Identity Console

Guía de instalación

Septiembre de 2022

Información legal

Para obtener información acerca de la información legal, las marcas comerciales, las renuncias de responsabilidad, las garantías, la exportación y otras restricciones de uso, los derechos del gobierno estadounidense, la directiva de patentes y el cumplimiento de la norma FIPS, consulte el sitio https://www.netiq.com/es-es/company/legal.

Tabla de contenido

Acerca de este libro y la biblioteca

La *Guía de instalación de Identity Console* proporciona información sobre cómo instalar y gestionar el producto NetIQ Identity Console (Identity Console). En este documento se define la terminología y se presentan escenarios de implementación.

A quién va dirigida

Esta guía está dirigida a administradores de red.

Otra información de la biblioteca

La biblioteca ofrece los siguientes recursos informativos:

Guía de instalación

Describe el procedimiento de instalación y actualización de Identity Console. Este libro está dirigido a administradores de red.

Acerca de NetIQ Corporation

Somos una empresa mundial de software empresarial, centrada en resolver los tres principales desafíos de su entorno, a saber, cambios, complejidad y riesgo, y en cómo podemos ayudarle a controlarlos.

Nuestro punto de vista

La adaptación a los cambios y la gestión de la complejidad y los riesgos no son conceptos nuevos

De hecho, de todos los desafíos a los que se enfrenta, quizá sean estas las variables más destacadas que le deniegan el control necesario para poder medir, supervisar y gestionar de forma segura sus entornos físico, virtual y de cloud computing.

Activación de servicios esenciales para el negocio de forma más rápida y eficiente

Creemos que la única forma de hacer posible una prestación de servicios más puntual y económica es dotar a las organizaciones de TI del mayor control posible. La presión continua de los cambios y la complejidad seguirá aumentando a medida que las organizaciones sigan creciendo y las tecnologías necesarias para gestionarlas se hagan intrínsecamente más complejas.

Nuestra filosofía

Vender soluciones inteligentes, no solo software

Para poder ofrecer un control fiable, debemos entender primero los escenarios reales en los que —día a día— operan las organizaciones de TI como la suya. Esa es la única forma de desarrollar soluciones de TI prácticas e inteligentes que proporcionen resultados mensurables con una eficacia demostrada. Y eso es mucho más satisfactorio que vender simplemente software.

Fomentar su éxito es nuestra pasión

Ayudarle a alcanzar el éxito es el objetivo primordial de nuestro trabajo. Desde la concepción a la implantación, sabemos que usted necesita soluciones de TI que funcionen bien y se integren a la perfección con su inversión existente; necesita asistencia continua y formación posterior a la implantación; y, para variar, también necesita trabajar con alguien que le facilite las cosas. En definitiva, su éxito será también el nuestro.

Nuestras soluciones

- Control de identidad y acceso
- Gestión de acceso
- Gestión de la seguridad

- Gestión de sistemas y aplicaciones
- Gestión del trabajo
- Gestión de servicios

Cómo ponerse en contacto con la asistencia para ventas

Para cualquier pregunta sobre nuestros productos, precios y capacidades, póngase en contacto con su representante local. Si no puede contactar con su representante local, comuníquese con nuestro equipo de Asistencia para ventas.

Oficinas mundiales:	www.netiq.com/about_netiq/officelocations.asp
Estados Unidos y Canadá:	1-888-323-6768
Correo electrónico:	info@netiq.com
Sitio Web:	www.netiq.com

Cómo ponerse en contacto con el personal de asistencia técnica

Para obtener información sobre problemas con productos específicos, póngase en contacto con nuestro equipo de asistencia técnica.

Oficinas mundiales:	www.netiq.com/support/contactinfo.asp
Norteamérica y Sudamérica:	1-713-418-5555
Europa, Oriente Medio y África:	+353 (0) 91-782 677
Correo electrónico:	support@netiq.com
Sitio Web:	www.netiq.com/support

Cómo ponerse en contacto con la asistencia para documentación

Nuestro objetivo es proporcionar documentación que satisfaga sus necesidades. Si tiene sugerencias de mejoras, haga clic en **Add Comment** (Agregar comentario) en la parte de abajo de cualquier página de las versiones HTML de la documentación publicada en www.netiq.com/documentation. Si lo desea, también puede enviar un correo electrónico a Documentation-Feedback@netiq.com. Agradecemos sus comentarios y estamos deseando oír sus sugerencias.

Cómo ponerse en contacto con la comunidad de usuarios en línea

Qmunity, la comunidad de NetIQ en línea, es una red de colaboración que le pone en contacto con sus colegas y con otros expertos de NetIQ. Qmunity le ayuda a dominar los conocimientos que necesita para hacer realidad todo el potencial de su inversión en TI de la que depende, al proporcionarle información inmediata, enlaces útiles a recursos prácticos y acceso a los expertos de NetIQ. Para obtener más información, visite la página http://community.netiq.com.

1 Planificación de la instalación de Identity Console

En este capítulo, se indican los requisitos del sistema y los requisitos previos para la instalación de Identity Console. Como Identity Console se puede ejecutar como contenedor de Docker o aplicación independiente, consulte las secciones correspondientes de los requisitos del sistema y los requisitos previos para ambos tipos de instalación.

Nota: Identity Console es compatible con eDirectory 9.2.4 HF2, Identity Manager Engine 4.8.3 HF2 y sus respectivas versiones posteriores. Debe actualizar las instancias de eDirectory e Identity Manager Engine antes de utilizar Identity Console.

Requisitos del sistema y requisitos previos para la instalación de Docker

En esta sección, se explican los requisitos del sistema y los requisitos previos para instalar Identity Console como un contenedor de Docker.

Requisitos del sistema

Como Identity Console se puede ejecutar como un contenedor de Docker, para obtener más información sobre los requisitos del sistema y las plataformas compatibles para la instalación de Identity Console, consulte la documentación de Docker.

Requisitos previos

- ❒ Instale Docker 20.10.9-ce o posterior. Para obtener más información acerca de cómo instalar Docker, consulte Instalación de Docker.
- ❒ Debe obtener un certificado de servidor pkcs12 con la clave privada para cifrar o descifrar el intercambio de datos entre el servidor de Identity Console y el servidor back-end. Este certificado de servidor se utiliza para proteger la conexión HTTP. Puede utilizar certificados de

servidor generados por una CA externa. Para obtener más información, consulte Creating Server Certificate Objects (Creación de objetos Certificado de servidor). El certificado del servidor debería contener el nombre alternativo del sujeto con la dirección IP y el DNS del servidor. Una vez creado el objeto Certificado de servidor, debe exportarlo en formato .pfx.

- ❒ Debe obtener un certificado de CA para todos los árboles en formato `.pem` a fin de validar la firma de CA de los certificados de servidor obtenidos en el paso anterior. Este certificado de CA raíz también garantiza el establecimiento de una comunicación LDAP protegida entre el cliente y el servidor de Identity Console. Por ejemplo, puede obtener el certificado de CA de eDirectory (`SSCert.pem`) desde `/var/opt/novell/eDirectory/data/SSCert.pem`.
- ❒ (Opcional) Con One SSO Provider (OSP), puede habilitar la autenticación de firma única para los usuarios en el portal de Identity Console. Debe instalar OSP antes de la instalación de Identity Console. Para configurar OSP para Identity Console, siga las instrucciones que aparecen en la pantalla y proporcione los valores necesarios para los parámetros de configuración. Para obtener más información, consulte "Distribución del contenedor OSP" en la página 22. Para registrar Identity Console en el servidor OSP existente, debe añadir manualmente lo siguiente al archivo `ism-configuration.properties` en la carpeta `/opt/netiq/idm/apps/tomcat/conf/`:

  ```
  com.netiq.edirapi.clientID = identityconsole
  com.netiq.edirapi.redirect.url = https://<Identity Console Server IP>:<Identity Console Listener Port>/eDirAPI/v1/<eDirectory Tree Name>/authcoderedirect
  com.netiq.edirapi.logout.url = https://<Identity Console Server IP>:<Identity Console Listener Port>/eDirAPI/v1/<eDirectory Tree Name>/logoutredirect
  com.netiq.edirapi.logout.return-param-name = logoutURL
  com.netiq.edirapi.response-types = code,token
  com.netiq.edirapi.clientPass._attr_obscurity = NONE
  com.netiq.edirapi.clientPass = novell
  ```

  Nota: Con OSP, solo puede conectarse a un único árbol de eDirectory, ya que este no admite el uso de varios.

- ❒ Asegúrese de que dispone de una entrada de DNS adecuada disponible para el equipo host en `/etc/hosts` con un nombre completo del host.
- ❒ Si desea utilizar Identity Console en el navegador Edge, debe descargar la versión más reciente de Microsoft Edge para obtener todas las funciones.

