



Identity Console Guide d'installation

Septembre 2022

Mentions légales

Pour plus d'informations sur les mentions légales, les marques, les exclusions de garantie, les garanties, les limitations en matière d'exportation et d'utilisation, les droits du gouvernement américain, la politique relative aux brevets et la compatibilité avec la norme FIPS, rendez-vous sur <https://www.netiq.com/company/legal>.

Copyright © 2022 NetIQ Corporation. Tous droits réservés.

Table des matières

À propos de ce guide et de la bibliothèque	5
À propos de NetIQ Corporation	7
1 Planification de l'installation d'Identity Console	9
Configuration système requise et conditions préalables concernant l'installation de Docker	9
Configuration système requise	9
Conditions préalables	9
Configuration de l'environnement	11
Configuration système requise et conditions préalables concernant l'installation autonome (non-Docker)	13
Configuration système requise	14
(Facultatif) Conditions préalables à la configuration d'OSP	15
Configuration système requise et conditions préalables pour le poste de travail	16
Configuration système requise	16
Vérification de la signature RPM	17
2 Déploiement d'Identity Console	19
Recommandations en matière de sécurité	19
Déploiement d'Identity Console en tant que conteneur Docker	20
Déploiement du conteneur OSP	20
Déploiement d'Identity Console en tant que conteneur Docker	22
Arborescences multiples avec Identity Console en tant que Docker	24
Déploiement d'Identity Console en mode autonome	25
Déploiement d'Identity Console en mode autonome (non-Docker)	25
Arborescences multiples avec Identity Console en mode autonome	26
Identity Console sous Windows en tant que poste de travail	27
Arborescences multiples avec Identity Console en mode Poste de travail	28
Arrêt et redémarrage d'Identity Console	28
Arrêt et redémarrage d'Identity Console en tant que conteneur Docker	28
Arrêt et redémarrage d'Identity Console en mode autonome	29
Fermeture et redémarrage du poste de travail Identity Console	29
Gestion de la persistance des données	29
Déploiement d'Identity Console dans Azure Kubernetes Service	30
Déploiement d'Identity Console dans une grappe AKS	30
Modification du certificat de serveur	36
Modification du certificat de serveur dans le conteneur Docker	36
Modification du certificat de serveur dans Identity Console en mode autonome	37
3 Mise à niveau d'Identity Console	39
Mise à niveau d'Identity Console en tant que conteneur Docker	39
Mise à niveau d'Identity Console en mode autonome (non-Docker)	41
Mise à niveau du conteneur OSP	42

4 Désinstallation d'Identity Console	45
Procédure de désinstallation pour l'environnement Docker	45
Procédure de désinstallation pour l'installation autonome d'Identity Console (non-Docker)	45

À propos de ce guide et de la bibliothèque

Le *Guide d'installation d'Identity Console* fournit des informations sur la procédure d'installation et de gestion du produit NetIQ Identity Console (Identity Console). Il définit la terminologie utilisée et inclut des scénarios d'implémentation.

Public

Il est destiné aux administrateurs réseau.

Autres documents dans la bibliothèque

La bibliothèque propose les manuels suivants :

Guide d'installation

Décrit comment installer et mettre à niveau Identity Console. Ce manuel est destiné aux administrateurs réseau.

À propos de NetIQ Corporation

Fournisseur international de logiciels d'entreprise, nos efforts sont constamment axés sur trois défis inhérents à votre environnement (le changement, la complexité et les risques) et la façon dont vous pouvez les contrôler.

Notre point de vue

Adaptation au changement et gestion de la complexité et des risques : rien de neuf

Parmi les défis auxquels vous êtes confronté, il s'agit peut-être des principaux aléas qui vous empêchent de disposer du contrôle nécessaire pour mesurer, surveiller et gérer en toute sécurité vos environnements informatiques physiques, virtuels et en nuage (cloud computing).

Services métiers critiques plus efficaces et plus rapidement opérationnels

Nous sommes convaincus qu'en proposant aux organisations informatiques un contrôle optimal, nous leur permettons de fournir des services dans les délais et de manière plus rentable. Les pressions liées au changement et à la complexité ne feront que s'accroître à mesure que les organisations évoluent et que les technologies nécessaires à leur gestion deviennent elles aussi plus complexes.

Notre philosophie

Vendre des solutions intelligentes et pas simplement des logiciels

Pour vous fournir un contrôle efficace, nous veillons avant tout à comprendre les scénarios réels qui caractérisent les organisations informatiques telles que la vôtre, et ce jour après jour. De cette manière, nous pouvons développer des solutions informatiques à la fois pratiques et intelligentes qui génèrent assurément des résultats éprouvés et mesurables. En même temps, c'est tellement plus gratifiant que la simple vente de logiciels.

Vous aider à réussir, telle est notre passion

Votre réussite constitue le fondement même de notre manière d'agir. Depuis la conception des produits jusqu'à leur déploiement, nous savons que vous avez besoin de solutions informatiques opérationnelles qui s'intègrent en toute transparence à vos investissements existants. En même temps, après le déploiement, vous avez besoin d'une formation et d'un support continus. En effet, il vous faut un partenaire avec qui la collaboration est aisée... pour changer. En fin de compte, votre réussite est aussi la nôtre.

Nos solutions

- ♦ Gouvernance des accès et des identités
- ♦ Gestion des accès
- ♦ Gestion de la sécurité

- ♦ Gestion des systèmes et des applications
- ♦ Gestion des charges de travail
- ♦ Gestion des services

Contacteur le support

Pour toute question concernant les produits, tarifs et fonctionnalités, contactez votre partenaire local. Si vous ne pouvez pas contacter votre partenaire, contactez notre équipe de support ventes.

Monde :	www.netiq.com/about_netiq/officelocations.asp
États-Unis et Canada :	1-888-323-6768
Courrier électronique :	info@netiq.com
Site Web :	www.netiq.com

Contacteur le support technique

Pour tout problème spécifique au produit, contactez notre équipe du support technique.

Monde :	www.netiq.com/support/contactinfo.asp
Amérique du Nord et du Sud :	1-713-418-5555
Europe, Moyen-Orient et Afrique :	+353 (0) 91-782 677
Courrier électronique :	support@netiq.com
Site Web :	www.netiq.com/support

Contacteur le support en charge de la documentation

Notre objectif est de vous proposer une documentation qui réponde à vos besoins. Si vous avez des suggestions d'améliorations, cliquez sur le bouton **Add Comment** (Ajouter un commentaire) au bas de chaque page dans les versions HTML de la documentation publiée à l'adresse www.netiq.com/documentation. Vous pouvez également envoyer un message électronique à l'adresse Documentation-Feedback@netiq.com. Nous accordons une grande importance à vos commentaires et sommes impatients de connaître vos impressions.

Contacteur la communauté d'utilisateurs en ligne

La communauté en ligne de NetIQ, Qmunity, est un réseau collaboratif vous mettant en relation avec vos homologues et des spécialistes de NetIQ. En proposant des informations immédiates, des liens utiles vers des ressources et un accès aux experts NetIQ, Qmunity vous aide à maîtriser les connaissances nécessaires pour tirer pleinement parti du potentiel de vos investissements informatiques. Pour plus d'informations, consultez le site <http://community.netiq.com>.

1 Planification de l'installation d'Identity Console

Ce chapitre décrit la configuration système requise et les conditions préalables concernant l'installation d'Identity Console. Identity Console peut être exécuté en tant que conteneur Docker ou en tant qu'application autonome. Par conséquent, reportez-vous aux sections respectives pour connaître la configuration système requise et les conditions préalables concernant les deux types d'installation.

REMARQUE : Identity Console prend en charge eDirectory 9.2.4 HF2, Identity Manager Engine 4.8.3 HF2 et leurs versions ultérieures respectives. Vous devez mettre à niveau vos instances eDirectory et Identity Manager Engine avant d'utiliser Identity Console.

