

Adicionar botões de ação novos em meus server

Índice

[Introdução](#)

[Antes de Começar](#)

[Requisitos](#)

[Componentes Utilizados](#)

[Adicionando botões novos a meu server](#)

[Desenhista IAC e de serviço](#)

[Adicionando um botão novo](#)

[Informações Relacionadas](#)

[Introdução](#)

Este documento explica como adicionar um botão novo em meu popover dos server e ligá-lo com um serviço novo.

[Antes de Começar](#)

[Requisitos](#)

Não existem requisitos específicos para este documento.

[Componentes Utilizados](#)

A informação neste documento é baseada nestes versão de hardware e software.

- A seguinte configuração aplica-se às versões 4.0 IAC.

[Adicionando botões novos a meu server](#)

Criar um serviço novo ou alterar um serviço existente são além do alcance deste documento.

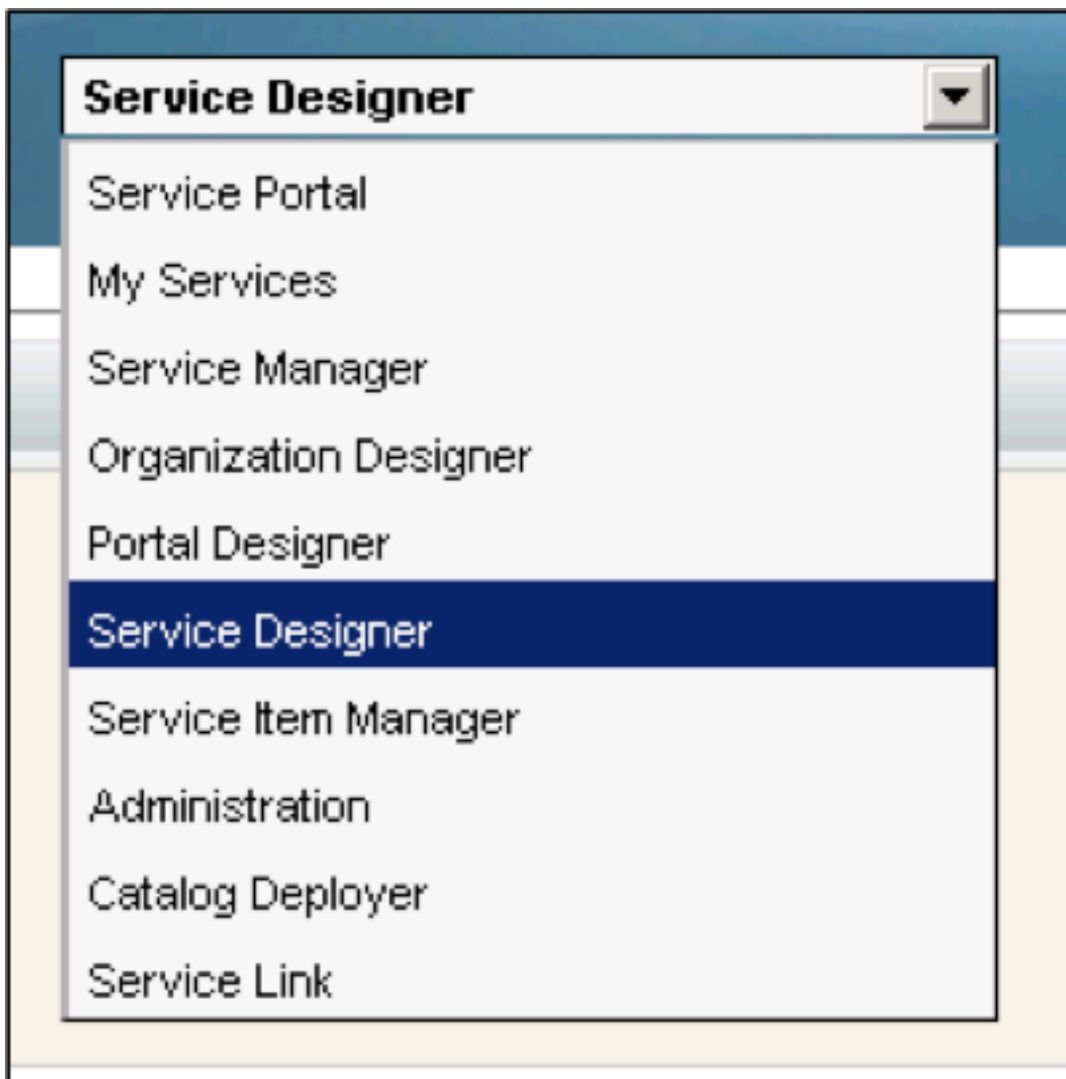
[Desenhista IAC e de serviço](#)

O procedimento para criar botões novos em meus server exige o uso do desenhista do serviço.

