

#CloudPorvExperts

¡Descarga gratuita sin registro!

El Libro de

Cloud por vExperts

Edición 2.0

16

**Bloggers unidos por un
nuevo Proyecto solidario.**

**Todo lo recaudado es donado a la
ONG NASCO feeding minds**



Xavier Caballé • Iván Camargo • Patricio Cerdá • Federico Cinalli • Celia Cristaldo •
Jorge de la Cruz • Xavier Genestos • Héctor Herrero • Ricard Ibáñez • Gorka Izquierdo •
Miquel Mariano • Daniel Romero • Ariel Sánchez • Elver Sena Sosa • Jorge Torres • Raúl Unzué

Prólogo por Rick Vanover

CLOUD POR VEXPERTS

© 2020 - Autores libro

Reservados todos los derechos. Esta publicación está protegida por las leyes de propiedad intelectual.

No se permite distribuir ni parcialmente, ni totalmente, la publicación a través de cualquier medio, soporte, sin autorización expresa de los autores.

Todas las marcas, nombres propios, que aparecen en el libro son marcas registradas de sus respectivos propietarios, siendo su utilización realizada exclusivamente a modo de referencia.

Los autores del libro no se hacen responsables de los problemas que pueda causar en su infraestructura, siendo responsabilidad de los administradores de sistemas realizar las copias de seguridad, planes de contingencia y laboratorio de pruebas previo antes de aplicar cualquier cambio en los servidores de producción.



This work is licensed under the **Creative Commons** Attribution-NonCommercial 4.0 International License. To view a copy of this license, visit

<http://creativecommons.org/licenses/by-nc/4.0/>.

LOS vEXPERTS

XAVIER CABALLÉ

Hola, mi nombre es Xavier Caballé, soy administrador de sistemas informáticos desde hace más de veinte años. Mi trabajo está orientado al soporte técnico, mantenimiento e instalación de infraestructuras.

Entre mis labores habituales se encuentran la administración, configuración y migración de entornos basados en un dominio de Active Directory, y todos los servicios asociados a él como pueden ser, los servidores de nombres, los servicios de DHCP, el servicio de impresión, la configuración y gestión de directivas de grupo, y el servidor de ficheros.

También me dedico a la instalación, gestión y migración de servidores de correo electrónico basados en Microsoft Exchange server u Office 365 con Exchange OnLine.

En la actualidad, una gran parte de mi tiempo laboral lo consume la tarea de configurar y administrar granjas de virtualización basadas en tecnología de VMware.

Tampoco podemos olvidar la gestión y programación de tareas de copia de seguridad con distintos productos de Backup como pueden ser, VEEAM, ARCServe o BackupExec.

Soporte técnico de Hardware de servidores de grandes marcas como HP, DELL, IBM, Fujitsu.

En la parte de storage me dedico a la instalación, configuración y gestión de cabinas de discos de HP, EMC y Fujitsu.

En lo personal, dedico gran parte de mi tiempo libre a la fotografía paisajística y estar al aire libre con mi familia.

También soy el autor del blog dedicado a la tecnología de la información <http://www.pantallazos.es> y del canal de YouTube <https://www.youtube.com/c/pantallazoses>



Iván Camargo

Hola, mi nombre es Iván Camargo, y actualmente trabajo como Associate Partner para IBM. Soy un entusiasta de la tecnología y de las comunidades. Mis pasiones en la vida son: Dios, mi familia y la tecnología.

Soy Ingeniero de Sistemas con más de 17 años de experiencia en el área de tecnologías de la información. Trabajé en diferentes roles como administrador de sistemas, ingeniero consultor, especialista en preventa, arquitecto de soluciones y, jefe de arquitectura y estrategia de nube.

He tenido el placer de ser nombrado VMware **vExpert** desde 2014, y en el año 2020 en la subcategoría de Modern Applications en VMware. He participado en comunidades del ecosistema en los programas como: Nutanix Technology Champion en 2017 y también NetApp United 2017, 2018. Tengo experiencia en tecnologías alrededor de centros de datos modernos y de TI a escala web: SDS, SDN, Cloud, DevOps, SME (Docker - Kubernetes) Aplicaciones nativas en la nube y soluciones como VMware vSphere Integrated Containers, Pivotal Container Services, Cloud PKS, RedHat Openshift, así como conocimientos y certificación Arquitectura Empresarial.

Con grandes valores profesionales y personales, Apasionado de la tecnología y el aporte que genera al mundo. Pasión por las comunidades tecnológicas co-founder del VMUG Colombia, expositor en BrownBag LATAM y co-founder del grupo de Cloud Native Computing Foundation (CNCF) community en Bogotá, Colombia.

Creo profundamente en compartir mis experiencias y conocimientos con la comunidad, para poder así retribuir lo que la comunidad ha aportado en mi formación profesional.

Además de la tecnología, soy líder y profesor de comunidades de niños y jóvenes en Bogotá, en donde los inspiramos a ir por un mundo mejor, disfruto del cine, la música y la lectura.

Me pueden encontrar en mi blog y en redes sociales:

Blog: <https://www.linkedin.com/in/iv%C3%A1n-camargo-5bb50846/detail/recent-activity/posts/>

Twitter: <https://twitter.com/ivanrcamargo>

LinkedIn: <https://www.linkedin.com/in/iv%C3%A1n-camargo-5bb50846/>

Instagram: <https://www.instagram.com/ivancho190/>

AGRADECIMIENTOS

Antes de empezar, quiero agradecer a Federico Cinalli y Ariel Sanchez por invitarme a hacer parte de esta iniciativa maravillosa. La comunidad me ha dado mucho y siempre disfruto el espacio para devolver un poco de todo lo que he recibido. Un día me atreví a involucrarme y fue una muy buena decisión. He conocido grandes amigos de todo el mundo. Gracias a todo el equipo de vExperts por este tiempo de esfuerzo y sacrificio para donar nuestro tiempo y conocimientos para un propósito espectacular, que es ayudar a NASCO Feeding Minds, una fundación que ayuda a los niños y niñas de Ghana.

Gracias a mi esposa Diana, y a mis hijos Juan José y Juan Esteban, por su paciencia y comprensión. A mi familia que siempre está ahí para dar un “bravo”. Por supuesto a Dios quien dispone todo para bien, para los que lo aman.

PATRICIO CERDA

Hola, mi nombre es Patricio Cerda, y actualmente trabajo como consultor independiente tanto en Europa como para Latinoamérica.

Comencé mi carrera profesional allá por el año 2001, cuando comencé trabajando como desarrollador por poco más de 1 año. Luego de pasar algunos años deambulando por muchas ramas de la informática, incluyendo unos años trabajando con soluciones de seguridad (Firewalls, IDS/IPS), así como también con soluciones Microsoft (AD, Exchange y Sharepoint), finalmente me encontré un día haciendo pruebas con VMware Server 1, creando mi primera máquina virtual para simular un entorno de producción. Era el año 2006 y sin saberlo comenzaba mi aventura en el mundo de la virtualización.

Muchos años han pasado desde aquello, y en el camino me he ido especializando en múltiples soluciones y tecnologías, comenzando con soluciones VMware como NSX-T, Virtual SAN y vRealize Suite, así como de otros fabricantes como AWS, Veeam, Nutanix, Brocade y Dell EMC. Durante este proceso además me convertí en instructor oficial VMware (VCI) el año 2014, para 3 años más tarde convertirme también en instructor oficial Veeam (VMCT).

Creo profundamente en compartir mis experiencias y conocimientos con la comunidad, para poder así retribuir lo que la comunidad me ha aportado en mi formación profesional. Pensando en eso es que comencé con mi blog el año 2009, y también esto fue lo que me llevó a dar el paso de convertirme en instructor.

Además de la tecnología, soy un aficionado a la fotografía, hobby que adopté luego de que mi labor como consultor me llevara a visitar múltiples países. Ahora tomo cada uno de estos viajes como una oportunidad para recorrer nuevas ciudades y poder así retratarlas con mi cámara.

Me pueden encontrar en mi blog y en redes sociales:

Blog: <https://patriciocerda.com>

Twitter: <https://twitter.com/patote83>

LinkedIn: <https://www.linkedin.com/in/prcerda/>

Instagram: <https://www.instagram.com/patricio.rcc/>

AGRADECIMIENTOS

En primer lugar, agradecer a Federico Cinalli por permitirme ser parte de este proyecto. Cuando Fede nos propuso la idea de escribir este libro, no dude ni un instante en sumarme a este esfuerzo conjunto, el cual además de ser un aporte a la comunidad, nos permite llevar a cabo una labor solidaria.

Un agradecimiento especial además a Dagoberto Araya, amigo desde mis tiempos de Universidad. El me abrió la puerta a una enorme oportunidad profesional 11 años atrás, que finalmente me ha permitido llegar a donde estoy ahora.

Agradecer a mi familia, a mi madre y a mis amigos. Casi todos ellos están a miles de kilómetros de distancia, pero son mi constante apoyo y un pilar fundamental en mi vida. La distancia no disminuye ni un centímetro el amor que siento por ellos.

Finalmente agradecer a los sponsors que han creído en este proyecto, y en el espíritu solidario del mismo. ¡Gracias a todos los lectores de este libro, espero que lo disfruten!

FEDERICO CINALLI

Mi nombre es Federico Cinalli (Fede para los amigos) y soy consultor independiente.

Me especializo en infraestructuras de virtualización con VMware, más específicamente con los productos vSphere, vSAN, NSX, vRealize Suite y vCloud Director.

Soy instructor oficial de VMware y miembro del equipo de instructores asociados de VMware EMEA dictando cursos oficiales en Europa, África y Oriente Medio para clientes de VMware.

Tengo el honor de formar parte del equipo de arquitectura de Cloud y servicios profesionales de Adistec, donde diseñamos e implementamos infraestructuras con tecnologías VMware como plataformas de Cloud Públicas con vCloud Director y el stack completo de VMware, además de soportar los cuatro datacenters que tiene la compañía repartidos por América para dar servicio de Cloud Pública a los canales. También suelo hacer presentaciones en eventos para Adistec.

A nivel de certificaciones, tengo actualmente VCIX6-DCA, VACP5-DCA y DCD, VCP6-CMA, VCP6-NV, VCP7-DT, VCI L2, y reconocimiento de **vExpert** desde el año 2014 con las especialidades de vSAN y Cloud Management.

Desde marzo de 2020 co-presento con mi gran amigo Héctor Herrero, Un Podcast para TI en la plataforma Ivoox, donde disfrutamos de lo lindo hablando de los temas que nos gustan y aprendiendo de los invitados, que son todos unos cracks.

Cuando dispongo de tiempo colabro con la comunidad, ya sea a través de mi Blog, en presentaciones en los VMworlds, vBrownBag y los VMUGs.

AGRADECIMIENTOS Y DEDICATORIA

Quiero agradecer especialmente a mis 15 compañeros que lo dieron todo para que este proyecto sea posible. Particularmente en este año tan difícil, con tanta carga laboral, le robaron tiempo a sus familias y seres queridos para llevar adelante este proyecto solidario.

Otro punto que me gustaría destacar es la humildad de cada uno de los participantes. Todos sabemos que en el mundo TI, y especialmente cuando se trata de profesionales que están expuestos todo el tiempo a redes sociales, los egos son enormes. Tanto la primera edición de este libro como esta segunda edición se caracterizó en que todos tiraron del carro a la par y en ningún momento nadie se creyó más importante que el compañero.

Otro agradecimiento es para los Sponsors que nos permiten que este proyecto sea doblemente solidario y que seamos capaces de, entre todos, poder colaborar con nuestro granito de arena a organizaciones tan nobles como NASCO feeding minds.

Y por último y no menos importante quiero dedicar mis páginas a las princesas que me aguantan todo el día en casa, mis hijas Chiara y Olivia, y mi mujer Roxana.

A mi perro Che también lo agrego a la dedicatoria 😊

CELIA CRISTALDO CANTERO

De Paraguay para el mundo... Llevo poco más de 13 años en el área de TI. Mi foco siempre fue mayormente virtualización de servidores (VMware) y respaldo de ambientes virtuales, pero me ha tocado aprender de todo. Actualmente trabajo como Systems Engineer en Veeam Software para el sur de Latinoamérica.

Tengo el título de Licenciada en Ciencias Informáticas con énfasis en Análisis de Sistemas, de la Universidad Nacional de Asunción, Paraguay. Cuento con la certificación VMware Certified Professional DCV desde la versión 3.5 hasta la 6.5; Veeam Certified Engineer, y además tengo el honor de ser **vExpert** desde 2013 hasta 2020... espero seguir sumando estrellas.

Hace 8 años me aventuré a mudarme a Santiago de Chile, en donde tengo residencia actualmente, aunque me dan ganas de ser nómada. Hoy día formo parte de los líderes del *VMware User Group* de Chile.

En paralelo a mi carrera en TI, estoy emprendiendo en el mundo de la Consejería terapéutica. Tengo un Diplomado en Sexualidad humana y otro en Terapia y Consejería Sexual. Actualmente me encuentro realizando prácticas supervisadas de consejería con pacientes. La Psicología ha sido un gran bastón en mi vida.

Me pueden encontrar en:

Twitter <https://twitter.com/celiacri>

Blog <https://www.v-celia.com/>

Linkedin <https://cl.linkedin.com/in/celiacristaldoc>

¡Pasan a saludar!

AGRADECIMIENTOS Y DEDICATORIA

Le dedico este logro a dos personas influyentes en mi vida, que ya no están físicamente: mi abuela Perla, que me dejó un legado de fortaleza, sensibilidad y resiliencia, siempre apoyándome en mi rol de oveja de la familia. Y mi abuelo Braulio, que me heredó sus genes de curiosidad y su obsesión por la ortografía xD, él me enseñó la importancia de los libros y la escritura.

Agradezco a mis padres, que en todo momento me incentivaron a estudiar, a mirar siempre más allá para alcanzar mis metas, que hicieron todo lo que podían con lo que tenían para criarme de la mejor manera posible, y, sobre todo, gracias por dejarme abrir mis alas y volar.

Gracias a mi hermano Osvaldo, a mis amigos y amigas, a mi persona especial, por siempre creer en mí, por celebrar mis aciertos y señalarme las cosas que puedo mejorar. Que están ahí a pesar de la distancia y el tiempo.

Gracias a cada empleador que apostó a mis conocimientos y aptitudes, por darme herramientas para seguir creciendo en esta profesión; gracias a cada compañero y compañera de trabajo que me apoyó y me apoya.

Gracias a los Sponsors que confiaron en este proyecto. Su ayuda será muy bien aprovechada.

Gracias a este grupo de soñadores por volcar su tiempo y conocimiento en estas páginas, y por invitarme a ser parte de este proyecto especial.

JORGE DE LA CRUZ

Mi nombre es Jorge de la Cruz, actualmente trabajo como Senior Systems Engineer para la empresa Veeam, en Reino Unido. Mi experiencia en la informática se remonta al año 2005, donde comencé como Administrador de Sistemas para un ciber-café, posteriormente desarrollé medio año en Java para Vector Software, y comencé pasado este tiempo en el mundo de la consultoría de sistemas.

Recuerdo todavía cómo fue mi primera incursión con VMware ESX 3.0 y VirtualCenter 2.0, y el posterior 3.5 y las increíbles novedades presentadas en vSphere 4.0. Comencé desplegando un single-node con almacenamiento en una HP MSA y ese entorno escaló a múltiples VNX con cientos de VMs ofreciendo servicio de Hosting.

Una de las experiencias más bonitas que tengo de mi tiempo como consultor fue en Anadat Consulting, para los que no lo conoczáis, Anadat es una empresa ubicada en Rivas-Vaciamadrid. Anadat Consulting ofrece todo tipo de servicios de virtualización, consultoría de sistemas, despliegues de entornos, Hiperconvergencia, Software-Defined Datacenter, y mucho más. Una empresa que me formó como profesional, y en la que tuve la oportunidad de conocer a muchos profesionales del sector, entre ellos Christiam, Óscar, Marcos, Héctor, Bernardo, etc.

Me uní posteriormente a la solución open source de correo electrónico y colaboración, Zimbra, donde estuve cuatro años creando contenido de todo tipo, Wikis (<https://wiki.zimbra.com>), blogs (<https://blog.zimbra.com/author/jcruz>) e incluso tuve la suerte de dirigir la estrategia de Product Marketing y Product Management.

Mis competencias profesionales se basan alrededor de la virtualización de centro de datos, esto es desde la Infraestructura como es Compute, Storage y Networking, hasta la capa de virtualización y abstracción del Hardware, pasando por monitorización usando herramientas open source y comerciales, y seguridad incluyendo Cisco, SonicWall, FortiGate y PaloAlto.

Actualmente escribo desde hace muchos años en <https://jorgedelacruz.es> donde tengo la suerte de contar con unos 3000 lectores cada día. Más datos y formas de contactarme:

Blog: <https://jorgedelacruz.es>

Twitter: <https://twitter.com/jorgedlcruz>

Linkedin: <https://www.linkedin.com/in/jorgedelacruzmigo/>

YouTube: <https://www.youtube.com/user/jorgedlcruz23>

Email: Jorge.delacruz@jorgedelacruz.es

Teléfono: +1 202 738 4705

AGRADECIMIENTOS

No puedo dejar de agradecer a mi mujer Irina y mi hija Victoria el tiempo que no dedico a ellas, porque lo dedico a seguir aprendiendo, trabajar o crear contenido, incluido este libro. Al lado de todos nosotros, tenemos muchas personas que nos entregan todo sin pedirnos nada a cambio, mis humildes capítulos se los dedico a ellas.

También agradecer a Federico Cinalli por dejarme participar en el proyecto, y por supuesto a todos los profesionales que han participado en este libro, yo ya me he leído sus capítulos y es lo mejor que he leído sobre VMware en toda mi vida.

Por último, agradecer a todos los patrocinadores que han contribuido al proyecto.

XAVIER GENESTOS

AUTOBIOGRAFÍA

- Administrador de sistemas: Entornos Microsoft, GNU/Linux, VMware, entre otros. (*Año 2001 hasta la actualidad*).
- Formador IT: Cursos prácticos sobre VMware, Active Directory, Exchange, GPOs, Linux, etc. (*Año 2006 hasta la actualidad*).
- Escritor de libros de IT: 13 libros publicados: (*Año 2012 hasta la actualidad*).
 - WS2012LABS - Windows Server 2012
 - EX2013ADM - Exchange Server 2013
 - WFS - Windows File Server
 - GPOIT - Group Policy Objects para administradores de IT
 - ADIT - Active Directory para administradores de IT
 - LinuXe - Linux para empresas
 - VBESXi - Veeam Backup sobre ESXi
 - EX2016ADM - Exchange Server 2016
 - WS2016LABS - Windows Server 2016
 - RDSIT - Remote Desktop Services para administradores de IT
 - WIN10IT – Windows 10 para administradores de IT
 - WS2019LABS - Windows Server 2019
 - BIT – BitLocker para Administradores de IT
- Blogger en <https://www.SYSADMIT.com>: (*Año 2013 hasta la actualidad*)

Enlaces:

Blog: <https://www.sysadmit.com>

Linkedin: <https://es.linkedin.com/in/xaviergenestos>

Grupo SYSADMIT en Linkedin: <https://www.linkedin.com/groups/8550757>

Twitter: <https://twitter.com/sysadmit>

AGRADECIMIENTOS

Quería aprovechar este fragmento para agradecer al resto de bloggers su gran trabajo y dedicación en este proyecto, también a la familia y amigos por su apoyo y por supuesto a todos los que habéis decidido dedicar vuestro tiempo a leer esta publicación conjunta: ¡Seguro que os gustará!

HÉCTOR HERRERO

¡Hola! Soy Héctor Herrero, un bilbaíno pura cepa, del año 81. Desde los 19 años en este sector dando guerra. Desde hace más de 15 años soy responsable de Open Services IT, una empresa orientada a los servicios TIC especializados, donde mimamos y cuidamos de nuestros clientes, ahí realizo proyectos, consultorías y formaciones, entre otros.

Aunque, seguramente alguno que otro igual me conoce del Blog Bujarra.com, que desde hace la tira de años voy escribiendo manuales y How To's de todo con lo que he ido aprendiendo. Ya lo sabéis, me encantan las Raspberry Pi, por todo lo que nos pueden aportar en el negocio y en el hogar, haciéndolo todo más inteligente.

Un apasionado de las tecnologías, últimamente con más profundidad en el mundo de la monitorización con Centreon, Grafana, NagVis o Stack de ELK, entre otros. Como sabéis, otro de los pilares es la virtualización, tanto de infraestructuras con productos de VMware, cómo virtualización de aplicaciones y escritorios con la familia de Citrix. Heredero del mundo Microsoft, donde he vencido batallas desde Windows NT o migraciones de Exchange hasta hoy Office 365. ¡Ah! adicto al Home Assistant y dotar de inteligencia al hogar.

Por cierto, junto a otro gran autor del Libro, como es Fede, tengo el honor de co-presentar "Un Podcast para TI", donde cada semana traemos un Podcast distinto con la intención de presentar material interesante de nuestro sector. Por último, pero no por ello menos importante, que, si queréis aprender del mundo Citrix, tengo un libro gratuito (388 páginas en español) que igual interesa, "Citrix para administradores de IT".

Mentalidad Open.

Compañía: www.openservices.eus

Blog: <http://www.bujarra.com>

Twitter: [@nheobug](https://twitter.com/nheobug)

Linked In: <https://www.linkedin.com/in/hectorherrero/>

Podcast: https://www.ivoox.com/podcast-un-podcast-para-ti_sq_f1866190_1.html

Libro Citrix: <https://www.bujarra.com/libro-citrix-para-administradores-de-it-gratis/>

AGRADECIMIENTOS

Quería agradecer a cada autor de este libro el poner su granito, el tiempo que ha dedicado a sus capítulos, y sobre todo a los que siempre empujaron porque esto saliese adelante.

Agradecer a la ONG NASCO Feeding Minds la labor que realiza cada día, gracias a todos los que habéis donado o aportado de alguna manera.

Y por supuesto, dedicarle estas líneas a mi madre, a mi padre, a mi hermano, mi txabala, mi moco, a la perra y al gato también, ya que estamos....

¡Compartir y ser felices!

RICARD IBÁÑEZ

Apasionado de mi familia, a la cual le intento dedicar todo mi tiempo libre y sacar un poco más para mis ratos de deporte.

Me encanta la tecnología desde que tengo memoria y, sobre todo, los gadgets que simplifican la vida, o a veces hasta me la complican.

Llevo en el mundo de la tecnología hace ya más de 10 años, donde he trabajado dando soporte Helpdesk, he viajado como consultor para una empresa de ámbito nacional implementando tecnologías de Microsoft, VMware y Veeam entre otras. También he peleado como administrador de sistemas en cliente final, lo cual, me ha enseñado que todo lo que diseñamos e implementamos siempre se puede mejorar.

Soy Blogger hace más de 7 años, escribiendo sobre todo tipo de tecnologías de una manera muy práctica, documentando procesos que simplifiquen las tareas a los administradores de sistemas y he sido recompensado con cinco estrellitas **vExpert** 16/17/18/19/20.

Blog: www.cenabit.com

Twitter [@ricardibanez](https://twitter.com/ricardibanez)

LinkedIn: [Ricard Ibáñez](https://www.linkedin.com/in/Ricard-Ibanez/)

AGRADECIMIENTOS

Esta segunda edición ha sido un poco accidentada debido al año que hemos vivido, pero desde el principio todo el equipo se volcó en hacer posible esta segunda edición con nuevos colaboradores, y por ello agradezco a todos hacer el esfuerzo de conseguir sacar adelante esta segunda edición.

También quiero agradecer a mi mujer Laura que aguante a un “informático” en su vida durante tantos años y consiga que cada día tenga más ganas de seguir avanzando en lo personal y profesional.

Por último, agradecer a todos los Sponsors del libro, los cuales consiguen que este proyecto tenga un impacto en la vida de mucha gente.



GORKA IZQUIERDO

Hace más de 10 años decidí dedicarme al mundo de la informática viniendo de un gremio que no tenía ver nada con este, consiguiendo grandes logros y reconocimientos como el **vExpert** (15-20) y el Veeam Vanguard en 2016.

Desde entonces no he parado de crecer profesionalmente, donde empecé dando soporte de primer nivel de forma remota e insitu de prácticas, y con 30 años.

He estado como responsable de IT de una empresa del sector de las energías renovables, donde desde ahí, he pasado a dar soporte de tecnologías como VMware, VMware Horizon, Veeam Backup, ComVMault, Azure, AWS, Netapp, Microsoft, y un largo etc.

Me podéis seguir en mi blog <https://aprendiendoavirtualizar.com>

Linkedin <https://www.linkedin.com/in/gorkaizquierdobizkarrondo/>

Twitter <https://twitter.com/vGorkon>

Facebook <https://www.facebook.com/aprendiendoavirtualizar>

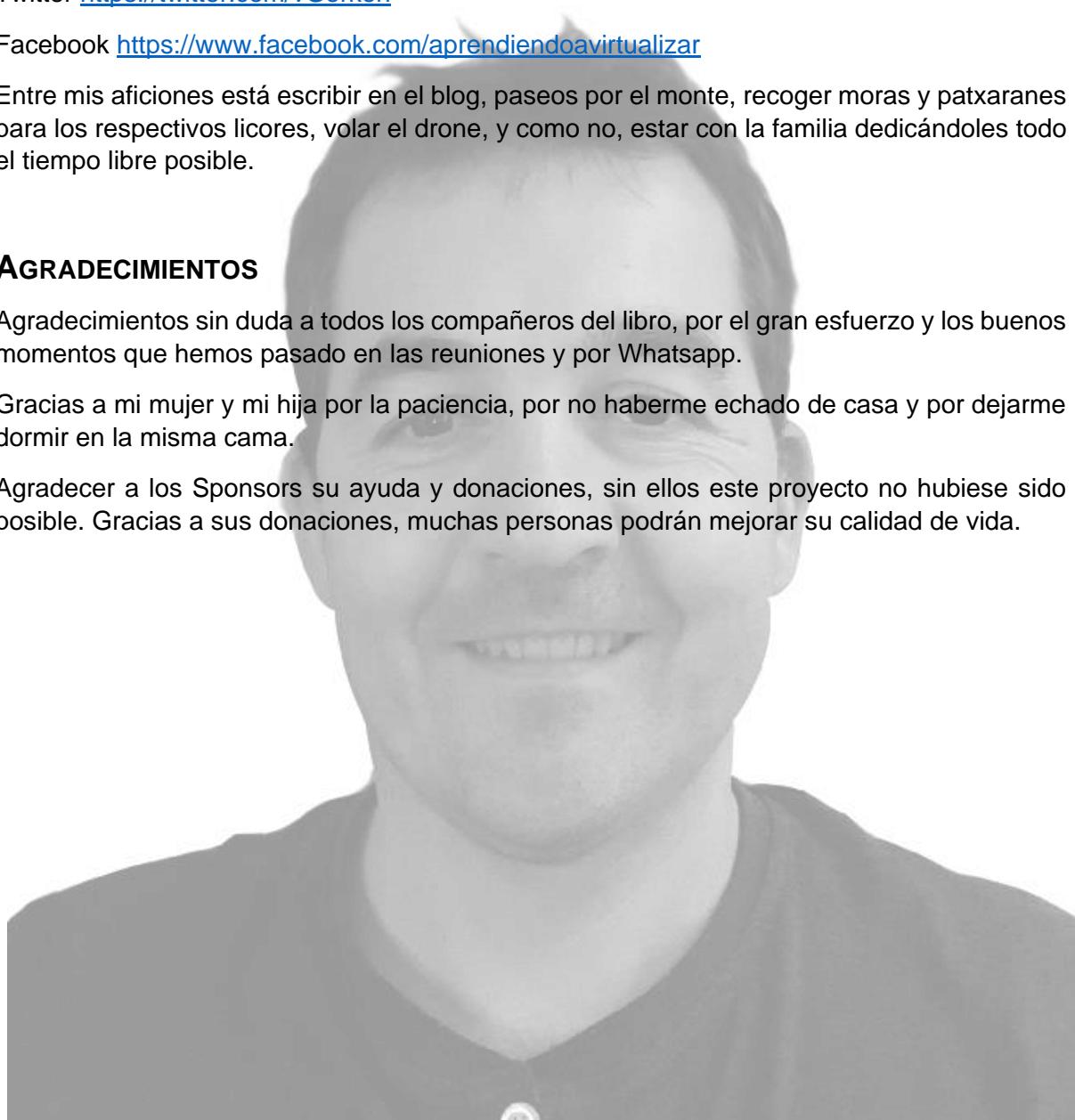
Entre mis aficiones está escribir en el blog, paseos por el monte, recoger moras y patxaranes para los respectivos licores, volar el drone, y como no, estar con la familia dedicándoles todo el tiempo libre posible.