- ♦ « [Configuration système requise et conditions préalables concernant l'installation de Docker](#) » page 9
- ♦ « [Configuration système requise et conditions préalables concernant l'installation autonome \(non-Docker\)](#) » page 13
- ♦ « [Configuration système requise et conditions préalables pour le poste de travail](#) » page 16
- ♦ « [Vérification de la signature RPM](#) » page 17

Configuration système requise et conditions préalables concernant l'installation de Docker

Cette section décrit la configuration système requise et les conditions préalables concernant l'installation d'Identity Console en tant que conteneur Docker.

- ♦ « [Configuration système requise](#) » page 9
- ♦ « [Conditions préalables](#) » page 9
- ♦ « [Configuration de l'environnement](#) » page 11

Configuration système requise

Identity Console peut être exécuté en tant que conteneur Docker. Par conséquent, pour plus d'informations sur la configuration système requise et sur les plates-formes prises en charge pour l'installation d'Identity Console, reportez-vous à la [documentation relative à Docker](#).

Conditions préalables

- Installez Docker 20.10.9-ce ou version ultérieure. Pour plus d'informations sur l'installation de Docker, reportez-vous à la section [Docker Installation](#) (Installation de Docker).

- ❑ Vous devez obtenir un certificat de serveur PKCS12 avec la clé privée pour chiffrer/déchiffrer l'échange de données entre le serveur Identity Console et le serveur dorsal. Ce certificat de serveur sert à sécuriser la connexion HTTP. Vous pouvez utiliser des certificats de serveur générés par une autorité de certification externe. Pour plus d'informations, reportez-vous à la section [Création d'objets Certificat de serveur](#). Le certificat de serveur doit contenir l'alias de l'objet avec l'adresse IP et le nom DNS du serveur Identity Console. Une fois l'objet Certificat de serveur créé, vous devez l'exporter au format .pfx.
- ❑ Vous devez obtenir un certificat d'autorité de certification (CA) pour toutes les arborescences au format .pem pour valider la signature CA des certificats de serveur obtenus à l'étape précédente. Ce certificat CA racine garantit également l'établissement d'une communication LDAP sécurisée entre le client et le serveur Identity Console. Par exemple, vous pouvez obtenir le certificat CA eDirectory (SSCert.pem) à partir de l'emplacement /var/opt/novell/eDirectory/data/SSCert.pem.
- ❑ (Facultatif) À l'aide de One SSO Provider (OSP), vous pouvez activer l'authentification Single Sign-on pour vos utilisateurs sur le portail Identity Console. Vous devez installer OSP avant Identity Console. Pour configurer OSP pour Identity Console, suivez les invites à l'écran et fournissez les valeurs requises pour les paramètres de configuration. Pour plus d'informations, reportez-vous à la section « [Déploiement du conteneur OSP](#) » page 20. Pour enregistrer Identity Console sur un serveur OSP existant, vous devez ajouter manuellement les éléments suivants au fichier ism-configuration.properties dans le dossier /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
```

REMARQUE : OSP ne vous permet de vous connecter qu'à une seule arborescence eDirectory, car il ne prend pas en charge plusieurs arborescences eDirectory.

- ❑ Veillez à ce qu'une entrée DNS appropriée soit disponible pour votre machine hôte dans /etc/hosts avec un nom d'hôte complet.
- ❑ Si vous souhaitez utiliser Identity Console dans le navigateur Edge, vous devez télécharger la dernière version de Microsoft Edge pour bénéficier de toutes les fonctionnalités.

REMARQUE : si vous utilisez Identity Console dans Mozilla Firefox, l'opération peut échouer avec le message d'erreur *Origin Mismatch* (Discordance d'origine). Pour résoudre ce problème, effectuez les étapes suivantes :

- 1 Mettez Firefox à jour vers la version la plus récente.
 - 2 Entrez `about:config` dans le champ d'URL de Firefox, puis appuyez sur Entrée.
 - 3 Recherchez `Origin`.
 - 4 Double-cliquez sur `network.http.sendOriginHeader`, puis remplacez sa valeur par 1.
-

Configuration de l'environnement

Il peut être nécessaire de créer un fichier de configuration contenant certains paramètres. Si vous souhaitez configurer Identity Console avec OSP, vous devez spécifier les paramètres propres à OSP dans le fichier de configuration. Par exemple, créez le fichier `edirapi.conf` ci-dessous avec les paramètres OSP :

REMARQUE : Vous devez indiquer le nom de votre arborescence eDirectory dans le champ `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 vous souhaitez configurer Identity Console sans OSP, créez un fichier de configuration comme indiqué ci-dessous, sans les paramètres OSP :

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

REMARQUE : si vous souhaitez configurer Identity Console avec plusieurs arborescences eDirectory, vous pouvez ignorer les paramètres « `ldapserver` », « `ldapuser` » et « `ldappassword` » et créer le fichier de configuration.

Tableau 1-1 Description des paramètres de configuration dans le fichier de configuration

Paramètre de configuration	Description
<code>listen</code>	Spécification du port 9000 en tant que port d'écoute du serveur Identity Console dans le conteneur.

Paramètre de configuration	Description
ldapservers	Spécification de l'adresse IP et du numéro de port du serveur hôte eDirectory.
ldapuser	Spécification du nom de l'utilisateur eDirectory. Ce paramètre est utilisé en tant qu'informations d'identification pour lancer des appels LDAP vers eDirectory à l'aide du contrôle d'autorisation par proxy en cas de connexion OSP. L'utilisateur LDAP doit disposer des droits de superviseur sur l'arborescence eDirectory.
ldappassword	Spécification du mot de passe de l'utilisateur LDAP.
pfpassword	Spécification du mot de passe du fichier de certificat de serveur PKCS12.
ospmode	Spécification de la valeur <code>true</code> (vrai) pour intégrer OSP à Identity Console. Si ce paramètre a la valeur <code>false</code> , Identity Console utilise la connexion LDAP.
osp-token-endpoint	Cette URL est utilisée pour extraire certains attributs du serveur OSP afin de vérifier la validité du jeton d'authentification.
osp-authorize-url	Cette URL est utilisée par l'utilisateur pour fournir les informations d'identification permettant d'obtenir un jeton d'authentification.
osp-logout-url	Cette URL permet de mettre fin à la session entre l'utilisateur et le serveur OSP.
osp-redirect-url	Le serveur OSP redirige l'utilisateur vers cette URL une fois le jeton d'authentification accordé. REMARQUE : veillez à spécifier le nom de l'arborescence eDirectory en minuscules lorsque vous configurez Identity Console. Dans le cas contraire, la connexion au serveur Identity Console risque d'échouer.
osp-client-id	Spécification de l'ID du client OSP qui a été fourni lors de l'enregistrement d'Identity Console avec OSP.
ospclientpass	Spécification du mot de passe du client OSP qui a été fourni lors de l'enregistrement d'Identity Console avec OSP.
ospcert	Spécification de l'emplacement du certificat CA du serveur OSP.
bcert	Spécification de l'emplacement du certificat d'autorité de certification d'Identity Console.
loglevel	Spécification des niveaux de consignation à inclure dans le fichier journal. Ce paramètre peut avoir la valeur « fatal » (irrécupérable), « error » (erreur), « warn » (avertissement) ou « info ».

Paramètre de configuration	Description
check-origin	Si ce paramètre a la valeur <code>true</code> (vrai), le serveur Identity Console compare la valeur d'origine des requêtes. Les options disponibles sont <code>true</code> (vrai) et <code>false</code> (faux). Le paramètre <code>origin</code> est obligatoire même si la valeur du paramètre <code>check-origin</code> est définie sur <code>false</code> (faux) lors de l'utilisation de la configuration DNS.
origin	Identity Console compare la valeur d'origine des requêtes aux valeurs spécifiées dans ce champ. REMARQUE : à partir d'Identity Console 1.4, ce paramètre est indépendant du paramètre <code>check-origin</code> et est obligatoire si la configuration DNS est utilisée.
maxclients	Nombre maximal de clients simultanés qui peuvent accéder à IDConsole. Tout client supplémentaire au-delà de cette limite est placé en file d'attente.

REMARQUE

- ♦ N'utilisez le paramètre de configuration `ospmode` que si vous envisagez d'intégrer OSP à Identity Console.
- ♦ Si Identity Applications (Identity Apps) est configuré en mode grappe dans Identity Manager, vous devez fournir le nom DNS du serveur équilibreur de charge dans les champs `osp-token-endpoint`, `osp-authorize-url` et `osp-logout-url` du fichier de configuration. Si vous indiquez les détails du serveur OSP dans ces champs, la connexion à Identity Console échoue.
- ♦ Si Identity Console est configuré avec la même instance OSP qu'Identity Apps et Identity Reporting, le service d'authentification Single Sign-on prend effet lorsque vous vous connectez au portail Identity Console.
- ♦ L'URL HTTPS OSP doit être validée avec des certificats contenant une clé 2 048 bits ou supérieure avec Identity Console 1.4 ou version ultérieure.
- ♦ Si vous souhaitez interdire l'accès au portail Identity Console à partir de différents domaines, définissez le paramètre `samesitecookie` sur la valeur `strict`. Si vous souhaitez autoriser l'accès au portail Identity Console à partir de différents domaines, définissez le paramètre `samesitecookie` sur la valeur `lax`. Si le paramètre n'est pas spécifié durant la configuration, les paramètres du navigateur sont respectés par défaut.

Dès que le fichier de configuration est prêt, procédez au déploiement du conteneur. Pour plus d'informations, reportez-vous au « [Déploiement d'Identity Console en tant que conteneur Docker](#) » page 20.

Configuration système requise et conditions préalables concernant l'installation autonome (non-Docker)

- ♦ « [Configuration système requise](#) » page 14
- ♦ « [\(Facultatif\) Conditions préalables à la configuration d'OSP](#) » page 15

Configuration système requise

Cette section décrit la configuration système requise et les conditions préalables concernant l'installation d'Identity Console en mode autonome.

Catégorie	Configuration minimale
Processeur	1,4 GHz 64 bits
Mémoire	2 Go
Espace disque	200 Mo sous Linux
Navigateur pris en charge	<ul style="list-style-type: none">◆ Dernière version de Microsoft Edge◆ Dernière version de Google Chrome◆ Dernière version de Mozilla Firefox <p>REMARQUE : si vous utilisez Identity Console dans Mozilla Firefox, l'opération peut échouer avec le message d'erreur <code>Origin Mismatch</code> (Discordance d'origine). Pour résoudre ce problème, effectuez les étapes suivantes :</p> <ol style="list-style-type: none">1 Mettez Firefox à jour vers la version la plus récente.2 Entrez <code>about:config</code> dans le champ d'URL de Firefox, puis appuyez sur Entrée.3 Recherchez Origin.4 Double-cliquez sur <code>network.http.sendOriginHeader</code>, puis remplacez sa valeur par 1.
Système d'exploitation pris en charge	<ul style="list-style-type: none">◆ Certifiés :<ul style="list-style-type: none">◆ SUSE Linux Enterprise Server (SLES) 15 SP1, SP2 et SP3◆ SUSE Linux Enterprise Server (SLES) 12 SP1, SP2, SP3, SP4 et SP5◆ Red Hat Enterprise Linux (RHEL) 7.8, 7.9, 8.0, 8.1, 8.2, 8.3, 8.4 et 8.5◆ OpenSUSE 15.1 et 15.2◆ Pris en charge : pris en charge sur les versions ultérieures des Support Packs des systèmes d'exploitation certifiés ci-dessus.

Catégorie	Configuration minimale
Certificats	<ul style="list-style-type: none"> Vous devez obtenir un certificat de serveur PKCS12 avec la clé privée pour chiffrer/déchiffrer l'échange de données entre le client et le serveur Identity Console. Ce certificat de serveur sert à sécuriser la connexion HTTP. Vous pouvez utiliser des certificats de serveur générés par une autorité de certification externe. Pour plus d'informations, reportez-vous à la section Création d'objets Certificat de serveur. Le certificat de serveur doit contenir l'alias de l'objet avec l'adresse IP et le nom DNS du serveur Identity Console. Une fois l'objet Certificat de serveur créé, vous devez l'exporter au format .pfx. Vous devez obtenir un certificat d'autorité de certification (CA) pour toutes les arborescences au format .pem pour valider la signature CA des certificats de serveur obtenus à l'étape précédente. Ce certificat CA racine garantit également l'établissement d'une communication LDAP sécurisée entre le client et le serveur Identity Console. Par exemple, vous pouvez obtenir le certificat CA eDirectory (SSCert.pem) à partir de l'emplacement /var/opt/novell/eDirectory/data/SSCert.pem.

Dès que vous êtes prêt, installez Identity Console. Pour plus d'informations, reportez-vous à la section « [Déploiement d'Identity Console en mode autonome](#) » page 25.

(Facultatif) Conditions préalables à la configuration d'OSP

À l'aide de One SSO Provider (OSP), vous pouvez activer l'authentification Single Sign-On pour vos utilisateurs sur le portail Identity Console. Vous devez installer OSP avant Identity Console. Pour configurer OSP pour Identity Console, suivez les invites à l'écran et fournissez les valeurs requises pour les paramètres de configuration. Pour plus d'informations, reportez-vous à la section « [Déploiement du conteneur OSP](#) » page 20. Pour enregistrer Identity Console sur un serveur OSP existant, vous devez ajouter manuellement les éléments suivants au fichier `ism-configuration.properties` dans le dossier `/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
```

