



Identity Console

Guia de Instalação

Setembro de 2022

Informações legais

Para saber mais sobre informações legais, marcas registradas, isenções de responsabilidades, garantias, exportação e outras restrições de uso, direitos restritos do Governo dos EUA, política de patente e conformidade de FIPS, consulte <https://www.netiq.com/company/legal>.

Copyright © 2022 NetIQ Corporation. Todos os direitos reservados.

Índice

Sobre este manual e a biblioteca	5
Sobre a NetIQ Corporation	7
1 Planejando a instalação do Identity Console	11
Requisitos do sistema e pré-requisitos para instalação do Docker	11
Requisitos do sistema	11
Pré-requisitos	11
Configurando seu ambiente	13
Requisitos do sistema e pré-requisitos para instalação autônoma (não Docker)	15
Requisitos do sistema	16
(Opcional) Pré-requisito para configuração do OSP	17
Requisitos do sistema e pré-requisitos para estação de trabalho	18
Requisitos do sistema	18
Verificação de autenticação RPM	19
2 Implantando o Identity Console	21
Recomendações de segurança	21
Implantação do Identity Console como container do Docker	22
Implantação do container do OSP	22
Implantação do Identity Console como um container do Docker	24
Multiárvore com o Identity Console como Docker	26
Implantação do Identity Console independente	26
Implantação do Identity Console independente (não Docker)	26
Multiárvore com Identity Console autônomo	28
Identity Console no Windows como estação de trabalho	28
Multiárvore com o Identity Console como estação de trabalho	29
Parada e reinício do Identity Console	30
Parada e reinício do Identity Console como container do Docker	30
Parada e reinício do Identity Console independente	30
Fechar e reiniciar a Identity Console Workstation	31
Gerenciamento da Persistência de Dados	31
Implantação do Identity Console nos Serviços de Kubernetes do Azure	31
Implantação do Identity Console no cluster do AKS	31
Modificando o certificado de servidor	38
Modificando o certificado de servidor no container do Docker	38
Modificando o certificado de servidor no Identity Console autônomo	38
3 Atualização do Identity Console	39
Upgrade do Identity Console como container do Docker	39
Atualizando o Identity Console autônomo (não Docker)	41
Upgrade do container do OSP	42

4	Desinstalação do Identity Console	43
	Procedimento de desinstalação para ambiente Docker	43
	Procedimento de desinstalação do Identity Console independente (não Docker)	43

Sobre este manual e a biblioteca

O *Guia de Instalação do Identity Console* fornece informações sobre como instalar e gerenciar o produto NetIQ Identity Console (Identity Console). Este manual define a terminologia e inclui cenários de implementação.

Público-alvo

Este guia é dirigido a administradores de rede.

Outras informações na biblioteca

A biblioteca fornece os seguintes recursos informativos:

Installation Guide (Guia de Instalação)

Descreve como instalar e fazer upgrade do Identity Console. Ele é dirigido aos administradores de rede.

Sobre a NetIQ Corporation

Nós somos uma empresa global de software corporativo com foco nos três desafios constantes do seu ambiente — mudança, complexidade e risco — e em como podemos ajudar você a controlá-los.

Nosso ponto de vista

Adaptar-se a mudanças e gerenciar complexidades e riscos não são novidades

De fato, dentre todos os desafios que você enfrenta, estas são provavelmente as variáveis mais proeminentes, que impedem que você obtenha o controle de que precisa para gerenciar, monitorar e medir de forma segura seus ambientes de computação físicos, virtuais e em nuvem.

Habilitando serviços essenciais para empresas de forma mais rápida e eficiente

Nós acreditamos que fornecer o máximo possível de controle para organizações de TI é a única maneira de possibilitar uma entrega de serviços mais oportuna e econômica. Pressões persistentes, como mudanças e complexidade, só continuarão a aumentar conforme as organizações continuarem a mudar e as tecnologias necessárias para gerenciá-las se tornarem inerentemente mais complexas.

Nossa filosofia

Vender soluções inteligentes, não somente software

Visando providenciar um controle seguro, primeiro nos certificamos de que entendemos os cenários do mundo real, nos quais organizações de TI como a sua operam todos os dias. Somente dessa maneira podemos desenvolver soluções de TI práticas e inteligentes, que geram com sucesso resultados comprovados e mensuráveis. E isso é muito mais recompensador do que simplesmente vender software.

Promover seu sucesso é nossa paixão

O seu sucesso encontra-se no âmbito de como fazemos negócios. Desde os primeiros esboços até a implantação de um produto, nós compreendemos que você precisa de soluções de TI que funcionem bem e se integrem perfeitamente a seus investimentos existentes, de suporte contínuo e treinamento pós-implantação, além de alguém com quem a colaboração seja verdadeiramente fácil, o que sabemos que não é muito comum. Em última análise, quando você é bem-sucedido, todos nós somos bem-sucedidos.

Nossas soluções

- ♦ Governança de acesso e identidade
- ♦ Gerenciamento de acesso
- ♦ Gerenciamento de segurança

- ♦ Gerenciamento de aplicativos e sistemas
- ♦ Gerenciamento de carga de trabalho
- ♦ Gerenciamento de serviços

Entrando em contato com o Suporte a vendas

Para esclarecer dúvidas sobre produtos, preços e recursos, entre em contato com seu parceiro local. Se não for possível entrar em contato com seu parceiro, entre em contato com nossa equipe de Suporte a vendas.

Mundial:	www.netiq.com/about_netiq/officelocations.asp
Estados Unidos e Canadá:	1-888-323-6768
E-mail:	info@netiq.com
Site na Web:	www.netiq.com

Entrando em contato com o Suporte técnico

Para questões sobre produtos específicos, entre em contato com nossa equipe de Suporte técnico.

Mundial:	www.netiq.com/support/contactinfo.asp
América do Norte e do Sul:	1-713-418-5555
Europa, Oriente Médio e África:	+353 (0) 91-782 677
E-mail:	support@netiq.com
Site na Web:	www.netiq.com/support

Entrando em contato com o Suporte de documentação

Nosso objetivo é fornecer uma documentação que atenda às suas necessidades. Se você tem sugestões de melhorias, clique em **Add Comment** (Adicionar Comentário) na parte inferior de qualquer página nas versões em HTML da documentação publicada em <https://www.netiq.com/pt-br/documentation/>. Você também pode enviar um e-mail para Documentation-Feedback@netiq.com. Nós valorizamos sua opinião e aguardamos seu contato.

Entrando em contato com a comunidade online de usuários

A Qmunity, a comunidade online da NetIQ, é uma rede colaborativa que conecta você, seus colegas e os especialistas da NetIQ. Fornecendo mais informações imediatas, links para recursos úteis e acesso aos especialistas da NetIQ, a Qmunity ajuda a garantir que você domine os conhecimentos de que precisa para utilizar todo o potencial dos investimentos de TI dos quais depende. Para obter mais informações, visite <http://community.netiq.com>.

1 Planejando a instalação do Identity Console

Este capítulo explica os requisitos do sistema e os pré-requisitos para instalar o Identity Console. Como o Identity Console pode ser executado tanto como um container do Docker quanto como aplicativo autônomo, consulte as respectivas seções para requisitos do sistema e pré-requisitos para ambos os tipos de instalação.

Observação: O Identity Console suporta o eDirectory 9.2.4 HF2, o Identity Manager Engine 4.8.3 HF2 e as respectivas versões posteriores. Você precisa fazer upgrade das suas instâncias do eDirectory e do Identity Manager Engine antes de usar o Identity Console.