Nota: Al utilizar Identity Console en Mozilla Firefox, la operación puede fallar con el mensaje de error `Origin Mismatch` (Error de coincidencia de origen). Para solucionar los problemas, realice los siguientes pasos:

1. Actualice Firefox a la versión más reciente.
2. Especifique `about:config` en el campo URL de Firefox y pulse Intro.
3. Busque Origen.
4. Haga doble clic en `network.http.SendOriginHeader` y cambie su valor a 1.

Configuración del entorno

Es posible que deba crear un archivo de configuración que contenga determinados parámetros. Si desea configurar Identity Console con OSP, debe especificar los parámetros específicos del OSP en el archivo de configuración. Por ejemplo, cree el archivo `edirapi.conf` que aparece a continuación con los parámetros de OSP:

Nota: Debe proporcionar el nombre del árbol de eDirectory en el campo `osp-redirect-url`.

```
listen = ":9000"
ldapserver = "2.168.1.1:636"
ldapuser = "cn=admin,ou=sa,o=system"
ldappassword = "novell"
pfxpassword = "novell"
ospmode = "true"
osp-token-endpoint = "https://10.10.10.10:8543/osp/a/idm/auth/oauth2/
getattributes"
osp-authorize-url = "https://10.10.10.10:8543/osp/a/idm/auth/oauth2/grant"
osp-logout-url = "https://10.10.10.10:8543/osp/a/idm/auth/app/logout"
osp-redirect-url = "https://10.10.10.10:9000/eDirAPI/v1/edirtree/
authcoderedirect"
osp-client-id = "identityconsole"
ospclientpass = "novell"
ospcert = "/etc/opt/novell/eDirAPI/cert/SSCert.pem"
bcert = "/etc/opt/novell/eDirAPI/cert/"
loglevel = "error"
check-origin = "true"
origin = "https://10.10.10.10:9000,https://192.168.1.1:8543"
```

Si desea configurar Identity Console sin OSP, cree un archivo de configuración, como se muestra a continuación, sin los parámetros de OSP:

```
listen = ":9000"
pfxpassword = "novell"
ospmode = "false"
bcert = "/etc/opt/novell/eDirAPI/cert/"
```

Nota: Si desea configurar Identity Console con varios árboles de eDirectory, puede omitir los parámetros "*ldapserver*", "*ldapuser*" y "*ldappassword*", y crear el archivo de configuración.

Tabla 1-1 *Descripción de los parámetros de configuración del archivo de configuración*

Parámetros de configuración	Descripción
listen	Especifique 9000 como el puerto de escucha del servidor de Identity Console en el contenedor.
ldapserver	Especifique la dirección IP del servidor host de eDirectory y el número de puerto.

Parámetros de configuración	Descripción
ldapuser	Especifique el nombre del usuario de eDirectory. Este parámetro se utiliza como credencial para iniciar las llamadas de LDAP a eDirectory mediante el control de autorización proxy en el caso de la entrada a OSP. El usuario LDAP debe tener derechos de supervisor en el árbol de eDirectory.
ldappassword	Especifique la contraseña del usuario LDAP.
pfxpassword	Especifique la contraseña del archivo de certificado de servidor pkcs12.
ospmode	Especifique `true` (verdadero) para integrar OSP con Identity Console. Si establece el valor en `false` (falso), Identity Console utilizará la entrada LDAP.
osp-token-endpoint	Esta dirección URL permite obtener determinados atributos del servidor OSP a fin de verificar la validez del testigo de autenticación..
osp-authorize-url	El usuario utiliza esta dirección URL para proporcionar las credenciales a fin de obtener un testigo de autenticación..
osp-logout-url	Utilice esta dirección URL para finalizar la sesión entre el usuario y el servidor OSP..
osp-redirect-url	El servidor OSP redirige al usuario a esta dirección URL después de otorgar el testigo de autenticación.. **Nota:** Asegúrese de especificar el nombre del árbol de eDirectory en minúsculas al configurar Identity Console. Si el nombre del árbol no se especifica en minúsculas, es posible que falle la entrada al servidor de Identity Console.
osp-client-id	Especifique el ID de cliente de OSP que se proporcionó durante el registro de Identity Console en OSP..
ospclientpass	Especifique la contraseña de cliente de OSP que se proporcionó durante el registro de Identity Console en OSP..
ospcert	Especifique la ubicación del certificado de CA del servidor OSP..
bcert	Especifique la ubicación del certificado de CA de Identity Console.
loglevel	Especifique los niveles de registro que desea incluir en el archivo de registro. Este parámetro se puede definir como "fatal", "error", "warn" o "info".

Parámetros de configuración	Descripción
check-origin	Si se ha establecido en `true` (verdadero), el servidor de Identity Console compara el valor de origen de las peticiones. Las opciones disponibles son `true` (verdadero) o `false` (falso). El parámetro *origin* (origen) es obligatorio, incluso aunque el valor del parámetro *check-origin* se haya establecido en `false` (falso) cuando se utiliza la configuración de DNS.
origin	Identity Console compara el valor de origen de las peticiones con los valores especificados en este campo. **Nota:** A partir de Identity Console 1.4, este parámetro es independiente del parámetro *check-origin* y es obligatorio si se utiliza la configuración de DNS.
maxclients	Número máximo de clientes simultáneos que pueden acceder a `IDConsole`. Cualquier cliente adicional que supere este límite tendrá que esperar en la cola.

Nota

- El parámetro de configuración `ospmode` solo debe utilizarse si tiene previsto integrar OSP junto con Identity Console.
- Si Identity Applications (Identity Apps) se ha establecido en modo de clúster en la configuración de Identity Manager, debe proporcionar el nombre DNS del servidor del equilibrador de carga en los campos `osp-token-endpoint`, `osp-authorize-url` y `osp-logout-url` del archivo de configuración. Si especifica los detalles del servidor OSP en estos campos, fallará la entrada a Identity Console.
- Si Identity Console se ha configurado con la misma instancia de OSP que Identity Apps e Identity Reporting, la entrada única (servicio de autenticación) se aplicará cuando entre al portal de Identity Console.
- La dirección URL HTTPS de OSP debe validarse con certificados que contengan una clave de 2048 bits o superior con Identity Console 1.4 en adelante.
- Si desea restringir el acceso al portal de Identity Console desde diferentes dominios, establezca el parámetro `samesitecookie` en `strict`. Si desea permitir el acceso al portal de Identity Console desde diferentes dominios, establezca el parámetro `samesitecookie` en `lax`. Si no se especifica el parámetro durante la configuración, los ajustes del navegador se respetarán por defecto.

Una vez que haya completado la creación del archivo de configuración, continúe con la distribución del contenedor. Para obtener más información, consulte "Distribución de Identity Console como contenedor de Docker" en la página 22.

Requisitos del sistema y requisitos previos para la instalación independiente (sin Docker)

Requisitos del sistema

En esta sección, se describen los requisitos del sistema y los requisitos previos para la instalación independiente de Identity Console.

Categoría	Requisito mínimo
Procesador	1,4 GHz, 64 bits
Memoria	2 GB
Espacio en disco	200 MB en Linux
Navegador compatible	◆ Versión más reciente de **Microsoft Edge** ◆ Versión más reciente de **Google Chrome** ◆ Versión más reciente de **Mozilla Firefox** **Nota:** Al utilizar Identity Console en Mozilla Firefox, la operación puede fallar con el mensaje de error `Origin Mismatch` (Error de coincidencia de origen). Para solucionar los problemas, realice los siguientes pasos: **1** Actualice Firefox a la versión más reciente. **2** Especifique `about:config` en el campo URL de Firefox y pulse Intro. **3** Busque Origen. **4** Haga doble clic en `network.http.SendOriginHeader` y cambie su valor a 1.
Sistema operativo compatible	◆ **Certificado**: ◆ SUSE Linux Enterprise Server (SLES) 15 SP1, SP2 y SP3 ◆ SUSE Linux Enterprise Server (SLES) 12 SP1, SP2, SP3, SP4 y SP5 ◆ Red Hat Enterprise Linux (RHEL) 7.8, 7.9, 8.0, 8.1, 8.2, 8.3, 8.4 y 8.5 ◆ OpenSUSE 15.1 y 15.2 ◆ **Admitido:** admitido en versiones posteriores de los paquetes de soporte de los sistemas operativos certificados anteriormente.

Categoría	Requisito mínimo
Certificados	◆ Debe obtener un certificado de servidor pkcs12 con la clave privada para cifrar o descifrar el intercambio de datos entre el cliente y el servidor de Identity Console. Este certificado de servidor se utiliza para proteger la conexión HTTP. Puede utilizar certificados de servidor generados por una CA externa. Para obtener más información, consulte Creating Server Certificate Objects (Creación de objetos Certificado de servidor). El certificado del servidor debería contener el nombre alternativo del sujeto con la dirección IP y el DNS del servidor. Una vez creado el objeto Certificado de servidor, debe exportarlo en formato .pfx. ◆ Debe obtener un certificado de CA para todos los árboles en formato `.pem` a fin de validar la firma de CA de los certificados de servidor obtenidos en el paso anterior. Este certificado de CA raíz también garantiza el establecimiento de una comunicación LDAP protegida entre el cliente y el servidor de Identity Console. Por ejemplo, puede obtener el certificado de CA de eDirectory (`SSCert.pem`) desde `/var/opt/novell/eDirectory/data/SSCert.pem`.