REMARQUE

- ♦ Si vous installez OSP pour la première fois, spécifiez l'option « γ » pour **Configure OSP with eDir API** (Configurer OSP avec l'API eDir), puis suivez les invites qui s'affichent à l'écran pour enregistrer Identity Console dans OSP.
 - ♦ Veillez à spécifier le nom de l'arborescence eDirectory en minuscules lorsque vous configurez Identity Console. Dans le cas contraire, la connexion au serveur Identity Console risque d'échouer.
 - ♦ OSP ne vous permet de vous connecter qu'à une seule arborescence eDirectory, car il ne prend pas en charge plusieurs arborescences eDirectory.
-

Configuration système requise et conditions préalables pour le poste de travail

- ♦ [« Configuration système requise » page 16](#)

Configuration système requise

Cette section décrit la configuration système requise et les conditions préalables concernant l'exécution d'Identity Console en mode Poste de travail.

Catégorie	Configuration minimale
Processeur	1.5 GHz 64 bits
Mémoire	2 Go
Espace disque	1 Go sous Windows
Système d'exploitation pris en charge	<ul style="list-style-type: none">♦ Certifiés :<ul style="list-style-type: none">♦ Windows Server 2016♦ Windows Server 2019♦ Windows Server 2022♦ Windows 10♦ Windows 11

Catégorie	Configuration minimale
Certificats	<ul style="list-style-type: none"> Vous devez obtenir un certificat de serveur au format PFX pour échanger des données entre le client Identity Console et le serveur REST. Ce certificat de serveur doit toujours être nommé <code>keys.pfx</code>. Pour plus d'informations, reportez-vous à la section Création d'objets Certificat de serveur (https://www.netiq.com/documentation/edirectory-92/edir_admin/data/b1j4tpo3.html#b1j4u0cm). Vous devez obtenir un certificat d'autorité de certification (CA) pour toutes les arborescences au format <code>.pem</code> pour valider la signature CA des certificats de serveur obtenus à l'étape précédente. Ce certificat CA racine garantit également l'établissement d'une communication LDAP sécurisée entre le client et le serveur Identity Console. <p>Par exemple, vous pouvez obtenir le certificat CA eDirectory <code>SSCert.pem</code> pour Linux à partir de l'emplacement <code>/var/opt/novell/eDirectory/data/SSCert.pem</code>.</p> <p>Obtenez le certificat CA eDirectory <code>SSCert.pem</code> pour Windows à partir de <code><emplacement d'installation de eDirectory>\NetIQ\eDirectory\DIBFiles\CertServ\SSCert.pem</code>.</p>

Dès que vous êtes prêt, déployez Identity Console. Pour plus d'informations, reportez-vous à la section « [Identity Console sous Windows en tant que poste de travail](#) » page 27.

Vérification de la signature RPM

Pour effectuer la vérification de la signature RPM, procédez comme suit :

- 1 Accédez au dossier dans lequel la version est extraite.

