

Sistema: Atualização HCM (Sem Gestão Ponto) Área: Infraestrutura, tecnologia e desenvolvimento

Sumário

1)	Introdução	2
2)		
3)	Documentações utilizadas Senior	2
4)	Preparação do ambiente	
5)		
6)		
	Possíveis erros	
. ,	Erro ao Criar Domínio no WildFly com Socket-Binding Customizado	_



1) Introdução

Com a evolução das tecnologias utilizadas pela Senior Sistemas, o servidor de aplicações GlassFish, amplamente utilizado em versões anteriores, está sendo gradualmente substituído pelo WildFly, uma plataforma mais moderna, leve, segura e compatível com os padrões atuais da especificação Java EE (Jakarta EE).

Este documento tem como objetivo orientar de forma clara e detalhada o processo de atualização do sistema Senior e a migração automática do ambiente do GlassFish para o WildFly, utilizando o utilitário oficial seniorupdater.exe.

O conteúdo abrange desde a verificação inicial dos serviços e coleta de dados no SeniorConfigCenter até a execução completa do processo de atualização, contemplando a reconfiguração do domínio, apontamento da nova estrutura de servidor e validações pós-migração.

Acompanhe os passos a seguir com atenção para garantir uma migração segura e funcional, minimizando riscos de indisponibilidade ou falhas na execução dos módulos da Senior.

2) Pré-requisitos

- 1. Backup do banco de dados
- 2. Backup da pasta Senior ou Snapshot do servidor
- 3. Midia de instalação na versão 6.10.4.*
- 4. Instalação do Wildfly

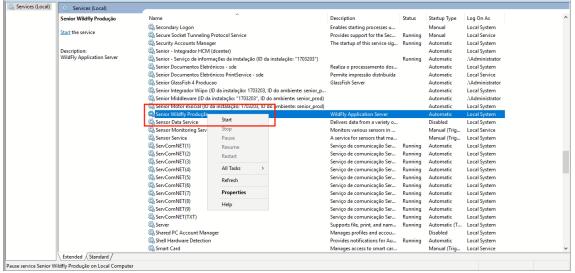
3) Documentações utilizadas Senior

https://documentacao.senior.com.br/tecnologia/5.10.4/informacoes-tecnicas/java/wildlfy/migracao-glassfish-wildfly.htm

4) Preparação do ambiente

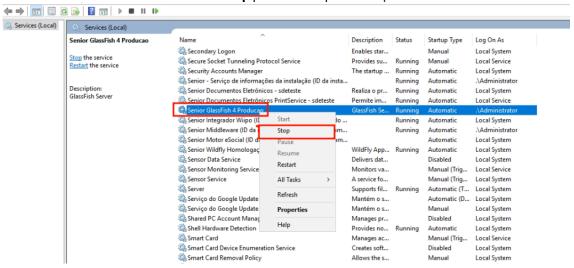
Antes de realizar qualquer validação no ambiente, é essencial garantir que o servidor de aplicações correto esteja ativo. Para isso, abra o **Gerenciador de Serviços do Windows** (services.msc) em cada um dos servidores ou domínios onde o sistema Senior está instalado.

Localize o serviço com o nome **Senior WildFly Produção** (ou nome equivalente configurado durante a instalação do novo ambiente) e verifique se seu status está como **Running** (em execução). Caso não esteja, clique com o botão direito sobre ele e selecione a opção **Start** para iniciá-lo.





Em seguida, ainda no Gerenciador de Serviços, localize o serviço antigo referente ao servidor GlassFish, geralmente nomeado como **Senior GlassFish 4 Produção**. Verifique se o status está **parado**. Caso ainda esteja em execução, clique com o botão direito sobre ele e selecione **Stop** para interromper o serviço.



Esse processo deve ser realizado em **todos os servidores/dominios** onde o ambiente Senior está instalado. Garantir que o **GlassFish esteja parado** e o **WildFly esteja em execução** evita conflitos de porta, problemas de carga dos módulos e garante que a aplicação seja executada corretamente no novo servidor de aplicação.

