Sistema: Certificado HTTPS no Wildfly
Área: Infraestrutura, tecnologia e desenvolvimento

Sumário

1) Introdução

Este manual tem como objetivo orientar tecnicamente a equipe de implantação na configuração de um certificado digital HTTPS no servidor WildFly, utilizando como ferramenta auxiliar o KeyStore Explorer para o gerenciamento do keystore.

A aplicação de um certificado SSL/TLS é essencial para garantir a comunicação segura entre cliente e servidor, protegendo os dados trafegados contra interceptações e adulterações.

O procedimento aqui descrito abrange:

- A criação e manipulação de um arquivo keystore com extensão .keystore, no formato JKS, contendo o certificado digital da empresa;
- O uso da ferramenta KeyStore Explorer para importar o certificado no padrão PKCS#12, aplicar senhas e definir alias;
- A parametrização do keystore dentro do WildFly, por meio da interface de administração via browser;
- A associação do certificado ao conector HTTPS utilizando os recursos de Security - Elytron, Key Manager, SSL Context, e Undertow (Listener);
- A reinicialização segura do domínio no ambiente modo domain, com posterior validação do acesso HTTPS em ambiente real.

Este manual segue as boas práticas de configuração indicadas na documentação oficial da Senior Sistemas, com foco em ambientes controlados, onde há exigência de identificação segura da aplicação via certificados digitais emitidos por autoridade certificadora confiável.

2) Pré-requisitos

- Acesso administrativo ao WildFly;
- Certificado digital no formato .pfx ou .p12 válido;
- Instalação do KeyStore Explorer;
- Conhecimento básico de administração de servidores de aplicação Java EE.

3) Documentações utilizadas Senior

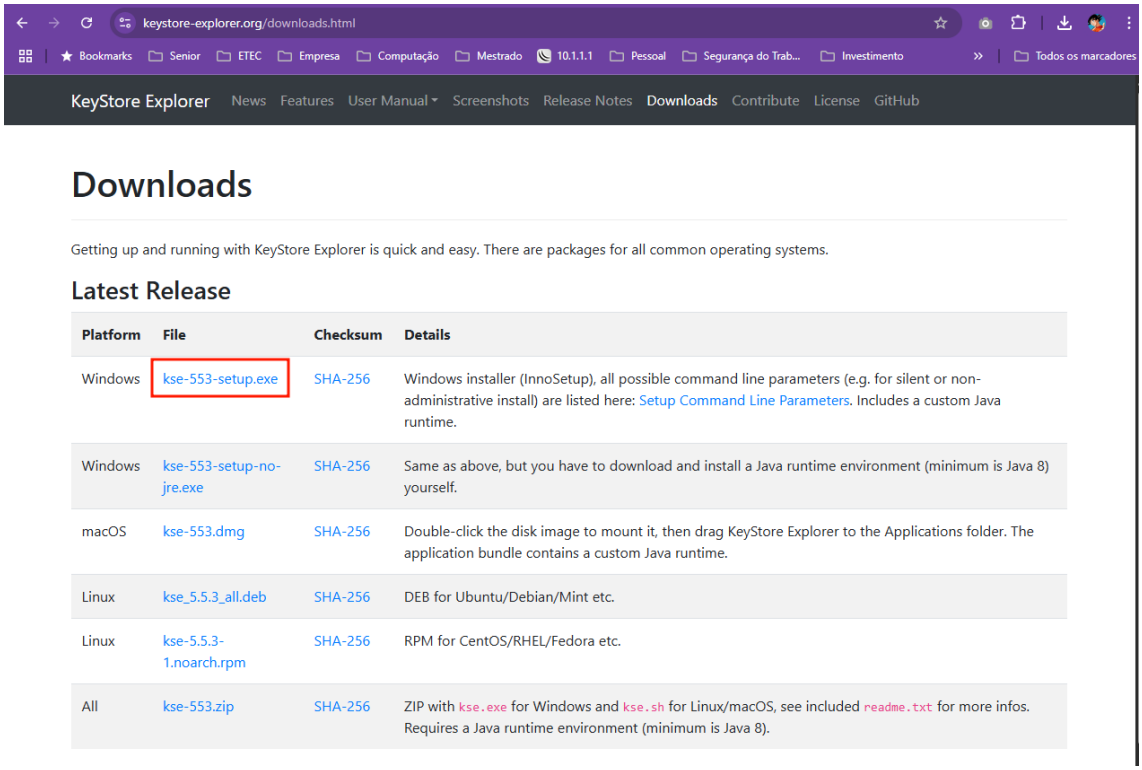
https://documentacao.senior.com.br/tecnologia/5.10.4/informacoes-tecnicas/java/wildlfy/wildfly.htm
https://documentacao.senior.com.br/tecnologia/5.10.4/informacoes-tecnicas/java/wildlfy/certificado-digital-para-wildfly.htm

4) KeyStore Explorer

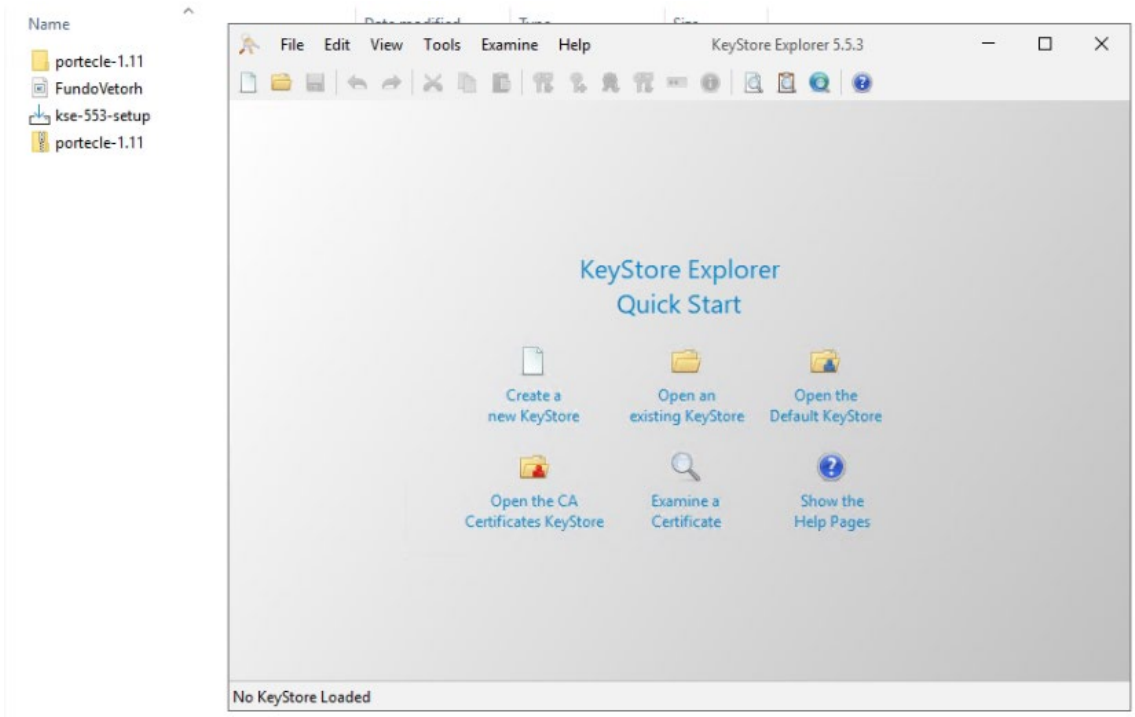
Download do KeyStore Explorer
Acesse o site oficial do KeyStore Explorer através do link:
https://keystore-explorer.org/downloads.html

Na seção Latest Release, localize a opção compatível com o sistema operacional Windows e clique no link kse-553-setup.exe (conforme destacado na imagem). Essa versão já inclui o ambiente Java necessário para execução da ferramenta, facilitando a instalação.

Após o download, prossiga com a instalação do programa normalmente.



Abertura do KeyStore Explorer
Após a instalação, abra o KeyStore Explorer. A tela inicial exibirá o menu Quick Start, com diversas opções para gerenciamento de arquivos keystore.

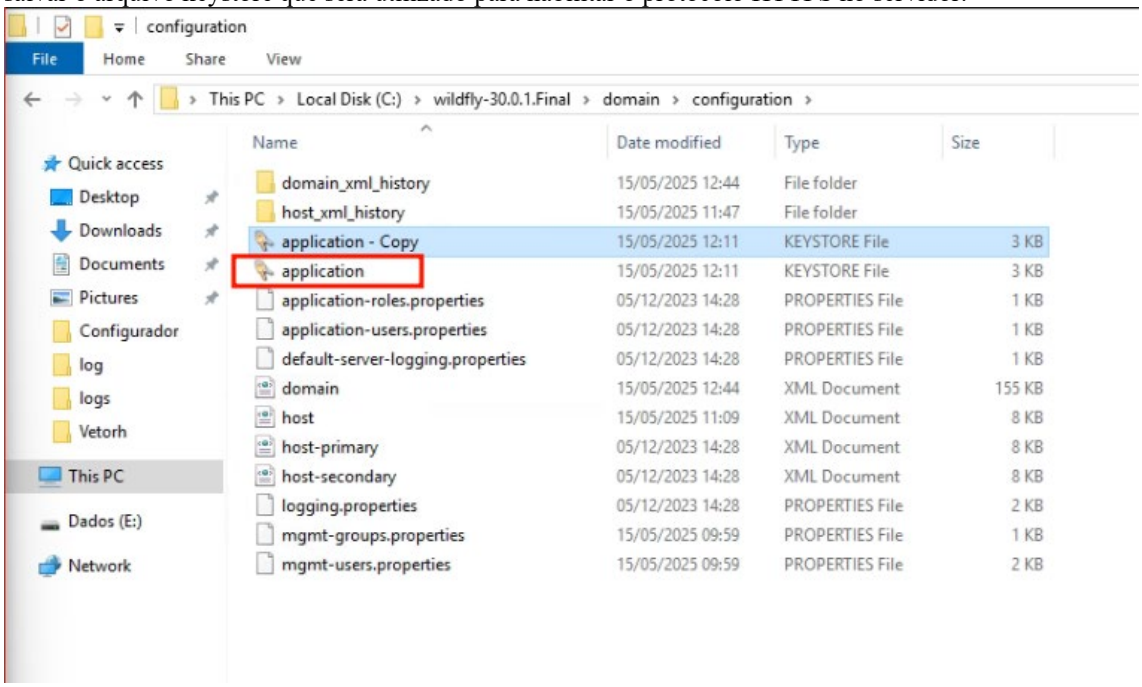


Neste momento, você poderá optar por:

- Create a new KeyStore: para gerar um novo keystore do zero;
- Open an existing KeyStore: para abrir um arquivo de keystore já existente;
- Open the Default KeyStore: abre o keystore padrão, se houver um configurado;

- Open the CA Certificates KeyStore: para abrir o repositório de autoridades certificadoras;
- Examine a Certificate: para inspecionar diretamente um certificado.

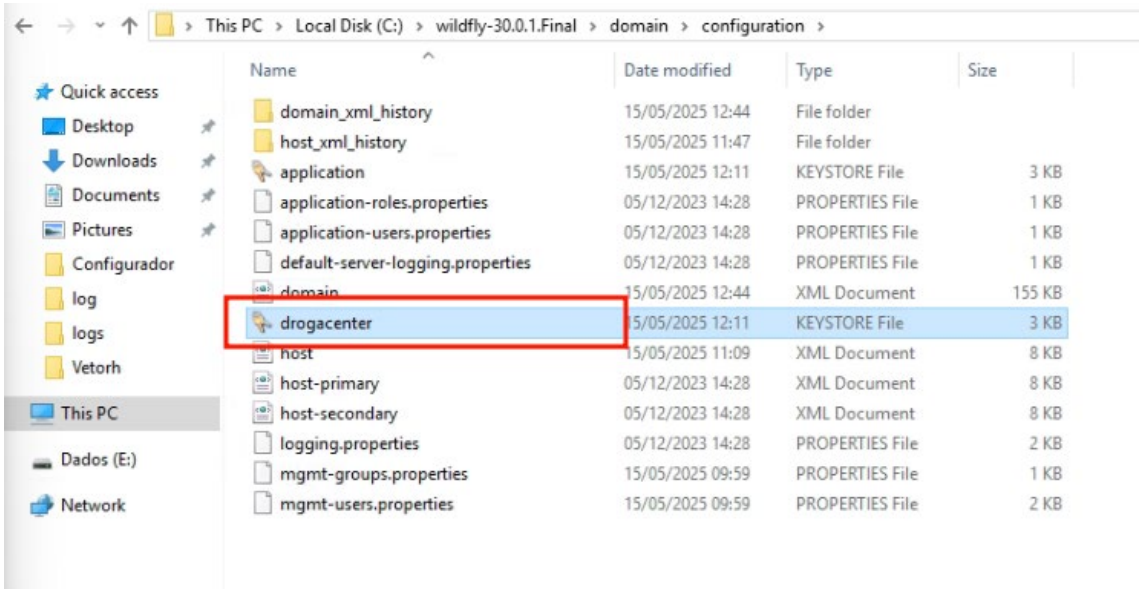
Dentro do diretório de configuração do WildFly, localizado em C:\wildfly-30.0.1.Final\domain\configuration, deve-se salvar o arquivo keystore que será utilizado para habilitar o protocolo HTTPS no servidor.