- ♦ [“Requisitos do sistema e pré-requisitos para instalação do Docker”](#) na página 11
- ♦ [“Requisitos do sistema e pré-requisitos para instalação autônoma \(não Docker\)”](#) na página 15
- ♦ [“Requisitos do sistema e pré-requisitos para estação de trabalho”](#) na página 18
- ♦ [“Verificação de autenticação RPM”](#) na página 19

Requisitos do sistema e pré-requisitos para instalação do Docker

Esta seção explica os requisitos do sistema e os pré-requisitos para instalar o Identity Console como container do Docker.

- ♦ [“Requisitos do sistema”](#) na página 11
- ♦ [“Pré-requisitos”](#) na página 11
- ♦ [“Configurando seu ambiente”](#) na página 13

Requisitos do sistema

Como o Identity Console pode ser executado como um container do Docker, para obter mais informações sobre requisitos do sistema e plataformas suportadas para instalação do Identity Console, consulte a [Documentação do Docker](#).

Pré-requisitos

- Instale o Docker 20.10.9-ce ou posterior. Para obter mais informações sobre como instalar o Docker, consulte a [Instalação do Docker](#).
- Você precisa obter um certificado de servidor pkcs12 com a chave privada para criptografar/descriptografar a troca de dados entre o servidor Identity Console e o servidor de backend. Esse certificado de servidor é usado para proteger a conexão http. Você pode usar certificados do servidor gerados por qualquer CA externa. Para obter mais informações, consulte [Creating](#)

[Server Certificate Objects](#) (Criando objetos certificação do servidor). O certificado de servidor deve conter o Nome Alternativo do Assunto com endereço IP e DNS do servidor do Identity Console. Uma vez criado o objeto certificação do servidor, você precisa exportá-lo em formato .pfx.

- ❑ Você precisa obter um certificado de CA em formato .pem para validar a autenticação de CA dos certificados de servidor obtidos na etapa anterior. Este certificado rootCA também garante o estabelecimento de uma comunicação LDAP segura entre o cliente e o servidor do Identity Console. Por exemplo, você pode obter o certificado de CA do eDirectory (SSCert.pem) de /var/opt/novell/eDirectory/data/SSCert.pem.

- ❑ (Opcional) Ao utilizar o One SSO Provider (OSP), você pode habilitar a autenticação de login único para os seus usuários no portal do Identity Console. Você precisa instalar o OSP antes de instalar o Identity Console. Para configurar o OSP para o Identity Console, siga os prompts na tela e forneça os valores necessários para os parâmetros de configuração. Para obter mais informações, consulte [“Implantação do container do OSP” na página 22](#). Para registrar o Identity Console em um servidor OSP existente, você deve adicionar manualmente o seguinte ao arquivo ism-configuration.properties na pasta /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
```

Observação: Com o OSP, você pode se conectar a apenas uma árvore do eDirectory, pois o OSP não suporta várias árvores do eDirectory.

- ❑ Verifique se você tem uma entrada DNS adequada disponível para a sua máquina host em /etc/hosts com um nome completo do host.
- ❑ Se você quiser usar o Identity Console no browser Edge, precisará fazer download da versão mais recente do Microsoft Edge para obter a funcionalidade completa.

Observação: Ao usar o Identity Console no Mozilla Firefox, a operação pode falhar com a mensagem de erro Incompatibilidade de Origem. Para solucionar o problema, execute as seguintes etapas:

- 1 Atualize o Firefox para a versão mais recente.
 - 2 Especifique about:config no campo URL do Firefox e pressione Enter.
 - 3 Pesquise Origin.
 - 4 Clique duas vezes em network.http.SendOriginHeader e mude o valor para 1.
-

Configurando seu ambiente

Pode ser necessário criar um arquivo de configuração contendo determinados parâmetros. Se você quiser configurar o Identity Console com o OSP, você precisará especificar os parâmetros específicos do OSP no arquivo de configuração. Por exemplo, crie o arquivo `edirapi.conf` abaixo com parâmetros do OSP:

Observação: Você deve fornecer o nome da árvore do eDirectory no campo `osp-redirect-url`.