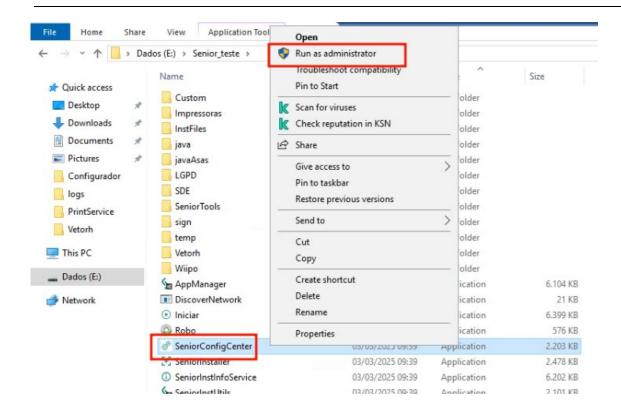
Com os serviços ajustados, o próximo passo é abrir o **SeniorConfigCenter** para coletar informações do servidor de aplicações atual no ambiente Senior.

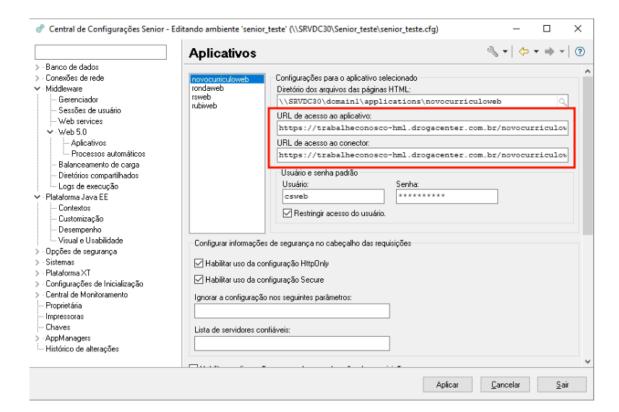
Acesse o diretório onde está localizado o ambiente, por exemplo: E:\Senior_teste

Dentro dessa pasta, localize e execute o utilitário **SeniorConfigCenter**. É recomendável clicar com o botão direito sobre o atalho ou executável e selecionar a opção **Executar como administrador**. Isso garante que todas as informações sejam acessadas corretamente e evita falhas ao manipular arquivos protegidos ou entradas de registro.

O utilitário exibirá informações como o caminho do servidor de aplicações em uso, a porta de administração, diretórios de deploy, entre outros parâmetros essenciais que deverão ser migrados para o novo ambiente com WildFly.







Com o SeniorConfigCenter aberto, navegue até o caminho:

Middleware > Web 5.0 > Aplicativos

Selecione o aplicativo desejado no painel da esquerda — por exemplo, novocurriculoweb. No painel da direita, serão exibidas as configurações de diretório e URLs associadas a esse aplicativo.



Anote as seguintes informações que serão reutilizadas na nova configuração com o WildFly:

- URL de acesso ao aplicativo
- URL de acesso ao conector
- Diretório dos arquivos das páginas HTML
- Usuário e senha padrão (se definidos)

Essas URLs apontam para o caminho onde o aplicativo estará acessível após a migração. Durante o processo, será necessário garantir que os mesmos caminhos estejam configurados corretamente no novo ambiente.

Essas informações também são fundamentais para validar a publicação após a substituição do servidor GlassFish pelo WildFly.

Copie essas informações e guarde em um lugar seguro para utilização posterior.

Faça isso para todas as aplicações web e contextos.

5) Atualização do Sistema (HCM)

Com todas as informações do ambiente salvas, é hora de executar o processo de atualização do sistema Senior, que também será responsável por realizar a migração automática do servidor de aplicações do GlassFish para o WildFly.

Acesse o diretório raiz do ambiente Senior — por exemplo:

E:\Senior teste

Localize o executável seniorupdater.exe, clique com o botão direito sobre ele e selecione a opção **Executar como administrador**.

Essa execução é obrigatória para que o sistema tenha as permissões necessárias para realizar alterações em arquivos protegidos, serviços do Windows e ajustes em configurações do ambiente.