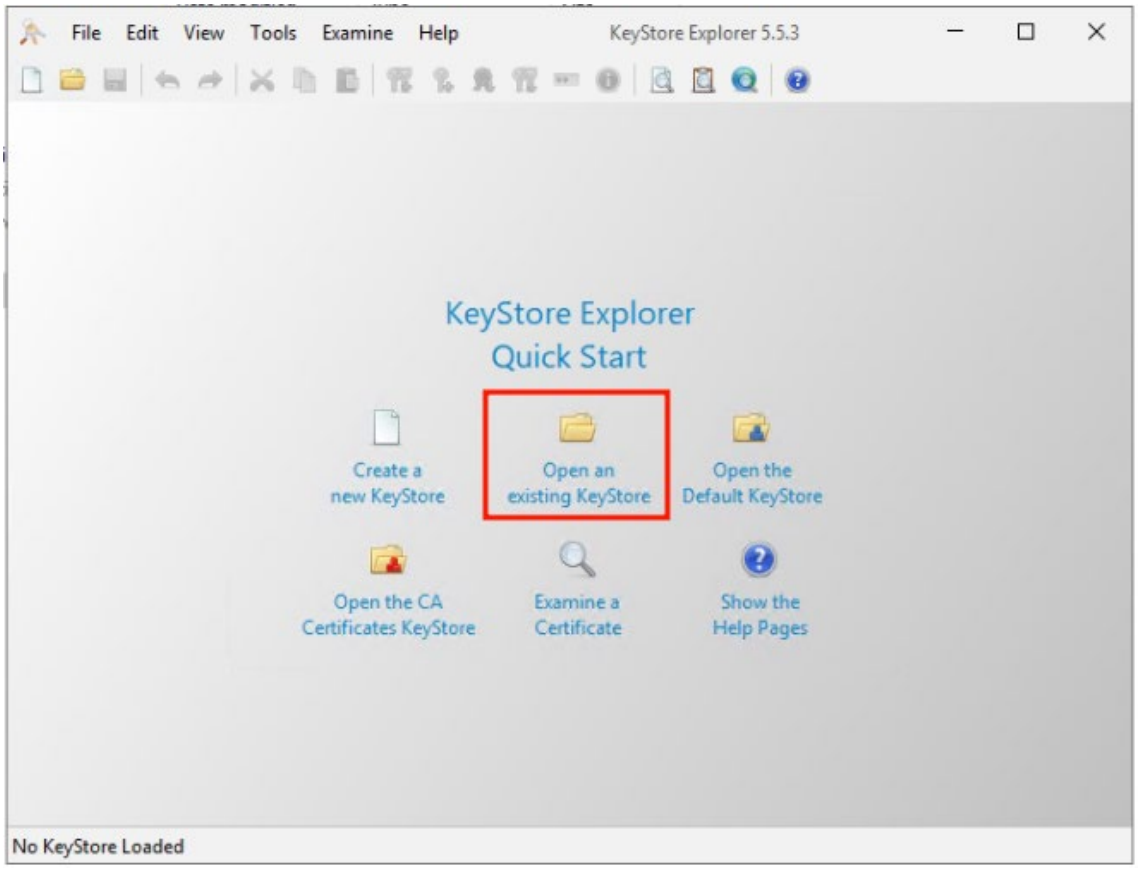
No exemplo apresentado, o arquivo gerado pelo KeyStore Explorer está salvo com o nome application, com a extensão .keystore. Uma cópia de segurança chamada application - Copy também foi criada como medida preventiva.

Este arquivo será referenciado posteriormente na configuração do conector HTTPS no arquivo host.xml ou domain.xml, conforme o modo de execução do WildFly. É essencial que o keystore esteja no diretório correto e com permissões adequadas para que o processo de inicialização do servidor consiga acessá-lo sem erros.

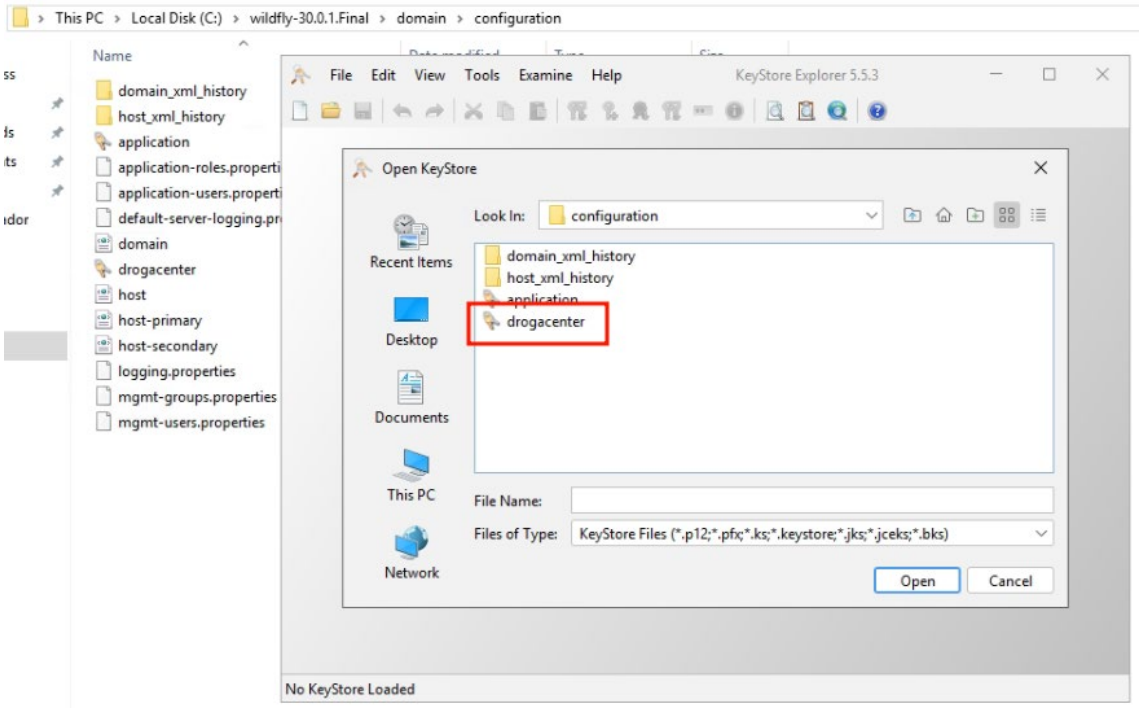
Também é possível nomear o arquivo .keystore com o nome do cliente, facilitando a identificação em ambientes com múltiplas instâncias ou configurações. No exemplo apresentado, o keystore foi salvo com o nome [nome_cliente].keystore, representando o nome do cliente correspondente.
Essa abordagem ajuda na organização dos arquivos, especialmente quando há necessidade de manter mais de um certificado no mesmo diretório. Lembre-se de que o nome utilizado aqui deverá ser referenciado corretamente no arquivo de configuração do servidor (host.xml ou domain.xml) durante a parametrização do conector HTTPS.



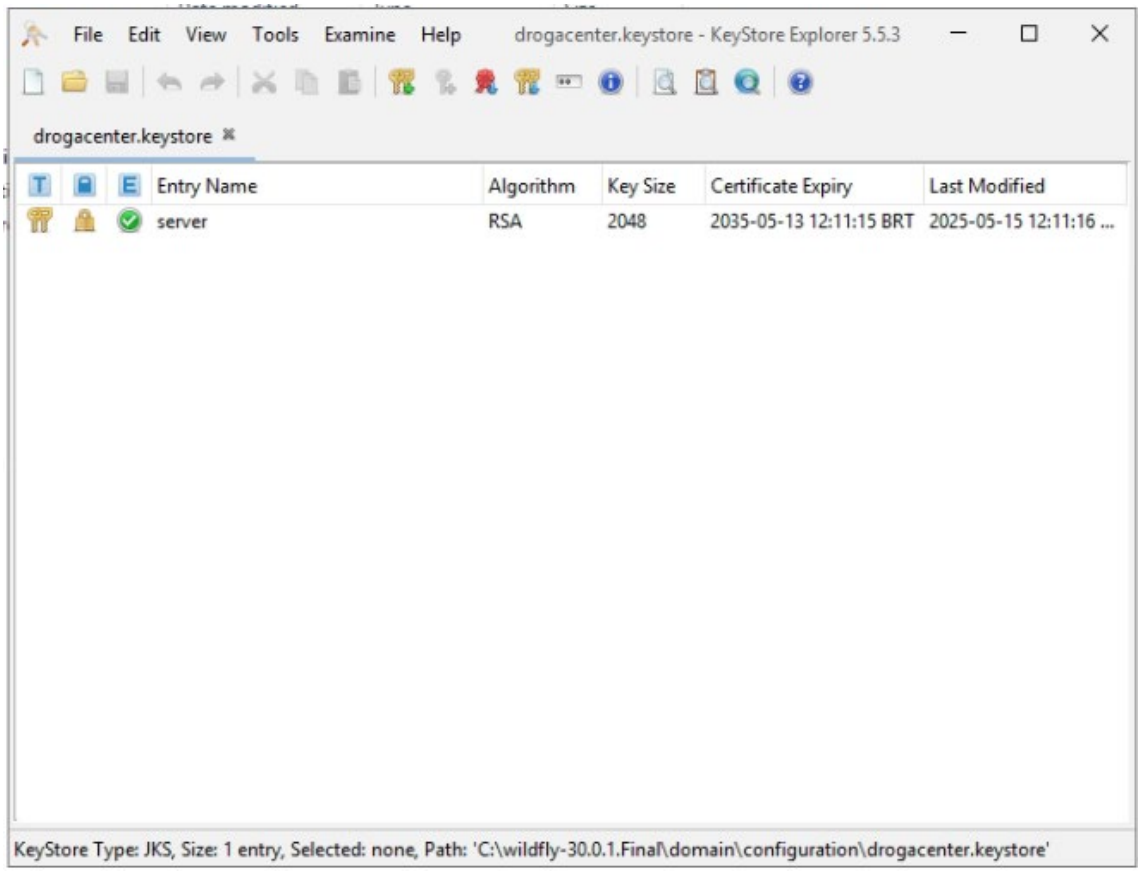
Com a opção "Open an existing KeyStore" selecionada, será exibida a janela de seleção de arquivos. Navegue até o diretório onde o keystore foi salvo — no caso deste exemplo, o diretório é C:\wildfly-30.0.1.Final\domain\configuration.



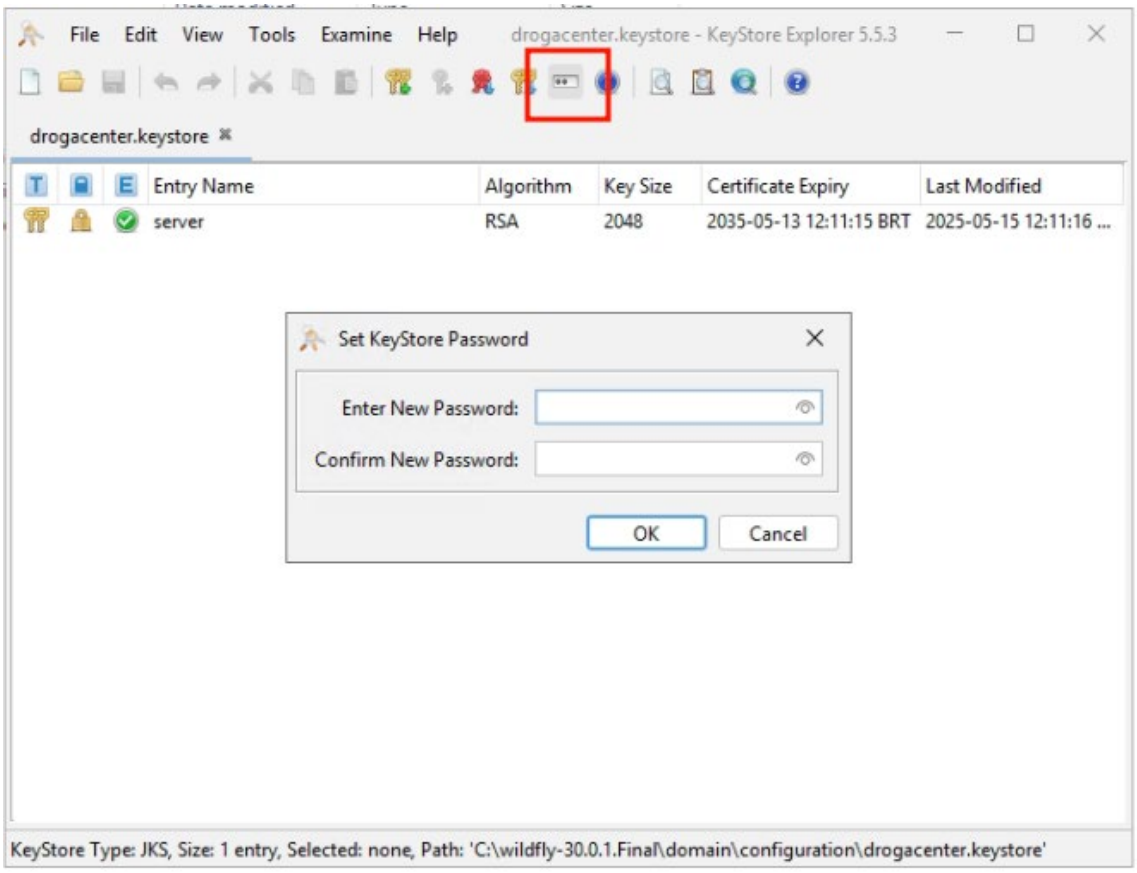
Selecione o arquivo desejado, que pode estar nomeado com o nome do cliente. No print apresentado, foi utilizado o nome [nome_cliente].