AGRADECIMIENTOS

Agradecimientos sin duda a todos los compañeros del libro, por el gran esfuerzo y los buenos momentos que hemos pasado en las reuniones y por Whatsapp.

Gracias a mi mujer y mi hija por la paciencia, por no haberme echado de casa y por dejarme dormir en la misma cama.

Agradecer a los Sponsors su ayuda y donaciones, sin ellos este proyecto no hubiese sido posible. Gracias a sus donaciones, muchas personas podrán mejorar su calidad de vida.



MIQUEL MARIANO

A parte de la tecnología, me encanta el deporte al aire libre. Aunque viva en una isla como Mallorca, soy más de montaña que de playa, así que es fácil encontrarme por el monte haciendo MTB, corriendo o simplemente practicando algo de senderismo.

En lo profesional,uento con más de 13 años de experiencia en el ámbito de sistemas, y en los últimos años he centrado mi trayectoria profesional especializándome en todo lo relacionado con la virtualización y automatización del datacenter:

- Diseño, implementación y administración de grandes infraestructuras, tanto SDDC como EUC
- Instalación, configuración y administración de almacenamiento Hitachi HUS, VSP Gx00, EMC VNXe, IBM Storwize, QNAP
- Instalación, configuración y administración de SAN con Brocade FC Switches
- Amplios conocimientos en protocolos de almacenamiento: iSCSI, FC, NFS, SMB/CIFS
- Instalación, configuración y administración de servidores Dell PowerEdge, HP Proliant, Hitachi Compute Blade
- Disaster Recovery, Veeam Backup & Replication, Cohesity Data Platform y continuidad de negocio
- Análisis de carga de trabajo, performance y dimensionamiento de máquinas virtuales
- Automatización con Ansible, scripting con PowerCLI, bash
- Monitorización con PRTG Network Monitor

He tenido la suerte de poder sacarme [algunas certificaciones de VMware](#) y en estos últimos 5 años, VMware me ha nombrado [#vExpert](#)

Blog: <https://miquelmariano.github.io/>

Twitter [@miquelMariano](#)

LinkedIn: Miquel Mariano

AGRADECIMIENTOS

Una vez más, agradecer profundamente a los impulsores de este segundo proyecto: Fede y Héctor. Sin ellos nada de esto hubiera sido posible. Agradecer también a todos los compañeros co-autores toda su implicación y el hacer posible esta segunda edición.

En segundo lugar, agradecer también a www.ncora.com todo lo que me aportan a nivel profesional para que pueda seguir creciendo día a día.

En tercer lugar, a mi familia. Al final ellas, mi mujer y mi hija han sido las grandes afectadas y las que han visto mermadas sus horas de dedicación en beneficio del proyecto.

Y para finalizar, como no, a todos los sponsors que han creído en el proyecto desde el minuto 0 y a todos los lectores que han decidido dedicar su tiempo en leer nuestro libro.

¡Gracias a todos!

DANIEL ROMERO

Hola Soy Daniel Romero Sánchez, actualmente trabajo como CTO en ilusia®

A nivel profesional llevo más de quince años dedicados al sector TI. Comencé mi andadura profesional como administrador de sistemas centrándome en el mundo de la monitorización, entendiendo las arquitecturas de sistemas y desarrollando scripts para monitorizarlas. Con el paso de los años comienza mi pasión por el mundo de la virtualización y el cloud computing, de ahí mi especialización en VMware, AWS o OpenStack.

Durante casi toda mi carrera he tenido siempre claro que no hay que realizar tareas repetitivas. Y como siempre he tenido bastante curiosidad por el desarrollo, he creado decenas de scripts para automatizar despliegues y tareas, liberando a los equipos de desarrollo de labores fuera de su alcance.

Estar en continuo contacto con la programación me ha llevado a aprender como diseñar arquitecturas de aplicaciones ya sean monolíticas o basadas en microservicios.

Ahora gestiono a un equipo de TI, a la vez que superviso la infraestructura, políticas de seguridad o nuevas tecnologías adaptables a nuestros servicios.

Me considero una persona autodidacta que no puede parar de conocer cómo funcionan las tecnologías emergentes. Todos mis conocimientos los plasmo en el blog www.dbigcloud.com.

BLOG: www.dbigcloud.com

Twitter: [@drsromero](https://twitter.com/drsromero)

Linkedin: [Daniel Romero](https://www.linkedin.com/in/danielromero/)

DEDICATORIA Y AGRADECIMIENTOS

Hace un año comenzamos una aventura, con mucho por recorrer, pero con un objetivo claro: compartir. Lo hicimos en equipo, catorce personas de diferentes partes del mundo, ayudándonos, aprendiendo los unos de los otros. Por aquel entonces no imaginábamos lo que iba a pasar, pero conseguimos algo muy grande. Este año nos hemos colgado la mochila a la espalda y volvemos a la carga con nuevas incorporaciones. De ahí que mis primeras palabras de agradecimiento sean para ustedes, mis compañeros de aventura, por el esfuerzo que habéis realizado en vuestro tiempo libre, sin pedir nada a cambio y en un año tan complicado. Pensad que no solo hemos escrito un libro, hemos construido las bases para hacer llegar nuestro conocimiento, sobre tecnología, a toda la comunidad de habla hispana. Además, debemos estar orgulloso de poder de ayudar a gente realmente necesitada. ¡Gracias, amigos!

No quiero olvidarme de mi familia, que me apoyan día a día y que me han ayudado a poder estar hoy aquí. Destacando a mis cuatro fantásticas sobrinas que son, hoy en día, mi inspiración, las que me hacen luchar por que exista un mañana mejor. ¡Gracias!

Por otro lado, quiero mencionar a toda la comunidad IT que, de forma desinteresada, comparten su conocimiento, en redes sociales, foros, blog, etc. Estoy muy agradecido de la gran labor que hacéis. En pocas profesiones hay una comunidad tan activa.

Por último, quiero agradecer a los patrocinadores que, siendo un año tan complicado, han puesto su granito de arena y han aportado recursos para ayudar a la ONG NASCO feeding minds.

Gracias a todos, volveremos el año que viene con más.

ARIEL SÁNCHEZ

Ariel Sánchez nació en Costa Rica y ha vivido en Estados Unidos desde el 2011. Completó una licenciatura en ingeniería electrónica y rápidamente se enfocó en TI.

Tuvo experiencia trabajando en ambientes financieros y de centros de contacto como empleado y supervisor del NOC, y gerente de soporte al usuario, pero se sintió más feliz trabajando como administrador de sistemas y especialista de tecnologías VMware.

Trabaja para VMware desde el 2017 como gerente técnico de cuenta senior y ha sido reconocido como un embajador para la oficina del CTO.

Ha alcanzado [certificaciones avanzadas](#) para virtualización de centros de datos y redes, pero está más orgulloso de ser un participante activo en la [vCommunity](#) desde el 2014. Ha sido reconocido como [vExpert PRO](#) para USA, y frecuentemente presenta charlas de comunidad en [VMUG Usercons](#).

Está particularmente orgulloso de formar parte de [vBrownBag](#), un equipo que produce videos técnicos gratuitos para la comunidad, participando en [Español](#) e [Inglés](#).

Adora interactuar con otros en Twitter, asistir a reuniones de usuarios de TI, discutir laboratorios caseros, jugar al tenis de mesa y aprender sobre la cultura japonesa.

Tiene debilidad por proyectos de código abierto como [OpenBSD](#), [Open vSwitch](#) y su colaboración con [Edgar Sánchez](#), [vDocumentation](#).

Puedes encontrarlo en LinkedIn con el URL www.linkedin.com/in/ariel-sanchez-mora pero prefiere ser contactado por Twitter <https://twitter.com/arielsanchezmor>

AGRADECIMIENTOS

A mis colegas en este esfuerzo, con los cuales he reído, trabajado y disfrutado estos proyectos. Particularmente a Celia, quien hizo muchísimas labores esenciales de edición y coordinación – se ganó el Ballon D'Or de esta segunda edición.

A los autores que vienen en las siguientes ediciones, porque sé que el trabajo les quitará tiempo personal, pero es sumamente importante para nuestra comunidad que ustedes den también su aporte.

A todos los patrocinadores, que con su tiempo, apoyo y contribuciones ayudan a amplificar el impacto de este trabajo. Me trae gran satisfacción ver cómo el proyecto sigue creciendo.

A mi esposa Amy, mis padres Dennis y Glenda, mis hermanos, familia extendida y amigos, porque son mi corazón. Todo lo que hago, es para que estén orgullosos de mí.

En particular, a mi esposa por estos diez años juntos, ella sabe cuánto la comunidad significa para mí, ha visto el impacto que ha tenido en mi carrera, y me deja seguir dando mi tiempo; hasta me sigue cuando la llevo por calles oscuras a conocer a un grupo de gente loca por La Rambla de Barcelona. ¡Te Amo!

A todas las amistades que he hecho a través de VMUG, vBrownbag, vBrownbag LATAM, la mafia LATAM, VMunderground – en fin, la #vCommunity que quiero que se convierta en la #vComunidad.

A ti, por leernos y ayudar a que nuestra comunidad siga creciendo y compartiendo conocimiento.

ELVER SENA SOSA

Elver Sena Sosa es un arquitecto de soluciones de infraestructura de centro de datos, con más de 20 años de experiencia. Luego de una temporada como maestro de matemáticas de secundaria/liceo, Elver se trasladó a el área de redes, obteniendo la certificación de CCIE #7321.

Aburrido de estar haciendo lo mismo varios años, Elver descubrió el mundo de la virtualización. Determinado a descubrir qué clase de brujería era esa vaina de vMotion, Elver dejó el mundo físico y se fue con todo su espíritu al mundo virtual, donde aún reside.

Luego de obtener el VCDX #154, Elver se ha interesado en entender mejor cómo posicionar la infraestructura (redes, almacenamiento, seguridad, alta disponibilidad) para ofrecer servicios más relevantes a los clientes.

Como todo en la vida, Elver va retornando cada vez más a sus raíces como maestro. Cuando Elver no está ayudando a los clientes a hacer mejor uso los servicios de infraestructura, pueden encontrarlo frente a un pizarrón (físico o virtual) compartiendo lo poco que sabe.

AGRADECIMIENTOS Y DEDICATORIA

Estoy super honrado de haber sido integrado a este tremendo grupo de expertos hispanohablantes para dar un poco de apoyo al Proyecto “Cloud por vExperts”.

Quiero dedicarle mi aporte a toda mi gente linda latina de tecnología.

Me pueden encontrar (de vez en cuando) en

Twitter [@ElverS_opinion](https://twitter.com/ElverS_opinion)

LinkedIn <https://www.linkedin.com/in/elversenasosa>

¡Un abrazo y besos a todos!



JORGE TORRES

Original de Colombia, llevo más de la mitad de mi vida viviendo en Estados Unidos donde antes trabajé como inspector de calidad y control, profesión que cambié por el mundo de TI a mediados de los 2000's, con un enfoque en virtualización en la última década.

Durante mi tiempo aprendiendo y practicando TI con diferentes tecnologías, pude obtener certificaciones como VCP, VCAP y ser reconocido como **vExpert** en los últimos 4 años.

Partípice de la entusiasta comunidad informática vCommunity, VMUG, vBrownBag desde el año 2012, aprecio mucho los vAmigos y las oportunidades que desde ahí se han forjado.

Intento contribuir, participar y crear iniciativas que fortalezcan los lazos de amistad con eventos como vSoccer. Disfruto mucho de los deportes, tanto practicarlos como verlos y ~~sufrirlos~~ vivirlos 😊. En un próximo VMworld presencial, búscame si te gusta jugar al fútbol.

Me pueden encontrar en:

Twitter: https://twitter.com/j_kolkes

Blog: <https://www.kolkes.com/>

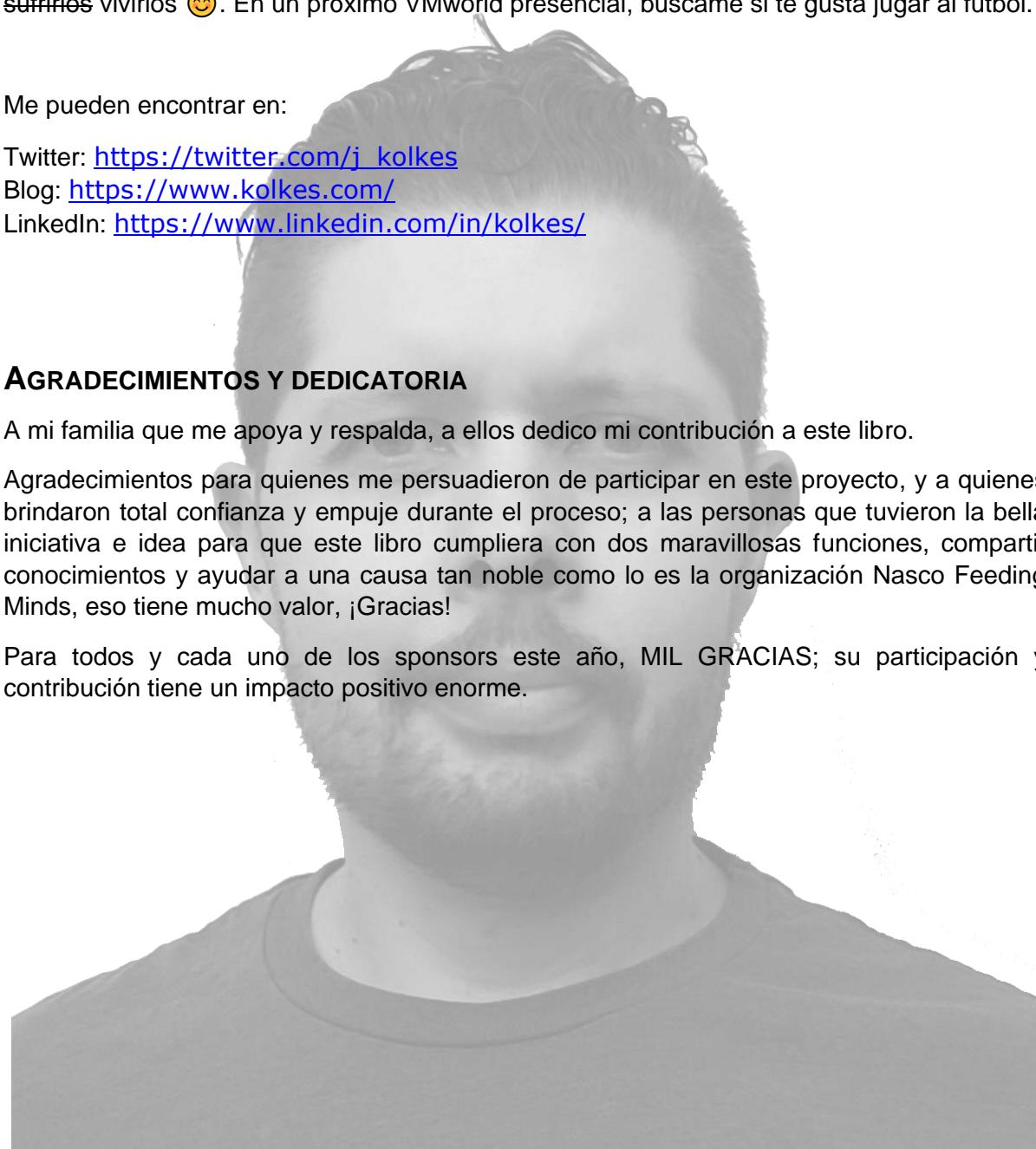
LinkedIn: <https://www.linkedin.com/in/kolkes/>

AGRADECIMIENTOS Y DEDICATORIA

A mi familia que me apoya y respalda, a ellos dedico mi contribución a este libro.

Agradecimientos para quienes me persuadieron de participar en este proyecto, y a quienes brindaron total confianza y empuje durante el proceso; a las personas que tuvieron la bella iniciativa e idea para que este libro cumpliera con dos maravillosas funciones, compartir conocimientos y ayudar a una causa tan noble como lo es la organización Nasco Feeding Minds, eso tiene mucho valor, ¡Gracias!

Para todos y cada uno de los sponsors este año, MIL GRACIAS; su participación y contribución tiene un impacto positivo enorme.



RAÚL UNZUÉ

¡Hola! Me llamo Raúl Unzué y tengo 40 años.

Soy emprendedor, curioso e inquieto. Cada día intento aprender, desarrollar y aplicar soluciones basadas en tecnología para traer el futuro al presente. O como era el eslogan de mi empresa como freelance, “hacer simple lo complejo” 😊

Durante estos años me he especializado en Virtualización (VMware y Citrix principalmente). Reconocido los últimos 8 años consecutivos con el premio VMware **vExpert** y en 2020 vExpert Pro.

He desarrollado durante más de 20 años diferentes puestos de trabajo (Responsable regional de Operaciones Técnicas (COO) en empresa TIC, Consultor Freelance, Analista de sistemas, Consultor SEO, Project Manager, Diseñador web, Especialista Infraestructuras, Linux o Comunicaciones, SysAdmin, Formador, fotógrafo certificado Google Street View... y como me gusta ponerme retos, aprendiz de DevOps.

Adicionalmente, he tenido la suerte de participar como coautor del ebook “VMware por vExperts”, por una causa benéfica. ¡Más de 19000 descargas y +27000€ recaudados!

¿Quieres saber más de mí? Visita mi Blog ☺

Me podéis seguir en mi blog <https://www.maquinasvirtuales.eu/>

Linkedin <https://www.linkedin.com/in/ra%C3%BAl-unzu%C3%A9-pulido-b11a4b48/>

Twitter <https://twitter.com/elblogdenegu>

Facebook <https://www.facebook.com/elblogdenegu>

DEDICATORIA Y AGRADECIMIENTOS

Se lo dedico a los que siempre están ahí para ayudarme a revisar el capítulo, aguantarme, animarme, a los que se alegran de mis logros y a los que no (¡jajig yo alguno habrá jeje!!)

Este año, una neuritis y el amigo Covid, me hicieron más difícil generar el contenido de este libro. Así que, se lo dedico en especial a Naiara y Vanesa, que son las que me cuidaron cuando peor estaba. Aunque parezca que no, vamos viendo la luz...

Agradecer por último a los sponsors que, como nosotros, ven en el libro una forma de ayudar a la comunidad y que, con sus donaciones, hacen posible que el libro tenga sentido, y podamos poner nuestro granito de arena por una buena causa.

CLOUD POR vEXPERTS

Las páginas que estás a punto de leer son producto del esfuerzo de 16 bloggers, 16 personas que de forma totalmente desinteresada nos fusionamos en un proyecto común de compromiso con la comunidad, para compartir nuestros conocimientos y experiencias.

La satisfacción es plena al poder terminar ya un segundo proyecto solidario y, de forma adicional, funcionar como un nexo entre diferentes empresas y una ONG como NASCO feeding minds.

Al fin y al cabo, todo se trata de un trabajo en equipo, de colaborar con la comunidad, y de ser solidarios aportando 16 granitos de arena contra viento y pandemia.

Este año 2020 es un año muy difícil en muchos sentidos, mucha carga de trabajo, excesivas reuniones online, ansiedad, estrés y objetivos trastocados debido a esta inesperada realidad. No obstante, decidimos seguir adelante entre todos porque sabemos que la recompensa es enorme.

En esta segunda edición, conseguimos que la familia crezca incorporando a excelentes personas como lo son Celia, Jorge, Iván y Elver, a la vez que ampliamos horizontes, pero siempre con el denominador común que es nuestro idioma, sumados a las ganas de compartir y aprender.

No hay mayor satisfacción que poder regalar conocimiento en cualquier geografía, sin importar la situación económica y sin pedir nada a cambio. Y como si fuera poco, y gracias a los sponsors que se comprometieron con nuestro proyecto, también podemos ayudar a la ONG de Ousman Umar a dar una mejor educación en Ghana para intentar mejorar el futuro de mucha gente.

“Si das de comer sacias el hambre por un día. Si alimentas la mente, la saciarás por cien años”

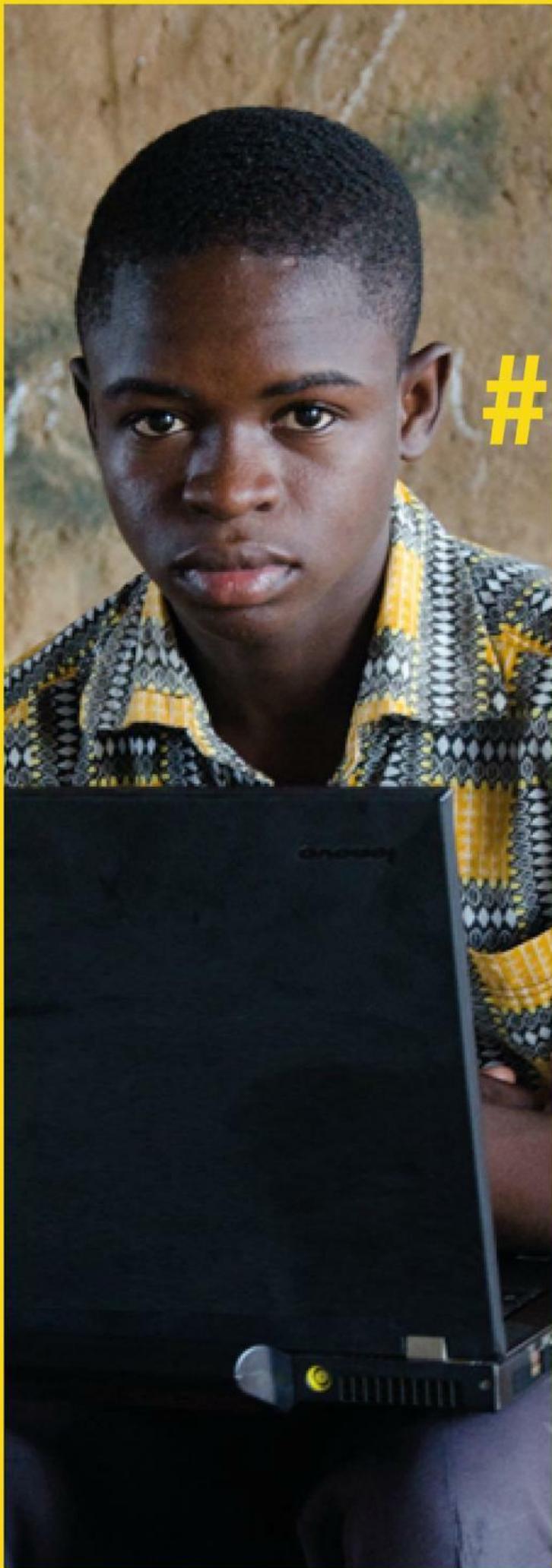
Ousman Umar, fundador de NASCO feeding minds

Simplemente esperamos que disfrutes de la lectura, que nos regales tu feedback, y que recuerdes que está en uno mismo aportar ese granito de arena de la forma que podamos.

Mi nombre es **Ousman Umar**. Crecí en la sabana africana, a los trece años atravesé el Sahara a pie y el mar en patera. Cuatro años más tarde llegué a España y, después de pasar tres meses durmiendo en la calle, una familia me acogió. En 2012 fundé **NASCO Feeding Minds**, una ONG que crea aulas informáticas en escuelas rurales de Ghana, para familiarizar a los niños con las herramientas digitales y facilitarles el acceso a la información.

Me gustaría agradecer a todos los autores y a los sponsors que han hecho posible que esa historia llegue a más personas, para poder dar a conocer esta realidad y salvar muchas vidas. Gracias por ayudarnos a cambiar la historia.





#CAMBIA LA HISTORIA

**"Si me alimentas
la barriga, sacias
mi hambre un
día, pero si
alimentas mi
mente, lo haces
para más de
cien años"**

NASCO
FEEDING
MINDS

PATROCINADORES

Gold Sponsors

veeAM

Qloudea
data solutions

openServices^{it}
eus

Silver Sponsors



Adistec

OVHcloud



EasyCloudFactory
Tu amigo de confianza en Cloud

Bronze Sponsors

ALTARO

EasyVirt



influxdata

Prólogo por Rick Vanover

PRÓLOGO

To all who read this eBook

You are taking the first step in bettering yourself. Knowledge is the key to making a positive change in your life and the world. The experts who have collaborated in this eBook are providing you technology information to empower your next move.

The experts who have contributed this book come from many backgrounds, many countries and many different stages in their IT careers. What is consistent is the drive to contribute to IT communities to feed hungry minds.

You will find that IT communities are an incredibly powerful network. They span geographies, they cross disciplines, they also allow the human race to put aside differences and align with technology to better the greater good. IT communities have that power.

By reading this book, you may not know it, but you are taking the first step towards engaging in an IT community. Whether it is the first IT community for you or a new IT community, you are taking that important first step. My advice to you is to maximize this opportunity. Engage with the authors on Twitter, read their blogs, learn a new area of technology.

Technical information here in this book around VMware, Veeam and topical content will advance your career. IT communities have made a positive impression on my career and I would not be where I am without that experience.

To the authors of this eBook, your tireless motivation to bring content to these hungry minds does not go unnoticed. This information is more important than you know, and the IT community thanks you for this effort.

Enjoy the Second Edition eBook!

Best Regards,

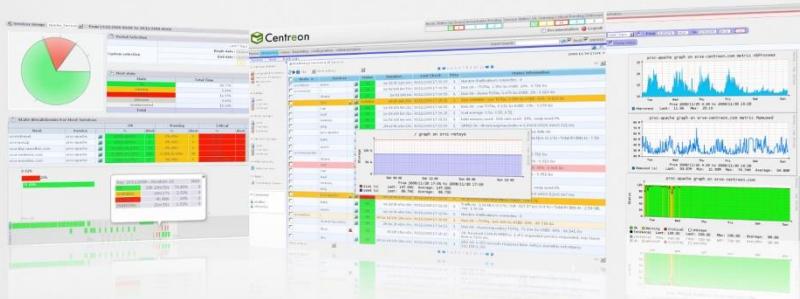


Rick W. Vanover *Microsoft VMP, VMware vExpert, Cisco Champion*
Senior Director, Product Strategy - Veeam Software
Twitter: [@RickVanover](https://twitter.com/@RickVanover)

ÍNDICE DE CAPÍTULOS

NOVEDADES DE VSphere 7.0.....	27
INSTALACIÓN Y CONFIGURACIÓN DE VSphere 7.0	46
ADMINISTRACIÓN DE ESXI DESDE LA LÍNEA DE COMANDOS.....	73
NSX-T	100
HORIZON 7: ACCESO EXTERNO Y CONSOLA HTML5	195
MONITORIZACIÓN DE NUEVA GENERACIÓN CON WAVEFRONT... MÁS ALLÁ DE VSphere.....	228
SITE RECOVERY MANAGER	268
VEEAM BACKUP & REPLICATION: NOVEDADES	328
CLOUD NATIVE APPS PARA ADMINISTRADORES DE VSphere.....	364
CLOUD NATIVE APPLICATIONS Y SU IMPACTO EN LAS PLATAFORMAS DE INFRAESTRUCTURA	396
KUBERNETES DESDE CERO.....	422
TANZU MISSION CONTROL	535
NSX-T Y MICROSERVICIOS	550
DEL DATACENTER FÍSICO A LAS NUBES	574
COMUNIDAD EN TI - VMUG, VEXPERT, VBROWNBAG, SOCIAL NETWORKS	600
EL ESTRÉS: UNA CONSTANTE EN TI.....	611

openServices^{it}_{.eus}



[Video demo](#)

Monitoriza tu entorno

Descansa & tenlo todo controlado

Expertos en monitorización

Monitorización de negocio

Da movilidad a tu negocio

Trabaja desde cualquier sitio

Desde cualquier dispositivo

De forma segura



Somos distintos

Trato cercano

Honestos



[Citrix para administradores de IT](#)

Descarga gratis
eBook 400pág.