```
listen = ":9000"
ldapservers = "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"
```

Caso você queira configurar o Identity Console sem o OSP, crie um arquivo de configuração conforme mostrado abaixo, sem os parâmetros do OSP:

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

Observação: Quando você quiser configurar o Identity Console com várias árvores do eDirectory, poderá ignorar os parâmetros `ldapservers`, `ldapuser` e `ldappassword` e criar o arquivo de configuração.

Tabela 1-1 Descrição dos parâmetros de configuração no arquivo de configuração

Parâmetros de configuração	Descrição
<code>listen</code>	Especifique 9000 como a porta de escuta do servidor do Identity Console dentro do container.
<code>ldapservers</code>	Especifique o IP do servidor de host do eDirectory e o número da porta.

Parâmetros de configuração	Descrição
ldapuser	Especifique o nome de usuário do eDirectory. Este parâmetro é usado como uma credencial para iniciar chamadas LDAP para eDirectory usando controle de autorização por proxy no caso de login OSP. O usuário LDAP precisa ter direitos de supervisor na árvore do eDirectory.
ldappassword	Especifique a senha do usuário LDAP.
pfpassword	Especifique a senha ou o arquivo de certificado de servidor pkcs12.
ospmode	Especifique <code>true</code> para integrar o OSP ao Identity Console. Se você definir como <code>false</code> , o Identity Console usará o login do LDAP.
osp-token-endpoint	Este URL é usado para buscar certos atributos do servidor OSP para verificar a validade do token de autenticação.
osp-authorize-url	Este URL é usado pelo usuário para fornecer credenciais para obter um token de autenticação.
osp-logout-url	Use este URL para terminar a sessão entre o usuário e o servidor OSP.
osp-redirect-url	O servidor OSP redireciona o usuário para este URL após conceder o token de autenticação. Observação: Especifique o nome da árvore do eDirectory em minúsculas ao configurar o Identity Console. Caso o nome da árvore não esteja especificado em minúsculas, o login no servidor Identity Console pode falhar.
osp-client-id	Especifique o ID do cliente OSP fornecido no momento do registro do Identity Console no OSP.
ospclientpass	Especifique a senha do cliente OSP fornecida no momento do registro do Identity Console no OSP.
ospcert	Especifique a localização do certificado CA do servidor OSP.
bcert	Especifique a localização do certificado de CA do Identity Console.
loglevel	Especifique os níveis de registro que você deseja incluir no arquivo de registro. Esse parâmetro pode ser definido como <code>"fatal"</code> , <code>"error"</code> , <code>"warn"</code> ou <code>"info"</code> .
check-origin	Se estiver definido como <code>true</code> , o servidor do Identity Console comparará o valor de origem das solicitações. As opções disponíveis são <code>true</code> ou <code>false</code> . O parâmetro <code>origin</code> é obrigatório mesmo se o valor do parâmetro <code>check-origin</code> é definido como <code>false</code> quando a configuração de DNS é usada.

Parâmetros de configuração	Descrição
origin	O Identity Console compara o valor de origem das solicitações com os valores especificados neste campo. Observação: A partir do Identity Console 1.4, esse parâmetro é independente do parâmetro <i>check-origin</i> e é obrigatório se a configuração de DNS é usada.
maxclients	Número máximo de clientes simultâneos que podem acessar o IDConsole. Qualquer cliente adicional além desse limite precisa esperar na fila.

Observação

- ♦ O parâmetro de configuração `ospmode` deve ser usado apenas se você planeja integrar o OSP junto com o Identity Console.
- ♦ Se o Identity Applications (Identity Apps) for configurado no modo cluster na configuração do Identity Manager, você deverá fornecer o nome DNS do servidor do balanceador de carga nos campos `osp-token-endpoint`, `osp-authorize-url` e `osp-logout-url` no arquivo de configuração. Se você fornecer os detalhes do servidor OSP nesses campos, o login do Identity Console falhará.
- ♦ Se o Identity Console estiver configurado com a mesma instância do OSP que o Identity Apps e o Identity Reporting, o Login Único (serviço de autenticação) entrará em vigor quando você estiver efetuando login no portal do Identity Console.
- ♦ O URL HTTPS do OSP deve ser validado com certificados contendo chave de 2048 bits ou mais, com o Identity Console 1.4 ou posterior.
- ♦ Se você quiser restringir o acesso de domínios diferentes ao portal do Identity Console, defina o parâmetro `samesitecookie` como `strict`. Se você quiser permitir o acesso de domínios diferentes ao portal do Identity Console, defina o parâmetro `samesitecookie` como `lax`. Se o parâmetro não for especificado durante a configuração, as configurações do browser serão seguidas por padrão.

Quando estiver pronto com o arquivo de configuração, continue a implantação do container. Para obter mais informações, consulte [“Implantação do Identity Console como container do Docker”](#) na página 22.

Requisitos do sistema e pré-requisitos para instalação autônoma (não Docker)

- ♦ [“Requisitos do sistema”](#) na página 16
- ♦ [“\(Opcional\) Pré-requisito para configuração do OSP”](#) na página 17

Requisitos do sistema

Esta seção explica os requisitos e pré-requisitos do sistema para instalar o Identity Console autônomo.

Categoria	Requisito mínimo
Processador	1,4 GHz, 64 bits
Memória	2 GB
Espaço em Disco	200 MB no Linux
Browser suportado	<ul style="list-style-type: none">♦ Versão mais recente do Microsoft Edge♦ Versão mais recente do Google Chrome♦ Versão mais recente do Mozilla Firefox <p>Observação: Ao usar o Identity Console no Mozilla Firefox, a operação pode falhar com a mensagem de erro Incompatibilidade de Origem. Para solucionar o problema, execute as seguintes etapas:</p> <ol style="list-style-type: none">1 Atualize o Firefox para a versão mais recente.2 Especifique <code>about:config</code> no campo URL do Firefox e pressione Enter.3 Pesquise Origin.4 Clique duas vezes em <code>network.http.SendOriginHeader</code> e mude o valor para 1.
Sistema Operacional Suportado	<ul style="list-style-type: none">♦ Certificado:<ul style="list-style-type: none">♦ SUSE Linux Enterprise Server (SLES) 15 SP1, SP2 e SP3♦ SUSE Linux Enterprise Server (SLES) 12 SP1, SP2, SP3, SP4 e SP5♦ Red Hat Enterprise Linux (RHEL) 7.8, 7.9, 8.0, 8.1, 8.2, 8.3, 8.4 e 8.5♦ OpenSUSE 15.1 e 15.2♦ Suportado: Suportado em versões posteriores de pacotes de suporte dos sistemas operacionais certificados acima.

Categoria	Requisito mínimo
Certificados	<ul style="list-style-type: none"> ♦ Você precisa obter um certificado de servidor pkcs12 com a chave privada para criptografar/descriptografar a troca de dados entre o cliente e o servidor do Identity Console. Esse certificado de servidor é usado para proteger a conexão http. Você pode usar certificados do servidor gerados por qualquer CA externa. Para obter mais informações, consulte Creating Server Certificate Objects (Criando objetos certificação do servidor). O certificado de servidor deve conter o Nome Alternativo do Assunto com endereço IP e DNS do servidor do Identity Console. Uma vez criado o objeto certificação do servidor, você precisa exportá-lo em formato .pfx. ♦ Você precisa obter um certificado de CA no formato .pem para validar a autenticação de CA dos certificados de servidor obtidos na etapa anterior. Este certificado rootCA também garante o estabelecimento de uma comunicação LDAP segura entre o cliente e o servidor do Identity Console. Por exemplo, você pode obter o certificado de CA do eDirectory (SSCert.pem) de /var/opt/novell/eDirectory/data/SSCert.pem.

Uma vez pronto, instale o Identity Console. Para obter mais informações, consulte [“Implantação do Identity Console independente”](#) na página 26.

(Opcional) Pré-requisito para configuração do OSP

Ao utilizar o OSP (One SSO Provider), você pode habilitar a autenticação de login único para seus usuários no portal do Identity Console. Você precisa instalar o OSP antes de instalar o Identity Console. Para configurar o OSP para o Identity Console, siga os prompts na tela e forneça os valores necessários para os parâmetros de configuração. Para obter mais informações, consulte [“Implantação do container do OSP”](#) na página 22. Para registrar o Identity Console em um servidor OSP existente, você deve adicionar manualmente o seguinte ao arquivo `ism-configuration.properties` na pasta `/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
```

Observação

- ♦ Se estiver instalando o OSP pela primeira vez, especifique a opção 'y' para **Configure OSP with eDir API** (Configurar o OSP com a API do eDir) e siga os prompts na tela para registrar o Identity Console com o OSP.
 - ♦ Especifique o nome da árvore do eDirectory em minúsculas ao configurar o Identity Console. Caso o nome da árvore não esteja especificado em minúsculas, o login no servidor Identity Console pode falhar.
 - ♦ Com o OSP, você pode se conectar a apenas uma árvore do eDirectory, pois o OSP não suporta várias árvores do eDirectory.
-

Requisitos do sistema e pré-requisitos para estação de trabalho

- ♦ [“Requisitos do sistema” na página 18](#)

Requisitos do sistema

Esta seção explica os requisitos e pré-requisitos do sistema para executar o Identity Console da estação de trabalho.

Categoria	Requisito mínimo
Processador	1.5 GHz, 64 bits
Memória	2 GB
Espaço em Disco	1 GB no Windows
Sistema Operacional Suportado	<ul style="list-style-type: none">♦ Certificado:<ul style="list-style-type: none">♦ Windows Server 2016♦ Windows Server 2019♦ Windows Server 2022♦ Windows 10♦ Windows 11

Categoria	Requisito mínimo
Certificados	<ul style="list-style-type: none"> ♦ Você precisa obter um certificado de servidor em formato pfx para trocar dados entre o cliente do Identity Console e o servidor REST. Este certificado de servidor precisa ser sempre nomeado keys.pfx. Para obter mais informações, consulte Creating Server Certificate Objects (https://www.netiq.com/documentation/edirectory-92/edir_admin/data/b1j4tpo3.html#b1j4u0cm) (Criando objetos certificação do servidor). ♦ Você precisa obter um certificado de CA no formato .pem para validar a autenticação de CA dos certificados de servidor obtidos na etapa anterior. Este certificado de CA raiz também garante o estabelecimento de uma comunicação LDAP segura entre o cliente e o servidor do Identity Console. Por exemplo, você pode obter o certificado de CA do eDirectory para Linux em SSCert.pem de /var/opt/novell/eDirectory/data/SSCert.pem. Obtenha o certificado de CA do eDirectory SSCert.pem para Windows do <local de instalação do eDirectory>\NetIQ\eDirectory\DIBFiles\CertServ\SSCert.pem.