Após selecionar o arquivo, clique em “Open”. Em seguida, será solicitada a senha de acesso ao keystore. Nenhuma senha foi definida durante a criação, basta deixar o campo em branco e confirmar com OK para prosseguir.



Com o keystore aberto no KeyStore Explorer, é possível realizar ajustes, como a definição de uma senha de proteção do arquivo. Esse passo é importante, pois a senha será exigida posteriormente na configuração do WildFly para acesso ao keystore durante a inicialização do conector HTTPS.

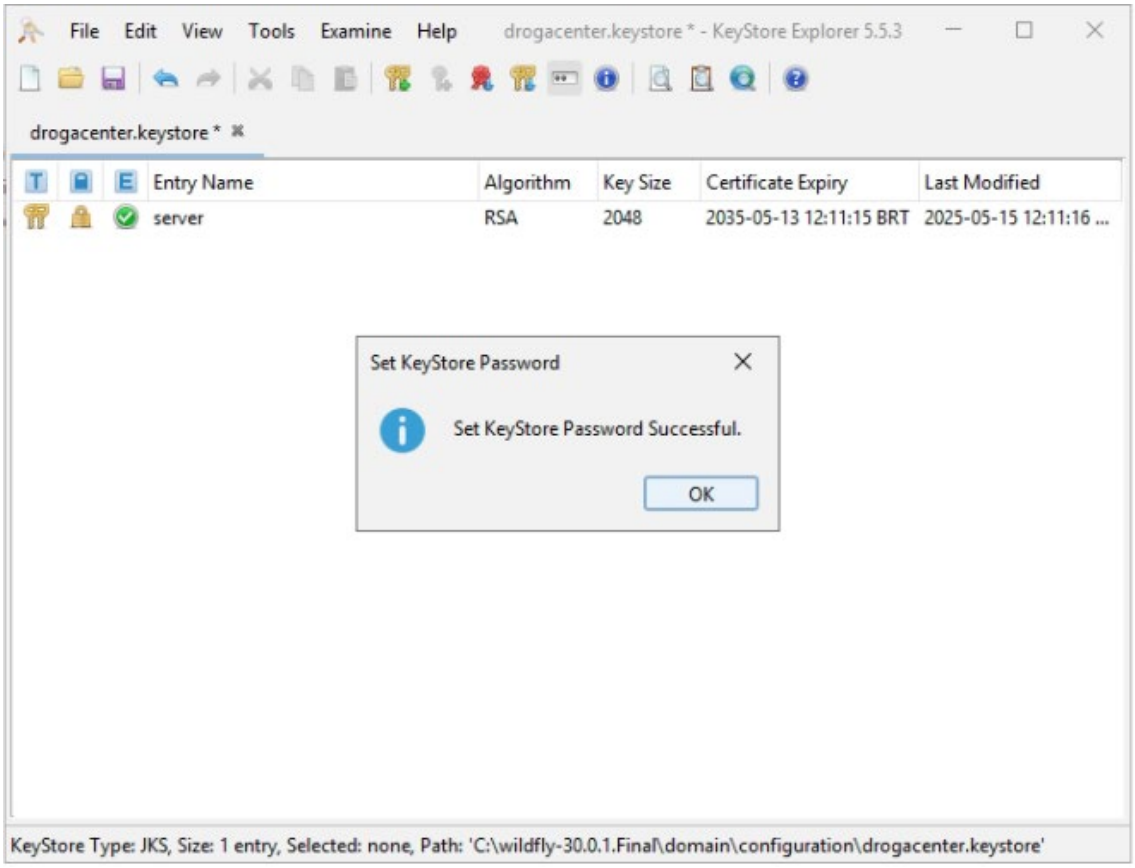


Clique no ícone com reticências (…), localizado na barra superior (conforme destacado na imagem), para definir a senha do keystore.

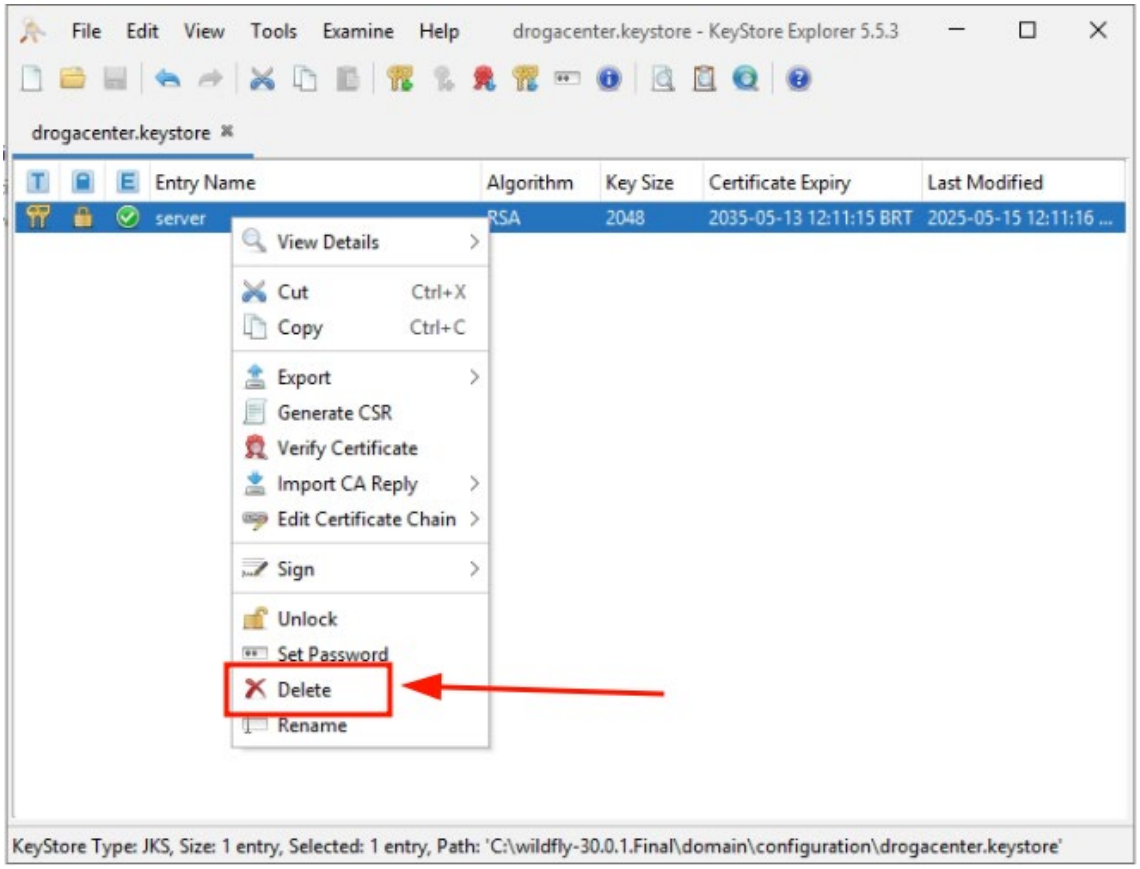
Na janela “Set KeyStore Password”, digite a nova senha nos dois campos ("Enter New Password" e "Confirm New Password") e clique em OK para confirmar.

Sugestão: utilize uma senha segura e padronizada, como changeit, que também é amplamente utilizada como padrão em aplicações Java. Apenas certifique-se de documentar essa senha para uso futuro.

Assim que a senha for definida, salve o arquivo novamente para que a alteração seja aplicada.



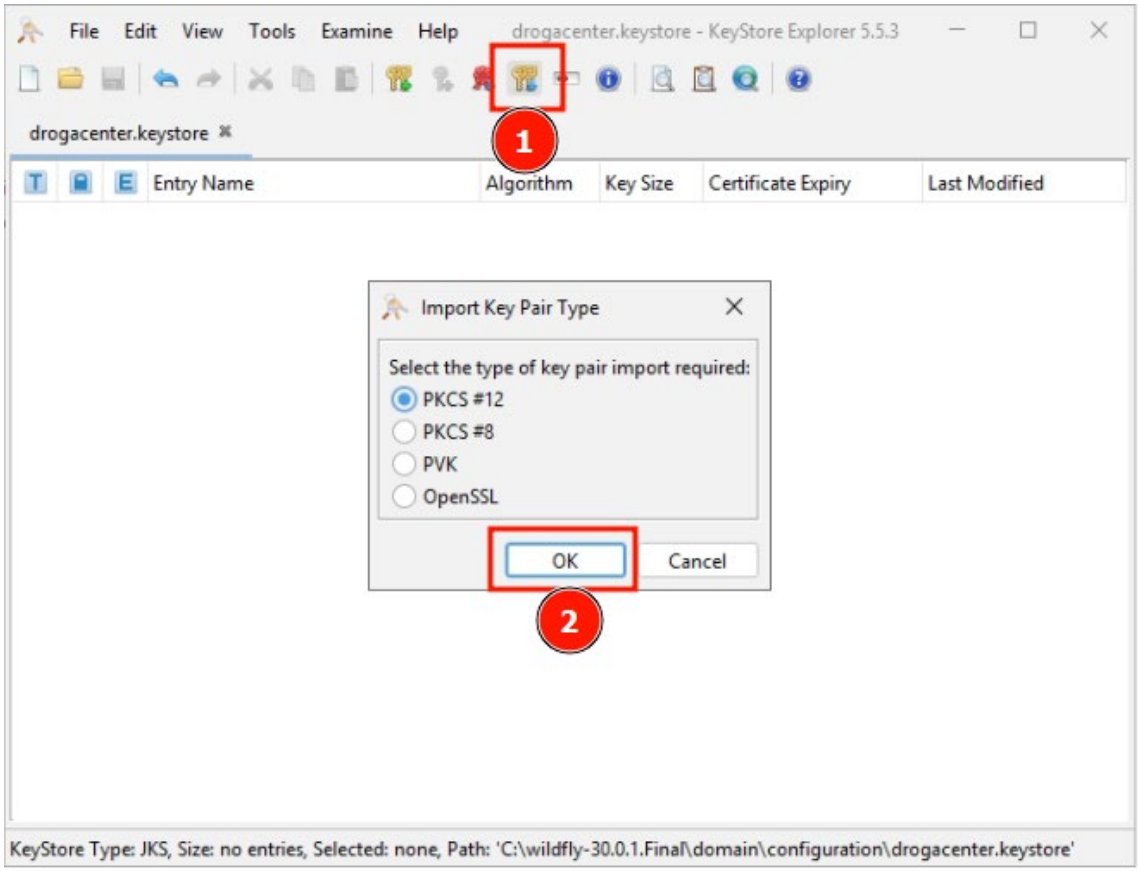
Ao abrir o keystore, é comum que ele contenha uma entrada padrão gerada automaticamente durante a criação do arquivo, geralmente com o nome server. Esse certificado é apenas ilustrativo e não possui validade real para uso em produção. Portanto, ele pode ser excluído sem prejuízo. Para isso, clique com o botão direito sobre a entrada server e selecione a opção Delete, conforme destacado na imagem.



Após a exclusão, lembre-se de salvar o keystore para que a alteração seja efetivada.

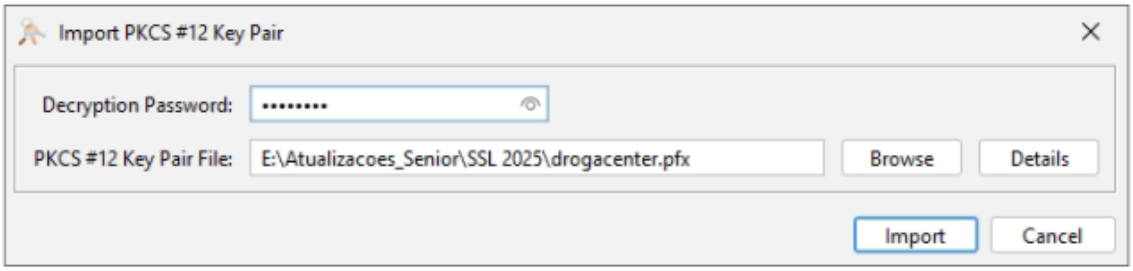
Com o keystore limpo (sem certificados padrão), o próximo passo é importar o certificado válido da empresa — normalmente fornecido no formato .pfx (PKCS #12), que contém a chave privada e o certificado público emitido por uma autoridade certificadora (AC).

Para realizar a importação:



Clique no ícone de importação de par de chaves (ícone com duas chaves amarelas), localizado na barra de ferramentas superior — conforme indicado no item 1 da imagem;
Na janela Import Key Pair Type, selecione a opção PKCS #12 (formato padrão de certificados corporativos);
Clique em OK para avançar.

