Autenticación mediante FreeIPA en Linux Centos

Nelson D. Insaurralde, Agustin F. Rios, Rodrigo Ernesto Zalazar[1]

[1] Fac. de Ciencias Exactas Naturales y Agrimensura, Universidad Nacional del Nordeste, Corrientes
milococoinsa@gmail.com, agustinfruni@gmail.com, rodrigo.zalazar@comunidad.unne.edu.ar

Abstract. Con la evolución de las organizaciones, los niveles de infraestructura de servicios informáticos y calidad de los mismos también avanzan tanto en capacidad de rendimiento y alcance como en la mejora de la posibilidad de ofrecer ayuda a cada vez más usuarios a un costo menor, a su vez, los productos de Software deben mantenerse capaces de responder a las expectativas como consecuencia del desarrollo de las plataformas donde se utilizan en términos de necesidades de seguridad, eficiencia y aplicabilidad.

En el marco del trabajo de investigación en equipo de la asignatura Redes de Datos, cuarto año de la Carrera LSI, proponemos la solución FreeIPA como solución de código abierto, para la integración de servicios que abarca desde la administración de identidades hasta la entrega y utilización de certificados, además de brindar la posibilidad de gestión mediante un web service interface integrado.

Keywords: Open Source, Authentication, Authorization, FreeIPA..

1 Introducción

Con la evolución de los sistemas informáticos, la cantidad de servicios disponibles aumentan y con ello se puede encontrar una gran variación entre ellos en términos de objetivos, nivel de detalle de sus aplicaciones, compatibilidad con otros servicios, alcance y seguridad lo que puede complicar el uso de los mismos debido a la falta de claridad en su funcionamiento. Los servicios de integración intentan mitigar este problema brindando una interfaz y una estructura definidas que actúa como contenedor para una solución de servicios comúnmente relacionados que permiten la interoperabilidad o compatibilidad en su uso.

Uno de los proveedores OpenSource[2] conocidos en grán medida por su sistema operativo empresarial “Red Hat Enterprise Linux”, Red Hat Inc, propone su propio conjunto completo de tecnologías de mensajería e integración, diseñado para conectar las aplicaciones y los datos en todas las infraestructuras híbridas [1].

Particularmente, nos interesa su propuesta FreeIPA[3] como herramienta adecuada para la gestión centralizada de una infraestructura Linux, de manera que el control de los diferentes equipos se pueda realizar de una forma sencilla, y de forma que puedan

suplirse las herramientas actuales que se puedan estar empleando en el centro de trabajo.

FreeIPA es un proyecto de suite de software de código fuente libre mantenida por el Proyecto Fedora, patrocinada por RedHat, el nombre significa Identida-Políticas-Auditoría Libre (fig.1).

Esta herramienta se utiliza para crear un controlador de dominio entre máquinas Linux y Unix ,por lo que no admite clientes Windows, pero si puede sincronizarse con un dominio de Active Directory para permitir la integración con servidores de Windows.

Los usuarios, máquinas, servicios y políticas están configuradas en un solo lugar, con la misma herramienta, lo cual le permite al administrador gestionarlo en función de las necesidades que quiera adoptar.

Realmente, FreeIPA no hace ninguna tarea que un administrador no pudiese realizar antes de su existencia, ya que lo único que hace es unir todas las herramientas y hacerlo más fácil y cómodo.

Los paquetes de software utilizados en esta propuesta son de naturaleza de código abierto -OpenSource-[2], y lo integran los siguientes productos:
- de sistema operativo Linux
- de administración de servicios y entidades,
- de autenticación y sesiones,
- de instalación y configuración de sistema
- de sincronización horaria
- de integración de certificados

La actividad principal consiste en la instalación de un servidor FreeIPA basado en linux y en la utilización de la interfaz web integrada para la administración de servicios, usuarios, obtención de informes, actualizaciones del sistema, obtención de información referidas a las características del sistema y del soporte físico involucrado.