Quando estiver pronto, prossiga com a implantação do Identity Console. Para obter mais informações, consulte [“Identity Console no Windows como estação de trabalho” na página 28.](#)

Verificação de autenticação RPM

Use as seguintes etapas para realizar a verificação de autenticação RPM:

- 1 Navegue até a pasta na qual o build é extraído.

Por exemplo: <local sem compactação tar do Identity Console>/IdentityConsole_150_Linux/license/MicroFocusGPGPackageSign.pub

- 2 Execute o seguinte comando para importar a Chave Pública:

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

- 3 (Opcional) Execute o seguinte comando para verificar a autenticação RPM: rpm --checksig -v <Nome do RPM>

Por exemplo:

```
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 Implantando o Identity Console

Este capítulo descreve o processo para implantar o Identity Console, além das recomendações de segurança. Para se preparar para a implantação, revise os pré-requisitos e requisitos do sistema fornecidos em [Capítulo 1, “Planejando a instalação do Identity Console”](#) na página 11.

- ♦ [“Recomendações de segurança”](#) na página 21
- ♦ [“Implantação do Identity Console como container do Docker”](#) na página 22
- ♦ [“Implantação do Identity Console independente”](#) na página 26
- ♦ [“Identity Console no Windows como estação de trabalho”](#) na página 28
- ♦ [“Parada e reinício do Identity Console”](#) na página 30
- ♦ [“Gerenciamento da Persistência de Dados”](#) na página 31
- ♦ [“Implantação do Identity Console nos Serviços de Kubernetes do Azure”](#) na página 31
- ♦ [“Modificando o certificado de servidor”](#) na página 38

Recomendações de segurança

- ♦ Os containers do Docker não têm nenhuma restrição de recurso por padrão. Isso fornece a todos os containers o acesso a todos os recursos de CPU e memória fornecidos pelo kernel do host. Você também deve definir limites para a quantidade de recursos que podem ser usados por um container para garantir que um container em execução não consuma mais recursos do que o limite, fazendo com que falte recursos para outros containers em execução.
 - ♦ O container do Docker deve garantir que um limite físico seja aplicado à memória usada pelo container usando o flag `--memory` no comando de execução do Docker.
 - ♦ O container do Docker deve garantir que um limite seja aplicado à quantidade de CPU usada por um container em execução usando o flag `--cpuset-cpus` no comando de execução do Docker.
- ♦ `--pids-limit` deve ser definido como 300 para restringir o número de threads do kernel gerados dentro do container a qualquer momento. Isso é para evitar ataques de DoS.
- ♦ Você deve definir a política de reinício do container em falha como 5 usando o flag `--restart` no comando de execução do Docker.
- ♦ Você deve usar o container do apenas quando o status de integridade aparecer como **Healthy** (Íntegro) depois que o container aparecer. Para verificar o status de saúde do container, execute o seguinte comando:

```
docker ps <container_name/ID>
```
- ♦ O container do Docker sempre começará como usuário não root (`nds`). Como uma medida de segurança adicional, habilite o remapeamento do namespace do usuário no daemon para evitar ataques de escalada de privilégios dentro do container. Para obter mais informações sobre o remapeamento de namespaces do usuário, consulte [Isolar containers com um namespace de usuário](#).

Implantação do Identity Console como container do Docker

Esta seção inclui os seguintes procedimentos:

- ♦ “Implantação do container do OSP” na página 22
- ♦ “Implantação do Identity Console como um container do Docker” na página 24
- ♦ “Multiárvore com o Identity Console como Docker” na página 26

Implantação do container do OSP

Execute as seguintes etapas para implantar o container do OSP:

- 1 Efetue login em [Software License and Download \(https://sld.microfocus.com/\)](https://sld.microfocus.com/) e navegue até a página Software Downloads (Downloads de software).
- 2 Selecione o seguinte:
 - ♦ Produto: eDirectory
 - ♦ Nome do produto: eDirectory per User Sub SW E-LTU
 - ♦ Versão: 9.2
- 3 Faça download do arquivo: IdentityConsole_<versão>_Containers_tar.zip.
- 4 Extraia o arquivo baixado para uma pasta.
- 5 Modifique o arquivo de propriedades silenciosas de acordo com seu requisito. Um arquivo de propriedades silenciosas de amostra é mostrado abaixo:

```
# 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

```

Observação: Para evitar restrições de espaço ao usar o arquivo de propriedades silenciosas (texto DOS), você deve converter o arquivo texto DOS para o formato UNIX usando a ferramenta dos2unix. Execute o comando abaixo para converter arquivos texto de finais de linha DOS para finais de linha Unix:

```

nome de arquivo dos2unix

Por exemplo,

arquivo de amostragem dos2unix

```

-
- 6 Gere um certificado de servidor (`cert.der`) usando o iManager e importe-o no keystore (`tomcat.ks`). Copie o arquivo de propriedades silenciosas e o keystore (`tomcat.ks`) para qualquer diretório. Por exemplo, `/data`. Siga estas etapas para criar um certificado de servidor e importá-lo no keystore:
 - 6a Execute o comando a seguir para criar um keystore (`tomcat.ks`). Gere a chave. O nome CN ou o nome de host totalmente qualificado da máquina precisa ser o endereço 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 Execute o comando a seguir para criar uma solicitação de autenticação de certificado. Por exemplo, `cert.csr`.

```

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

```
 - 6c Passe esse `cert.csr` para o iManager e obtenha o certificado de servidor `osp.der`. Verifique se você selecionou o tipo de chave como Personalizado, as opções de uso de chave como Criptografia de Dados, Criptografia de Chave e Assinatura Digital, e o campo Nome(s) Alternativo(s) do Assunto do certificado para conter endereço IP ou nome de host do servidor OSP. Para mais informações, veja [Criando um Objeto Certificação do Servidor](#).
 - 6d Execute os comandos a seguir para importar o certificado CA (`SSCert.der`) e o certificado de servidor (`cert.der`) no 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 Execute o seguinte comando para carregar a imagem do OSP:

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

8 Implante o container usando o seguinte comando:

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

Por exemplo,

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

Implantação do Identity Console como um container do Docker

Esta seção explica o procedimento para implantar o Identity Console como um container do Docker:

Observação: Os parâmetros de configuração, os valores amostrais e os exemplos mencionados neste procedimento são apenas para fins de referência. Não os use diretamente em seu ambiente de produção.

- 1 Efetue login em SLD: [Software License and Download \(https://sld.microfocus.com/\)](https://sld.microfocus.com/) e navegue até a página Software Downloads (Downloads de software).
- 2 Selecione o seguinte:
 - ◆ Produto: eDirectory
 - ◆ Nome do produto: eDirectory per User Sub SW E-LTU
 - ◆ Versão: 9.2

3 Faça download do arquivo: IdentityConsole_<version>_Container.tar.zip.

4 A imagem precisa ser carregada no registro local do Docker. Extraia e carregue o arquivo IdentityConsole_<versão>_Containers.tar.gz usando os comandos abaixo:

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

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

5 Crie o container do Docker do Identity Console usando o seguinte comando:

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

Por exemplo,

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

Observação

- ♦ Você pode aceitar o EULA definindo a variável de ambiente `ACCEPT_EULA` como `'Y'`. Você também pode aceitar o EULA do prompt da tela enquanto inicia o container usando a opção `-it` no comando de criação do Docker para o modo interativo.
- ♦ O parâmetro `--volume` no comando acima criará um volume para armazenamento de dados de configuração e de registro. Neste caso, criamos um volume de amostra chamado `IDConsole-volume`.

-
- 6 Copie o arquivo de certificado de servidor do sistema de arquivos local para o container como `/etc/opt/novell/eDirAPI/cert/keys.pfx` usando o comando a seguir. Para obter mais informações sobre a criação do certificado do servidor, consulte [“Pré-requisitos” na página 11](#):

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

Por exemplo,

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

Quando você se conecta a várias árvores do eDirectory, precisa obter pelo menos um certificado de servidor `keys.pfx` para todas as árvores conectadas.

- 7 Copie o arquivo de certificado CA (`.pem`) do sistema de arquivos local para o container como `/etc/opt/novell/eDirAPI/cert/sscert.pem` usando o comando a seguir. Para obter mais informações sobre a obtenção do certificado CA, consulte [“Pré-requisitos” na página 11](#):

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

Por exemplo,

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

Se o usuário precisar se conectar a várias árvores do eDirectory, consulte a seção: [“Multiárvore com o Identity Console como Docker” na página 26](#)

- 8 Modifique o arquivo de configuração de acordo com suas necessidades e copie o arquivo de configuração (`edirapi.conf`) do seu sistema de arquivos local para o container como `/etc/opt/novell/eDirAPI/conf/edirapi.conf` usando o seguinte comando:

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

Por exemplo,

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

- 9 Inicie o container do Docker usando o seguinte comando:

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

Por exemplo,