Cuando esté preparado, continúe con la instalación de Identity Console. Para obtener más información, consulte “Distribución de una instancia independiente de Identity Console” en la página 27.

(Opcional) Requisito previo para la configuración de OSP

Con One SSO Provider (OSP), puede habilitar la autenticación de firma única para los usuarios en el portal de Identity Console. Debe instalar OSP antes de la instalación de Identity Console. Para configurar OSP para Identity Console, siga las instrucciones que aparecen en la pantalla y proporcione los valores necesarios para los parámetros de configuración. Para obtener más información, consulte “Distribución del contenedor OSP” en la página 22. Para registrar Identity Console en el servidor OSP existente, debe añadir manualmente lo siguiente al archivo `ism-configuration.properties` en la carpeta `/opt/netiq/idm/apps/tomcat/conf/`:

```
com.netiq.edirapi.clientID = identityconsole
com.netiq.edirapi.redirect.url = https://<Identity Console Server
IP>:<Identity Console Listener Port>/eDirAPI/v1/<eDirectory Tree Name>/
authcoderedirect
com.netiq.edirapi.logout.url = https://<Identity Console Server
IP>:<Identity Console Listener Port>/eDirAPI/v1/<eDirectory Tree Name>/
logoutredirect
com.netiq.edirapi.logout.return-param-name = logoutURL
com.netiq.edirapi.response-types = code,token
com.netiq.edirapi.clientPass._attr_obscurity = NONE
com.netiq.edirapi.clientPass = novell
```

Nota

- Si va a instalar OSP por primera vez, especifique y para **Configure OSP with eDir API** (Configurar OSP con la API de eDir) y siga las instrucciones que aparecen en la pantalla para registrar Identity Console en OSP.
- Asegúrese de especificar el nombre del árbol de eDirectory en minúsculas al configurar Identity Console. Si el nombre del árbol no se especifica en minúsculas, es posible que falle la entrada al servidor de Identity Console.
- Con OSP, solo puede conectarse a un único árbol de eDirectory, ya que este no admite el uso de varios.

Requisitos del sistema y previos para la estación de trabajo

- "Requisitos del sistema" en la página 18

Requisitos del sistema

En esta sección, se describen los requisitos del sistema y previos para la ejecución de una estación de trabajo de Identity Console.

Categoría	Requisito mínimo
Procesador	1.5 GHz, 64 bits
Memoria	2 GB
Espacio en disco	1 GB en Windows

Sistema operativo compatible	• **Certificado**:
	  • Windows Server 2016.
	  • Windows Server 2019
	  • Windows Server 2022
	  • Windows 10
	  • Windows 11

Categoría	Requisito mínimo
Certificados	◆ Debe obtener un certificado de servidor en formato pfx para intercambiar datos entre el cliente Identity Console y el servidor REST. Este certificado de servidor debe denominarse siempre keys.pfx. Para obtener más información, consulte Creating Server Certificate Objects (https://www.netiq.com/documentation/edirectory-92/edir_admin/data/b1j4tpo3.html#b1j4u0cm) (Creación de objetos Certificado de servidor). ◆ Debe obtener un certificado de CA para todos los árboles en formato `.pem` a fin de validar la firma de CA de los certificados de servidor obtenidos en el paso anterior. Este certificado de CA raíz también garantiza el establecimiento de una comunicación LDAP protegida entre el cliente y el servidor de Identity Console. Por ejemplo, puede obtener el certificado de CA de eDirectory para Linux en `SSCert.pem` desde `/var/opt/novell/eDirectory/data/SSCert.pem`. Obtenga el certificado de CA de eDirectory `SSCert.pem` para Windows desde `<Ubicación de instalación de eDirectory>\NetIQ\eDirectory\DIBFiles\CertServ\SSCert.pem`.

Cuando esté preparado, continúe con la distribución de Identity Console. Para obtener más información, consulte “Identity Console en Windows como estación de trabajo” en la página 29.

Verificación de firma de RPM

Lleve a cabo los pasos siguientes para realizar la verificación de firma de RPM:

1. Acceda a la carpeta en la que se ha extraído la compilación.

   Por ejemplo: `<ubicación sin archivo tar de Identity Console>/IdentityConsole_150_Linux/license/MicroFocusGPGPackageSign.pub`.

2. Ejecute el siguiente comando para importar la clave pública:

   ```
   rpm --import MicroFocusGPGPackageSign.pub
   ```

3. (Opcional) Ejecute el siguiente comando para verificar la firma de RPM: `rpm --checksig -v <Nombre de RPM>`

   Por ejemplo:

   ```
   rpm --checksig -v identityconsole-1.5.0000.x86_64.rpm
   identityconsole-1.5.0000.x86_64.rpm:
   Header V4 RSA/SHA256 Signature, OK, key ID 786ec7c0: OK
   ```

```
Header SHA1 digest: OK
Header SHA256 digest: OK
Payload SHA256 digest: OK
V4 RSA/SHA256 Signature, key ID 786ec7c0: OK
MD5 digest: OK
```

2 Distribución de Identity Console

En este capítulo, se describe el proceso de distribución de Identity Console junto con las recomendaciones de seguridad. Para preparar la distribución, revise los requisitos previos y los requisitos del sistema en el Capítulo 1, “Planificación de la instalación de Identity Console”, en la página 11.

Recomendaciones de seguridad

- Los contenedores de Docker no presentan ninguna restricción de recursos por defecto. Esto brinda a todos los contenedores acceso a todos los recursos de CPU y memoria proporcionados por el kernel del host. También debe asegurarse de que un único contenedor en ejecución no consuma más recursos y prive de ellos a otros contenedores en ejecución. Para ello, establezca límites en cuanto a la cantidad de recursos que puede utilizar un contenedor.
  - El contenedor de Docker debe garantizar que se haya aplicado un límite rígido para la memoria utilizada por el contenedor mediante el uso del indicador `--memory` del comando de ejecución de Docker.
  - El contenedor de Docker debe garantizar que se haya aplicado un límite a la cantidad de CPU utilizada por un contenedor en ejecución mediante el uso del indicador `--cpuset-cpus` del comando de ejecución de Docker.
- `--pids-limit` debería establecerse en 300 para restringir los subprocesos del kernel generados en el contenedor en cualquier momento específico. Esto impide los ataques de DoS.
- Debe establecer la directiva de reinicio del contenedor en caso de error en 5 mediante el indicador `--restart` del comando de ejecución de Docker.
- Solo debe utilizar el contenedor cuando el estado de actividad se muestre como **Healthy** (En buen estado) después de que aparezca el contenedor. Para comprobar el estado de actividad del contenedor, ejecute el comando siguiente:

```
docker ps <container_name/ID>
```

- El contenedor de Docker siempre se iniciará como un usuario no root (nds). Como medida de seguridad adicional, habilite la nueva asignación de espacio de nombres de usuario en el daemon para evitar ataques de derivación de privilegios desde el contenedor. Para obtener más información sobre la nueva asignación de espacio de nombres de usuario, consulte Isolate containers with a user namespace (Aislar contenedores con un espacio de nombres de usuario).

Distribución de Identity Console como contenedor de Docker

En esta sección se incluyen los procedimientos siguientes:

- "Distribución del contenedor OSP" en la página 22
- "Distribución de Identity Console como un contenedor de Docker" en la página 24
- "Varios árboles con Identity Console como Docker" en la página 26

Distribución del contenedor OSP

Lleve a cabo los pasos siguientes para distribuir el contenedor OSP:

1. Entre a Software License and Download (https://sld.microfocus.com/) y desplácese a la página Descargas de software.
2. Seleccione lo siguiente:
   - Producto: eDirectory
   - Nombre del producto: eDirectory per User Sub SW E-LTU
   - Versión: 9.2
3. Descargue el archivo: `IdentityConsole_<versión>_Containers_tar.zip`.
4. Extraiga el archivo descargado en una carpeta.
5. Modifique el archivo de propiedades silencioso según sus requisitos. A continuación, se muestra un ejemplo de archivo de propiedades silencioso:

```
# Silent file for osp with edirapi
## Static contents Do not edit - starts
INSTALL_OSP=true
DOCKER_CONTAINER=y
EDIRAPI_PROMPT_NEEDED=y
UA_PROMPT_NEEDED=n
SSPR_PROMPT_NEEDED=n
RPT_PROMPT_NEEDED=n
CUSTOM_OSP_CERTIFICATE=y
## Static contents Do not edit - ends

# OSP Details
SSO_SERVER_HOST=osp.example.com
SSO_SERVER_SSL_PORT=8543
OSP_COMM_TOMCAT_KEYSTORE_FILE=/config/tomcat.ks
OSP_COMM_TOMCAT_KEYSTORE_PWD=novell
SSO_SERVICE_PWD=novell
```

```
OSP_KEYSTORE_PWD=novell
IDM_KEYSTORE_PWD=novell
OSP_CUSTOM_NAME="Identity Console"
USER_CONTAINER="o=novell"
ADMIN_CONTAINER="o=novell"

# IDConsole Details
IDCONSOLE_HOST=192.168.1.1
IDCONSOLE_PORT=9000
EDIRAPI_TREENAME=ed913

#If ENABLE_CUSTOM_CONTAINER_CREATION is set to y
#ie., when you have user and admin container different from o=data
#    and they need to be created in eDir
#then CUSTOM_CONTAINER_LDIF_PATH should be entered as well
ENABLE_CUSTOM_CONTAINER_CREATION=n
#ENABLE_CUSTOM_CONTAINER_CREATION=y
#CUSTOM_CONTAINER_LDIF_PATH=/config/custom-osp.ldif

# eDir Details
ID_VAULT_HOST=192.168.1.1
ID_VAULT_LDAPS_PORT=636
ID_VAULT_ADMIN_LDAP="cn=admin,o=novell"
ID_VAULT_PASSWORD=novell
```

Nota: Para evitar restricciones de espacio mientras utiliza el archivo de propiedades silenciosas (texto DOS), debe convertir el archivo de texto DOS al formato UNIX mediante la herramienta dos2unix. Ejecute el siguiente comando para convertir un archivo de texto de finales de línea DOS a finales de línea Unix:

```
dos2unix nombre_archivo
```

Por ejemplo:

```
dos2unix archivo_ejemplo
```

---

6 Genere un certificado de servidor (`cert.der`) mediante iManager e impórtelo mediante el almacén de claves (`tomcat.ks`). Copie el archivo de propiedades silencioso y el almacén de claves (`tomcat.ks`) en cualquier directorio. Por ejemplo, `/data`. Realice los siguientes pasos para crear un certificado de servidor e importarlo en el almacén de claves:

6a Ejecute el siguiente comando para crear un almacén de claves (`tomcat.ks`). Genere la clave y asegúrese de que el nombre CN o el nombre completo de host del equipo sea la dirección IP.

```
keytool -genkey -alias osp -keyalg RSA -storetype pkcs12 -keystore /
opt/certs/tomcat.ks -validity 3650 -keysize 2048 -dname "CN=blr-
osp48-demo.labs.blr.novell.com" -keypass novell -storepass novell
```

6b Ejecute el siguiente comando para crear una petición de firma de certificado. Por ejemplo, `cert.csr`.

```
keytool -certreq -v -alias osp -file /opt/certs/cert.csr -keypass
novell -keystore /opt/certs/tomcat.ks -storepass novell
```

6c Transfiera `cert.csr` a iManager y obtenga el certificado del servidor `osp.der`. Asegúrese de que selecciona el tipo de clave como Personalizado; las opciones de uso de la clave como Cifrado de datos, Cifrado de clave y Firma digital, y el campo Nombre(s) alternativo(s) del sujeto del certificado para que contenga la dirección IP o el nombre de host del servidor OSP. Para obtener más información, consulte Creación de un objeto Certificado de servidor.

6d Ejecute los siguientes comandos para importar el certificado de CA (`SSCert.der`) y el certificado de servidor (`cert.der`) en el almacén de claves `tomcat.ks`.

```
keytool -import -trustcacerts -alias root -keystore /opt/certs/
tomcat.ks -file /opt/certs/SSCert.der -storepass novell -noprompt
```

```
keytool -import -alias osp -keystore /opt/certs/tomcat.ks -file /
opt/certs/cert.der -storepass novell -noprompt
```

7 Ejecute el siguiente comando para cargar la imagen de OSP:

```
docker load --input osp.tar.gz
```

8 Distribuya el contenedor mediante el siguiente comando:

```
docker run -d --name OSP_Container --network=host -e
SILENT_INSTALL_FILE=/config/silent.properties -v /data:/config
osp:<version>
```

Por ejemplo:

```
docker run -d --name OSP_Container --network=host -e
SILENT_INSTALL_FILE=/config/silent.properties -v /data:/config
osp:6.3.9
```

Distribución de Identity Console como un contenedor de Docker

En esta sección, se describe el procedimiento de distribución de Identity Console como contenedor de Docker:

Nota: Los parámetros de configuración, los valores de muestra y los ejemplos mencionados en este procedimiento se proporcionan solo como referencia. Debe asegurarse de no utilizarlos directamente en el entorno de producción.

---

1 Entre a SLD, Software License and Download (https://sld.microfocus.com/), y desplácese a la página Descargas de software.

2 Seleccione lo siguiente:

- Producto: eDirectory
- Nombre del producto: eDirectory per User Sub SW E-LTU
- Versión: 9.2

3 Descargue el archivo: `IdentityConsole_<versión>_Container.tar.zip`.

4 La imagen debe cargarse en el registro local de Docker. Extraiga y cargue el archivo `IdentityConsole_<versión>_Containers.tar.gz` mediante los siguientes comandos:

```
tar -xvf IdentityConsole_<version>_Containers.tar.gz
```

```
docker load --input identityconsole.tar.gz
```

5 Cree el contenedor de Docker para Identity Console mediante el siguiente comando:

```
docker create --name <identityconsole-container-name> --env
ACCEPT_EULA=Y --network=<network-type> --volume <volume-name>:/config/
identityconsole:<version>
```

Por ejemplo:

```
docker create --name identityconsole-container --env ACCEPT_EULA=Y --
network=host --volume IDConsole-volume:/config/
identityconsole:1.5.0.0000.
```

Nota

- Puede aceptar el CLUF. Para ello, establezca la variable de entorno ACCEPT_EULA en Y. También puede aceptar el CLUF en las instrucciones en pantalla mientras inicia el contenedor utilizando la opción `-it` del comando create de Docker y acceder al modo interactivo.
- El parámetro `--volume` del comando anterior creará un volumen para almacenar los datos de configuración y registro. En este caso, hemos creado un volumen de ejemplo denominado `IDConsole-volume`.

---

6 Copie el archivo de certificado de servidor del sistema de archivos local en el contenedor como `/etc/opt/novell/eDirAPI/cert/keys.pfx` mediante el siguiente comando. Para obtener más información sobre cómo crear el certificado de servidor, consulte "Requisitos previos" en la página 11:

```
docker cp <absolute path of server certificate file> <identityconsole-
container-name>:/etc/opt/novell/eDirAPI/cert/keys.pfx
```

Por ejemplo:

```
docker cp /home/user/keys.pfx identityconsole-container:/etc/opt/
novell/eDirAPI/cert/keys.pfx
```

Al conectarse a varios árboles de eDirectory, debe asegurarse de obtener al menos un certificado de servidor keys.pfx para todos los árboles conectados.

7 Copie el archivo de certificado de CA (`.pem`) del sistema de archivos local en el contenedor como `/etc/opt/novell/eDirAPI/cert/SSCert.pem` mediante el siguiente comando. Para obtener más información sobre cómo obtener el certificado de CA, consulte "Requisitos previos" en la página 11:

```
docker cp <absolute path of CA certificate file> <identityconsole-
container-name>:/etc/opt/novell/eDirAPI/cert/SSCert.pem
```

Por ejemplo:

```
docker cp /home/user/SSCert.pem identityconsole-container:/etc/opt/
novell/eDirAPI/cert/SSCert.pem
```

Si el usuario necesita conectarse a varios árboles de eDirectory, consulte la sección: "Varios árboles con Identity Console como Docker" en la página 26

8 Modifique el archivo de configuración (`edirapi.conf`) en función de sus requisitos y cópielo del sistema de archivos local al contenedor como `/etc/opt/novell/eDirAPI/conf/edirapi.conf` mediante el siguiente comando:

```
docker cp <absolute path of configuration file> <identityconsole-container-name>:/etc/opt/novell/eDirAPI/conf/edirapi.conf
```

Por ejemplo:

```
docker cp /home/user/edirapi.conf identityconsole-container:/etc/opt/novell/eDirAPI/conf/edirapi.conf
```

9 Inicie el contenedor de Docker mediante el siguiente comando:

```
docker start <identityconsole-container-name>
```

Por ejemplo:

```
docker start identityconsole-container
```

Nota: Puede encontrar los siguientes archivos de registro en el directorio `/var/lib/docker/volumes/<nombre_volumen>/_data/eDirAPI/var/log`:

- edirapi.log : se utiliza para registrar diferentes eventos en edirapi y los problemas de depuración.
- edirapi_audit.log: se utiliza para registrar eventos de auditoría de edirapi. Los registros siguen el formato de auditoría de CEF.
- container-startup.log: se utiliza para capturar registros de instalación del contenedor de Docker para Identity Console.