Par exemple : `<emplacement désarchivé d'Identity Console>/IdentityConsole_150_Linux/license/MicroFocusGPGPackageSign.pub`.

- 2 Exécutez la commande suivante pour importer la clé publique :

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

- 3 (Facultatif) Exécutez la commande suivante pour vérifier la signature RPM : `rpm --checksig -v <nom RPM>`.

Par exemple :

```
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 Déploiement d'Identity Console

Ce chapitre décrit la procédure de déploiement d'Identity Console ainsi que les recommandations en matière de sécurité. Pour préparer le déploiement, consultez les conditions préalables et la configuration système requise mentionnées au [Chapitre 1, « Planification de l'installation d'Identity Console »](#), page 9.

- ♦ « [Recommandations en matière de sécurité](#) » page 19
- ♦ « [Déploiement d'Identity Console en tant que conteneur Docker](#) » page 20
- ♦ « [Déploiement d'Identity Console en mode autonome](#) » page 25
- ♦ « [Identity Console sous Windows en tant que poste de travail](#) » page 27
- ♦ « [Arrêt et redémarrage d'Identity Console](#) » page 28
- ♦ « [Gestion de la persistance des données](#) » page 29
- ♦ « [Déploiement d'Identity Console dans Azure Kubernetes Service](#) » page 30
- ♦ « [Modification du certificat de serveur](#) » page 36

Recommandations en matière de sécurité

- ♦ Aucune contrainte de ressources n'est définie par défaut pour les conteneurs Docker. Ainsi, chaque conteneur dispose d'un accès à toutes les ressources de l'UC et mémoire fournies par le kernel de l'hôte. Vous devez également vous assurer qu'un conteneur en cours d'exécution ne consomme pas plus de ressources et n'épuise pas les autres conteneurs en cours d'exécution en définissant des limites pour le volume de ressources utilisables par un conteneur.
 - ♦ Le conteneur Docker doit veiller à ce qu'une limite fixe soit appliquée à la mémoire utilisée par le conteneur à l'aide du drapeau `--memory` dans la commande `docker run`.
 - ♦ Le conteneur Docker doit veiller à ce qu'une limite soit appliquée à la quantité d'UC utilisée par un conteneur en cours d'exécution à l'aide du drapeau `--cpuset-cpus` dans la commande `docker run`.
- ♦ Le drapeau `--pids-limit` doit avoir la valeur 300 pour restreindre le nombre de threads du kernel générés à tout moment dans le conteneur. Ceci permet d'éviter les attaques par déni de service (DoS).
- ♦ Vous devez définir la stratégie de redémarrage de conteneur en cas d'échec sur la valeur 5 à l'aide du drapeau `--restart` dans la commande `docker run`.
- ♦ Vous ne devez utiliser le conteneur qu'une fois que l'état de santé est **Healthy** (Sain) lorsque le conteneur est activé. Pour vérifier l'état de santé du conteneur, exécutez la commande suivante :

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

- ♦ Le conteneur Docker démarre toujours en tant qu'utilisateur non-root (n_{ds}). Par mesure de sécurité supplémentaire, activez la réassignation de l'espace de noms utilisateur sur le daemon pour éviter les attaques par élévation de privilèges à partir du conteneur. Pour plus d'informations sur la réassignation de l'espace de noms utilisateur, reportez-vous à la section [Isolate containers with a user namespace](#) (Isoler les conteneurs avec un espace de noms).

Déploiement d'Identity Console en tant que conteneur Docker

Cette section explique les procédures suivantes :

- ♦ « [Déploiement du conteneur OSP](#) » page 20
- ♦ « [Déploiement d'Identity Console en tant que conteneur Docker](#) » page 22
- ♦ « [Arborescences multiples avec Identity Console en tant que Docker](#) » page 24

Déploiement du conteneur OSP

Pour déployer le conteneur OSP, procédez comme suit :

- 1 Connectez-vous au [portail de téléchargements et de licences de logiciels \(Software License and Download\)](https://sld.microfocus.com/) (<https://sld.microfocus.com/>) et accédez à la page de téléchargements de logiciels.
- 2 Sélectionnez les éléments suivants :
 - ♦ Produit : eDirectory
 - ♦ Nom du produit : eDirectory per User Sub SW E-LTU
 - ♦ Version : 9.2
- 3 Téléchargez le fichier `IdentityConsole_<version>_Containers_tar.zip`.
- 4 Extrayez le fichier téléchargé dans un dossier.
- 5 Modifiez le fichier de propriétés en mode silencieux selon vos besoins. Voici un exemple de fichier de propriétés en mode silencieux :

```
# 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

```

REMARQUE : pour éviter les contraintes d'espace lors de l'utilisation du fichier de propriétés silencieuses (texte DOS), vous devez convertir le fichier texte DOS au format UNIX à l'aide de l'outil dos2unix. Exécutez la commande ci-dessous pour convertir le fichier texte de fins de ligne DOS en fins de ligne UNIX :

```
dos2unix nom_fichier
```

Par exemple :

```
dos2unix exemple_fichier
```

-
- 6 Générez un certificat de serveur (`cert.der`) à l'aide d'iManager, puis importez-le dans le fichier Keystore (`tomcat.ks`). Copiez le fichier de propriétés en mode silencieux et le fichier Keystore (`tomcat.ks`) dans n'importe quel répertoire. Par exemple : `/data`. Pour créer un certificat de serveur et l'importer dans le fichier Keystore, procédez comme suit :

- 6a Exécutez la commande suivante pour créer un fichier Keystore (`tomcat.ks`). Générez la clé et veillez à ce que le nom CN ou le nom d'hôte complet de la machine corresponde à l'adresse 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 Exécutez la commande suivante pour créer une requête de signature de certificat. Par exemple : `cert.csr`.

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

6c Transmettez le fichier `cert.csr` à iManager et obtenez le certificat de serveur `osp.der`.
Veillez à sélectionner le type de clé Personnalisé, les options d'utilisation de la clé
Chiffrement des données, Chiffrement des clés et Signature numérique, ainsi que le champ
d'alias de l'objet du certificat pour qu'il contienne l'adresse IP ou le nom d'hôte du
serveur OSP. Pour plus d'informations, reportez-vous à la section [Création d'un objet
Certificat de serveur](#).

6d Exécutez les commandes suivantes pour importer le certificat d'autorité de certification
(`SSCert.der`) et le certificat de serveur (`cert.der`) dans le fichier Keystore `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 Exécutez la commande suivante pour charger l'image OSP :

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

8 Déployez le conteneur à l'aide de la commande suivante :

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

Par exemple :

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

Déploiement d'Identity Console en tant que conteneur Docker

Cette section décrit la procédure à suivre pour déployer Identity Console en tant que conteneur Docker :

REMARQUE : les paramètres de configuration, les exemples de valeurs et les exemples mentionnés dans cette procédure sont fournis uniquement à des fins de référence. Veillez à ne pas les utiliser directement dans votre environnement de production.

- 1 Connectez-vous au [portail de téléchargements et de licences de logiciels \(Software License and Download\)](https://sld.microfocus.com/) (<https://sld.microfocus.com/>) et accédez à la page de téléchargements de logiciels.
- 2 Sélectionnez les éléments suivants :
 - ◆ Produit : eDirectory
 - ◆ Nom du produit : eDirectory per User Sub SW E-LTU
 - ◆ Version : 9.2
- 3 Téléchargez le fichier `IdentityConsole_<version>_Container.tar.zip`.
- 4 L'image doit être chargée dans le registre Docker local. Extrayez et chargez le fichier `IdentityConsole_<version>_Containers.tar.gz` à l'aide des commandes suivantes :

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

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

5 Créez le conteneur Docker Identity Console à l'aide de la commande suivante :

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

Par exemple :

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

REMARQUE

- ♦ Vous pouvez accepter le contrat de licence EULA en affectant la valeur « Y » à la variable d'environnement ACCEPT_EULA. Vous pouvez également accepter ce contrat dans l'invite qui s'affiche au démarrage du conteneur à l'aide de l'option `-it` de la commande Docker create en mode interactif.
- ♦ Le paramètre `--volume` de la commande ci-dessus crée un volume destiné au stockage des données de configuration et de journal. Dans ce cas, nous avons créé un exemple de volume nommé `IDConsole-volume`.

6 Copiez le fichier de certificat de serveur de votre système de fichiers local vers le conteneur en tant que `/etc/opt/novell/eDirAPI/cert/keys.pfx` à l'aide de la commande ci-dessous. Pour plus d'informations sur la création du certificat de serveur, reportez-vous à la section « [Conditions préalables](#) » page 9.

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

Par exemple :

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

Lorsque vous vous connectez à plusieurs arborescences eDirectory, vous devez veiller à obtenir au moins un certificat de serveur `keys.pfx` pour toutes les arborescences connectées.

7 Copiez le fichier de certificat d'autorité de certification (`.pem`) de votre système de fichiers local vers le conteneur en tant que `/etc/opt/novell/eDirAPI/cert/SSCert.pem` à l'aide de la commande ci-dessous. Pour plus d'informations sur l'obtention du certificat d'autorité de certification, reportez-vous à la section « [Conditions préalables](#) » page 9.

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

Par exemple :

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

Si vous devez vous connecter à plusieurs arborescences eDirectory, reportez-vous à la section « [Arborescences multiples avec Identity Console en tant que Docker](#) » page 24.

- 8 Modifiez le fichier de configuration selon vos besoins et copiez-le (`edirapi.conf`) de votre système de fichiers local vers le conteneur en tant que `/etc/opt/novell/eDirAPI/conf/edirapi.conf` à l'aide de la commande suivante :

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

Par exemple :

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

- 9 Démarrez le conteneur Docker à l'aide de la commande suivante :

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

Par exemple :