Capítulo 1

NOVEDADES DE VSphere 7.0



Héctor Herrero

@nheobug

NOVEDADES DE VMWARE VSHERE 7

En este primer capítulo del libro Cloud por vExperts vamos a analizar y repasar lo que son novedades de producto dentro de la plataforma VMware vSphere. Siendo como todos conocemos el corazón de nuestras infraestructuras de virtualización, pero como iremos viendo y avanzando en el libro, ni mucho menos será lo novedoso. Ya que esta gran esperada versión nos va a proporcionar nuevas herramientas y soluciones para gestionar nuestros centros de datos definidos por software.

Cómo todos sabemos, las tecnologías avanzan, no sólo con nuevas funcionalidades o mejoras, sino que también la manera en que son consumidas por los usuarios. Por fin vamos entrando en la época donde todo comienza a ser digital, cada vez tenemos más recursos y podemos hacer las cosas de manera diferente a lo tradicional, incluso con valores añadidos. Estamos en la era dorada del software, cada vez tenemos más aplicaciones que nos ayudan con el desempeño diario, y VMware se posiciona para facilitar su disponibilidad y servicios para las nubes híbridas modernas.

Evolucionamos, optimizamos los recursos, securizamos y agilizamos los accesos, llegan otras maneras de entregar dichos recursos a los usuarios finales; ahí está el grueso de esta nueva edición, una nueva solución que cubre todas estas necesidades. Cómo no, fundamentalmente la presencia más notable del mundo Cloud, en aportar soluciones robustas que se integren, permiten escalabilidad, en definitiva, que nos den más flexibilidad a la hora de trabajar con entornos on-premises, híbridos o multi-cloud. Por supuesto que una de las grandes novedades de esta nueva versión, es cubrir la carencia que existía en cuanto al mundo de las modern apps bajo contenedores, ahora con Tanzu podremos finalmente gobernar de manera nativa Kubernetes, y sus contenedores, simplificando su entrega y su gestión. Destacar por último que NSX-T evoluciona, sustituyendo NSX-V, diseñado para abordar muchos de los casos de uso para los que no se diseñó NSX-V, como los multi-hipervisores, los contenedores o la nube pública entre otros. Es por ello recomendable leer cada capítulo del libro, donde no sólo aprenderemos de distintas tecnologías, sino que podremos comprender hacia dónde está yendo el mercado IT y qué opciones tenemos para enfrentarnos a él.

VMware pone el foco en ayudarnos a gestionar el tan importante ciclo de vida, con herramientas que nos facilitarán nuestra calidad de vida, permitiendo gestionar las actualizaciones, aplicar parches o asegurarnos que disponemos de configuraciones idénticas entre otros. Ofreciendo además una seguridad intrínseca mediante vSphere Trust Authority y Identity Federation. Y, por último, comentaremos las mejoras existentes en DRS y vMotion a la hora de acelerar nuestras cargas de trabajo.

Así que, lo comentado anteriormente, en este capítulo trataremos las principales novedades de la parte VMware vSphere 7. Hablaremos de los cambios que hay, de nuevas funcionalidades, y nuevas maneras de desempeñar nuestra labor.

vSPHERE LIFECYCLE MANAGER

Empezamos si os parece bien presentando vSphere Lifecycle Manager, una nueva herramienta para la gestión del ciclo de vida de nuestros datacenters. Sustituyendo y ampliando las capacidades de nuestro querido y ya antiguo vSphere Update Manager, lo usaremos para actualizar cualquier componente de nuestra infraestructura.

VMware cambia el modelo de gestionar el versionado de nuestros componentes, no sólo podremos (como hasta ahora) gestionar los parches o upgrade de nuestros hipervisores, sino que también añade la posibilidad de parchear y actualizar nuestro vCenter Server Appliance. Mucho más interesante es también la posibilidad de centralizar desde aquí dicha gestión sobre el firmware o drivers del hardware físico. Recordad que, hasta ahora, durante los upgrade, era bastante tedioso tener que controlar manualmente esto, así como tener que andar verificando matrices de compatibilidad entre los distintos productos.

The screenshot shows the vSphere Lifecycle Manager interface. At the top, there's a navigation bar with tabs: Resumen, Supervisar, Configurar, Permisos, Hosts, Máquinas virtuales, Almacenes de datos, Redes, and **Actualizaciones**. Below the navigation bar, there's a sidebar with a dropdown menu set to 'Hosts' and a list of options: Líneas base, **Imagen**, VMware Tools, and Hardware de máquina virtual. The main content area has a title 'Administrar con una sola imagen' (Manage with one image). It explains that Lifecycle Manager allows all hosts in a cluster to inherit the same image, eliminating variability between hosts. It also mentions that using a single image allows for faster updates, improved reliability, and easier maintenance. Below this text, there's a section titled 'Requisitos previos del host' (Prerequisites for the host) with a bulleted list: 'Los hosts deben ejecutar ESXi 7.0 o una versión posterior', 'Todos los hosts del clúster deben ser del mismo proveedor', and 'Los hosts deben tener estado'. At the bottom of the main content area are two buttons: 'CONFIGURAR IMAGEN' (Configure image) and 'IMPORTAR IMAGEN' (Import image).

Con todo ello, obtendremos una gestión más sencilla de todo lo que compone nuestro entorno virtual y, sobre todo, sufriremos menos a la hora de distribuir parches de seguridad o cuando nos toque actualizar el versionado.

vCENTER SERVER UPDATE PLANNER

The screenshot shows the 'Update Planner' section of the vSphere Lifecycle Manager. It displays a table of available updates with columns for Release Date, Version, Build, Type, Severity, Reboot Required, and Release Notes. A blue button labeled 'GENERATE REPORT' is visible below the table. At the bottom, there are two buttons: 'Interoperability' and 'Pre-Update Checks'.

Release Date	Version	Build	Type	Severity	Reboot Required	Release Notes
11/18/2019	6.7.0.42000	15132721	Update	Moderate	No	Link
09/27/2019	6.7.0.41000	14836122	Update	Moderate	No	Link
07/08/2019	6.7.0.40000	14367737	Update	Moderate	No	Link
07/02/2019	6.7.0.32000	14070457	Update	Moderate	No	Link

Dentro de vSphere Lifecycle Manager disponemos de vCenter Server Update Planner, que nos proporciona herramientas nativas para ayudarnos a planificar, descubrir y actualizar nuestros entornos. Siendo fácil de usar, intuitivo y todo ello integrado en la misma GUI. Nos permitirá diseñar la estrategia a la hora de desplegar upgrade/parches, con el objetivo de dejar un entorno homogéneo y actualizado.

The screenshot shows the 'Pre-Update Checks' section of the vSphere Lifecycle Manager. On the left, there is a sidebar with navigation links: Summary, Monitor, Configure, Permissions, Datacenters, Hosts & Clusters, VMs, Datastores, Networks, and a 'vCenter Server' dropdown menu which is currently set to 'Update Planner'. Below this is a 'Hosts' dropdown menu with options: Images, Baselines, VMware Tools, and VM Hardware. The main area displays a table titled 'Pre-Update Checks' with columns for 'Result' and 'Description'. There are three rows: 1) An 'Error' row stating 'Cannot collect component requirements. For more details check out the server logs' with a resolution note about checking logs and collecting support bundles. 2) A 'Warning' row stating 'The component 'VMware vCenter Server High-Availability' precheck will be skipped.' with a similar resolution note. 3) Another 'Warning' row for 'vCenter External Extensions' with a note to ensure compatibility and re-register extensions after upgrade. A blue 'EXPORT' button is located at the top left of the table area.

Result	Description	Resolution
⚠ Error	Cannot collect component requirements. For more details check out the server logs	For more information check the VMware logs. Please search for these symptoms in the VMware Knowledge Base for any known issues and possible resolutions. If none can be found, collect a support bundle and open a support request.
⚠ Warning	The component 'VMware vCenter Server High-Availability' precheck will be skipped.	For more information check the VMware logs. Please search for these symptoms in the VMware Knowledge Base for any known issues and possible resolutions. If none can be found, collect a support bundle and open a support request.
⚠ Warning	vCenter External Extensions	Please ensure extensions are compatible with the new vCenter Server and re-register extensions with the new vCenter Server after upgrade. Please refer to the vSphere documentation on extensions, and the upgrade and interoperability guides.

Recibiremos notificaciones en vSphere Client cuando hay actualizaciones listas para nuestros equipos, pudiendo además simular escenarios tipo “qué pasa si”, ejecutando una serie de chequeos pre-update... y, por último, pero no menos importante, es recordaros que tenemos la posibilidad de generar informes y generar reportes con los datos de nuestros equipos.

Interoperability ⓘ			
 Before upgrading, check the compatibility of 3rd party products that are registered with vCenter Server.			
MODIFY PRODUCT LIST		EXPORT	
Product	Current Version	Compatible Version(s)	Release Notes
ⓘ VMware vCloud Director	9.7	No compatible version	Not Available
ⓘ VMware vSphere Hypervisor (ESXi) (5)	7.0.0	No compatible version	Not Available
ⓘ VMware vRealize Automation	7.5.0	<u>8.0.0</u> ▾	Not Available
ⓘ VMware vRealize Log Insight	4.8.0	<u>8.0.0</u> ▾	Link
ⓘ VMware vRealize Operations Manager	6.7.0.000000	<u>7.0.0</u> ▾	Link
ⓘ Hybrid Cloud Extension (HCX)	3.5.1	3.5.1	Not Available
6 items			

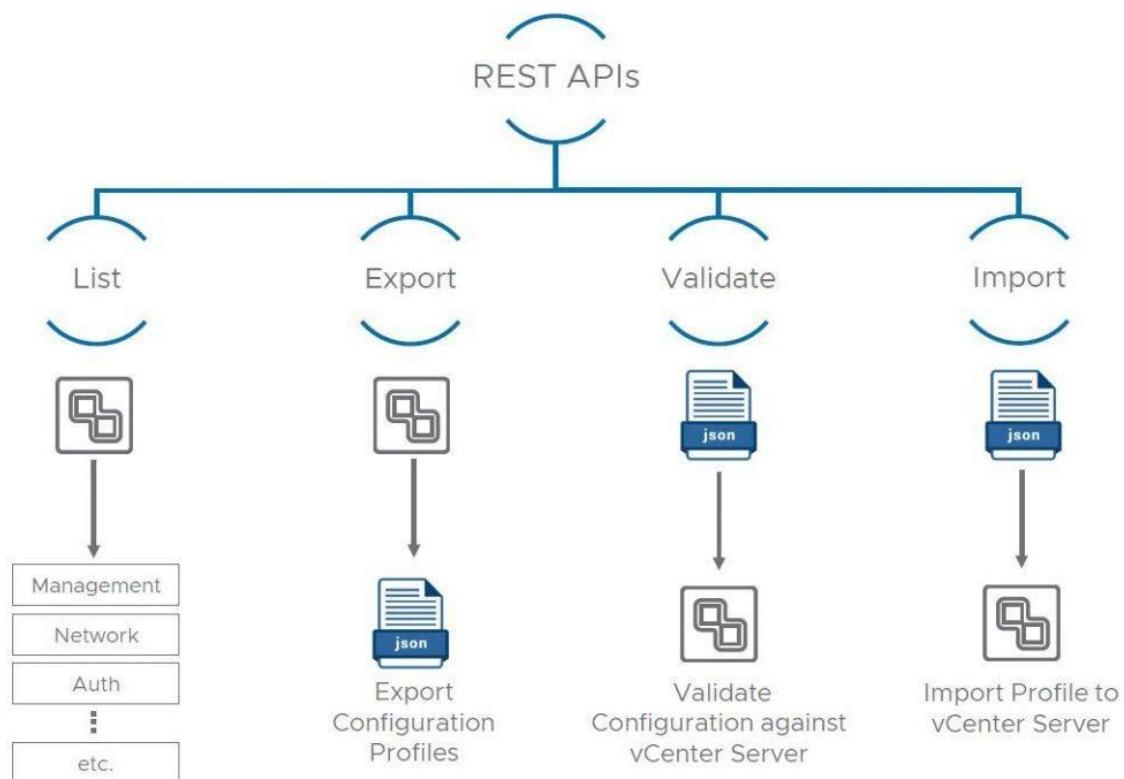
El maravilloso asistente de Interoperability, como comentamos nos hará la vida más fácil a la hora de subir de versionado, ya que nos mostrará las dependencias que tenemos con otros productos de VMware, así de fácil. Nos indicará si nuestro versionado es compatible o no, así como, nos ofrecerá enlaces a las distintas Release Notes de cada producto, donde tendremos más información.

NEXT-GEN INFRASTRUCTURE IMAGE MANAGEMENT

Next-Gen Infrastructure Image Management es dentro de Lifecycle Manager una nueva funcionalidad que nos permitirá gestionar las imágenes de infraestructura de nuestros hosts ESXi para poder aplicar parches o actualizar clústeres mediante el modelo deseado. Nos permitirán aplicar configuraciones en los hosts, así como monitorizarlas asegurando de que las cumplen y disponen todos ellos de la configuración correcta.

vCENTER SERVER PROFILES

En vSphere 7 llega vCenter Server Profiles, una nueva funcionalidad que nos permitirá disponer de un entorno homogéneo y consistente. Gracias a vCenter Server Profiles podremos tener la configuración de nuestros diferentes vCenter Servers controlada y homogénea, esto es, podremos saber si nuestros vCenter disponen de la configuración debida o disponen de alguna configuración que no es igual al resto. No debemos confundirlo con los Host Profiles que, como sabemos aplicarían a hosts, en este caso hablamos únicamente de vCenter Server. Pensado quizás para entornos donde tenemos varios vCenter Server, bien en el mismo sitio o en delegaciones separadas (o datacenters), podremos exportar la configuración o importar la configuración de manera global o de manera individual; esto es, como son los parámetros de red, la definición de la autenticación o sus usuarios, licencias, bibliotecas de contenido, certificados.



Todo esto lo haremos mediante 4 API's que nos permitirán: listar, exportar, validar o importar la configuración; como decimos, de manera global o de manera individual. También tendremos en cuenta, que los servidores vCenter Server, se podrán suscribir a otros servidores vCenter para obtener y mantener su configuración como es debido. Además, estas configuraciones podremos exportarlas y trabajarlas en formato JSON, así como podremos desplegarlas hasta en 100 distintos servidores vCenter. Una nota muy importante, es que en el proceso de Validate o en el de Import, nos comprobará si la configuración es correcta y se puede aplicar en el vCenter destino seleccionado.

CONTENT LIBRARY

Como no, en las Bibliotecas de contenido o Content Library tenemos novedades, algo que surgió en la versión 6.0 y ha ido mejorando bastante hasta la 6.7 donde hasta ahora lo conocemos como un repositorio de Plantillas de VMs, Scripts, Imágenes ISO o ficheros de texto.

Una de las pegas que tenía la gestión de las Plantillas, es que el proceso de actualización era manual y complicado si no trabajamos de manera ordenada. Ahora, VMware nos da la solución mediante el uso de VM Template Management, una gestión más intuitiva y sencilla para nuestras plantillas, con información y comentarios.

Con ello, controlaremos mejor el versionado de las plantillas, así como su mantenimiento diario o control de solicitudes por cada plantilla, pudiendo tener mayor visibilidad de qué pasa con ellas. Mediante Check-Out podremos buscar y desplegar el template; y mediante el Check-In grabaremos los cambios realizados en la nueva versión, sin tener que registrar nuevos Templates.

The screenshot shows the VMware vSphere Web Client interface for managing VM templates. At the top, there's a navigation bar with tabs: Summary (which is selected), Monitor, Configure, Permissions, Datastores, Versioning, and Updates. Below the navigation bar, there's a summary card for an 'APP-Server (3)' VM. The card displays the following details:

- Guest OS: CoreOS Linux (64-bit)
- Compatibility: ESXi 6.0 and later (VM version 11)
- VMware Tools: Not running, not installed
- More info
- DNS Name:
- IP Addresses: 10.173.184.15
- Host:
- Managed By: BCN Publisher

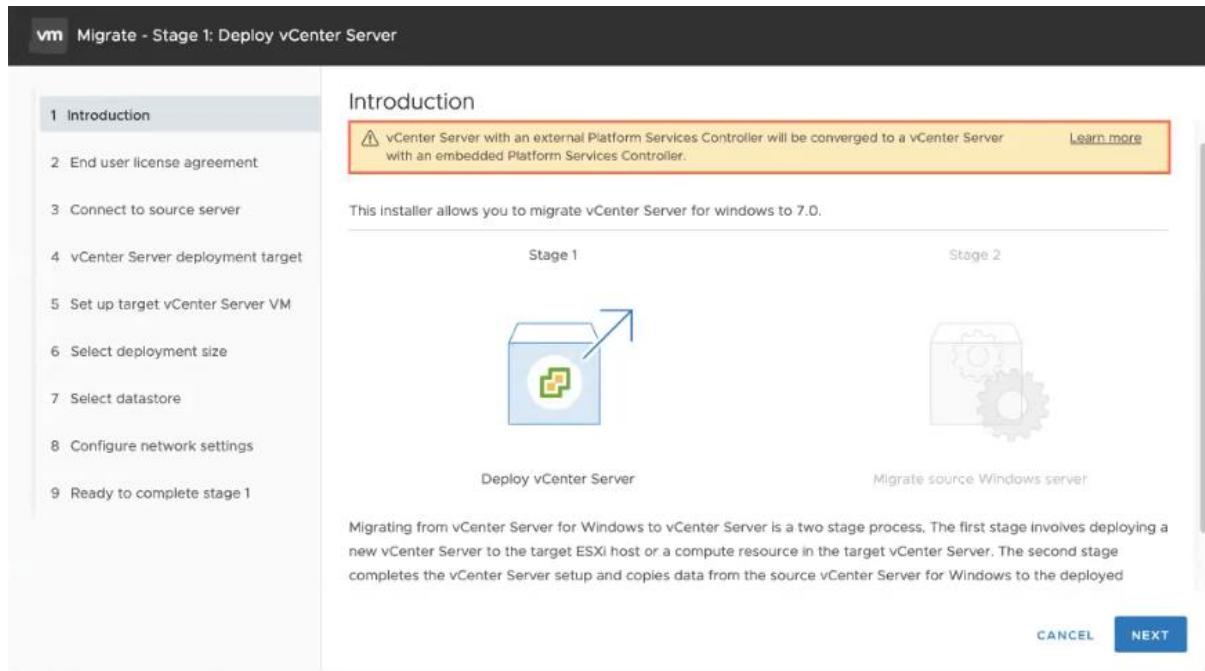
On the right side of the card, there's a storage usage indicator showing 872 MB. A 'SWITCH TO NEW VIEW' button is located at the top right of the card. Below the summary card, there are two main sections: 'Versioning' and 'VM Hardware'. The 'Versioning' section shows two versions of the 'APP-Server' template:

- APP-Server (3)**: Created on 02/15/2020, 1:00:21 PM by VSPHERE.LOCAL\Administrator. This version has a note: 'Added 2nd NIC. Edited RAM to 6GB from 4GB. Applied all OS patches. - VI Admin'. It includes buttons for 'CHECK OUT VM FROM THIS TEMPL...' and 'Delete Version'.
- APP-Server (2)**: Created on 02/15/2020, 12:55:59 PM by VSPHERE.LOCAL\Administrator. This version has a note: 'Application Server Template ...'. It includes buttons for 'Revert to This Version' and 'Delete Version'.

The 'VM Hardware' section shows a table for 'Tags' with three columns: Assigned Tag, Category, and Description. The table is currently empty, displaying 'No items to display'. There are also buttons for 'Assign...' and 'Remove...'.

vCENTER SERVER 7

Con la nueva versión de vCenter Server, debemos recordar que ya no se soportan arquitecturas con PSC's externos; así que, a la hora de actualizar nuestra infraestructura, estos se convertirán en vCenter Servers embedded, todo junto. Para ello, VMware nos proporcionará a la hora de actualizar la posibilidad de convertirlos de manera automatizada y sin que tengamos que preocuparnos de más.



En esta ocasión nos encontraremos un entorno GUI 100% basado en HTML5, completamente rediseñado, consiguiendo un aspecto más fresco, con más información detallada y mucho más sencillo de encontrar los recursos. Además, podremos ver desde la GUI los updates pendientes que tiene el entorno, así como la fecha del último backup del vCenter Server.

Quiero comentar que, como suele ser habitual, en esta nueva versión también han escalado los máximos frente a la 6.7, si antes podíamos gobernar 2.000 hosts por cada vCenter Server, ahora podremos hasta los 2.500. Así como hasta ahora soportaba 25.000 máquinas virtuales encendidas, este número sube hasta las 30.000. Al igual que si utilizamos Enhanced Linked Mode para unir la gestión de distintos vCenter Server, ahora podremos controlar 15.000 hosts frente a los 5.000 de la anterior versión. También, con Enhanced Linked Mode sube de 50.000 a 150.000 VMs que podremos gestionar desde una única consola. Finalmente podemos observar que han subido la latencia máxima entre servidores vCenter Server, cuando hasta ahora si teníamos más de 100ms entre ellos era un entorno no soportado, ahora nos permitirá disponer de hasta 150ms, que quizá en muchos sitios, esto era un problema.

vSPHERE DEVELOPER CENTER

Ahora, en la GUI totalmente integrada, disponemos de un nuevo sitio, un centro para perfiles de tipo desarrollador o perfiles DevOps, un lugar donde podremos trabajar con API's. Siendo ideal para crear automatizaciones o tareas que requieran tiempo. Desde este sitio, podremos también capturar cualquier operación que hacemos desde vSphere Client, con idea de que podamos obtener códigos para generar nuestros propios scripts.

No sé si lo recordaréis, pero la idea está basada en el mítico fling de VMware llamado Onyx que nos daba todos los comandos en vSphere PowerCLI de todo lo que hacíamos con el ratón. Bueno, pues ahora a parte de PowerCLI, también podremos obtener el código en vRO Javascript, en Go o en el maravilloso Python. Tenemos también un API Explorer, que nos permitirá de manera más sencilla el poder conocer la estructura para poder interactuar mediante llamadas REST API y gestionar de otra manera nuestro ambiente de vSphere.

The screenshots illustrate the vSphere Client Developer Center interface, specifically the API Explorer and Code Capture tabs.

API Explorer Tab:

- The left sidebar shows the "Developer Center" section under "Developer Center".
- The main area displays the "API Explorer" tab, which includes a "Select Endpoint" dropdown set to "upg-dhcp-l570-vm-078.cpbl.lab" and a "Select API" dropdown showing options like "vcenter", "cis", "stats", "appliance", "vapi", "esx", and "content".
- A tooltip is visible over the "Select API" dropdown.

Code Capture Tab:

- The left sidebar shows the "Developer Center" section under "Developer Center".
- The main area displays the "Code Capture" tab, which includes a message "Your session has been recorded.", an "Enable Code Capture" toggle switch (which is turned on), and three buttons: "CLEAR AND START ANOTHER", "STOP RECORDING", and "COPY".
- The "Language" dropdown is set to "PowerCLI".
- The code editor displays a PowerShell script for cloning a VM:

```
1 #----- Start of code capture -----
2
3 #-----ListKmipServers-----
4 $this = Get-View -Id 'CryptoManagerKmip-CryptoManager'
5 $this.ListKmipServers($null)
6
7 #-----CheckClone_Task-----
8 $vm = New-Object VMware.Vim.ManagedObjectReference
9 $vm.Type = 'VirtualMachine'
10 $vm.Value = 'vm-36'
11 $folder = New-Object VMware.Vim.ManagedObjectReference
12 $folder.Type = 'Folder'
13 $folder.Value = 'group-v4'
14 $name = 'Test-VM-01'
15 $spec = New-Object VMware.Vim.VirtualMachineCloneSpec
16 $spec.Template = $false
17 $spec.PowerOn = $false
18 $spec.Location = New-Object VMware.Vim.VirtualMachineRelocateSpec
19 $spec.Location.Pool = New-Object VMware.Vim.ManagedObjectReference
20 $spec.Location.Pool.Type = 'ResourcePool'
21 $spec.Location.Pool.Value = 'resgroup-9'
22 $stestType[0] = 'sourceTests'
23 $stestType[1] = 'resourcePoolTests'
24 $stestType[2] = 'hostTests'
```

GESTIÓN DE CERTIFICADOS

The screenshot shows the vSphere Client interface with the following details:

- Header:** VM, vSphere Client, Menú, Buscar en todos los entornos.
- Left Sidebar (Administración):**
 - Control de acceso (Funciones, Permisos globales)
 - Licencias (Licencias)
 - Soluciones (Complementos del cliente, Extensiones de vCenter Server)
 - Implementación (Configuración del sistema, Programa de mejora de la experiencia...)
 - Soporte (Cargar archivo a la solicitud de servicio)
 - Single Sign On (Usuarios y grupos, Configuración)
 - Certificados (selected)
- Right Panel - Certificados:**
 - Administración de certificados**
 - Certificado SSL de máquina**
 - __MACHINE_CERT**
 - Válido hasta 20 ago. 2022
 - Clave privada y cadena de certificados
 - VER DETALLES** ACCIONES ▾
 - Renovar (highlighted)
 - Importar y reemplazar certificado
 - Generar solicitud de firma del certificado (CS...)
 - Certificados raíz de confianza** | AGREGAR
 - 04C3117305D335A...
 - Válido hasta 27 abr. 2022
 - Certificado
 - 05B2F555FE806F3...
 - Válido hasta 14 ago. 2030
 - Certificado

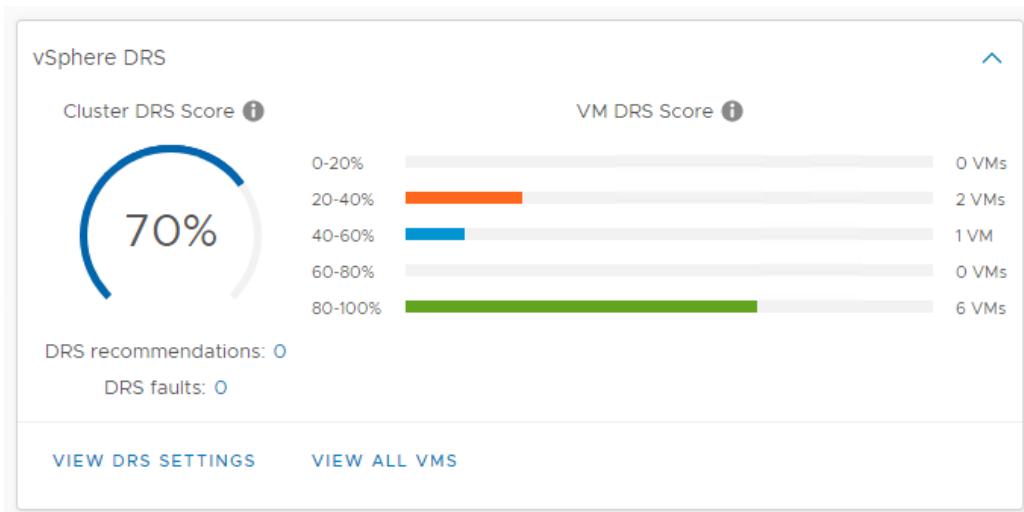
Otra de las grandes sorpresas y que muchos esperábamos, no es otra que la simplificación a la hora de gestionar los certificados de vSphere. Como muchos conocemos, hasta ahora esto venía siendo entre una tarea tediosa y una pesadilla; nos referimos a cuándo teníamos que instalar certificados en nuestro vCenter Server o en sus hosts.