2 Instrumentación

Para el despliegue de la solución se requiere disponer de los siguientes paquetes de software:
- · Sistema operativo Linux de 64 bits
- · FreeIPA
- · 389 Directory Server (LDAP)
- · Mit Kerberos
- · Cockpit

De igual forma, dependiendo de la versión linux utilizada,la instalación del paquete FreeIpa ya cuenta con todos los elementos necesarios para su uso. Las distribuciones de Linux soportadas por el servidor son Fedora, RedHat, Centos, OpenSuSe y Debian .

Específicamente para las pruebas, se ha elegido CentOS Stream 9. En la actualidad gran parte de las infraestructuras informáticas del mundo están soportadas por Linux y tecnologías libres y de código abierto. Las ventajas de usar software de este estilo son varias. Algunos ejemplos serían los siguientes: gran eficiencia, costes muy bajos (o incluso nulos), y una gran comunidad de desarrolladores y colaboradores.

Componentes

Los elementos que integran esta solución de software son:

389 Directory Service

El directorio activo está constituido sobre un servidor LDAP, el cual se encarga de la gestión de identidades, autenticación basada en Kerberos, servicios de autorización y otras políticas. Para hacer que la manipulación de las entradas sea más fácil, FreeIPA proporciona una interfaz CLI y Web para el usuario, de forma que le resulte más sencillo el manejo de las herramientas y la modificación de los datos.

Podemos destacar los siguientes aspectos:

•Protocolo ligero de acceso a directorios.
•Base de datos optimizada de lectura jerárquica.
•Consiste en objetos con atributos.
•Las consultas a LDAP se pueden realizar usando el cliente ldap o usando kerberos.

KERBEROS

Kerberos proporciona servicios de autenticación para todo el dominio de FreeIPA, para los usuarios, servicios y componentes. Su función es permitir que, en una red insegura, poder demostrar las identidades de los clientes de forma segura. Para ello utiliza un sistema mediante tickets, los cuales se utilizan para demostrar la autenticidad de los usuarios.

Se necesita un tercero (un Centro de distribución de claves) para autenticarse entre un cliente y un servicio o equipo host. Básicamente, Kerberos se reduce a:

•Un protocolo para autenticación.
•Usa tickets para autenticarse.
•Evita almacenar contraseñas localmente o enviarlas a través de Internet.
•Involucra a un tercero de confianza (kdc).
•Construido en la criptografía de clave simétrica.
•Sso es un procedimiento de autenticación que permite al usuario acceder a varios servicios con un solo inicio de sesión.

El funcionamiento básico, consiste en:

- El cliente envía un mensaje al servidor (AS) que autentica los usuarios.
- Una vez autenticado, el servidor (TGS) le da un “ticket” al cliente válido por X tiempo, el cual incluye la información del cliente
- El cliente puede acceder a la red y a los recursos a los que su ticket le dé acceso durante el tiempo que este sea válido (fig.2).

NTP

Otro servicio que ofrece FreeIPA es NTP (Network Time Protocol) el cual es un protocolo de Internet para sincronizar los relojes de los sistemas que integran el dominio. Es muy importante tener el reloj sincronizado en todas las máquinas integradas en el reino Kerberos, porque los tickets tienen marca de tiempo y una diferencia mínima de tiempo (máximo 5 min) puede generar graves problemas, tales como que no funcione el uso de tickets. Si tiene problemas con kerberos, verifique que su reloj esté sincronizado con el reloj del servidor.

DNS

DNS son las iniciales de Domain Name System (sistema de nombres de dominio) y es una tecnología basada en una base de datos que sirve para resolver nombres en las redes, es decir, para conocer la dirección IP de la máquina donde está alojado el dominio al que queremos acceder.

FreeIPA permite gestionar y servir registros DNS en el dominio utilizado la interfaz web o CLI. La integración de DNS se basa en el proyecto bind9-dyndb-ldap, que mejora el servidor de nombres BIND para poder utilizar instancias LDAP.

Como era de esperar, su funcionamiento es a través de OpenSSL:

- Bind9 se usa como un servidor DNS.
- Usa el backend de LDAP para leer zonas y registros.
- Las zonas se pueden crear / actualizar a través de la web de FreeIPA, la API o mediante el uso de actualizaciones dinámicas.

