net>scaler

NetScaler ADC CPX 12.1

Machine translated content

Disclaimer

La versión oficial de este contenido está en inglés. Para mayor comodidad, parte del contenido de la documentación de Cloud Software Group solo tiene traducción automática. Cloud Software Group no puede controlar el contenido con traducción automática, que puede contener errores, imprecisiones o un lenguaje inadecuado. No se ofrece ninguna garantía, ni implícita ni explícita, en cuanto a la exactitud, la fiabilidad, la idoneidad o la precisión de las traducciones realizadas del original en inglés a cualquier otro idioma, o que su producto o servicio de Cloud Software Group se ajusten a cualquier contenido con traducción automática, y cualquier garantía provista bajo el contrato de licencia del usuario final o las condiciones de servicio, o cualquier otro contrato con Cloud Software Group, de que el producto o el servicio se ajusten a la documentación no se aplicará en cuanto dicha documentación se ha traducido automáticamente. Cloud Software Group no se hace responsable de los daños o los problemas que puedan surgir del uso del contenido traducido automáticamente.

Contents

Acerca de NetScaler CPX	2
Arquitectura y flujo de tráfico	2
Licencias de NetScaler CPX	6
Implementación de una instancia de NetScaler CPX en Docker	14
Agregar instancias de NetScaler CPX a Citrix ADM	21
Configuración de NetScaler CPX	23
Configuración de NetScaler CPX mediante un archivo de configuración	28
Configuración de controladores de registro de Docker	29
Compatibilidad con redirección dinámica en NetScaler CPX	30
Actualización de una instancia de NetScaler CPX	33
Uso de servidores virtuales comodín en la instancia de NetScaler CPX	35
Implementar NetScaler CPX como proxy para permitir el flujo de tráfico de este a oeste	36
Implementación de NetScaler CPX en una red de host único	40
Implementación de NetScaler CPX en una red de varios hosts	41
Implementar NetScaler CPX con acceso directo a la red	46
Configurar NetScaler CPX en Kubernetes mediante ConfigMaps	47
Implementar el proxy de NetScaler CPX en Google Compute Engine	50
Solución de problemas de NetScaler CPX	70

Acerca de NetScaler CPX

November 23, 2023

NetScaler CPX es un controlador de entrega de aplicaciones basado en contenedores que se puede aprovisionar en un host Docker. NetScaler CPX permite a los clientes aprovechar las capacidades del motor de Docker y utilizar las funciones de equilibrio de carga y administración de tráfico de NetScaler para aplicaciones basadas en contenedores. Puede implementar una o más instancias de NetScaler CPX como instancias independientes en un host de Docker.

Una instancia de NetScaler CPX proporciona un rendimiento de hasta 1 Gbps.

En este documento se asume que está familiarizado con Docker y su funcionamiento. Para obtener información sobre Docker, consulte la documentación de Docker en https://docs.docker.com.

Funciones disponibles

NetScaler CPX admite las siguientes funciones:

- Disponibilidad de aplicaciones
 - Equilibrio de carga L4 y conmutación de contenido L7
 - Descarga de SSL
 - Traducción de protocolos IPv6
- Seguridad de las aplicaciones
 - Reescritura y respuesta de L7
- Capacidad de administración sencilla
 - Registro web
 - AppFlow

Arquitectura y flujo de tráfico

November 23, 2023

Cuando aprovisiona una instancia de NetScaler CPX en un host de Docker, el motor de Docker crea una interfaz virtual, eth0, en la instancia CPX. Esta interfaz eth0 está conectada directamente a una interfaz virtual (veth*) en el puente docker0. El motor de Docker también asigna una dirección IP a la instancia de NetScaler CPX en la red 172.17.0.0/16.

La puerta de enlace predeterminada para la instancia CPX es la dirección IP del puente docker0, lo que significa que cualquier comunicación con la instancia de NetScaler CPX se realiza a través de la red Docker. Todo el tráfico entrante recibido desde el puente docker0 lo recibe la interfaz eth0 en la instancia NetScaler CPX y el motor de paquetes NetScaler CPX lo procesa.

Esta ilustración ilustra la arquitectura de una instancia de NetScaler CPX en un host Docker.



Cómo funciona una dirección IP única en NetScaler CPX

Un dispositivo NetScaler MPX o VPX normal requiere al menos tres direcciones IP para funcionar:

- Dirección IP de administración denominada dirección IP de NetScaler (NSIP)
- Dirección IP de subred (SNIP) para comunicarse con el conjunto de servidores
- Direcciones IP (VIP) del servidor virtual para aceptar solicitudes de clientes

Una instancia de NetScaler CPX funciona con una sola dirección IP que se utiliza tanto para la administración como para el tráfico de datos.

Durante el aprovisionamiento, el motor de Docker solo asigna una dirección IP privada (dirección IP única) a una instancia de NetScaler CPX. Las tres funciones IP de una instancia de NetScaler se multiplexan en una dirección IP. Esta única dirección IP utiliza diferentes números de puerto para funcionar como NSIP, SNIP y VIP (s).

La siguiente imagen ilustra cómo se usa una sola dirección IP para realizar las funciones de NSIP, SNIP y VIP (s).



Flujo de tráfico para solicitudes que se originan en la instancia de NetScaler CPX

Docker configura implícitamente tablas IP y una regla NAT para dirigir el tráfico que se origina en la instancia de NetScaler CPX a la dirección IP de docker0.

Esta ilustración ilustra cómo una solicitud ping que se origina en una instancia de NetScaler CPX llega al destino.



En este ejemplo, el motor de paquetes envía la solicitud ping en la interfaz eth0 con la dirección IP de origen como la dirección IP de NetScaler CPX (172.17.0.4). A continuación, el host de Docker realiza la traducción de direcciones de red (NAT) para agregar la dirección IP del host (192.68.x.x) como dirección IP de origen y envía la solicitud al destino (216.58.x.x). La respuesta de la dirección IP de destino sigue la misma ruta a la inversa. El host Docker realiza NAT en la respuesta y reenvía la respuesta a la instancia de NetScaler CPX en la interfaz eth0.

Flujo de tráfico para solicitudes que se originan en la red externa

Para habilitar la comunicación externa, al aprovisionar NetScaler CPX, debe establecer parámetros de modo que Docker exponga ciertos puertos, como 80, 22 y cualquier otro puerto que quiera. Si no ha configurado ningún puerto para que esté expuesto durante el aprovisionamiento, debe configurar reglas NAT en el host de Docker para que estos puertos estén disponibles.

La solicitud del cliente que se origina en Internet es recibida por el host de Docker, que luego realiza la traducción de direcciones de puerto (PAT) para asignar la dirección IP pública y el puerto a la dirección IP y el puerto únicos de la instancia de NetScaler CPX, y reenvía el tráfico a la instancia.

Esta ilustración muestra cómo el host Docker realiza la traducción de direcciones de puerto para dirigir el tráfico a la dirección IP única y al puerto NetScaler CPX.



En este ejemplo, la dirección IP del host de Docker es 192.68.x.x y la dirección IP única de la instancia de NetScaler CPX es 172.17.0.4. El puerto SSH 22 de la instancia de NetScaler CPX se asigna al puerto 1100 en el host Docker. La solicitud SSH del cliente se recibe en la dirección IP 192.68.x.x en el puerto 1100. El host de Docker realiza la traducción de direcciones de puerto para asignar esta dirección y puerto a la dirección IP única 172.17.0.4 en el puerto 22 y reenvía la solicitud del cliente.

Licencias de NetScaler CPX

July 22, 2024

NetScaler CPX es un controlador de entrega de aplicaciones basado en contenedores que se puede aprovisionar en un host Docker para equilibrar la carga de las aplicaciones basadas en microservicios. Necesita CPX con licencia para obtener un mejor rendimiento de la entrega de aplicaciones. NetScaler CPX admite las licencias de grupos. Citrix ADM puede actuar como su servidor de licencias para licenciar sus instancias de NetScaler CPX.

Citrix ADM también está disponible en las instalaciones y en la nube. Puede usar Citrix ADM para administrar licencias de capacidad agrupadas para todos los factores de forma de Citrix ADC.

Para obtener información sobre Citrix ADM local, consulte Citrix ADM local. Para obtener información sobre el servicio Citrix ADM, consulte el servicio Citrix ADM.

Tipos de licencias de NetScaler CPX

NetScaler CPX admite licencias de ancho de banda y grupos de CPU virtual (núcleo) para implementaciones locales y basadas en la nube.

Grupo de anchode banda: las licencias de NetScaler CPX se pueden asignar en función del consumo de ancho de banda de las instancias. Puede utilizar las licencias agrupadas para maximizar la utilización del ancho de banda al garantizar la asignación de ancho de banda necesaria a una instancia y no más de lo que se requiere. Para obtener más información sobre el grupo de ancho de banda, consulte Licencias basadas en el ancho

Grupo de vCPU: en las licencias virtuales basadas en el uso de CPU, la licencia especifica el número de CPU a las que tiene derecho una instancia de NetScaler CPX en particular. Por lo tanto, NetScaler CPX puede extraer licencias solo para la cantidad de CPU virtuales del servidor de licencias. NetScaler CPX desprotege las licencias en función del número de CPU que se ejecutan en el sistema. Para obtener más información sobre el grupo de vCPU, consulte Licencias de CPU virtuales de Citrix ADC.

Producto	Ancho de banda máximo	Ancho de banda mínimo	Instancias mínimas	Instancias máximas	Unidad mínima de ancho de banda
NetScaler CPX	40000 Nota :	20 Mbps	1	16	10 Mbps
	Depende de la				
	la CPU, la				
	generación,				
	etc.				

Capacidad agrupada admitida para instancias NetScaler CPX

Nota: Citrix está trabajando actualmente en un modelo de licencias basado en el consumo de NetScaler CPX o basado en el pago a medida que crece para ofertas públicas basadas en la nube. Una vez listo, estará disponible en el mercado de la nube pública para su consumo.

¿Cómo funcionan las licencias de NetScaler CPX?

Capacidad agrupada de NetScaler CPX: un grupo de licencias común desde el que la instancia de NetScaler CPX puede extraer una licencia de instancia y solo el ancho de banda que necesite. Cuando la instancia ya no requiere estos recursos, los vuelve a registrar en el grupo común, lo que hace que los recursos estén disponibles para otras instancias que necesitan estas licencias.

Licencias de registro y salida de NetScaler CPX: Citrix ADM asigna licencias de instancias NetScaler CPX a pedido. Una instancia de NetScaler CPX puede retirar la licencia de Citrix ADM cuando se aprovisiona una instancia de NetScaler CPX y volver a registrar su licencia en Citrix ADM cuando se destruye una instancia.

Comportamiento de NetScaler CPX: una sola instancia de NetScaler CPX que desprotege un rendimiento de hasta 1 Gbps, solo se desprotege del grupo de instancias y no del grupo de licencias de ancho de banda. NetScaler CPX funciona de esta manera hasta 1 Gbps de utilización de ancho de banda. Por ejemplo, si una instancia CPX consume un ancho de banda de 200 Mbps, utiliza el grupo de instancias de licencia, en lugar del conjunto de ancho de banda. Sin embargo, si una instancia de NetScaler CPX consume 1200 Mbps de rendimiento, los primeros 1000 Mbps se utilizan del grupo de instancias y los 200 Mbps restantes se consumen del grupo de ancho de banda.

NetScaler CPX Express

NetScaler CPX Express es una edición de software gratuita para implementaciones locales y en la nube. Cuando descarga la instancia de NetScaler CPX del repositorio de Quay, esta es la capacidad predeterminada disponible para los POC que no requieren un archivo de licencia y viene con estas funciones:

- Ancho de banda 20 Mbps
- Máximo 250 sesiones SSL
- Rendimiento SSL de 20 Mbps

Debe obtener una licencia de su instancia de NetScaler CPX para realizar la actualización y lograr un mejor rendimiento y implementaciones de producción.

Modelos de licencia de NetScaler CPX

Citrix ofrece una gama de modelos de licencias de productos para que NetScaler CPX cumpla con los requisitos de su organización. Puede seleccionar opciones como vCPU o ancho de banda y local o en la nube.

Según sus requisitos, puede elegir cualquiera de los siguientes modelos:

- Licencias basadas en ancho de banda para NetScaler CPX desde el servicio ADM
- Licencias basadas en vCPU para NetScaler CPX desde el servicio ADM

- Licencias basadas en ancho de banda para NetScaler CPX desde ADM local
- Licencias basadas en vCPU para NetScaler CPX desde ADM local

Aprovisionar licencias basadas en ancho de banda y basadas en CPU virtuales desde el servicio Citrix ADM para NetScaler CPX

Realice los siguientes pasos para aprovisionar licencias basadas en ancho de banda y licencias basadas en CPU virtuales para NetScaler CPX desde el servicio Citrix ADM.

1. Configure Citrix ADM.

Asegúrese de que la configuración del servicio Citrix ADM funcione con el agente ADM de NetScaler. Debe tener un servicio Citrix ADM y una cuenta de agente NetScaler ADM para que las licencias de NetScaler CPX funcionen. Para obtener información sobre la configuración de Citrix ADM Service y el agente NetScaler ADM, consulte Citrix ADM Service.

Nota: En este procedimiento, se utiliza una configuración de agente NetScaler ADM de hipervisor (local). En la siguiente imagen, 10.106.102.199 se muestra el agente local que se utiliza para obtener licencias de NetScaler CPX.

≡	citrix A	pplicati	ion Delivery Management	▲ ② シトリックス・システ > OrgiD: htt54cd3ba2
Q Sear	rch in Menu	<:	Networks > Agents	
Applica	ations	>	Agents (2)	Set Up Agent Settings Generate Activation Code 2 🛃 ? 🖸
Networ	ks	\sim		
In	frastructure Analytics		View Details Delete Reboot Rediscover Attach Site View Fingerprint Provision Select Action 🗸	÷2+
In	istances	\sim	Q Click here to search or you can enter Key : Value format	0
	Citrix ADC	لما	IP ADDRESS O HOST NAME O VERSION O STATE A PLATFORM O CPU USAGE (%) O	DISK USAGE (%) MEMORY USAGE (%) COUNTRY REGION CIT
	Citrix SD-WAN Preview		10.106.102.199 akshay-199 13.0-82.22 • Up KVM 1	22 13
	HAProxy Preview		10.98.161.89 akshay-adm-agent 13.0-77.26 () Op Kubernetes	47 0
In	stance Advisory	>	Total 2	25 Per Page ∨ Page 1 of 1 <
In	stance Groups			
A	utoScale Groups			
☆ A	gents			
Li	icenses	>		
E	vents	>		
s	SL Dashboard	>		
с	onfiguration Jobs	>		
с	onfiguration Audit	>		
S	ites			
A	ppEnvironments			
IF	PAM			
D	omain Names			
N	letwork Functions	>		
N	letwork Reporting			
Analyti	cs	>		
Orches	tration	>		
Accoun	nt	>		
Setting	8	>		
Help Co	enter			

2. Agregue el grupo de licencias de instancias de Citrix ADC al servicio Citrix ADM.

Se supone que tiene un conjunto de licencias de ancho de banda disponibles para el servicio ADM. Para obtener información sobre cómo cargar un archivo de licencia en Citrix ADM, consulte Configurar la capacidad agrupada. En la siguiente imagen, CNS_INST_200CC_Retail.lic se usa como el ancho de banda y el grupo de licencias de instancias.

≡ citrix A	pplicati	on Delivery Management				4 1 -	シトリックス・システ ∨ OrgID: int54cd3ba2
Q Search in Menu	<:	Networks > License Settings					(?)
Applications	>	License Files					
Networks	\sim	The following license files are present	on this server. Select Add New License to	uninad more licenses. To delete a license, sel	ect the license and click Delete		
Infrastructure Analytics		To manually Download licenses from C	itrix licensing portal please visit http://ww	ww.mycitrix.com and use the Host ID: fedf8321	39ee		
Instances	>	Add License File Apply License	es Delete Download				
Instance Groups	>	O Click here to search or you can ent	r Key : Value format				0
AutoScale Groups		NAME					\$176
Agents		CNS INST 5CCS Ret	illic		Mon Sep 07 2020 8:22 PM		1.03 KB
☆ Licenses	>	CNS_PBW10MB_50GE	L_RetailS.lic		Thu Aug 20 2020 1:52 PM		1.22 KB
Events	>	CNS_CP1000_100SSE	RVER_RetailS.lic		Mon Sep 07 2020 8:22 PM		1.04 KB
SSL Dashboard	>	CNS_PBW10MB_10GB	_RetailS.lic		Mon Sep 07 2020 8:22 PM		1.24 KB
Configuration Jobs	>	CNS_1VCPUP_100SSE	RVER_RetailS.lic		Thu Aug 20 2020 2:20 PM		1.13 KB
Configuration Audit	>	Total 5				25 Per F	Page V Page 1 of 1 4 >
AppEnvironments							
IPAM		License Expiry Information					
Domain Names		FEATURE	 cour 	NT	DAYS TO EXPIRY		
Network Functions	>	No nems					
Network Reporting		Notification Settings					1
Analytics	>	Email Profile	Slack Profile	PagerDuty Profile	ServiceNow Profile	Alert Threshold	Dave To Expiry
Orchestration	>	Not configured	Not configured	Not configured	Not configured	90%	30
Settings	>						
Help Center	,						

3. Implemente la instancia de NetScaler CPX en el clúster de Kubernetes. Asegúrese de que las siguientes variables de entorno se agreguen al archivo YAML de NetScaler CPX para licenciar la instancia de NetScaler CPX.

Para las licencias basadas en ancho de banda del servicio Citrix ADM, especifique las siguientes variables de entorno en el archivo YAML:

• nombre: "LS_IP"

valor: "10.105.158.166"//IP del agente ADM como se mencionó en el paso 1

- nombre: "LS_PORT" valor: "27000"// Puerto que escucha el servidor de licencias ADM
- nombre: "BANDWIDTH" valor: "3000"//Capacidad en Mbps que quiere asignar a CPX

Para las licencias basadas en CPU virtuales del servicio Citrix ADM, especifique las siguientes variables de entorno en el archivo YAML:

• nombre: "LS_IP"

valor: "10.102.216.173"//IP del agente ADM como se mencionó en el paso 1

- nombre: "LS_PORT" valor: "27000"// Puerto que escucha el servidor de licencias ADM
 nombre: "CPX_CORES"
 - valor: "4"// Cantidad de núcleos que quiere asignar
- 4. Descargue el cpx-bandwidth-license-adm-service.yaml archivo mediante el siguiente comando:

1 kubectl create namespace bandwidth

- 2 wget https://raw.githubusercontent.com/citrix/cloud-nativegetting-started/master/cpx-licensing/manifest/cpx-bandwidthlicense-adm-service.yaml
- 5. Implemente el YAML modificado en el clúster de Kubernetes con el siguiente comando:

```
kubectl create -f cpx-bandwidth-license-adm-service.yaml -n
bandwidth
```

6. Inicie sesión en NetScaler CPX para verificar la información de instancias mediante el siguiente comando:

```
1 kubectl exec -it 'cpx-pod-ip-name' bash -n bandwidth
```

7. Para ver la información de licencias de la instancia de NetScaler CPX determinada, ejecute los siguientes comandos:

```
1 cli_script.sh "show licenseserver"
2 cli_script.sh "show capacity"
```

Puede realizar un seguimiento del ancho de banda asignado y la capacidad de vCPU en el portal de servicios ADM.