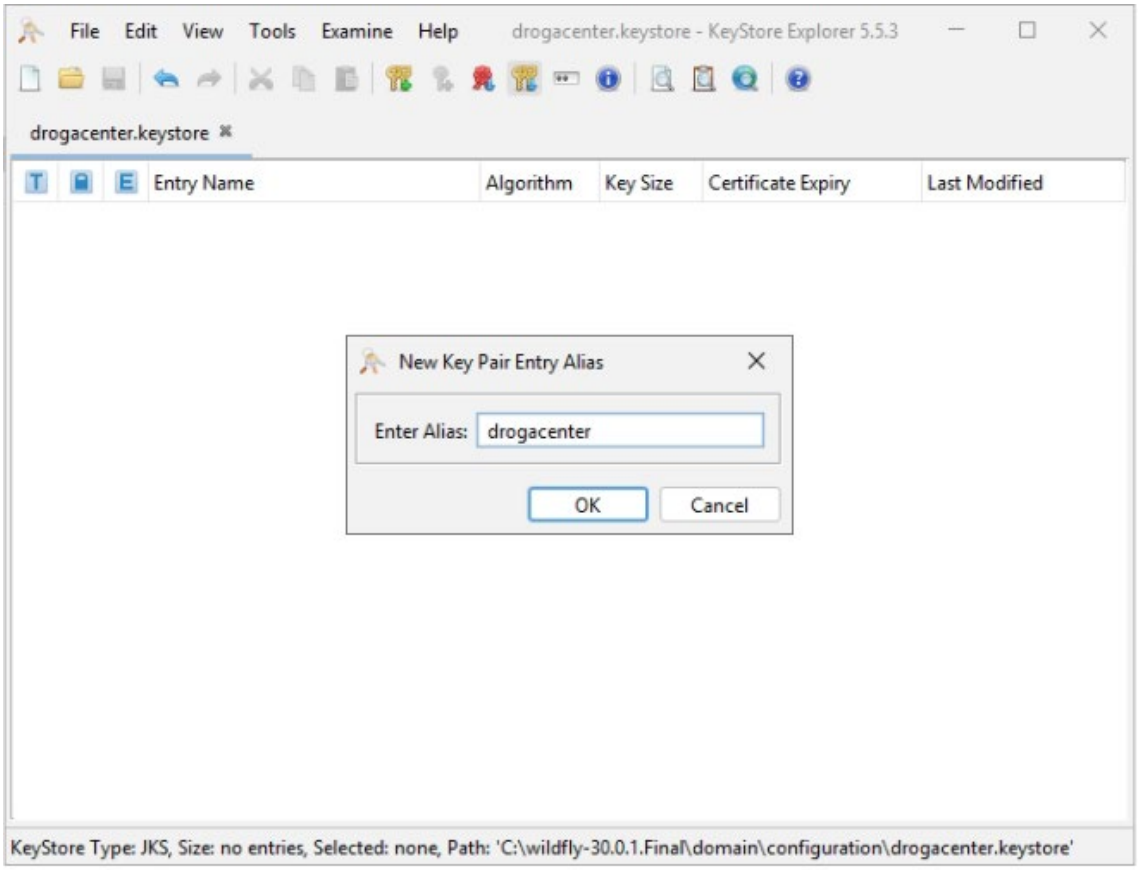
Na próxima etapa, será necessário selecionar o arquivo .pfx e inserir a senha definida no momento da geração do certificado.



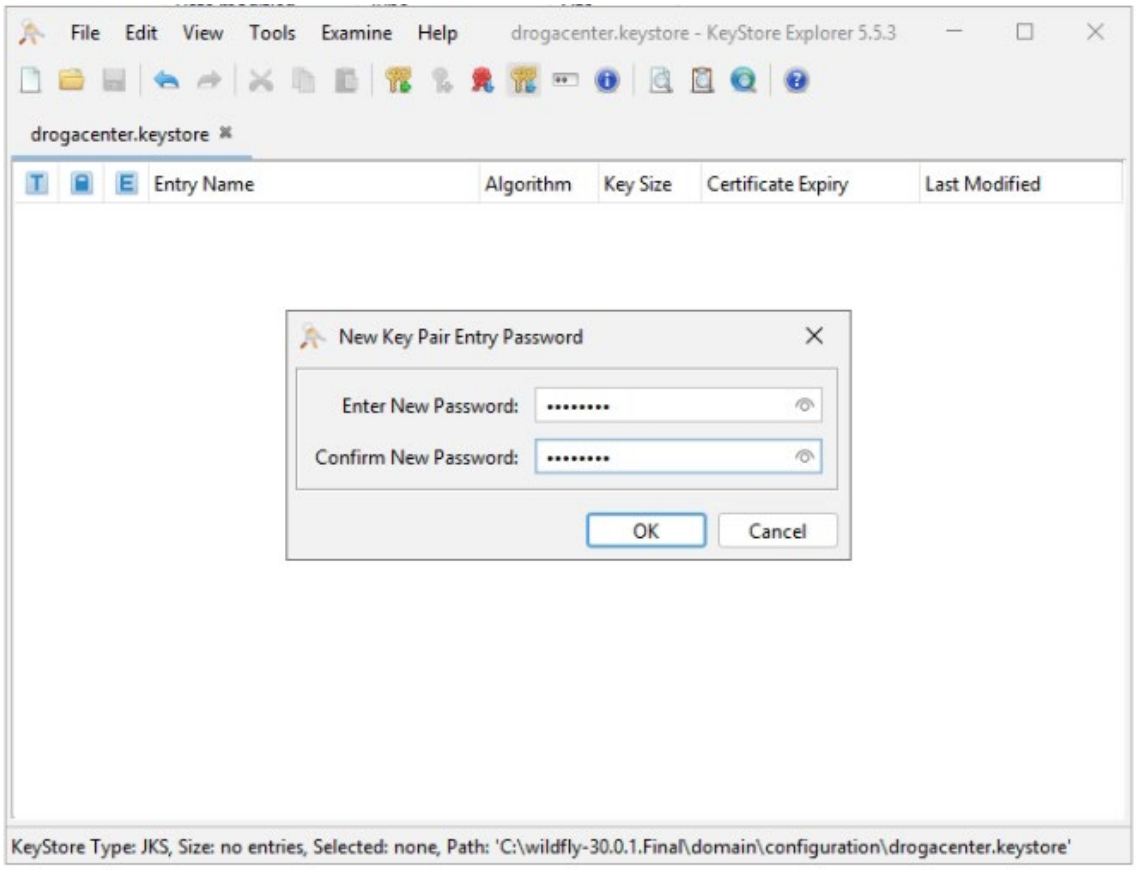
Importante: Certificado ser no formato PFX, pois ele já é completo. Geralmente este certificado precisa de senha.

Após confirmar a importação do certificado no formato PKCS #12, será solicitado o preenchimento de dois dados importantes:

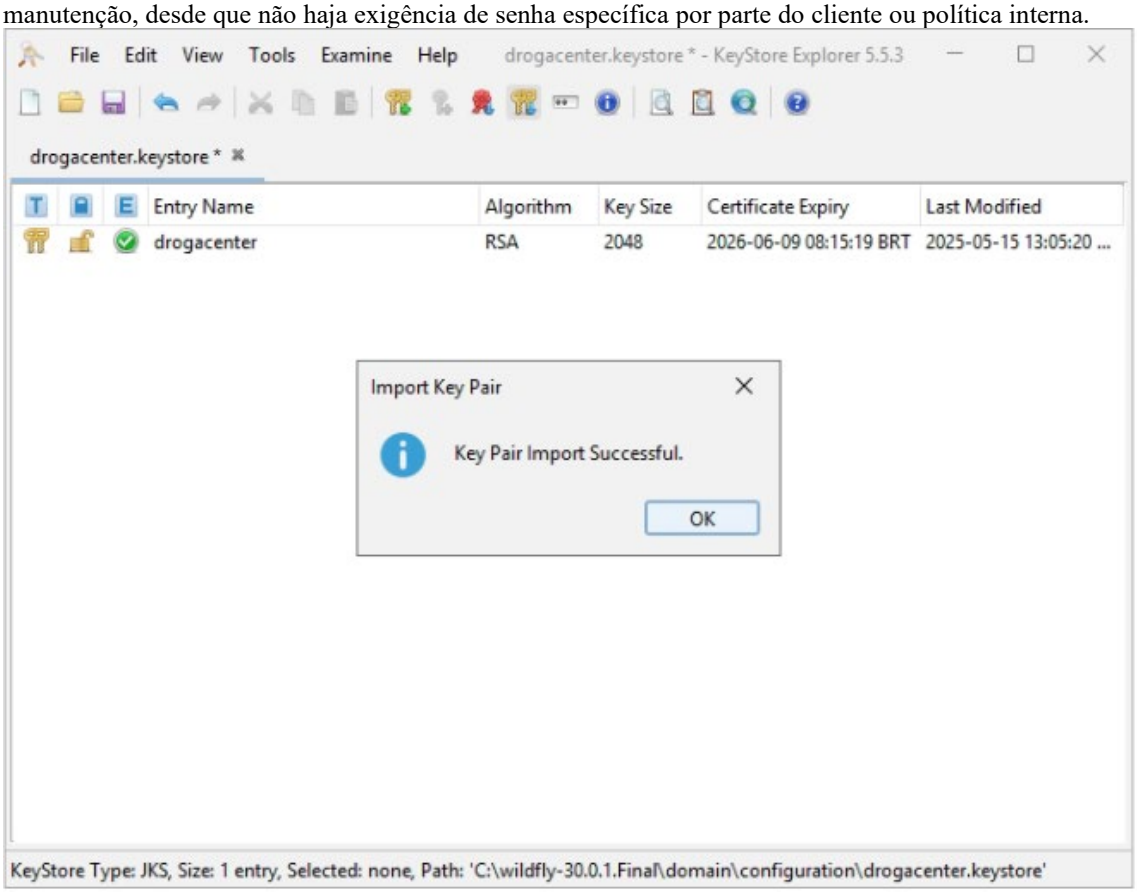
Alias da entrada: insira um nome identificador para o par de chaves (chave privada + certificado público). No exemplo, foi utilizado o nome [nome_cliente], refletindo o nome do cliente. Esse alias será utilizado posteriormente na configuração do WildFly e deve ser mantido de forma padronizada e descritiva.



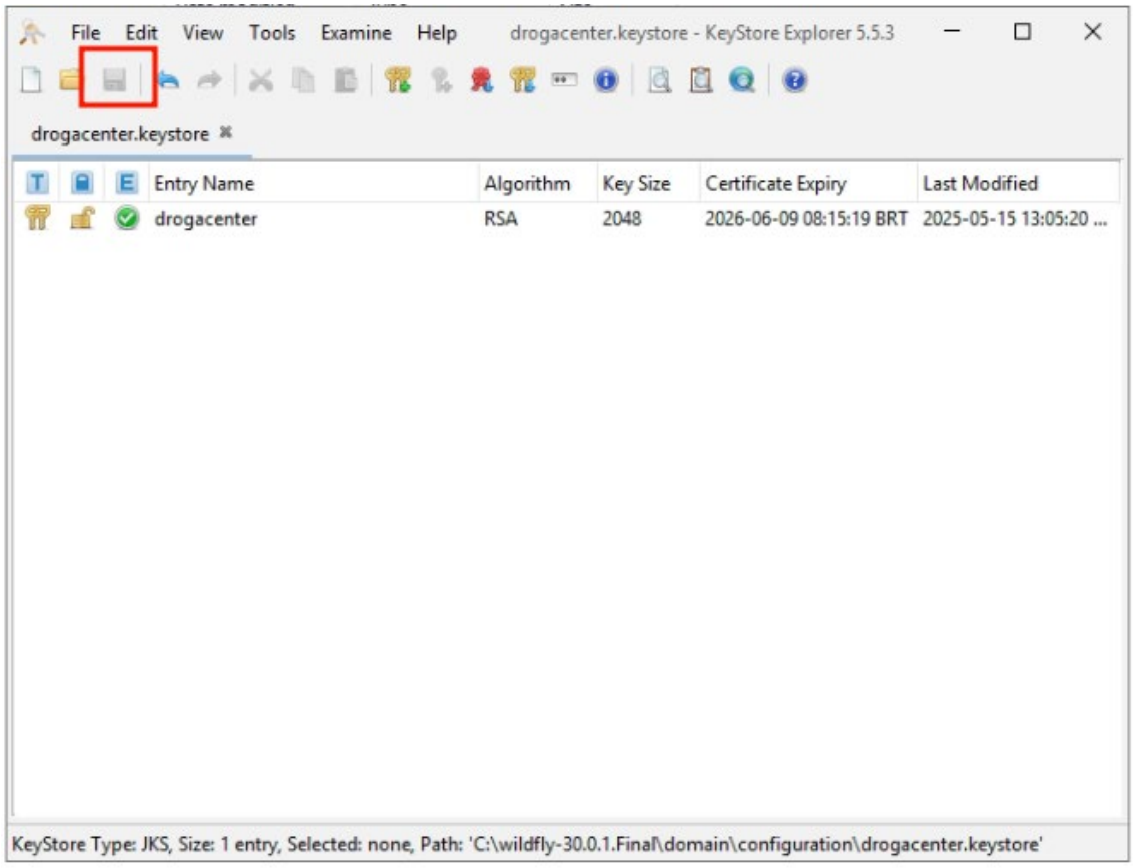
Senha da entrada: defina uma senha para proteger especificamente essa entrada dentro do keystore.



Recomenda-se utilizar a senha padrão changeit por questões de compatibilidade com servidores Java e facilidade de manutenção, desde que não haja exigência de senha específica por parte do cliente ou política interna.



Após concluir essas etapas, será exibida a mensagem "Key Pair Import Successful", indicando que o certificado foi importado com sucesso para dentro do keystore.



Por fim, clique no ícone de salvar no canto superior esquerdo (ícone de disquete) para persistir todas as alterações realizadas.