Varios árboles con Identity Console como Docker

Identity Console permite al usuario conectarse a varios árboles mediante la obtención de un certificado de CA individual para cada árbol.

Por ejemplo, si se conecta a tres árboles de eDirectory, debe copiar los tres certificados de CA en el contenedor de Docker:

```
docker cp /home/user/SSCert1.pem identityconsole-container:/etc/opt/novell/eDirAPI/cert/SSCert.pem

docker cp /home/user/SSCert2.pem identityconsole-container:/etc/opt/novell/eDirAPI/cert/SSCert1.pem

docker cp /home/user/SSCert3.pem identityconsole-container:/etc/opt/novell/eDirAPI/cert/SSCert2.pem
```

Ejecute los siguientes comandos para reiniciar Identity Console:

```
docker restart <identityconsole-container-name>
```

Distribución de una instancia independiente de Identity Console

Distribución de una instancia independiente de Identity Console (sin Docker)

En esta sección, se explica el procedimiento para distribuir una instancia independiente de Identity Console:

1. Entre a SLD, Software License and Download (https://sld.microfocus.com/), y desplácese a la página Descargas de software.
2. Seleccione lo siguiente:
   - Producto: eDirectory
   - Nombre del producto: eDirectory per User Sub SW E-LTU
   - Versión: 9.2
3. Descargue la compilación más reciente de Identity Console.
4. Extraiga el archivo descargado en una carpeta.
5. Abra un shell y desplácese a la carpeta en la que ha extraído la versión de Identity Console.
6. Ejecute el siguiente comando mientras está conectado como usuario root o equivalente:

   ```
   ./identityconsole_install
   ```

7. Lea la introducción y haga clic en **ENTER**.
8. Haga clic en S para aceptar el Acuerdo de licencia. Esta acción instalará todos los RPM necesarios en el sistema.
9. Introduzca el nombre de host (nombre completo) del servidor de Identity Console o su dirección IP.
10. Introduzca el número de puerto de escucha de Identity Console. El valor por defecto es 9000.
11. Introduzca la opción para integrar OSP con Identity Console o Identity Console para utilizar la entrada LDAP.
12. Si desea integrar OSP con Identity Console:
    1. Introduzca el nombre de dominio o la dirección IP del servidor del repositorio seguro de identidades/eDirectory con el número de puerto LDAPS.

       Por ejemplo:

       192.168.1.1:636
    2. Introduzca el nombre del repositorio seguro de identidades/eDirectory.

       Por ejemplo:

       cn=admin,ou=org_unit,o=org
    3. Introduzca la contraseña del repositorio seguro de identidades/eDirectory.

4. Vuelva a introducir la contraseña del depósito de identidades/eDirectory para confirmarla.
5. Introduzca el nombre de dominio o la dirección IP del servidor OSP con el número de puerto SSL del servidor SSO.
6. Introduzca el ID del cliente OSP.
7. Introduzca la contraseña del cliente OSP.
8. Introduzca el nombre de árbol del repositorio seguro de identidades/eDirectory.

13 Introduzca la vía de los certificados raíz de confianza (SSCert.pem), incluida la carpeta.

Por ejemplo:

```
/home/Identity_Console/certs
```

Nota: El usuario debe asegurarse de no crear un subdirectorio dentro de la carpeta cert.

14 Introduzca la vía del certificado del servidor (keys.pfx), incluido el nombre de archivo.

Por ejemplo:

```
/home/Identity_Console/keys.pfx
```

15 Introduzca la contraseña del certificado del servidor. Para confirmar que ha introducido correctamente la contraseña, vuelva a introducir la contraseña del certificado de servidor. Se iniciará la instalación.

Nota: Puede encontrar los siguientes archivos de registro en el directorio /var/opt/novell/eDirAPI/log:

- edirapi.log : se utiliza para registrar diferentes eventos en edirapi y los problemas de depuración.
- edirapi_audit.log: se utiliza para registrar eventos de auditoría de edirapi. Los registros siguen el formato de auditoría de CEF.
- identityconsole_install.log: se utiliza para capturar registros de instalación de Identity Console.

Los registros de Inicio/parada del proceso de Identity Console se encuentran en el archivo /var/log/messages.

Nota: NetIQ recomienda que, al instalar Identity Console y eDirectory en el mismo equipo, este tenga al menos una instancia de eDirectory disponible.

Varios árboles con una instancia independiente de Identity Console

Al conectarse a varios árboles de eDirectory, debe asegurarse de obtener un certificado de CA individual para cada árbol.

Por ejemplo, si se conecta a tres árboles de eDirectory, debe copiar los tres certificados de CA en el directorio etc/opt/novell/eDirAPI/cert/:

```
cp /home/user/SSCert.pem /etc/opt/novell/eDirAPI/cert/SSCert1.pem
```

```
cp /home/user/SSCert.pem /etc/opt/novell/eDirAPI/cert/SSCert2.pem
```

```
cp /home/user/SSCert.pem /etc/opt/novell/eDirAPI/cert/SSCert3.pem
```

Ejecute uno de los siguientes comandos para reiniciar Identity Console:

```
/usr/bin/identityconsole restart
```

o bien

```
systemctl restart netiq-identityconsole.service
```

Identity Console en Windows como estación de trabajo

Identity Console se puede iniciar en Windows como estación de trabajo y requiere que se ejecuten los servicios REST. Por lo tanto, al iniciarse, se ejecuta un proceso eDirAPI en el símbolo del sistema de edirapi.exe. Si se ha cerrado este terminal edirapi.exe, Identity Console dejará de funcionar.

En el siguiente procedimiento, se describe cómo ejecutar Identity Console en Windows.

1. Entre a SLD, Software License and Download (https://sldlogin.microfocus.com/nidp/idff/sso?id=5&sid=0&option=credential&sid=0), y desplácese a la página Descargas de software.
2. Seleccione lo siguiente:
   - Producto: eDirectory
   - Nombre del producto: eDirectory per User Sub SW E-LTU
   - Versión: 9.2
3. Descargue el archivo `IdentityConsole_<versión>_workstation_win_x86_64.zip`.
4. Extraiga el archivo `IdentityConsole_<versión>_workstation_win_x86_64.zip` en una carpeta.
5. Desplácese a la carpeta extraída `IdentityConsole_150_workstation_win_x86_64\eDirAPI\cert` y copie el certificado raíz de confianza de CA `SSCert.pem` y el certificado de servidor `keys.pfx`.

   Para obtener los certificados, consulte la sección: “Requisitos del sistema y previos para la estación de trabajo” en la página 18

   Si el usuario necesita conectarse a varios árboles de eDirectory, consulte la sección: “Varios árboles con Identity Console como estación de trabajo” en la página 30

   Nota: El nombre del certificado de servidor debe ser siempre `keys.pfx`.
6. Acceda a la carpeta en la que ha extraído la compilación y haga doble clic en el archivo `run.bat` (archivo por lotes de Windows).
7. Introduzca la contraseña del certificado de servidor (`keys.pfx`) en el símbolo del sistema.

   El terminal de proceso eDirAPI (edirapi.exe) comienza a ejecutarse y aparece la página de entrada a la sesión de Identity Console.

Nota:

- Si el terminal de proceso `eDirAPI` (edirapi.exe) ya se está ejecutando, inicie `identityconsole.exe` desde la carpeta de extracción de la compilación.

- Los usuarios pueden encontrar los siguientes registros en: `\IdentityConsole_150_workstation_win_x86_64\eDirAPI\log`.

  `edirapi.log`: se utiliza para registrar diferentes eventos de `edirapi` y problemas de depuración.

  `edirapi_audit.log`: se utiliza para registrar eventos de auditoría de `edirapi`. Los registros siguen el formato de auditoría de CEF.

- No se admite la entrada a la sesión basada en OSP en el modo estación de trabajo.
- Identity Console Workstation solo supervisa el `puerto 9000`. No modifique el archivo `edirapi_win.conf`.

Varios árboles con Identity Console como estación de trabajo

Identity Console permite al usuario conectarse a varios árboles mediante la obtención de un certificado de CA individual para cada árbol.

1. Cierre la estación de trabajo de Identity Console y el terminal eDirAPI.
2. Copie los certificados de CA `SSCert.pem` en la ubicación: `IdentityConsole_150_workstation_win_x86_64\eDirAPI\cert`.

   Por ejemplo, si desea conectarse a tres directorios de eDirectory, copie los certificados de CA como `SSCert1.pem`, `SSCert2.pem` y `SSCert3.pem` respectivamente.
3. Acceda a la carpeta en la que ha extraído la compilación y haga doble clic en el archivo `run.bat` (archivo por lotes de Windows).
4. Introduzca la contraseña de `keys.pfx` en el indicador del terminal y entre al árbol de eDirectory que desee.