Aprovisionar licencias basadas en ancho de banda y licencias basadas en vCPU para NetScaler CPX desde Citrix ADM local

Realice los siguientes pasos para aprovisionar en función del ancho de banda y de la CPU virtual en NetScaler CPX desde Citrix ADM local.

1. Configure Citrix ADM.

Asegúrese de que la configuración local de ADM esté lista. Asegúrese de que Citrix ADM local con o sin la implementación del agente ADM para las licencias de NetScaler CPX esté funcionando.

Para obtener información sobre cómo configurar Citrix ADM local y el agente NetScaler ADM, consulte Citrix ADM Service.

Nota: En este ejemplo, se utiliza un agente ADM integrado con ADM local. En la siguiente imagen, puede ver que no hay ningún agente implementado.

× Citrix Applica	tior	n Delivery Management Apr 26 2021 06:49:10 GMT	<u></u> ,	sroot 🗸
Q Search in Menu		Networks > Agents		
★ Favorites	>	Agents 0	C	
III Menu	\sim	View Details Delete Reboot Attach Site View Fingerprint No action V		¢
Applications	>			
Networks	\sim	Q Click here to search or you can enter Key ' Value format		0
Infrastructure Analytics		IP ADDRESS COUNTRY COUNTRY CITY	≎ SIT	÷
Instances	>	No items		
Instance Groups				
습 Agents				
Licenses	>			
Events	>			
SSL Dashboard	>			
Configuration Jobs	>			
Configuration Audit	>			
Sites				
IPAM				
Domain Names				
Network Functions	>			
Network Reporting	>			
Analytics	>			
Orchestration	>			
System	>			
Help Center				

2. Agregue el grupo de licencias de instancias de Citrix ADC a ADM local.

Se supone que tiene un conjunto de licencias de ancho de banda disponible para ADM local. Para obtener información sobre cómo cargar un archivo de licencia en Citrix ADM, consulte Licencias. En la siguiente imagen, CNS_INST_200CC_Retail.lic se usa como el ancho de banda y el grupo de licencias de instancias.

× Citrix Applica	ion Delivery Management		Apr 26 2021 06:50:11 GMT 🥂 nsroot 🗸
Q Search in Menu	Networks > License Settings		0
★ Favorites	License Server Port Settings		/
Menu Applications Networks Infrastructure Analytics	Proxy Server Port 0 The Proxy Server Port used by Citrix ADC Instances to access the Citrix licensing portal for license allocation	License Server Port 27000 The License Server Port used by Citrix ADC instances to communicate with the license server	Vendor Daemon Port 7279 The Daemon Port used by Citrix ADC instances to communicate with the license server
Instances Instance Groups Agents	License Files The following license files are present on this server. Select Add New License to up with http://www. Amaulay Download licenses from Citrix licensing portal please visit http://www.	bload more licenses. To delete a license, select the license and click Delete. wmydthccom and use the Host ID. 02208e8fa59	
Events SSL Dashboard	Add License File Apply Licenses Delete Download Q Click here to search or you can enter Key : Value format		0
Configuration Jobs Configuration Audit Sites IPAM Domain Names Network Functions Network Reporting Analytics Orchestration	NAME CIS_ERVIDARE_2008_Paulic CIS_ERVIDARE_2008_Paulic CIS_ERVIDARE_2008_Paulic CIS_PRVIDARE_2008_Paulic CIS_PRVIDARE_2008_Paulic CIS_PRVIDARE_2008_Paulic CIS_PRVIDARE_2008_RetailSile CIS_UNST_2002CS_RetailIc Total 6	LAST MODIFIED Mon Apr 26 2021 2217 PM Mon Apr 26 2021 1217 PM	52E : 124 K8 : 127 K8 : 123 K8 : 124 K8 :
System Help Center		COUNT COUNT 250,000 200,00000000	DAVS TO EXPIREY : 219 219 229 229 229 229 229 229 Per Page V Page 1 of 1 V V

En la siguiente imagen, CP1000 se usa como el grupo de licencias de vCPU.

× Citrix Appl	icatio	n Delivery Management		Apr 26 2021 06:52:17 GMT
Q Search in Menu		Networks > License Settings		Ø
★ Favorites	>	License Server Port Settings		/
Henu Applications	~ > ~	Proxy Server Port 0 The Proxy Server Port used by Clinix ADC instances to access the Clinix licensing portal for license allocation	License Server Port 27000 The License Server Port used by Citrix ADC instances to communicate with the license server	Vendor Daemon Port 7279 The Daemon Port used by Clirix ADC Instances to communicate with the license server
Infrastructure Analytics Instances Instance Groups Agents	>	License Files The following license files are present on this server. Select Add New License to up To manually Download licenses from Citrix licensing portal please vidit http://www	load more licenses. To delete a license, select the license and click Delete. myclifuc.com and use the Host ID: 02a208e5fa59	
Licenses Events SSL Dashboard	 > > 	Add License File Apply Licenses Delete Download Q Click here to search or you can enter Key : Value format Click here to search or you can enter Key : Value format Click here to search or you can enter Key : Value format Click here to search or you can enter Key : Value format Click here to search or you can enter Key : Value format Click here to search or you can enter Key : Value format Click here to search or you can enter Key : Value format Click here to search or you can enter Key : Value format Click here to search or you can enter Key : Value format Click here to search or you can enter Key : Value format Click here to search or you can enter Key : Value format Click here to search or you can enter Key : Value format Click here to search or you can enter Key : Value format Click here to search or you can enter Key : Value format Click here to search or you can enter Key : Value format Click here to search or you can enter Key : Value format Click here to search or you can enter Key : Value format Click here to search or you can enter Key : Value format Click here to search or you can enter Key : Value format Click here to search or you can enter Key : Value format Click here to search or you can enter Key : Value format Click here to search or you can enter Key : Value format Click here to search or you can enter Key : Value format Click here to search or you can enter Key : Value format Click here to search or you can enter Key : Value format Click here to searc		0
Configuration Jobs Configuration Audit	>	NAME CNS_EBW10MB_200GB_Eval.lic	C LAST MODIFIED	 SIZE 1.24 KB
Sites		CNS_SBW10MB_500GB_RetailLlic	Mon Apr 26 2021 12:29 PM Mon Apr 26 2021 12:29 PM	1.17 KB
Domain Names Network Functions	>	CNS_CP1000_100SSERVER_RetailS.lic	Mon Apr 26 2021 12:29 PM	1.04 KB
Network Reporting Analytics	> >	CNS_P8W10MB_50GB_RetailS.lic	Mon Apr 26 2021 12:29 PM	1.13 KB
Orchestration System	> >	Total 7	Mon Apr 26 2021 12:29 PM	1.04 KB 25 Per Page ✓ Page 1 of 1 < ►
Help Center		License Expiry Information		
		FEATURE Standard Bandwidth	COUNT 0	DAYS TO EXPIRY ¢
		Instance Platinum Bandwidth	200	219 219
		Enterprise Bandwidth Total 4	200,000	219 25 Per Page > Page 1 of 1 <

3. Implemente la instancia de NetScaler CPX en el clúster de Kubernetes. Asegúrese de que las siguientes variables de entorno se agreguen al archivo YAML de NetScaler CPX para licenciar la instancia de NetScaler CPX.

Para las licencias basadas en ancho de banda de Citrix ADM local, especifique las siguientes variables de entorno en el archivo YAML:

nombre: "LS_IP"

valor: "10.105.158.144"// IP de instancia local de ADM, si ha implementado el agente ADM, esta es la dirección IP de su agente como se describe en el paso 1

- nombre: "LS_PORT" valor: "27000"// Puerto que escucha el servidor de licencias ADM
 nombre: "BANDWIDTH"
- valor: "3000"//Capacidad en Mbps que quiere asignar a CPX

Para las licencias basadas en CPU virtuales de Citrix ADM local, especifique las siguientes variables de entorno en el archivo YAML:

- nombre: "LS_IP"
 valor: "10.105.158.144"// IP de instancia local de ADM, si tiene implementación de agente ADM, esta será la IP de su agente como se describe en el paso 1
- nombre: "LS_PORT" valor: "27000"// Puerto que escucha el servidor de licencias ADM
- nombre: "CPX_CORES" valor: "4"// La cantidad de núcleos que quiere asignar

4. Descargue el cpx-bandwidth-license-adm-onprem.yaml archivo mediante el siguiente comando:

```
1 kubectl create namespace bandwidth
2 wget https://raw.githubusercontent.com/citrix/cloud-native-
getting-started/master/cpx-licensing/manifest/cpx-bandwidth-
license-adm-onprem.yaml
```

5. Implemente el YAML modificado en el clúster de Kubernetes con el siguiente comando:

```
1 kubectl create -f cpx-bandwidth-license-adm-onprem.yaml -n
bandwidth
```

6. Inicie sesión en NetScaler CPX para verificar la información de instancias mediante el siguiente comando:

1 kubectl exec -it <cpx-pod-ip-name> bash -n bandwidth

7. Para ver la información de licencias de la instancia de NetScaler CPX, ejecute los siguientes comandos:

```
1 cli_script.sh "show licenseserver"
2 cli_script.sh "show capacity"
```

Puede realizar un seguimiento del ancho de banda asignado y la capacidad de vCPU en el portal local de ADM.

Comandos para limpiar las implementaciones

Puede usar los siguientes comandos para limpiar las distintas implementaciones de YAML:

```
1 kubectl delete -f cpx-bandwidth-license-adm-service.yaml -n bandwidth
2 kubectl delete -f cpx-core-license-adm-service.yaml -n core
3 kubectl delete -f cpx-bandwidth-license-adm-onprem.yaml -n bandwidth
4 kubectl delete -f cpx-core-license-adm-onprem.yaml -n core
5 kubectl delete namespace bandwidth
6 kubectl delete namespace core
```

Implementación de una instancia de NetScaler CPX en Docker

November 23, 2023

Las instancias de NetScaler CPX están disponibles como un archivo de imagen de Docker en el registro de contenedores de Quay. Para implementar una instancia, descargue la imagen de NetScaler CPX

del registro de contenedores Quay y, a continuación, implemente la instancia mediante el comando docker run o la herramienta de redacción de Docker.

Requisitos previos

Asegúrese de que:

- El sistema host de Docker tiene al menos:
 - 1 CPU
 - 2 GB de RAM

Nota:Para obtener un mejor rendimiento de NetScaler CPX, puede definir la cantidad de motores de procesamiento que quiere que inicie la instancia de NetScaler CPX. Para cada motor de procesamiento adicional que agregue, asegúrese de que el host de Docker contenga la cantidad equivalente de CPU virtuales y la cantidad de memoria en GB. Por ejemplo, si quiere agregar 4 motores de procesamiento, el host de Docker debe contener 4 vCPU y 4 GB de memoria.

- El sistema host Docker ejecuta Linux Ubuntu versión 14.04 o posterior.
- La versión 1.12 de Docker está instalada en el sistema host. Para obtener información sobre la instalación de Docker en Linux, consulte ladocumentación de Docker.
- El host de Docker tiene conectividad a Internet.

Nota: NetScaler CPX tiene problemas cuando se ejecuta en ubuntu versión 16.04.5, kernel versión 4.4.0-131-generic. Por lo tanto, no se recomienda ejecutar NetScaler CPX en ubuntu versión 16.04.5 kernel versión 4.4.0-131-generic.

Descarga de la imagen de NetScaler CPX desde Quay

Puede descargar la imagen de NetScaler CPX desde el registro de contenedores de Quay mediante el comando docker pull e implementarla en su entorno. Use el siguiente comando para descargar la imagen de NetScaler CPX del registro de contenedores de Quay:

1 docker pull quay.io/citrix/citrix-k8s-cpx-ingress:tag

En este comando, la etiqueta especifica la imagen de NetScaler CPX de Citrix.

Por ejemplo, si quiere descargar la versión 12.1-51.16, utilice el siguiente comando:

1 docker pull quay.io/citrix/citrix-k8s-cpx-ingress:12.1-51.16

Use el siguiente comando para comprobar si la imagen de NetScaler CPX está instalada en imágenes de docker:

```
1root@ubuntu:~# docker images | grep 'citrix-k8s-cpx-ingress'2quay.io/citrix/citrix-k8s-cpx-ingress12.1-51.16952a04e731012 months ago469 MB
```

Nota: Se recomienda utilizar la imagen más reciente de NetScaler CPX del registro de contenedores Quay para aprovechar las funciones más recientes de NetScaler CPX.

Implementación de la instancia de NetScaler CPX mediante el comando docker run

En el host, puede instalar una instancia de NetScaler CPX en el contenedor de Docker mediante la imagen de Docker de NetScaler CPX que cargó en el host. Mediante el comando docker run, instale la instancia de NetScaler CPX con la configuración predeterminada de NetScaler CPX.

Importante:

Si ha descargado NetScaler CPX Express de https://www.citrix.com/products/netscaler-adc/cpxexpress.html, asegúrese de leer y comprender el Contrato de licencia de usuario final (EULA) disponible en:https://www.citrix.com/products/netscaler-adc/cpx-express.htmly acepte el EULA al implementar la instancia de NetScaler CPX.

Instale la instancia de NetScaler CPX en el contenedor de Docker mediante el siguiente comando *docker run*:

```
1 docker run -dt --privileged=true --net=host -e NS_NETMODE="HOST" -e
	CPX_NW_DEV='eth1 eth2' -e CPX_CORES=5 - e CPX_CONFIG='{
2 "YIELD":"No" }
3 ' -e LS_IP=10.102.38.134 -e PLATFORM=CP1000 -v /var/cpx:/cpx --name
	cpx_host cpx:12.1-48.xx
```

En este ejemplo se crea un contenedor denominado mycpx basado en la imagen de Docker de NetScaler CPX.

El-Pparámetro es obligatorio. Indica a Docker que asigne los puertos expuestos en el contenedor por la imagen de NetScaler CPX Docker. Esto significa asignar los puertos 9080, 22, 9443 y 161/UDP a los puertos del host Docker que se seleccionan aleatoriamente del intervalo definido por el usuario. Este mapeo se hace para evitar conflictos. Si más tarde crea varios contenedores NetScaler CPX en el mismo host de Docker. Las asignaciones de puertos son dinámicas y se configuran cada vez que se inicia o se reinicia el contenedor. Los puertos se usan de la siguiente manera:

- 9080 se usa para HTTP
- 9443 se usa para HTTPs
- 22 utilizados para SSH
- 161/UDP se utiliza para SNMP.

Si quiere asignaciones de puertos estáticas, use el parámetro -p para configurarlas manualmente.

La opción --privileged=**true** se usa para ejecutar el contenedor en modo privilegiado. Si ejecuta NetScaler CPX con varios núcleos, debe proporcionar todos los privilegios del sistema a NetScaler CPX. Si quiere ejecutar NetScaler CPX con un solo núcleo, en lugar de esta opción, debe usar la --cap -add=NET_ADMINopción. La --cap-add=NET_ADMINopción le permite ejecutar el contenedor NetScaler CPX con todos los privilegios de red.

**--net=host Es una opción de comando de ejecución de docker estándar que especifica que el contenedor se ejecuta en la pila de red del host y tiene acceso a todos los dispositivos de red.

Nota

Si ejecuta NetScaler CPX en una red puente o ninguna, omita esta opción.

-e NS_NETMODE="HOST" es una variable de entorno específica de NetScaler CPX que le permite especificar que NetScaler CPX se inicie en modo host. Una vez que NetScaler CPX se inicia en modo host, configura cuatro reglas iptable predeterminadas en la máquina host para el acceso de administración a NetScaler CPX. Utiliza los siguientes puertos:

- 9995 para HTTP
- 9996 para HTTPS
- 9997 para SSH
- 9998 para SNMP

Si quiere especificar puertos diferentes, puede usar las siguientes variables de entorno:

- -e NS_HTTP_PORT=
- -e NS_HTTPS_PORT=
- -e NS_SSH_PORT=
- -e NS_SNMP_PORT=

Nota

Si ejecuta NetScaler CPX en puente o en ninguna red, ignore esta variable de entorno.

Se-e CPX_COREStrata de una variable de entorno opcional específica de NetScaler CPX. Puede usarlo para mejorar el rendimiento de la instancia de NetScaler CPX definiendo el número de motores de procesamiento que quiere que se inicie el contenedor de NetScaler CPX.

Nota

Por cada motor de procesamiento adicional que agregue, asegúrese de que el host de Docker contenga la cantidad equivalente de CPU virtuales y la cantidad de memoria en GB. Por ejemplo, si quiere agregar 4 motores de procesamiento, el host de Docker debe contener 4 vCPU y 4 GB de memoria.

-e EULA = yesSe trata de una variable de entorno específica de NetScaler CPX obligatoria, que se requiere para comprobar que ha leído y comprendido el Contrato de licencia de usuario final (EULA) disponible en:https://www.citrix.com/products/netscaler-adc/cpx-express.html.

El-e PLATFORM=CP1000parámetro especifica el tipo de licencia de NetScaler CPX.

Si ejecuta Docker en una red host, puede asignar interfaces de red dedicadas al contenedor NetScaler CPX mediante la variable de entorno –e CPX_NW_DEV. Debe definir las interfaces de red separadas por un espacio en blanco. Las interfaces de red que defina se guardan en el contenedor NetScaler CPX hasta que desinstala el contenedor NetScaler CPX. Cuando se aprovisiona el contenedor NetScaler CPX, todas las interfaces de red asignadas se agregan al espacio de nombres de redes NetScaler.

Nota

Si ejecuta NetScaler CPX en una red puente, puede cambiar la red de contenedores, por ejemplo, configurar otra conexión de red al contenedor o eliminar una red existente. A continuación, asegúrese de reiniciar el contenedor NetScaler CPX para usar la red actualizada.

```
1 docker run -dt --privileged=true --net=host -e NS_NETMODE="HOST" -e
EULA=yes -e CPX_NW_DEV='eth1 eth2' -e CPX_CORES=5 -e PLATFORM=CP1000
--name cpx_host cpx:12.0-53.x
```

-e CPX_CONFIG es una variable de entorno específica de NetScaler CPX que le permite controlar el rendimiento de rendimiento del contenedor NetScaler CPX. Cuando NetScaler CPX no recibe ningún tráfico entrante para procesar, produce la CPU durante este tiempo de inactividad, lo que resulta en un rendimiento bajo. Puede usar la variable de entorno CPX_CONFIG para controlar el rendimiento del rendimiento del contenedor NetScaler CPX en tales casos. Debe proporcionar los siguientes valores a la variable de entorno CPX_CONFIG para controlar el se valores a la variable de entorno CPX_CONFIG para controlar el se valores del contenedor NetScaler CPX en tales casos. Debe proporcionar los siguientes valores a la variable de entorno CPX_CONFIG para controlar el se valores del contenedor NetScaler CPX en tales casos. Debe proporcionar los siguientes valores a la variable de entorno CPX_CONFIG para controlar el se valores del contenedor NetScaler CPX en tales casos. Debe proporcionar los siguientes valores de la variable de entorno CPX_CONFIG para controlar el se valores del contenedor NetScaler CPX en tales casos. Debe proporcionar los siguientes valores de la variable de entorno CPX_CONFIG en formato JSON:

- Si quiere que el contenedor NetScaler CPX genere CPU en casos inactivos, defina { "YIELD"
 : "Yes" }
- Si quiere que el contenedor NetScaler CPX evite producir la CPU en casos inactivos para que pueda obtener un rendimiento de alto rendimiento, defina { "YIELD" : "No" }

```
1 docker run -dt --privileged=true --net=host -e NS_NETMODE="HOST" -e
	EULA=yes -e CPX_CORES=5 - e CPX_CONFIG='{
2 "YIELD":"No" }
3 ' -e PLATFORM=CP1000 --name cpx_host cpx:12.0-51.x
```

```
1 docker run -dt --privileged=true --net=host -e NS_NETMODE="HOST" -e
	EULA=yes -e CPX_CORES=5 - e CPX_CONFIG='{
2 "YIELD":"Yes" }
3 ' -e PLATFORM=CP1000 --name cpx_host cpx:12.0-51.xx
```

El – vparámetro es un parámetro opcional que especifica el punto de montaje del directorio de montaje de NetScaler CPX,/cpx. Un punto de montaje es un directorio en el host, en el que se monta el/cpxdirectorio. El/cpxdirectorio almacena los registros, los archivos de configuración, los certificados SSL y los archivos de volcado de memoria. En el ejemplo, el punto de montaje es/var/cpx y el directorio de montaje de NetScaler CPX es/cpx.

