura, búsqueda y modificación de información en directorios.

**Funcionamiento de LDAP**

El funcionamiento de LDAP se centra en la transferencia de mensajes entre el cliente y el servidor LDAP a través de conexiones seguras. Utiliza un modelo de datos basado en árbol, donde cada entrada en el directorio está identificada por un Nombre Distinguido (*Distinguished Name*, visto en la **Tabla 1**) único. Esto proporciona una estructura flexible para almacenar y acceder a información sobre usuarios, dispositivos y recursos dentro del directorio.

**Características principales de LDAP**

LDAP ofrece varias características clave que lo hacen ideal para la gestión de directorios en entornos empresariales:

* **Eficiencia**: permite búsquedas rápidas y eficientes de información en grandes volúmenes de datos.
* **Seguridad**: soporta protocolos de seguridad como TLS (*Transport Layer Security*) para proteger la integridad y confidencialidad de los datos transmitidos.
* **Escalabilidad**: capacidad para manejar grandes cantidades de datos y usuarios dentro de un directorio, adaptándose a las necesidades de crecimiento de una organización.
* **Interoperabilidad**: estándar abierto compatible con una amplia gama de plataformas y sistemas de directorio.

**Aplicaciones de LDAP**

LDAP se utiliza ampliamente en la autenticación de usuarios, control de acceso y gestión de identidades en sistemas operativos, aplicaciones web y servicios de correo electrónico. Su flexibilidad y robustez lo convierten en una herramienta fundamental para la integración y administración de infraestructuras de TI empresariales.

## 1.3 Tecnologías y herramientas existentes

En el ámbito de la gestión de AD, existen diversas tecnologías y herramientas diseñadas para facilitar y optimizar esta tarea esencial en la administración de sistemas informáticos. Estas herramientas no solo permiten una gestión más eficiente de los recursos y usuarios dentro de una organización, sino que también contribuyen a mejorar la seguridad y el control de acceso.

Este epígrafe se centrará en revisar las herramientas actuales más utilizadas para la gestión de AD, detallando sus principales funcionalidades y características. Asimismo, se analizarán las ventajas y limitaciones de estas soluciones.

### 1.3.1 Herramientas de gestion de AD

Para gestionar efectivamente un AD, es crucial contar con herramientas que simplifiquen las tareas administrativas y ofrezcan opciones flexibles de personalización y despliegue. La capacidad de personalización se refiere a la flexibilidad que ofrece la herramienta para ajustar su interfaz y funcionalidades según las necesidades específicas del usuario u organización. Por otro lado, la simplicidad de despliegue se refiere a la facilidad con la que una herramienta puede ser instalada y configurada, asegurando una implementación eficiente y sin complicaciones.

En la **Tabla 2** se presentan algunas herramientas para la gestión de AD, resaltando sus características principales en términos de personalización y simplicidad de despliegue.

Tabla 2 Comparación de tecnologías existentes en cuanto a capacidad de personalizacion y simplicidad de despliegue

|  |  |  |
| --- | --- | --- |
| **Herramienta** | **Capacidad de personalización (interfaz y apariencia)** | **Simplicidad de despliegue** |
| RSAT | Limitada, la personalización se limita a ajustes mínimos dentro del entorno de Windows | Compleja, requiere conocimientos técnicos específicos y sólo disponible para ciertas versiones de Windows |
| Webmin | Moderada, permite cierta personalización a través de temas y ajustes de interfaz, pero con limitaciones en la profundidad de las modificaciones | Relativamente fácil de instalar, pero la configuración inicial puede ser confusa y tiene una curva de aprendizaje |
| samba4-manager | Limitada, diseñada específicamente para la gestión de Samba4, con opciones de personalización limitadas | Tedioso, especialmente en entornos grandes o complejos |
| ADwebmanager | Limitada, diseñada para funciones comunes de AD, con mínimas opciones de personalización de interfaz | Sencillo para usuarios con experiencia en AD, pero puede presentar desafíos para usuarios sin experiencia |

### 1.3.2 Frameworks para el desarrollo web

En el ámbito del desarrollo web, los frameworks juegan un papel crucial al proporcionar estructuras y herramientas que simplifican y aceleran la creación de aplicaciones. Estos frameworks ofrecen una base sólida para la implementación de funcionalidades complejas, facilitando la interacción entre el diseño front-end y la lógica de negocio back-end. Esta sección explora diversos frameworks destacados en el panorama actual, analizando sus características, ventajas y aplicaciones específicas en el desarrollo web moderno.