```
docker start identityconsole-container
```

REMARQUE : Vous trouverez les fichiers journaux suivants dans le répertoire `/var/lib/docker/volumes/<nom_volume>/_data/eDirAPI/var/log` :

- ♦ `edirapi.log` : ce fichier sert à consigner les différents événements dans `edirapi` et à déboguer les problèmes.
 - ♦ `edirapi_audit.log` : ce fichier sert à consigner les événements d'audit d'`edirapi`. Ce journal respecte le format d'audit CEF.
 - ♦ `container-startup.log` : ce fichier sert à capturer les journaux d'installation du conteneur Docker d'Identity Console.
-

Arborescences multiples avec Identity Console en tant que Docker

Identity Console permet de se connecter à plusieurs arborescences en obtenant un seul certificat CA de l'arborescence.

Par exemple, si vous vous connectez à trois arborescences `eDirectory`, vous devez copier les trois certificats CA dans le conteneur 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
```

Exécutez les commandes suivantes pour redémarrer Identity Console :

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

Déploiement d'Identity Console en mode autonome

- ♦ « [Déploiement d'Identity Console en mode autonome \(non-Docker\)](#) » page 25
- ♦ « [Arborescences multiples avec Identity Console en mode autonome](#) » page 26

Déploiement d'Identity Console en mode autonome (non-Docker)

Cette section décrit la procédure à suivre pour déployer Identity Console en mode autonome :

1 Connectez-vous au [portail de téléchargements et de licences de logiciels \(Software License and Download\)](https://sld.microfocus.com/) (<https://sld.microfocus.com/>) et accédez à la page de téléchargements de logiciels.

2 Sélectionnez les éléments suivants :

- ♦ Produit : eDirectory
- ♦ Nom du produit : eDirectory per User Sub SW E-LTU
- ♦ Version : 9.2

3 Téléchargez la dernière version d'Identity Console.

4 Extrayez le fichier téléchargé dans un dossier.

5 Ouvrez un shell et accédez au dossier dans lequel vous avez extrait la build d'Identity Console.

6 Exécutez la commande suivante lorsque vous êtes connecté en tant qu'utilisateur root ou équivalent :

```
./identityconsole_install
```

7 Lisez l'introduction, puis cliquez sur **ENTER** (ENTRÉE).

8 Cliquez sur « Y » (O) pour accepter le contrat de licence. Tous les paquetages RPM requis sont alors installés sur votre système.

9 Entrez le nom d'hôte (nom de domaine complet) ou l'adresse IP du serveur Identity Console.

10 Entrez le numéro de port d'écoute d'Identity Console. La valeur par défaut est 9000.

11 Entrez l'option d'intégration d'OSP à Identity Console ou pour qu'Identity Console utilise la connexion LDAP.

12 Si vous souhaitez intégrer OSP à Identity Console :

1. Entrez le nom de domaine/l'adresse IP du serveur eDirectory/Coffre-fort d'identité avec le numéro de port LDAPS.

Par exemple :

```
192.168.1.1:636
```

2. Entrez le nom d'utilisateur eDirectory/Coffre-fort d'identité.

Par exemple :

```
cn=admin,ou=org_unit,o=org
```

3. Entrez le mot de passe eDirectory/Coffre-fort d'identité.

4. Entrez de nouveau le mot de passe eDirectory/Coffre-fort d'identité pour le confirmer.

5. Entrez le nom de domaine/l'adresse IP du serveur OSP avec le numéro de port SSL du serveur SSO.

6. Entrez l'ID du client OSP.
 7. Entrez le mot de passe du client OSP.
 8. Entrez le nom de l'arborescence eDirectory/Coffre-fort d'identité.
- 13** Entrez le chemin d'accès au certificat racine approuvé (`SSCert.pem`), y compris le dossier.

Par exemple :

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

REMARQUE : vous devez veiller à ne pas créer de sous-répertoire dans le dossier cert.

- 14** Entrez le chemin d'accès au certificat de serveur (`keys.pfx`), y compris le nom de fichier.

Par exemple :

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

- 15** Entrez le mot de passe du certificat de serveur. Pour confirmer que vous avez entré correctement le mot de passe, entrez à nouveau le mot de passe du certificat de serveur. L'installation démarre.

REMARQUE : Vous trouverez les fichiers journaux suivants dans le répertoire `/var/opt/novell/eDirAPI/log` :

- ♦ `edirapi.log` : ce fichier sert à consigner les différents événements dans `edirapi` et à déboguer les problèmes.
- ♦ `edirapi_audit.log` : ce fichier sert à consigner les événements d'audit d'`edirapi`. Ce journal respecte le format d'audit CEF.
- ♦ `identityconsole_install.log` : ce fichier sert à capturer les journaux d'installation d'Identity Console.

Les journaux correspondant au démarrage/arrêt de processus Identity Console se trouvent dans le fichier `/var/log/messages`.

REMARQUE : NetIQ recommande que, lors de l'installation d'Identity Console et d'eDirectory sur une même machine, celle-ci dispose d'au moins une instance d'eDirectory disponible.

Arborescences multiples avec Identity Console en mode autonome

Lorsque vous vous connectez à plusieurs arborescences eDirectory, vous devez veiller à obtenir un seul certificat CA de l'arborescence.