DOGTAG

El sistema de certificado de Dogtag es una autoridad de certificación de código abierta (CA) de clase empresarial, el cual integra dos servicios Pki y Certmonger.

PKI

PKI firma y publica certificados para hosts y servicios de FreeIPA. La interfaz de gestión de FreeIPA proporciona una API para solicitar, mostrar y encontrar certificados. Dado que los certificados utilizados por los hosts y servicios de los clientes de FreeIPA tienen una validez limitada, la infraestructura también necesita una renovación confiable de los certificados.

Para tal fin, se ejecuta un daemon, Certmonger en todos los clientes y se encarga de la renovación de forma transparente para los servicios que lo utilizan.

CERTMONGER

Es un demonio que supervisa los certificados y alerta de una inminente expiración.

COCKPIT

Cockpit ofrece una interfaz gráfica para su servidor, a la que se puede acceder a través de un navegador web. Cockpit facilita el inicio de contenedores, la administración del almacenamiento, la configuración de redes, la inspección de

registros y la realización de tareas del sistema con un ratón. Puedes pensar en Cockpit como una "interfaz de escritorio" gráfica, pero para servidores individuales (fig.3).

Entendemos que el Active Directory se encuentra también en cluster, pero lo mostramos como un servidor por simplicidad, dado que la arquitectura del AD se encuentra fuera del alcance de este documento. Igualmente, FreeIPA aparece como un cluster de tres servidores, aunque podrían emplearse dos.

A nivel de red aparece un único router, pero entendemos que entre las diferentes redes podrán existir diferentes routers, firewalls y demás elementos de electrónica de red. Por último, indicar que el servidor de Foreman se ha ubicado en la parte inferior por mostrar un esquema sencillo, pero también debería tener conectividad directa en la red de gestión.

3 Ejecución

Una vez instalados los paquetes necesarios y montado el servidor FreeIPA podemos ingresar a la interfaz web ingresando a la dirección que hayamos configurado, por defecto en la versión CentOS 9 se accede mediante el puerto 9090 debido a la configuración de Cockpit [4], proyecto en el que se basa la interfaz web como alternativa a la utilización del puerto 80 o 443 (fig.4).

Como usuario root es posible operar cualquier configuración del servidor FreeIPA.
Al iniciar sesión como usuario root, haremos hincapié en la administración de ciertas configuraciones que este tiene acceso a realizar (fig.5).

Luego de iniciar sesión podemos obtener información sobre el sistema en el que hayamos instalado el servidor (fig.6 y fig.7).

Una de las funcionalidades disponibles es realizar conexión con un servicio ISCSI que tengamos disponible para conectar con volúmenes de disco (fig.8).

En la opción de menú Servicios, es posible observar los servicios que se encuentren en ejecución, con la disposición de información detallada (fig.9). Si se selecciona un servicio que se encuentra activo, se observa la configuración e información, es posible cambiar su estado a desactivado si fuese necesario.

Desde la misma interfaz es posible crear cuentas de acceso, para lo cual se debe definir la contraseña deseada, si debe cambiarla al ingresar o si se debe iniciar sesión solo una vez (fig.10). Por defecto, las cuentas creadas tienen asignado un nivel de permisos básicos, es posible cambiar el rol de las cuentas, lo que implica que un usuario obtenga más poder sobre el servidor, es decir, podrá llevar a cabo cambios a configuraciones, acceso cierta información delicada de archivos y procedimientos. En esta oportunidad mostraremos cómo el usuario root cambia el rol de la cuenta “administrador”, ahora este pasará a obtener privilegios como administrador del servidor (fig.11).

Desde la línea de comandos, es posible el inicio de sesión para obtener una clave Kerberos a los efectos de interactuar con los sistemas de información y de almacenamiento (fig.12).