Si compró una licencia o tiene una licencia de evaluación, puede cargar la licencia en un servidor de licencias y especificar la ubicación del servidor de licencias con el comando docker run, mediante ele LS_IP=<LS_IP_ADDRESS> -e LS_PORT=<LS_PORT>parámetro. En este caso, no necesita aceptar el EULA.

Donde:

- LS_IP_ADDRESS es la dirección IP del servidor de licencias.
- LS_PORT es el puerto del servidor de licencias.

Puede ver las imágenes que se ejecutan en su sistema y los puertos asignados a los puertos estándar mediante el comando: docker ps

Implementación de instancias de NetScaler CPX mediante Docker Compose

Puede usar la herramienta Redactar de Docker para aprovisionar una sola instancia de NetScaler CPX o varias instancias de NetScaler CPX. Para aprovisionar instancias de NetScaler CPX mediante Compose, primero debe escribir un archivo de redacción. Este archivo especifica la imagen de NetScaler CPX, los puertos que quiere abrir para la instancia de NetScaler CPX y los privilegios de la instancia de NetScaler CPX.

Importante

Asegúrese de haber instalado la herramienta Docker Compose en el host.

Para aprovisionar varias instancias de NetScaler CPX:

1. Escribe un archivo de redacción, donde:

- <service-name> es el nombre del servicio que quieres aprovisionar.
- **image::** <repository><tag>indica el repositorio y las versiones de la imagen de NetScaler CPX.
- privileged: true proporciona todos los privilegios de root a la instancia de NetScaler CPX.
- **cap_add** proporciona privilegios de red a la instancia de NetScaler CPX.
- ** < host_directory_path > indica el directorio en el host de la ventana acoplable que quiere montar para la instancia de NetScaler CPX.
- ** <number_processing_engine>es el número de motores de procesamiento que quiere que se inicie la instancia de NetScaler CPX. Para cada motor de procesamiento adicional, asegúrese de que el host de Docker contenga la cantidad equivalente de CPU virtuales y la cantidad de memoria en GB. Por ejemplo, si quiere agregar 4 motores de procesamiento, el host de Docker debe contener 4 vCPU y 4 GB de memoria.

El archivo de redacción suele tener un formato similar al siguiente:

1	<service-name>:</service-name>
2	container_name:
3	<pre>image: <repository>:<tag></tag></repository></pre>
4	ports:
5	- 22
6	- 9080
7	- 9443
8	- 161/udp
9	- 35021-35030
10	tty: true
11	cap_add:
12	- NET_ADMIN
13	ulimits:
14	core: -1
15	volumes:
16	<pre>- <host_directory_path>:/cpx</host_directory_path></pre>
17	environment:
18	- EULA=yes
19	- CPX_CORES= <number_processing_engine></number_processing_engine>
20	- CPX_CONFIG='{
21	"YIELD":"Yes" }
22	

```
CPX_0:
1
2
       container_name: CPX_0
3
       image: cpx:12.0-53.xx
4
       ports:
5
           - 9443
           -
6
              22
7
           -
             9080
8
           - 161/udp
9
           - 35021-35030
       tty: true
10
11
       cap_add:
           - NET_ADMIN
12
13
       ulimits:
```

Si quiere aprovisionar una sola instancia de NetScaler CPX, debe agregar la siguiente línea al archivo de redacción: container_name:<name_of_container>

Ejecute el siguiente comando para aprovisionar varias instancias de NetScaler CPX:

```
docker-compose -f <compose_file_name> scale <service-name>=<number of
instances> up -d docker-compose -f docker-compose.yml scale cpxlb=3
 up -d
```

Si quiere aprovisionar una sola instancia de NetScaler CPX, ejecute el siguiente comando: dockercompose -f <compose_file_name> up -d

Agregar instancias de NetScaler CPX a Citrix ADM

November 23, 2023

Debe agregar las instancias de NetScaler CPX instaladas en un host de Docker al software Citrix Application Delivery Management (ADM) si quiere administrar y supervisar estas instancias.

Puede agregar instancias al configurar ADM por primera vez o más tarde.

Para agregar instancias, debes crear un perfil de instancia y especificar el nombre de host o la dirección IP de cada instancia, o un intervalo de direcciones IP. Este perfil de instancia contiene el nombre de usuario y la contraseña de las instancias que quiere agregar a Citrix ADM. Para cada tipo de instancia, está disponible un perfil predeterminado. Por ejemplo, ns-root-profile es el perfil predeterminado para las instancias de Citrix ADC. Este perfil se define mediante las credenciales de administrador de ADC predeterminadas. Si ha cambiado las credenciales de administrador predeterminadas de las instancias, puede definir perfiles de instancia personalizados para esas instancias. Si cambia las credenciales de una instancia después de que se descubre la instancia, debe modificar el perfil de la instancia o crear un perfil nuevo y, a continuación, volver a descubrir la instancia.

Requisitos previos

Asegúrese de lo siguiente:

• Se instaló el software Citrix ADM en Citrix XenServer. Para obtener más información, consulte la documentación de Citrix ADM.

• Se instalaron las instancias de NetScaler CPX en un host de Docker.

Para agregar instancias de NetScaler CPX a ADM:

- 1. En un explorador web, escriba la dirección IP de **Citrix Application Delivery Management** (por ejemplo, http://192.168.100.1).
- 2. En los campos **Nombre de usuario** y **Contraseña**, introduzca las credenciales de administrador. Las credenciales de administrador predeterminadas son **nsroot** y **nsroot**.
- 3. Vaya a Redes > Instancias > Citrix ADC y haga clic en la ficha CPX.
- 4. Haga clic en Agregar para agregar nuevas instancias CPX en Citrix ADM.
- 5. Se abrirá la página Add NetScaler CPX. Introduzca los valores de los siguientes parámetros:
 - a) Puede agregar instancias de CPX proporcionando la dirección IP accesible de la instancia CPX o la dirección IP del contenedor Docker donde está alojada la instancia de CPX.
 - b) Seleccione el perfil de la instancia CPX.
 - c) Seleccione el sitio en el que se van a implementar las instancias.
 - d) Seleccione el agente.
 - e) Como opción, puede introducir el par clave-valor en la instancia. Agregar par clave-valor hace que sea fácil para usted buscar la instancia más adelante.

G Add Citrix ADC CPX

Enter Device IP Address	Import from file
Enter one or more hostnames,	IP addresses , and/or a range of IP addresses (for example, 10.102.40.30-10.102.40.45) using a comma separator.
Routable IP/ Docker IP*	
172.31.32.161	0
Profile Name*	
Docker-profile	✓ Add Edit
Site*	
Ohio-site	V Add Edit
Agent	
Click to select	>
Tags	
Кеу	Value +
OK Close	

6. Haga clic en **Aceptar**.

Nota

Si quiere volver a detectar una instancia, vaya a **Redes > Instancias > Citrix ADC > CPX**, seleccione la instancia que quiere redescubrir y, a continuación, en la lista desplegable **Seleccionar acción**, haga clic en **Redescubrir**.

Agregar instancias de NetScaler CPX a Citrix ADM mediante variables de entorno

También puede agregar las instancias de NetScaler CPX a Citrix ADM mediante variables de entorno. Para agregar instancias, debe configurar las siguientes variables de entorno para la instancia de NetScaler CPX.

- NS_MGMT_SERVER Dirección IP ADM/FQDN
- HOST Dirección IP del nodo
- NS_HTTP_PORT Puerto HTTP asignado en el nodo
- NS_HTTPS_PORT- Puerto HTTPS asignado en el nodo
- NS_SSH_PORT Puerto SSH mapeado en el nodo
- NS_SNMP_PORT Puerto SNMP asignado en el nodo
- NS_ROUTABLE (La dirección IP del pod de NetScaler CPX no se puede redirigir desde el exterior).

A continuación, se muestra un comando docker run de ejemplo para agregar una instancia de NetScaler CPX a Citrix ADM.

```
1 docker run -dt --privileged=true -p 9080:9080 -p 9443:9443 -p 9022:22
-p 9161:161 -e EULA=yes -e NS_MGMT_SERVER=abc-mgmt-server.com -e
HOST=10.1.1.1 -e NS_HTTP_PORT=9080 -e NS_HTTPS_PORT=9443 -e
NS_SSH_PORT=9022 -e NS_SNMP_PORT=9161 -e NS_ROUTABLE=0 --ulimit
core=-1 - name test cpx:latest
```

Configuración de NetScaler CPX

November 23, 2023

Puede configurar una instancia de NetScaler CPX accediendo a la solicitud de la CLI a través del host de Linux Docker o mediante las API de NetScaler NITRO.

Configuración de una instancia de NetScaler CPX mediante la interfaz de línea de comandos

Acceda al host de Docker e inicie sesión en el indicador SSH de la instancia, como se muestra en esta ilustración. Las credenciales de administrador predeterminadas para iniciar sesión en una instancia de NetScaler CPX son root/linux.

```
root@ubuntu:~# ssh -p 32777 root@127.0.0.1
root@127.0.0.1's password:
Welcome to Ubuntu 14.04.3 LTS (GNU/Linux 3.19.0-25-generic x86_64)
* Documentation: https://help.ubuntu.com/
Last login: Tue Dec 15 02:45:42 2015 from 172.17.0.1
root@10:~#
```

Escriba el siguiente comando para usar el símbolo de línea de comandos de la instancia para ejecutar los comandos de la CLI: **cli_script.sh** "<comando>"

Ejemplo:

root@	10:~# cli_script.sh	n "show ip"	
exec:	show ip		
	Ipaddress	Traffic Domain	Туре
1)	172.17.0.4	0	NetScaler IP VIP
2)	192.0.0.1	0	SNIP

Para cerrar sesión en el símbolo de la instancia, escriba **logout**.

Configuración de una instancia de NetScaler CPX mediante la API de NITRO

Puede usar la API NetScaler NITRO para configurar instancias de NetScaler CPX.

Para configurar instancias de NetScaler CPX mediante la API de Nitro, en un explorador web, escriba:

http://<host_IP_address>:<port>/nitro/v1/config/<resource-type\</pre>

Para recuperar estadísticas mediante la API de Nitro, en un explorador web, escriba:

http://\<host_IP_address\>:\<port\>/nitro/v1/stat/\<resource-type
\</pre>

Para obtener más información sobre el uso de la API de NITRO, consulte Servicios web REST. Para NetScaler CPX, utilice CPX IP address:port donde netscaler-ip-address se menciona.

Configuración de una instancia de NetScaler CPX mediante trabajos

Puede configurar instancias de NetScaler CPX creando y ejecutando trabajos en Citrix ADM. Puede usar las configuraciones de las plantillas de configuración, extraer configuraciones disponibles en otros dispositivos y usar configuraciones guardadas en archivos de texto. También puede registrar las configuraciones realizadas mediante la utilidad de configuración de otras instancias. A continuación, Citrix ADM muestra los comandos de la CLI correspondientes para que los utilice en la instancia de NetScaler CPX. Después de seleccionar la configuración, debe seleccionar las instancias de NetScaler CPX en las que quiere cargar la configuración, especificar los valores de las variables y ejecutar el trabajo.

Para configurar instancias de NetScaler CPX mediante Trabajos:

- 1. Inicie sesión en Citrix ADM con las credenciales administrativas.
- 2. Vaya a Redes > Trabajos de configuracióny, a continuación, haga clic en Crear trabajo.
- 3. Especifique los valores requeridos y seleccione el origen de configuración. También puede escribir los comandos que quiere ejecutar.

Select Configuration Select In	nstances	Specify Variable Values	
bb Name*	Instance Type*		
px-single-host	NetScaler 🗸		
Configuration Source	=	New	
Configuration Template \checkmark	1 SSH 🔻	add service db1 172.17.0.10 HTTP 80	
Drag and drop the template to the Commands	2 SSH 🔻	add service db2 172.17.0.11 HTTP 80	
field in the right pane. You can also edit the configuration and save the template with a	3 SSH 🔻	add lb vserver cpx-vip HTTP 172.17.0.4 81	
different name	4 SSH ▼	bind lb vserver cpx-vip db1	

4. Seleccione las instancias de NetScaler CPX en las que quiera ejecutar la configuración y haga clic en **Siguiente**.

~	Select Computation	Select Instances	specify variable values
Select t	he target instances on which you v	vant to run the configuration.	
Add	Instances		
-	IP Address		Name
0			
	172.17.0.150		10.102.31.190

5. Especifique la configuración de ejecución y haga clic en Finalizar para ejecutar los comandos en la instancia de NetScaler CPX. Si quiere guardar la configuración y ejecutarla más tarde, haga clic en **Guardar y salir**.

You can either run the commands now or schedule to run the commands at a later time. On Command Failure* Ignore error and continue Execution Mode* Now Execution Settings • Execute in Sequence • Execute in Parallel	Select Configuration	Select Instances	Specify Variable Values	C/> Execute
On Command Failure* Ignore error and continue Execution Mode* Now Execution Settings Execute in Sequence Execute in Parallel	You can either run the commands nov	<i>i</i> or schedule to run the commands at a	later time.	
Ignore error and continue Execution Mode* Now Execution Settings • Execute in Sequence • Execute in Parallel	On Command Failure*			
Execution Mode* Now Execution Settings Execute in Sequence Execute in Parallel	Ignore error and continue	•		
Now Execution Settings Execute in Sequence Execute in Parallel	Execution Mode*			
Execution Settings Execute in Sequence Execute in Parallel	Now			
Receive Execution Report Through	Execution Settings Execute in Sequence Execute in Parallel Receive Execution Report Through Email			

Configuración de la transmisión de registros en una instancia de NetScaler CPX

Puede configurar la transmisión de registros en una instancia de NetScaler CPX para recopilar datos de rendimiento de páginas web, información a nivel de flujo y sesión de usuario e información de base

de datos necesaria para la supervisión y el análisis del rendimiento de las aplicaciones. Estos registros de datos se envían a Citrix ADM, donde puede ver informes históricos y en tiempo real de todas sus aplicaciones.

Para configurar la transmisión de registros en una instancia de NetScaler CPX, primero debe habilitar la función AppFlow y el demonio ulfd en la instancia de NetScaler CPX. Al habilitar el demonio ulfd, debe especificar la dirección IP de Citrix ADM en la que quiere supervisar los informes históricos y en tiempo real. A continuación, debe configurar un recopilador, una acción y una directiva de AppFlow y vincular la directiva de AppFlow de forma global.

El demonio ulfd envía todos los registros de flujo a Citrix ADM en un único formato de registro unificado.

Puede configurar la transmisión de registros mediante la interfaz de línea de comandos de una instancia de NetScaler CPX o mediante la función Trabajos de Citrix ADM.

Para poder supervisar los registros, debe agregar la instancia de NetScaler CPX a Citrix ADM. Para obtener más información sobre cómo agregar una instancia de NetScaler CPX a Citrix ADM, consulte Instalación de una instancia de NetScaler CPX mediante Citrix ADM.

Para configurar la transmisión de registros en una instancia de NetScaler CPX:

- 1. Habilite la función AppFlow ejecutando el siguiente comando: enable ns feature AppFlow
- 2. Habilite el demonio ulfd mediante el siguiente comando: set ns param -ulfd ENABLED -loggerip <NUMS_IP_Address>
- 3. Ejecute los siguientes comandos para configurar un recopilador, una acción y una directiva de AppFlow, y para vincular la directiva de forma global:

```
add appflow collector <name> -IPAddress <ipaddress>`
1
2
      set appflow param -templateRefresh 3600 -httpUrl ENABLED -
3
          httpCookie ENABLED -httpReferer ENABLED -httpMethod ENABLED -
          httpHost ENABLED -httpUserAgent ENABLED -httpContentType ENABLED
           -httpAuthorization ENABLED -httpVia ENABLED -httpXForwardedFor
          ENABLED -httpLocation ENABLED -httpSetCookie ENABLED -
          httpSetCookie2 ENABLED -connectionChaining ENABLED -httpDomain
          Enabled
4
      add appflow action <name> --collectors <string> ... [-
5
          clientSideMeasurements (Enabled Disabled) ]
6
      add appflow policy <name> true <action>
7
8
      bind appflow global <policyName> <priority> [<</pre>
          gotoPriorityExpression [-type <type>]
```

Nota:

Debe configurar el recopilador de AppFlow con una dirección IP ficticia.

Configuración de NetScaler CPX mediante un archivo de configuración

November 23, 2023

En lugar de utilizar los trabajos de configuración de interfaz de línea de comandos (cli_script.sh), API de NITRO o NetScaler Management and Analytics (MAS) para configurar NetScaler CPX, puede configurar NetScaler CPX mediante un archivo de configuración estática al implementar la instancia de NetScaler CPX.

Puede proporcionar un archivo de configuración estática como archivo de entrada mientras implementa el contenedor NetScaler CPX. Durante el inicio del contenedor NetScaler CPX, el contenedor se configura en función de la configuración especificada en el archivo de configuración estática. Esta configuración incluye la configuración específica de NetScaler y los comandos bash shell que se pueden ejecutar dinámicamente en el contenedor NetScaler CPX.

Estructura del archivo de configuración estática

Como se mencionó anteriormente, cuando se implementa NetScaler CPX, se configura en función de las configuraciones especificadas en el archivo de configuración estática.

El archivo de configuración estática es un archivo .conf que incluye dos etiquetas #NetScaler Commands y #Shell Commands. Debajo de la etiqueta #NetScaler Commands, debe agregar todos los comandos de NetScaler para configurar la configuración específica de NetScaler en NetScaler CPX. Debajo de la etiqueta #Shell Commands, debe agregar los comandos de shell que quiere ejecutar en NetScaler CPX.

Durante la implementación del contenedor NetScaler CPX, los comandos de NetScaler y los comandos de shell se ejecutan en el contenedor en el orden especificado en el archivo de configuración.

Importante:

- Las etiquetas se pueden repetir varias veces en el archivo de configuración.
- Las etiquetas no distinguen entre mayúsculas y minúsculas.
- El archivo de configuración debe estar presente en el directorio /etc como archivo cpx . conf en el sistema de archivos del contenedor.
- El archivo de configuración también puede incluir comentarios. Debe agregar un carácter "#"antes de sus comentarios.