```
docker start identityconsole-container
```

Observação: É possível encontrar os seguintes arquivos de registro no diretório `/var/lib/docker/volumes/<nome_do_volume>/_data/eDirAPI/var/log`:

- ♦ `edirapi.log` — É usado para registrar diferentes eventos em questões de edirapi e de depuração.
 - ♦ `edirapi_audit.log` — É usado para registrar eventos de auditoria do edirapi. Os registros seguem o formato de auditoria CEF.
 - ♦ `container-startup.log` — É usado para capturar registros de instalação do container do Docker do Identity Console.
-

Multiárvore com o Identity Console como Docker

O Identity Console permite que o usuário se conecte a várias árvores obtendo o certificado de CA individual da árvore.

Por exemplo, se você se conectar a três árvores do eDirectory, precisará copiar todos os três certificados de CA no container do 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
```

Execute os seguintes comandos para reiniciar o Identity Console:

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

Implantação do Identity Console independente

- ♦ [“Implantação do Identity Console independente \(não Docker\)”](#) na página 26
- ♦ [“Multiárvore com Identity Console autônomo”](#) na página 28

Implantação do Identity Console independente (não Docker)

Esta seção explica o procedimento para implantar o Identity Console autônomo:

- 1 Efetue login em SLD: [Software License and Download \(https://sld.microfocus.com/\)](https://sld.microfocus.com/) e navegue até a página Software Downloads (Downloads de software).
- 2 Selecione o seguinte:
 - ♦ Produto: eDirectory
 - ♦ Nome do produto: eDirectory per User Sub SW E-LTU
 - ♦ Versão: 9.2

- 3 Faça download do build mais recente do Identity Console.
- 4 Extraia o arquivo baixado em uma pasta.
- 5 Abra um shell e navegue até a pasta na qual você extraiu o build do Identity Console.
- 6 Execute o seguinte comando enquanto estiver logado como usuário root ou equivalente:

```
./identityconsole_install
```
- 7 Leia a Introdução e clique em **ENTER**.
- 8 Clique em 'Y' para aceitar o Contrato de Licença. Isso instalará todos os RPMs necessários no seu sistema.
- 9 Digite o nome de host do servidor do Identity Console (FQDN)/endereço IP.
- 10 Digite o número da porta para o Identity Console escutar. O valor padrão é 9000.
- 11 Digite a opção de integrar OSP com o Identity Console ou de o Identity Console usar login LDAP.
- 12 Se você quiser integrar OSP com o Identity Console:

1. Digite o nome de domínio/endereço IP do servidor do eDirectory/cofre de identidade com o número da porta LDAPS.

Por exemplo,

192.168.1.1:636

2. Digite o nome de usuário do eDirectory/cofre de identidade.

Por exemplo,

cn=admin,ou=org_unit,o=org

3. Digite a senha do eDirectory/cofre de identidade.
4. Digite novamente a senha do eDirectory/cofre de identidade para confirmá-la.
5. Digite o nome de domínio/endereço IP do servidor do OSP com o número da porta SSL do servidor SSO.
6. Digite o ID do cliente do OSP.
7. Digite a senha do cliente do OSP.
8. Digite o nome da árvore do eDirectory/cofre de identidade.

- 13 Digite o caminho dos certificados raiz confiável (`SSCert.pem`), incluindo a pasta.

Por exemplo,

`/home/Identity_Console/certs`

Observação: O usuário não deve criar subdiretórios dentro da pasta de certificados.

- 14 Digite o caminho da certificação do servidor (`keys.pfx`), incluindo o nome do arquivo.

Por exemplo,

`/home/Identity_Console/keys.pfx`

- 15 Digite a senha do certificado de servidor. Para confirmar se inseriu a senha corretamente, redigite a senha do certificado de servidor. A instalação foi iniciada.

Observação: É possível encontrar os seguintes arquivos de registro no diretório `/var/opt/novell/eDirAPI/log`:

- ♦ `edirapi.log` — É usado para registrar diferentes eventos em questões de edirapi e de depuração.
- ♦ `edirapi_audit.log` — É usado para registrar eventos de auditoria do edirapi. Os registros seguem o formato de auditoria CEF.
- ♦ `identityconsole_install.log` — É usado para capturar registros de instalação do Identity Console.

Os registros para o início/interrupção do processo do Identity Console estão no arquivo `/var/log/messages`.

Observação: O NetIQ recomenda que, ao instalar o Identity Console e o eDirectory na mesma máquina, a máquina tenha pelo menos uma instância do eDirectory disponível.

Multiárvore com Identity Console autônomo

Ao se conectar a várias árvores do eDirectory, você precisa obter um certificado de CA individual da árvore.