Cómo detener y reiniciar Identity Console

Cómo detener y reiniciar Identity Console como contenedor de Docker

Para detener Identity Console, ejecute el siguiente comando:

```
docker stop <identityconsole-container-name>
```

Para reiniciar Identity Console, ejecute el siguiente comando:

```
docker restart <identityconsole-container-name>
```

Para iniciar Identity Console, ejecute el siguiente comando:

```
docker start <identityconsole-container-name>
```

Cómo detener y reiniciar una instancia independiente de Identity Console

Para detener Identity Console, ejecute uno de los siguientes comandos:

```
/usr/bin/identityconsole stop
```

o bien

```
systemctl stop netiq-identityconsole.service
```

Para reiniciar Identity Console, ejecute uno de los siguientes comandos:

```
/usr/bin/identityconsole restart
```

o bien

```
systemctl restart netiq-identityconsole.service
```

Para iniciar Identity Console, ejecute uno de los siguientes comandos:

```
/usr/bin/identityconsole start
```

o bien

```
systemctl start netiq-identityconsole.service
```

Cierre y reinicio de la estación de trabajo de Identity Console

Para cerrar la aplicación y el proceso, siga este procedimiento:

1. Cierre la aplicación de escritorio para Windows de Identity Console.
2. Detenga el proceso eDirAPI. Para ello, cierre el terminal de proceso `eDirAPI`.

Para reiniciar la estación de trabajo de Identity Console, acceda a la carpeta en la que ha extraído la compilación y haga doble clic en el archivo `run.bat` (archivo por lotes de Windows).

Nota: Si el terminal de proceso `eDirAPI` ya se está ejecutando, inicie `identityconsole.exe` desde la carpeta de extracción de la compilación para reiniciar la estación de trabajo de Identity Console.

Gestión de la persistencia de datos

Junto con los contenedores de Identity Console, también se crean volúmenes para la persistencia de datos. Para utilizar los parámetros de configuración de un contenedor anterior mediante los volúmenes, realice los siguientes pasos:

1. Detenga el contenedor de Docker actual mediante el siguiente comando:

   `docker stop` *`identityconsole-container`*

2. Cree el segundo contenedor mediante los datos de la aplicación del contenedor anterior almacenado en el volumen de Docker (`edirapi-volume-1`):

```
docker create --name identityconsole-container-2 --network=host --
volume edirapi-volume-1:/config/ identityconsole:1.0.0
```

3 Inicie el segundo contenedor mediante el siguiente comando:

```
docker start identityconsole-container-2
```

4 (Opcional) Ahora el primer contenedor se puede eliminar mediante el siguiente comando:

```
docker rm identityconsole-container
```

Distribución de Identity Console en Azure Kubernetes Service

Azure Kubernetes Service (AKS) es un servicio administrado de Kubernetes que le permite distribuir y gestionar clústeres. En esta sección se incluyen los procedimientos siguientes:

Distribución de Identity Console en un clúster de AKS

En esta sección, se describen los siguientes procedimientos para distribuir Identity Console en un clúster de AKS:

- “Creación de un registro de contenedor de Azure (ACR)” en la página 32
- “Configuración de un clúster de Kubernetes” en la página 33
- “Creación de una dirección IP pública de SKU estándar” en la página 34
- “Configuración de Cloud Shell y conexión al clúster de Kubernetes” en la página 34
- “Distribución de la aplicación” en la página 34

Creación de un registro de contenedor de Azure (ACR)

Un registro de contenedor de Azure (ACR) es un registro privado basado en Azure para las imágenes del contenedor de Docker.

Para obtener más información, consulte Create an Azure container registry using the Azure portal (Crear un registro de contenedor de Azure mediante el portal de Azure) en la sección "Create container registry - Portal" (Crear registro de contenedor - Portal) o realice los siguientes pasos para crear un registro de contenedor de Azure (ACR):

1. Inicie sesión en el portal de Azure.
2. Vaya a **Crear un recurso** > **Contenedores** > **Registro de contenedor**.
3. En la pestaña **Básico**, especifique los valores de **Grupo de recursos** y **Nombre del registro**. El nombre del registro debe ser exclusivo en Azure y contener un mínimo de 5 caracteres alfanuméricos y un máximo de 50.

   Acepte los valores por defecto para los ajustes restantes.
4. Haga clic en **Revisar y crear**.
5. Haga clic en **Crear**.
6. Entre en la CLI de Azure y ejecute el comando siguiente para entrar en el registro de contenedor de Azure.

```
az acr login --name registryname
```

Por ejemplo:

```
az acr login --name < idconsole >
```

7. Recupere el servidor de entrada a la sesión del registro de contenedor de Azure mediante el siguiente comando:

   ```
   az acr show --name registryname --query loginServer --output table
   ```

   Por ejemplo:

   ```
   az acr show --name < idconsole > --query loginServer --output table
   ```

8. Etiquete la imagen local de Identity Console con el nombre del servidor de entrada a la sesión de ACR (registryname.azureacr.io) mediante el siguiente comando:

   ```
   docker tag idconsole-image <login server>/idconsole-image
   ```

   Por ejemplo:

   ```
   docker tag identityconsole:<version> registryname.azurecr.io/
   identityconsole:<version>
   ```

9. Envíe la imagen etiquetada al registro.

   ```
   docker push <login server>/idconsole: <version>
   ```

   Por ejemplo:

   ```
   docker push registryname.azurecr.io/identityconsole:<version>
   ```

10. Recupere la lista de imágenes del registro mediante el siguiente comando:

    ```
    az acr show --name registryname --query loginServer --output table
    ```

Configuración de un clúster de Kubernetes

Cree un recurso de servicio de Kubernetes mediante el portal o la CLI de Azure.

Para obtener pasos más detallados para la creación de un recurso de servicio de Kubernetes en Azure con un nodo, consulte Create an AKS Cluster (Crear un clúster de AKS) en Azure Quickstart.

Nota:

- Asegúrese de seleccionar Azure CNI como red.
- Seleccione la red virtual existente (en la que se ha distribuido el servidor de eDirectory en la subred).
- Seleccione el registro de contenedor existente en el que está disponible la imagen de Identity Console.

Creación de una dirección IP pública de SKU estándar

Un recurso de dirección IP pública en el grupo de recursos del clúster de Kubernetes actúa como IP del equilibrador de carga para la aplicación.

Para ver los pasos detallados, consulte la sección Create a public IP address using the Azure portal (Crear una dirección IP pública mediante el portal de Azure) en "Create public IP address – Portal" (Crear dirección IP pública - Portal).

Configuración de Cloud Shell y conexión al clúster de Kubernetes

Utilice Cloud Shell, disponible en el portal de Azure, para todas las operaciones.

Para iniciar Cloud Shell en el portal de Azure, consulte Start Cloud Shell (Iniciar Cloud Shell) en Bash – Quickstart o lleve a cabo los siguientes pasos para configurar Cloud Shell y conectarse al clúster de Kubernetes:

1. En el portal de Azure, haga clic en el botón  para abrir Cloud Shell.

   Nota: Para gestionar un clúster de Kubernetes, utilice el cliente de línea de comandos de Kubernetes, `kubectl`. Si se utiliza Azure Cloud Shell, kubectl ya estará instalado.

2. Configure kubectl para conectarse al clúster de Kubernetes mediante el siguiente comando:

   ```
   az aks get-credentials --resource-group "resource group name" --name "Kubernetes cluster name"
   ```

   Por ejemplo:

   ```
   az aks get-credentials --resource-group myResourceGroup --name myAKSCluster
   ```

3. Compruebe la lista de nodos del clúster mediante el siguiente comando:

   ```
   kubectl get nodes
   ```

Distribución de la aplicación

Para distribuir Identity Console, puede utilizar los archivos de ejemplo `idc-services.yaml`, `idc-statefulset.yaml`, `idc-storageclass.yaml` y `idc-pvc.yaml`.

También puede crear sus propios archivos yaml según sea necesario.

1. Cree un recurso de clase de almacenamiento mediante el comando siguiente:

   ```
   kubectl apply -f <location of the YAML file>
   ```

   Por ejemplo:

   ```
   kubectl apply -f idc-storageclass.yaml
   ```

(Opcional) Para obtener más información sobre cómo crear y utilizar dinámicamente un volumen persistente con un recurso compartido de archivos de Azure, consulte Dynamically create and use a persistent volume with Azure Files in Azure Kubernetes Service (AKS) (Crear y utilizar dinámicamente un volumen persistente con archivos de Azure en Azure Kubernetes Service).

A continuación, se muestra un archivo de recursos de clase de almacenamiento de ejemplo:

```
kind: StorageClass
apiVersion: storage.k8s.io/v1
metadata:
  name: azurefilesc
provisioner: kubernetes.io/azure-file
mountOptions:
  - dir_mode=0777
  - file_mode=0777
  - uid=0
  - gid=0
  - mfsymlinks
  - cache=strict
  - actimeo=30
parameters:
  skuName: Standard_LRS
  shareName: fileshare
~
```

Un recurso de clase de almacenamiento permite el aprovisionamiento dinámico de almacenamiento. Se utiliza para definir cómo se crea un recurso compartido de archivo de Azure.