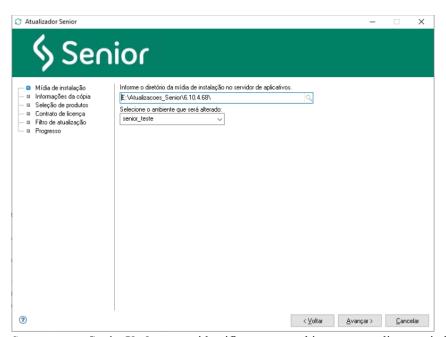


Com o SeniorUpdater iniciado, será exibida a tela para seleção da mídia de instalação e do ambiente que será atualizado. No campo de diretório, informe o caminho onde está localizada a mídia de atualização da Senior — por exemplo: E:\Atualizacoes_Senior\6.10.4.68\



Abaixo, no campo de seleção de ambiente, escolha o ambiente desejado — no exemplo, senior_teste. Essa escolha garante que todas as alterações aplicadas pelo atualizador afetem exclusivamente esse ambiente, sem comprometer outras instalações eventualmente presentes no servidor.

Após preencher os campos corretamente, clique em Avançar para prosseguir com a atualização.



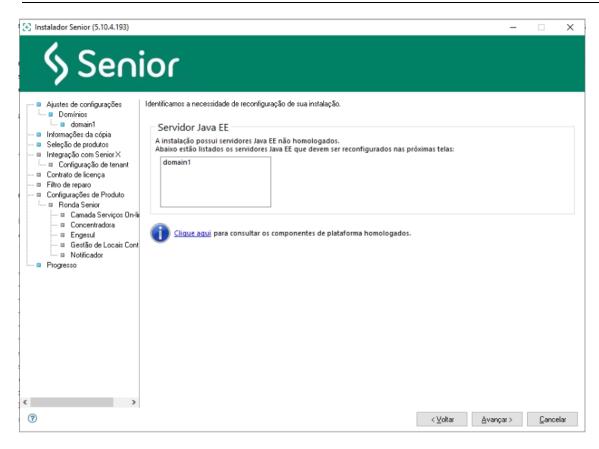
Sempre que o **SeniorUpdater.exe** identificar que o ambiente em atualização ainda utiliza um servidor **GlassFish**, será exibida automaticamente uma tela indicando a **necessidade de reconfiguração da instalação**.

Essa etapa informa que o ambiente possui **servidores Java EE não homologados** (ou seja, ainda baseados em GlassFish) e que é necessária a migração para um servidor homologado, como o **WildFly**.

Na lista apresentada, aparecerá o nome do domínio que será migrado — por exemplo, domain1. Esse domínio será reconfigurado nas próximas telas do instalador, onde as estruturas, caminhos e parâmetros específicos do WildFly serão aplicados automaticamente.

Clique em Avançar para prosseguir com a migração assistida.

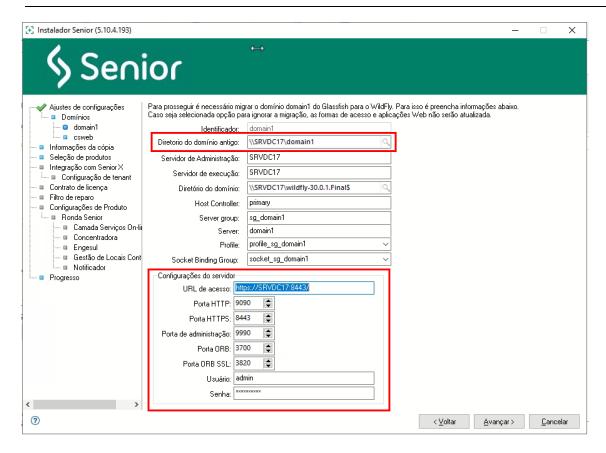




Na próxima etapa do processo de migração, será necessário configurar os parâmetros do novo domínio que será utilizado pelo **WildFly**, substituindo o antigo domínio baseado em **GlassFish**.

O instalador exibirá automaticamente os dados do domínio atual (como domain1). É importante ir passando os campos com a tecla de tabulação (TAB) para verificar se as informações estão corretas. Isso preencherá alguns campos com base na estrutura já existente no ambiente.





Você deverá indicar o novo diretório de domínio, apontando para o caminho onde o WildFly foi previamente instalado.

Por exemplo:

$\SRVDC30\$ wildfly-30.0.1. Final \$

Essa configuração informa ao sistema que, a partir de agora, o domínio domain1 será executado com base na nova estrutura do WildFly, localizada na pasta indicada.