Por exemplo, se você se conectar a três árvores do eDirectory, precisará copiar todos os três certificados de CA para o diretório `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
```

Execute um dos seguintes comandos para reiniciar o Identity Console:

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

ou

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

Identity Console no Windows como estação de trabalho

O Identity Console pode ser lançado no Windows como estação de trabalho e requer os serviços REST em execução. Portanto, quando é iniciado, um processo eDirAPI é executado no prompt `cmd` `edirapi.exe`. Se esse terminal `edirapi.exe` estiver fechado, o Identity Console se tornará não funcional.

O procedimento a seguir descreve como executar o Identity Console no Windows.

- 1 Efetue login em SLD [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) e navegue até a página Software Downloads (Downloads de software).
- 2 Selecione o seguinte:
 - ♦ Produto: eDirectory

- ♦ Nome do produto: eDirectory per User Sub SW E-LTU
 - ♦ Versão: 9.2
- 3 Faça download do arquivo
`IdentityConsole_<versão>_workstation_win_x86_64.zip`.
 - 4 Extraia o arquivo `IdentityConsole_<versão >_workstation_win_x86_64.zip` baixado para uma pasta.
 - 5 Navegue até a pasta extraída:
`IdentityConsole_150_workstation_win_x86_64\edirAPI\cert` e copie o certificado CA raiz confiável `SSCert.pem` e o certificado de servidor `keys.pfx`.

Para obter os certificados, consulte a seção: [“Requisitos do sistema e pré-requisitos para estação de trabalho” na página 18](#)

Se o usuário precisar se conectar a várias árvores do eDirectory, consulte a seção: [“Multiárvore com o Identity Console como estação de trabalho” na página 29](#)
-
- Observação:** O nome do certificado de servidor deve ser sempre como `keys.pfx`.
-
- 6 Navegue até a pasta na qual o build é extraído e clique duas vezes no arquivo `run.bat` (arquivo de lote do Windows).
 - 7 Digite a senha do certificado de servidor (`keys.pfx`) no prompt de comando.
O terminal de processos eDirAPI (`edirapi.exe`) começa a funcionar e a página de login do Identity Console é exibida.

Observação:

- ♦ Se o terminal de processos eDirAPI (`edirapi.exe`) já estiver em execução, execute `identityconsole.exe` da pasta em que o build foi extraído.
 - ♦ Os usuários podem encontrar os seguintes logs em:
`\IdentityConsole_150_workstation_win_x86_64\edirAPI\log`
`edirapi.log` — É usado para registrar diferentes eventos em problemas de depuração e `edirapi`.
`edirapi_audit.log` — É usado para registrar eventos de auditoria do `edirapi`. Os registros seguem o formato de auditoria CEF.
 - ♦ O login baseado em OSP não é suportado no modo de estação de trabalho.
 - ♦ A Identity Console Workstation escuta apenas na porta 9000. Não modifique o arquivo `edirapi_win.conf`.
-

Multiárvore com o Identity Console como estação de trabalho

O Identity Console permite que o usuário se conecte a várias árvores obtendo o certificado de CA individual da árvore.

- 1 Feche a Identity Console Workstation e o terminal eDirAPI.
- 2 Copie os certificados de CA `SSCert.pem` para o local:
`IdentityConsole_150_workstation_win_x86_64\edirAPI\cert`.

Por exemplo, se você quiser se conectar a três árvores do eDirectory, copie os certificados de CA como `SSCert1.pem`, `SSCert2.pem` e `SSCert3.pem`, respectivamente.

- 3 Navegue até a pasta na qual o build é extraído e clique duas vezes no arquivo `run.bat` (arquivo de lote do Windows).
- 4 Digite a senha `keys.pfx` no prompt do terminal e efetue login na árvore do eDirectory desejada.

Parada e reinício do Identity Console

- ♦ [“Parada e reinício do Identity Console como container do Docker”](#) na página 30
- ♦ [“Parada e reinício do Identity Console independente”](#) na página 30
- ♦ [“Fechar e reiniciar a Identity Console Workstation”](#) na página 31

Parada e reinício do Identity Console como container do Docker

Para parar o Identity Console, execute o seguinte comando:

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

Para reiniciar o Identity Console, execute o seguinte comando:

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

Para iniciar o Identity Console, execute o seguinte comando:

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

Parada e reinício do Identity Console independente

Para parar o Identity Console, execute um dos seguintes comandos:

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

ou

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

Para reiniciar o Identity Console, execute um dos seguintes comandos:

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

ou

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

Para iniciar o Identity Console, execute um dos seguintes comandos:

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

ou

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

Fechar e reiniciar a Identity Console Workstation

Para fechar o aplicativo e o processo, siga o procedimento:

- 1 Feche o aplicativo da área de serviço do Identity Console para Windows.
- 2 Pare o processo eDirAPI fechando o terminal de processos eDirAPI.

Para reiniciar a Identity Console Workstation, navegue até a pasta na qual o build é extraído e clique duas vezes no arquivo `run.bat` (arquivo de lote do Windows).

Observação: Se o terminal de processo eDirAPI já estiver em execução, execute `identityconsole.exe` da pasta em que o build foi extraído para reiniciar a Identity Console Workstation.

Gerenciamento da Persistência de Dados

Volumes de persistência de dados também são criados juntamente com os containers do Identity Console. Para usar os parâmetros de configuração de um container antigo usando os volumes, execute as seguintes etapas:

- 1 Pare o seu container do Docker atual usando o seguinte comando:

```
docker stop identityconsole-container
```

- 2 Crie o segundo container usando os dados de aplicativo do container antigo armazenado no volume do Docker (`edirapi-volume-1`):

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

- 3 Inicie o segundo container usando o seguinte comando:

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

- 4 (Opcional) Agora, o primeiro container pode ser removido usando o seguinte comando:

```
docker rm identityconsole-container
```

Implantação do Identity Console nos Serviços de Kubernetes do Azure

O AKS (Serviços de Kubernetes do Azure) é um serviço gerenciado de Kubernetes que permite implantar e gerenciar clusters. Esta seção inclui os seguintes procedimentos:

Implantação do Identity Console no cluster do AKS

Esta seção explica os seguintes procedimentos para implantar o Identity Console no cluster do AKS:

- ♦ [“Criando um ACR \(Registro de Contêiner do Azure\)” na página 32](#)
- ♦ [“Definindo um cluster do Kubernetes” na página 33](#)

- ♦ “Criando um endereço IP público de SKU padrão” na página 33
- ♦ “Configurando o Cloud Shell e conectando ao cluster do Kubernetes” na página 33
- ♦ “Implantando o aplicativo” na página 34

Criando um ACR (Registro de Contêiner do Azure)

O ACR (Registro de Contêiner do Azure) é um registro privado baseado no Azure, para imagens de containers do Docker.

Para obter mais detalhes, consulte a seção [Create an Azure container registry using the Azure portal](#) (Criar um Registro de Contêiner do Azure usando o portal do Azure) em [Create container registry - Portal](#) (Criar registro de container — Portal) ou execute as seguintes etapas para criar um ACR (Registro de Container do Azure):

1. Efetue login no [portal do Azure](#).
2. Acesse **Create a resource** > **Containers** > **Container Registry** (Criar um recurso > Contêineres > Registro de Contêiner).
3. Na guia **Basics** (Noções Básicas), especifique valores para **Resource group** (Grupo de recursos) e **Registry name** (Nome do registro). O nome do registro precisa ser único no Azure e conter no mínimo 5 e no máximo 50 caracteres alfanuméricos.

Aceite os valores padrão para as configurações restantes.

4. Clique em **Examinar + criar**.
5. Clique em **Criar**.
6. Efetue login na CLI do Azure e execute o comando a seguir para efetuar login no Registro de Contêiner do Azure

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

Por exemplo:

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

7. Recupere o servidor de login do Registro de Contêiner do Azure usando o comando:

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

Por exemplo:

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

8. Marque a imagem local do Identity Console com o nome do servidor de login do ACR (registryname.azurecr.io) usando o seguinte comando:

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

Por exemplo,

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

9. Envie por push a imagem marcada para o registro.

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

Por exemplo,

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

10. Recupere a lista de imagens no registro usando o comando:

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

Definindo um cluster do Kubernetes

Crie um recurso de serviço Kubernetes usando o portal do Azure ou a CLI.

Para obter mais detalhes para criar um recurso de serviço Kubernetes no Azure com um nó, confira [Create an AKS Cluster](#) (Criar um cluster do AKS) no [Azure Quickstart](#) (Início Rápido do Azure).

Observação:

- ♦ Verifique se você selecionou a CNI do Azure como a rede.
 - ♦ Selecione a rede virtual existente (em que o servidor eDirectory é implantado na sub-rede).
 - ♦ Selecione o registro de container existente no qual a imagem do Identity Console está disponível.
-

Criando um endereço IP público de SKU padrão


Um recurso de endereço IP público sob o grupo de recursos do cluster do Kubernetes atua como IP do balanceador de carga para o aplicativo.

Para obter etapas detalhadas, consulte [Create a public IP address using the Azure portal](#) (Criar um endereço IP público usando o portal do Azure) em [Create public IP address – Portal](#) (Criar endereço IP público — Portal).

Configurando o Cloud Shell e conectando ao cluster do Kubernetes

Use o Cloud Shell, disponível no portal do Azure, para todas as operações.

Para configurar o Cloud Shell no portal do Azure, consulte a seção [Start Cloud Shell](#) (Iniciar o Cloud Shell) em [Bash – Quickstart](#) (Bash — Início Rápido) ou execute as seguintes etapas para configurar o Cloud Shell e conecte-se ao cluster do Kubernetes:

1. No portal do Azure, clique no botão  para abrir o Cloud Shell.

Observação: Para gerenciar um cluster do Kubernetes, use o cliente de linha de comando do Kubernetes, `kubectl`. Se você usa o Azure Cloud Shell, o `kubectl` já está instalado.

2. Configure o `kubectl` para se conectar ao seu cluster do Kubernetes usando o seguinte comando:

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

Por exemplo,

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

3. Verifique a lista dos nós de cluster usando o comando:

```
kubectl get nodes
```

Implantando o aplicativo

Para implantar o Identity Console, você pode usar os arquivos de exemplo `idc-services.yaml`, `idc-statefulset.yaml`, `idc-storageclass.yaml` e `idc-pvc.yaml`.

Você também pode criar arquivos yaml próprios conforme suas necessidades.

1. Crie um recurso de classe de armazenamento usando o comando abaixo:

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

Por exemplo,

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

(Opcional) Para obter mais informações sobre como criar e usar dinamicamente o volume de persistência com o compartilhamento de arquivos do Azure, consulte [Dynamically create and use a persistent volume with Azure Files in Azure Kubernetes Service \(AKS\)](#) (Criar e usar dinamicamente um volume persistente com Arquivos do Azure no AKS [Serviço de Kubernetes do Azure]).

Um arquivo de recurso de classe de armazenamento de exemplo foi mostrado abaixo:

```
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
~
```

Um recurso de classe de armazenamento permite o provisionamento dinâmico de armazenamento. Ele é usado para definir como um compartilhamento de arquivos do Azure é criado.

2. Veja os detalhes da classe de armazenamento usando o comando abaixo:

```
kubectl get sc
```

3. Crie um recurso de pvc usando o arquivo `idc-pvc.yaml`:


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

Por exemplo,

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

Um arquivo de recurso de pvc de exemplo foi mostrado abaixo:

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

Um recurso de reivindicação de volume persistente cria o compartilhamento de arquivos. Uma reivindicação de volume persistente (persistent volume claim — PVC) usa o objeto da classe de armazenamento para provisionar dinamicamente um compartilhamento de arquivos do Azure.

4. Faça upload do certificado `edirapi.conf`, do certificado CA e do certificado de servidor para o Cloud Shell.

Clique no ícone de botão **Upload/Download files** (Carregar/Baixar arquivos)  no Cloud Shell e faça upload dos arquivos `edirapi.conf`, `SSCert.pem` e `keys.pfx`.

Observação: `edirapi.conf` tem um parâmetro “origem”. Aqui precisamos fornecer endereço IP com o qual acessaremos o aplicativo Identity Console. (use o endereço IP que é criado na seção [“Criando um endereço IP público de SKU padrão” na página 33.](#))

A implantação do Identity Console requer um certificado de servidor (`keys.pfx`).

Ao criar o certificado de servidor, certifique-se de fornecer nome DNS válido em Nome(s) Alternativo(s) do Assunto.

Etapas para criar um nome DNS válido:

Um pod típico implantado usando StatefulSet tem um nome DNS como o abaixo — `{nomedestatefulset}-{ordinal}.{nomedoserviço}.{namespace}.svc.cluster.local`

- ♦ Se o nome de StatefulSet no arquivo `idconsole-statefulset.yaml` for `idconsole-app`, então `statefulsetname = idconsole-app`
- ♦ Se o pod for o primeiro, então `ordinal = 0`
- ♦ Se você definir `serviceName` no arquivo `idconsole-statefulset.yaml` como `idconsole`, então `serviceName = idconsole`
- ♦ Se o namespace for o padrão, então `namespace=default`

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

5. Crie um recurso de configmap no cluster do Kubernetes que armazena os arquivos de configuração junto com os certificados.

Antes de executar o comando, certifique-se de que os arquivos (`edirapi.conf`, `SSCert.pem` e `keys.pfx`) estejam presentes no diretório.

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

Por exemplo,

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

6. Veja os detalhes do objeto configmap usando o comando kubectl describe:

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

Por exemplo,

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

7. Crie um recurso StatefulSet para implantar um container.

Execute o comando abaixo para implantar o container:

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

Por exemplo,

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

Um arquivo de recurso StatefulSet de exemplo foi mostrado abaixo:

```
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. Execute o seguinte comando para verificar o status do pod implantado:

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

9. Crie um recurso de serviço do tipo loadBalancer.

O tipo do serviço especificado no arquivo yaml é loadBalancer.

Crie um recurso de serviço usando o comando abaixo:

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

Por exemplo,

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

Um arquivo de recurso de serviço de exemplo foi mostrado abaixo:

```

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

```

Verifique o endereço EXTERNAL-IP (ou o loadBalancerIP) usando o comando abaixo:

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

10. Inicie o URL usando EXTERNAL-IP (ou o endereço loadBalancerIP).

Por exemplo,

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

Modificando o certificado de servidor

Esta seção fornece informações sobre a modificação do certificado de servidor no Container do Docker e no Identity Console Autônomo.

- ♦ [“Modificando o certificado de servidor no container do Docker” na página 38](#)
- ♦ [“Modificando o certificado de servidor no Identity Console autônomo” na página 38](#)

Modificando o certificado de servidor no container do Docker

Execute as seguintes etapas para modificar o certificado de servidor no container do Docker:

- 1 Execute o seguinte comando para copiar o novo certificado de servidor em qualquer local do seu container.

Por exemplo,

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

- 2 Efetue login no container usando o seguinte comando:

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

- 3 Execute o NLPCERT para armazenar as chaves como um pseudousuário:

```
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 Saia do console do container usando o comando:

```
exit
```

- 5 Reinicie o container digitando:

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

Modificando o certificado de servidor no Identity Console autônomo

Execute as seguintes etapas para modificar o certificado de servidor no container autônomo:

- 1 Execute NLPCERT para armazenar as chaves:

```
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 o Identity Console:

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

3 Atualização do Identity Console

Este capítulo descreve o processo de atualização do Identity Console para as suas versões mais recentes. Para preparar-se para o upgrade, revise os pré-requisitos e requisitos do sistema fornecidos em [Capítulo 1, “Planejando a instalação do Identity Console”](#) na página 11.