Tabla 3 Comparación entre frameworks de desarrollo wrb

|  |  |  |  |
| --- | --- | --- | --- |
| **Característica** | **SvelteKit** | **Next.js** | **Nuxt.js** |
| **Eficiencia** | Alto rendimiento con compilación previa y sitios estáticos | Buena eficiencia con renderizado del lado del servidor y en el cliente | Eficiencia en el desarrollo de aplicaciones web, con facilidades para la creación de aplicaciones universal y estáticamente generadas |
| **Flexibilidad** | Gran flexibilidad en personalización y configuración | Flexible para la creación de diferentes tipos de aplicaciones web | Flexible y adaptable a diferentes tipos de proyectos, con un enfoque en la simplicidad y facilidad de uso |
| **Escalabilidad** | Altamente escalable para proyectos de diferentes tamaños | Buena capacidad de escalar y manejar proyectos de gran envergadura | Puede escalar adecuadamente para manejar proyectos de diversos tamaños y complejidades |
| **Curva de aprendizaje** | Muy baja, con sintaxis simple y familiar | Moderada, requiere familiarizarse con sus conceptos y funcionalidades | Moderada, especialmente para aquellos que están familiarizados con Vue.js |
| **Comunidad** | En crecimiento, con soporte activo y recursos disponibles | Amplia comunidad de desarrolladores, con gran cantidad de recursos y soporte en línea | Comunidad activa y en crecimiento, con soporte y recursos disponibles |

Es importante destacar que realizar una comparación exhaustiva de todos los frameworks disponibles es complicado por varias razones:

* Abundancia de opciones: Actualmente, existen cientos de frameworks de desarrollo web, cada uno con características y ventajas únicas.
* Complejidad inherente: Los frameworks son complejos y ofrecen una amplia gama de características, lo que dificulta una comparación exhaustiva y detallada.
* Variabilidad en los requisitos: Las aplicaciones web tienen requisitos diversos, lo que significa que un framework ideal para una aplicación puede no serlo para otra.

Por lo tanto, es difícil determinar que un framework sea superior a otro de manera generalizada. La elección del mejor framework para una aplicación específica depende en gran medida de los requisitos particulares de esa aplicación y de las preferencias del equipo de desarrollo. En última instancia, la selección de un framework adecuado se basa en su capacidad para resolver los problemas específicos de desarrollo que se presentan en el contexto de la aplicación deseada.

### 1.3.3 Clientes LDAP

## 1.4 Archivos de configuración

### 1.4.1 Variables de entorno (.env)

### 1.4.1 Esquema JSON (JSON Schema)

### 1.4.1 YAML

### 1.4.1 TOML

## 1.5 Requisitos de la aplicación y personalización del sistema

* Documentación de requisitos funcionales y no funcionales.
* Análisis de diferentes casos de uso para identificar opciones de personalización.
* Requisitos de personalización incluyendo interfaz de usuario y ajustes de seguridad.

# Capítulo 2 Solución propuesta

## 2.1 Arquitectura de la aplicación web

* Descripción de la arquitectura base del proyecto.
* Configuración del ambiente de desarrollo necesario.
* Selección de tecnologías y herramientas para el desarrollo.

## 2.2 Implementación de funciones básicas

* Configuración del cliente LDAP seleccionado.
* Desarrollo de mecanismos de autenticación que interactúen con el Directorio Activo.
* Implementación de funcionalidades críticas para la gestión de usuarios y grupos.

# Capítulo 3 Validación de la solución

## 3.1 Pruebas de integración

* Diseño y ejecución de pruebas de integración que verifiquen el funcionamiento del sistema desde la autenticación hasta la gestión de recursos.
* Documentación de resultados de las pruebas y ajustes necesarios basados en los hallazgos.