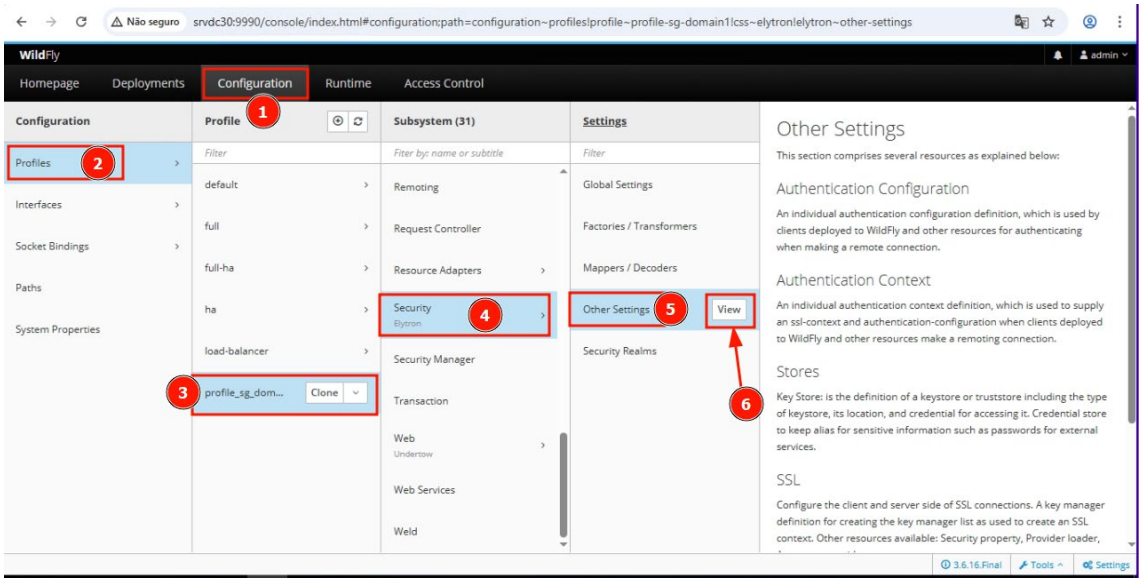
Com isso, o keystore está pronto para ser referenciado na configuração HTTPS do WildFly.

5) Configuração do WildFly

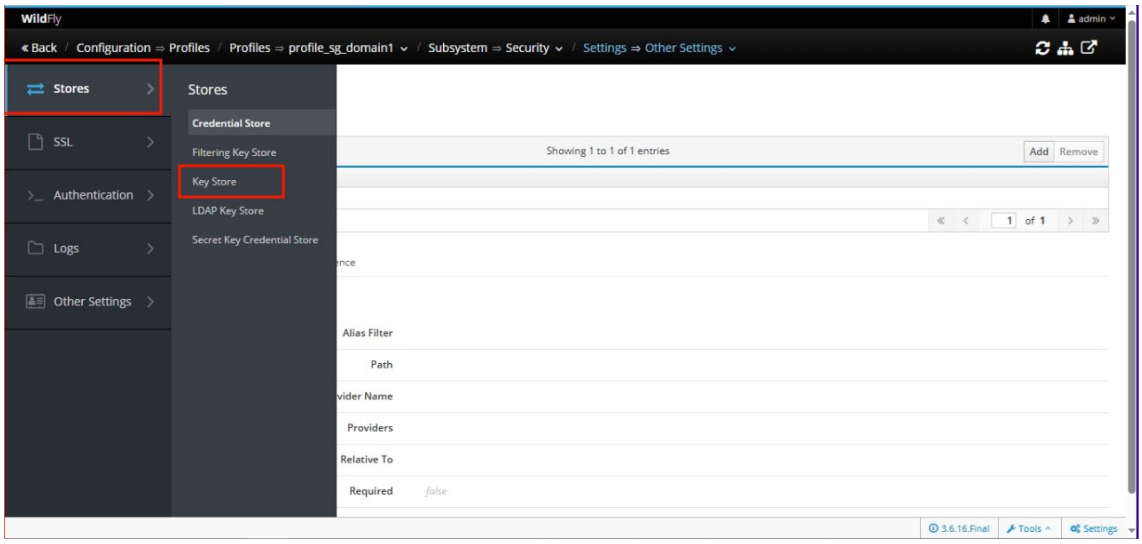
Com o keystore devidamente configurado, o próximo passo é registrar sua utilização no WildFly, vinculando-o ao serviço de HTTPS do servidor de aplicações.

Para isso, acesse o Console de Administração Web do WildFly, normalmente acessado via porta 9990 (ex: http://localhost:9990 ou http://<ip-servidor>:9990), e siga os passos abaixo:

- Clique na aba Configuration no topo do console;
- No menu lateral, selecione Profiles;
- Escolha o profile utilizado pelo domínio da aplicação — no exemplo, foi selecionado o perfil profile_sg_domain1;
- Dentro do profile, vá até o item Security, expandindo-o;
- Clique em Other Settings;
- Em seguida, clique no botão View, no canto superior direito da seção.



Essa navegação leva à tela onde serão configurados os detalhes de segurança necessários, como definição do keystore, alias e senha, além das propriedades associadas ao conector HTTPS.



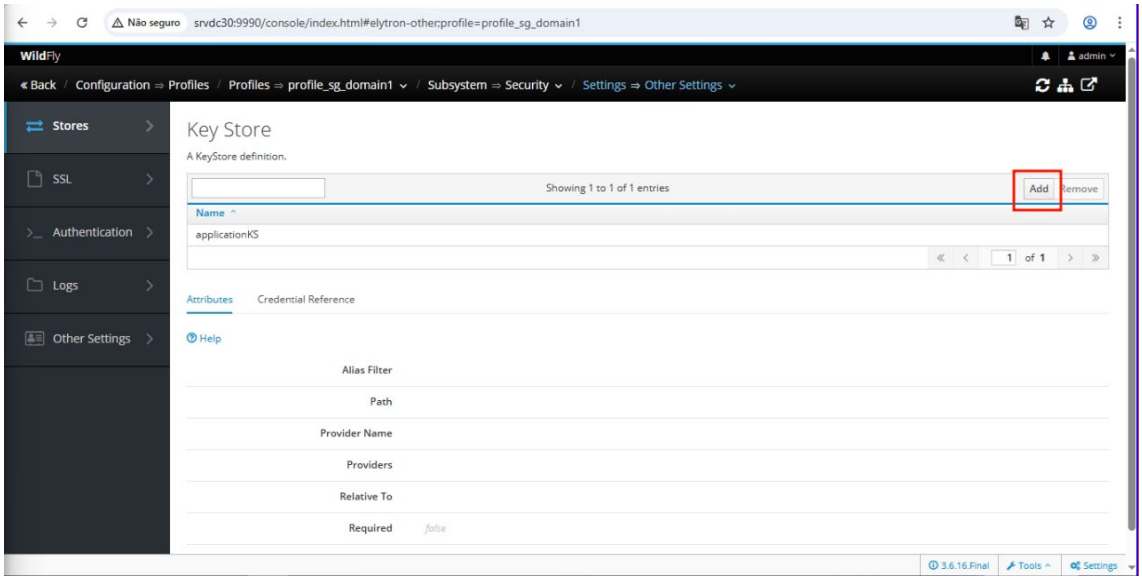
Com a tela Other Settings aberta, o próximo passo é realizar o cadastro do Key Store que será utilizado para a comunicação segura via HTTPS.

No menu lateral esquerdo:

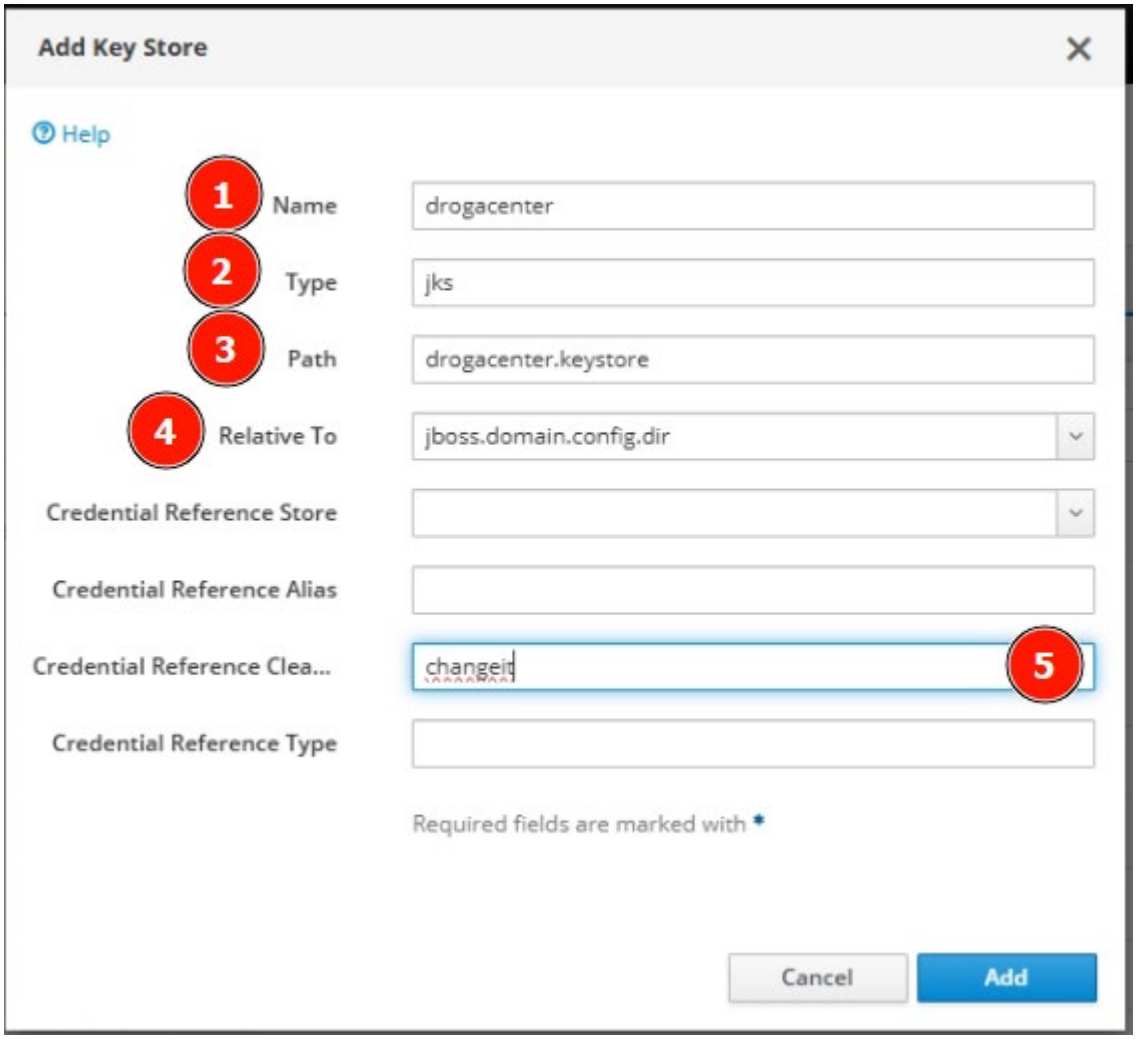
- Expanda a opção Stores;
- Em seguida, clique na opção Key Store.

Essa seção é responsável por cadastrar os arquivos .keystore ou .jks que o WildFly utilizará para a leitura do certificado digital. Será necessário informar dados como o caminho do arquivo, nome do provider, senha de acesso, alias e outras propriedades associadas.

Clique em Add no canto superior direito para iniciar o preenchimento de um novo registro de keystore.



Após clicar em Add na tela de Key Stores, preencha os campos conforme abaixo:



Explicação dos campos:

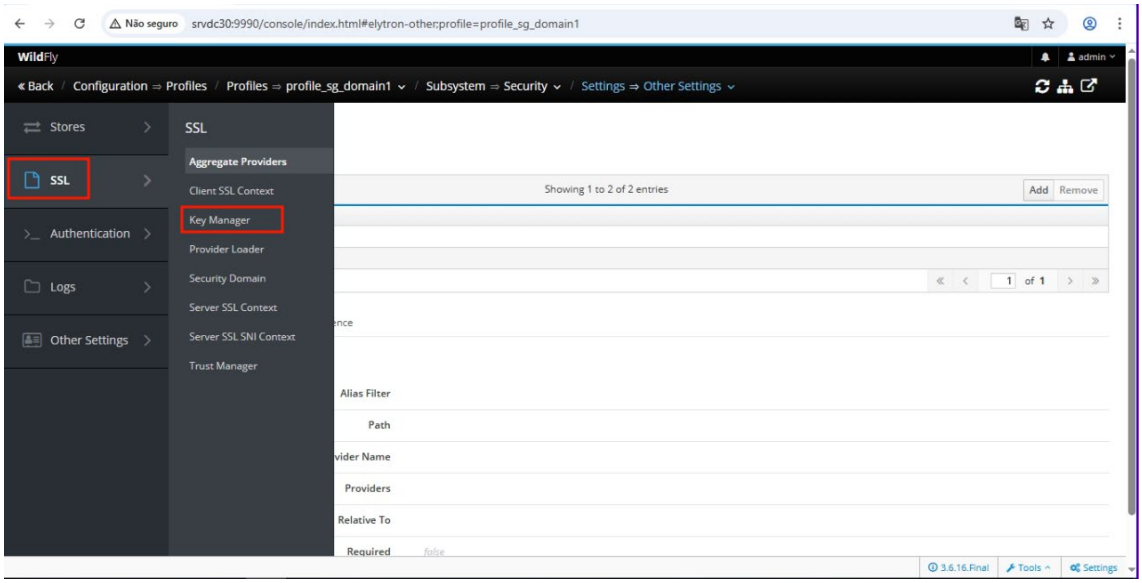
- Name: insira um identificador para este keystore dentro do WildFly. No exemplo, foi utilizado [nome_cliente], refletindo o nome do cliente.
- Type: informe o tipo de keystore. Para arquivos .keystore ou .jks, use jks.
- Path: informe o nome do arquivo salvo na pasta de configuração. Neste caso, [nome_cliente].keystore.
- Relative To: selecione a base de diretório onde o arquivo está salvo. Para keystores armazenados na pasta configuration, utilize a opção jboss.domain.config.dir.
- Credential Reference Clear-Text: insira a senha definida para acesso ao keystore. No exemplo, foi utilizada a senha padrão changeit.