Adicionando um botão novo

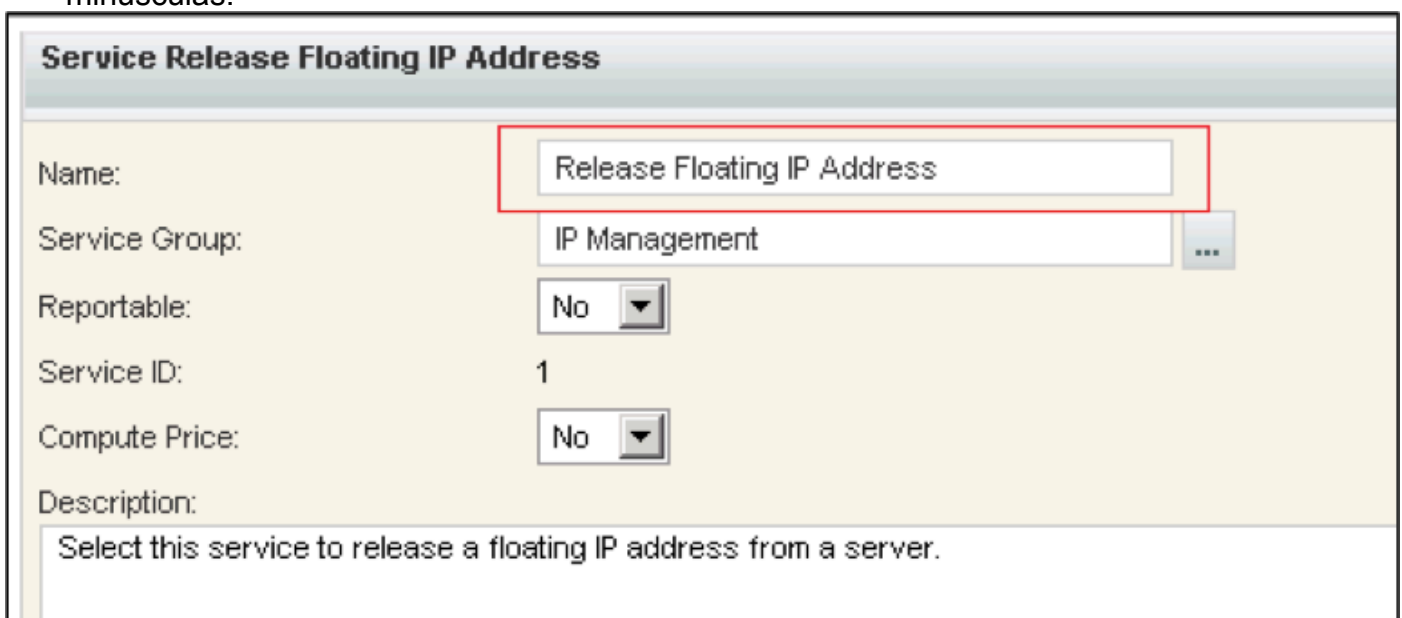
O exemplo seguinte ilustra as etapas para adicionar um botão novo “para liberar o endereço IP de Um ou Mais Servidores Cisco ICM NT de flutuação.”

Passo 1: Vá ao desenhista do serviço como mostrado abaixo no diagrama 1.



Passo 2: Crie um serviço e salvar seu nome para uso futuro.

- No diagrama 2 mostrado abaixo, o nome da “endereço IP de Um ou Mais Servidores Cisco ICM NT de flutuação liberação” é atribuído ao botão novo e é diferenciando maiúsculas e minúsculas.

A screenshot of a form titled "Service Release Floating IP Address". The form has a light yellow background and a grey header. It contains several fields: "Name:" with a text input field containing "Release Floating IP Address" (highlighted with a red border); "Service Group:" with a dropdown menu showing "IP Management" and a small grey button with three dots; "Reportable:" with a dropdown menu showing "No"; "Service ID:" with a text input field containing "1"; "Compute Price:" with a dropdown menu showing "No"; and "Description:" with a text area containing "Select this service to release a floating IP address from a server."

Passo 3: Encontre o diretório do desenvolvimento IAC e abra o arquivo “my-servers.html” para editar.

- “\ RequestCenterServer \ disposições \ RequestCenter.war \ costume \ IAC \ portlets \ my-servers.html”

Passo 4: Adicionar a nova linha à extremidade da seção de ServiceIds como mostrado abaixo no diagrama 3.