Ahora, lo podremos hacer todo mediante sencillos pasos en la GUI, podremos fácilmente reemplazar los certificados. Así como importar certificados de entidades emisoras de certificados externos (o CA's externas), certificados existentes, así como generar solicitudes para las CA's.

DRS

DRS ha sufrido un lavado de cara también muy importante, aunque claro que se mantiene el concepto o idea como tal, pero cambia radicalmente el funcionamiento de lo que venimos conociendo como DRS (1.0).

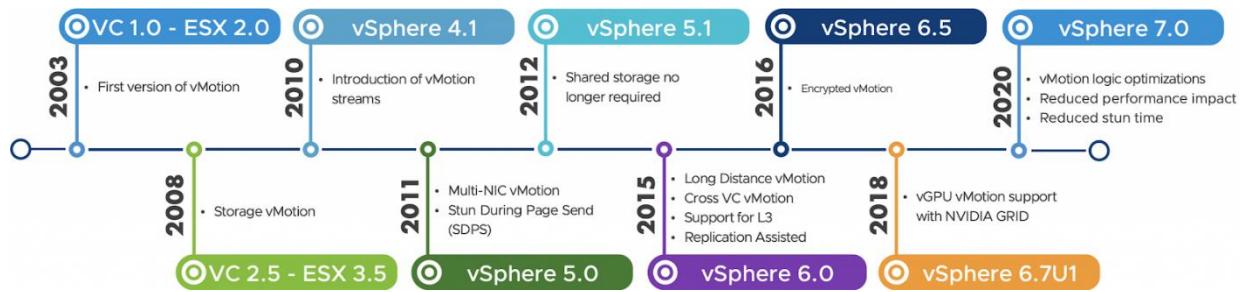
En esta nueva versión 7, aparte de soportar los contenedores, DRS está centrado en las cargas de trabajo, buscando siempre la eficiencia de cada máquina virtual; hasta ahora estaba basado en la carga del clúster intentando siempre tener equilibrada la carga de los hosts. Si recordamos, teníamos un nivel de carga en el clúster, y era la manera de visualizar la repartición de dicha carga, y el algoritmo nos recomendaba hacer vMotion cuando no estaba equilibrado. Ahora DRS se basará en puntuaciones por distintas métricas, por cada VM, llamado DRS Score, usando métricas interesantes como puedan ser el CPU Ready o Memoria Swap entre otros.



Por tanto, DRS se preocupará ahora por la felicidad de nuestras VMs, buscando la mayor eficiencia en cada una de ellas, sin olvidar que también DRS tiene en cuenta la capacidad de los ESXi, como hasta ahora. Algo que también tendrá en cuenta es el costo en cuanto a moverla mediante vMotion, si otro host ESXi puede proporcionar una puntuación más alta para la VM, el DRS considerará la migración.

Por último, cambia el periodo de programación de DRS, hasta ahora se ejecutaba cada 5 minutos, ahora lo hará cada 1 minuto, mejorando notablemente la respuesta.

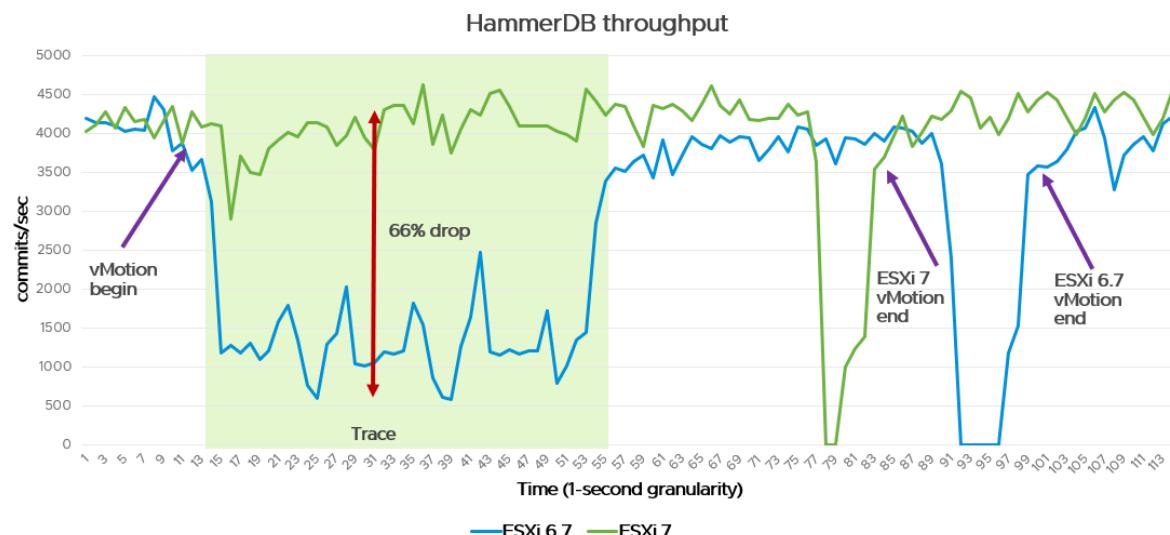
vMOTION



Cuánto hace que vivimos por primera vez aquello que parecía magia, aquella primera máquina virtual que se movió entre hosts en caliente, sin parada, cuánto ha llovido desde entonces. En esta ocasión, con vSphere 7, vMotion también ha sido mejorado, junto a su compañero DRS, vMotion ha sido adaptado para soportar las cargas de trabajo de hoy en día, ya que era conocido que ciertas máquinas virtuales con grandes recursos y uso intensivo de CPU o RAM sufrían la migración, tipo aplicaciones SAP o grandes BBDD,

Por tanto, máquinas con 128GB de RAM ya no serán problemáticas a la hora de moverse en caliente entre hosts. El objetivo ha sido evitar que caiga el rendimiento durante el proceso de migración de la VM, en vSphere 7, se utiliza una sola vCPU dedicada al rastreo de páginas, logrando así que la VM pueda seguir funcionando con total normalidad y no se vea afectada por el proceso de vMotion. Así como se ha mejorado también el proceso de copia de memoria, hasta ahora la memoria se transfería en páginas de 4k, y ahora con páginas de 1GB. Con todo ello se optimiza y se agiliza el movimiento de cargas en producción.

Recordad que ahora también es posible el realizar vMotion de manera totalmente cifrada, entre distintos clústeres de vCenter o entre nubes híbridas, eso sí, intercambiando el par de claves. Y, por último, pero no menos importante, es que tendremos nuevas plantillas EVC para ampliar el soporte con distintos hosts.



VM HARDWARE VERSIÓN 17

Con cada versión de vSphere, tenemos una nueva versión de hardware virtual, recordar que es algo que debemos actualizar también por cada VM, con objeto que disfrute de las nuevas funcionalidades. Y por supuesto lo haremos después de actualizar las VMware Tools en las máquinas virtuales.

Nueva máquina virtual

✓ 1 Seleccionar un tipo de cr... Personalizar hardware
✓ 2 Seleccionar un nombre y ... Permite configurar el hardware de la máquina virtual.
✓ 3 Seleccionar un recurso in...
✓ 4 Seleccionar almacenamiento...
✓ 5 Seleccionar compatibilid...
✓ 6 Seleccionar un sistema o...
7 Personalizar hardware
8 Listo para completar

Hardware virtual Opciones de máquina virtual

> CPU	2
> Memoria	4 GB
> Nuevo disco duro *	30 GB
> Nueva controladora SCSI *	LSI Logic SAS
> Nueva red *	VM Network
> Nueva unidad de CD/DVD *	Dispositivo cliente
> Nueva controladora USB	USB 3.1
> Tarjeta de video *	Especificar configuración personalizada
> Dispositivos de seguridad	Sin configurar
Dispositivo VMCI	
Nueva controladora SATA	Nueva controladora SATA

AGREGAR NUEVO DISPOSITIVO

Discos, unidades y almacenamiento
Disco duro
Disco duro existente
Disco RDM
Dispositivo USB de host
NVDIMM
Unidad de CD/DVD
Controladores
Controladora de NVMe
Controladora SATA
Controladora SCSI
Controladora USB
Otros dispositivos
Dispositivo PCI
Temporizador de Watchdog
Reloj de precisión
Puerto serie
Red
Adaptador de red

Compatibilidad: ESXi 7.0

CANCEL BACK NEXT

Bien, esta versión 17, con el Hardware Asignable permitimos asignar virtual hardware, tales como son gráficas NVIDIA y hasta 6 dispositivos PCIe con DirectPath IO. Si bien es cierto que ahora se gestiona de manera diferente, esto es, basado en reglas, dando flexibilidad a la VM, no asociándola al hardware específico de un host específico; sino hosts que proporcionen dicho hardware. Ahora HA y DRS tendrán en cuenta a la hora de mover una VM entre hosts, que las VMs siempre tengan acceso a dichos dispositivos. Por tanto, la idea es hacer adjuntar dinámicamente a las VMs los recursos basado en reglas para funcionar perfectamente con vMotion, es así como cuando se presenta un hardware de este tipo a la VM, se visualizarían todos los dispositivos disponibles por el clúster.

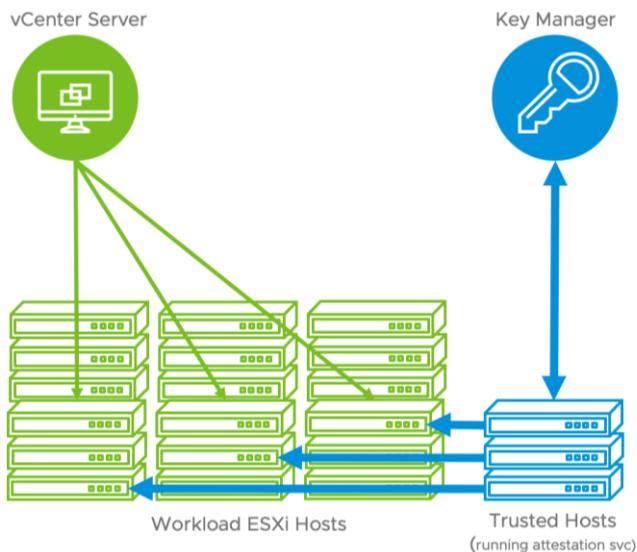
Tenemos la posibilidad de agregar un dispositivo de temporizador de Watchdog virtual (Virtual Watchdog Timer, VWDT) en las VMs. Nos permitirá monitorizar servicios críticos en VMs independientemente de su SO, ya que, si el SO deja de responder y no se recupera, Watchdog Timer podrá reiniciar el sistema tras un periodo de espera predefinido y solventar el problema. Es ideal para aplicativos en clúster o miembros de MSCS que puedan recuperarse de manera automatizada.

También vSphere 7 ofrece compatibilidad con PTP, podremos añadir un dispositivo de tipo PTP (Precision Time Protocol), que es un reloj de milisecondo con timestamp de milisecondo. Siendo una nueva opción para controlar el tiempo en aplicativos críticos que requieran una latencia mucho más baja que el tradicional NTP, normalmente usados en el sector financiero, científico, etc.

Tenemos aquí a modo de resumen las principales características del vHW:

Característica	7.0	6.7 U2	Característica	7.0	6.7 U2
Versión de Hardware virtual	17	15	USB 1.x y 2.0	N	S
Memoria máxima (GB)	6128	6128	USB 3.1 SuperSpeed	S	S
Número máximo de procesadores lógicos	256	256	USB 3.1 SuperSpeedPlus	S	N
Número máximo de cores (CPUs virtuales) por socket	64	64	Memoria máxima de video (MB)	128	128
Adaptadores SCSI máximos	4	4	Memoria máxima para gráficos 3D (GB)	4	2
Adaptadores Bus Logic	S	S	Pantallas SVGA	10	10
Adaptadores LSI Logic	S	S	Aceleración hardware SVGA 3D	S	S
Adaptadores LSI Logic SAS	S	S	VMCI	S	S
Controladoras VMware Paravirtual	S	S	PCI passthrough	16	16
Controladoras SATA	4	4	Dynamic DirectPath	S	N
Controladoras NVMe	4	4	Soporte PCI Hot add	S	S
Disco Virtual SCSI	S	S	Dispositivo Virtual Precision Clock	S	N
SCSI passthrough	S	S	Dispositivo Virtual Watchdog Timer	S	N
Soporte de SCSI hot add	S	S	Dispositivo Virtual SGX	S	N
Nodos IDE	S	S	Soporte Nested HV	S	S
Disco Virtual IDE	S	S	Soporte vPMC	S	S
Virtual IDE CD-ROM	S	S	Número máximo de puertos serie	32	32
Soporte IDE hot add	N	N	Puertos paralelos	3	3
Número máximo de NICs	10	10	Unidades de disquete	2	2
PCNet32	S	S	Virtual RDMA	S	S
VMXNet	S	S	Controladoras NVDIMM	1	1
VMXNet2	S	S	Dispositivos NVDIMM	64	64
VMXNet3	S	S	Virtual I/O MMU	S	S
E1000	S	S	Virtual TPM	S	S
E1000e	S	S	Microsoft VBS	S	S

vSPHERE TRUST AUTHORITY



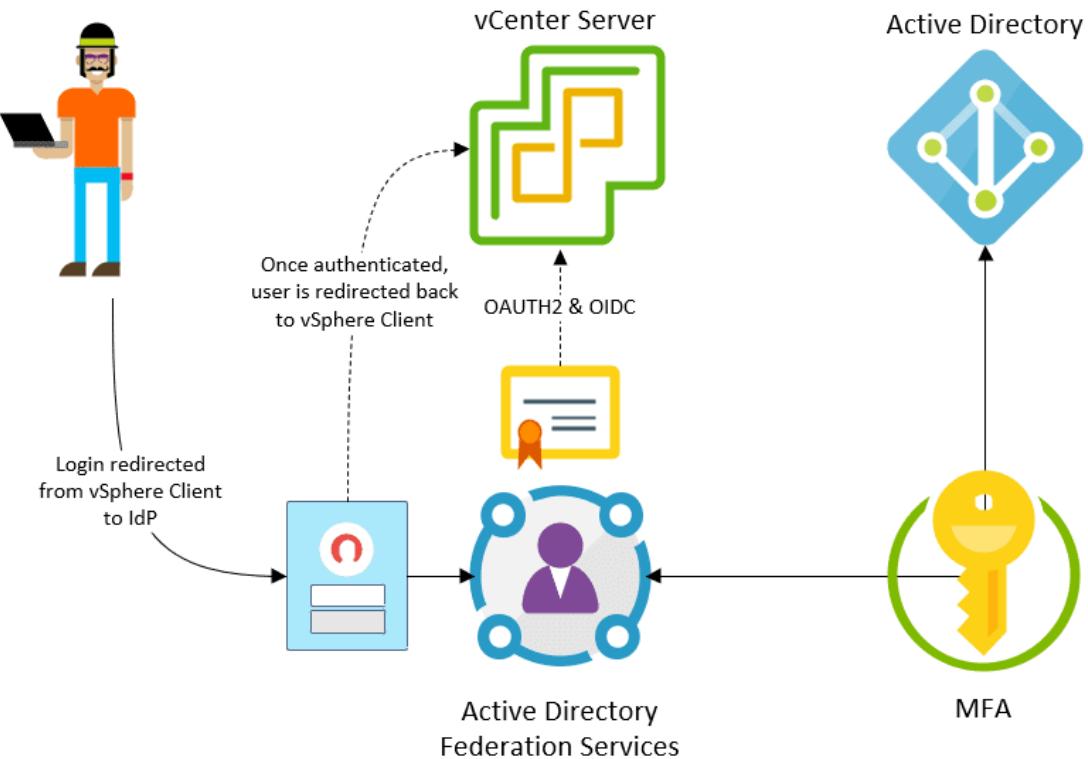
vTA, pieza tecnológica de VMware que presenta una certificación remota para cargas de trabajo confidenciales, asegurando que ciertos entornos cumplan unos mínimos de seguridad. Esto es, vSphere 7 nos asegura que podemos cumplimentar la integridad de nuestra infraestructura basándonos en una imagen de host de confianza, podremos realizar comprobaciones de seguridad en el resto de los hosts, con objeto de validar el SO, el firmware y las credenciales.

El hipervisor ESXi se apoya en los chips TPM (o Trusted Platform Module), éstos, son procesadores capaces de almacenar claves cifradas de datos confidenciales con el fin de proteger información, siendo un estándar en la industria de los procesadores seguros. Los chips TPM 2.0 aseguran la identidad de los hosts ESXi, avalan el estado del software del host, firmware, credenciales... y mediante arranque seguro UEFI, cargará únicamente software firmado; el chip registrará y almacenará de forma segura los módulos de software cargados por el sistema que el servidor vCenter Server validará; y que por supuesto podremos verificar en el vSphere Client el estado de confianza del hardware.

The screenshot shows the vSphere Client interface. The left sidebar has a tree view under 'vcenter-1.70.fcotr.org' with categories like Datacenter, Management, Workloads, and vCenter Server. The 'Monitor' tab is selected in the top navigation bar. The main content area is titled 'Security' and contains a table with the following data:

Name	Attestation	Last verified	TPM version
esx-1.70.fcotr.org	Passed	04/21/2020, 11:36 PM	2.0
esx-2.70.fcotr.org	Passed	04/21/2020, 11:36 PM	2.0
esx-3.70.fcotr.org	Passed	04/21/2020, 11:36 PM	2.0
esx-4.70.fcotr.org	Passed	04/21/2020, 11:36 PM	2.0
esx-5.70.fcotr.org	Passed	04/21/2020, 11:36 PM	2.0
esx-6.70.fcotr.org	Passed	04/21/2020, 11:36 PM	2.0
esx-7.70.fcotr.org	Passed	04/21/2020, 11:36 PM	2.0

IDENTITY FEDERATION



Con idea de mejorar y descargar el proceso de autenticación, vCenter Server ahora, puede integrarse con un proveedor de identidad empresarial externo. Una característica deseada por muchos para poder cumplimentar políticas de seguridad que hasta ahora no podíamos cubrir; algo interesante será la posibilidad de poder integrarlo con implementaciones de autenticación de múltiple factor o MFA (Multi-Factor Authentication); así como también el poder integrarlo en instalaciones de ADFS (Servicios de federación de Active Directory).

VSPHERE CON KUBERNETES

Hoy día VMware sigue siendo el líder indiscutible del mercado en infraestructuras on-premise, pero debido a la transformación digital, a la modernización de las aplicaciones. Estas aplicaciones están siendo reescritas o bien las nuevas aplicaciones ya son creadas para trabajar con tecnologías Cloud Native, trabajando sobre la base de la microsegmentación, y, esto significa que trabajan en contenedores, que algunos son serverless, otros tienen datos persistentes y trabajan de forma complementaria con Kubernetes, para orquestar lo que no es solamente el despliegue, balanceo, alta disponibilidad o escalabilidad. Esa es la idea, que las Apps sean fáciles de distribuir, automatizables, puedan escalar sin problemas y de forma granular.

Así que, al ser esta la base de las nuevas aplicaciones, cada día vamos a ir teniendo menos aplicaciones tradicionales o monolíticas que corran bajo máquinas virtuales. Sumado a esto la proliferación de las todas las cloud públicas, VMware se quiere posicionar como un proveedor de infraestructura híbrida para dar soporte tanto a máquinas virtuales como también a Cloud Native Apps, lo que serían, contenedores. A partir de vSphere 7, con lo que conocemos de vSphere, de Kubernetes hablamos del ESXi del futuro, que de forma nativa dará soporte a MVs y contenedores, esa es la gran diferencia; y no sólo eso, ya que también dando soporte nativo de Kubernetes, todo desde la misma GUI.

También como iremos leyendo en el libro, conocemos que han desarrollado de cero NSX-T, con la idea de que sea multihipervisor, adaptado al mundo de los contenedores y, además pueda trabajar con entornos multi cloud. VMware on Tanzu ofrece a sus clientes una plataforma de desarrollo, automatización, empaquetado, distribución, control, catálogo... de las modern apps.



qloudea

ALMACENAMIENTO
Y SEGURIDAD DE DATOS

BACKUP
NAS - SAN - NUBE

QLOUDEA.COM

vmware®

NUTANIX

Microsoft
Hyper-V



Windows
Server



Office 365

Capítulo 2

INSTALACIÓN Y CONFIGURACIÓN DE VSphere 7.0



Xavier Caballé

@screenshotsit

VMWARE ESXI 7.0: INSTALAR UN NUEVO SERVIDOR HOST.

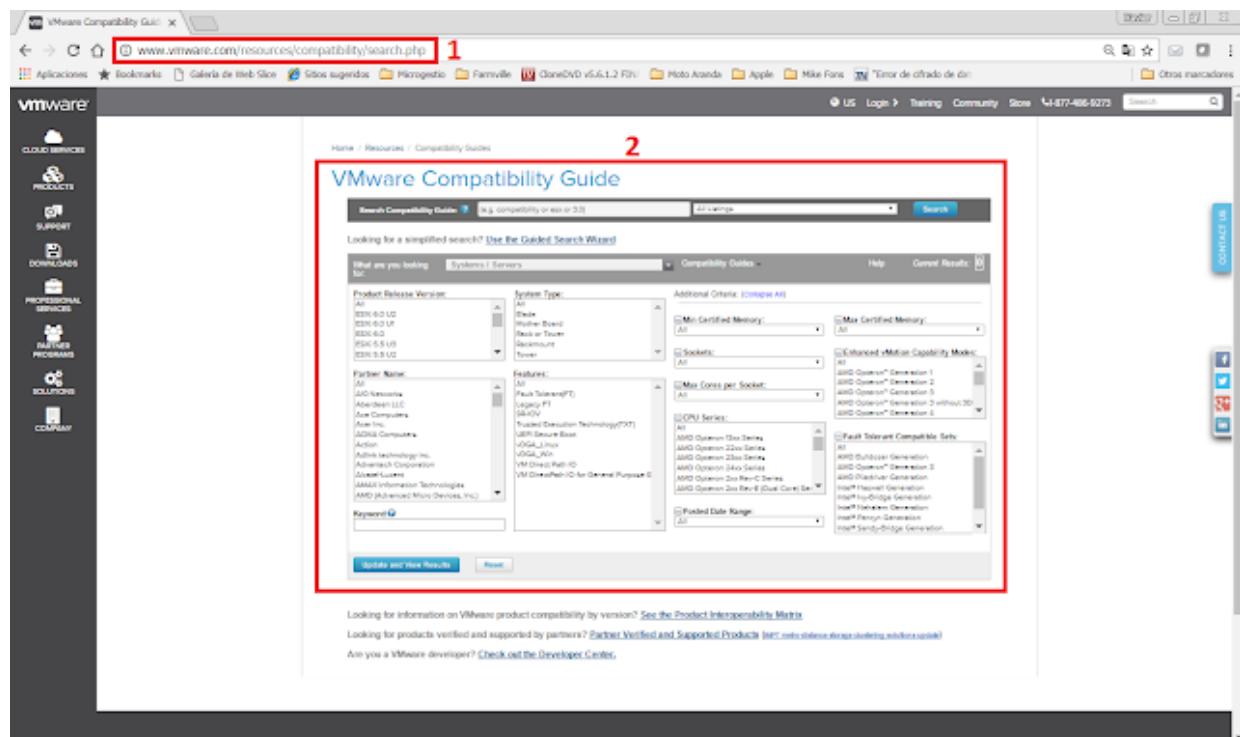
1. REQUISITOS PREVIOS

Lo primero que tenemos que hacer, antes de lanzarnos directamente a la instalación o actualización de un nuevo servidor host de virtualización basado en VMware, es comprobar que el hardware del que disponemos cumple con los requisitos mínimos de instalación marcados por el fabricante.

Para conseguir saber, si nuestros equipos son compatibles o no, con una versión concreta de VMware usaremos la herramienta llamada **VMware Compatibility Guide**.

Sabiendo previamente la marca y modelo de los nuestros servidores y también los modelos específicos de CPU, nos dirigiremos al enlace de la página oficial de VMware que mostramos a continuación:

VMware Compatibility Guide: <https://www.VMware.com/resources/compatibility/search.php>



Rellenaremos el formulario de VMware Compatibility Guide con los datos técnicos del hardware disponible en nuestra infraestructura. Los datos que vamos a usar en como ejemplo serán los siguientes:

- Partner Name: HP
 - System Type: Rackmount
 - CPU Series: Intel Xeon E5-2600v4 Series

- Enhanced vMotion Capability Modes: All

Seguidamente, pulsaremos el botón llamado **Update and view Results** y seleccionaremos el modelo de servidor de la marca **HP** que tenemos, en nuestro ejemplo se trata de un servidos **Proliant DL380e de la generación 8**.

Una vez tengamos el detalle de nuestro host en pantalla. Podremos comprobar en la columna llamada **Supported Releases**, que el hardware del que disponemos solo soporta como máximo la versión **ESXi 6.0 U2**. Así pues, no podemos instalar **vSphere 7.0** en el servidor que tenemos. El uso de esta práctica nos evitará muchos dolores de cabeza.

Home / Resources / Compatibility Guides

VMware Compatibility Guide

Search Compatibility Guide: [?](#)
All Listings
[Search](#)

Looking for a simplified search? [Use the Guided Search Wizard](#)

What are you looking for:
Compatibility Guides +
Help
Current Results: 18

Product Release Version:

- All
- ESXi 6.0 U2
- ESXi 6.0 U1
- ESXi 6.0
- ESXi 5.5 U3
- ESXi 5.5 U2

System Type:

- All
- Blade
- Mother Board
- Rack or Tower
- Rackmount
- Tower

Additional Criteria: [\(Collapse All\)](#)

- Min Certified Memory:
- Max Certified Memory:
- Enhanced vMotion Capability Modes: All
- Sockets:
- Max Cores per Socket:
- CPU Series: Intel Xeon E5-2400 Series
- Fault Tolerant Compatible Sets: All
- Posted Date Range:

1 **ESXi 6.0 U2**
2 **HP**
3 **Rackmount**
4 **Intel Xeon E5-2400 Series**
5 **Enhanced vMotion Capability Modes: All**
6 **Update and View Results**

Server Device and Model Information

The detailed lists show actual vendor devices that are either physically tested or are similar to the devices tested by VMware or VMware partners. VMware provides support only for the devices that are listed in this document.

Click on the 'Model' to view more details and to subscribe to RSS feeds.