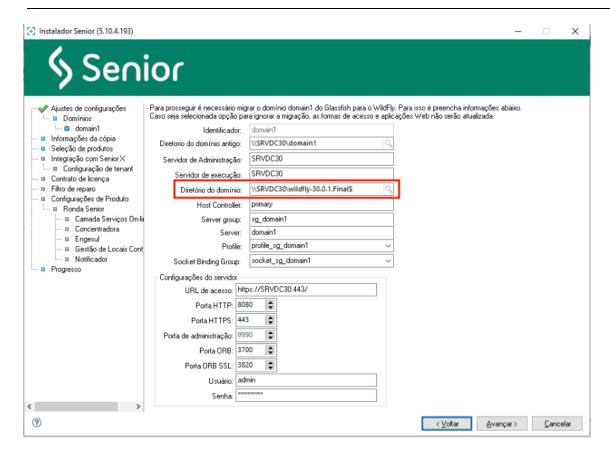
Revise também as demais configurações da tela:

- Host Controller: geralmente primary
- Grupo de Servidores, Server, Profile e Socket Binding Group: normalmente sugeridos automaticamente com base no domínio anterior
- Portas: revise as portas utilizadas no ambiente atual (HTTP, HTTPS, administração, ORB e ORB SSL)
- Usuário e Senha: utilize o login configurado previamente no WildFly (por exemplo, admin/adminadmin)

Atenção: Cuidado com as portas com valores baixos. Nessa caso, no momento da criação irá apresentar erro. Troque para uma porta com valor maior, como no print abaixo.

Caso apresentar erro no deploy, verifique na sessão de erros deste documento.



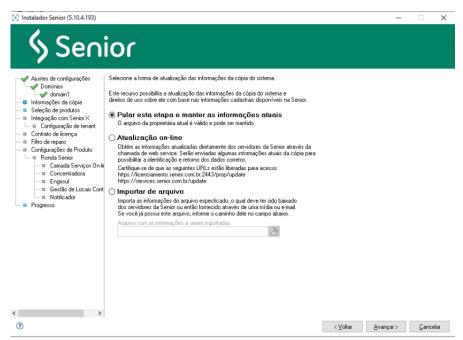


Após preencher todas as informações corretamente, clique em Avançar para prosseguir com a migração.

Na próxima tela, o instalador solicitará a forma de atualização das **informações da cópia do sistema**, também conhecidas como **metadados da instalação**. Essas informações incluem dados sobre a licença, cliente, módulos habilitados e registros de propriedade. Pode deixar selecionada a opção:

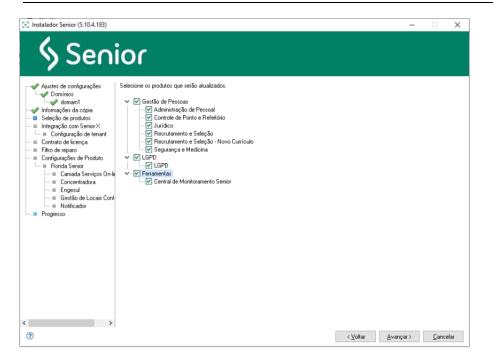
Pular esta etapa e manter as informações atuais.

Esta é a opção padrão e mais utilizada quando o ambiente já está configurado corretamente. O instalador manterá os dados existentes da cópia atual, sem modificações.

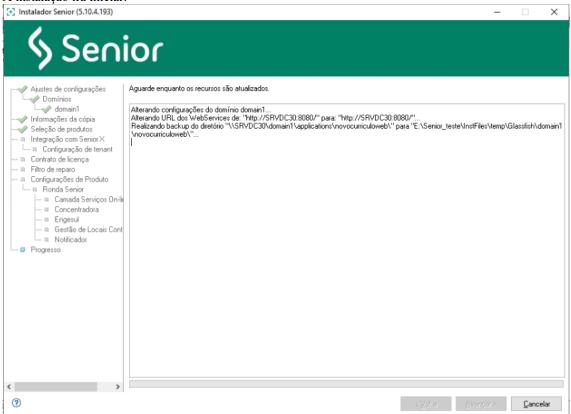


Selecione os produtos a serem atualizados no ambiente (recomendado todos).