Após preencher todos os campos obrigatórios, clique em Add para registrar o keystore no WildFly.

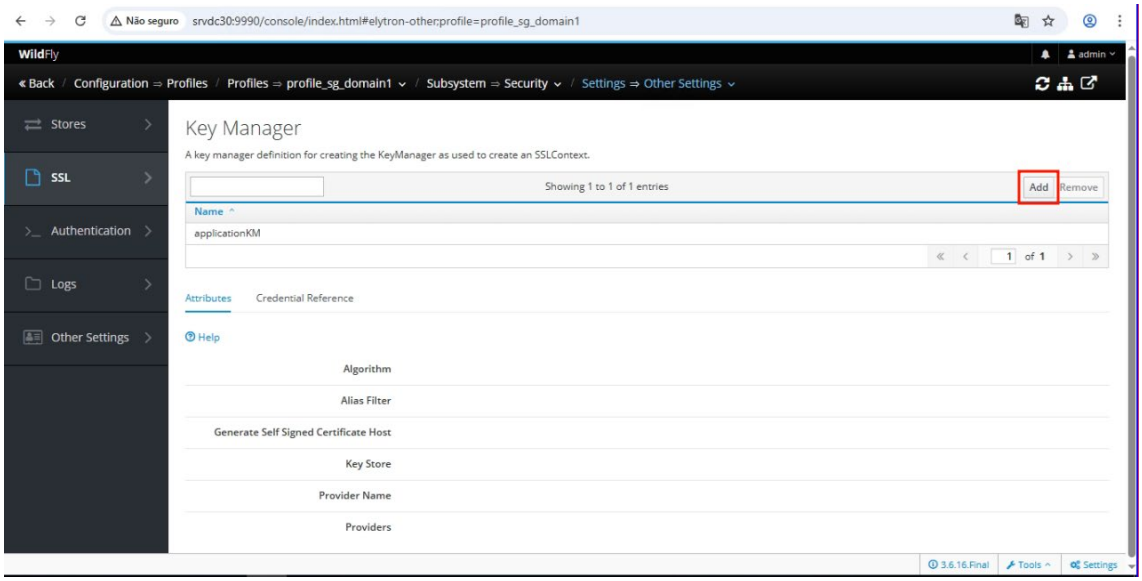
Com o Key Store cadastrado, o próximo passo é configurar o Key Manager, que será responsável por gerenciar as chaves privadas associadas ao keystore — etapa essencial para habilitar conexões SSL/TLS no WildFly.

No menu lateral esquerdo:

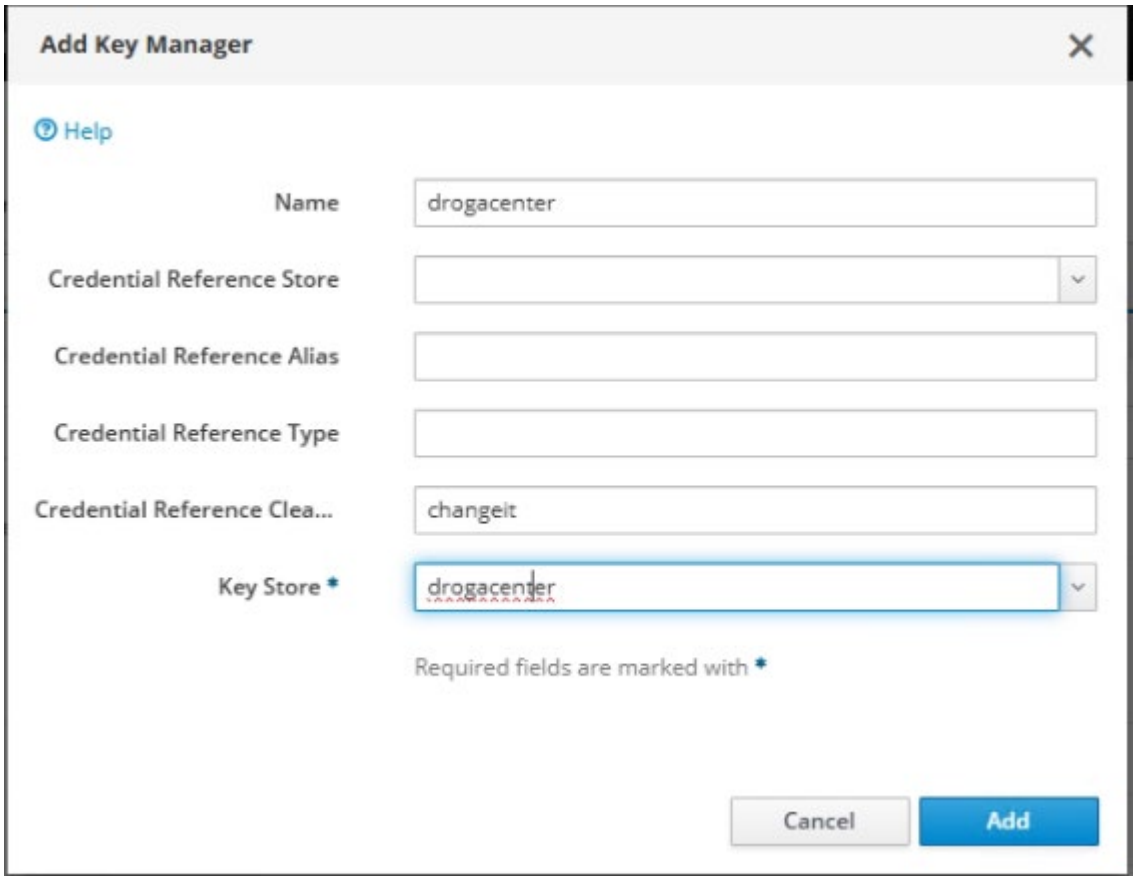
- Acesse a seção SSL;
- Em seguida, clique em Key Manager.



Essa configuração estabelece o vínculo entre o keystore (adicionado na etapa anterior) e o alias que identifica o par de chaves (certificado).
Após abrir essa seção, clique em Add para criar um novo gerenciador de chaves.



Na janela Add Key Manager, preencha os campos conforme os dados do keystore configurado anteriormente:



Explicação dos campos:

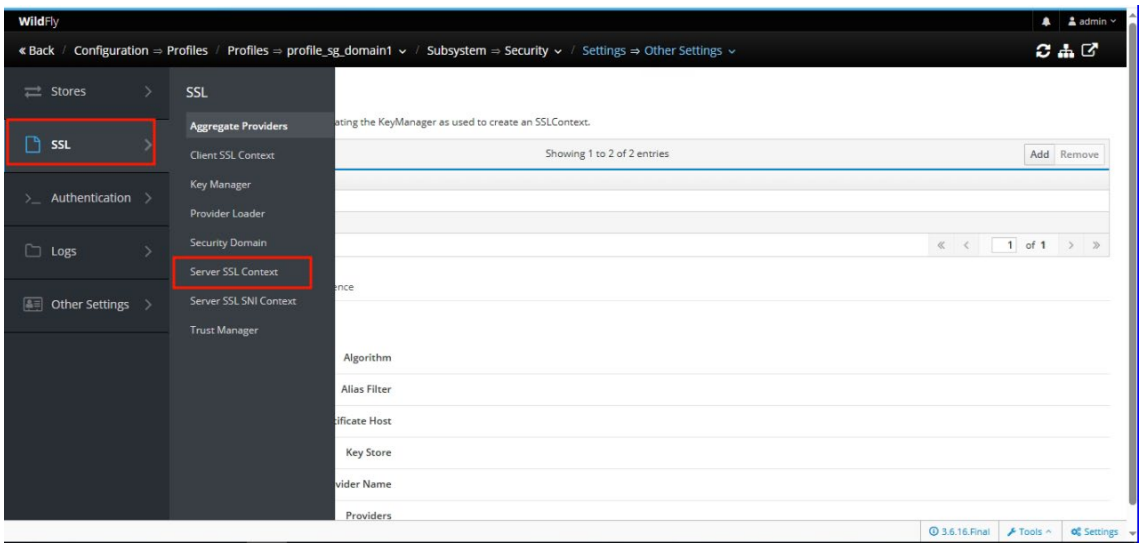
- Name: defina um nome para o gerenciador. No exemplo, foi utilizado [nome_cliente], mantendo o padrão de nome do cliente.
- Credential Reference Clear-Text: insira a senha utilizada para acessar a entrada do keystore. Aqui, foi usada a senha padrão changeit.
- Key Store: selecione o keystore que foi previamente registrado — no caso, [nome_cliente].

Os demais campos (Store, Alias e Type) podem ser deixados em branco, a menos que exista uma política de segurança mais avançada que exija preenchimento.

Clique em Add para concluir o cadastro do Key Manager.

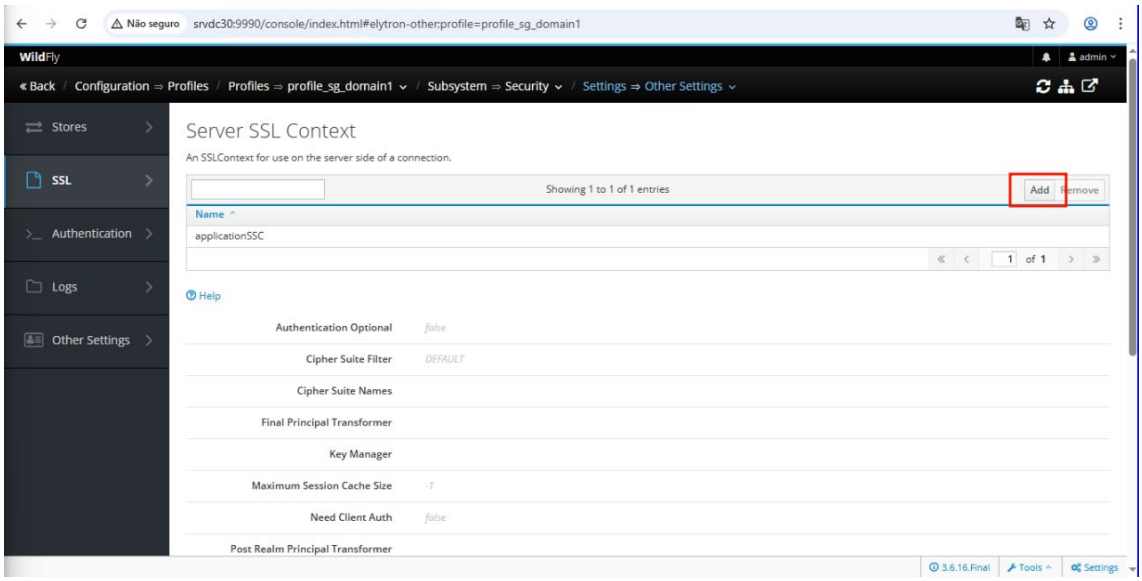
Após configurar o Key Manager, o próximo passo é criar o Server SSL Context, que representa o contexto SSL completo e será utilizado para habilitar conexões seguras no WildFly.

Esse contexto consolida o uso do keystore e do key manager configurados anteriormente, permitindo que o servidor realize comunicações HTTPS com base no certificado importado.



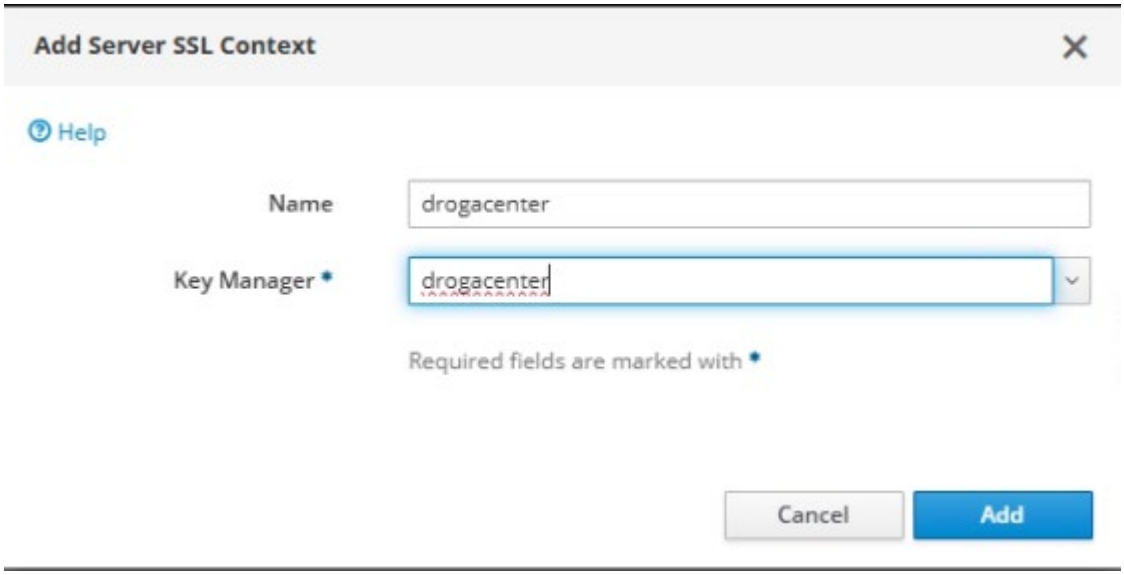
Para isso:

- Ainda no menu lateral esquerdo, dentro da seção SSL, clique na opção Server SSL Context.
- Em seguida, clique no botão Add no canto superior direito da tela para adicionar um novo contexto SSL.