2. Consulte la información de la clase de almacenamiento mediante el comando siguiente:

   ```
   kubectl get sc
   ```

3. Cree un recurso de PVC mediante el archivo `idc-pvc.yaml`:

   ```
   kubectl apply -f <location of the YAML file>
   ```

   Por ejemplo:

   ```
   kubectl apply -f idc.pvc.yaml
   ```

   A continuación, se muestra un archivo de recursos de PVC de ejemplo:

   ```
   apiVersion: v1
   kind: PersistentVolumeClaim
   metadata:
     name: pvcforsc
   spec:
     accessModes:
       - ReadWriteMany
     storageClassName: azurefilesc
     resources:
       requests:
         storage: 5Gi
   ```

Un recurso de reclamación de volumen persistente crea el recurso compartido de archivo. Una reclamación de volumen persistente (PVC) utiliza el objeto de clase de almacenamiento para aprovisionar dinámicamente un recurso compartido de archivo de Azure.

4. Cargue el archivo edirapi.conf, el certificado de CA y el certificado del servidor en Cloud Shell.

   Haga clic en el icono del botón para **cargar/descargar archivos**  en Cloud Shell y cargue los archivos `edirapi.conf`, `SSCert.pem` y `keys.pfx`.

   Nota: edirapi.conf presenta un parámetro "origin" (origen). Aquí es necesario proporcionar la dirección IP con la que se accederá a la aplicación de Identity Console. (Utilice la dirección IP creada en la sección "Creación de una dirección IP pública de SKU estándar" en la página 34).

   La distribución de Identity Console requiere el certificado de servidor (`keys.pfx`).

   Al crear el certificado de servidor, asegúrese de proporcionar un nombre DNS válido en el nombre alternativo del sujeto.

   Pasos para crear un nombre DNS válido:

   Un pod típico distribuido mediante StatefulSet presenta un nombre DNS como el siguiente: {statefulsetname}-{ordinal}.{servicename}.{namespace}.svc.cluster.local

   - Si el nombre de StatefulSet del archivo "idconsole-statefulset.yaml" es "idconsole-app", "statefulsetname" es igual a "idconsole-app".
   - Si es el primer pod, "ordinal" es igual a 0.
   - Si se define serviceName en el archivo "idconsole -statefulset.yaml" como "idconsole", serviceName es igual a "idconsole".
   - Si es el espacio de nombres por defecto, "namespace" es igual a "default".

   Salida: idconsole-app-0.idcosole.default.svc.cluster.local

   ---

5. Cree un recurso configmap en el clúster de Kubernetes que almacenará los archivos de configuración junto con los certificados.

   Antes de ejecutar el comando, asegúrese de que los archivos (`edirapi.conf`, `SSCert.pem` y `keys.pfx`) estén presentes en el directorio.

   ```
   kubectl create configmap <confgimapName> --from-file= "path where the files are present"
   ```

   Por ejemplo:

   ```
   kubectl create configmap config-data --from-file=/data
   ```

6. Vea los detalles del objeto configmap mediante el comando kuitemtl describe:

   ```
   kubectl describe configmap <configmapName>
   ```

   Por ejemplo:

   ```
   kubectl describe configmap confg-data
   ```

7. Cree un recurso StatefulSet para distribuir el contenedor.

   Ejecute el siguiente comando para distribuir el contenedor:

   ```
   kubectl apply -f <location of the YAML file>
   ```

   Por ejemplo:

```
kubectl apply -f idc-statefulset.yaml
```

A continuación, se muestra un archivo de recursos StatefulSet de ejemplo:

```
apiVersion: apps/v1
kind: StatefulSet
metadata:
  name: idconsole-app
spec:
  serviceName: idconsole
  selector:
    matchLabels:
      app: idconsole
  replicas: 1
  template:
    metadata:
      labels:
        app: idconsole
    spec:
      containers:
      - name: idconsole-container
        image: registryname.azurecr.io/identityconsole:<version>
        env:
        - name: ACCEPT_EULA
          value: "Y"
        ports:
        - containerPort: 9000
        volumeMounts:
          - name: configfiles
            mountPath: /config/data
          - name: datapersistenceandlog
            mountPath: /config
            subPath: log
      volumes:
        - name: configfiles
          configMap:
            name: config-data
        - name: datapersistenceandlog
          persistentVolumeClaim:
            claimName: pvcforsc
```

8. Ejecute el siguiente comando para verificar el estado del pod distribuido:

   ```
   kubectl get pods -o wide
   ```

9. Cree un recurso de servicio de tipo loadBalancer.

   El tipo de servicio especificado en el archivo yaml es loadBalancer.

   Cree un recurso de servicio mediante el siguiente comando:

   ```
   kubectl apply -f <location of the YAML file>
   ```

   Por ejemplo:

   ```
   kubectl apply -f ids-service.yaml
   ```

   A continuación, se muestra un archivo de recursos de servicio de ejemplo:

```
apiVersion: v1
kind: Service
metadata:
  name: idconsole-service
  labels:
    run: idconsole-service
spec:
  type: LoadBalancer
  loadBalancerIP: xx.xx.xx.xx
  selector:
    app: idconsole
  ports:
   - port: 9000
     targetPort: 9000
     protocol: TCP
```

Compruebe la dirección EXTERNAL-IP (o loadBalancerIP) mediante el siguiente comando:

```
kubectl get svc -o wide
```

10. Inicie la dirección URL mediante EXTERNAL-IP (o la dirección de loadBalancerIP).

    Por ejemplo:

    https://<EXTERNAL-IP>:9000/identityconsole

Modificación del certificado de servidor

En esta sección, se proporciona información sobre cómo modificar el certificado de servidor en el contenedor de Docker y en la instancia independiente de Identity Console.

- "Modificación del certificado de servidor en el contenedor de Docker" en la página 38
- "Modificación del certificado de servidor en una instancia independiente de Identity Console" en la página 39

Modificación del certificado de servidor en el contenedor de Docker

Lleve a cabo los pasos siguientes para modificar el certificado de servidor en el contenedor de Docker:

1 Ejecute el siguiente comando para copiar el nuevo certificado de servidor en cualquier ubicación del contenedor.

  Por ejemplo:

  ```
  docker cp /path/to/new-keys.pfx <container_id/name>:/tmp/new-keys.pfx
  ```

2 Entre al contenedor mediante el siguiente comando:

  ```
  docker exec -it <container_name> bash
  ```

3 Ejecute NLPCERT para almacenar las claves como pseudousuario:

```
LD_LIBRARY_PATH=/opt/novell/lib64/:/opt/novell/eDirectory/lib64/:/opt/
netiq/common/openssl/lib64/ /opt/novell/eDirAPI/sbin/nlpcert -i /tmp/
new-keys.pfx -o /etc/opt/novell/eDirAPI/conf/ssl/private/cert.pem
```

4 Salga de la consola del contenedor mediante el siguiente comando:

```
exit
```

5 Reinicie el contenedor. Para ello, introduzca:

```
docker restart <container name>
```

Modificación del certificado de servidor en una instancia independiente de Identity Console

Lleve a cabo los pasos siguientes para modificar el certificado de servidor en un contenedor independiente:

1 Ejecute NLPCERT para almacenar las claves:

```
su - nds -c "LD_LIBRARY_PATH=/opt/novell/lib64/:/opt/novell/eDirectory/
lib64/:/opt/netiq/common/openssl/lib64/ /opt/novell/eDirAPI/sbin/
nlpcert -i /Expiredcert/noexpire/new-keys.pfx -o /etc/opt/novell/
eDirAPI/conf/ssl/private/cert.pem"
```

2 Reinicie Identity Console:

```
systemctl restart netiq-identityconsole.service
```

3 Actualización de Identity Console

En este capítulo, se describe el proceso de actualización de Identity Console a las versiones más recientes. Para preparar la actualización, revise los requisitos previos y los requisitos del sistema en Capítulo 1, “Planificación de la instalación de Identity Console”, en la página 11.