# Conclusiones

# Recomendaciones

# Bibliografía

[1]M. A. Thakur y R. Gaikwad, «User identity and Access Management trends in IT infrastructure- an overview», en 2015 International Conference on Pervasive Computing (ICPC), Pune, India: IEEE, 2015, pp. 1-4. doi: 10.1109/PERVASIVE.2015.7086972. Disponible en: http://ieeexplore.ieee.org/document/7086972/. [Accedido: 11 de junio de 2024]  
   
   
 [2]A. Josang et al., «Local user-centric identity management», J Trust Manag, vol. 2, n.o 1, p. 1, 2015, doi: 10.1186/s40493-014-0009-6. Disponible en: http://www.journaloftrustmanagement.com/content/2/1/1. [Accedido: 11 de junio de 2024]  
   
   
 [3]J. M. Kizza, «Access Control and Authorization», en Guide to Computer Network Security, Cham: Springer International Publishing, 2024, pp. 195-214. doi: 10.1007/978-3-031-47549-8\_9. Disponible en: https://link.springer.com/10.1007/978-3-031-47549-8\_9. [Accedido: 11 de junio de 2024]  
   
   
 [4]G. Carter, LDAP System Administration. O’Reilly Media, Inc., 2003.  
   
   
 [5]R. Harrison, «Lightweight Directory Access Protocol (LDAP): Authentication Methods and Security Mechanisms». IETF, junio de 2006. Disponible en: https://datatracker.ietf.org/doc/html/rfc4513. [Accedido: 15 de mayo de 2024]  
   
   
 [6]M. A. Thakur y R. Gaikwad, «User identity and lifecycle management using LDAP directory server on distributed network», en 2015 International Conference on Pervasive Computing (ICPC), Pune, India: IEEE, 2015, pp. 1-3. doi: 10.1109/PERVASIVE.2015.7086970. Disponible en: http://ieeexplore.ieee.org/document/7086970/. [Accedido: 11 de junio de 2024]  
   
   
 [7]J. Sermersheim, «Lightweight Directory Access Protocol (LDAP): The Protocol», www.rfc-editor.org, jun. 2006, doi: 10.17487/RFC4511. Disponible en: https://www.rfc-editor.org/rfc/rfc4511  
   
   
 [8]A. Bartlett, «Samba 4 -Active Directory». 2005. Disponible en: https://www.samba.org/samba/news/articles/abartlet\_thesis.pdf  
   
   
 [9]R. E. Voglmaier, The ABCs of LDAP. CRC Press, 2003.  
   
   
 [10]RedHat, «What is LDAP authentication?» RedHat, 3 de junio de 2022. Disponible en: https://www.redhat.com/en/topics/security/what-is-ldap-authentication  
   
   
 [11]R. Janice, «LDAP authentication with Microsoft Entra ID - Microsoft Entra», 23 de octubre de 2023. Disponible en: https://learn.microsoft.com/en-us/entra/architecture/auth-ldap. [Accedido: 28 de junio de 2024]  
   
   
 [12]S. Graber, stgraber/samba4-manager. 2024. Disponible en: https://github.com/stgraber/samba4-manager. [Accedido: 28 de junio de 2024]  
   
   
 [13]V. S. G. Jerez, VicentGJ/AD-webmanager. 2024. Disponible en: https://github.com/VicentGJ/AD-webmanager. [Accedido: 28 de junio de 2024]  
   
   
 [14]D. Han, «Remote Server Administration Tools - Windows Server». Microsoft, 3 de mayo de 2024. Disponible en: https://learn.microsoft.com/en-us/troubleshoot/windows-server/system-management-components/remote-server-administration-tools. [Accedido: 10 de junio de 2024]  
   
   
 [15]M. Karzyński, Webmin Administrator’s Cookbook. Packt Publishing Ltd, 2014.  
   
   
 [16]T. Howes y M. Smith, LDAP: programming directory-enabled applications with lightweight directory access protocol. en Macmillan technology series. Indianapolis, Ind: Macmillan Technical Publishing, 1997.  
   
   
 [17]Dansimp, «Active Directory Accounts», 18 de mayo de 2023. Disponible en: https://learn.microsoft.com/en-us/windows-server/identity/ad-ds/manage/understand-default-user-accounts. [Accedido: 30 de junio de 2024]  
   
   
 [18]A. Imanudin, Active Directory Berbasis Linux Samba 4. Excellent Publishing, 2019.