Par exemple, si vous vous connectez à trois arborescences eDirectory, vous devez copier les trois certificats CA dans le répertoire `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
```

Exécutez l'une des commandes suivantes pour redémarrer Identity Console :

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

ou

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

Identity Console sous Windows en tant que poste de travail

Identity Console peut être lancé sous Windows en tant que poste de travail et nécessite l'exécution des services REST. Par conséquent, lorsqu'il est lancé, un processus eDirAPI s'exécute à l'invite de commande edirapi.exe. Si ce terminal edirapi.exe est fermé, Identity Console ne fonctionne plus.

La procédure ci-dessous explique comment exécuter Identity Console sous Windows.

- 1 Connectez-vous au [portail de téléchargements et de licences de logiciels \(Software License and Download\)](https://sldlogin.microfocus.com/nidp/idff/sso?id=5&sid=0&option=credential&sid=0) (<https://sldlogin.microfocus.com/nidp/idff/sso?id=5&sid=0&option=credential&sid=0>) et accédez à la page de téléchargements de logiciels.
- 2 Sélectionnez les éléments suivants :
 - ◆ Produit : eDirectory
 - ◆ Nom du produit : eDirectory per User Sub SW E-LTU
 - ◆ Version : 9.2
- 3 Téléchargez le fichier IdentityConsole_<version>_workstation_win_x86_64.zip.
- 4 Extrayez le fichier IdentityConsole_<version>_workstation_win_x86_64.zip téléchargé dans un dossier.
- 5 Accédez au dossier extrait
IdentityConsole_150_workstation_win_x86_64\edirAPI\cert, puis copiez le certificat CA racine approuvé SSCert.pem et le certificat de serveur keys.pfx.
Pour obtenir les certificats, reportez-vous à la section « [Configuration système requise et conditions préalables pour le poste de travail](#) » page 16.
Si vous devez vous connecter à plusieurs arborescences eDirectory, reportez-vous à la section « [Arborescences multiples avec Identity Console en mode Poste de travail](#) » page 28.

REMARQUE : le nom du certificat de serveur doit toujours être keys.pfx.

- 6 Accédez au dossier dans lequel la version est extraite, puis double-cliquez sur le fichier run.bat (fichier de traitement par lots Windows).
- 7 Entrez le mot de passe du certificat de serveur (keys.pfx) à l'invite de commande.
Le terminal de processus eDirAPI (edirapi.exe) démarre, et la page de connexion d'Identity Console s'affiche.

REMARQUE :

- ◆ Si le terminal de processus eDirAPI (edirapi.exe) est déjà en cours d'exécution, exécutez identityconsole.exe à partir du dossier dans lequel la version est extraite.

- ♦ Vous trouverez les journaux suivants dans
`\IdentityConsole_150_workstation_win_x86_64\edirapi\log`:
`edirapi.log` : ce fichier sert à consigner les différents événements dans edirapi et à déboguer les problèmes.
`edirapi_audit.log` : ce fichier sert à consigner les événements d'audit d'edirapi. Ce journal respecte le format d'audit CEF.
 - ♦ La connexion OSP n'est pas prise en charge en mode Poste de travail.
 - ♦ L'écoute du poste de travail Identity Console se fait uniquement sur le port 9000. Ne modifiez pas le fichier `edirapi_win.conf`.
-

Arborescences multiples avec Identity Console en mode Poste de travail

Identity Console permet de se connecter à plusieurs arborescences en obtenant un seul certificat CA de l'arborescence.

- 1 Fermez le poste de travail Identity Console et le terminal eDirAPI.
- 2 Copiez les certificats CA `SSCert.pem` à l'emplacement
`IdentityConsole_150_workstation_win_x86_64\edirapi\cert`.
Par exemple, si vous souhaitez vous connecter à trois arborescences eDirectory, copiez respectivement les certificats CA en tant que `SSCert1.pem`, `SSCert2.pem` et `SSCert3.pem`.
- 3 Accédez au dossier dans lequel la version est extraite, puis double-cliquez sur le fichier `run.bat` (fichier de traitement par lots Windows).
- 4 Entrez le mot de passe `keys.pfx` à l'invite du terminal, puis connectez-vous à l'arborescence eDirectory souhaitée.

Arrêt et redémarrage d'Identity Console

- ♦ [« Arrêt et redémarrage d'Identity Console en tant que conteneur Docker » page 28](#)
- ♦ [« Arrêt et redémarrage d'Identity Console en mode autonome » page 29](#)
- ♦ [« Fermeture et redémarrage du poste de travail Identity Console » page 29](#)

Arrêt et redémarrage d'Identity Console en tant que conteneur Docker

Pour arrêter Identity Console, exécutez la commande suivante :

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

Pour redémarrer Identity Console, exécutez la commande suivante :

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

Pour démarrer Identity Console, exécutez la commande suivante :

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

Arrêt et redémarrage d'Identity Console en mode autonome

Pour arrêter Identity Console, exécutez l'une des commandes suivantes :

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

ou

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

Pour redémarrer Identity Console, exécutez l'une des commandes suivantes :

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

ou

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

Pour démarrer Identity Console, exécutez l'une des commandes suivantes :

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

ou

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

Fermeture et redémarrage du poste de travail Identity Console

Pour fermer l'application et le processus, procédez comme suit :

- 1 Fermez l'application Windows de bureau Identity Console.
- 2 Arrêtez le processus eDirAPI en fermant le terminal de processus eDirAPI.

Pour relancer le poste de travail Identity Console, accédez au dossier dans lequel la version est extraite, puis double-cliquez sur le fichier `run.bat` (fichier de traitement par lots Windows).

REMARQUE : si le terminal de processus eDirAPI est déjà en cours d'exécution, exécutez `identityconsole.exe` à partir du dossier dans lequel la version est extraite pour relancer le poste de travail Identity Console.

Gestion de la persistance des données

En plus des conteneurs Identity Console, des volumes sont également créés pour la persistance des données. Pour utiliser les paramètres de configuration d'un ancien conteneur qui utilise les volumes, procédez comme suit :

- 1 Arrêtez le conteneur Docker actuel à l'aide de la commande suivante :

```
docker stop identityconsole-container
```

- 2 Créez le second conteneur en utilisant les données d'application de l'ancien conteneur stockées dans le volume Docker (`edirapi-volume-1`) :

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

3 Démarrez le second conteneur à l'aide de la commande suivante :

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

4 (Facultatif) Vous pouvez à présent supprimer le premier conteneur à l'aide de la commande suivante :

```
docker rm identityconsole-container
```

Déploiement d'Identity Console dans Azure Kubernetes Service

Azure Kubernetes Service (AKS) est un service Kubernetes géré qui vous permet de déployer et de gérer des grappes. Cette section contient les procédures suivantes :

Déploiement d'Identity Console dans une grappe AKS

Cette section explique les procédures ci-dessous à suivre pour déployer Identity Console dans une grappe AKS :

- ♦ [« Création d'une instance Azure Container Registry \(ACR\) » page 30](#)
- ♦ [« Définition d'une grappe Kubernetes » page 31](#)
- ♦ [« Création d'une adresse IP publique SKU standard » page 32](#)
- ♦ [« Configuration de Cloud Shell et connexion à la grappe Kubernetes » page 32](#)
- ♦ [« Déploiement de l'application » page 32](#)

Création d'une instance Azure Container Registry (ACR)

Une instance Azure Container Registry (ACR) est un registre privé Azure destiné aux images de conteneur Docker.

Pour obtenir les étapes détaillées, reportez-vous à la section [Créer un registre de conteneurs dans l'article Créer un registre de conteneurs Azure à l'aide du portail Azure](#) ou procédez comme suit pour créer une instance Azure Container Registry (ACR) :

1. Connectez-vous au [portail Azure](#).
2. Accédez à **Create a resource** (Créer une ressource) > **Containers** (Conteneurs) > **Container Registry** (Registre de conteneurs).
3. Dans l'onglet **Basics** (Concepts de base), spécifiez les valeurs des options **Resource Group** (Groupe de ressources) et **Registry name** (Nom du registre). Le nom de registre doit être unique dans Azure et contenir minimum 5 et maximum 50 caractères alphanumériques.
Acceptez les valeurs par défaut des paramètres restants.
4. Cliquez sur **Review + create** (Réviser + créer).
5. Cliquez sur **Create** (Créer).
6. Connectez-vous à Azure CLI, puis exécutez la commande suivante pour vous connecter à Azure Container Registry :

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

Par exemple :

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

7. Récupérez le serveur de connexion de l'instance Azure Container Registry à l'aide de la commande suivante :

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

Par exemple :

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

8. Marquez l'image locale d'Identity Console avec le nom du serveur de connexion ACR (registryname.azurecr.io) à l'aide de la commande suivante :

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

Par exemple :

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

9. Distribuez l'image marquée dans le registre.

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

Par exemple :

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

10. Récupérez la liste d'images dans le registre à l'aide de la commande suivante :

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

Définition d'une grappe Kubernetes

Créez une ressource de service Kubernetes à l'aide du portail Azure ou de CLI.

Pour obtenir la procédure détaillée à suivre pour créer une ressource de service Kubernetes dans Azure avec un nœud, reportez-vous à la section [Créer un cluster AKS](#) dans le [guide de démarrage rapide Azure](#).

REMARQUE :

- ♦ Veillez à sélectionner Azure CNI en tant que réseau.
 - ♦ Sélectionnez le réseau virtuel existant (sur lequel le serveur eDirectory est déployé dans le sous-réseau).
 - ♦ Sélectionnez le registre de conteneurs existant dans lequel l'image Identity Console est disponible.
-

Création d'une adresse IP publique SKU standard

Une ressource d'adresse IP publique sous le groupe de ressources de grappe Kubernetes fait office d'adresse IP d'équilibreur de charge pour l'application.

Pour obtenir la procédure détaillée, reportez-vous à l'article [Créer une adresse IP publique à l'aide du portail Azure](#) dans le guide [Créer une adresse IP publique - Portail](#).

Configuration de Cloud Shell et connexion à la grappe Kubernetes

Utilisez Cloud Shell, disponible sur le portail Azure pour toutes les opérations.

Pour configurer Cloud Shell sur le portail Azure, reportez-vous à la section [Démarrer Cloud Shell](#) dans le guide [Bash - Démarrage rapide](#) ou procédez comme suit pour configurer Cloud Shell et vous connecter à la grappe Kubernetes :

1. Dans le portail Azure, cliquez sur le bouton  pour ouvrir Cloud Shell.

REMARQUE : pour gérer une grappe Kubernetes, utilisez le client de ligne de commande Kubernetes, `kubectl`. `kubectl` est déjà installé si vous utilisez Azure Cloud Shell.

2. Configurez `kubectl` pour vous connecter à votre grappe Kubernetes à l'aide de la commande suivante :

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

Par exemple :

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

3. Vérifiez la liste des nœuds de la grappe à l'aide de la commande suivante :