Bookmark | Print | Export to CSV

Partner Name	Model	CPU Series	7	Supported Releases
HP	ProLiant DL380e Gen8	Intel Xeon E5-2400 Series	ESX	4.1 U3 4.1 U2

2. INSTALACIÓN PASO A PASO ESXI:

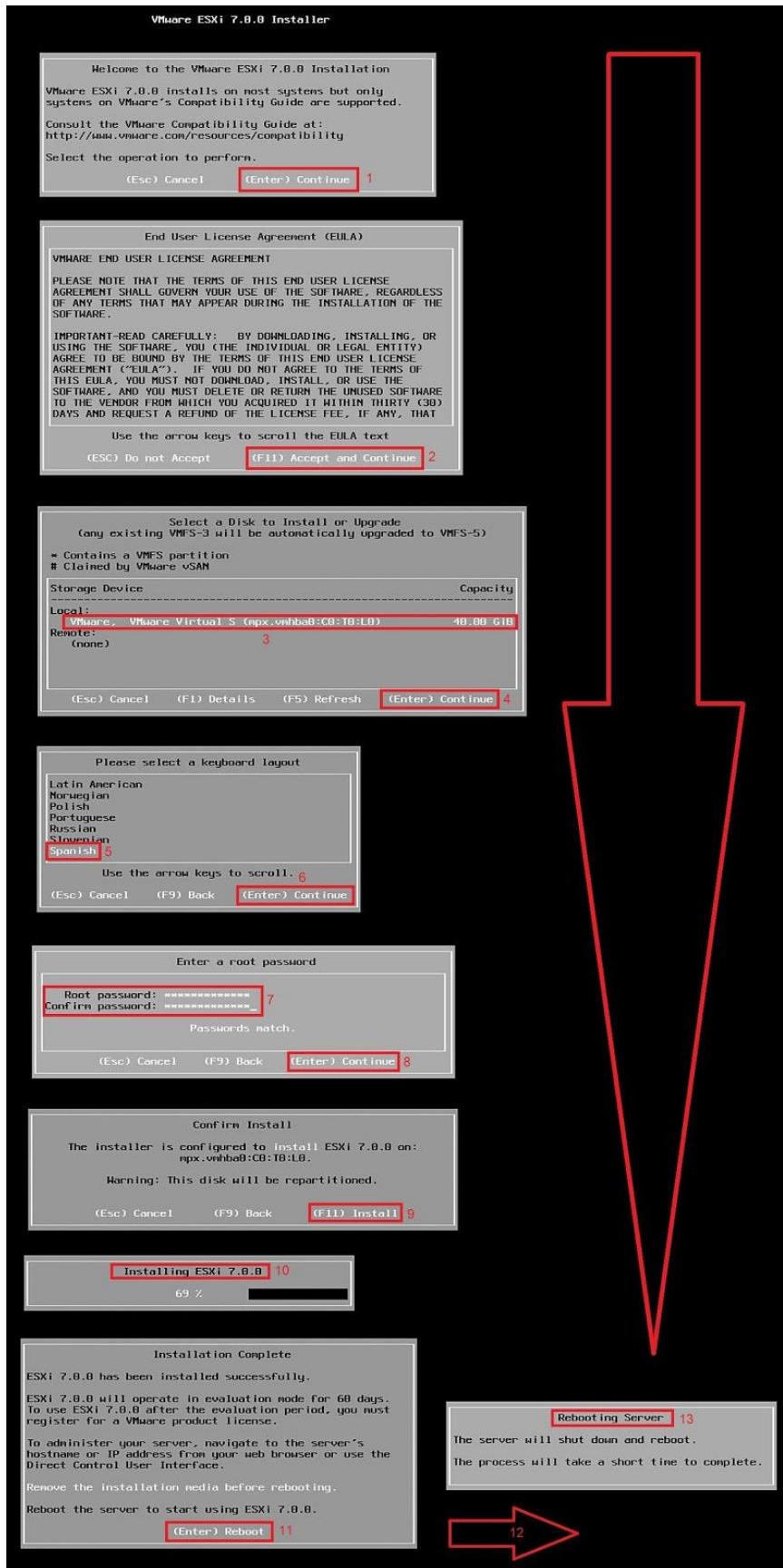
El proceso de instalación de un servidor **ESXi** a la versión **7.0**, no varía mucho de la forma en que VMware nos tenía acostumbrados con sus predecesores.

- a. En primer lugar, debemos descargar la imagen ISO del producto de la página oficial del fabricante. En la página de descargas de VMware encontraremos también

disponibles imágenes personalizadas para servidores de marca **HP**, **Lenovo**, **Fujitsu**, **CISCO** y **Hitachi**.

- b. Seguidamente, montaremos la imagen en nuestro servidor, ya sea pasándola a un soporte **USB** o **CD**, también podríamos usar la ILO o idrac de nuestro equipo físico para montar la imagen ISO directamente.
- c. Arrancaremos desde la imagen, hecho esto, empezará el proceso de carga de la instalación del producto, en nuestro laboratorio instalaremos la versión 7.0.
- d. En primer lugar, nos aparecerá la pantalla de bienvenida del asistente de instalación, pulsaremos la tecla **Enter** para continuar y seguidamente debemos aceptar la **End User License Agreement** o **EULA** pulsando la tecla **F11**.
- e. Seguidamente entraremos en materia, el asistente nos pedirá que seleccionemos el dispositivo de disco en el que deseamos instalar nuestro sistema operativo. En nuestro laboratorio de ejemplo disponemos de un volumen de disco de 40 gigas para poder realizar la instalación, lo seleccionaremos y presionaremos una vez más la tecla **Enter**.
- f. Hemos de prestar la máxima atención a este punto, en caso de tener un volumen SAN conectado a nuestro host, o si servidor dispone de algún otro volumen de disco local además del destinado a la instalación del hypervisor. Si no seleccionamos el volumen de disco correcto destinado a la instalación, podríamos tener perdida de datos.
- g. Seguidamente, nos preguntará el idioma de nuestro teclado, escogeremos el que más se adecue a nuestras necesidades y presionaremos **Enter** para continuar.
- h. Terminaremos la configuración del instalador asignando una contraseña segura al nuevo usuario **Root** de nuestro nuevo servidor **VMware vSphere ESXi 7.0**.

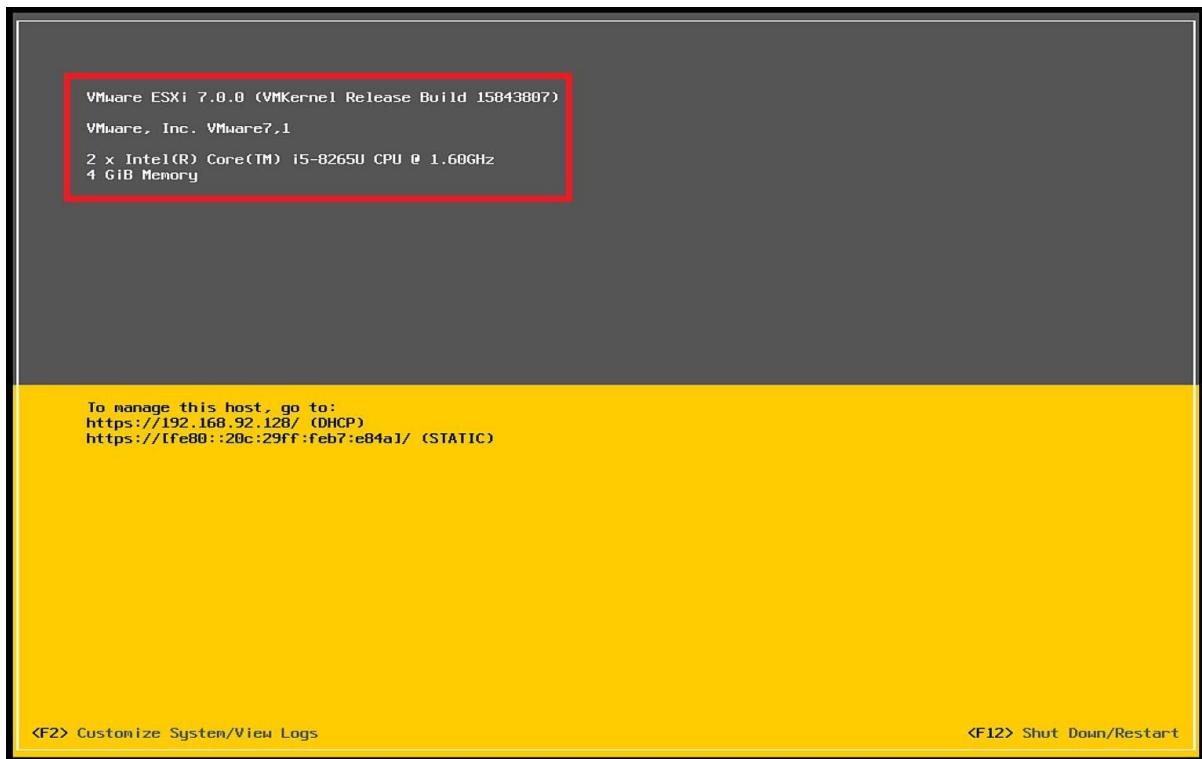
Una vez tengamos configuradas todas las opciones, para proceder con la instalación del nuevo producto, debemos confirmar la instalación del producto pulsando la tecla **F11** de nuestro teclado.



Una vez hayamos presionado **F11**, empezará automáticamente la instalación. Debemos esperar pacientemente a su culminación, en ese momento, nos mostrará una ventana donde

se nos informará del éxito de la nueva instalación, y también de la necesidad de reiniciar el nuevo **Host ESXi 7.0**, presionaremos por última vez la tecla **Enter** para reiniciar el servidor.

Arrancaremos por primera vez nuestro host **ESXi 7.0** y podremos comenzar con las tareas de configuración del nuevo servidor.



3. CONFIGURAR RED DE GESTIÓN- ADAPTADORES DE RED.

En este apartado del libro, vamos a configurar algunas opciones de la Red de gestión en un servidor host **VMware ESXi 7.0**.

Usaremos el menú **Personalización del sistema** (Modo texto) de nuestro nuevo host, para agregar varios adaptadores de red a la red de gestión y así conseguir redundar la red de administración del servidor host.

Cuando iniciamos un servidor **VMware ESXi vSphere 7.0**, la pantalla de la consola que tengamos conectada físicamente a nuestro servidor nos mostrará la imagen que tenemos a continuación. En ella, podremos ver:

- Versión de VMware.
- Release del Kernel.
- CPU física instalada en nuestro host.
- Memoria RAM física instalada en nuestro host.
- Parámetros de configuración de red.

Para empezar la configuración, pulsaremos la tecla F2 de nuestro teclado. De este modo, accederemos a la opción llamada **<F2> Personalizar sistema / Ver registros**.

Nos aparecerá una ventana emergente, donde nos solicitará las credenciales de acceso a nuestro servidor. Introduciremos las credenciales de acceso de nuestro usuario root y, seguidamente, pulsaremos en la tecla **Enter**.



En el menú de la pantalla **Personalización del sistema**, usaremos las flechas del teclado conectado a nuestro servidor host para descender a la opción del menú lateral llamada, **Configurar red de administración**.

El aspecto de la pantalla cambiará y nos aparecerá un nuevo menú con las opciones de configuración de la red de administración.

Para configurar los adaptadores físicos asignados a la red de administración, seleccionaremos la primera de las opciones de menú de la sección **Configurar red de administración**, llamada **Adaptadores de red**.

Aparecerá una pequeña ventana emergente, donde podremos seleccionar usando el teclado conectado a nuestro servidor, los adaptadores de red que formarán parte del grupo de **Management Network**.

De este modo, podremos redundar la conexión de administración de nuestros servidores.

Finalizada la configuración, pulsaremos la tecla **Enter** de nuestro teclado para guardar los cambios y cerrar la ventana de configuración de los adaptadores de red.

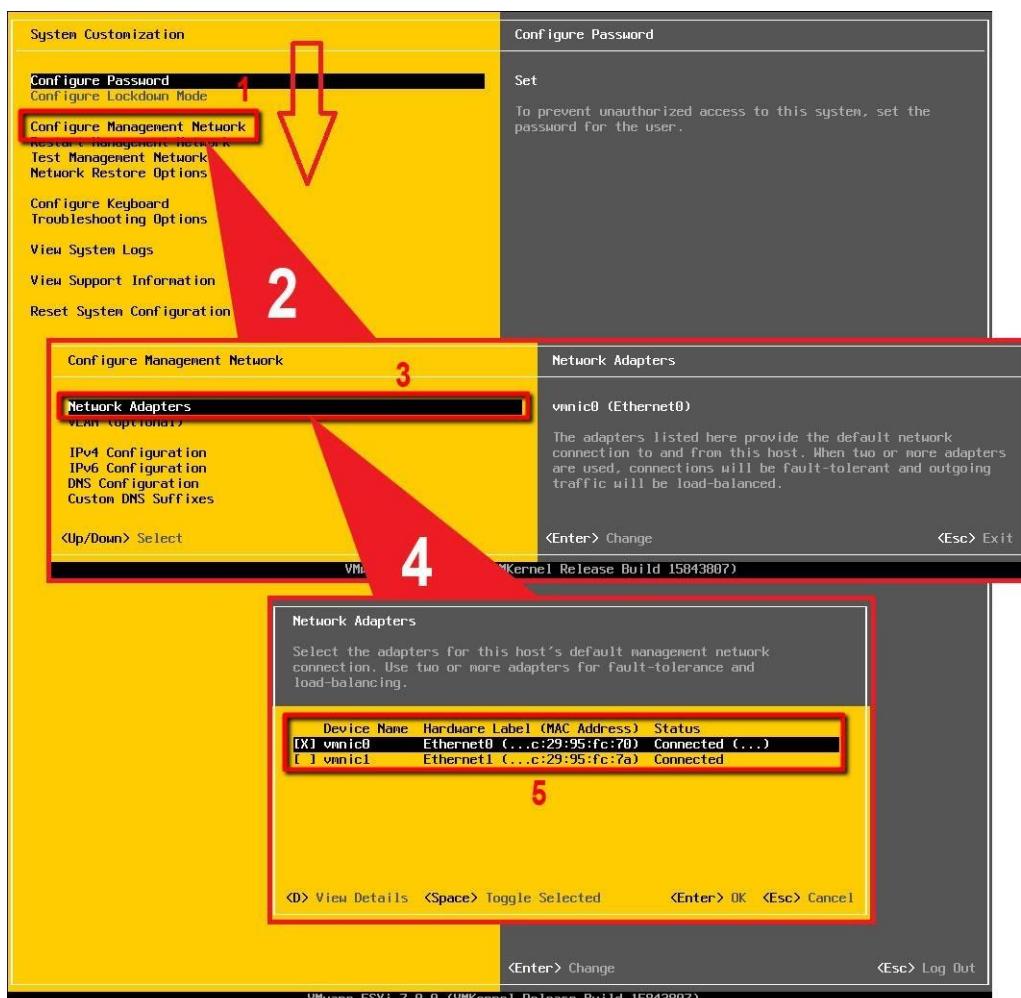
Terminada la configuración, presionaremos la tecla **Esc** de nuestro teclado para salir del menú **Configurar red de administración**.

Nos aparecerá una última ventana emergente, para confirmar todos los cambios que hemos realizado durante la configuración de las tarjetas de red de nuestro host **VMware ESXi vSphere 7.0**.

You have mode changes to the host's management network. Applying these changes may result in a brief network outage, disconnect remote management software and affect running virtual machines. In case IPv6 has been enabled or disabled this will restart your host

“Usted ha realizado cambios en la red de administración del host. La aplicación de estos cambios puede ocasionar una breve interrupción de la red, desconectar el software de administración remota y afectar la ejecución de máquinas virtuales. En caso de que IPv6 se haya habilitado o deshabilitado, esto reiniciará su host.”

Confirmaremos que todo es correcto, pulsando la tecla **Y**, nuestro servidor host reiniciará. Una vez termine el reinicio del servidor, aparecerá la pantalla principal de **VMware ESXi vSphere 7.0**, donde podremos volver a acceder al menú para comprobar todos los cambios se han aplicado correctamente.



4. CONFIGURAR SERVIDORES DNS Y LOS SUFIJOS DE DNS.

En esta sección, vamos a ver cómo usar el menú llamado personalizar sistema de nuestro host **VMware ESXi 7.0**, para conseguir configurar los servidores de nombres o servidores DNS y también los sufijos de DNS.

Como ya hicimos en el apartado anterior, para empezar la configuración, pulsaremos la tecla **F2** de nuestro teclado. De este modo, accederemos a la opción llamada **<F2> Personalizar sistema / Ver registros**.



Nos aparecerá una ventana emergente, donde nos solicitará las credenciales de acceso a nuestro servidor. Introduciremos las credenciales de acceso de nuestro usuario **root** y, seguidamente, pulsaremos la tecla **Enter** en nuestro teclado.

En la pantalla **personalizar sistema**, usaremos de las flechas de nuestro teclado para descender a la opción del menú llamada **Configurar red de administración**.

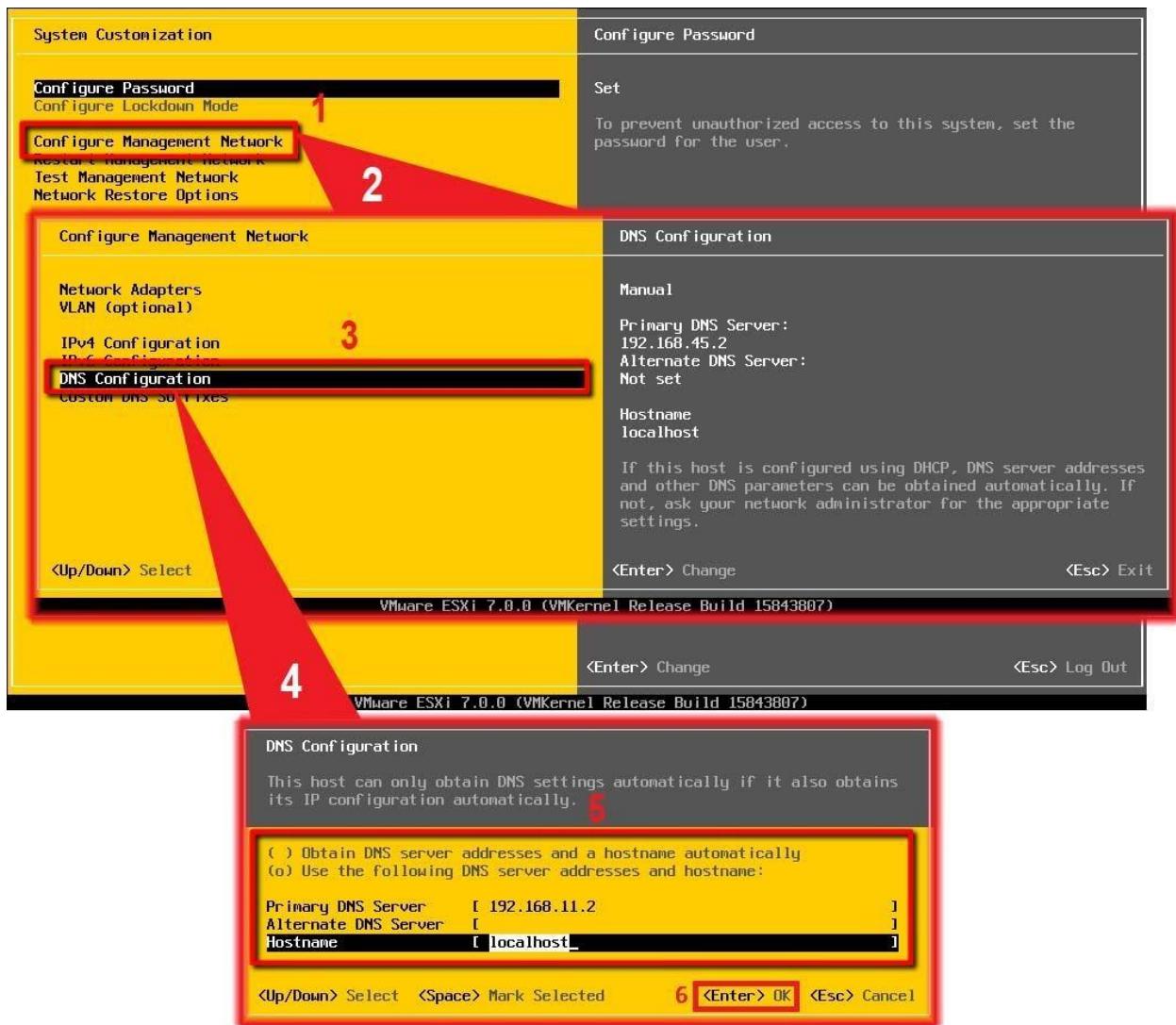
El aspecto de la pantalla cambiará y nos aparecerá un nuevo menú con las opciones de **Configuración de la red de Administración**.

Para configurar los servidores de **DNS** de la red de administración, seleccionaremos la quinta de las opciones de menú de la sección Configurar red de administración, llamada **Configuración DNS**.

Aparecerá una pequeña ventana emergente llamada **Configuración DNS**, donde podremos introducir las direcciones IP de los servidores de nombres de nuestra red.

También nos permitirá asignar un nombre a nuestro servidor host.

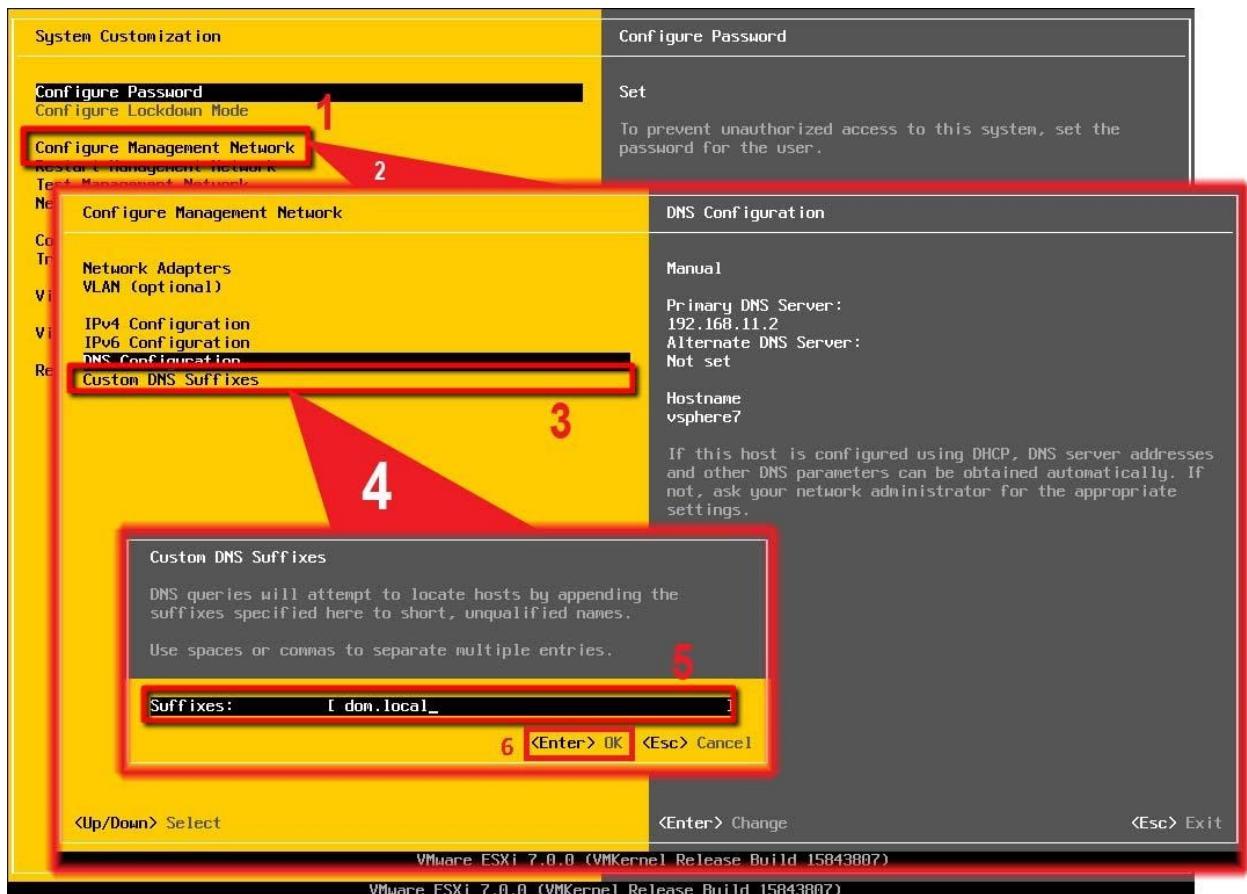
Finalizada la configuración, pulsaremos la tecla **Enter** de nuestro teclado para guardar los cambios y cerrar la ventana de configuración de los adaptadores de red.



Terminada la configuración, presionaremos la tecla **Esc** de nuestro teclado para salir del menú llamado **Configurar red de administración**.

Seguidamente, configuraremos los **sufijos DNS personalizados** asignados a la red de administración de nuestro host, seleccionaremos la sexta de las opciones de menú de la sección **Configurar red de administración**, llamada **Sufijos DNS personalizados**.

Aparecerá una pequeña ventana emergente llamada **Sufijos DNS personalizados**, donde podremos configurar las opciones de **Sufijos DNS personalizados** de la tarjeta de red de nuestro host.



Finalizada la configuración, pulsaremos la tecla **Enter** de nuestro teclado para guardar los cambios y cerrar la ventana de configuración de los adaptadores de red.

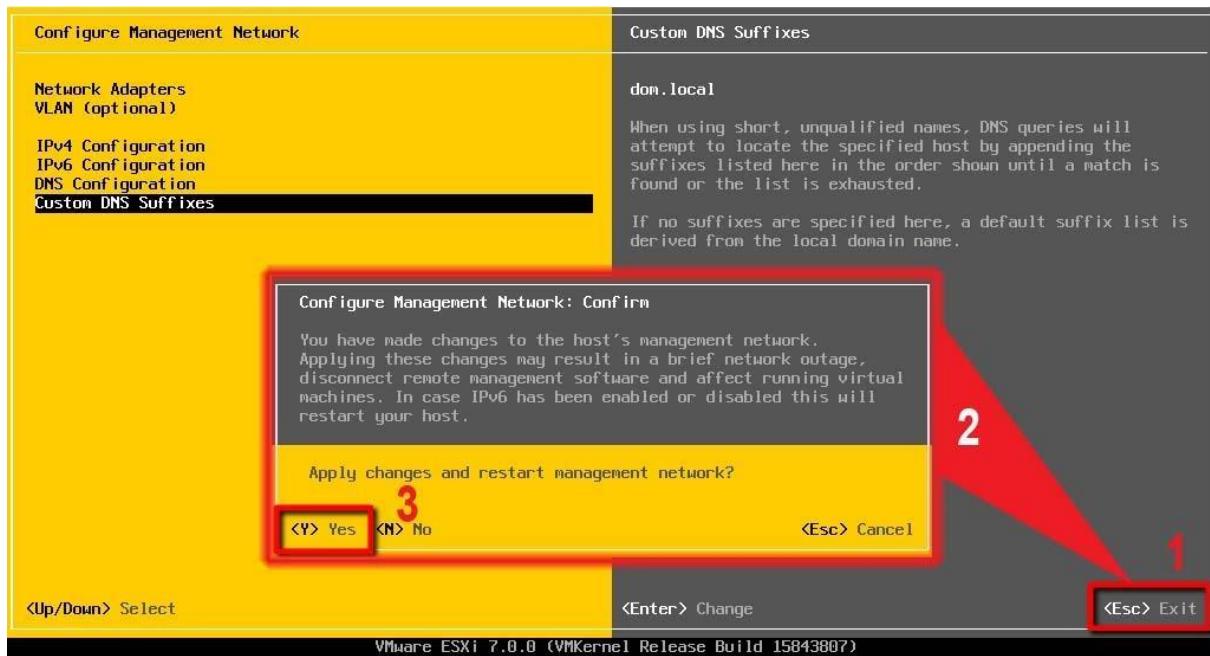
Una vez aplicada la nueva configuración, presionaremos la tecla **Esc** de nuestro teclado para salir del menú Configurar red de administración

En este momento aparecerá una nueva advertencia.

You have made changes to the host's management network. Applying these changes may result in a brief network outage, disconnect remote management software and affect running virtual machines. In case Ipv6 has been enabled or disabled this will restart your host

"Usted ha realizado cambios en la red de administración del host. La aplicación de estos cambios puede ocasionar una breve interrupción de la red, desconectar el software de administración remota y afectar la ejecución de máquinas virtuales. En caso de que IPv6 se haya habilitado o deshabilitado, esto reiniciará su host."

Confirmaremos que todo es correcto pulsando la tecla **Y**.



Nuestro servidor host no reiniciará, si no hemos realizado cambios en la configuración del protocolo **IPv6**.

Aparecerá nuevamente el menú principal llamado **Personalización del sistema**, donde podremos comprobar todos los cambios ya aplicados.

Pulsaremos una vez más la tecla **Esc** en el teclado de nuestro host para volver a la pantalla de bienvenida de **VMware ESXi vSphere 7.0**

5. CONFIGURE MANAGEMENT NETWORK - CONFIGURACIÓN IPv4.

En este apartado del libro, veremos cómo podemos configurar la dirección **IPv4** en un host **VMware ESXi 7.0**.