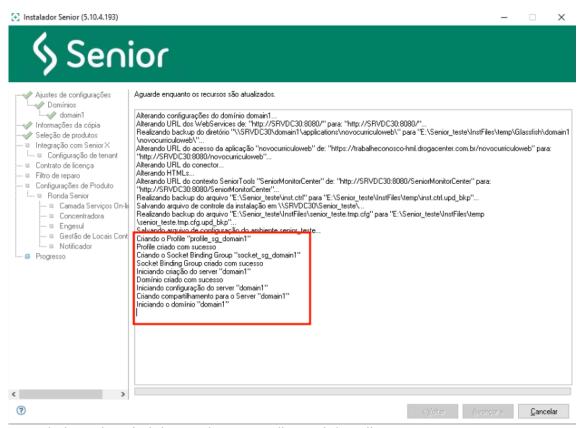


A instalação irá iniciar.

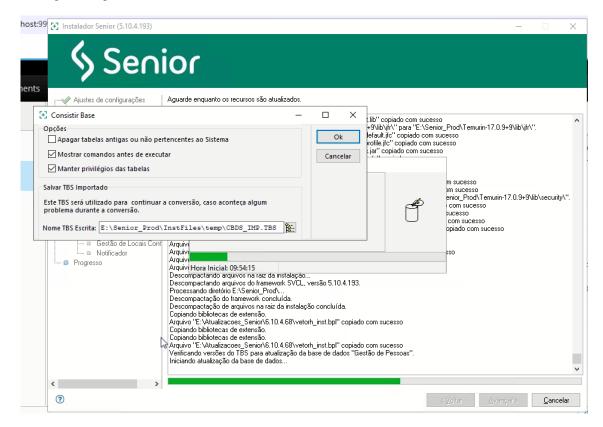


No log, é possível ver a criação de domínios no Wildfly.



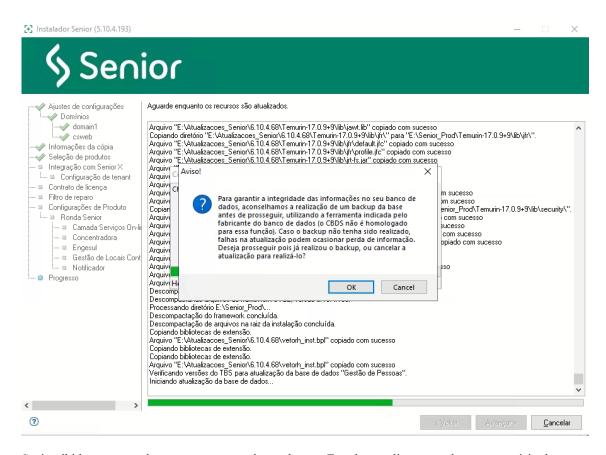


Uma tela de consistência de base se abre para atualizar os dados. Clique sempre na opção: Manter privilégios das tabelas Em seguida clique em OK.

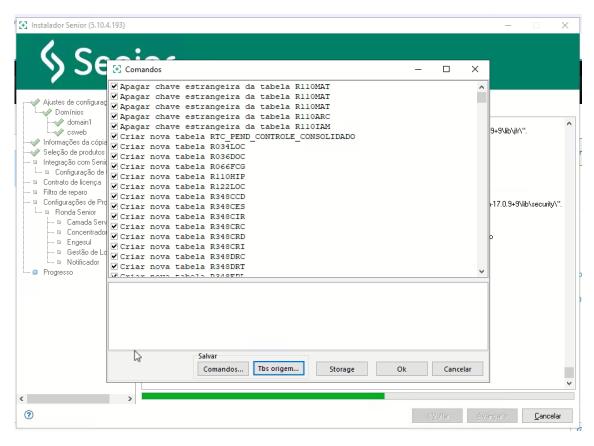


Após isso, uma mensagem de alerta de atualização da estrutura do banco de dados é exibida. Nela é exibida um aviso sobre a consistência e a importância do backup do ambiente.

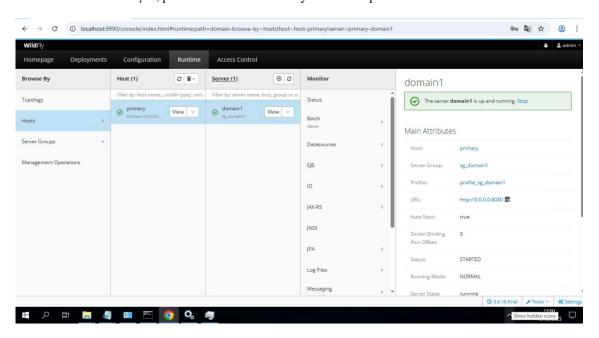




Será exibido os comandos a serem executados no banco. Em alguns clientes, pode ser necessário desmarcar algumas tabelas (USU_). Clique em Ok.