Esse contexto será referenciado posteriormente no conector HTTPS configurado nas interfaces do servidor.

Na janela Add Server SSL Context, preencha os campos com as informações do Key Manager criado anteriormente:



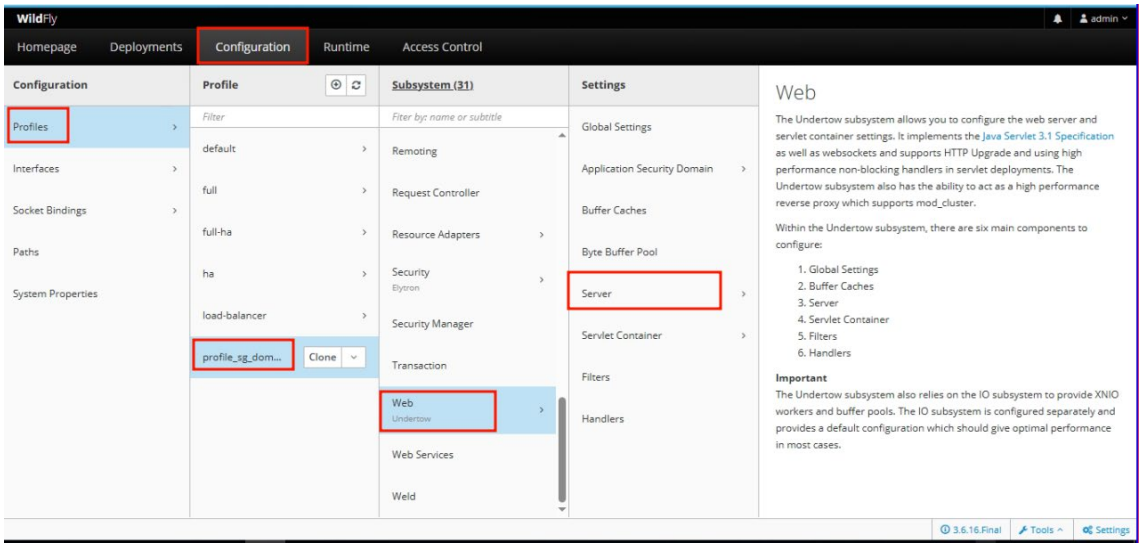
- Name: defina o nome do contexto SSL. No exemplo, foi utilizado [nome_cliente], mantendo a padronização do cliente.
- Key Manager: selecione o Key Manager configurado previamente — também com o nome [nome_cliente].

Esse vínculo garante que o contexto SSL utilize corretamente o certificado armazenado no keystore.
Após preencher os campos, clique em Add para concluir a criação do contexto SSL.
Esse contexto agora está pronto para ser associado ao conector HTTPS do WildFly.

Agora que o Server SSL Context foi configurado, o próximo passo é associá-lo ao conector HTTPS do servidor. Para isso, é necessário acessar a configuração do subsistema Web (Undertow), responsável pelo gerenciamento das conexões HTTP e HTTPS no WildFly.

Siga os passos abaixo:

- Vá novamente à aba Configuration no topo;
- No menu lateral, clique em Profiles;
- Selecione o perfil utilizado pela aplicação, como profile_sg_domain1;
- No painel do meio, acesse o subsistema Web (Undertow);
- No painel da direita, clique em Server, onde será possível editar as interfaces e listeners do servidor web.

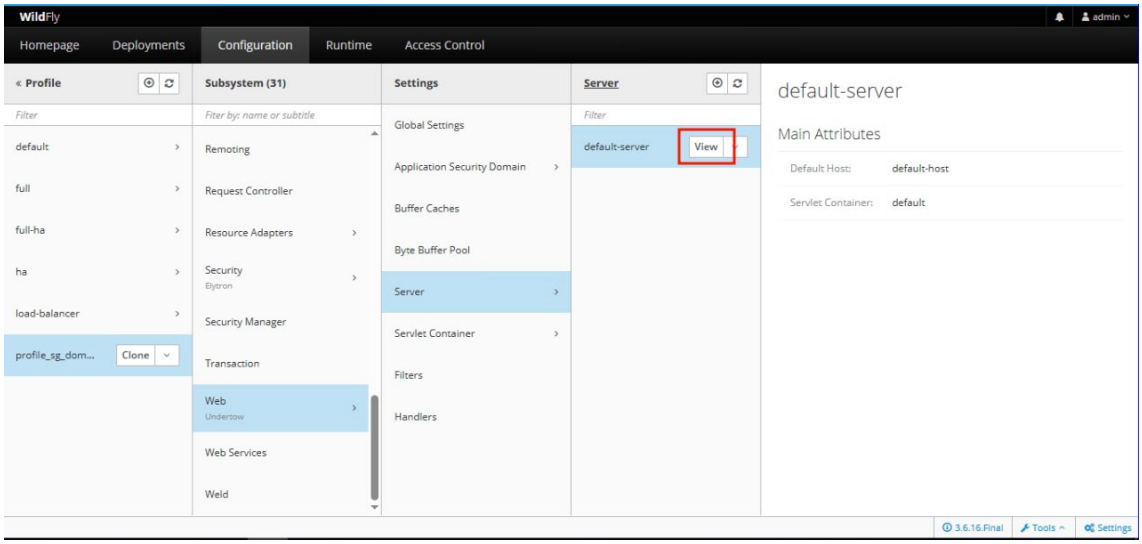


Essa seção é onde configuraremos a porta HTTPS e associaremos o contexto SSL previamente criado.

Na tela de configuração do subsistema Web, dentro da seção Server, será exibida a lista de servidores disponíveis. Por padrão, o WildFly utiliza o servidor chamado default-server.

Para continuar:

- Localize o item default-server na coluna do meio;
- Clique no botão View correspondente a ele, conforme destacado na imagem.

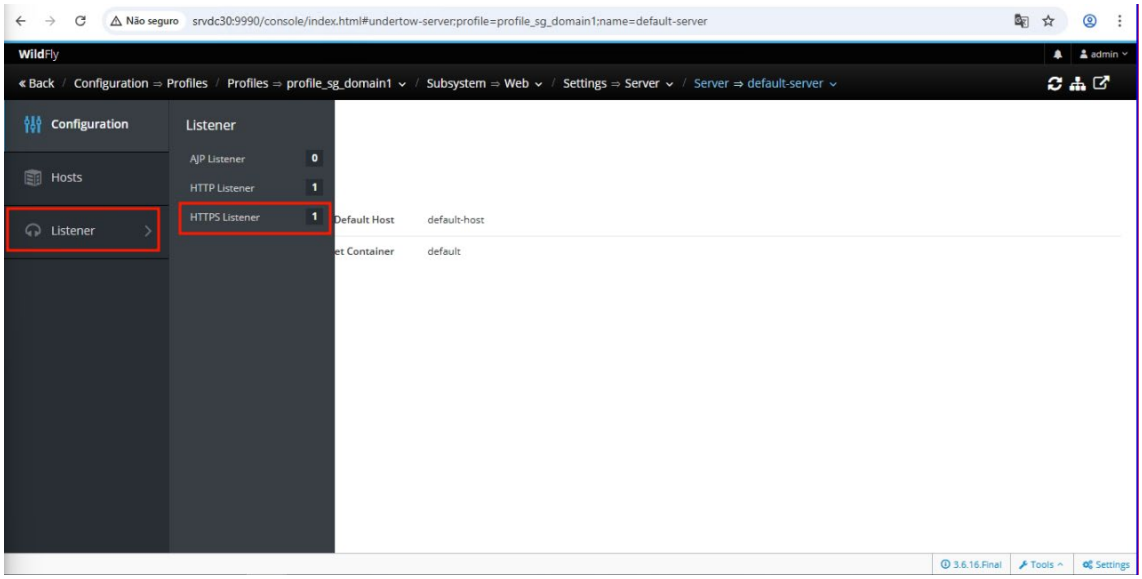


Esse acesso permitirá configurar os listeners, incluindo o conector HTTPS que utilizará o contexto SSL criado nas etapas anteriores.

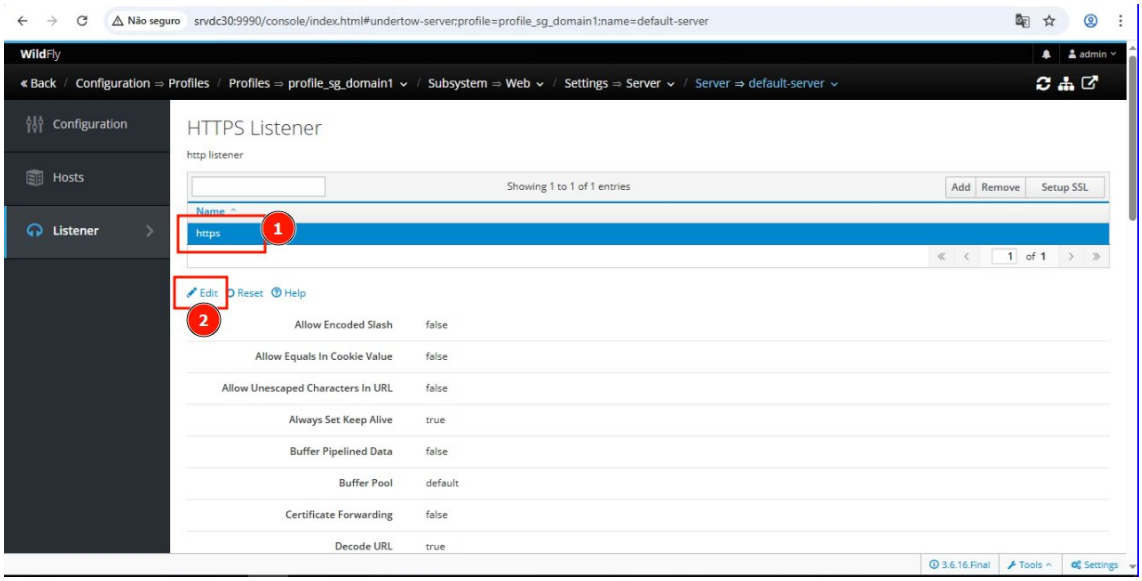
Dentro da configuração do default-server, acesse agora os conectores de escuta (listeners), responsáveis por gerenciar as portas de comunicação.

No menu lateral esquerdo:

- Clique em Listener;
- Em seguida, selecione HTTPS Listener.



Essa seção exibirá os listeners HTTPS configurados no servidor. Caso ainda não exista um listener configurado, será necessário adicioná-lo; caso já exista (como no exemplo), ele pode ser editado para associar o Server SSL Context criado anteriormente.
Na próxima etapa, vamos associar o contexto SSL correto ao listener HTTPS.

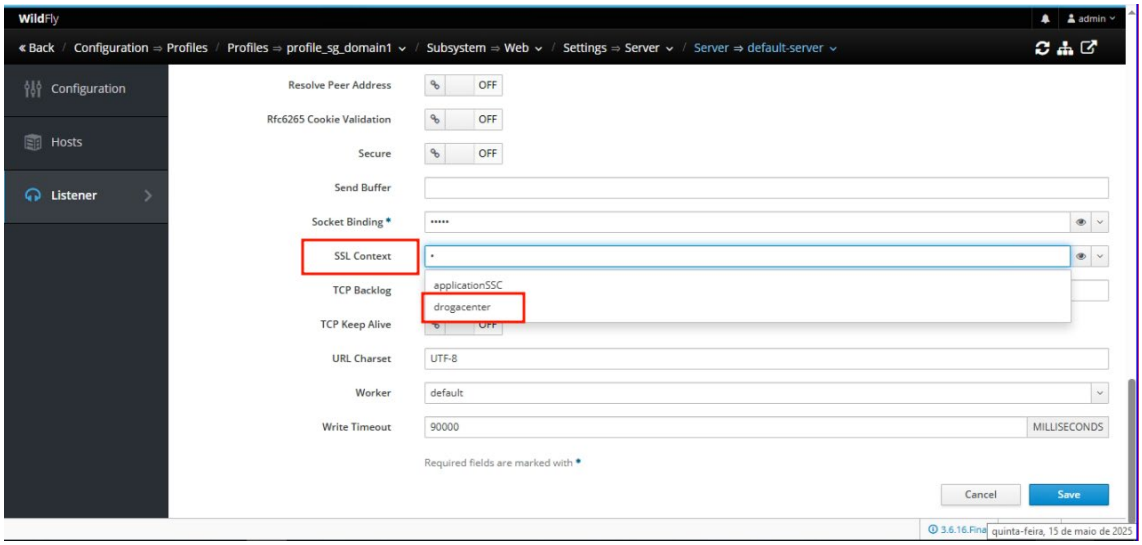


Na tela de HTTPS Listener, será exibido o listener existente, normalmente com o nome padrão https.
Para realizar a associação com o contexto SSL configurado:

- Clique sobre o nome do listener https para selecioná-lo;
- Em seguida, clique no botão Edit, localizado na parte inferior do painel lateral esquerdo.

Essa ação permitirá editar os atributos do listener, incluindo o vínculo com o Server SSL Context que foi criado anteriormente (ex: [nome_cliente]), viabilizando o funcionamento seguro da aplicação via protocolo HTTPS.