Como ya hicimos en los apartados anteriores del libro, para empezar la configuración, pulsaremos la tecla **F2** de nuestro teclado. De este modo, accederemos a la opción llamada **<F2> Personalizar sistema / Ver registros**.

Nos aparecerá una ventana emergente, donde nos solicitará las credenciales de acceso a nuestro servidor. Introduciremos las credenciales de acceso de nuestro usuario **root** y, seguidamente, pulsaremos la tecla **Enter**.



En la pantalla **System Customization**, usaremos de las flechas de nuestro teclado para descender a la opción del menú lateral llamada **Configure Management Network**.

El aspecto de la pantalla cambiará y nos aparecerá un nuevo menú con las opciones de **Configuración de la red de administración (Management Network)**.

Para configurar de dirección **IPv4** asignada a la red de administración, seleccionaremos la tercera de las opciones del menú de la sección **Configure Management Network**, llamada **IPv4 Configuration**.

Aparecerá una pequeña ventana emergente llamada **IPv4 Configuration**, donde podremos configurar las opciones de red de nuestra tarjeta de red.

- Dirección IPv4.
- Mascara de subred.
- Puerta de enlace.

Finalizada la configuración, pulsaremos la tecla **Enter** de nuestro teclado para guardar los cambios y cerrar la ventana de configuración de los adaptadores de red.

Terminada la configuración, presionaremos la tecla **Esc** de nuestro teclado para salir del menú **Configure Management Network**.

Nos aparecerá una última ventana emergente para que confirmemos todos los cambios que hemos realizado durante la configuración de las tarjetas de red de nuestro host **VMware ESXi vSphere 7.0**.

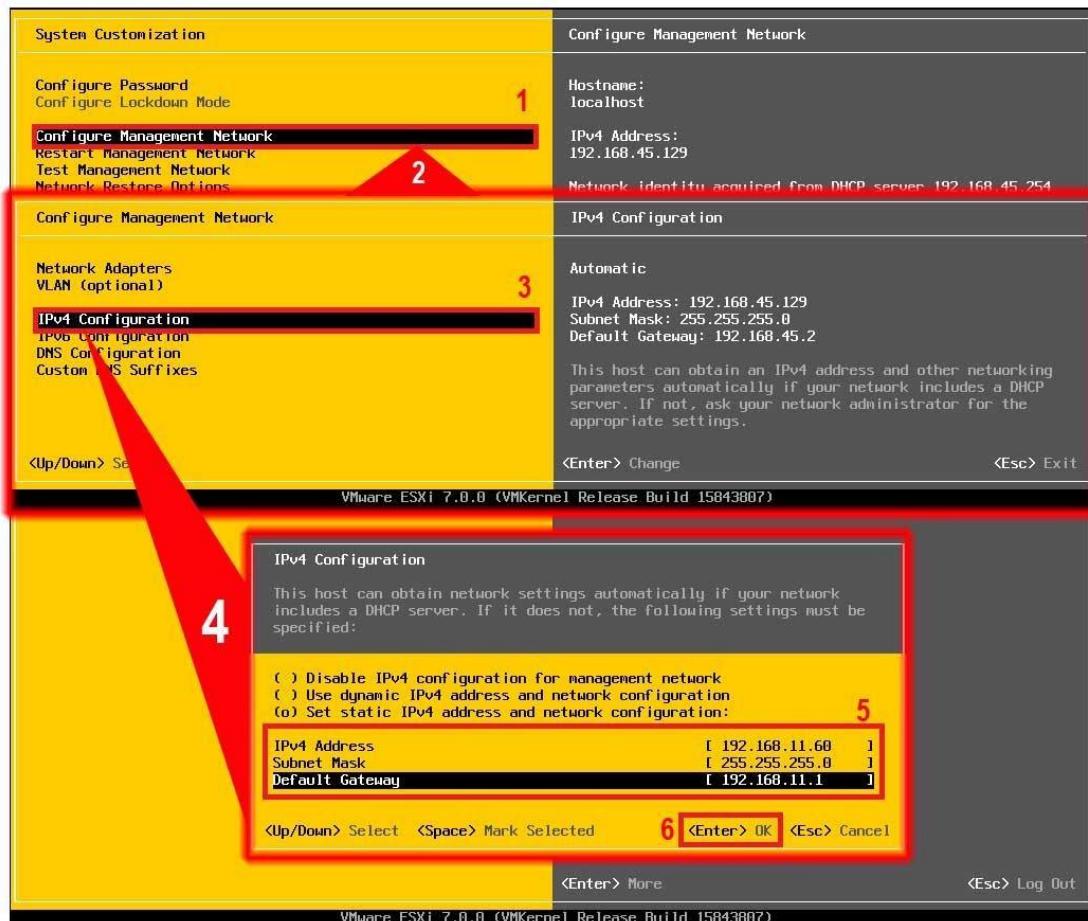
You have made changes to the host's management network. Applying these changes may result in a brief network outage, disconnect remote management software and affect running virtual machines. In case IPv6 has been enabled or disabled this will restart your host

“Usted ha realizado cambios en la red de administración del host. La aplicación de estos cambios puede ocasionar una breve interrupción de la red, desconectar el software de administración remota y afectar la ejecución de máquinas virtuales. En caso de que IPv6 se haya habilitado o deshabilitado, esto reiniciará su host.”

Confirmaremos que todo es correcto pulsando la tecla **Y**. Si hemos modificado las opciones de IPv6 nuestro servidor host reiniciará.

Aparecerá el menú principal **System Customization**, donde podremos comprobar todos los cambios ya aplicados.

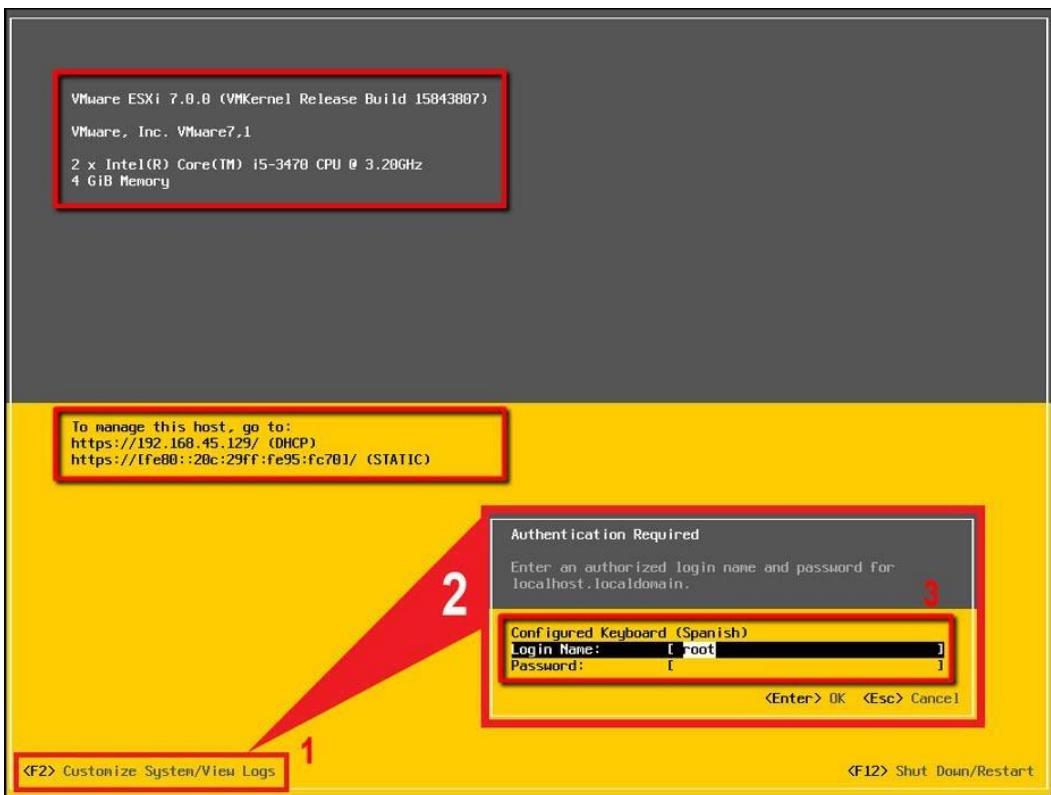
Pulsaremos una vez más la tecla **Esc** en el teclado de nuestro host para volver a la pantalla de bienvenida de **VMware ESXi vSphere 7.0**



6. CONFIGURE MANAGEMENT NETWORK - CONFIGURACIÓN IPv6.

Por último, vamos a configurar la dirección **IPv6** en un host **VMware ESXi 7.0**.

Nos aparecerá una ventana emergente, donde nos solicitará las credenciales de acceso a nuestro servidor. Introduciremos las credenciales de acceso de nuestro usuario **root** y, seguidamente, pulsaremos la tecla **Enter**.



En la pantalla **System Customization**, usaremos de las flechas de nuestro teclado para descender a la opción del menú lateral llamada **Configure Management Network**.

El aspecto de la pantalla cambiará y nos aparecerá un nuevo menú con las opciones de **Configuración de la red de Administración (Management Network)**.

Para configurar de dirección **IPv6** asignada a la red de administración, seleccionaremos la cuarta de las opciones de menú de la sección **Configure Management Network**, llamada **IPv6 Configuration**.

Aparecerá una pequeña ventana emergente llamada **IPv6 Configuration**, donde podremos configurar las opciones de nuestra tarjeta de red.

En nuestro laboratorio, no usamos la configuración IPv6 ni tampoco tenemos un servidor de **DHCP** que despliegue la configuración **IPv6** en nuestro entorno, así pues, vamos a deshabilitar **IPv6** de nuestro servidor host.

Finalizada la configuración pulsaremos la tecla **Enter** de nuestro teclado para guardar los cambios y cerrar la ventana de configuración de los adaptadores de red.

Terminada la configuración presionaremos la tecla **Esc** de nuestro teclado para salir del menú **Configure Management Network**.

Nos aparecerá una última ventana emergente para que confirmemos todos los cambios que hemos realizado durante la configuración de las tarjetas de red de nuestro host **VMware ESXi vSphere 7.0**.

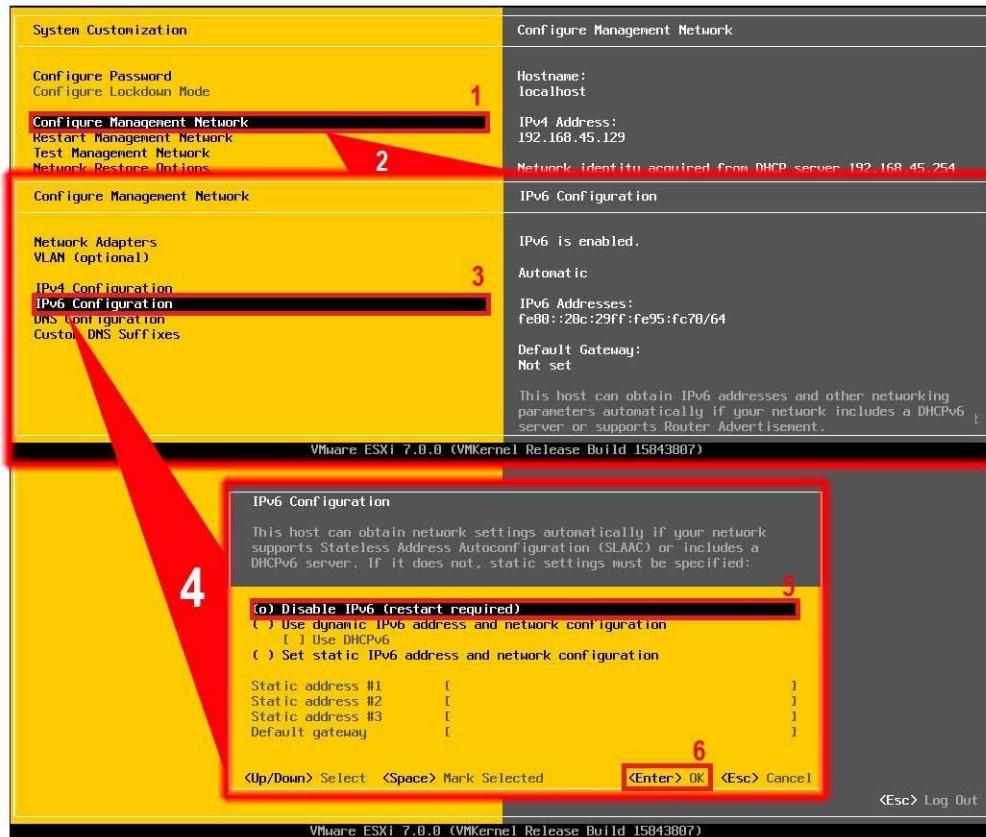
You have made changes to the host's management network. Applying these changes may result in a brief network outage, disconnect remote management software and affect running virtual machines. In case IPv6 has been enabled or disabled this will restart your host

“Usted ha realizado cambios en la red de administración del host. La aplicación de estos cambios puede ocasionar una breve interrupción de la red, desconectar el software de administración remota y afectar la ejecución de máquinas virtuales. En caso de que IPv6 se haya habilitado o deshabilitado, esto reiniciará su host.”

Confirmaremos que todo es correcto pulsando la tecla **Y**. Nuestro servidor host reiniciará.

Una vez termine el reinicio del servidor aparecerá el menú principal **System Customization** donde podremos comprobar todos los cambios ya aplicados.

Pulsaremos una vez más la tecla **Esc** en el teclado de nuestro host para volver a la pantalla de bienvenida de **VMware ESXi vSphere 7.0**.



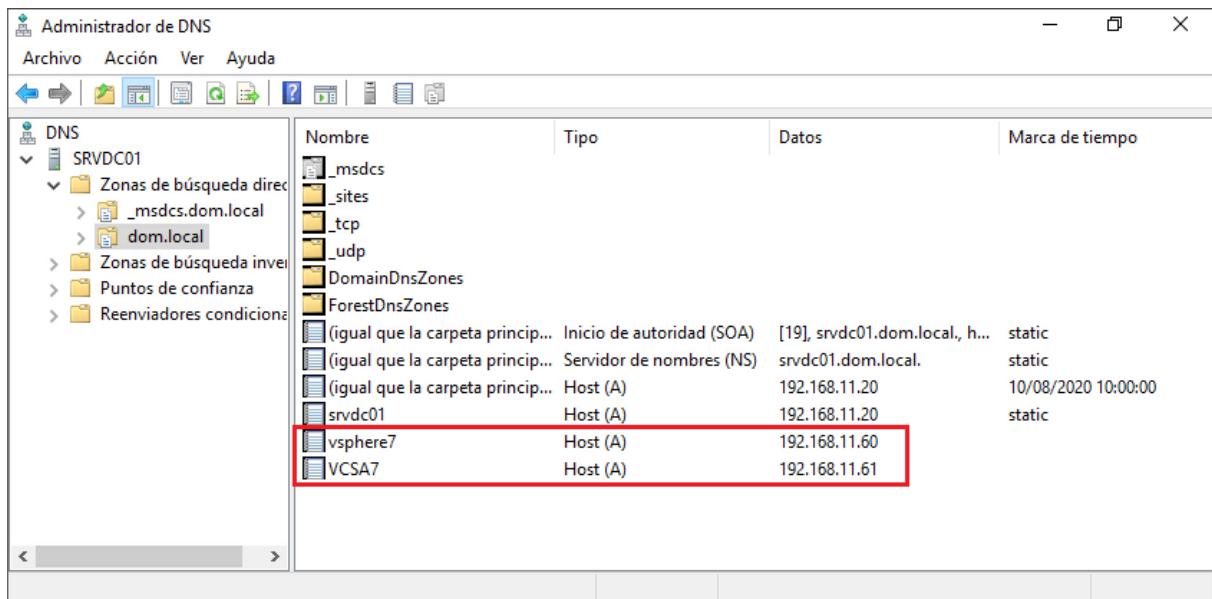
7. DESPLIEGUE DE UN NUEVO vCENTER SERVER APPLIANCE. (VCSA)

En este apartado, vamos a ver como instalar un nuevo servidor de **Virtual Center Appliance de la versión 7.0**.

El asistente de despliegue de un servidor de **vCenter Appliance** está dividido en dos fases, la primera correspondiente al despliegue del nuevo servidor y la segunda dedicada a la configuración. Así que también hemos dividido el proceso en dos laboratorios distintos. Este es el primer laboratorio está dedicado a la primera fase del asistente **Stage1 Deploy Appliance**.

En primer lugar y antes de empezar con el asistente de instalación de nuestro nuevo **vCenter Server Appliance (VCSA)**, debemos asegurarnos de tener previamente configuradas en el servidor de nombres de nuestro dominio, las entradas de **DNS** correspondientes a nuestros servidores host **vSphere** y también el nombre que queremos asignar al nuevo servidor de **vCenter Server Appliance (VCSA)** que vamos a desplegar.

De no ser así, las configuraremos antes de avanzar, si no, corremos el riesgo que la segunda parte del asistente dedicado a la Instalación de **(VCSA)** falle en el inicio del proceso.



Una vez hayamos configurado los nuevos registros de nombres, tendremos que descargar la imagen de DVD ISO del instalador del producto desde la página oficial de VMware.

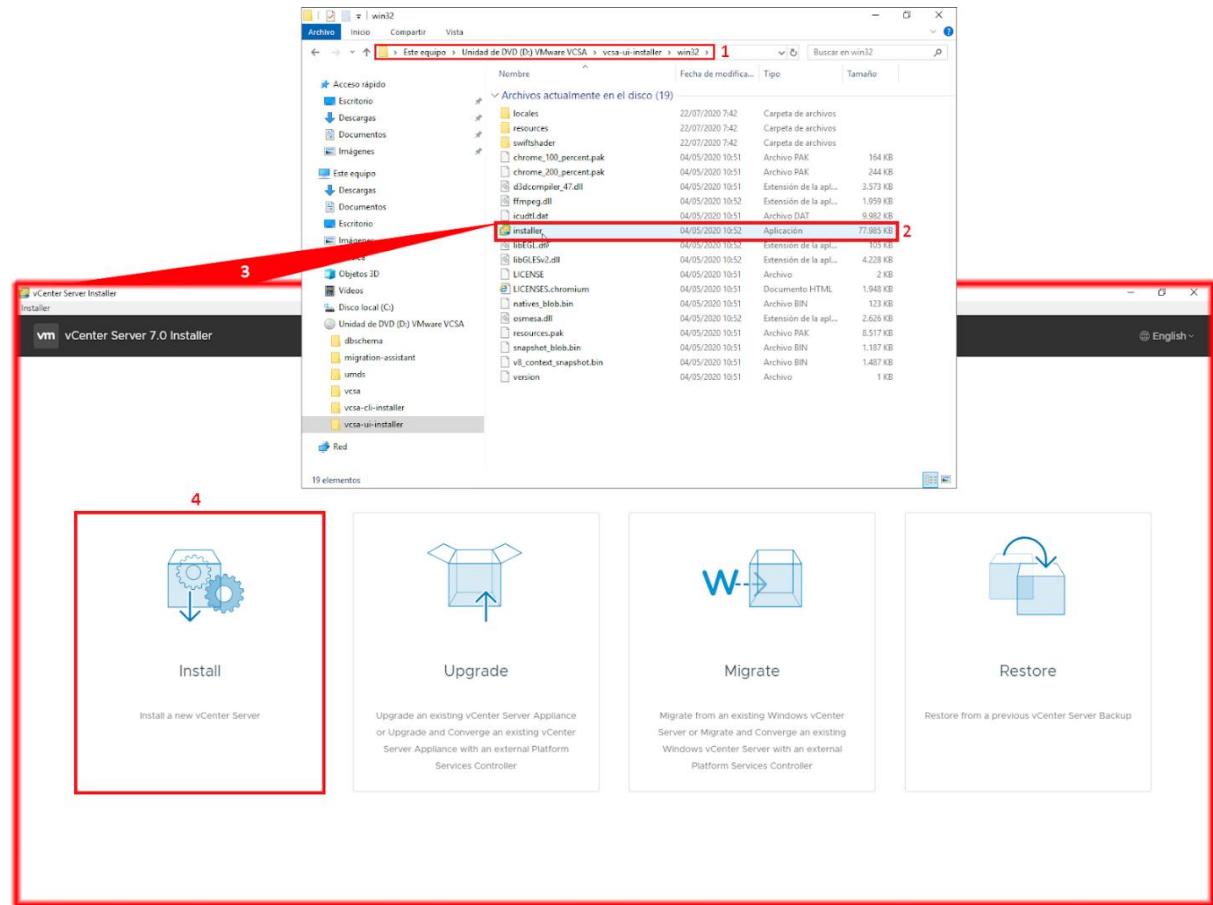
Una vez hayamos descargado el DVD de instalación en nuestro equipo, ejecutaremos el instalador que se encuentra en la ruta que mostramos a continuación:

[CD/DVD] : \VCSA-UI-INSTALLER\WIN32\INSTALLER.EXE

Aparecerá una nueva ventana emergente con el menú principal de despliegue, que nos permitirá elegir entre varias opciones, como por pueden ser:

- Instalar
- Actualizar
- Migrar
- Restaurar

Elegiremos la opción llamada Instalar y aparecerá la primera ventana del asistente, llamada **Introduction**, donde nos describirá el proceso de instalación del producto.



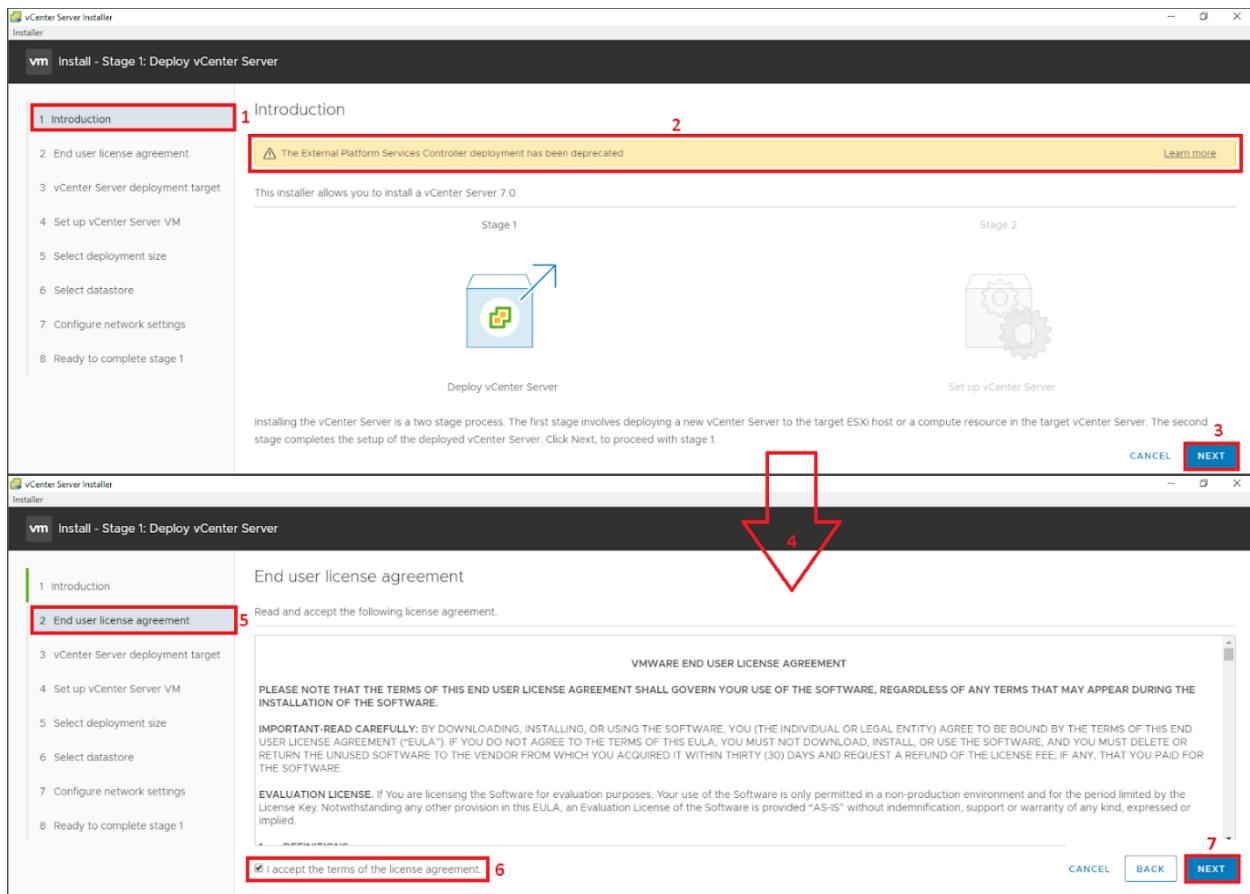
Como ya hemos comentado con anterioridad, el proceso de despliegue estará formado por dos escenarios. El primero será el despliegue de nuestro nuevo **vCenter Appliance** y el segundo escenario será la configuración de este.

Comprobaremos que la única instalación posible es en la versión 7.0 del producto es **vCenter Server with Embedded Platform Services Controller**, que instalará **Platform Services Controller** dentro del propio **vCenter Server Appliance (VCSA)**.

La posibilidad de instalar vCenter Server With External Platform Services Controller: es una opción obsoleta en VCSA 7.0.

Seguidamente, presionaremos el botón **Next** para comenzar despliegue del nuevo servidor **VCSA**.

Seguidamente, nos encontraremos en la sección llamada **Acuerdo de licencia de usuario final**. En ella debemos aceptar el **Acuerdo de licencia de usuario final de VMware** y, a continuación, presionaremos el botón **Next** para continuar con el asistente.



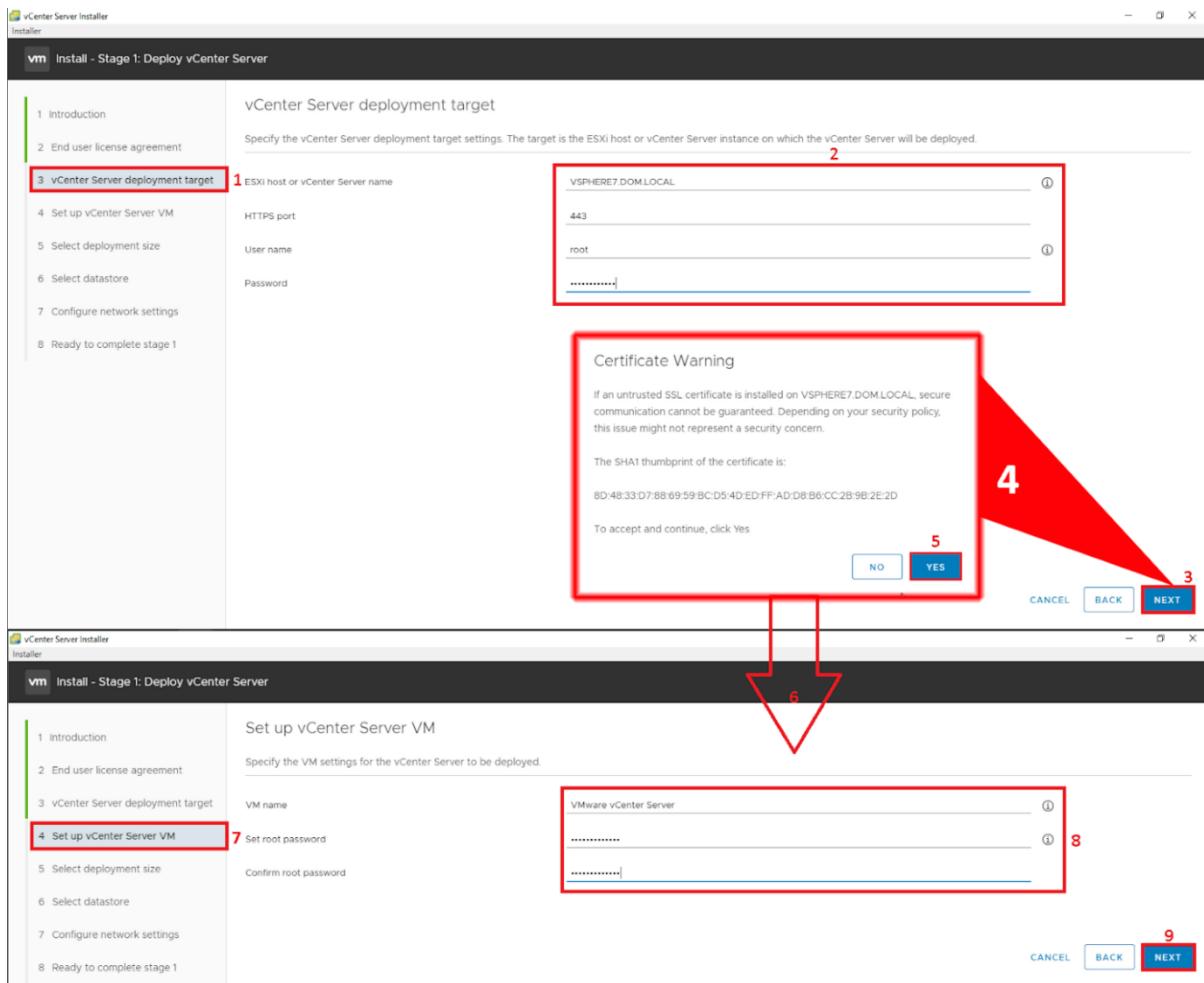
Seguidamente, nos encontraremos en la ventana llamada **Destino de despliegue del dispositivo**. En esta nueva sección, debemos llenar el formulario con los datos de uno de nuestros servidores host, será el servidor host donde queremos desplegar nuestro nuevo **vCenter Server Appliance (VCSA)**.