No término da atualização, podemos acessar o Wildfly e conferir que o domínio está no ar.



Lembre-se sempre de que ao final das atualizações, é necessário logar nos módulos e realizar as compilações das regras, bem como telas SGI.

Abaixo um exemplo de um WSDL nativo funcionando no Wildfly.

6) Pontos de atenção

Conferir variáveis de ambiente para iniciar o Wildfly corretamente Conferir que o Glassfish encontra-se parado no momento da atualização, pois isso pode resultar em conflito de portas

7) Possíveis erros

a) Erro ao Criar Domínio no WildFly com Socket-Binding Customizado

Durante a criação de um novo domínio no **WildFly 30.0.1.Final**, com a definição de um socket-bindinggroup customizado (socket_sg_csweb), foi identificado um erro crítico de boot no servidor de aplicações. Isso ocorre quando as portas estão abaixo do configurado (ex: 80, 443).

Erro Apresentado

Ao iniciar o domínio, a seguinte mensagem de erro foi exibida no LOG:

WFLYCTL0117: -6288 is an invalid value for parameter port. A minimum value of 0 is required

Além disso, o WildFly subia com falhas nos seguintes serviços:

WFLYCTL0013: Operation ("add") failed - address: [("subsystem" => "transactions")] Required services that are not installed:

- org.wildfly.network.socket-binding.txn-status-manager
- org.wildfly.network.socket-binding.txn-recovery-environment

E também:

WFLYCTL0013: Operation ("add") failed - address: [("subsystem" => "mail"), ("mail-session" => "default")]

Required services that are not installed:

- org.wildfly.network.outbound-socket-binding.mail-smtp

Diagnóstico

- O grupo de socket customizado socket_sg_csweb estava corretamente definido no domain.xml, porém as falhas ocorriam como se os bindings obrigatórios (txn-status-manager, txn-recovery-environment e mail-smtp) estivessem ausentes.
- A estrutura de servidor e servidor-grupo já referenciava corretamente socket_sg_csweb.

Conclusão: o erro estava relacionado ao fato de que o perfil (profile_sg_csweb) exigia bindings que **não estavam disponíveis no grupo socket_sg_csweb no momento da resolução** — o WildFly aparentemente ainda referenciava bindings do standard-sockets devido a alguma inconsistência no ciclo de boot ou fallback interno.

Solução Aplicada

Foi necessário garantir as seguintes correções no domain.xml:

1. Inclusão dos bindings obrigatórios no grupo socket_sg_csweb

2. Garantia de que o server-group usava o grupo de socket correto

```
<server-group name="sg_csweb" profile="profile_sg_csweb">
  <socket-binding-group ref="socket_sg_csweb"/>
</server-group>
```

3. Garantia de que o servidor utilizava o server-group e o grupo de socket corretamente

```
No host.xml:
```

4. Verificação da consistência no <profile name="profile_sg_csweb">

```
Foi validado que os subsistemas transactions e mail referenciavam corretamente os bindings:

<subsystem xmlns="urn:jboss:domain:transactions:..." ...>

<recovery-environment socket-binding="txn-recovery-environment"/>

<status-socket-binding socket-binding="txn-status-manager"/>

</subsystem>

<subsystem xmlns="urn:jboss:domain:mail:..." ...>

<mail-session name="default" jndi-name="java:jboss/mail/Default">

<smtp-server outbound-socket-binding-ref="mail-smtp"/>

</mail-session>
```

Após realização dos passos, é necessário reiniciar o serviço do Wildfly.

Resultado Final

</subsystem>

Após a aplicação das correções:

- O servidor csweb subiu corretamente no domínio.
- Todos os serviços foram inicializados com sucesso.
- O erro WFLYCTL0117 foi resolvido.
- As dependências de transações e e-mail foram atendidas.

Log final esperado:

WFLYSRV0025: WildFly Full 30.0.1. Final started in 6xxx ms

Observações

- Esse tipo de erro ocorre frequentemente quando há inconsistência entre os nomes de bindings utilizados nos perfis e os efetivamente disponíveis nos grupos de socket.
- Sempre verifique se o server-group e o profile apontam corretamente para os grupos de socket antes de iniciar o domínio.



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