En esta sección se incluyen los procedimientos siguientes:

Actualización de Identity Console como contenedor de Docker

Si hay disponible una nueva versión de la imagen de Identity Console, el administrador puede realizar un procedimiento de actualización para distribuir el contenedor con la versión más reciente de Identity Console. Antes de llevar a cabo una actualización, asegúrese de almacenar de forma permanente todos los datos necesarios de la aplicación en los volúmenes de Docker. Realice los siguientes pasos para actualizar Identity Console mediante el contenedor de Docker:

1. Descargue y cargue la versión más reciente de la imagen de Docker desde Software License and Download (https://sld.microfocus.com/) y lleve a cabo los pasos necesarios para instalar la versión más reciente de Identity Console, como se indica en “Distribución de Identity Console” en la página 21.
2. Una vez cargada la imagen más reciente de Docker, detenga el contenedor de Docker actual mediante el siguiente comando:

   ```
   docker stop identityconsole-container
   ```

3. (Opcional) Obtenga la copia de seguridad del volumen compartido.
4. Ejecute el siguiente comando para suprimir el contenedor de Identity Console existente:

   ```
   docker rm <container name>
   ```

   Por ejemplo:

   ```
   docker rm identityconsole-container
   ```

5. (Opcional) Ejecute el siguiente comando para suprimir la imagen de Docker de Identity Console obsoleta:

   ```
   docker rmi identityconsole
   ```

6. Cree el contenedor de Docker para Identity Console mediante el siguiente comando:

```
docker create --name <identityconsole-container-name> --env
ACCEPT_EULA=Y --network=<network-type> --volume <volume-name>:/config/
identityconsole:<version>
```

Por ejemplo:

```
docker create --name identityconsole-container --env ACCEPT_EULA=Y --
network=host --volume IDConsole-volume:/config/
identityconsole:1.5.0.0000
```

Nota

- Puede aceptar el CLUF. Para ello, establezca la variable de entorno ACCEPT_EULA en Y. También puede aceptar el CLUF en las instrucciones en pantalla mientras inicia el contenedor utilizando la opción `-it` del comando create de Docker y acceder al modo interactivo.
- El parámetro `--volume` del comando anterior creará un volumen para almacenar los datos de configuración y registro. En este caso, hemos creado un volumen de ejemplo denominado `IDConsole-volume`.

7 Copie el archivo de certificado de servidor del sistema de archivos local en el contenedor recién creado como `/etc/opt/novell/eDirAPI/cert/keys.pfx` mediante el siguiente comando:

```
docker cp <absolute path of server certificate file> identityconsole-
container:/etc/opt/novell/eDirAPI/cert/keys.pfx
```

Por ejemplo:

```
docker cp /home/user/keys.pfx identityconsole-container:/etc/opt/
novell/eDirAPI/cert/keys.pfx
```

Al conectarse a varios árboles de eDirectory, debe asegurarse de copiar al menos un certificado de servidor keys.pfx para todos los árboles conectados.

8 Copie el archivo de certificado de CA (`.pem`) del sistema de archivos local en el contenedor recién creado como `/etc/opt/novell/eDirAPI/cert/SSCert.pem` mediante el siguiente comando:

```
docker cp <absolute path of CA certificate file> identityconsole-
container:/etc/opt/novell/eDirAPI/cert/SScert.pem
```

Por ejemplo:

```
docker cp /home/user/SSCert.pem identityconsole-container:/etc/opt/
novell/eDirAPI/cert/SSCert.pem
```

Al conectarse a varios árboles de eDirectory, debe asegurarse de obtener un certificado de CA individual para todos los árboles conectados. Por ejemplo, si se conecta a tres árboles de eDirectory, debe copiar los tres certificados de CA en el contenedor de Docker:

```
docker cp /home/user/SSCert.pem identityconsole-container:/etc/opt/
novell/eDirAPI/cert/SSCert.pem
docker cp /home/user/SSCert1.pem identityconsole-container:/etc/opt/
novell/eDirAPI/cert/SSCert1.pem
docker cp /home/user/SSCert2.pem identityconsole-container:/etc/opt/
novell/eDirAPI/cert/SSCert2.pem
```

Nota: A partir de Identity Console 1.4, el archivo de configuración (edirapi.conf) no incluye explícitamente los parámetros "*ldapuser*", "*ldappassword*" y "*ldapserver*". El valor del parámetro "*bcert*" debe incluir la vía del directorio de los certificados raíz de confianza. Por ejemplo, bcert = "/etc/opt/novell/eDirAPI/cert/". Además, el parámetro "*origin*" (origen) es independiente del parámetro "*check-origin*" y es obligatorio cuando se utiliza la configuración de DNS.

9 Copie el archivo de configuración (`edirapi.conf`) del sistema de archivos local en el contenedor recién creado como `/etc/opt/novell/eDirAPI/conf/edirapi.conf` mediante el siguiente comando:

```
docker cp <absolute path of configuration file> identityconsole-container:/etc/opt/novell/eDirAPI/conf/edirapi.conf
```

Por ejemplo:

```
docker cp /home/user/edirapi.conf identityconsole-container:/etc/opt/novell/eDirAPI/conf/edirapi.conf
```

10 Inicie el segundo contenedor mediante el siguiente comando:

```
docker start identityconsole-container
```

11 Para comprobar el estado del contenedor en ejecución, inicie el siguiente comando:

```
docker ps -a
```

Actualización de una instancia independiente de Identity Console (sin Docker)

En esta sección, se explica el procedimiento de actualización de una instancia independiente de Identity Console:

1 Descargue `IdentityConsole_<versión>_Containers.tar.gz` desde Software License and Download (https://sld.microfocus.com/).

2 Entre a SLD, desplácese a la página Descarga de software de SLD y haga clic en **Descargar**.

3 Desplácese mediante la selección de Producto: **eDirectory** > Nombre del producto: **eDirectory per User Sub SW E-LTU** > Versión: **9.2**.

4 Descargue la compilación más reciente de Identity Console.

5 Extraiga el archivo descargado mediante el siguiente comando:

```
tar -zxvf IdentityConsole_<version>_Linux.tar.gz
```

6 Acceda a la carpeta en la que ha extraído la compilación de Identity Console.

7 Copie todos los certificados raíz de confianza de los árboles de eDirectory a los que desea conectarse en una carpeta. Para copiar el certificado raíz de confianza en la carpeta, ejecute el siguiente comando:

```
cp /var/opt/novell/eDirectory/data/SSCert.pem <folder path>
```

Por ejemplo:

```
cp /var/opt/novell/eDirectory/data/SSCert.pem /home/Identity_Console/
certs
```

8 Ejecute el comando siguiente:

```
./identityconsole_install
```

9 Especifique la vía de la carpeta de certificados raíz de confianza utilizada en el **paso 4**.

10 Identity Console se actualiza correctamente.

Actualización del contenedor OSP

Lleve a cabo los pasos siguientes para actualizar el contenedor OSP:

1 Descargue y cargue la versión más reciente de la imagen de OSP desde Software License and Download (https://sld.microfocus.com/).

Por ejemplo:

```
docker load --input osp.tar.gz
```

2 Una vez cargada la imagen más reciente de OSP, detenga el contenedor OSP actual mediante el siguiente comando:

```
docker stop <OSP container name>
```

3 (Opcional) Obtenga la copia de seguridad del volumen compartido.

4 Ejecute el siguiente comando para suprimir el contenedor OSP existente:

```
docker rm <OSP container name>
```

Por ejemplo:

```
docker rm OSP_Container
```

5 Vaya al directorio que contiene el almacén de claves (`tomcat.ks`) y el archivo de propiedades silenciosas, suprima el almacén de claves (`tomcat.ks`) y conserve la carpeta OSP existente. Genere un almacén de claves nuevo (`tomcat.ks`) con un tamaño de clave de 2048. Para obtener más información, consulte el **paso 4** de la sección Distribución del contenedor de OSP de la Guía de instalación de Identity Console.

6 Distribuya el contenedor mediante el siguiente comando:

```
docker run -d --name OSP_Container --network=host -e
SILENT_INSTALL_FILE=/config/silent.properties -v /data:/config
osp:<version>
```

Por ejemplo:

```
docker run -d --name OSP_Container --network=host -e
SILENT_INSTALL_FILE=/config/silent.properties -v /data:/config
osp:6.5.3
```

4 Desinstalación de Identity Console

En este capítulo, se describe el proceso de desinstalación de Identity Console:

Procedimiento de desinstalación para el entorno de Docker

Para desinstalar el contenedor de Docker de Identity Console, realice los siguientes pasos:

1. Detenga el contenedor de Identity Console:

   ```
   docker stop <container-name>
   ```

2. Ejecute el siguiente comando para eliminar el contenedor de Docker de Identity Console:

   ```
   docker rm -f <container_name>
   ```

3. Ejecute el siguiente comando para eliminar la imagen de Docker:

   ```
   docker rmi -f <docker_image_id>
   ```

4. Elimine el volumen de Docker:

   ```
   docker volume rm <docker-volume>
   ```

Nota: Si elimina el volumen, los datos también se suprimirán del servidor.

Procedimiento de desinstalación de la instancia independiente de Identity Console (sin Docker)

Para desinstalar la instancia independiente de Identity Console, realice los siguientes pasos:

1. Acceda al directorio `/usr/bin` en el equipo en el que se ha instalado Identity Console.
2. Ejecute el comando siguiente:

   ```
   ./identityconsoleUninstall
   ```

3. Identity Console se desinstala correctamente.

Nota: Cuando se instala eDirectory u otro producto de NetIQ en el equipo, el usuario debe desinstalar manualmente *nici* y *openssl*.