```
kubectl get nodes
```

Déploiement de l'application

Pour déployer Identity Console, vous pouvez utiliser les exemples de fichiers `idc-services.yaml`, `idc-statefulset.yaml`, `idc-storageclass.yaml` et `idc-pvc.yaml`.

Vous pouvez également créer vos propres fichiers yaml selon vos besoins.

1. Créez une ressource de classe de stockage à l'aide de la commande suivante :

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

Par exemple :

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

(Facultatif) Pour plus d'informations sur la création et l'utilisation dynamiques d'un volume de persistance avec un partage de fichiers Azure, reportez-vous à l'article [Créer et utiliser un volume persistant de manière dynamique avec Azure Files dans Azure Kubernetes Service \(AKS\)](#).

Voici un exemple de fichier de ressource de classe de stockage :

```
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
~
```

Une ressource de classe de stockage active le provisioning de stockage dynamique. Elle sert à définir la manière dont un partage de fichiers Azure est créé.

2. Consultez les détails de la classe de stockage à l'aide de la commande suivante :

```
kubectl get sc
```

3. Créez une ressource pvc à l'aide du fichier `idc-pvc.yaml` :

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

Par exemple :

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

Voici un exemple de fichier de ressource pvc :

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

Une ressource de revendication de volume persistant crée le partage de fichiers. Une revendication de volume persistant (PVC) utilise l'objet Classe de stockage pour provisionner dynamiquement un partage de fichiers Azure.

4. Téléchargez `edirapi.conf`, le certificat CA et le certificat de serveur dans Cloud Shell.

Cliquez sur l'icône de **téléchargement de fichiers**  dans Cloud Shell et téléchargez les fichiers `edirapi.conf`, `SSCert.pem` et `keys.pfx`.

REMARQUE : le fichier `edirapi.conf` contient un paramètre « `origin` » qui permet d'indiquer l'adresse IP à utiliser pour accéder à l'application Identity Console. (Utilisez l'adresse IP créée dans la section « [Création d'une adresse IP publique SKU standard](#) » page 32.)

Le déploiement d'Identity Console nécessite un certificat de serveur (`keys.pfx`).

Lors de la création du certificat de serveur, veillez à fournir un nom DNS valide dans l'alias de l'objet.

Étapes de création d'un nom DNS valide :

Un pod standard déployé à l'aide de StatefulSet possède un nom DNS au format `{nom_statefulset}-{ordinal}.{nom_service}.{espace_noms}.svc.cluster.local`

- ♦ Si le nom StatefulSet du fichier `idconsole-statefulset.yaml` est `idconsole-app`, `nom_statefulset = idconsole-app`.
- ♦ S'il s'agit du 1er pod, `ordinal = 0`.
- ♦ Si vous définissez `nom_service` dans le fichier `idconsole-statefulset.yaml` en tant que `idconsole`, `nom_service = idconsole`.
- ♦ S'il s'agit de l'espace de noms par défaut, `espace_noms = default`.

Sortie : `idconsole-app-0.idconsole.default.svc.cluster.local`

5. Créez une ressource `configmap` dans la grappe Kubernetes qui stocke les fichiers de configuration avec les certificats.

Avant d'exécuter la commande, veillez à ce que les fichiers (`edirapi.conf`, `SSCert.pem` et `keys.pfx`) soient présents dans le répertoire.

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

Par exemple :

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

6. Consultez les détails de l'objet `configmap` à l'aide de la commande `kubectl describe` :

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

Par exemple :

```
kubectl describe configmap config-data
```

7. Créez une ressource `StatefulSet` pour déployer le conteneur.

Exécutez la commande suivante pour déployer le conteneur :

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

Par exemple :

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

Voici un exemple de fichier de ressource `StatefulSet` :

```

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: pvcforsec

```

8. Exécutez la commande suivante pour vérifier l'état du pod déployé :

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

9. Créez une ressource de service du type loadBalancer.

Le type de service spécifié dans le fichier yaml est loadBalancer.

Créez une ressource de service à l'aide de la commande suivante :

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

Par exemple :

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

Voici un exemple de fichier de ressource de service :

```
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
```

Vérifiez l'adresse IP externe (ou loadBalancerIP) à l'aide de la commande suivante :

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

10. Lancez l'URL à l'aide de l'adresse IP externe (ou de l'adresse loadBalancerIP).

Par exemple :

```
https://<adresse_IP_externe>:9000/identityconsole
```

Modification du certificat de serveur

Cette section contient des informations sur la modification du certificat de serveur dans le conteneur Docker et Identity Console en mode autonome.

- ♦ [« Modification du certificat de serveur dans le conteneur Docker » page 36](#)
- ♦ [« Modification du certificat de serveur dans Identity Console en mode autonome » page 37](#)

Modification du certificat de serveur dans le conteneur Docker

Pour modifier le certificat de serveur dans le conteneur Docker, procédez comme suit :

- 1 Exécutez la commande suivante pour copier le nouveau certificat de serveur à un emplacement du conteneur.

Par exemple :

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

- 2 Connectez-vous au conteneur à l'aide de la commande suivante :

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

- 3 Exécutez NLPcert pour stocker les clés en tant que pseudo-utilisateur :

```
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 Quittez la console du conteneur à l'aide de la commande suivante :

```
exit
```

- 5 Redémarrez le conteneur à l'aide de la commande suivante :

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

Modification du certificat de serveur dans Identity Console en mode autonome

Pour modifier le certificat de serveur dans le conteneur autonome, procédez comme suit :

- 1 Exécutez NLPCERT pour stocker les clés:

```
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 Redémarrez Identity Console :

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

3 Mise à niveau d'Identity Console

Ce chapitre décrit la procédure à suivre pour effectuer la mise à niveau d'Identity Console vers la version la plus récente. Pour préparer le déploiement, consultez les conditions préalables et la configuration système requise mentionnées au [Chapitre 1, « Planification de l'installation d'Identity Console », page 9](#).

Cette section contient les procédures suivantes :

- ♦ « [Mise à niveau d'Identity Console en tant que conteneur Docker](#) » page 39
- ♦ « [Mise à niveau d'Identity Console en mode autonome \(non-Docker\)](#) » page 41
- ♦ « [Mise à niveau du conteneur OSP](#) » page 42

Mise à niveau d'Identity Console en tant que conteneur Docker

Quand une nouvelle version de l'image Identity Console est disponible, l'administrateur peut effectuer une mise à niveau pour déployer le conteneur à l'aide de la dernière version d'Identity Console. Veillez à assurer le stockage persistant de toutes les données nécessaires liées à l'application dans les volumes Docker avant d'effectuer une mise à niveau. Procédez comme suit pour effectuer la mise à niveau d'Identity Console à l'aide d'un conteneur Docker :

- 1 Téléchargez et chargez la dernière version de l'image Docker à partir du [portail de téléchargements et de licences de logiciels \(Software License and Download\) \(https://sld.microfocus.com/\)](https://sld.microfocus.com/), puis effectuez la procédure à suivre pour installer la dernière version d'Identity Console mentionnée à la section « [Déploiement d'Identity Console](#) » page 19.
- 2 Une fois la dernière image Docker chargée, arrêtez votre conteneur Docker actuel à l'aide de la commande suivante :