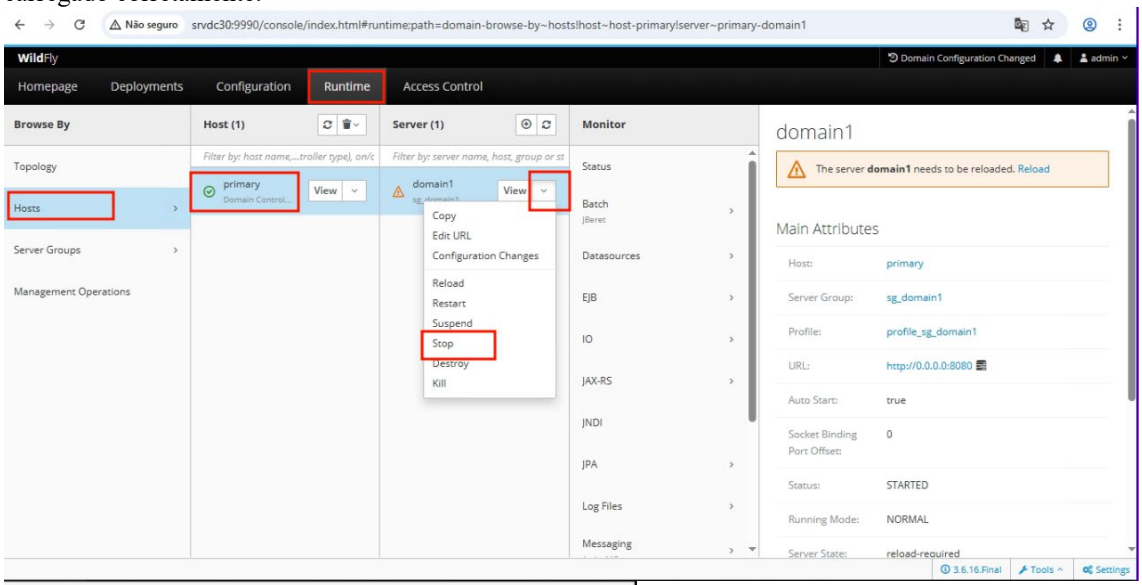
Na tela de edição do HTTPS Listener, localize o campo SSL Context, que define qual contexto de segurança será utilizado para este conector.

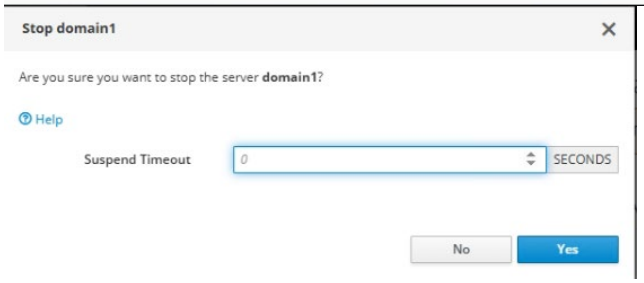


Clique no campo e selecione o Server SSL Context configurado anteriormente — no exemplo, [nome_cliente].
Essa seleção vincula o listener HTTPS ao certificado armazenado no keystore, garantindo que as conexões seguras sejam estabelecidas com base nas configurações definidas.

Após selecionar o contexto, clique em Save para aplicar as alterações.

Com todas as configurações aplicadas, é necessário reiniciar o domínio do WildFly para que o novo certificado digital seja carregado corretamente.



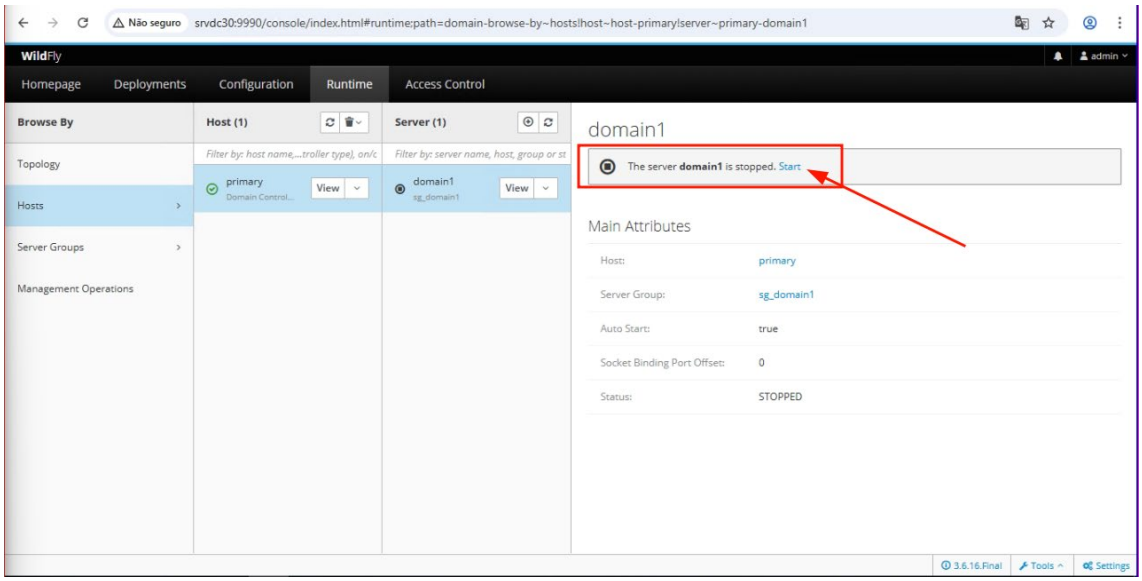


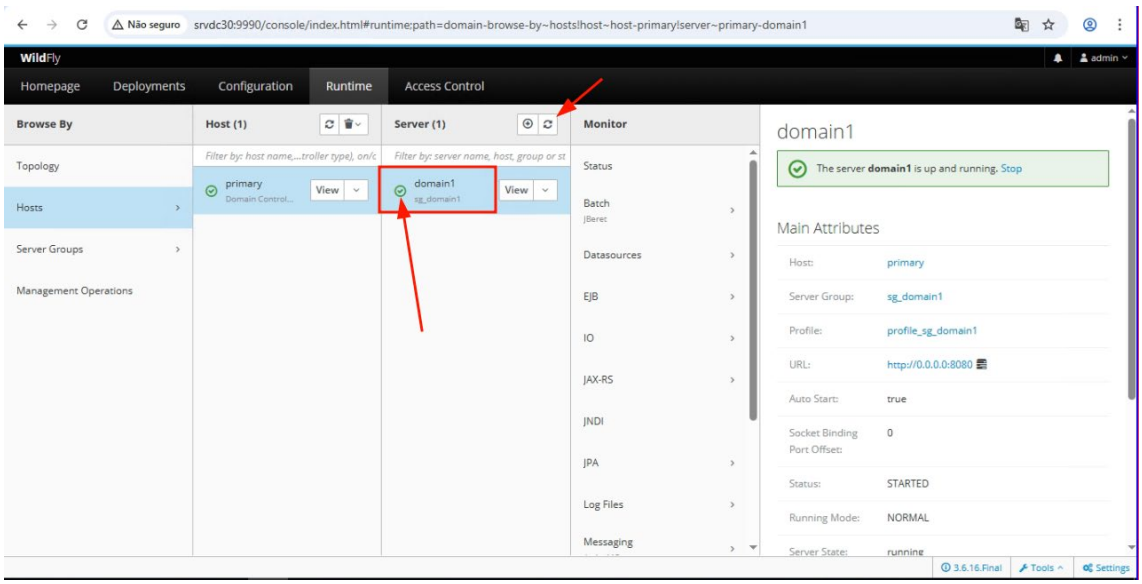
Reiniciando o domínio

- Acesse a aba Runtime no console do WildFly;
- No menu lateral esquerdo, clique em Hosts;

- Selecione o host principal (primary) e localize o servidor correspondente (ex: domain1);
- Clique na seta ao lado de View e selecione a opção Stop;
- Confirme a parada no diálogo exibido, mantendo o tempo de suspensão em 0 segundos;
- Aguarde até que o status do servidor mude para STOPPED;
- Clique no botão Start para iniciar novamente o domínio.

Durante esse processo, o console pode exibir alertas de timeout ou mensagens de erro temporárias. Isso é esperado. Aguarde até que o ícone do servidor fique verde, indicando que ele está executando normalmente.



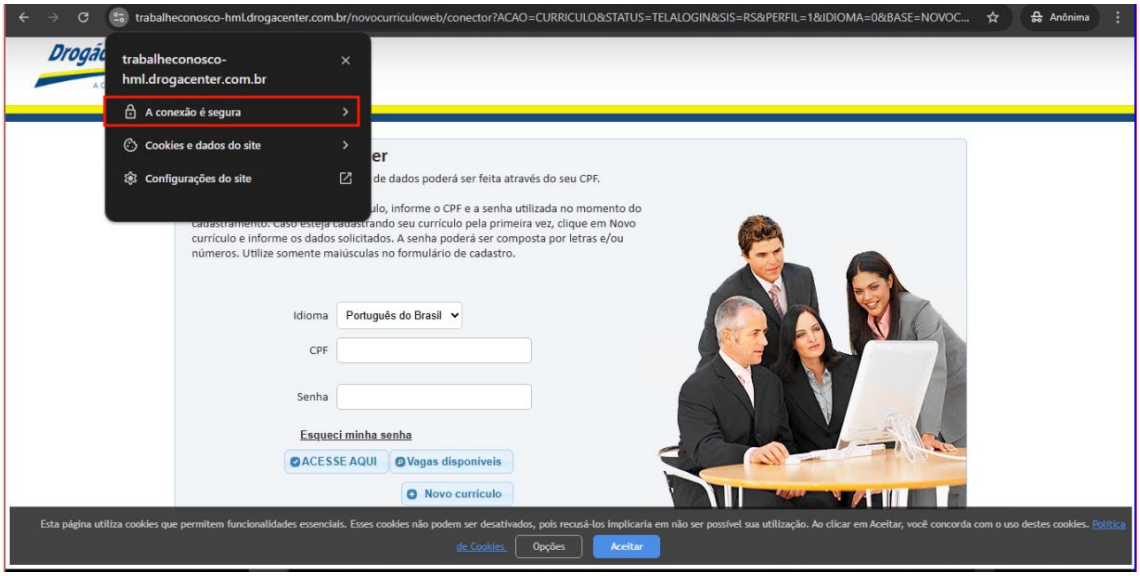


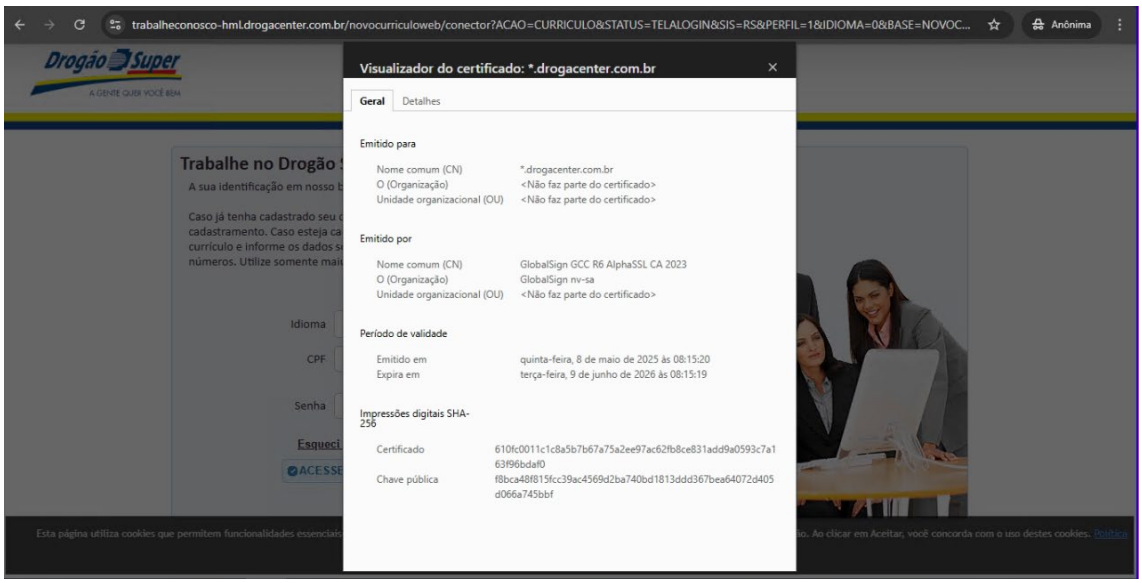
6) Validando o acesso HTTPS

Para validar se o certificado foi corretamente aplicado:

- Abra uma guia anônima no navegador (ou feche todas as abas abertas antes de testar);
- Acesse o endereço configurado (ex: https://trabalheconosco-html.[nome_cliente].com.br);
- Clique no ícone de cadeado ao lado da URL e verifique se aparece a mensagem "A conexão é segura";
- Visualize o certificado e confira os dados de validade, emissor, e domínio associado.

Se tudo estiver correto, o site estará operando com HTTPS funcional e certificado válido.





7) Pontos de atenção

Verificar se o certificado do cliente está no formato PFX, pois isso garante a cadeia completa dos certificados.
Solicitar senha do certificado gerado pelo cliente para implantação correta.
Substitua o texto [nome_cliente] pelo nome do cliente que está aplicando sem espaços, caracteres especiais ou letras maiusculas, por exemplo seniornoroeste.

8) Possíveis erros

Ainda não tivemos erros para reportar neste documento

Versão	Autor	Data	Comentários
1	Maicon Monttozo Batista	22/05/2025	Versão inicial