Contenido

Introducción
Antes de comenzar
Requisitos
Componentes Utilizados
Agregar los nuevos botones a mi VDC
Diseñador IAC y del servicio
Agregar un nuevo botón
Información Relacionada

Introducción

Este documento explica cómo agregar un nuevo botón en mi popover VDC y atarlo con un nuevo servicio.

Antes de comenzar

Requisitos

No hay requisitos específicos para este documento.

Componentes Utilizados

La información en este documento se basa en estas versiones de software y hardware.

- Cisco Intelligent Automation for Cloud 4.0 (cualquier edición) y servicio Deisgner

Agregar los nuevos botones a mi VDC

En esta sección, le presentan con los pasos para agregar un nuevo botón de acción al mi popover VDC. Crear un nuevo servicio o la modificación de un servicio existente está fuera del alcance de este documento.

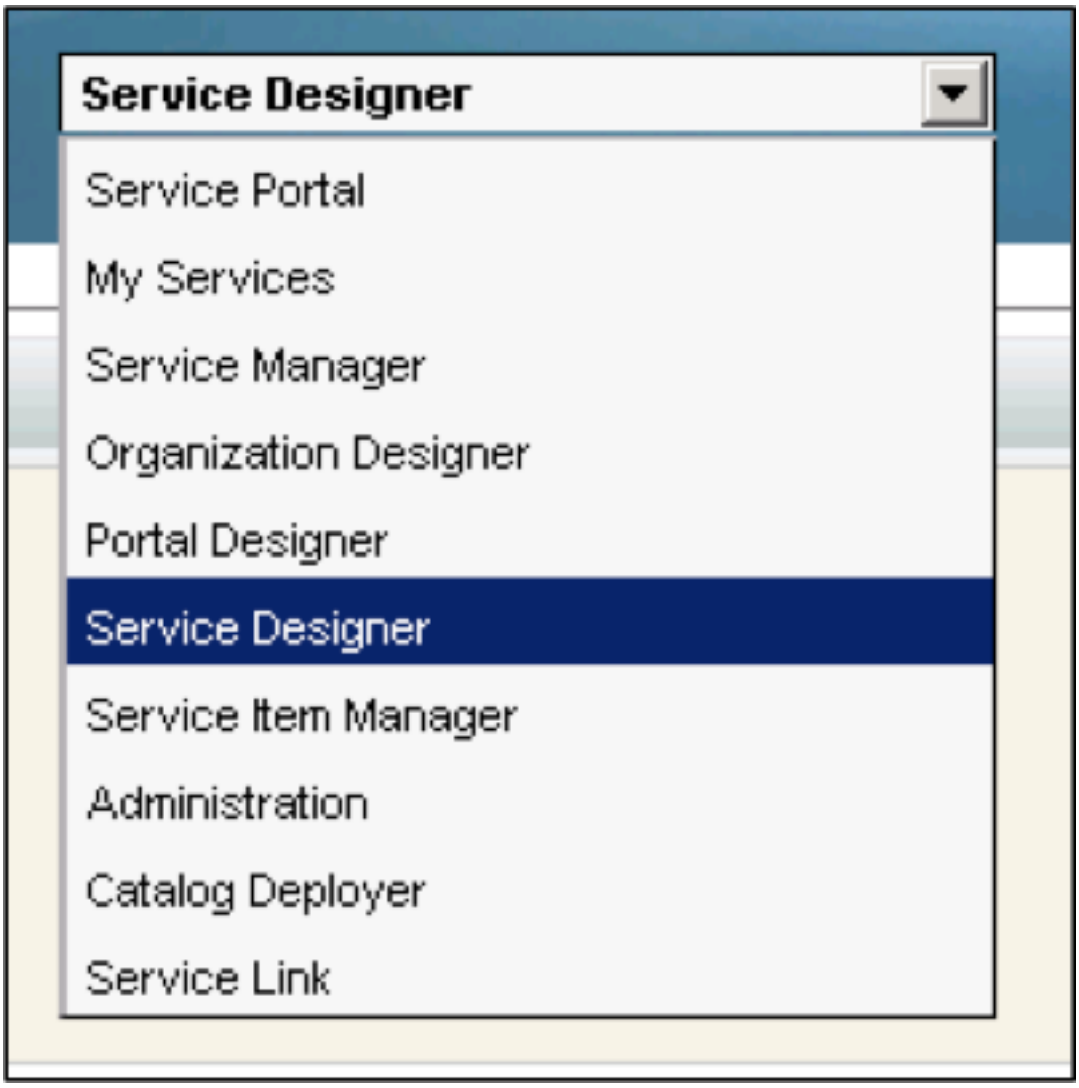
Diseñador IAC y del servicio

El procedimiento para crear los nuevos botones en mis VDC requiere el uso del diseñador del servicio.

Agregar un nuevo botón

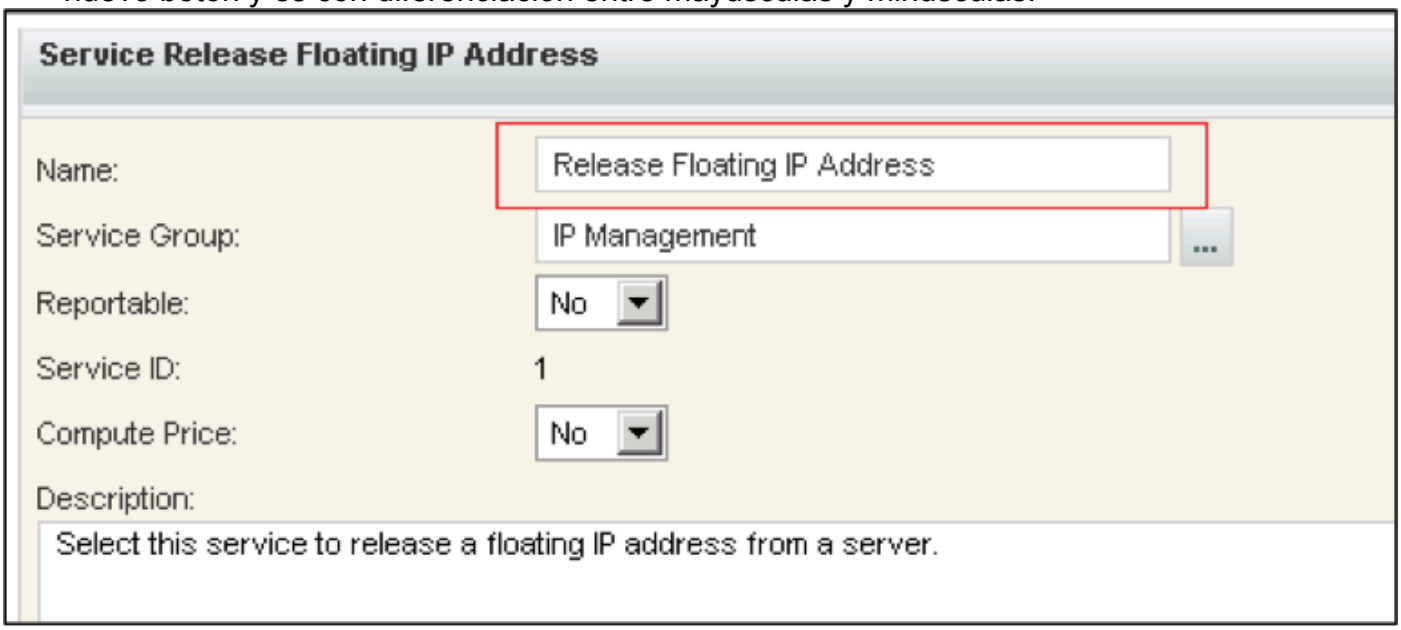
El siguiente ejemplo ilustra los pasos para agregar un nuevo botón “para liberar la dirección IP flotante.”

Paso 1: Entre al diseñador del servicio como se muestra abajo en el diagrama 1.



Paso 2: Cree un servicio y salve su nombre para uso futuro.

- En el diagrama 2 mostrado abajo, el nombre “dirección IP flotante de la versión” se asigna al nuevo botón y es con diferenciación entre mayúsculas y minúsculas.



Paso 3: Localice el directorio del despliegue IAC y abra el archivo “my-vdcs.html” para editar.