Introduciremos la dirección IP o **FQDN** completo de nuestro servidor host, **Puerto HTTPS**, **Nombre de usuario (p.e. root)** y finalmente la contraseña de acceso.

Una vez hayamos cumplimentado todo el formulario, pulsaremos el botón **Next** para avanzar a la siguiente sección del asistente de despliegue de **vCenter Server Appliance (VCSA)**.

La siguiente sección será la **Configuración de la nueva máquina virtual de vCenter Server Appliance**, en ella vamos a definir el nombre de la nueva máquina virtual y la contraseña que queremos asignar en el usuario **root** del servidor de **vCenter Server Appliance (VCSA)**.

La nueva contraseña definida en este formulario, la usaremos para poder realizar las tareas de gestión desde la UI.



Una vez hayamos pulsado el botón **Siguiente**, nos encoraremos en la sección **Seleccione el tamaño de la implementación**. Donde, mediante unos sencillos menús desplegables, conseguiremos definir el tamaño de la infraestructura que tendrá que soportar nuestro nuevo servidor de **vCenter Server Appliance (VCSA)**.

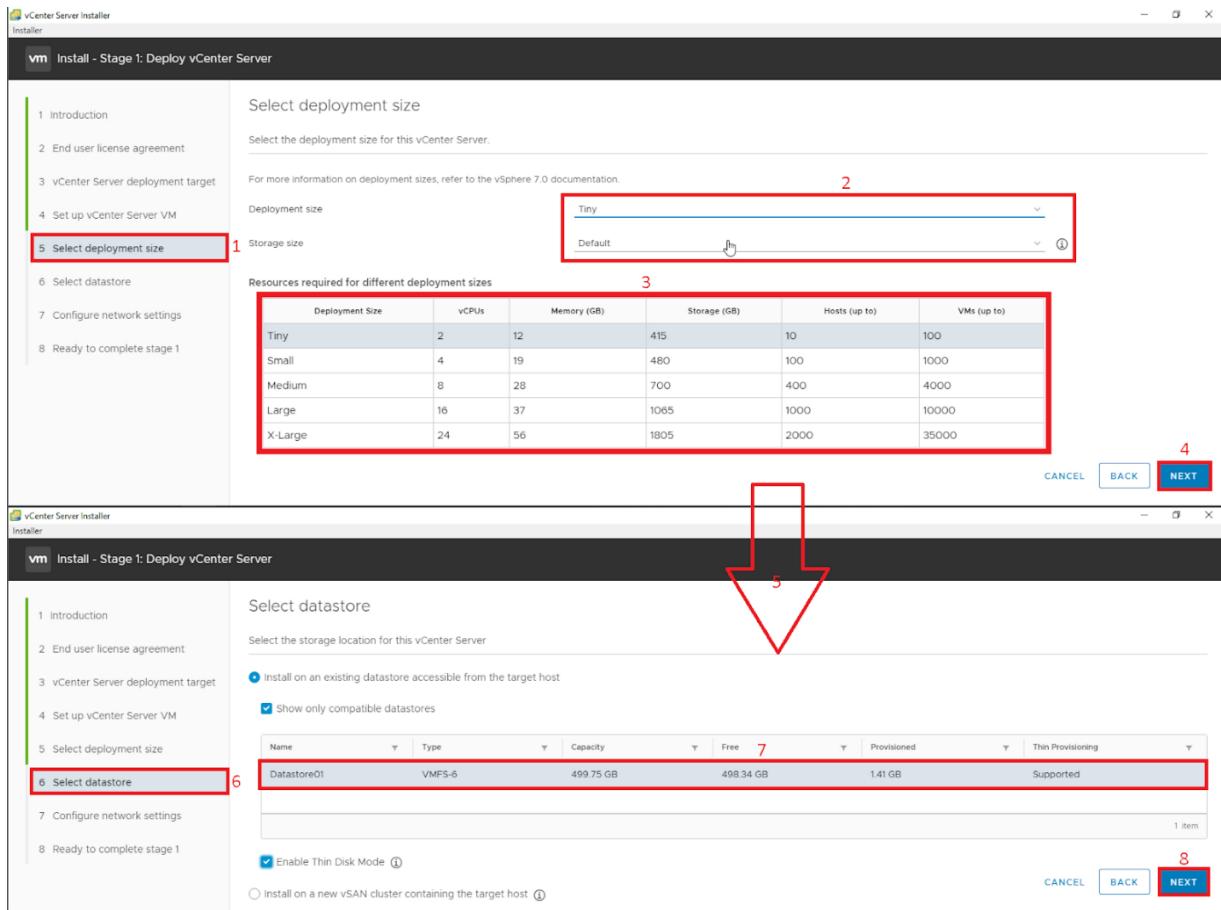
Para seleccionar el tamaño correcto para el nuevo servidor **vCenter Server Appliance (VCSA)**, debemos tener en cuenta cuántas máquinas virtuales y servidores host **ESXi** tendrá que gestionar. En nuestro laboratorio, hemos seleccionado la opción **Minúscula**, ya que en nuestro entorno solo tendremos dos servidores host y menos de **100** máquinas virtuales.

Dependiendo de nuestras selecciones, el tamaño de los recursos necesarios para desplegar nuestro nuevo servidor de **vCenter Server Appliance (VCSA)** variará.

La siguiente sección del asistente será **seleccionar almacén de datos**, en ella tendremos que seleccionar un almacén de datos con espacio suficiente para albergar nuestro nuevo **vCenter Server Appliance (VCSA)**.

Una vez hayamos seleccionado el almacén de datos, presionaremos una vez más el botón de **Siguiente** para avanzar en el asistente de despliegue del producto.

También, podremos elegir el modo de aprovisionamiento de los discos duros virtuales de nuestro nuevo servidor de **vCenter Server Appliance (VCSA)**, en nuestro laboratorio seleccionaremos un aprovisionamiento delgado.



Llegaremos a la sección llamada **Configurar ajustes de red**, donde configuraremos los ajustes de red de nuestro nuevo servidor de **vCenter Server Appliance (VCSA)**.

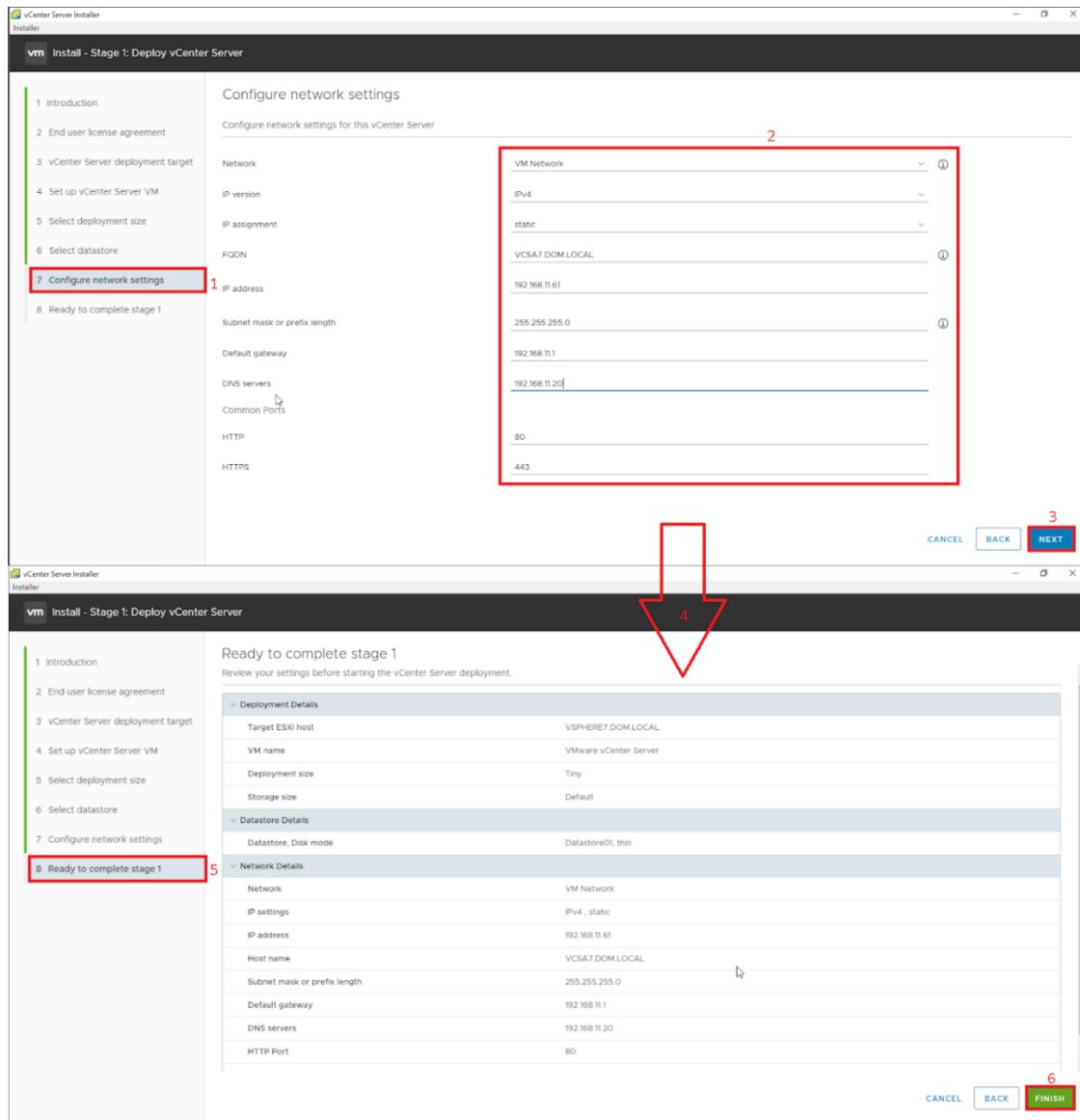
Los parámetros para cumplimentar del formulario serán los siguientes:

- **Network:** Seleccionaremos la red virtual donde queremos conectar nuestro nuevo servidor de vCenter.
- **IP Version:** IPv4 o IPv6.
- **IP Assignment:** Tipo de asignación de direcciones IP estática o Dinámica.
- **System Name:** Nombre FQDN de nuestro nuevo servidor de vCenter.
- **IP address:** Dirección IP que asignaremos de forma estática a nuestro nuevo servidor de vCenter.
- **Subnet mask o prefix length:** Máscara de red asignada a nuestro nuevo servidor de vCenter.
- **Default Gateway:** Puerta de enlace de nuestra infraestructura de red.
- **DNS Servers:** Servidores de nombres de nuestra infraestructura de red.

Es muy recomendable que nos aseguremos que la configuración de nuestro servidor de nombres sea correcta.

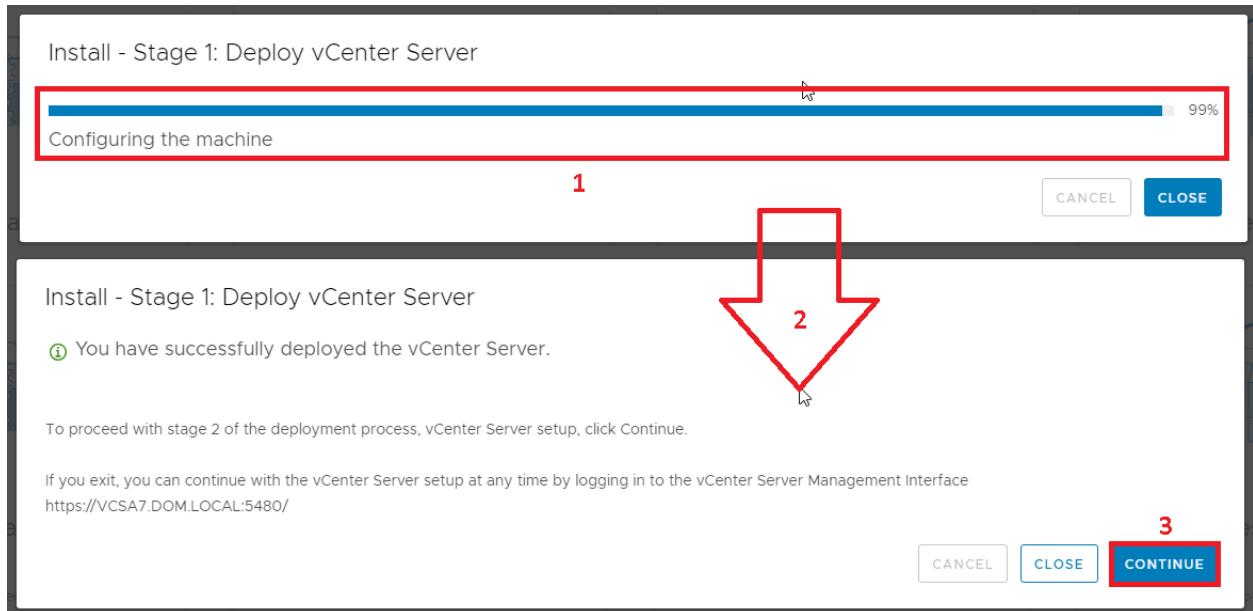
Como anteriormente hemos comentado, debemos asegurarnos de tener previamente configuradas las entradas de **DNS** correspondientes para el servidor de **vCenter Server Appliance (VCSA)** y nuestros servidores host en los servidores de nombres de nuestra organización. De no ser así, las configuraremos antes de avanzar más en el asistente de despliegue, si no, corremos el riesgo que la segunda parte del asistente dedicado a la Instalación de **vCenter Server Appliance (VCSA)** falle en el inicio de su ejecución.

Después de presionar el botón **siguiente**, nos encontraremos en la sección llamada **Listo para completar**, comprobaremos en el resumen y si todas las configuraciones realizadas durante el asistente son correctas, y, seguidamente podremos presionar finalizar.



El proceso de despliegue tardará más o menos, dependiendo de los recursos disponibles en nuestro host **ESXi**, en nuestro laboratorio tardó unos veinte minutos en finalizar.

Una vez terminado el despliegue de la nueva máquina virtual, presionaremos el botón **Continue** y acceder a la segunda fase del asistente.



8. CONFIGURACIÓN DE UN NUEVO VCENTER SERVER APPLIANCE. (VCSA)

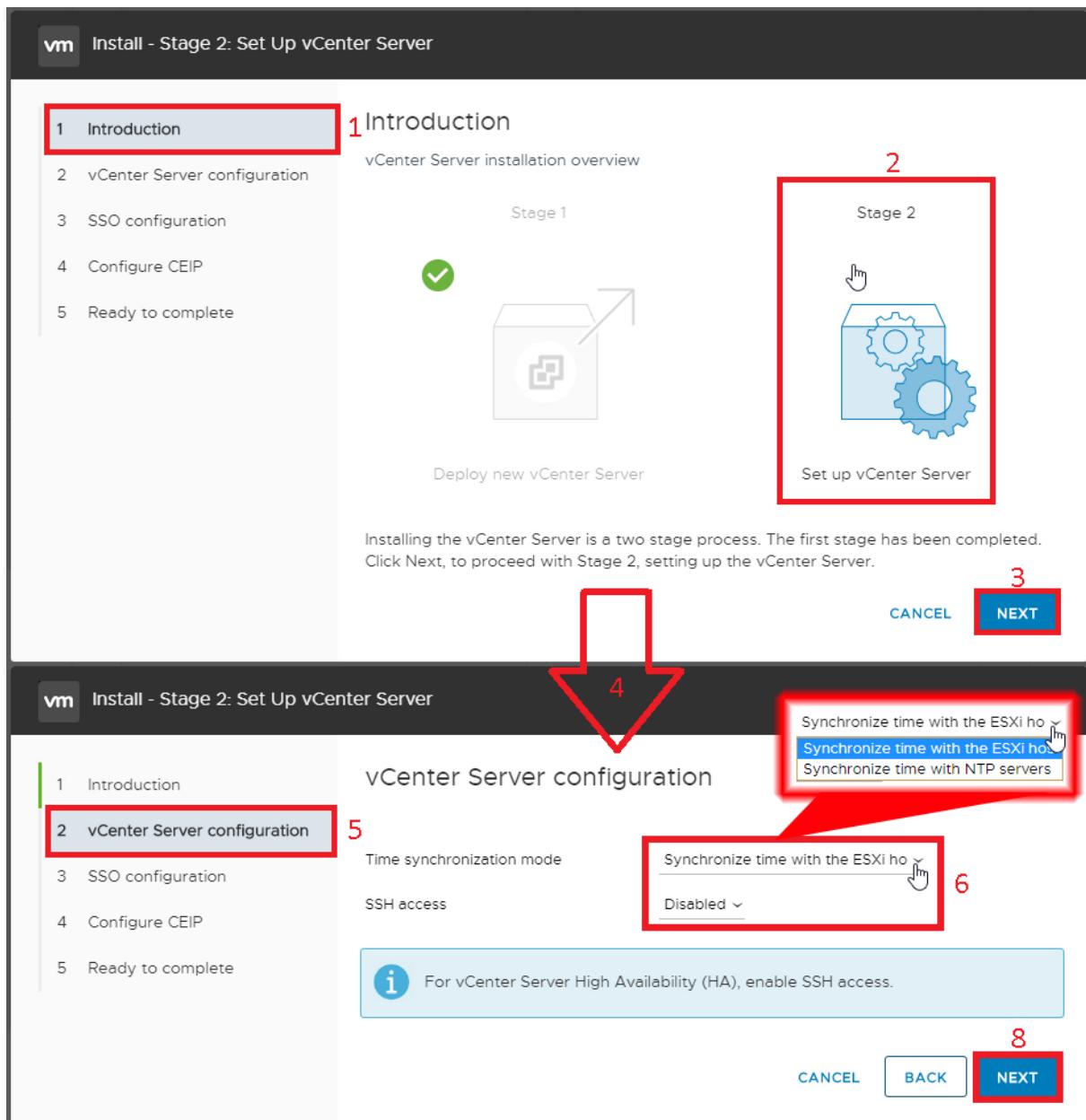
Una vez terminada la fase del despliegue del producto, pulsaremos el botón llamado **continuar** situado en la esquina inferior derecha de la ventana del asistente.

Seguidamente, aparecerá una nueva ventana emergente con el asistente de Instalar el servidor de **vCenter Server Appliance (VCSA)**, el cual nos dará acceso a la fase de configuración de nuestro nuevo servidor de **vCenter Server Appliance (VCSA)**.

Pulsaremos el botón **siguiente**, para saltar la ventana llamada **Introduction** que nos situará en la fase del asistente que estamos cursando. Seguidamente, nos encontraremos en la sección llamada **Configuración del Appliance**, dónde se nos permitirá seleccionar la configuración de nuestros servidores de tiempo.

En nuestro laboratorio, hemos seleccionado que queremos sincronizar los servicios de tiempo con el propio servidor host de **ESXi**, pero podríamos definir cualquier otro servidor de tiempo.

También, podremos **habilitar** o **deshabilitar** el acceso mediante **SSH**, para las futuras configuraciones.



A continuación, nos aparecerá la sección llamada **SSO Configuration**, dónde encontraremos un formulario que nos permitirá configurar un nuevo dominio de **Single Sign-on** o agregar nuestro nuevo servidor de vCenter a un dominio existente.

Para crear un nuevo dominio **SSO** tendremos que proporcionar los datos siguientes:

- Un nombre para nuestro nuevo dominio SSO
- Un nombre de usuario para el administrador
- Una contraseña para el usuario administrador

Podéis configurar cada una de estas opciones, con los parámetros que más se adecuen a vuestras necesidades empresariales.

La contraseña predeterminada del administrador de **vCenter Single Sign-On**, se especifica en la directiva de contraseñas de **vCenter Single Sign-On**.

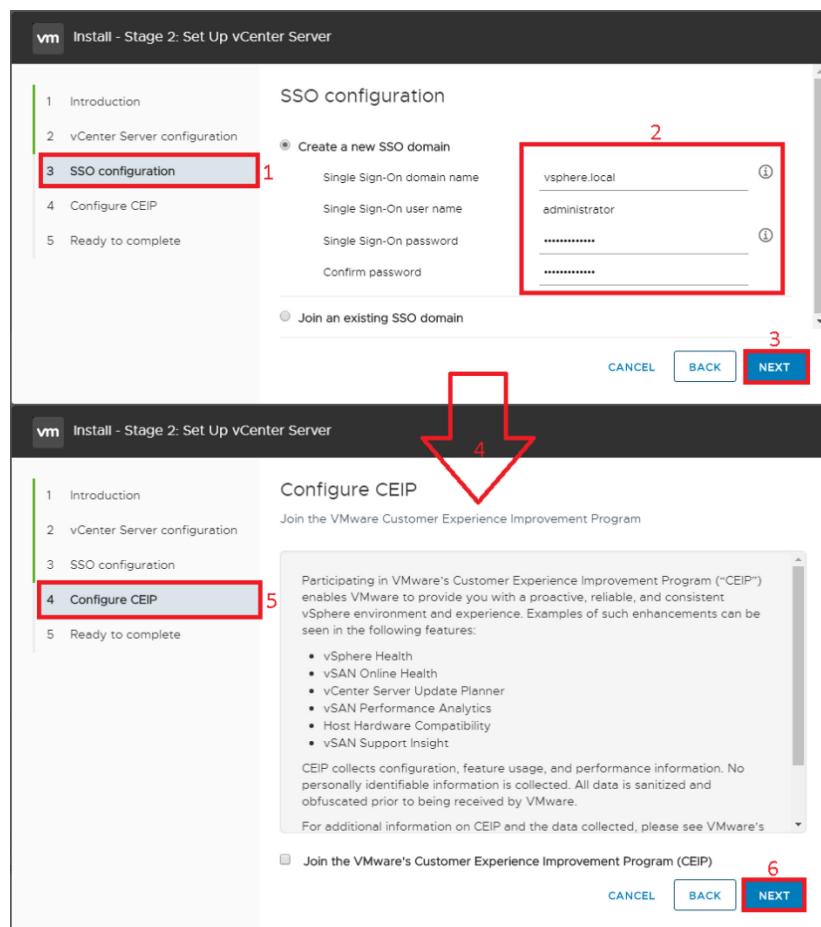
De manera predeterminada, esta contraseña debe cumplir con los siguientes requisitos:

Debe Tener al menos ocho caracteres, debe tener como mínimo un carácter en minúscula, al menos un carácter numérico y finalmente debe tener al menos un carácter especial.

La contraseña del usuario no puede superar los 20 caracteres. A partir de **vSphere 6.0**, se permiten caracteres que no son **ASCII**. Los administradores pueden cambiar la directiva de contraseñas predeterminada. Una vez tengamos definida la nueva contraseña, pulsaremos nuevamente el botón siguiente para continuar.

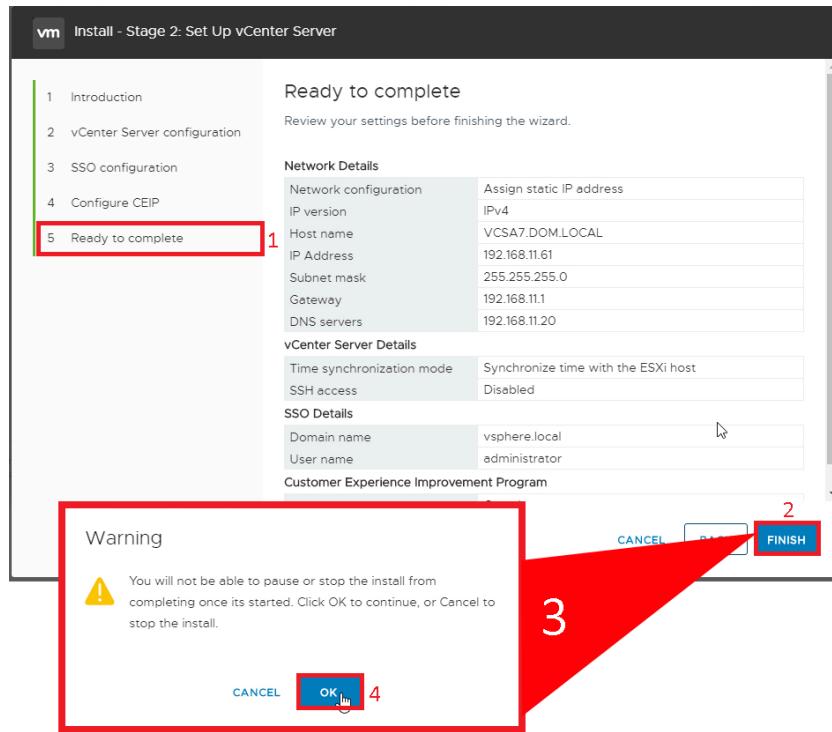
La siguiente sección será la configuración del programa **Programa de mejora de la experiencia del cliente de VMware (CEIP)** para facilitar información que ayudará a VMware a mejorar sus productos y servicios, a solucionar problemas y también a asesorar sobre la mejor forma de implementar y utilizar sus productos.

Podremos seleccionar si queremos unirnos al programa de mejora de **VMware** o, al contrario, queremos desactivarlo.



La última sección del **Stage 2** del asistente de **configuración del servidor vCenter Appliance 7.0** será lista para completar.

Si todos los parámetros que hemos configurado durante el asistente son correctos, estaremos en disposición de presionar **Finish** para configurar nuestra nueva máquina virtual de **vCenter Server Appliance 7.0 (VCSA)**.