- Si hay casos de error al implementar el contenedor NetScaler CPX con el archivo de configuración, los errores se registran en el archivo ns.log en el contenedor.
- Al reiniciar el contenedor NetScaler CPX, el archivo de configuración se vuelve a aplicar en el contenedor.

```
1 #NetScaler Commands
3
  add lb vserver v1 http 1.1.1.1 80
4
5 add service s1 2.2.2.2 http 80
6
  bind lb vserver v1 s1
7
8
  #Shell Commands
9
10
11 touch /etc/a.txt
13 echo "this is a" > /etc/a.txt
14
15 #NetScaler Commands
16
17 add lb vserver v2 http
18
19 #Shell Commands
20
21 echo "this is a 1" >> /etc/a.txt
23 #NetScaler Commands
24
25
   add lb vserver v3 http
26
27
   #This is a test configuration file
```

Para instalar un contenedor NetScaler CPX y configurar dinámicamente el contenedor NetScaler CPX en función de un archivo de configuración, monte el archivo de configuración estática mediante la opción – v del comando docker run:

```
1 docker run -dt --privileged=true -e EULA=yes --ulimit core=-1 -v /tmp/
cpx.conf:/etc/cpx.conf --name mycpx store/citrix/netscalercpx
:12.1-48.13
```

Configuración de controladores de registro de Docker

November 23, 2023

Docker incluye mecanismos de registro denominados "controladores de registro"para ayudarlo a obtener información de los contenedores en ejecución. Puede configurar un contenedor NetScaler CPX para que reenvíe los registros que genere a los controladores de registro de Docker. Para obtener más información sobre los controladores de registro de docker, consulte https://docs.docker.com/config/containers/logging/configure/.

De forma predeterminada, todos los registros generados por el contenedor NetScaler CPX se almacenan en un archivo /cpx/log/ns.log en el host de la ventana acoplable. Cuando inicia el contenedor NetScaler CPX con el comando docker run, puede configurarlo para que reenvíe todos los registros generados a un controlador de registro de docker mediante la opción --log-driver. Si el controlador de registro tiene parámetros configurables, puede configurarlos mediante la opción --log-opt <NAME>=<VALUE>.

En el siguiente ejemplo, el contenedor NetScaler CPX está configurado para reenviar todos los registros generados mediante syslog como controlador de registro.

```
1 docker run -dt --privileged=true --log-driver syslog --log-opt syslog-
address=udp://10.106.102.190:514 -e EULA=yes --ulimit core=-1 --name
test store/citrix/netscalercpx:12.1-48.13
```

Del mismo modo, en el siguiente ejemplo, el contenedor NetScaler CPX está configurado para reenviar todos los registros generados mediante Splunk como controlador de registro.

```
1 docker run -dt --privileged=true --log-driver=splunk --log-opt splunk-
token=176FCEBF-4CF5-4EDF-91BC-703796522D20 --log-opt splunk-url=
https://splunkhost:8088 -e EULA=yes --ulimit core=-1 --name test
store/citrix/netscalercpx:12.1-48.13
```

Compatibilidad con redirección dinámica en NetScaler CPX

November 23, 2023

NetScaler CPX admite el protocolo de redirección dinámica BGP. El objetivo clave del protocolo de redirección dinámica es anunciar la dirección IP del servidor virtual en función del estado de los servicios, vinculados al servidor virtual. Ayuda a un enrutador ascendente a elegir la mejor entre varias rutas a un servidor virtual distribuido topográficamente.

Nota: El protocolo de redirección dinámica BGP se admite en la compilación 52.15 de la versión 12.1 de Citrix ADC y versiones posteriores.

Para obtener información sobre la contraseña no predeterminada en NetScaler CPX, consulte la sección Support **for** using a non-**default** password in NetScaler CPX del documento Configuración de NetScaler CPX.

En una red de host único, el cliente, los servidores y la instancia de NetScaler CPX se implementan como contenedores en el mismo host de Docker. Todos los contenedores están conectados a través

del puente docker0. En este entorno, la instancia de NetScaler CPX actúa como proxy para las aplicaciones aprovisionadas como contenedores en el mismo host de Docker. Para obtener información sobre la implementación en modo de red de host de NetScaler CPX, consulte Modo de red de host.





En esta topología, los servidores virtuales se configuran y anuncian (según el estado de los servicios) a la red o enrutador ascendente mediante BGP.

Realice los siguientes pasos para configurar BGP en NetScaler CPX en un solo host Docker con el modo de conexión en red puente.

Configurar la inyección de estado de ruta basada en BGP mediante la API REST en NetScaler CPX

1. Cree un contenedor a partir de la imagen de NetScaler CPX mediante el siguiente comando:

```
1 docker run -dt --privileged=true -p 22 -p 80 -p 161 -e EULA=yes --
ulimit core=-1 cpx: <tag>
```

Por ejemplo:

```
1 docker run -dt --privileged=true -p 22 -p 80 -p 161 -e EULA=yes --
ulimit core=-1 cpx:12.1-50.16
```

2. Inicie sesión en el contenedor con el siguiente comando:

1 docker exec -it <container id> bash

3. Habilite la función BGP con el siguiente comando:

1 cli_script.sh "enable ns feature bgp"

4. Obtenga el NSIP mediante el comando show ns ip:

1 cli_script.sh "show ns ip"

5. Agregue el servidor virtual mediante el siguiente comando:

1 cli_script.sh "add lb vserver <vserver_name> http <VIP> <PORT>"

- 6. Agregue servicios y enlace servicios al servidor virtual.
- 7. Habilite hostroute para el VIP mediante el siguiente comando:

1 cli_script.sh "set ns ip <VIP> -hostroute enabled"

Salga del contenedor y envíe comandos BGP NITRO desde el host al NSIP en el puerto 9080.

8. Configure el enrutador BGP:

Por ejemplo, si quieres configurar:

```
    router bgp 100
    Neighbour 172.17.0.2 remote-as 101
    Redistribute kernel
```

Especifique el comando de la siguiente manera:

```
curl -u username:password http://<NSIP>:9080/nitro/v1/config/ -X
       POST --data 'object={
    "routerDynamicRouting": {
2
   "bgpRouter" : {
3
   "localAS":100, "neighbor": [{
4
    "address": "172.17.0.2", "remoteAS": 101 }
5
   ], "afParams":{
6
    "addressFamily": "ipv4", "redistribute": {
7
   "protocol": "kernel" }
8
9
    }
10
    }
11
     }
12
     }
    r.
13
```

9. Instale las rutas BGP aprendidas en el PE mediante el siguiente comando NITRO:

10. Verifique el estado de adyacencia de BGP mediante el siguiente comando NITRO:

```
1 curl -u username:password http://<NSIP>:9080/nitro/v1/config/
routerDynamicRouting/bgpRouter
```

Salida de muestra:

```
root@ubuntu:~# curl -u username:password http://172.17.0.3:9080/
        nitro/v1/config/routerDynamicRouting/bgpRouter
2
    {
     "errorcode": 0, "message": "Done", "severity": "NONE", "
3
         routerDynamicRouting":{
    "bgpRouter":[{
4
     "localAS": 100, "routerId": "172.17.0.3", "afParams": [ {
5
     "addressFamily": "ipv4" }
6
7
    , {
    "addressFamily": "ipv6"
8
                                 ļ
9
     ], "neighbor": [ {
10
     "address": "172.17.0.2", "remoteAS": 101, "ASOriginationInterval
         ": 15, "advertisementInterval": 30, "holdTimer": 90, "
         keepaliveTimer": 30, "state": "Connect", "singlehopBfd":
false, "multihopBfd": false, "afParams": [ {
     "addressFamily": "ipv4", "activate": true }
11
12
    , {
     "addressFamily": "ipv6", "activate": false }
13
14
     1
```

11. Compruebe que las rutas aprendidas a través de BGP estén instaladas en el motor de paquetes con el siguiente comando:

1 cli_script.sh " show route "

12. Guarde la configuración mediante el siguiente comando:

1 cli_script.sh "save config"

La configuración de redirección dinámica se guarda en el archivo /nsconfig/ZebOS.conf.

Actualización de una instancia de NetScaler CPX

November 23, 2023

Para actualizar una instancia de NetScaler CPX, apáguela, instale la versión más reciente en el mismo punto de montaje y, a continuación, elimine la instancia anterior. Un punto de montaje es un directorio en el que se monta el directorio **/cpx** en el host.

Por ejemplo, para montar el directorio **/cpx** de la instancia de NetScaler CPX existente en el directorio **/var/cpx** del host, el punto de montaje es **/var/cpx** y el directorio de montaje de NetScaler CPX es **/cpx**,

como se muestra a continuación:

```
1 root@ubuntu:~# docker run -dt -e EULA=yes --name mycpx -v /var/cpx
:/cpx --ulimit core=-1 cpx:12.1-48.xx
```

Requisitos previos

Asegúrese de lo siguiente:

 Detalles del directorio host en el que montó el directorio /cpx de la instancia de NetScaler CPX existente. Puede usar el comando docker inspect <containerName>, donde < containerName> es el nombre del contenedor NetScaler CPX, para mostrar información sobre el directorio host.

El resultado del comando proporciona los detalles de las configuraciones del contenedor, incluidos los volúmenes. En la entrada "**Mounts**", la subentrada"**Source**"muestra la ubicación del directorio del host en el host.



 Descargue el archivo de imagen Docker de NetScaler CPX más reciente https://www.microl oadbalancer.com/get-it-now y, a continuación, cargue la imagen de Docker de NetScaler CPX. Para cargar la imagen, navega hasta el directorio en el que guardaste el archivo de imagen de Docker. Use el comando docker load -i <image_name> para cargar la imagen. Después de cargar la imagen de NetScaler CPX, puede introducir el comando docker images para mostrar información sobre la imagen:

1	root@ubuntu:	~# docker lo	ad -i cpx-12.	0-41.10.gz	
2					
3	root@ubuntu:	~# docker או	lages		
5	REPOSITORY	TAG	IMAGE ID	CREATED	VIRTUAL SIZE
6					
7	срх	12.0-41.10	2e97aadf918b	43 hours ago	414.5 MB

Para actualizar una instancia de NetScaler CPX

1. Detenga la instancia de NetScaler CPX existente introduciendo el comando docker stop <containerName>, donde <containerName> es el nombre de la instancia de NetScaler CPX.

```
1 root@ubuntu:~# docker stop mycpx
2 mycpx
```

2. Con el comando docker run, implemente la última instancia de NetScaler CPX desde la imagen de NetScaler CPX que cargó en el host. Asegúrese de implementar la instancia en el mismo punto de montaje (por ejemplo, /var/cpx:/cpx) que usó para la instancia de NetScaler CPX existente.

```
1 root@ubuntu:~# docker run -dt -P -e CPX_CORES=1 --name latestcpx
--ulimit core=-1 -e EULA=yes -v /var/cpx:/cpx --cap-add=
NET_ADMIN cpx:12.0-41.10
```

Puede introducir el comando docker ps para comprobar que la instancia de NetScaler CPX implementada es la versión más reciente.



3. Después de comprobar que implementó la instancia CPX correcta de NetScaler, introduzca el comando **docker rm** <containerName> para eliminar la instancia anterior.

```
1 root@ubuntu:~# docker rm mycpx
2 mycpx
```

Uso de servidores virtuales comodín en la instancia de NetScaler CPX

November 23, 2023

Cuando aprovisiona una instancia de NetScaler, el motor de Docker solo asigna una dirección IP privada (una dirección IP) a una instancia de NetScaler CPX. Las tres funciones IP de una instancia de NetScaler se multiplexan en una dirección IP. Esta única dirección IP utiliza diferentes números de puerto para funcionar como NSIP, SNIP y VIP.

La dirección IP única que asigna el motor de Docker es dinámica. Agregue los servidores virtuales de equilibrio de carga (LB) o conmutación de contenido (CS) mediante la dirección IP única o con la dirección IP 127.0.0.1. Los servidores virtuales creados con 127.0.0.1 se denominan Servidores virtuales
comodín. De forma predeterminada, cuando crea un servidor virtual comodín, NetScaler CPX reemplaza la dirección IP asignada del servidor virtual comodín. La dirección IP asignada es 127.0.0.1, con el NSIP asignado a la instancia NetScaler CPX por el motor Docker.

En implementaciones de NetScaler CPX de alta disponibilidad, puede agregar servidores virtuales comodín en una de las instancias de NetScaler CPX y copiar el archivo **ns.conf** de la instancia a otras instancias de NetScaler CPX en la implementación y garantizar que la configuración de NetScaler sea coherente en todos los Instancias de NetScaler CPX en la implementación en lugar de identificar la dirección IP única asignada por el motor de Docker a la instancia de NetScaler y crear los servidores virtuales LB o CS en función de la dirección IP única en todas las instancias de NetScaler CPX de la implementación.

Puntos a tener en cuenta:

- Asegúrese de que el número de puerto que asigna al servidor virtual comodín no lo utilice ningún otro servidor virtual de la implementación.
- La incorporación de servidor virtual comodín falla si el número de puerto que asigna al servidor virtual comodín ya está siendo utilizado por los servicios internos.
- El servidor virtual comodín no admite el carácter *.

Para crear un servidor virtual de equilibrio de carga comodín, en el símbolo del sistema, introduzca el siguiente comando:

```
1 add lb vserver <name> <serviceType> 127.0.0.1 <port>
2
3 add lb vserver testlbvserver HTTP 127.0.0.1 30000
```

Para crear un servidor virtual de conmutación de contenido comodín, en el símbolo del sistema, escriba el siguiente comando:

```
1 add cs vserver <name> <serviceType> 127.0.0.1 <port>
2
3 add cs vserver testcsvserver HTTP 127.0.0.1 30000
```

Implementar NetScaler CPX como proxy para permitir el flujo de tráfico de este a oeste

November 23, 2023

En esta implementación, la instancia de NetScaler CPX actúa como un proxy para permitir la comunicación entre los contenedores de aplicaciones que residen en varios hosts. La instancia de NetScaler CPX se aprovisiona junto con las aplicaciones en varios hosts y proporciona la ruta más corta para la comunicación. La siguiente imagen ilustra el flujo de tráfico entre dos aplicaciones a través de las instancias de NetScaler CPX.



Esta imagen muestra el flujo de tráfico entre la aplicación C y la aplicación B y entre la aplicación A y la aplicación B. Cuando la aplicación C (en cualquiera de los hosts) envía una solicitud a B, la solicitud se recibe primero en el contenedor NetScaler CPX en el mismo host que la aplicación C. A continuación, el contenedor NetScaler CPX pasa el tráfico al contenedor NetScaler CPX alojado en el mismo host que la aplicación B y, a continuación, el tráfico se reenvía a la aplicación B. Se sigue una ruta de tráfico similar cuando la aplicación A envía una solicitud a la aplicación B.

En este ejemplo, también se implementa un NetScaler MPX para permitir el tráfico a las aplicaciones desde Internet a través de un VIP global. El tráfico de NetScaler MPX se recibe en los contenedores de NetScaler CPX, que luego distribuyen el tráfico entre los contenedores de aplicaciones.

El siguiente diagrama ilustra esta topología con las configuraciones que deben establecerse para que se produzca la comunicación.



Entities configured on the CPX container

VIPs	Services	
VIP-A1	Svc-A1	
VIP-B1	Svc-B1	
VIP-C1	Svc-VIP-C2 Svc-VIP-C3	

	Entities	configured	on	the	CPX	container	
--	----------	------------	----	-----	-----	-----------	--

VIPs	Services
VIP-A2	Svc-A2
VIP-B2	Svc-VPI-B1
VIP-C2	Svc-C2

ntities configured on the CPX contain

F

VIPs	Services
VIP-A3	Svc-VIP-A1 Svc-VIP-A2
VIP-B3	Svc-VIP-B1
VIP-C3	Svc-C3

En la siguiente tabla se enumeran las direcciones IP y los puertos que se configuran en las instancias de NetScaler CPX en este ejemplo de configuración.

Docke	er Host 1	Docke	er Host 2	Docker Host 3			
VIPs	Services VIPs Services			VIPs	Services		
	Bound to the		Bound to the		Bound to the		
	VIP		VIP		VIP		
VIP-A1	SVC-A1	VIP-A2	SVC-A2	VIP-A3	SVC-VIP-A1		
172.17.0.2:3	10.102.29.100:	172.17.0.3:3	10.102.29.105:	172.17.0.4:3	10.102.29.100:		
0000	80	0000	80	0000	30000		
					SVC-VIP-A2		
					10.102.29.105:		
					30000		
VIP-B1	SVC-B1	VIP-B2	SVC-VIP-B1	VIP-B3	SVC-VIP-B1		
172.17.0.2:3	10.102.29.100:	172.17.0.3:3	10.102.29.100:	172.17.0.4:3	10.102.29.100:		
0001	90	0001	30001	0001	30001		
VIP-C1	SVC-VIP-C2	VIP-C2	SVC-C2	VIP-C3	SVC-C3		
172.17.02:30	10.102.29.105:	172.17.0.3:3	10.102.29.105:	172.17.0.4:3	10.102.29.110:		
002	30002	0002	70	0002	70		
	SVC-VIP-C3						
	10.102.29.110:						
	30002						

Para configurar este caso de ejemplo, ejecute el siguiente comando en el símbolo del shell de Linux mientras crea el contenedor NetScaler CPX en los tres hosts de Docker:

Ejecute los siguientes comandos mediante la función Trabajos en NetScaler MAS o mediante las API de NITRO.

En la instancia de NetScaler CPX en Docker Host 1:

```
add lb vserver VIP-A1 HTTP 172.17.0.2 30000
1
       add service svc-A1 10.102.29.100 HTTP 80
2
       bind lb vserver VIP-A1 svc-A1
3
       add lb vserver VIP-B1 HTTP 172.17.0.2 30001
4
       add service svc-B1 10.102.29.100 HTTP 90
5
6
       bind lb vserver VIP-B1 svc-B1
       add lb vserver VIP-C1 HTTP 172.17.0.2 30002
7
       add service svc-VIP-C2 10.102.29.105 HTTP 30002
8
9
       add service svc-VIP-C3 10.102.29.110 HTTP 30002
       bind lb vserver VIP-C1 svc-VIP-C2
10
11
       bind lb vserver VIP-C1 svc-VIP-C3
```

En la instancia de NetScaler CPX en el host 2 de Docker:

```
add lb vserver VIP-A2 HTTP 172.17.0.3 30000
1
      add service svc-A2 10.102.29.105 HTTP 80
2
      bind lb vserver VIP-A2 svc-A2
3
      add lb vserver VIP-B2 HTTP 172.17.0.3 30001
4
5
      add service svc-VIP-B1 10.102.29.100 HTTP 30001
      bind lb vserver VIP-B2 svc-VIP-B1
6
      add lb vserver VIP-C2 HTTP 172.17.0.3 30002
7
8
      add service svc-C2 10.102.29.105 HTTP 70
9
      bind lb vserver VIP-C2 svc-C2
```

En la instancia de NetScaler CPX en el host 3 de Docker:

```
add lb vserver VIP-A3 HTTP 172.17.0.4 30000
1
2
       add service svc-VIP-A1 10.102.29.100 HTTP 30000
       add service svc-VIP-A2 10.102.29.105 HTTP 30000
3
       bind lb vserver VIP-A3 svc-VIP-A1
4
       bind lb vserver VIP-A3 svc-VIP-A2
5
       add lb vserver VIP-B3 HTTP 172.17.0.4 30001
6
7
       add service svc-VIP-B1 10.102.29.100 HTTP 30001
       bind lb vserver VIP-B3 svc-VIP-B1
8
       add lb vserver VIP-C3 HTTP 172.17.0.4 30002
9
       add service svc-C3 10.102.29.110 HTTP 70
10
       bind lb vserver VIP-C3 svc-C3
11
```

Implementación de NetScaler CPX en una red de host único

November 23, 2023

En una red de host único, la instancia de NetScaler CPX actúa como proxy entre los contenedores de aplicaciones en el mismo host. En esta capacidad, la instancia de NetScaler CPX proporciona escalabilidad y seguridad a las aplicaciones basadas en contenedores. Además, optimiza el rendimiento y también proporciona información sobre los datos de telemetría.