NOTA: O nome do serviço é diferenciando maiúsculas e minúsculas, e usa o “serviceId” como um segundo argumento.

```
223     serviceIds.push(getServiceProperties("Power Down Physical Server", "serviceId"));
224     serviceIds.push(getServiceProperties("Power Cycle Physical Server", "serviceId"));
225     serviceIds.push(getServiceProperties("Decommission Physical Server", "serviceId"));
226
227     serviceIds.push(getServiceProperties("Allocate Floating IP Address", "serviceId"));
228     serviceIds.push(getServiceProperties("Delete VM Firewall Rule", "serviceId"));
229     serviceIds.push(getServiceProperties("Release Floating IP Address", "serviceId"));
230
```

Passo 5: Abra o arquivo “IACServersPopover.js” para editar isso é ficado situado no “\ src \ widget \ iac \ IACServersPopover.js”

- Encontre a declaração variável: “buttonHTML”. Deve ser o número de linha próximo 160 como mostrado abaixo no diagrama 4.

```
159
160     var buttonHTML = '';
161     if(status == "INACTIVE") {
162
163         buttonHTML = "<p style=\`font-weight:bold; color: #455A73; margin: 5px 0 0 10px;\`>This "+serve
164     }
165     else {
166
167         if(serverType == "Virtual Machine (VM)" && powerState == "Off") { Start of conditional block
168
169             if(podType == "OpenStack Cloud Manager"){
170                 serviceNames = ["Power Up Virtual Machine", "Modify Configuration", "View Snapshots",
171                 serviceIDs = [idsArray[0], idsArray[3], idsArray[4], idsArray[5], idsArray[6], idsArray
172                 cssClassNames = ["powerUp", "modifyConfiguration", "viewSnapshots", "takeSnapshot", "
173             }
174             else if(podType == "Amazon EC2") {
175                 serviceNames = ["Power Up Virtual Machine", "View Snapshots", "Take Snapshot", "Revert
176                 serviceIDs = [idsArray[0], idsArray[4], idsArray[5], idsArray[6], idsArray[7], idsArra
177                 cssClassNames = ["powerUp", "viewSnapshots", "takeSnapshot", "revertSnapshot", "deleteS
178             }
179             else {
180                 //All actions except Power Down, Power Cycle
181                 serviceNames = ["Power Up Virtual Machine", "Modify Configuration", "View Snapshots", "
182                 serviceIDs = [idsArray[0], idsArray[3], idsArray[4], idsArray[5], idsArray[6], idsArray
183                 cssClassNames = ["powerUp", "modifyConfiguration", "viewSnapshots", "takeSnapshot", "re
184                 if(podType == "VMware vCenter Server") {
185                     serviceNames.push("Convert VM to Template");
186                     serviceIDs.push(idsArray[10]);
187                     cssClassNames.push("convertVMtoTemp");
```

- Depois da declaração do “buttonHTML” são as circunstâncias adicionais arranjadas pelo tipo, pelo estado e pelos papéis de usuário VM.

Passo 6: Introduza o seguinte código com circunstâncias apropriadas para mostrar a ação para cada papel específico e tipo PE, antes de chamar o código dos “generateActionButton” (situado antes da linha no.: 198) como mostrado abaixo no diagrama 5.

```

201
202     serviceNames.push("Release Floating IP Address");
203     serviceIDs.push(idsArray[20]);
204     cssClassNames.push("releaseFIP");

```

- Nos serviceNames, empurre o nome do serviço, neste exemplo da “endereço IP de Um ou Mais Servidores Cisco ICM NT de flutuação liberação.”
- Nos serviceIDs, empurre o valor de índice apropriado de idsArray, neste exemplo que é o deslocamento predeterminado 20, o último esforço do serviço dos “getServiceProperties” em my-servers.html.
- Nos cssClassNames, empurre a classe CSS de botão.

Passo 7: O arquivo em aberto “IACGenerateActionButtons.js” para editar e encontra a função: “generateMyServersGroupButtons”.

- Na extremidade deste laço da função mostrado abaixo no diagrama 6, mude os valores destacados de acordo com seu serviço:

```

        else if (serviceNames[i] == "Release Floating IP Address" && add
extra validations if required){
            networkServicesGroup+="<div class=\"popover_main\"
id=\"releaseFIPCont\"><span class=\"popover_sub\"><button type=\"button\"
id=\"releaseFIP\" class=\"\" + cssClassNames[i] + \"\" title=\"\" + serviceNames[i] +
\"\"+ alt=\"\" + serviceNames[i] + \"\"></button></span><span class=\"popover_label
popover content\" id=\"releaseFIPLabel\">"+serviceNames[i]+\"</span></div>";
        }

```

Você pode mudar o networkServicesGroup a um do seguinte baseado em suas exigências:

- lifecycleManagementGroup
- networkServicesGroup
- serverOpsGroup

Passo 8: A construção, é executado e distribuí para verificar que o botão novo está criado.

[Informações Relacionadas](#)

- [Cisco Intelligent Automation para a nuvem](#)