Esta seção inclui os seguintes procedimentos:

- ♦ “Upgrade do Identity Console como container do Docker” na página 39
- ♦ “Atualizando o Identity Console autônomo (não Docker)” na página 41
- ♦ “Upgrade do container do OSP” na página 42

Upgrade do Identity Console como container do Docker

Quando uma nova versão da imagem do Identity Console estiver disponível, o administrador poderá executar um procedimento de upgrade para implantar o container com a versão mais recente do Identity Console. Verifique se armazenou nos volumes do Docker, de modo persistente, todos os dados necessários relacionados ao aplicativo antes de fazer um upgrade. Realize as seguintes etapas para fazer upgrade do Identity Console usando o container do Docker:

- 1 Faça download da versão mais recente da imagem do Docker e carregue-a de [Software License and Download \(https://sld.microfocus.com/\)](https://sld.microfocus.com/). Depois, execute as etapas para instalar a versão mais recente do Identity Console, conforme mencionado em [“Implantando o Identity Console”](#) na página 21.

- 2 Após a última imagem do Docker ser carregada, pare o seu container do Docker atual usando o seguinte comando:

```
docker stop identityconsole-container
```

- 3 (Opcional) Use o backup do volume compartilhado.

- 4 Apague o container existente do Identity Console executando o seguinte comando:

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

Por exemplo,

```
docker rm identityconsole-container
```

- 5 (Opcional) Apague a imagem obsoleta do Docker do Identity Console executando o seguinte comando:

```
docker rmi identityconsole
```

- 6 Crie o container do Docker do Identity Console usando o seguinte comando:

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

Por exemplo:

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

Observação

- ♦ Você pode aceitar o EULA definindo a variável de ambiente `ACCEPT_EULA` como `'Y'`. Você também pode aceitar o EULA do prompt da tela enquanto inicia o container usando a opção `-it` no comando de criação do Docker para o modo interativo.
- ♦ O parâmetro `--volume` no comando acima criará um volume para armazenamento de dados de configuração e de registro. Neste caso, criamos um volume de amostra chamado `IDConsole-volume`.

-
- 7 Copie o arquivo de certificado do servidor do sistema de arquivos local para o container recém-criado como `/etc/opt/novell/eDirAPI/cert/keys.pfx` usando o seguinte comando:

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

Por exemplo,

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

Quando conectado a várias árvores do eDirectory, verifique se você copiou pelo menos um certificado de servidor `keys.pfx` para todas as árvores conectadas.

- 8 Copie o arquivo de certificado CA (`.pem`) do sistema de arquivos local para o container recém-criado como `/etc/opt/novell/eDirAPI/cert/sscert.pem` usando o seguinte comando:

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

Por exemplo,

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

Ao se conectar a várias árvores do eDirectory, verifique se você obteve um certificado de CA individual para todas as árvores conectadas. Por exemplo, se você se conectar a três árvores do eDirectory, precisará copiar todos os três certificados de CA no container do 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
```

Observação: A partir do Identity Console 1.4, o arquivo de configuração (`edirapi.conf`) não inclui explicitamente os parâmetros `"ldapuser"`, `"ldappassword"` e `"ldapsver"`. O valor do parâmetro `"bcert"` precisa incluir o caminho de diretório para certificados raiz confiável. Por exemplo, `bcert = "/etc/opt/novell/eDirAPI/cert/"`. E o parâmetro `"origin"` é independente do parâmetro `"check-origin"` e é obrigatório quando a configuração de DNS é usada.

- 9 Copie o arquivo de configuração (`edirapi.conf`) do sistema de arquivos local para o container recém-criado como `/etc/opt/novell/eDirAPI/conf/edirapi.conf` usando o seguinte comando:

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

Por exemplo,

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

- 10 Inicie o segundo container usando o seguinte comando:

```
docker start identityconsole-container
```

- 11 Para verificar o status do container em execução, execute o seguinte comando:

```
docker ps -a
```

Atualizando o Identity Console autônomo (não Docker)

Esta seção explica o procedimento para fazer upgrade do Identity Console autônomo:

- 1 Faça download de `IdentityConsole_<versão>_Containers.tar.gz` de [Software License and Download \(https://sld.microfocus.com/\)](https://sld.microfocus.com/).
- 2 Efetue login no SLD, navegue até a página de download de software do SLD e clique em **Fazer Download**.
- 3 Navegue selecionando Product: **eDirectory** > Product Name **eDirectory per User Sub SW E-LTU** > Version: **9.2** (Produto: eDirectory > Nome do Produto: eDirectory per User Sub SW E-LTU > Versão: 9.2)

- 4 Faça download do build mais recente do Identity Console.
- 5 Extraia o arquivo baixado usando o seguinte comando:

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

- 6 Navegue até a pasta na qual você extraiu o build do Identity Console.
- 7 Copie todos os certificados raiz confiável das árvores do eDirectory que você deseja conectar em uma pasta. Para copiar o certificado raiz confiável na pasta, execute o seguinte comando:

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

Por exemplo,

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

- 8 Execute o seguinte comando:

```
./identityconsole_install
```

- 9 Especifique o caminho da pasta de certificados raiz confiável usado na **etapa 4**.
- 10 O Identity Console recebe o upgrade com sucesso.

Upgrade do container do OSP

Execute as seguintes etapas para fazer upgrade do container do OSP:

- 1 Faça download da versão mais recente da imagem do OSP e carregue-a de [Software License and Download \(https://sld.microfocus.com/\)](https://sld.microfocus.com/).

Por exemplo,

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

- 2 Após a última imagem do Docker ser carregada, pare o seu container do OSP atual usando o seguinte comando:

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

- 3 (Opcional) Use o backup do volume compartilhado.

- 4 Apague o container do OSP existente executando o seguinte comando:

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

Por exemplo,

```
docker rm OSP_Container
```

- 5 Vá para o diretório que contém o keystore (`tomcat.ks`) e o arquivo de propriedades silenciosas, apague o keystore existente (`tomcat.ks`) e mantenha a pasta do OSP existente. Gere uma nova keystore (`tomcat.ks`) com o tamanho de chave como 2048. Para obter mais informações, consulte a **etapa 4** na seção [Deploying the OSP Container](#) (Implantando o container do OSP) do [Identity Console Installation Guide](#) (Guia de Instalação do Identity Console).

- 6 Implante o container usando o seguinte comando:

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

Por exemplo,

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


4 Desinstalação do Identity Console

Este capítulo descreve o processo de desinstalação do Identity Console:

- ♦ “Procedimento de desinstalação para ambiente Docker” na página 43
- ♦ “Procedimento de desinstalação do Identity Console independente (não Docker)” na página 43

Procedimento de desinstalação para ambiente Docker

Para desinstalar o container do Docker do Identity Console, execute as seguintes etapas:

- 1 Pare o container do Identity Console:

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

- 2 Execute o seguinte comando para remover o container do Docker do Identity Console:

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

- 3 Execute o seguinte comando para remover a imagem do Docker:

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

- 4 Remova o volume do Docker:

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

Observação: Se você remover o volume, os dados também serão removidos do servidor.

Procedimento de desinstalação do Identity Console independente (não Docker)

Para desinstalar o Identity Console independente, execute as seguintes etapas:

- 1 Navegue até o diretório `/usr/bin` na máquina na qual o Identity Console está instalado.

- 2 Execute o seguinte comando:

```
./identityconsoleUninstall
```

- 3 O Identity Console é desinstalado com êxito.

Observação: Quando o eDirectory ou outro produto NetIQ estiver instalado na máquina, o usuário precisará desinstalar manualmente o *nici* e o *openssl*.