En una red de host único, el cliente, los servidores y la instancia de NetScaler CPX se implementan como contenedores en el mismo host de Docker. Todos los contenedores están conectados a través del puente docker0.

En este entorno, la instancia de NetScaler CPX actúa como proxy para las aplicaciones aprovisionadas como contenedores en el mismo host de Docker.



Esta ilustración ilustra la topología de host único.

En este ejemplo, un contenedor de aplicaciones web (172.17.0.2) es el cliente y los dos contenedores de base de datos, DB1 (172.17.0.10) y DB2 (172.17.0.11), son los servidores. El contenedor NetScaler CPX (172.17.0.4) se encuentra entre el cliente y los servidores que actúan como proxy.

Para permitir que la aplicación web se comunique con los contenedores de base de datos a través de NetScaler CPX, primero debe configurar dos servicios en el contenedor NetScaler CPX para que representen a los dos servidores. A continuación, configure un servidor virtual mediante la dirección IP de NetScaler CPX y un puerto HTTP no estándar (como 81) porque NetScaler CPX reserva el puerto HTTP estándar 80 para la comunicación NITRO.

En esta topología, no tiene que configurar ninguna regla de NAT porque el cliente y el servidor están en la misma red.

Para configurar este caso, ejecute los siguientes comandos mediante la función Trabajos en NetScaler MAS o mediante las API de NITRO:

```
    add service db1 HTTP 172.17.0.10 80
    add service db2 HTTP 172.17.0.11 80
    add lb vserver cpx-vip HTTP 172.17.0.4 81
    bind lb vserver cpx-vip db1
    bind lb vserver cpx-vip db2
```

Implementación de NetScaler CPX en una red de varios hosts

November 23, 2023

Una instancia de NetScaler CPX en una red multihost se puede configurar en una implementación de producción en el centro de datos, donde proporciona funciones de equilibrio de carga. Además, puede proporcionar funciones de supervisión y datos de análisis.

En una red de varios hosts, las instancias de NetScaler CPX, los servidores backend y los clientes se implementan en hosts diferentes. Puede utilizar topologías de varios hosts en implementaciones de producción en las que la instancia de NetScaler CPX equilibra la carga de un conjunto de aplicaciones y servidores basados en contenedores o incluso servidores físicos.

Topología 1: servidores NetScaler CPX y backend en el mismo host; cliente en una red diferente

En esta topología, la instancia de NetScaler CPX y los servidores de base de datos se aprovisionan en el mismo host de Docker, pero el tráfico del cliente se origina en otro lugar de la red. Esta topología se puede utilizar en una implementación de producción en la que la instancia de NetScaler CPX equilibra la carga de un conjunto de aplicaciones o servidores basados en contenedores.

El siguiente diagrama ilustra esta topología.



Linux Docker Host: 10.102.29.100

En este ejemplo, la instancia de NetScaler CPX (172.17.0.4) y los dos servidores, DB1 (172.17.0.10) y DB2 (172.17.0.11) se aprovisionan en el mismo host de Docker con la dirección IP 10.102.29.100. El cliente reside en otra parte de la red.

Las solicitudes de clientes que se originan en Internet se reciben en el VIP configurado en la instancia de NetScaler CPX, que luego distribuye las solicitudes entre los dos servidores.

Hay dos métodos que puede usar para configurar esta topología:

Método 1: uso de una dirección IP adicional y un puerto estándar para el VIP

- 1. Configure el VIP en el contenedor NetScaler CPX mediante una dirección IP adicional.
- 2. Configure una dirección IP adicional para el host de Docker.
- 3. Configure las reglas NAT para reenviar todo el tráfico recibido en la dirección IP adicional del host Docker a la dirección IP adicional del VIP.
- 4. Configure los dos servidores como servicios en la instancia de NetScaler CPX.
- 5. Por último, vincule los servicios al VIP.

Tenga en cuenta que en este ejemplo de configuración, la red 10.x.x.x indica una red pública.

Para configurar este caso de ejemplo, ejecute los siguientes comandos mediante la función Trabajos en NetScaler MAS o mediante las API de NITRO:

1 add service s1 172.17.0.10 HTTP 80

```
2 add service s2 172.17.0.11 HTTP 80
3 add lb vserver cpx-vip HTTP 172.17.4.100 80
4 bind lb vserver cpx-vip s1
5 bind lb vserver cpx-vip s2
```

Configure una dirección IP pública adicional para el host de Docker y una regla NAT ejecutando los siguientes comandos en el símbolo del shell de Linux:

```
ip addr add 10.102.29.103/24 dev eth0
iptables -t nat -A PREROUTING -p ip -d 10.102.29.103 -j DNAT --to-
destination 172.17.4.100
```

Método 2: utilizar la dirección IP de NetScaler CPX para el VIP y configurar la asignación de puertos:

- 1. Configure la VIP y los dos servicios en la instancia de NetScaler CPX. Use un puerto no estándar, 81, con el VIP.
- 2. Vincula los servicios al VIP.
- 3. Configure una regla NAT para reenviar todo el tráfico recibido en el puerto 50000 del host Docker a la VIP y al puerto 81.

Para configurar este caso de ejemplo, ejecute el siguiente comando en el símbolo del shell de Linux mientras crea el contenedor NetScaler CPX en los tres hosts de Docker:

Después de aprovisionar la instancia de NetScaler CPX, ejecute los siguientes comandos mediante la función Trabajos en NetScaler MAS o mediante las API de NITRO:

```
    add service s1 172.17.0.10 http 80
    add service s2 172.17.0.11 http 80
    add lb vserver cpx-vip HTTP 172.17.0.4 81
    bind lb vserver cpx-vip s1
    bind lb vserver cpx-vip s2
```

Nota:

Si no ha configurado la asignación de puertos durante el aprovisionamiento de la instancia de NetScaler CPX, configure una regla NAT ejecutando los siguientes comandos en el símbolo del shell de Linux:

iptables -t nat -A PREROUTING -p tcp -m addrtype –dst-type LOCAL -m tcp –dport 50000 -j DNAT –to-destination 172.17.0.4:81

Topología 2: NetScaler CPX con servidores físicos y cliente

En esta topología, solo la instancia de NetScaler CPX se aprovisiona en un host Docker. El cliente y los servidores no están basados en contenedores y residen en otra parte de la red.

En este entorno, puede configurar la instancia de NetScaler CPX para equilibrar la carga del tráfico en los servidores físicos.

En esta ilustración se ilustra esta topología.



En este ejemplo, el contenedor NetScaler CPX (172.17.0.4) se encuentra entre el cliente y los servidores físicos que actúan como proxy. Los servidores, DB1 (10.102.29.105) y DB2 (10.102.29.110), residen fuera de un host Docker en la red. La solicitud del cliente se origina en Internet y se recibe en NetScaler CPX, que la distribuye en los dos servidores.

Para habilitar esta comunicación entre el cliente y los servidores a través de NetScaler CPX, primero debe configurar la asignación de puertos al crear el contenedor NetScaler CPX. A continuación, configure los dos servicios en el contenedor NetScaler CPX para que representen los dos servidores. Y, por último, configure un servidor virtual mediante la dirección IP de NetScaler CPX y el puerto HTTP asignado no estándar 8080.

Tenga en cuenta que en la configuración de ejemplo, la red 10.x.x.x indica una red pública.

Para configurar este caso de ejemplo, ejecute el siguiente comando en el símbolo del shell de Linux mientras crea el contenedor NetScaler CPX:

docker run -dt -p 22 -p 80 -p 161/udp -p 8080:8080 --ulimit core=-1 --privileged=**true** cpx:6.2

1

A continuación, ejecute los siguientes comandos mediante la función Trabajos en NetScaler MAS o mediante las API de NITRO:

```
    add service s1 HTTP 10.102.29.105 80
    add service s2 HTTP 10.102.29.110 80
    add lb vserver cpx-vip HTTP 172.17.0.4 8080
    bind lb vserver cpx-vip s1
    bind lb vserver cpx-vip s2
```

Topología 3: NetScaler CPX y servidores aprovisionados en diferentes hosts

En esta topología, la instancia de NetScaler CPX y los servidores de base de datos se aprovisionan en diferentes hosts de Docker y el tráfico del cliente se origina en Internet. Esta topología se puede utilizar en una implementación de producción en la que la instancia de NetScaler CPX equilibra la carga de un conjunto de aplicaciones o servidores basados en contenedores.

El siguiente diagrama ilustra esta topología.



En este ejemplo, la instancia de NetScaler CPX y un servidor (DB1) se aprovisionan en el mismo host de Docker con la dirección IP 10.102.29.100. Otros cuatro servidores (DB2, DB3, DB4 y DB5) se aprovisionan en dos hosts Docker diferentes, 10.102.29.105 y 10.102.29.110.

Las solicitudes de clientes que se originan en Internet se reciben en la instancia de NetScaler CPX,

que luego distribuye las solicitudes en los cinco servidores. Para habilitar esta comunicación, debe configurar lo siguiente:

- Configure la asignación de puertos al crear su contenedor NetScaler CPX. En este ejemplo, esto significa que debe reenviar el puerto 8080 del contenedor al puerto 8080 del host. Cuando la solicitud del cliente llega al puerto 8080 del host, se asigna al puerto 8080 del contenedor CPX.
- 2. Configure los cinco servidores como servicios en la instancia de NetScaler CPX. Debe usar una combinación de la dirección IP del host Docker respectivo y el puerto asignado para configurar estos servicios.
- 3. Configure una VIP en la instancia de NetScaler CPX para recibir la solicitud del cliente. Este VIP debe estar representado por la dirección IP de NetScaler CPX y el puerto 8080 que se asignó al puerto 8080 del host.
- 4. Por último, vincule los servicios al VIP.

Tenga en cuenta que en la configuración de ejemplo, la red 10.x.x.x indica una red pública.

Para configurar este caso de ejemplo, ejecute el siguiente comando en el símbolo del shell de Linux mientras crea el contenedor NetScaler CPX:

Ejecute los siguientes comandos mediante la función Trabajos en NetScaler MAS o mediante las API de NITRO:

1	add service s1 10.102.29.100 HTTP 8081
2	add service s2 10.102.29.105 HTTP 8081
3	add service s3 10.102.29.105 HTTP 8082
4	add service s4 10.102.29.110 HTTP 8081
5	add service s5 10.102.29.110 HTTP 8082
6	add lb vserver cpx-vip HTTP 172.17.0.2 8080
7	bind lb vserver cpx-vip s1
8	bind lb vserver cpx-vip s2
9	bind lb vserver cpx-vip s3
10	bind lb vserver cpx-vip s4
11	bind lb vserver cpx-vip s5

Implementar NetScaler CPX con acceso directo a la red

November 23, 2023

En el modo de red puente, puede configurar la instancia de NetScaler CPX para que tenga acceso directo a la red. En este caso, el tráfico entrante se recibe directamente en la IP (VIP) del servidor virtual NetScaler CPX.

Para habilitar esta comunicación, primero debe configurar una dirección IP pública en el puente docker0. A continuación, elimine la dirección IP pública del puerto de red eth0 y vincule el puerto de red al puente docker0.

Configure el equilibrio de carga agregando los dos servicios y, a continuación, configure una dirección IP pública de red como VIP en la instancia de NetScaler CPX. Las solicitudes de los clientes se reciben directamente en el VIP.

En la configuración de ejemplo, la red 10.x.x.x indica una red pública.

Para configurar este caso, ejecute el siguiente comando en el símbolo del shell de Linux:

```
ip addr add 10.102.29.100/24 dev docker0;
ip addr del 10.102.29.100/24 dev eth0;
brctl addif docker0 eth0;
ip route del default;
ip route add default via 10.102.29.1 dev docker0
```

Ya sea mediante la función Trabajos en NetScaler MAS o mediante las API de NITRO, ejecute los siguientes comandos:

```
    add service s1 172.17.0.8 http 80
    add service s2 172.17.0.9 http 80
    add lb vserver cpx-vip HTTP 10.102.29.102 80
    bind lb vserver cpx-vip s1
    bind lb vserver cpx-vip s2
```

Configurar NetScaler CPX en Kubernetes mediante ConfigMaps

November 23, 2023

En Kubernetes, puede configurar la instancia de NetScaler CPX mediante ConfigMaps. Con ConfigMaps, puede configurar dinámicamente la instancia de NetScaler CPX durante el inicio de la instancia.

Cree un archivo de configuración cpx.conf que incluya la configuración específica de NetScaler y los comandos bash shell que quiera ejecutar dinámicamente en la instancia de NetScaler CPX. La estructura del archivo de configuración requiere dos tipos de etiquetas, #NetScaler Commands y #Shell Commands. Debajo de la etiqueta #NetScaler Commands, debe agregar todos los comandos de NetScaler para configurar la configuración específica de NetScaler en la instancia de NetScaler CPX. Debajo de la etiqueta #Shell Commands, debe agregar los comandos de shell que quiere ejecutar en la instancia de NetScaler CPX.

Importante:

- Las etiquetas se pueden repetir varias veces en el archivo de configuración.
- El archivo de configuración también puede incluir comentarios. Agregue un carácter "#" antes de los comentarios.
- Las etiquetas no distinguen entre mayúsculas y minúsculas.
- Si hay casos de error al implementar el contenedor NetScaler CPX con el archivo de configuración, los errores se registran en el archivo ns.log.
- Una vez iniciada la instancia de NetScaler CPX, si cambia el ConfigMap, la configuración actualizada se aplica solo cuando se reinicia la instancia de NetScaler CPX.

A continuación se muestra un archivo de configuración de ejemplo:

```
1 #NetScaler Commands
2 add lb vserver v1 http 1.1.1.1 80
3 add service s1 2.2.2.2 http 80
4 bind lb vserver v1 s1
5 #Shell Commands
6 touch /etc/a.txt
7 echo "this is a" > /etc/a.txt
8 #NetScaler Commands
9 add lb vserver v2 http
10 #Shell Commands
11 echo "this is a 1" >> /etc/a.txt
12 #NetScaler Commands
13 add lb vserver v3 http
```

Una vez que haya creado el archivo de configuración, debe crear un ConfigMap a partir del archivo de configuración mediante el comando kubectl create configmap.

1 kubectl create configmap cpx-config --from-file=cpx.conf

En el ejemplo anterior, puede crear un ConfigMap, cpx-config basado en el archivo de configuración cpx.conf. A continuación, puede usar este ConfigMap en el archivo YAML que se utiliza para implementar la instancia de NetScaler CPX.

Puede ver el ConfigMap creado mediante el comando kubectl get configmap.
root@node1:~/yaml# kubectl get configmap cpx-config -o yaml

Muestra:

```
1 apiVersion: v1
2 data:
3 cpx.conf: |
4 #NetScaler Commands
5 add lb vserver v1 http 1.1.1.1 80
6 add service s1 2.2.2.2 http 80
7 bind lb vserver v1 s1
8 #Shell Commands
```

```
9
               touch /etc/a.txt
10
                echo "this is a" > /etc/a.txt
                echo "this is the file" >> /etc/a.txt
11
               ls >> /etc/a.txt
12
           #NetScaler Commands
13
14
               add lb vserver v2 http
           #Shell Commands
15
               echo "this is a 1" >> /etc/a.txt
16
17
           #NetScaler Commands
18
               add lb vserver v3 http
19
           #end of file
       kind: ConfigMap
20
21
       metadata:
           creationTimestamp: 2017-12-26T06:26:50Z
23
           name: cpx-config
24
           namespace: default
25
           resourceVersion: "8865149"
26
           selfLink: /api/v1/namespaces/default/configmaps/cpx-config
27
           uid: c1c7cb5b-ea05-11e7-914a-926745c10b02
```

Puede especificar el ConfigMap creado, cpx-config en el archivo YAML utilizado para implementar la instancia de NetScaler CPX, como se muestra a continuación:

```
1 apiVersion: v1
2 kind: Pod
3 metadata:
4 name: cpx-1
5 labels:
6
       app: cpx-daemon
7
     annotations:
       NETSCALER_AS_APP: "True"
8
9 spec:
10 hostNetwork: true
11
   containers:
12
     - name: cpx
       image: "quay.io/citrix/citrix-k8s-cpx-ingress:12.1-51.16"
13
14
       securityContext:
15
         privileged: true
16
       volumeMounts:
17
       - name: config-volume
18
        mountPath: /cpx/conf
19
       env:
         - name: "EULA"
          value: "yes"
21
         - name: "NS_NETMODE"
          value: "HOST"
23
24
         - name: "kubernetes_url"
25
          value: "https://10.90.248.101:6443"
         - name: "NS_MGMT_SERVER"
26
27
          value: "10.90.248.99"
28
         - name: "NS_MGMT_FINGER_PRINT"
           value: "19:71:A3:36:85:0A:2B:62:24:65:0F:7E:72:CC:DC:AD:B8:BF
29
               :53:1E"
```

```
- name: "NS_ROUTABLE"
31
           value: "FALSE"
32
         - name: "KUBERNETES_TASK_ID"
33
           valueFrom:
34
              fieldRef:
35
                fieldPath: metadata.name
       imagePullPolicy: Never
37
     volumes:
     - name: config-volume
38
39
       configMap:
40
         name: cpx-config
```

Una vez que se implementa la instancia de NetScaler CPX is e inicia la configuración especificada en ConfigMap, cpx-config se aplica a la instancia de NetScaler CPX.

Implementar el proxy de NetScaler CPX en Google Compute Engine

November 23, 2023

En esta guía de implementación, se describe cómo puede implementar NetScaler CPX con Docker en Google Compute Engine (GCE) de Google Cloud con NetScaler MAS ejecutándose dentro de la red empresarial. En esta implementación, NetScaler CPX instalado en GCE equilibra la carga de dos servidores back-end, y NetScaler MAS proporciona soluciones de licencias y análisis.

NetScaler CPX es un proxy basado en contenedores que admite la funcionalidad completa de capa 7, la descarga de SSL, varios protocolos y la API de NITRO. NetScaler MAS proporciona soluciones de administración, licencias y análisis. Como servidor de licencias, NetScaler MAS proporciona derechos a instancias de NetScaler CPX que se ejecutan en las instalaciones o en la nube.

CPX y CPX Express son las mismas imágenes. Cuando NetScaler MAS se licencia y se instala, la imagen CPX en Docker App Store (versión 11 o 12) se convierte en una instancia CPX completa. Sin licencia, la imagen CPX se convierte en una instancia CPX Express que admite conexiones SSL de 20 Mbps y 250.

Requisitos previos

- 2 GB de memoria y 1 vCPU dedicada a NetScaler CPX
- Código abierto de Docker disponible en GCE
- NetScaler MAS ejecutándose en las instalaciones con conexión a Internet o VPN a GCE

Nota

Para obtener información sobre cómo implementar NetScaler MAS, consulte Implementación de NetScaler MAS.

Pasos de configuración

Debe realizar los siguientes pasos para configurar esta implementación.

- 1. Instala Docker en una VM de GCE.
- 2. Configure la comunicación de API remota con la instancia de Docker.
- 3. Instale la imagen de NetScaler CPX.
- 4. Cree una instancia CPX.
- 5. Licencia de NetScaler CPX a través de NetScaler MAS.
- 6. Configure los servicios de equilibrio de carga en NetScaler CPX y verifique la configuración.
 - a) Instale los servidores web NGINX.
 - b) Configure CPX para el equilibrio de carga y verifique la distribución de la carga a ambos servicios web.