- \ RequestCenterServer \ implementaciones \ RequestCenter.war \ aduana \ IAC \ portlets \ my-vdcs.html

Paso 4: Agregue la línea nueva en el extremo de la sección de ServiceIds y dentro del bloque condicional apropiado que determina la accesibilidad del servicio a través de diversos rol del usuario como se muestra abajo en el diagrama 3.

NOTA: El nombre del servicio es con diferenciación entre mayúsculas y minúsculas, y utiliza el "serviceId" como segundo argumento.

```
151        //=====================================================================
152        //  Get the service IDs for all the action buttons to pass to the popover
153        //=====================================================================
154        if(personRole != "Cloud Provider Business Administrator" && personRole != "Tenant Business Administrator")
155        {
156            if(personRole == "Cloud Provider Technical Administrator" || personRole == "Organization Technical Administrator" || personRole == "Tenant Technical Administrator")
157            {
158                var VDCModifySize = getServiceProperties("Modify VDC Size", "serviceId");
159                var VDCAddNetwork = getServiceProperties("Add Network to VDC", "serviceId");
160                var VDCModifyNetwork = getServiceProperties("Modify VDC Network Properties", "serviceId");
161                var VDCRemoveNetwork = getServiceProperties("Remove Network from VDC", "serviceId");
162                var VDCDecommission = getServiceProperties("Decommission Virtual Data Center", "serviceId");
163                var VDCManageVDCAccess = getServiceProperties("Manage Access to VDC", "serviceId");
164                var createVDCFirewallRule = getServiceProperties("Create VDC Firewall Rule", "serviceId");
165                var deleteVDCFirewallRule = getServiceProperties("Delete VDC Firewall Rule", "serviceId");
166            }
167            if(personRole != "Virtual Server Owner")
168            {
169                var VDCAddPhysServ = getServiceProperties("Order a Physical Server", "serviceId");
170            }
171            var VDCAddVMTemplate = getServiceProperties("Order a Virtual Machine From Template", "serviceId");
172            var VDCAddVMOS = getServiceProperties("Order a Virtual Machine and Install an OS", "serviceId");
173            var createServerGroup = getServiceProperties("Create Server Group", "serviceId");
174            var deleteServerGroup = getServiceProperties("Delete Server Group", "serviceId");
175            var serviceIds = VDCModifySize+","+VDCDecommission+","+VDCManageVDCAccess+","+VDCAddNetwork+","+VDCRemoveNetwork+","+VDCModifyNetwork+","+VDCAddVMTemplate+","+VDCAddVMOS+
               ","+VDCAddPhysServ+","+createVDCFirewallRule+","+deleteVDCFirewallRule+","+createServerGroup+","+deleteServerGroup;
176            console.log(serviceIds);
177        }
```

Paso 5: Abra el archivo "IACServersPopover.js" para editar eso se localiza en:

- "\ src \ aparato \ iac \ IACVDCPopover.js"
- Inserte el código siguiente mostrado abajo en el diagrama 4 con las condiciones apropiadas para mostrar la acción para cada papel específico y el tipo PE, antes de llamar el código de los "generateActionButtons" (situado antes de la línea no.: 198).
- En los serviceNames, avance el nombre del servicio, en este ejemplo "dirección IP flotante de la versión."
- En los serviceids avance el valor del índice apropiado del arsenal de los serviceIds, en este ejemplo que es index13, la búsqueda más reciente del servicio de los "getServiceProperties" en my-vdcs.html.
- El valor del "generalCheck" será "verdad" para los papeles CPTA, OTA y TTA.

- En la muestra antedicha la nueva “acción de la dirección IP flotante de la versión” no se muestra para los VDC de la “VAINA de OpenStack” (tipo PE), no obstante el control se puede modificar según las necesidades.

```
200            if( generalCheck && PEType !== "OpenStack Cloud Manager" ) {
201                serviceNames.push("Release Floating IP Address");
202                ids.push(serviceIds[13]);
203                cssClassNames.push("releaseIP");
204            }
```

Paso 6: Se crea la estructura, se ejecuta y despliega para verificar el nuevo botón.

Información Relacionada

- Cisco Intelligent Automation for Cloud