4 Implementación de un caso de uso en JAVA

La manera de testear el funcionamiento de la infraestructura instalada, la vamos a desarrollar mediante una aplicación de java y la utilización de sockets plasmamos como un usuario que pertenece a un grupo determinado tendrá ciertas funcionalidades disponibles y en su defecto no, dicha aplicación es un servidor de socket de consultas que verifica con el servidor freeIPA los grupos a los que pertenece el usuario y habilitar ciertas funciones, en este se podrá consultar por diversas características del sistema, así como la versión del sistema operativo, versión del lenguaje de programación, etc. De tal forma, es posible restringir el acceso a las funciones para cada usuario mediante un chequeo de los grupos a los que pertenece y así poder implementar un servicio de tipo RBAC otorgando de esta manera mayor seguridad al ambiente del sistema operativo.

En este caso, la ejecución de la aplicación la realiza el usuario “programador2”, podemos observar como mediante FreeIPA la aplicación obtiene información de dicho usuario, de esta forma se verifica a qué grupo/s pertenece este. Los usuarios pertenecientes al grupo “Programadores” tienen el acceso a realizar 5 consultas y así obtener determinada información (fig.13). A continuación, se lleva a cabo la ejecución de la aplicación con una cuenta diferente, perteneciente al grupo “Administradores”. Es posible observar que la interacción define respuestas diferentes a las consultas realizadas en el punto anterior (fig.14).

Como último paso, mediante la verificación que se obtuvo del servidor FreeIPA, la ejecución la lleva a cabo un usuario invitado, es decir, un usuario que no pertenece a ningún grupo, por lo tanto solo tiene disponible realizar una consulta. Teniendo en cuenta que es un usuario con rol de invitado, de ser necesario, se pueden indicar acciones disponibles para este (fig.15).

5 Conclusión y posibles mejoras

Existen muchos recursos disponibles en Internet y el gran abanico de posibilidades puede abrumar no solo a los nuevos usuarios de la comunidad sino también a las entidades empresariales que deben tomar una decisión de confianza en sobre quién confiar sus recursos. El ofrecimiento de una alternativa opensource para este tipo de problemas se puede ver en los proyectos de integración.

El uso de software libre permite el ahorro de costos asociados con licencias de software. Por este motivo recomendamos el uso no solo de tecnologías de integración sino también de software libre, si bien en ocasiones puede que no resulte la mejor elección en ciertos entornos, el informático deberá conocer dichas alternativas para

poder valorar todas las opciones disponibles y seleccionar la más adecuada en cada caso.

El objetivo de este trabajo es incentivar sobre el cambio de tecnologías, y que promuevan el uso de software libre como Linux como soporte en los centros de datos, sean públicos y privados.

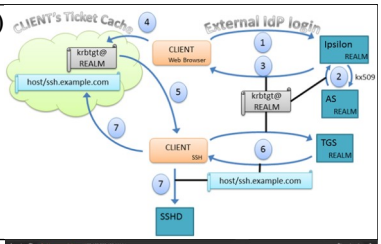
6 Referencias

[1] Red Hat Integration. https://www.redhat.com/es/products/integration
[2] https://es.wikipedia.org/wiki/Software_libre
[3] https://www.freeipa.org/page/Main_Page
[4] https://cockpit-project.org/
[5] https://www.server-world.info/en/note?os=CentOS_Stream_9&p=freeipa&f=1
[6] https://administradordesistema.files.wordpress.com/2018/08/freeipa.pdf

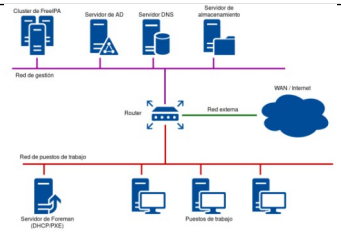
Lista de Figuras.