Paso 1: Instalar Docker en una VM de GCE

Desde GCE, crea una VM Linux Ubuntu. A continuación, instale Docker en la VM mediante los comandos que se muestran en el siguiente ejemplo:

```
1 $ sudo curl - ssl https://get.docker.com/ | sh
2 % Total % Received % Xferd Average Speed Time Time Time Current
3 Dload Upload Total Spent Left Speed
4 0 0 0 0 0 0 0 0 --:--:-- --:-- 0curl: (6) Could not resolve
       host: xn--ssl-1n0a
5 100 17409 100 17409 0 0 21510 0 --:--:-- --:-- 21492
6 apparmor is enabled in the kernel and apparmor utils were already
      installed
7 + sudo -E sh -c apt-key add -
8 + echo -----BEGIN PGP PUBLIC KEY BLOCK-----
9 Version: GnuPG v1
10
11 mQINBFWln24BEADrBl5p99uKh8+rpvqJ48u4eTtjeXAWbslJotmC/CakbNSqOb9o
12 ddfzRvGVeJVERt/Q/mlvEqgnyTQy+e6oEYN2Y2kqXceUhXagThnqCoxcEJ3+KM4R
13 mYdoe/BJ/J/6rH0jq70mk24z2qB3RU1uAv57iY5VGw5p45uZB4C4pNNsBJXoCvPn
14 TGAs/7IrekFZDDgVraPx/hdiwopQ8NltSfZCyu/jPpWFK28TR8yfVlzYFwibj5WK
15 dHM7ZTqlA1tHIG+agyPf3Rae0jPMsHR6q+arXVwMccy0i+ULU0z8mHUJ3iEMIrpT
16 X+80KaN/ZjibfsBOCjcfiJSB/acn4nxQQgNZigna32velafhQivsNREFeJpzENiG
17 HOoyC6qVe0gKrRiKxzymj0FIMLru/iFF5pSWcBQB7PYlt8J0G80lAcPr6VCiN+4c
```

```
18 NKv03SdvA69dC0j79Pu09IIvQsJXsSq96HB+TeEmmL+xSdpGtGdCJHHM1fDeCqkZ
19 hT+RtBGQL2SEdWjxbF43oQopocT8cHvyX6Zaltn0svoGs+wX3Z/H6/8P5anog43U
20 65c0A+64Jj00rNDr8j31izhtQMRo892kGeQAaaxg4Pz6HnS7hRC+cOMHUU4HA7iM
21 zHrouAdYeTZeZEQ0A7SxtCME9ZnGwe2grxPXh/U/80WJGkzLFNcTKdv+rwARAQAB
22 tDdEb2NrZXIgUmVsZWFzZSBUb29sIChyZWxlYXNlZG9ja2VyKSA8ZG9ja2VyQGRv
23
   Y2tlci5jb20+iQIcBBABCgAGBQJWw7vdAAoJEFyzYeVS+w0QHysP/i37m4SyoOCV
   cnybl18vzwBEcp4VCRbXvHv0Xty1gccVIV8/aJqNKgBV97lY3vrp0yiIeB8ETQeg
24
   srxFE7t/Gz0rsLObqfLEHdmn5iBJRkhLfCpzjeOnyB3Z0IJB6Uog0/msQVYe5CXJ
25
   l6uwr0AmoiCBLrVlDAktxVh9RWch0l0KZRX2FpHu8h+uM0/zySqIidlYfLa3y5oH
26
27
   scU+nGU1i6ImwDTD3ysZC5jp9aVfvUmcESyAb4vvdcAHR+bXhA/RW8QHeeMFliWw
28
   7Z2jYHyuHmDnWG2yUrnCqAJTrWV+0fKRIzzJFBs4e88ru5h2ZIXdRepw/+COYj34
   LyzxR2cxr2u/xvxwXCkSMe7F4KZAphD+1ws61FhnUMi/PERMYfTFuvPrCkq4gyBj
29
30 t3fFpZ2NR/fKW87QOeVcn1ivXl9id3MMs9KXJsg7QasT7mCsee2VIFsxrkFQ2jNp
   D+JAERRn9Fj4ArHL5TbwkkFbZZvSi6fr5h2GbCAXIGhIXKnjjorPY/YDX6X8AaH0
32
   W1zblWy/CFr6VFl963jrjJgag0G6tNtBZLrclZgWh0QpeZZ5Lbvz2ZA5CqRrfAVc
33
   wPNW1f0bFIRtqV6vuVluF0PCMAAnOnqR02w9t17iVQj03oVN0mbQi9vjuExXh1Yo
   ScVeti06LSmlQfVEVRTqHLMgXyR/EMo7iQIcBBABCgAGBQJXSWBlAAoJEFyzYeVS
34
   +w0QeH0QAI6btAfYwYPuAjfRUy9qlnPhZ+xt1rnwsUzsbmo8K3XTNh+l/R08nu0d
   sczw30Q1wju28fh1N8ay223+69f0+yICaXqR18AbGgFGKX7vo0gfEVaxdItUN3eH
37
   NydGFzmeOKbAlrxIMECnSTG/TkFVYO9Ntlv9vSN2BupmTagTRErxLZKnVsWRzp+X
38
   -----END PGP PUBLIC KEY BLOCK-----
39
40
41
   OK
   + sudo -E sh -c mkdir -p /etc/apt/sources.list.d
42
43 + dpkg --print-architecture
44 + sudo -E sh -c echo deb \\[arch=amd64\\] https://apt.dockerproject.org
       /repo ubuntu-yakkety main > /etc/apt/sources.list.d/docker.list
45 + sudo -E sh -c sleep 3; apt-get update; apt-get install -y -q docker-
      engine
46 Hit:1 http://us-west1.gce.archive.ubuntu.com/ubuntu yakkety InRelease
   Get:2 http://us-west1.gce.archive.ubuntu.com/ubuntu yakkety-updates
47
      InRelease [102 kB]
48
   Get:3 http://us-west1.gce.archive.ubuntu.com/ubuntu yakkety-backports
      InRelease [102 kB]
   Get:4 http://us-west1.gce.archive.ubuntu.com/ubuntu yakkety/restricted
49
       Sources [5,376 B]
50 Get:5 http://us-west1.gce.archive.ubuntu.com/ubuntu yakkety/multiverse
       Sources [181 kB]
51 Get:6 http://us-west1.gce.archive.ubuntu.com/ubuntu yakkety/universe
       Sources [8,044 kB]
52 Get:7 http://archive.canonical.com/ubuntu yakkety InRelease [11.5 kB]
53 Get:8 http://security.ubuntu.com/ubuntu yakkety-security InRelease [102
       kB]
54 Get:9 https://apt.dockerproject.org/repo ubuntu-yakkety InRelease [47.3
       kB1
   Get:10 http://us-west1.gce.archive.ubuntu.com/ubuntu yakkety/main
      Sources [903 kB]
56 Get:11 http://us-west1.gce.archive.ubuntu.com/ubuntu yakkety-updates/
      restricted Sources [2,688 B]
  Get:12 http://us-west1.gce.archive.ubuntu.com/ubuntu yakkety-updates/
57
      universe Sources [57.9 kB]
58 Get:13 http://us-west1.gce.archive.ubuntu.com/ubuntu yakkety-updates/
```

multiverse Sources [3,172 B] 59 Get:14 http://us-west1.gce.archive.ubuntu.com/ubuntu yakkety-updates/ main Sources [107 kB] 60 Get:15 http://us-west1.gce.archive.ubuntu.com/ubuntu yakkety-updates/ main amd64 Packages [268 kB] 61 Get:16 http://us-west1.gce.archive.ubuntu.com/ubuntu yakkety-updates/ main Translation-en [122 kB] 62 Get:17 http://us-west1.gce.archive.ubuntu.com/ubuntu yakkety-updates/ universe amd64 Packages [164 kB] 63 Get:18 http://us-west1.gce.archive.ubuntu.com/ubuntu yakkety-updates/ universe Translation-en [92.4 kB] 64 Get:19 http://us-west1.gce.archive.ubuntu.com/ubuntu yakkety-updates/ multiverse amd64 Packages [4,840 B] 65 Get:20 http://us-west1.gce.archive.ubuntu.com/ubuntu yakkety-updates/ multiverse Translation-en [2,708 B] 66 Get:21 http://us-west1.gce.archive.ubuntu.com/ubuntu yakkety-backports/ universe Sources [2,468 B] 67 Get:22 http://us-west1.gce.archive.ubuntu.com/ubuntu yakkety-backports/ main Sources [2,480 B] 68 Get:23 http://us-west1.gce.archive.ubuntu.com/ubuntu yakkety-backports/ main amd64 Packages [3,500 B] 69 Get:24 http://us-west1.gce.archive.ubuntu.com/ubuntu yakkety-backports/ universe amd64 Packages [3,820 B] 70 Get:25 http://us-west1.gce.archive.ubuntu.com/ubuntu yakkety-backports/ universe Translation-en [1,592 B] 71 Get:26 http://archive.canonical.com/ubuntu yakkety/partner amd64 Packages [2,480 B] 72 Get:27 http://security.ubuntu.com/ubuntu yakkety-security/main Sources [47.7 kB] 73 Get:28 https://apt.dockerproject.org/repo ubuntu-yakkety/main amd64 Packages [2,453 B] 74 Get:29 http://security.ubuntu.com/ubuntu yakkety-security/universe Sources [20.7 kB] 75 Get:30 http://security.ubuntu.com/ubuntu yakkety-security/multiverse Sources [1,140 B] 76 Get:31 http://security.ubuntu.com/ubuntu yakkety-security/restricted Sources [2,292 B] 77 Get:32 http://security.ubuntu.com/ubuntu yakkety-security/main amd64 Packages [150 kB] 78 Get:33 http://security.ubuntu.com/ubuntu yakkety-security/main Translation-en [68.0 kB] 79 Get:34 http://security.ubuntu.com/ubuntu yakkety-security/universe amd64 Packages [77.2 kB] 80 Get:35 http://security.ubuntu.com/ubuntu yakkety-security/universe Translation-en [47.3 kB] 81 Get:36 http://security.ubuntu.com/ubuntu yakkety-security/multiverse amd64 Packages [2,832 B] 82 Fetched 10.8 MB in 2s (4,206 kB/s) 83 Reading **package** lists... Done 84 Reading **package** lists... 85 Building dependency tree... 86 Reading state information... 87 The following additional packages will be installed:

```
88 aufs-tools cgroupfs-mount libltdl7
89 The following NEW packages will be installed:
90 aufs-tools cgroupfs-mount docker-engine libltdl7
91 0 upgraded, 4 newly installed, 0 to remove and 37 not upgraded.
92 Need to get 21.2 MB of archives.
93 After this operation, 111 MB of additional disk space will be used.
94 Get:1 http://us-west1.gce.archive.ubuntu.com/ubuntu yakkety/universe
       amd64 aufs-tools amd64 1:3.2+20130722-1.1ubuntu1 [92.9 kB]
95 Get:2 http://us-west1.gce.archive.ubuntu.com/ubuntu yakkety/universe
       amd64 cgroupfs-mount all 1.3 [5,778 B]
96 Get:3 http://us-west1.gce.archive.ubuntu.com/ubuntu yakkety/main amd64
       libltdl7 amd64 2.4.6-1 [38.6 kB]
97 Get:4 https://apt.dockerproject.org/repo ubuntu-yakkety/main amd64
       docker-engine amd64 17.05.0~ce-0~ubuntu-yakkety [21.1 MB]
98 Fetched 21.2 MB in 1s (19.8 MB/s)
99 Selecting previously unselected package aufs-tools.
100 (Reading database ... 63593 files and directories currently installed.)
101 Preparing to unpack .../aufs-tools_1%3a3.2+20130722-1.1ubuntu1_amd64.
       deb ...
102 Unpacking aufs-tools (1:3.2+20130722-1.1ubuntu1) ...
103 Selecting previously unselected package cgroupfs-mount.
104 Preparing to unpack .../cgroupfs-mount_1.3_all.deb ...
105 Unpacking cgroupfs-mount (1.3) ...
106 Selecting previously unselected package libltdl7:amd64.
107 Preparing to unpack .../libltdl7_2.4.6-1_amd64.deb ...
108 Unpacking libltdl7:amd64 (2.4.6-1) ...
109 Selecting previously unselected package docker-engine.
110 Preparing to unpack .../docker-engine_17.05.0~ce-0~ubuntu-yakkety_amd64
       .deb ...
111 Unpacking docker-engine (17.05.0~ce-0~ubuntu-yakkety) ...
112 Setting up aufs-tools (1:3.2+20130722-1.1ubuntu1) ...
113 Processing triggers for ureadahead (0.100.0-19) ...
114 Setting up cgroupfs-mount (1.3) ...
115 Processing triggers for libc-bin (2.24-3ubuntu2) ...
116 Processing triggers for systemd (231-9ubuntu4) ...
117 Setting up libltdl7:amd64 (2.4.6-1) ...
118 Processing triggers for man-db (2.7.5-1) ...
119 Setting up docker-engine (17.05.0~ce-0~ubuntu-yakkety) ...
120 Created symlink /etc/system/system/multi-user.target.wants/docker.
       service > /lib/systemd/system/docker.service.
121 Created symlink /etc/systemd/system/sockets.target.wants/docker.socket
       → /lib/systemd/system/docker.socket.
122 Processing triggers for ureadahead (0.100.0-19) ...
123 Processing triggers for libc-bin (2.24-3ubuntu2) ...
124 Processing triggers for systemd (231-9ubuntu4) ...
125 + sudo -E sh -c docker version
126 Client:
127 Version: 17.05.0-ce
128 API version: 1.29
129 Go version: go1.7.5
130 Git commit: 89658be
131 Built: Thu May 4 22:15:36 2017
132 OS/Arch: linux/amd64
```

```
133
134 Server:
135 Version: 17.05.0-ce
136 API version: 1.29 (minimum version 1.12)
137 Go version: go1.7.5
138 Git commit: 89658be
139 Built: Thu May 4 22:15:36 2017
140 OS/Arch: linux/amd64
141 Experimental: false
142
143 If you would like to use Docker as a non-root user, you should now
       consider
144 adding your user to the "docker" group with something like:
145
146 sudo usermod -aG docker albert_lee
147
148 Remember that you will have to log out and back in for this to take
       effect.
149
150 WARNING: Adding a user to the "docker" group will grant the ability to
       run
151 containers which can be used to obtain root privileges on the
152 docker host.
153 Refer to https://docs.docker.com/engine/security/security/#docker-
       daemon-attack-surface
154 for more information.
155
156 $
157
158 \*\*$ sudo docker info\*\*
159 Containers: 0
160 Running: 0
161 Paused: 0
162 Stopped: 0
163 Images: 0
164 Server Version: 17.05.0-ce
165 Storage Driver: aufs
166 Root Dir: /var/lib/docker/aufs
167 Backing Filesystem: extfs
168 Dirs: 0
169 Dirperm1 Supported: true
170 Logging Driver: json-file
171 Cgroup Driver: cgroupfs
172 Plugins:
173 Volume: local
174 Network: bridge host macvlan null overlay
175 Swarm: inactive
176 Runtimes: runc
177 Default Runtime: runc
178 Init Binary: docker-init
179 containerd version: 9048e5e50717ea4497b757314bad98ea3763c145
180 runc version: 9c2d8d184e5da67c95d601382adf14862e4f2228
181 init version: 949e6fa
```

```
182 Security Options:
183 apparmor
184 seccomp
185 Profile: default
186 Kernel Version: 4.8.0-51-generic
187 Operating System: Ubuntu 16.10
188 OSType: linux
189 Architecture: x86_64
190 CPUs: 1
191 Total Memory: 3.613GiB
192 Name: docker-7
193 ID: R5TW:VKXK:EKGR:GHWM:UNU4:LPJH:IQY5:X77G:NNRQ:HWBY:LIUD:4ELQ
194 Docker Root Dir: /var/lib/docker
195 Debug Mode (client): false
196 Debug Mode (server): false
197 Registry: https://index.docker.io/v1/
198 Experimental: false
199 Insecure Registries:
200 127.0.0.0/8
201 Live Restore Enabled: false
203 WARNING: No swap limit support
204 $
205
206 \*\*$ sudo docker images\*\*
207 REPOSITORY TAG IMAGE ID CREATED SIZE
208 $
209
210 \*\*$ sudo docker ps\*\*
211 CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
212 $
```

Paso 2: Configurar la comunicación de API remota con la instancia de Docker

Abra el puerto 4243 para la comunicación de la API con la instancia de Docker. Esto es necesario para que NetScaler MAS se comunique con la instancia de Docker.

```
1 $
2 $ \*\*cd /etc/systemd/system\*\*
3 $ \*\*sudo vi docker-tcp.socket\*\*
4 $
5
6 $ \*\*cat docker-tcp.socket\*\*
7 [Unit]
8 \*\*Description=Docker Socket for the API
9 [Socket]
10 ListenStream=4243
11 BindIPv6Only=both
12 Service=docker.service
13 [Install]
14 WantedBy=sockets.target\*\*
```

```
15 $
16
17 $ \*\*sudo systemctl enable docker-tcp.socket\*\*
18 Created symlink /etc/system/sockets.target.wants/docker-tcp.
       socket → /etc/systemd/system/docker-tcp.socket.
19 $ \*\*sudo systemctl enable docker.socket\*\*
20 $ \*\*sudo systemctl stop docker\*\*
21 $ \*\*sudo systemctl start docker-tcp.socket\*\*
22 $ \*\*sudo systemctl start docker\*\*
23 $ \*\*sudo systemctl status docker\*\*
24 ● docker.service - Docker Application Container Engine
25 Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor
       preset: enabled)
26 Active: \*\*active (running)\*\* since Wed 2017-05-31 12:52:17 UTC; 2s
       ago
27 Docs: https://docs.docker.com
28 Main PID: 4133 (dockerd)
29 Tasks: 16 (limit: 4915)
30 Memory: 30.1M
31 CPU: 184ms
32 CGroup: /system.slice/docker.service
33 -4133 /usr/bin/dockerd -H fd://
   4137 docker-containerd -l unix:///var/run/docker/libcontainerd/docker
34
       -containerd.sock --metrics-interval=0 --start-timeout 2m -
36 May 31 12:52:17 docker-7 dockerd[4133]: time="2017-05-31T12
       :52:17.300890402Z" level=warning msg="Your kernel does not support
       cgroup rt peri
37 May 31 12:52:17 docker-7 dockerd[4133]: time="2017-05-31T12
       :52:17.301079754Z" level=warning msg="Your kernel does not support
       cgroup rt runt
38 May 31 12:52:17 docker-7 dockerd[4133]: time="2017-05-31T12
       :52:17.301681794Z" level=info msg="Loading containers: start."
39 May 31 12:52:17 docker-7 dockerd[4133]: time="2017-05-31T12"
       :52:17.417539064Z" level=info msg="Default bridge (docker0) is
      assigned with an I
40 May 31 12:52:17 docker-7 dockerd[4133]: time="2017-05-31T12
       :52:17.465011600Z" level=info msg="Loading containers: done."
41 May 31 12:52:17 docker-7 dockerd[4133]: time="2017-05-31T12
       :52:17.484747909Z" level=info msg="Daemon has completed
       initialization"
42 May 31 12:52:17 docker-7 dockerd[4133]: time="2017-05-31T12
       :52:17.485119478Z" level=info msg="Docker daemon" commit=89658be
      graphdriver=aufs
43 May 31 12:52:17 docker-7 systemd[1]: Started Docker Application
       Container Engine.
44 May 31 12:52:17 docker-7 dockerd[4133]: time="2017-05-31T12
       :52:17.503832254Z" level=info msg="API listen on /var/run/docker.
       sock"
45 May 31 12:52:17 docker-7 dockerd[4133]: time="2017-05-31T12
       :52:17.504061522Z" level=info msg="API listen on [::]:4243"
46 $
47
```

```
48 (external)$ \*\*curl 104.199.209.157:4243/version\*\*
49 {
50 "Version":"17.05.0-ce","ApiVersion":"1.29","MinAPIVersion":"1.12","
    GitCommit":"89658be","GoVersion":"go1.7.5","0s":"linux","Arch":"
    amd64","KernelVersion":"4.8.0-52-generic","BuildTime":"2017-05-04
    T22:15:36.071254972+00:00" }
51
52 (external)$
```