```
docker stop identityconsole-container
```

- 3 (Facultatif) Récupérez la sauvegarde du volume partagé.
- 4 Supprimez le conteneur Identity Console existant en exécutant la commande suivante :

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

Par exemple :

```
docker rm identityconsole-container
```

- 5 (Facultatif) Supprimez l'image Docker obsolète d'Identity Console en exécutant la commande suivante :

```
docker rmi identityconsole
```

- 6 Créez le conteneur Docker Identity Console à l'aide de la commande suivante :

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

Par exemple :

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

REMARQUE

- ♦ Vous pouvez accepter le contrat de licence EULA en affectant la valeur « Y » à la variable d'environnement ACCEPT_EULA. Vous pouvez également accepter ce contrat dans l'invite qui s'affiche au démarrage du conteneur à l'aide de l'option `-it` de la commande Docker `create` en mode interactif.
- ♦ Le paramètre `--volume` de la commande ci-dessus crée un volume destiné au stockage des données de configuration et de journal. Dans ce cas, nous avons créé un exemple de volume nommé `IDConsole-volume`.

-
- 7 Copiez le fichier de certificat de serveur de votre système de fichiers local vers le nouveau conteneur en tant que `/etc/opt/novell/eDirAPI/cert/keys.pfx` à l'aide de la commande suivante :

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

Par exemple :

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

Lorsque vous vous connectez à plusieurs arborescences eDirectory, vous devez veiller à copier au moins un certificat de serveur `keys.pfx` pour toutes les arborescences connectées.

- 8 Copiez le fichier de certificat d'autorité de certification (`.pem`) de votre système de fichiers local vers le nouveau conteneur en tant que `/etc/opt/novell/eDirAPI/cert/SSCert.pem` à l'aide de la commande suivante :

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

Par exemple :

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

Lorsque vous vous connectez à plusieurs arborescences eDirectory, vous devez veiller à obtenir un seul certificat CA pour toutes les arborescences connectées. Par exemple, si vous vous connectez à trois arborescences eDirectory, vous devez copier les trois certificats CA dans le conteneur 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
```

REMARQUE : À partir d'Identity Console 1.4, le fichier de configuration (`edirapi.conf`) n'inclut pas explicitement les paramètres « `ldapuser` », « `ldappassword` » et « `ldapserver` ». La valeur du paramètre « `vbcert` » doit inclure le chemin d'accès au répertoire des certificats racine approuvés. Par exemple : `bcert = "/etc/opt/novell/eDirAPI/cert/"`. De plus, le paramètre « `origin` » est indépendant du paramètre « `check-origin` » et est obligatoire lorsque la configuration DNS est utilisée.

- 9 Copiez le fichier de configuration (`edirapi.conf`) de votre système de fichiers local vers le nouveau conteneur en tant que `/etc/opt/novell/eDirAPI/conf/edirapi.conf` à l'aide de la commande suivante :

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

Par exemple :

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

- 10 Démarrez le second conteneur à l'aide de la commande suivante :

```
docker start identityconsole-container
```

- 11 Consultez l'état du conteneur en cours d'exécution à l'aide de la commande suivante :

```
docker ps -a
```

Mise à niveau d'Identity Console en mode autonome (non-Docker)

Cette section décrit la procédure à suivre pour mettre à niveau Identity Console en mode autonome :

- 1 Téléchargez `IdentityConsole_<version>_Containers.tar.gz` à partir du [portail de téléchargements et de licences de logiciels \(Software License and Download\) \(https://sld.microfocus.com/\)](https://sld.microfocus.com/).
- 2 Connectez-vous à ce portail, accédez à la page de téléchargements de logiciels, puis cliquez sur **Download** (Télécharger).
- 3 Sélectionnez les éléments suivants : Produit : **eDirectory** > Nom du produit : **eDirectory per User Sub SW E-LTU** > Version : **9.2**.
- 4 Téléchargez la dernière version d'Identity Console.
- 5 Extrayez le fichier téléchargé à l'aide de la commande suivante :

```
tar -zxvf IdentityConsole_<version>_Linux.tar.gz
```
- 6 Accédez au dossier dans lequel vous avez extrait la version d'Identity Console.

- 7 Copiez dans un dossier tous les certificats racine approuvés des arborescences eDirectory auxquelles vous souhaitez vous connecter. Pour copier le certificat racine approuvé dans le dossier, exécutez la commande suivante :

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

Par exemple :

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

- 8 Exécutez la commande suivante :

```
./identityconsole_install
```

- 9 Indiquez le chemin d'accès au dossier des certificats racine approuvés utilisé à l'**étape 4**.

- 10 La mise à niveau d'Identity Console est effectuée.

Mise à niveau du conteneur OSP

Pour mettre à niveau le conteneur OSP, procédez comme suit :

- 1 Téléchargez et chargez la dernière version de l'image OSP à partir du [portail de téléchargements et de licences de logiciels \(Software License and Download\)](https://sld.microfocus.com/) (<https://sld.microfocus.com/>).

Par exemple :

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

- 2 Une fois la dernière image OSP chargée, arrêtez votre conteneur OSP actuel à l'aide de la commande suivante :

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

- 3 (Facultatif) Récupérez la sauvegarde du volume partagé.

- 4 Supprimez le conteneur OSP existant en exécutant la commande suivante :

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

Par exemple :

```
docker rm OSP_Container
```

- 5 Accédez au répertoire qui contient le fichier Keystore (`tomcat.ks`) et le fichier de propriétés en mode silencieux, supprimez le fichier Keystore existant (`tomcat.ks`) et conservez le dossier OSP existant. Générez un nouveau fichier Keystore (`tomcat.ks`) selon une taille de clé de 2 048. Pour plus d'informations, reportez-vous à l'**étape 4** de la section [Déploiement du conteneur OSP](#) du [Guide d'installation d'Identity Console](#).

- 6 Déployez le conteneur à l'aide de la commande suivante :

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

Par exemple :

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

4 Désinstallation d'Identity Console

Ce chapitre décrit la procédure de désinstallation d'Identity Console :

- ♦ « Procédure de désinstallation pour l'environnement Docker » page 45
- ♦ « Procédure de désinstallation pour l'installation autonome d'Identity Console (non-Docker) » page 45

Procédure de désinstallation pour l'environnement Docker

Pour désinstaller le conteneur Docker Identity Console, procédez comme suit :

- 1 Arrêtez le conteneur Identity Console :

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

- 2 Exécutez la commande suivante pour supprimer le conteneur Docker Identity Console :

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

- 3 Exécutez la commande suivante pour supprimer l'image Docker :

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

- 4 Supprimez le volume Docker :

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

REMARQUE : Si vous supprimez le volume, les données sont également supprimées de votre serveur.

Procédure de désinstallation pour l'installation autonome d'Identity Console (non-Docker)

Pour désinstaller l'installation autonome d'Identity Console, procédez comme suit :

- 1 Accédez au répertoire `/usr/bin` sur la machine sur laquelle Identity Console est installé.
- 2 Exécutez la commande suivante :

```
./identityconsoleUninstall
```
- 3 La désinstallation d'Identity Console est effectuée.

REMARQUE : lorsque eDirectory ou un autre produit NetIQ est installé sur la machine, vous devez désinstaller manuellement *nici* et *openssl*.