(fig.1)

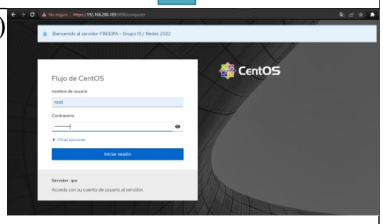
(fig.2)



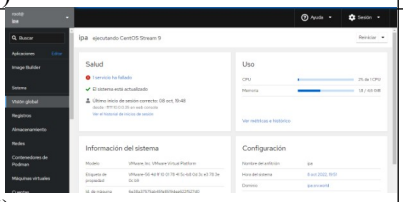
(fig.3)



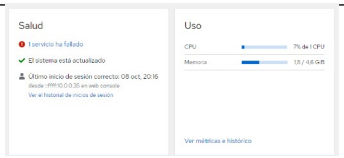
(fig.4)



(fig.5)

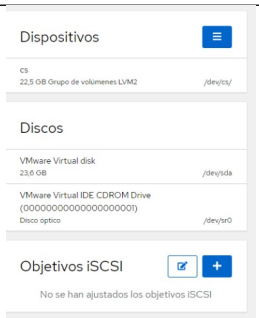


(fig.6)

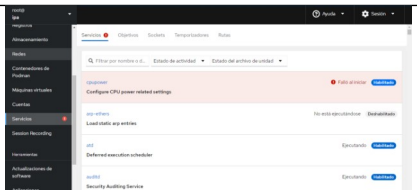


(fig.7)

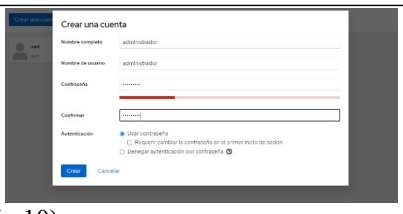
(fig.8)

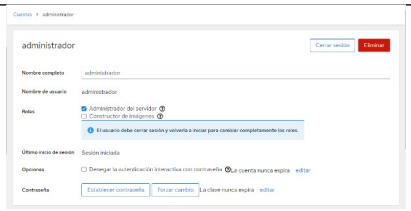


(fig.9)



(fig.10)





(fig.11)

(fig.12)

```
[root@ipa ~]# kinit admin
Password for admin@IPA.SRV.WORLD:
[root@ipa ~]# klist
Ticket cache: KCM:0
Default principal: admin@IPA.SRV.WORLD

Valid starting     Expires            Service principal
08/10/22 20:14:19  09/10/22 19:15:50  krbtgt/IPA.SRV.WORLD@IPA.SRV.WORLD
```

```
Ingresa la IP [localhost por defecto]:
Puerto [5050 por defecto]: 5052
Realizando Conexión -[programador2] => Conectado a :localhost


Bienvenido =>
[Servidor] => Usuario: programador2
 Rol de usuario: Programadores
[programador2] =>
[Servidor] =>
--------------------
Opciones Disponibles:
0 - Volver a mostrar opciones
1 - Nombre del OS
2 - Dirección del directorio JAVA
3 - Arquitectura del OS
4 - Versión del OS
5 - Versión JAVA
exit - Finalizar conexión
---------------

[programador2] =>
```

(fig.13)

```
Ingresa la IP [localhost por defecto]:
Puerto [5050 por defecto]: 5051
Realizando Conexión -[administrador1] => Conectado a :localhost


Bienvenido =>
[Servidor] => Usuario: administrador1
 Rol de usuario: Administradores
[administrador1] =>
[Servidor] =>
--------------------
Opciones Disponibles:
0 - Volver a mostrar opciones
1 - Nombre del OS
2 - Nombre del User
3 - Dirección del directorio JAVA
4 - Arquitectura del OS
5 - Dirección HOME del USER
6 - Dirección del ejecutable
7 - Versión del OS
8 - Versión JAVA
9 - Nombre del proveedor JAVA
exit - Finalizar conexión
---------------

[administrador1] =>
```

(fig.14)

```
Ingresa la IP [localhost por defecto]:
Puerto [5050 por defecto]: 5054
Realizando Conexión -[usercomun] => Conectado a :localhost


Bienvenido =>
[Servidor] => Usuario: usercomun
 Rol de usuario: Invitado
[usercomun] =>
[Servidor] =>
--------------------
Opciones Disponibles:
0 - Volver a mostrar opciones
1 - Nombre del sistema operativo
exit - Finalizar conexión
--------------------

[usercomun] =>
```

(fig.15)