Paso 3: instalar NetScaler CPX Image

Obtenga la imagen de NetScaler CPX en Docker App Store. Tenga en cuenta que el CPX Express y el CPX tienen la misma imagen. Cuando NetScaler MAS se licencia e instala, la imagen se convierte en una instancia CPX completa con un rendimiento de 1 Gbps. Sin licencia, la imagen se convierte en una instancia CPX Express que admite conexiones SSL de 20 Mbps y 250.

```
1 $ \*\*sudo docker pull store/citrix/netscalercpx:12.0-41.16\*\*
2 12.0-41.16: Pulling from store/citrix/netscalercpx
3 4e1f679e8ab4: Pull complete
4 a3ed95caeb02: Pull complete
5 2931a926d44b: Pull complete
6 362cd40c5745: Pull complete
7 d10118725a7a: Pull complete
8 1e570419a7e5: Pull complete
9 d19e06114233: Pull complete
10 d3230f008ffd: Pull complete
11 22bdb10a70ec: Pull complete
12 1a5183d7324d: Pull complete
13 241868d4ebff: Pull complete
14 3f963e7ae2fc: Pull complete
15 fd254cf1ea7c: Pull complete
16 33689c749176: Pull complete
17 59c27bad28f5: Pull complete
18 588f5003e10f: Pull complete
19 Digest: sha256:31
      a65cfa38833c747721c6fbc142faec6051e5f7b567d8b212d912b69b4f1ebe
20 Status: Downloaded newer image for store/citrix/netscalercpx:12.0-41.16
21 $
23 $ \*\*sudo docker images\*\*
24 REPOSITORY TAG IMAGE ID CREATED SIZE
25 store/citrix/netscalercpx 12.0-41.16 6fa57c38803f 3 weeks ago 415MB
26 $
```

Paso 4: crear una instancia CPX

Instale la imagen CPX en el host de Docker. Abra puertos para servicios específicos, como se muestra en el siguiente ejemplo, y especifique una dirección IP para NetScaler MAS:

```
1 bash-2.05b# \*\*CHOST=${
   1:-localhost }
2
3
    \*\*
4 bash-2.05b# \*\*echo | openssl s_client -connect $CHOST:443 | openssl
      x509 -fingerprint -noout | cut -d'=' -f2\*\*
5 depth=0 C = US, ST = California, L = San Jose, O = Citrix NetScaler, OU
       = Internal, CN = Test Only Cert
6 verify error:num=18:self signed certificate
7 verify return:1
8 depth=0 C = US, ST = California, L = San Jose, O = Citrix NetScaler, OU
       = Internal, CN = Test Only Cert
9 verify return:1
10 DONE
11 24:AA:8B:91:7B:72:5E:6E:C1:FD:86:FA:09:B6:42:49:FC:1E:86:A4
12 bash-2.05b#
13
14 $ \*\*sudo docker run -dt -p 50000:88 -p 5080:80 -p 5022:22 -p 5443:443
       -p 5163:161/udp -e NS_HTTP_PORT=5080 -e NS_HTTPS_PORT=5443 -e
      NS_SSH_PORT=5022 -e NS_SNMP_PORT=5163 -e EULA=yes -e LS_IP=xx.xx.xx.
      xx -e PLATFORM=CP1000 --privileged=true --ulimit core=-1 -e
      NS_MGMT_SERVER=xx.xx.xx.xx:xxxx -e NS_MGMT_FINGER_PRINT=24:AA:8B
      :91:7B:72:5E:6E:C1:FD:86:FA:09:B6:42:49:FC:1E:86:A4 --env
      NS_ROUTABLE=false -- env HOST=104.199.209.157 store/citrix/
      netscalercpx:12.0-41.16\*\*
15 44calc6c0907e17a10ffcb9ffe33cd3e9f71898d8812f816e714821870fa3538
16 $
17
18 $ \*\*sudo docker ps\*\*
19 CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
20 44calc6c0907 store/citrix/netscalercpx:12.0-41.16 "/bin/sh -c 'bash ...
       " 19 seconds ago Up 17 seconds 0.0.0.0:5022->22/tcp,
      0.0.0.0:5080->80/tcp, 0.0.0.0:50000->88/tcp, 0.0.0.0:5163->161/udp,
      0.0.0.0:5443->443/tcp gifted_perlman
21 $
22
23 $ \*\*ssh -p 5022 root@localhost\*\*
24 root@localhost's password:
25 Welcome to nsoslx 1.0 (GNU/Linux 4.8.0-52-generic x86_64)
26
27 * Documentation: https://www.citrix.com/
28 Last login: Mon Jun 5 18:58:51 2017 from xx.xx.xx.xx
29 root@44ca1c6c0907:~#
30 root@44ca1c6c0907:~#
31 root@44ca1c6c0907:~# \*\*cli_script.sh 'show ns ip'\*\*
32 exec: show ns ip
33 Ipaddress Traffic Domain Type Mode Arp Icmp Vserver State
34 ----
35 1) 172.17.0.2 0 NetScaler IP Active Enabled Enabled NA Enabled
36 2) 192.0.0.1 0 SNIP Active Enabled Enabled NA Enabled
37 Done
38 root@44ca1c6c0907:~# \*\*cli_script.sh 'show licenseserver'\*\*
39 exec: show licenseserver
40 1) ServerName: xx.xx.xxPort: 27000 Status: 1 Grace: 0 Gptimeleft: 0
```

```
41 Done
42 root@44ca1c6c0907:~# cli_script.sh 'show capacity'
43 exec: show capacity
44 Actualbandwidth: 1000 Platform: CP1000 Unit: Mbps Maxbandwidth: 3000
      Minbandwidth: 20 Instancecount: 0
45 Done
46 root@44ca1c6c0907:~#
47
49 Chain PREROUTING (policy ACCEPT)
50 target prot opt source destination
51 DOCKER all -- 0.0.0.0/0 0.0.0.0/0 ADDRTYPE match dst-type LOCAL
52
53 Chain INPUT (policy ACCEPT)
54 target prot opt source destination
56 Chain OUTPUT (policy ACCEPT)
57 target prot opt source destination
58 DOCKER all -- 0.0.0.0/0 !127.0.0.0/8 ADDRTYPE match dst-type LOCAL
59
60 Chain POSTROUTING (policy ACCEPT)
61 target prot opt source destination
62 MASQUERADE all -- 172.17.0.0/16 0.0.0.0/0
63 MASQUERADE tcp -- 172.17.0.2 172.17.0.2 tcp dpt:443
64 MASQUERADE udp -- 172.17.0.2 172.17.0.2 udp dpt:161
65 MASQUERADE tcp -- 172.17.0.2 172.17.0.2 tcp dpt:88
66 MASQUERADE tcp -- 172.17.0.2 172.17.0.2 tcp dpt:80
67 MASQUERADE tcp -- 172.17.0.2 172.17.0.2 tcp dpt:22
68
69 Chain DOCKER (2 references)
70 target prot opt source destination
71 RETURN all -- 0.0.0.0/0 0.0.0.0/0
72 DNAT tcp -- 0.0.0.0/0 0.0.0/0 tcp dpt:5443 to:172.17.0.2:443
73 DNAT udp -- 0.0.0.0/0 0.0.0.0/0 udp dpt:5163 to:172.17.0.2:161
74 DNAT tcp -- 0.0.0.0/0 0.0.0.0/0 tcp dpt:50000 to:172.17.0.2:88
75 DNAT tcp -- 0.0.0.0/0 0.0.0.0/0 tcp dpt:5080 to:172.17.0.2:80
76 DNAT tcp -- 0.0.0.0/0 0.0.0.0/0 tcp dpt:5022 to:172.17.0.2:22
77 $
```

Paso 5: Licencia de NetScaler CPX a través de NetScaler MAS

Suponiendo que NetScaler MAS se ejecuta en las instalaciones, debería poder validar que CPX se está comunicando con MAS y enviando información. Las siguientes imágenes muestran a CPX recuperando una licencia de NetScaler MAS.

Citrix NetScaler Management and Analytics System	User Name nsroot Password Log On	

×	Citrix NetS	caler	Management and Analytics System May 31 2017 13:14:20 GMT n:						
Q , Sea	rch here		Networks	> License Settings					
Applicat	tions	>	Linner	in the Dent Cattlines				1	
Networ	rks	\sim	License S	server Port Settings				/	
Di	ashboard			Proxy Server Port	License Server Port		Vendor Daemon Port		
In	istances	>		0	27000		7279		
In	stance Groups								
Li	censes	\sim	License F	iles					
	System Licenses		The follow	ing license files are present on this server. Sele	ect Add New License to upload more licenses. To del	ete a license, select the lice	nse and click Delete .		
	Third Party Licenses								
	CPX Licenses		Add Lice	Apply Licenses Delete	e Download				
Ev	vents	>		Name		Last Modified	Size		
SS	SL Dashboard	>		FID3bf0b423_15ba7640cc6_2664.lic		2017-05-23 21:30:15	1.10 KB		
Co	onfiguration Jobs	>		FID3bf0b423_15ba7640cc6_2672.lic		1.10 KB			
Co	onfiguration Audit	>		FID2a2386a8_15b93284902_487e.lic		1.10 KB			
Si	ites	>		FID2a2386a8_15b93284902_4878.lic		2017-05-23 21:30:15	1.10 KB		
N	etwork Functions	>		FID3bf0b423_15ba7640cc6_5281.lic		2017-05-23 21:30:15	1.10 KB		
N	etwork Reporting	Ś		FID_28238688_15D93284902_4870.llc		2017-05-23 21:30:15	1.10 KB		
Analytic		ĺ.		FID		2017-05-23 21:30:15	1.10 KB		
Oreheet		(FID_3bf0b423_15ba7640cc6_5275 lic		2017-05-23 21:30:15	1.10 KB		
Custom	tration	(FID 2a2386a8 15b93284902 4864.lic		2017-05-23 21:30:15	1.10 KB		
System	- 4-								
Downlo	bads		License E	expiry Information					
			Feature		Count	Days To Expiry			
					No items				
×	Citrix NetS	Scale	r Manag	ement and Analytics Sys	tem		Jun 05 2017 15:09:41 GMT	nsroot 🗸	

Q Search here		Networks > Instances > NetScaler CPX									
Applications	>	NotSc	NetScaler CBX								
Networks	\sim	NELSC									
Dashboard		Instances 3 Docker Host 0 Add Provision Delete Dashboard Reboot Action • Search • & •									
Instances NetScaler MPX	~								:h - o -		
NetScaler VPX			IP Address	Host Name	State	Docker Host	Port Range	SSH Port	HTTP Port	HTTPS Port	SNMP Port
NetScaler SDX			172.17.0.2	-NA-	Out of Service	104.196.190.229		32770	32769	32768	32768
NetScaler CPX			172.17.0.5	-NA-	Out of Service	10.10.15.159	88-88	32785	32784	32783	32773
NetScaler Gateway			172.17.0.2	-NA-	● Up	104.199.209.157		5022	5080	5443	5163
NetScaler SD-WAN WO											

×	Citrix NetSc	aler	n 05 2017 15:10:23 GMT	nsroot 🗸					
Q Search	A Search here Networks > License Settings > CPX Licenses								
Applicatio	ns	>	CPX Licenses						~
Dash	nboard		Instances						
Insta Insta Licer Sy Th	ances ance Groups nses ystem Licenses nird Party Licenses PX Licenses	>	10.0% Total 10 Used 1						
Even	its	>	The following instances are consun	ning Instance license.					
SSL	Dashboard	>	Name	IP Address	Instance Type	\mathbf{v}	Allocation Status	Allocated Capacity	
Cont	figuration Jobs	>	e516b1b61939	172.17.0.2	NetScaler CPX		Not available 🕛	1	
Cont	figuration Audit	>							

Paso 6: Configurar los servicios de equilibrio de carga en NetScaler CPX y verificar la configuración

Primero, instale los servidores web NGINX en el host de Docker. A continuación, configure el equilibrio de carga en NetScaler CPX para equilibrar la carga de los dos servidores web y pruebe la configuración.

Instalar servidores web NGINX Use los comandos que se muestran en el siguiente ejemplo para instalar servidores web NGINX.

```
1 $ sudo docker pull nginx
2 Using default tag: latest
3 latest: Pulling from library/nginx
4 Digest: sha256:41
      ad9967ea448d7c2b203c699b429abe1ed5af331cd92533900c6d77490e0268
5 Status: Image is up to date for nginx:latest
6
7
8 \*\*$ sudo docker run -d -p 81:80 nginx\*\*
9 098a77974818f451c052ecd172080a7d45e446239479d9213cd4ea6a3678616f
10
11
12 \*\*$ sudo docker run -d -p 82:80 nginx\*\*
13 bbdac2920bb4085f70b588292697813e5975389dd546c0512daf45079798db65
14
15
16 \times sudo iptables -t nat -L -n\times
17 Chain PREROUTING (policy ACCEPT)
18 target prot opt source destination
19 DOCKER all -- 0.0.0.0/0 0.0.0.0/0 ADDRTYPE match dst-type LOCAL
20
21 Chain INPUT (policy ACCEPT)
22
  target prot opt source destination
23
```

```
24 Chain OUTPUT (policy ACCEPT)
25 target prot opt source destination
26 DOCKER all -- 0.0.0.0/0 !127.0.0.0/8 ADDRTYPE match dst-type LOCAL
27
28 Chain POSTROUTING (policy ACCEPT)
29 target prot opt source destination
30 MASQUERADE all -- 172.17.0.0/16 0.0.0.0/0
31 MASQUERADE tcp -- 172.17.0.2 172.17.0.2 tcp dpt:443
32 MASQUERADE udp -- 172.17.0.2 172.17.0.2 udp dpt:161
33 MASQUERADE tcp -- 172.17.0.2 172.17.0.2 tcp dpt:88
34 MASQUERADE tcp -- 172.17.0.2 172.17.0.2 tcp dpt:80
35 MASQUERADE tcp -- 172.17.0.2 172.17.0.2 tcp dpt:22
36 MASQUERADE tcp -- 172.17.0.3 172.17.0.3 tcp dpt:80
37 MASQUERADE tcp -- 172.17.0.4 172.17.0.4 tcp dpt:80
38
39 Chain DOCKER (2 references)
40 target prot opt source destination
41 RETURN all -- 0.0.0.0/0 0.0.0.0/0
42 DNAT tcp -- 0.0.0.0/0 0.0.0.0/0 tcp dpt:5443 to:172.17.0.2:443
43 DNAT udp -- 0.0.0.0/0 0.0.0.0/0 udp dpt:5163 to:172.17.0.2:161
44 DNAT tcp -- 0.0.0.0/0 0.0.0.0/0 tcp dpt:50000 to:172.17.0.2:88
45 DNAT tcp -- 0.0.0.0/0 0.0.0.0/0 tcp dpt:5080 to:172.17.0.2:80
46 DNAT tcp -- 0.0.0.0/0 0.0.0.0/0 tcp dpt:5022 to:172.17.0.2:22
47 DNAT tcp -- 0.0.0.0/0 0.0.0.0/0 tcp dpt:81 to:172.17.0.3:80
48 DNAT tcp -- 0.0.0.0/0 0.0.0.0/0 tcp dpt:82 to:172.17.0.4:80
49 $
```

Configurar CPX para el equilibrio de carga y verificar la distribución de la carga a ambos

```
servicios web
1 $ \*\*ssh -p 5022 root@localhost\*\*
2 root@localhost's password:
3 Welcome to nsoslx 1.0 (GNU/Linux 4.8.0-52-generic x86_64)
4
5 * Documentation: https://www.citrix.com/
6 Last login: Mon Jun 5 18:58:54 2017 from 172.17.0.1
7 root@44ca1c6c0907:~#
8 root@44ca1c6c0907:~#
9 root@44ca1c6c0907:~#
10 root@44ca1c6c0907:~#
11 root@44ca1c6c0907:~# \*\*cli_script.sh "add service web1 172.17.0.3
      HTTP 80"\*\*
12 exec: add service web1 172.17.0.3 HTTP 80
13 Done
14 root@44ca1c6c0907:~# \*\*cli_script.sh "add service web2 172.17.0.4
      HTTP 80"\*\*
15 exec: add service web2 172.17.0.4 HTTP 80
16 Done
17 root@44ca1c6c0907:~# \*\*cli_script.sh "add lb vserver cpx-vip HTTP
      172.17.0.2 88"\*\*
18 exec: add lb vserver cpx-vip HTTP 172.17.0.2 88
19 Done
20 root@44ca1c6c0907:~# \*\*cli_script.sh "bind lb vserver cpx-vip web1
   "\*\*
```

```
21 exec: bind lb vserver cpx-vip web1
22 Done
23 root@44ca1c6c0907:~# \*\*cli_script.sh "bind lb vserver cpx-vip web2
      "\*\*
24 exec: bind lb vserver cpx-vip web2
25 Done
26 root@44ca1c6c0907:~#
27
28 root@44ca1c6c0907:~# \*\*cli_script.sh 'show lb vserver cpx-vip'\*\*
29 exec: show lb vserver cpx-vip
31 cpx-vip (172.17.0.2:88) - HTTP Type: ADDRESS
32 State: UP
33 Last state change was at Mon Jun 5 19:01:49 2017
34 Time since last state change: 0 days, 00:00:42.620
35 Effective State: UP
36 Client Idle Timeout: 180 sec
37 Down state flush: ENABLED
38 Disable Primary Vserver On Down : DISABLED
39 Appflow logging: ENABLED
40 Port Rewrite : DISABLED
41 No. of Bound Services : 2 (Total) 2 (Active)
42 Configured Method: LEASTCONNECTION
43 Current Method: Round Robin, Reason: A new service is bound
       BackupMethod: ROUNDROBIN
44 Mode: IP
45 Persistence: NONE
46 Vserver IP and Port insertion: OFF
47 Push: DISABLED Push VServer:
48 Push Multi Clients: NO
49 Push Label Rule: none
50 L2Conn: OFF
51 Skip Persistency: None
52 Listen Policy: NONE
53 IcmpResponse: PASSIVE
54 RHIstate: PASSIVE
55 New Service Startup Request Rate: 0 PER_SECOND, Increment Interval: 0
56 Mac mode Retain Vlan: DISABLED
57 DBS_LB: DISABLED
58 Process Local: DISABLED
59 Traffic Domain: 0
60 TROFS Persistence honored: ENABLED
61 Retain Connections on Cluster: NO
63 2) web1 (172.17.0.3: 80) - HTTP State: UP Weight: 1
64 3) web2 (172.17.0.4: 80) - HTTP State: UP Weight: 1
65 Done
66 root@44ca1c6c0907:~#
67
68 (external)$ \*\*curl 104.199.209.157:50000\*\*
69 \\<\\!DOCTYPE html\\>
70 \<html\>
71 \langle head \rangle
```

```
72 \<title\>Welcome to nginx\!\</title\>
73 < style >
74 body {
75
76 width: 35em;
77
   margin: 0 auto;
   font-family: Tahoma, Verdana, Arial, sans-serif;
78
79
    }
81
   \\</style\\>
82
   \\</head\\>
83 \ \ \ )
84 \<h1\>Welcome to nginx\!\</h1\>
85 \<p\>If you see this page, the nginx web server is successfully
       installed and
86 working. Further configuration is required.\\</p\\>
87
88 \>For online documentation and support please refer to
   \\<a href="http://nginx.org/"\\>nginx.org\\</a\\>.\\<br/\\>
90 Commercial support is available at
91
   \\<a href="http://nginx.com/"\\>nginx.com\\</a\\>.\\</p\\>
92
   ><em>>Thank you for using nginx.</em>>>
93
94
   \<\box{body}\>
95
   \\</html\\>
96
   (external)$
97
   ![localized image](/en-us/cpx/12-1/media/cpx-image-5.png)
98
99
   (external)$ for i in {
100
101
    1..100 }
      ; \*\*do curl http://104.199.209.157:50000 -o /dev/null ; done\*\*
102
                                                                      Time
104
     % Total
                 % Received % Xferd Average Speed
                                                     Time
                                                             Time
        Current
105
106
                                     Dload Upload
                                                     Total
                                                             Spent
                                                                      Left
        Speed
107
   100 612 100
                           0
                                      1767
                    612
                                  \mathbf{O}
                                                0 --:--:--
       --:-- 1768
109
                 % Received % Xferd Average Speed
                                                             Time
                                                                      Time
110
     % Total
                                                     Time
        Current
111
112
                                     Dload Upload
                                                     Total
                                                             Spent
                                                                      Left
        Speed
113
114
   100
        612 100
                     612
                            0
                                  0
                                      1893
                                                0 --:--:--
       --:-- 1894
115
116
      % Total
                % Received % Xferd Average Speed
                                                     Time
                                                             Time
                                                                      Time
        Current
```

117 118 Dload Upload Total Spent Left Speed 119 120 100 612 100 612 0 0 1884 0 --:--:----:-- 1883 121 % Total % Received % Xferd Average Speed Time Time Time 122 Current 123 124 Dload Upload Total Spent Left Speed 125 126 100 612 100 612 0 0 1917 0 --:--:----:-- 1924 127 % Total % Received % Xferd Average Speed Time Time 128 Time Current 129 Dload Upload Total Spent 130 Left Speed 131 132 100 612 100 612 0 0 1877 0 --:--:----:-- 1883 133 134 % Total % Received % Xferd Average Speed Time Time Time Current 135 Dload Upload Total Spent 136 Left Speed 137 138 100 612 100 612 0 0 1852 0 --:--:----:-- 1848 139 % Total % Received % Xferd Average Speed Time Time Time 140 Current 141 Dload Upload Total Spent 142 Left Speed 143 144 100 612 100 612 0 0 1860 0 --:--:----:-- 1865 145 146 % Total % Received % Xferd Average Speed Time Time Time Current 147 Dload Upload Total Spent Left 148 Speed 149 150 100 612 100 612 0 0 1887 0 --:--:----:-- 1888 151 152 % Total % Received % Xferd Average Speed Time Time Time Current 153 Dload Upload Total Spent Left 154 Speed 155 156 100 612 100 612 0 0 1802 0 --:--:----:-- 1800 157 % Total % Received % Xferd Average Speed Time Time 158 Time Current 159 Dload Upload Total Spent 160 Left Speed 162 100 612 100 612 0 0 1902 0 --:--:----:-- 1906 163 % Total % Received % Xferd Average Speed Time 164 Time Time Current 165 166 Dload Upload Total Spent Left Speed 167 168 100 612 100 612 0 0 1843 0 --:--:----:-- 1848 169 170 171 % Total % Received % Xferd Average Speed Time Time 172 Time Current 173 Dload Upload Total Spent Left 174 Speed 175 176 100 612 100 612 0 0 1862 0 --:--:----:-- 1860 177 % Total % Received % Xferd Average Speed Time Time 178 Time Current 179 Dload Upload Total Spent 180 Left Speed 181 182 100 612 100 612 0 0 1806 0 --:--:----:-- 1810 183 % Total % Received % Xferd Average Speed Time Time 184 Time Current 185 Dload Upload Total Spent Left 186 Speed 187 188 100 612 100 612 0 0 1702 0 --:--:--

```
--:-- 1704
189
190 (external)$
191
192
193
194
195
196 root@44ca1c6c0907:~# \*\*cli_script.sh 'stat lb vserver cpx-vip'\*\*
197
198 exec: stat lb vserver cpx-vip
199
200
202 Virtual Server Summary
203
204
                      vsvrIP port Protocol State Health
     actSvcs
206 cpx-vip 172.17.0.2 88 HTTP UP 100
2
207
208
209
       inactSvcs
211
            Θ
212 cpx-vip
213
214
215
216 Virtual Server Statistics
217
                                         Rate (/s)
218
     Total
219
220 Vserver hits
                                                 0
     101
221
222 Requests
                                                 0
    101
223
224 Responses
                                                 0
              101
225
226 Request bytes
                                                 0
     8585
227
228 Response bytes
                                                 0
     85850
229
230 Total Packets rcvd
                                                 0
     708
231
```

232	Total Packets sent 408	0
233 234	Current client connections	
235	Θ	
236	Current Client Est connections 0	
238	Current server connections 0	
239 240	Current Persistence Sessions 0	
241		
242	Requests in surge queue 0	
244	Requests in vserver's surgeQ 0	
245		
246	Requests in service's surgeQs 0	
248	Spill Over Threshold	
249	0	
250	Spill Over Hits 0	
251 252	Labeled Connection	
253	Ŭ	
254	Push Labeled Connection 0	
255		
256	0 0	0
258	Invalid Request/Response 0	
259		
260	Invalid Request/Response Dropped 0	
261	Vserver Down Backup Hits 0	
263		
264	Current Multipath TCP sessions 0	
265	Current Multingth TCP subflows	
200	0	

267 268	Apdex for client response times 1.00						
269							
270	Average client TTLB 0						
271							
272	web1		172.17.0.3	80	HTTP	UP	51
		0/s					
273							
274	web2		172.17.0.4	80	HTTP	UP	50
		0/s					
275							
276	Done						
277							
278	root@44ca1c6c0907:~#						

Solución de problemas de NetScaler CPX

November 23, 2023

En este documento se explica cómo solucionar los problemas que puede encontrar al utilizar NetScaler CPX. Con este documento, puede recopilar registros para determinar las causas y aplicar soluciones para algunos de los problemas comunes relacionados con la instalación y la configuración de NetScaler CPX.

• ¿Cómo puedo ver los registros de NetScaler CPX?

Puede ver los registros de NetScaler CPX mediante el comando kubectl logs si NetScaler CPX se implementa con la opción tty:true. Puede ejecutar el siguiente comando para mostrar los registros:

```
1 kubectl logs <pod-name> [-c <container-name>] [-n <namespace-name
>]
```

Por ejemplo,

1 kubectl logs cpx-ingress1-69b9b8c648-t8bgn -c cpx -n citrix-adc

A continuación, se muestra un ejemplo de la implementación de pods de NetScaler CPX con la opción tty:true:

```
1 containers:
2 - name: cpx-ingress
3 image: "quay.io/citrix/citrix-k8s-cpx-ingress:13.0-58.30"
4 tty: true
5 securityContext:
```

6 privileged: **true** 7 env:

Puede encontrar más registros de arranque en el archivo /*cpx/log/boot.log* del sistema de archivos NetScaler CPX.

Nota: Para obtener el nombre del pod, ejecute el comando kubectl get pods -o wide.

· ¿Cómo puedo recopilar el paquete de soporte técnico de NetScaler CPX?

Puede ejecutar el siguiente comando en la interfaz shell del nodo principal de Kubernetes para recopilar el paquete de soporte técnico de NetScaler CPX:

```
1 kubectl exec <cpx-pod-name> [-c <cpx-container-name>] [-n <
            namespace-name>] /var/netscaler/bins/cli_script.sh "show
            techsupport"
```

Puede ver el paquete de soporte técnico en el directorio /var/tmp/support del sistema de archivos de NetScaler CPX. Use scp o kubectl cp para copiar el paquete de soporte técnico de NetScaler CPX al destino deseado.

Ejemplo:

```
root@localhost# kubectl exec cpx-ingress1-55b9b6fc75-t5kc6 -c cpx
1
        -n citrix-adc /var/netscaler/bins/cli_script.sh "show
       techsupport"
2
   exec: show techsupport
3
     Scope: NODE
4
  Done
5
   root@localhost# kubectl cp cpx-ingress1-55b9b6fc75-t5kc6:var/tmp/
       support/collector_P_192.168.29.232_31Aug2020_07_30.tar.gz /tmp
       /collector_P_192.168.29.232_31Aug2020_07_30.tar.gz -c cpx
   root@localhost# ll /tmp/collector_P_192.168.29.232
6
       _31Aug2020_07_30.tar.gz
   -rw-r--r-- 1 root root 1648109 Aug 31 13:23 /tmp/collector_P_192
7
       .168.29.232_31Aug2020_07_30.tar.gz
```

• ¿Por qué se bloquea el pod de NetScaler CPX durante el arranque?

Puede comprobar el estado del pod con el comando kubectl describe pods. Ejecute el siguiente comando para conocer el estado del pod:

Ejemplo:

1 kubectl describe pods cpx-ingress1-69b9b8c648-t8bgn

Si los eventos del pod muestran que el contenedor se inició, debe comprobar los registros del pod.
• ¿Cómo copio archivos entre el pod de NetScaler CPX y el nodo maestro de Kubernetes?

Se recomienda utilizar la función de montaje de volumen de docker para montar el directorio / cpx en el sistema de archivos del host. Si un contenedor NetScaler CPX sale de los volcados de núcleo, los registros y otros datos importantes están disponibles en el punto de montaje.

Puede usar cualquiera de los siguientes comandos para copiar archivos entre el pod de NetScaler CPX y el nodo maestro de Kubernetes:

kubectl cp: Puede ejecutar este comando para copiar archivos del pod al nodo:

Ejemplo:

```
1 root@localhost:~# kubectl cp cpx-ingress-596d56bb6-zbx6h:cpx/log/
	boot.log /tmp/cpx-boot.log -c cpx-ingress
2 root@localhost:~# ll /tmp/cpx-boot.log
3 -rw-r--r-- 1 root root 7880 Sep 11 00:07 /tmp/cpx-boot.log
```

scp: puede usar el comando para copiar archivos entre el pod de NetScaler CPX y el nodo de Kubernetes. Ejecute el siguiente comando para copiar archivos del pod al nodo. Cuando se le pida la contraseña, proporcione la contraseña para el usuario SSH:

scp <user>@<pod-ip>:<absolute-src-path> <dst-path>

Ejemplo:

```
1 root@localhost:~# scp nsroot@192.168.29.198:/cpx/log/boot.log /
2 nsroot@192.168.29.198's password:
3 boot.log
4 100% 7880 5.1MB/s 00:00
5 root@localhost:~#
```

· ¿Cómo capturo paquetes en NetScaler CPX?

Para capturar paquetes en NetScaler CPX, inicie la interfaz shell de NetScaler CPX mediante el comando kubectl exec. Ejecute el siguiente comando para iniciar la interfaz shell del pod de NetScaler CPX:

```
1 kubectl exec -it pod-name [-c container-name] [-n namespace-
name] bash
```

Ejemplo:

```
1 kubectl exec -it cpx-ingress1-69b9b8c648-t8bgn -c cpx -n
citrix-adc bash
```

Y ejecute el siguiente comando para iniciar la captura de paquetes:

1 cli_script.sh "start nstrace -size 0"

Si quiere detener la captura de paquetes en curso, ejecute el siguiente comando:

1 cli_script.sh "stop nstrace"

Puede ver los paquetes capturados en un archivo *.cap* en el directorio */cpx/nstrace/time-stamp* en el sistema de archivos NetScaler CPX.

 ¿Por qué el servidor de licencias no está configurado incluso cuando NetScaler CPX se implementa con la variable de entorno LS_IP=<ADM-IP>?

Asegúrese de que se pueda acceder al servidor de licencias desde el nodo en el que se implementa NetScaler CPX. Puede usar el comando ping <ADM-IP> para verificar la conectividad del nodo NetScaler CPX a Citrix ADM.

Si se puede acceder a Citrix ADM desde el nodo, debe verificar los registros de configuración del servidor de licencias en el archivo /*cpx/log/boot.log*. También puede comprobar la configuración del servidor de licencias mediante el siguiente comando en la interfaz shell del pod de NetScaler CPX:

1 cli_script.sh "show licenseserver"

Ejemplo:

```
1 root@cpx-ingress-596d56bb6-zbx6h:/cpx/log# cli_script.sh "show
licenseserver"
2 exec: show licenseserver
3 ServerName: 10.106.102.199Port: 27000 Status: 1 Grace: 0
Gptimeleft: 720
4 Done
```

• ¿Por qué la licencia agrupada no se configura en NetScaler CPX incluso después de una configuración correcta del servidor de licencias en NetScaler CPX?

Compruebe los registros de configuración de licencias en el archivo /*cpx*/*log*/*boot.log*. También puede verificar la licencia agrupada configurada en NetScaler CPX mediante el siguiente comando en la interfaz shell del pod de NetScaler CPX:

1 cli_script.sh "show capacity"

Por ejemplo,

4 Done

Además, asegúrese de que los archivos de licencia requeridos se carguen en el servidor de licencias. También puede verificar las licencias disponibles en el servidor de licencias una vez que se haya configurado correctamente en NetScaler CPX mediante el siguiente comando. Ejecute el comando en la interfaz shell del pod de NetScaler CPX:

cli_script.sh " sh licenseserverpool "

Ejemplo:

1	root@cpx-ingress-596d56bb6-zbx6h:/cpx	/log# cli_script.sh "show
	licenseserverpool"	
2	exec: show licenseserverpool	
3	Instance Total	: 5
4	Instance Available	: 4
5	Standard Bandwidth Total	: O Mbps
6	Standard Bandwidth Availabe	: O Mbps
7	Enterprise Bandwidth Total	: O Mbps
8	Enterprise Bandwidth Available	: O Mbps
9	Platinum Bandwidth Total	: 10.00 Gbps
10	Platinum Bandwidth Available	: 9.99 Gbps
11	CP1000 Instance Total	: 100
12	CP1000 Instance Available	: 100
13	Done	

• ¿Por qué las llamadas a la API NITRO obtienen *una respuesta de rechazo de conexión* de NetScaler CPX?

El puerto predeterminado para las API de NITRO es 9080 (no seguro) y 9443 (seguro) a partir de la versión 12.1 de NetScaler CPX en adelante. Asegúrese de que el puerto NITRO de NetScaler CPX al que intenta acceder esté expuesto en el pod. Puede ejecutar el comando kubectl describe para ver el puerto expuesto y asignado del contenedor NetScaler CPX en la sección contenedor NetScaler CPX:

kubectl describe pods <pod-name> | grep -i port

Ejemplo:

1	ng472 grep -i p	ort		
2	Ports:	80/TCP, 4	443/TCP, 9080/TCP, 9443/TCP	
3	Host Ports:	0/TCP, 0/	/TCP, 0/TCP, 0/TCP	
4	NS_HTTP_PORT	-:	9080	
5	NS_HTTPS_POF	RT:	9443	
6	Port:	<none></none>		
7	Host Port:	<none></none>		
8	NS_PORT:		80	

• ¿Por qué el proceso NSPPE en NetScaler CPX consume la mayor parte del uso de la CPU incluso cuando no hay tráfico o hay poco tráfico?

Si NetScaler CPX se implementa con la variable de entorno CPX_CONFIG='{ "YIELD": "NO "} , el proceso NSPPE consume el 100 por ciento de uso de la CPU incluso cuando no hay tráfico o hay poco tráfico. Si quiere que el proceso NSPPE no consuma el uso de la CPU, debe implementar NetScaler CPX sin la variable de entorno CPX_CONFIG='{ "YIELD": "NO"}. De forma predeterminada, el proceso NSPPE en CPX está configurado para no acaparar ni consumir el uso de la CPU.

• ¿Por qué NetScaler CPX no figura en Citrix ADM incluso cuando se implementó con las variables de entorno necesarias para registrarse en Citrix ADM?

Puede encontrar los registros para el registro de NetScaler CPX con Citrix ADM en el archivo /cpx/log/boot.log del sistema de archivos NetScaler CPX.

Puede comprobar la accesibilidad de la dirección IP de Citrix ADM desde el pod de NetScaler CPX mediante el comando ping. Además, asegúrese de que todas las variables de entorno necesarias para el registro de Citrix ADM estén configuradas para el contenedor NetScaler CPX.

- NS_MGMT_SERVER=<ADM-IP>
- NS_MGMT_FINGER_PRINT=<ADM-Fingerprint>
- HOST=<Node-IP>
- NS_HTTP_PORT=<Mapped-HTTP-Port-on-Node>
- NS_HTTPS_PORT=<Mapped-HTTPS-Port-on-Node>
- NS_SSH_PORT=<Mapped-SSH-Port-on-Node>
- NS_SNMP_PORT=<Mapped-SNMP-Port-on-Node>
- NS_ROUTABLE=0 (la dirección IP del pod de NetScaler CPX no se puede redirigir desde el exterior).
- ¿Por qué cli_script.sh se muestra un mensaje de error de *nombre de usuario o contraseña no válidos* después de cambiar la contraseña del usuario nsroot?

El comando cli_script.sh es una utilidad de envoltura para NSCLI en NetScaler CPX. Ejecuta el primer argumento como cadena de comandos o ruta de archivo y el segundo argumento es opcional, que son las credenciales. Si se cambia la contraseña del usuario nsroot, debe proporcionar credenciales como segundo argumento para cli_script.sh. Puede ejecutar el siguiente comando para ejecutar NSCLI con credenciales:

```
1 cli_script.sh " <command> " ":<username>:<password> "
```

Ejemplo:

7							
8	exec: show ns ip						
9							
10	Ipaddress	Tra	ffic Domain	Туре	Mode		
	Arp	Icmp	Vserver	State			
11			-				
12	172.17.0.3	Θ		NetScaler IP	Active		
	Enabled	Enabled	NA E	nabled			
13	192.0.0.1	Θ		SNIP	Active		
	Enabled	Enabled	NA E	nabled			
14	Done						

net>scaler

© 2025 Cloud Software Group, Inc. All rights reserved. This document is subject to U.S. and international copyright laws and treaties. No part of this document may be reproduced in any form without the written authorization of Cloud Software Group, Inc. This and other products of Cloud Software Group may be covered by registered patents. For details, please refer to the Virtual Patent Marking document located at https://www.cloud.com/legal. Citrix, the Citrix logo, NetScaler, and the NetScaler logo and other marks appearing herein are either registered trademarks or trademarks of Cloud Software Group, Inc. and/or its subsidiaries in the United States and/or other countries. Other marks are the property of their respective owner(s) and are mentioned for identification purposes only. Please refer to Cloud SG's Trademark Guidelines and Third Party Trademark Notices (https://www.cloud.com/legal) for more information.

© 1999–2025 Cloud Software Group, Inc. All rights reserved.