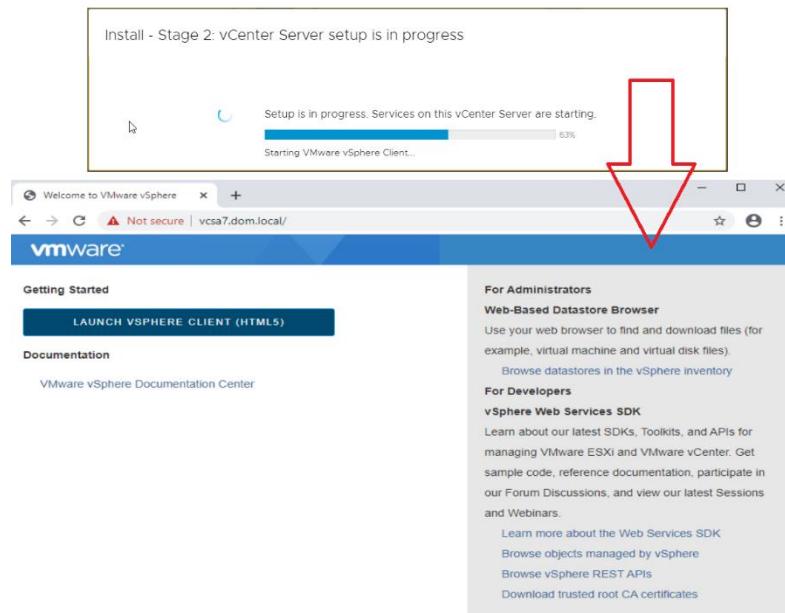


El proceso de configuración tardará unos diez minutos. Una vez haya completado, ya podremos tener acceso a nuestro **vCenter Web client**.

vSphere Web Client: https://IP_o_FQDN:443/vsphere-client

Podremos ver que tenemos acceso al cliente de vSphere HTML5.

Introduciremos nuestras credenciales de SSO, ya podremos acceder a nuestro vSphere Client.



Copyright © 1986-2020 VMware, Inc. All rights reserved. This product is protected by U.S. and international copyright and intellectual property laws. VMware products are covered by one or more patents listed at <http://www.vmware.com/go/patents>. VMware is a registered trademark or trademark of VMware, Inc. in the United States and/or other jurisdictions. All other marks and names mentioned herein may be trademarks of their respective companies. VMware products may contain individual open source software components, each of which has its own copyright and applicable license conditions. Please visit <http://www.vmware.com/info?id=1127> for more information.



EasyCloudFactory

Tu amigo de confianza en Cloud

En EasyCloudFactory encontrarás la flexibilidad que otras plataformas Cloud ni tan siquiera soñarían.

Somos especialistas en Infraestructura con muchos años de experiencia que trabaja como una extensión de tu departamento de TI.

Te invitamos a que nos pruebes y crezcas en la nube junto a nosotros.



EasyCloudFactory
BaaS

Backup como Servicio. Repositorio Cloud redundante sin límite de caudal ni tráfico para tus copias externas de Veeam Backup, Nakivo y otras soluciones de Backup On-Premise. ECF es el sitio ideal para almacenar tus Backups de Microsoft 365.

Y además...

DaaS

Solución de Desktop as a Service

MaaS

Monitorización como Servicio

Sistemas gestionados

Monitorización de tu Cloud

DRaaS

Recuperación ante Desastres

Security as a Service

Firewalls de nueva generación

Nextcloud

Plataforma de Colaboración Privada en Cloud



EasyCloudFactory
IaaS

Infraestructura como Servicio. Una plataforma Cloud estable, con un rendimiento Clase A++, monitorizada, gestionada y adaptada a tus necesidades. Puedes incrementar y reducir los servicios contratados en el momento que el negocio lo necesite.



-20%

Descuento exclusivo eBook.
Usa el código de descuento **vExperts** al configurar tu presupuesto en nuestra web.

Capítulo 3

ADMINISTRACIÓN DE ESXI DESDE LA LÍNEA DE COMANDOS



Gorka Izquierdo

@vGorkon

ADMINISTRACIÓN DE ESXI DESDE LA LÍNEA DE COMANDOS

INTRODUCCIÓN

El hypervisor ESXi incluye una Shell para poder administrar a este a través de la línea de comandos, ya sea para solucionar un problema o para labores de mantenimiento ya que muchas veces puede que nos encontremos con problemas para acceder a la interfaz gráfica y tengamos que hacer uso de los comandos desde la Shell

Desde la Shell puedes administrar el hypervisor ESXi de la misma forma que lo harías desde la interfaz Gráfica, incluso puedes llegar a hacer ciertas tareas administrativas que con la interfaz gráfica sería más complicado llevarlas a cabo.

Aquí tienes un par de ejemplos en los siguientes enlaces.

<https://www.sysadmit.com/2018/10/VMware-saber-ip-idrac-dell.html>

<https://www.sysadmit.com/2018/05/VMware-esxi-ver-modulos-de-ram.html>

Una vez visto el potencial que tiene, aunque parezca extraño, acabamos utilizando más la línea de comandos que la interfaz gráfica para según qué cosas.

Existen varias herramientas para la administración desde la línea de comandos donde algunas tienen un uso concreto y único, en este capítulo os explicaré estas herramientas.

En este capítulo veremos cómo se usan estas herramientas, para qué sirven, y se añadirán una serie de ejemplos de cada una de estas herramientas que más utilizo en el día a día.

También, veremos cómo algunas de estas herramientas tienen usos parecidos para según qué cosas y cuál de estas es más efectiva.

Una parte importante que se verá en este capítulo es como habilitar y/o deshabilitar la Shell y el acceso remoto SSH, ya que por defecto vienen deshabilitados por seguridad.

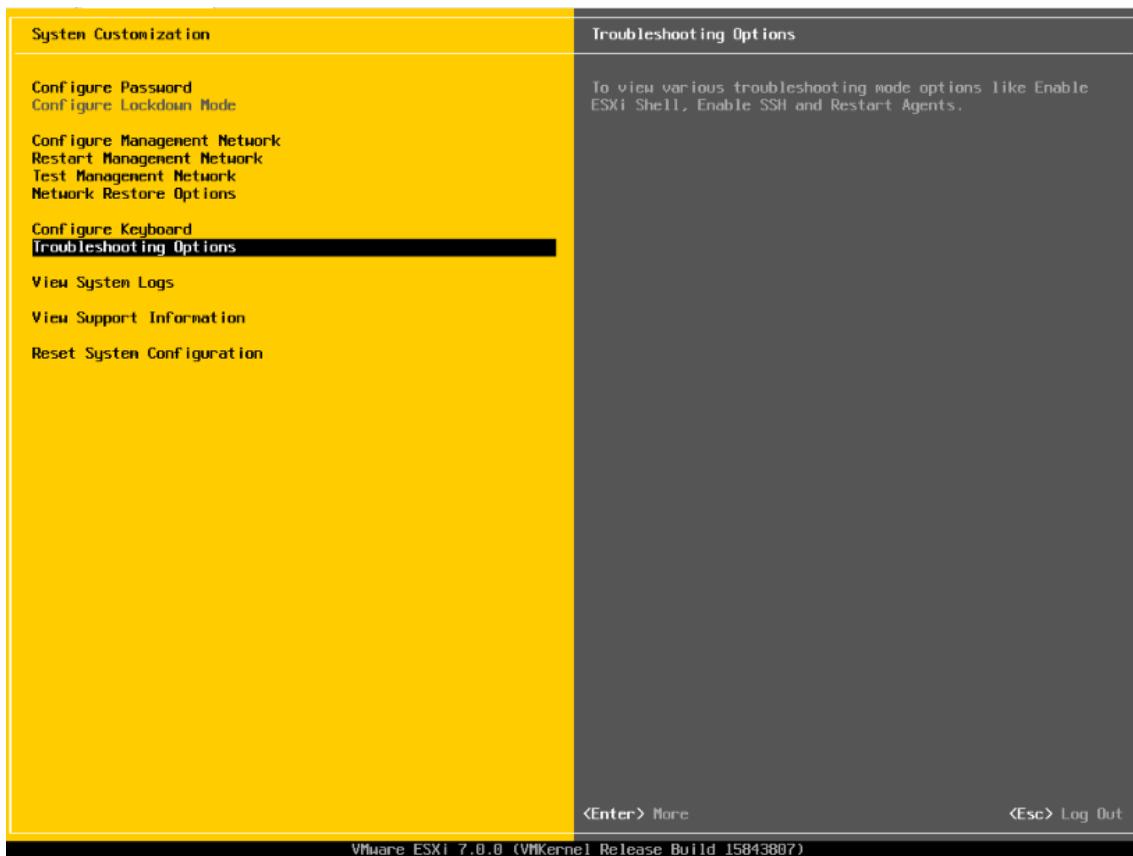
HABILITAR ESXi SHELL Y ACCESO REMOTO SSH

Para poder acceder a la consola de forma local o remota (cliente SSH), lo primero que tendremos que hacer es habilitarla, ya que por defecto y por temas de seguridad estas dos opciones vienen deshabilitadas.

Añadir y comentar que el acceso SSH y a la Shell podemos habilitarlo desde el vCenter.

HABILITAR ESXi SHELL

Para habilitar el acceso a la Shell de forma local, pulsaremos F2 desde la pantalla principal y nos dirigiremos ha **Troubleshooting Options**.

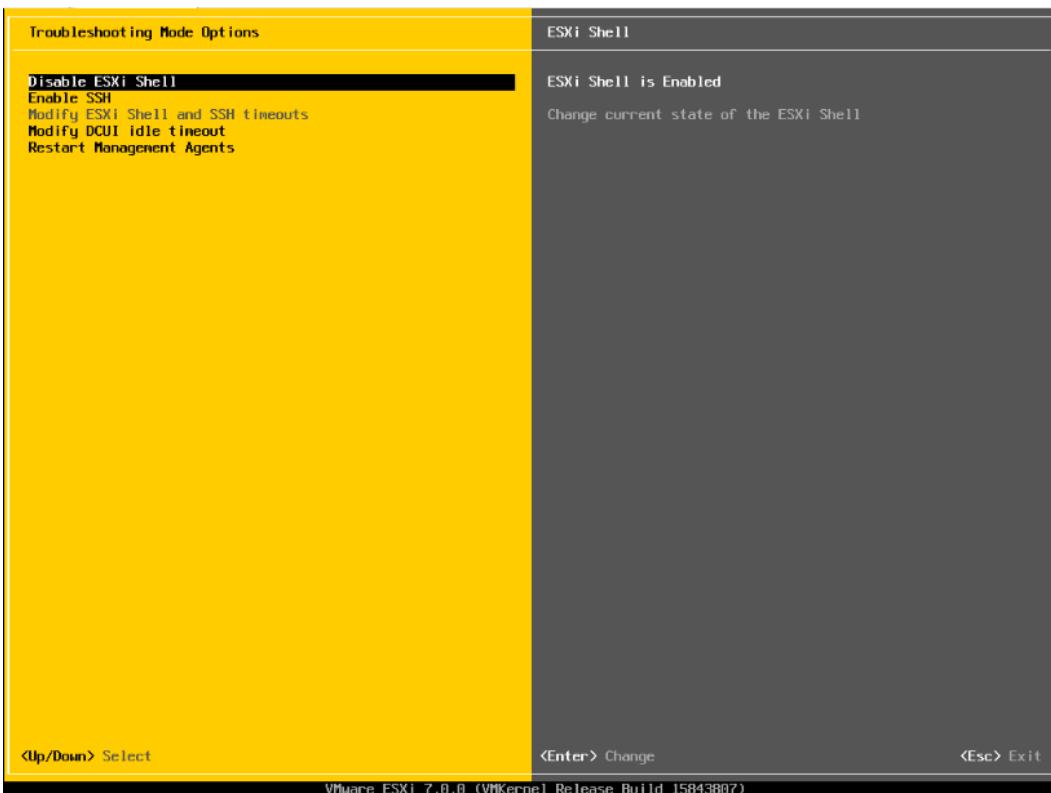


En este menú nos encontraremos con todas las opciones relacionadas con este temario, exceptuando la última opción que sería la de reinicio de los agentes de administración.

Para habilitar la ESXi Shell, nos posicionamos sobre **Enable ESXi Shell** y pulsamos **Enter**.



Enseguida veremos cómo ha cambiado el estado a **Enabled**.



Realizado este cambio y pulsado las teclas **Alt + F1**, podremos acceder a la Shell del ESXi

```

ESXi 7.0.0 http://www.vmware.com
Copyright (c) 2007-2020 VMware, Inc.

esxivi7 login: root
Password:
The time and date of this login have been sent to the system logs.

WARNING:
All commands run on the ESXi shell are logged and may be included in
support bundles. Do not provide passwords directly on the command line.
Most tools can prompt for secrets or accept them from standard input.

VMware offers supported, powerful system administration tools. Please
see www.vmware.com/go/sysadmintools for details.

The ESXi Shell can be disabled by an administrative user. See the
vSphere Security documentation for more information.

[root@esxivi7:~]

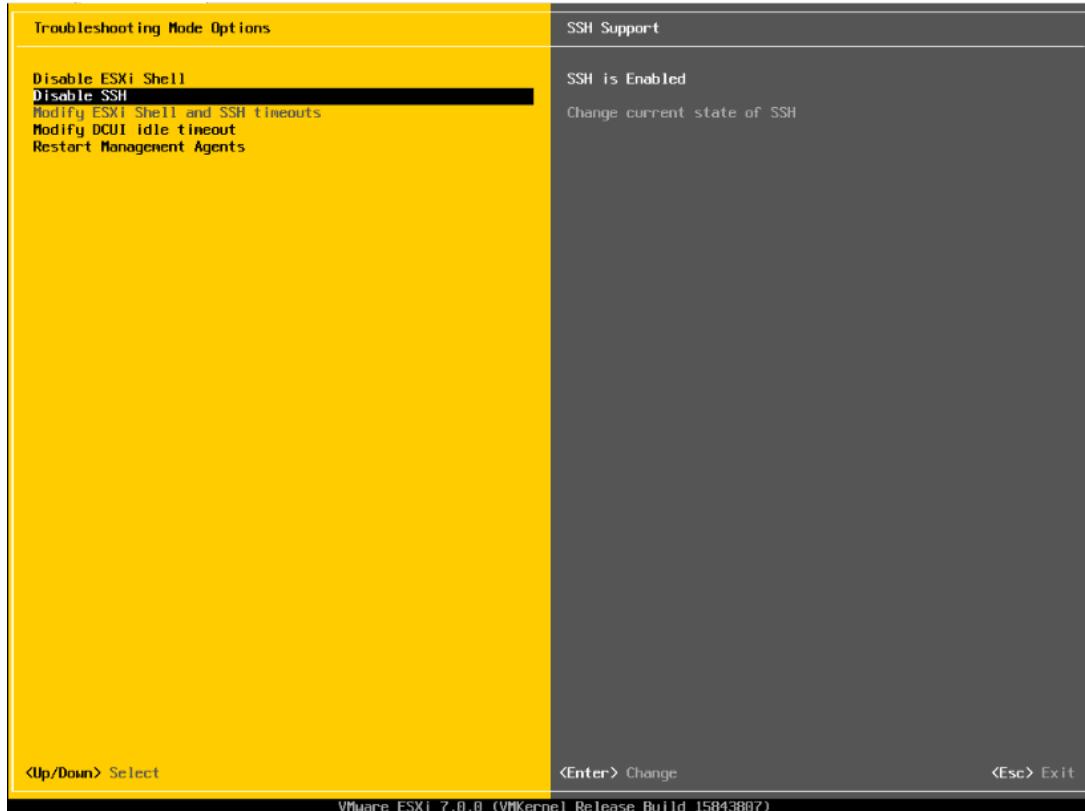
```

También podemos habilitar y deshabilitar la Shell con **vim-cmd** de la siguiente forma

```
[root@esxivi7:~] vim-cmd hostsvc/start_esx_shell
[root@esxivi7:~] vim-cmd hostsvc/stop_esx_shell
[root@esxivi7:~] █
```

HABILITAR ACCESO REMOTO SSH

Para habilitar el acceso remoto por SSH procederemos de la misma forma, nos posicionaremos sobre **Disable SSH** y pulsaremos la tecla **Enter** para habilitarlo.



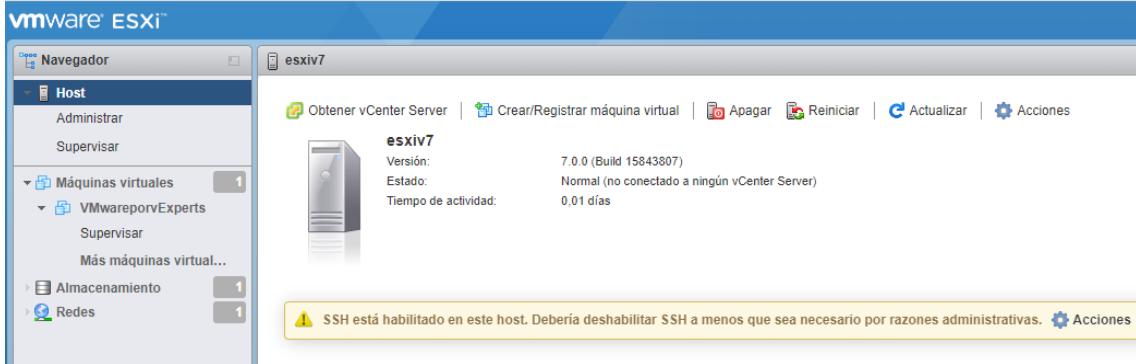
Realizado este cambio y usando un cliente SSH probaremos la conexión.

```
[root@esxiv7:~]
[root@esxiv7:~]
[root@esxiv7:~] hostname
esxiv7
[root@esxiv7:~]
```

También podemos habilitarlo con **vim-cmd** de la siguiente forma

```
[root@esxiv7:~] vim-cmd hostsvc/disable_ssh
[root@esxiv7:~] vim-cmd hostsvc/stop_ssh
[root@esxiv7:~]
```

Cuando habilitemos el servicio SSH en la interfaz web veremos un mensaje de advertencia



En siguiente enlace veremos cómo quitar esta advertencia en caso de querer tener el servicio SSH habilitado.

<https://aprendiendoavirtualizar.com/deshabilitar-warning-ssh-habilitado-esxi/>

VIM-CMD

QUE ES VIM-CMD

Vim-cmd (Virtual Infrastructure Management), vSphere Api command que está construida sobre el hostd que implementa la API.

<https://communities.VMware.com/docs/DOC-31025>

En ESXi, vim-cmd está ubicada en la ruta **/bin/vim-cmd**, que en realidad es un enlace simbólico a un host ejecutable como se muestra a continuación.

```
[root@esxiv7:~] ls -l /bin/vim-cmd
lrwxrwxrwx 1 root      root          10 Mar 16 10:37 /bin/vim-cmd -> /bin/hostd
[root@esxiv7:~]
```

Con vim-cmd puedes realizar operaciones relacionadas con el ESXi, máquinas virtuales, vSphere Replication. Es una de las herramientas de líneas de comando más completas, pero menos intuitiva y más difícil de usar (*esto bajo mi humilde opinión*).

Para conocer todos los subcomandos que podemos utilizar con vim-cmd, simplemente bastará con que escribamos vim-cmd y muestre las siguientes categorías.

```
[root@esxiv7:~] vim-cmd
Commands available under /:
hbrsvc/      internalsvc/  solo/          vmsvc/
hostsvc/     proxysvc/    vimsvc/        help
```

En vim-cmd existen 7 categorías, vamos a llamarles categorías de subcomandos, donde cada una de estas categorías tiene sus comandos correspondientes o incluso más subcategorías por debajo de estas categorías, la octava sería la ayuda (help),

Ejemplo:

Para usar la ayuda, pondremos siempre el **help** después de **vim-cmd** y seguido el comando completo.

```
[root@esxiv7:~] vim-cmd help /hostsvc/maintenance_mode_enter
Usage: maintenance_mode_enter [timeout] [vsanmode]
Put the host in maintenance mode.

[root@esxiv7:~] ■
```

Puede que en ocasiones el comando **help** no nos aporte mucho y el resultado puede que sea igual que si no lo ponemos, con la diferencia de que nos dice el siguiente mensaje **“Insufficient arguments”**, en cambio en el ejemplo anterior, antes de ejecutar el comando de poner en modo mantenimiento al ESXi, poniendo el comando **help** nos mostrará el uso de este comando.

```
[root@esxiv7:~] vim-cmd help vmsvc/device.diskadd
Usage: device.diskadd vmid size controller_number unit_number datastore [ctrlr_type] ← Sub comando Help añadido
Add a disk to this virtual machine.

[root@esxiv7:~] vim-cmd vmsvc/device.diskadd ← Sin añadir subcomando Help
Insufficient arguments.
Usage: device.diskadd vmid size controller_number unit_number datastore [ctrlr_type]
Add a disk to this virtual machine.
```

Cada una de estas categorías tiene una serie de comandos relacionados con una serie de objetos y usos.

Por ejemplo, en la categoría **vmsvc**, sus comandos están relacionados con operaciones de Máquinas Virtuales, información, Snapshots, estados de energía de las VMs.

```
[root@esxiv7:~] vim-cmd vmsvc
Commands available under vmsvc:
acquiremksticket           get.snapshotinfo
acquireticket                get.spaceNeededForConsolidation
createdummyvm               get.summary
destroy                      get.tasklist
device.connection            getallvms
device.connnusbdev          gethostconstraints
device.ctlradd                message
device.ctlrrmove              power.getstate
device.disconnusbdev         power.hibernate
device.diskadd                power.off
device.diskadexisting         power.on
device.diskextend             power.reboot
device.diskremove              power.reset
device.getdevices             power.shutdown
device.nvdimmadd              power.suspend
device.nvdimmremove           power.suspendResume
device.toolsSyncSet            queryftcompat
devices.createnic             reload
get.capability                 setscreeneres
get.config                     snapshot.create
get.config.cpuidmask          snapshot.dumpoption
get.configoption               snapshot.get
get.datastores                  snapshot.remove
get.disabledmethods            snapshot.removeall
get.environment                 snapshot.revert
get.filelayout                snapshot.setoption
get.filelayoutex               tools.cancelinstall
get.guest                      tools.install
get.guestheartbeatstatus       tools.upgrade
get.managedentitystatus        unregister
get.networks                   upgrade
get.runtime
```

En cambio, en la categoría **hostsvc**, todos sus comandos estarán relacionados en la administración del ESXi tales como habilitar, deshabilitar, parar e iniciar servicios, configuración de las reglas del firewall, etc.

```
[root@esx1v7:~] vim-cmd hostsvc:  
Commands available under hostsvc:  
advopt/           enable_ssh      refresh_services  
autostartmanager/   firewall_disable_ruleset  reset_service  
datastore/          firewall_enable_ruleset  runtimeinfo  
datastorebrowser/    get_service_status  set_hostid  
firmware/          hostconfig       standby_mode_enter  
net/               hosthardware    standby_mode_exit  
lsrc/              hostsummary     start_esx_shell  
storage/            maintenance_mode_enter  start_service  
summary/           maintenance_mode_exit  start_ssh  
vmotion/            pci_add         stop_esx_shell  
cpuinfo/            pci_remove      stop_service  
disable_esx_shell  queryconnectioninfo  stop_ssh  
enable_esx_shell   querydisabledmethods  task_list  
enable_esx_shell   refresh_firewall  updateSSLThumbprintsInfo  
[root@esx1v7:~]
```

COMO FUNCIONA VIM-CMD

Utilizar la herramienta vim-cmd puede ser poco intuitiva y difícil de entender, pero si aprendemos su uso, puede venirnos muy bien en situaciones difíciles

Para poder usar **vim-cmd**, tendremos que usar añadiendo la ruta entera, por ejemplo

Vim-cmd – categoría - comando – opciones

Si usásemos el comando help quedaría de la siguiente forma

Vim-cmd – help - categoría - comando – opciones

Con el siguiente ejemplo, veremos cómo podemos añadir un virtual disk a una máquina virtual

- Si escribimos **vim-cmd**, nos mostrará las categorías.

```
[root@esx1v7:~] vim-cmd  
Commands available under /:  
hbrsvc/      internalsvc/  solo/        vmsvc/  
hostsvc/     proxysvc/     vimsvc/     help
```

- Escribimos **vim-cmd** más la categoría **vmsvc** y nos mostrará los comandos que están dentro de la categoría.

```
[root@esx1v7:~] vim-cmd vmsvc  
Commands available under vmsvc:  
acquiremksticket  get.snapshotinfo  
acquiresicket     get.spaceNeededForConsolidation  
createdummyvm     get.summary  
destroy           get.tasklist  
device.connection  getallvms  
device.connsusbdev  gethostconstraints  
device.ctlradd     message  
device.ctlrremove   power.getstate  
device.disconusbdev  power.hibernate  
device.diskadd     power.off  
device.diskadexisting  power.on  
device.diskextend   power.reboot  
device.diskremove   power.reset  
device.getdevices   power.shutdown  
device.nvdimmadd    power.suspend  
device.nvdimmremove  power.suspendResume  
device.toolsSyncSet  queryfcompat  
devices.createnic   reload  
get.capability     setscreenes  
get.config         snapshot.create  
get.config.cidmask  snapshot.dumpoption  
get.config.option   snapshot.get  
get.datastores     snapshot.remove  
get.disabledmethods  snapshot.removeall  
get.environment    snapshot.revert  
get.filelayout     snapshot.setoption  
get.filelayoutex   tools.cancelinstall  
get.guest          tools.install  
get.guestheartbeat  tools.upgrade  
get.managedentitystatus  unregister  
get.networks       upgrade  
get.runtime
```

- Escribimos **vim-cmd vmsvc** más comando **device.diskadd** y nos dirá la información del comando y los parámetros que necesitaremos para ejecutar correctamente el comando completo.

```
[root@esxiv7:~] vim-cmd vmsvc/device.diskadd
Insufficient arguments.
Usage: device.diskadd vmid size controller_number unit_number datastore [ctrlr_type]
Add a disk to this virtual machine.
```

- Finalización del comando, donde añadiremos las opciones indicadas anteriormente, siendo

Vmid = **1**

Size = **5242880** (en KB)

Controller_number = **scsi0**

Unit_number = **1**

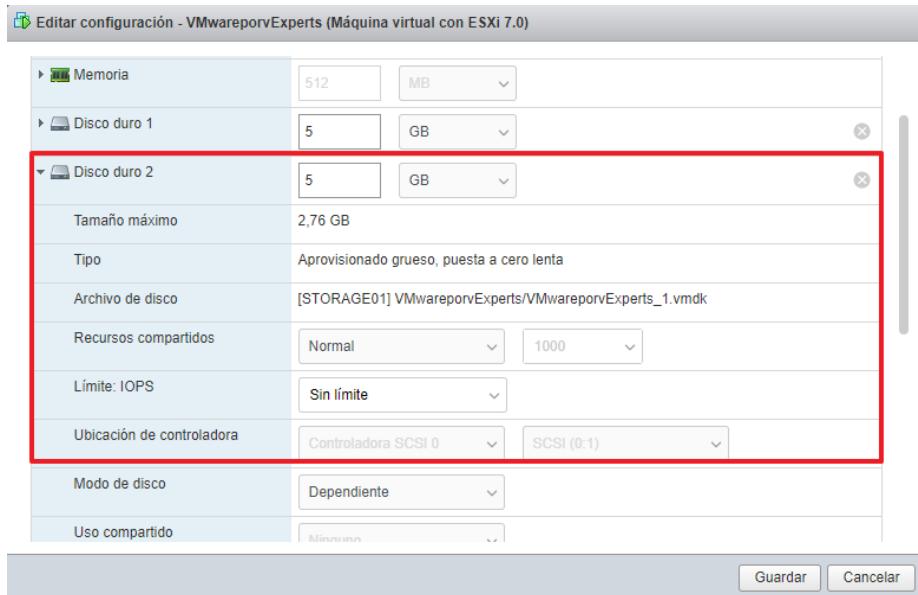
Datastore = **STORAGE01**

```
[root@esxiv7:~] vim-cmd vmsvc/device.diskadd 1 5242880 scsi0 1 STORAGE01
[root@esxiv7:~]
```

Si la salida del comando no muestra nada, significará que el comando se ha aplicado correctamente.

Vista grafica del resultado del comando anterior.

La creación del disco por defecto la crea Think provisioning



Observaciones

Dependiendo de los comandos que queramos utilizar también nos encontraremos con más categorías de subcomandos debajo de las primeras categorías, estas están marcadas con el símbolo "/" como se ve en la siguiente imagen.

```

[root@esxiv7:~] vim-cmd hostsvc
Commands available under hostsvc/:
advopt/           enable_ssh          refresh_services
autostartmanager/ firewall_disable_ruleset  reset_service
datastore/         firewall_enable_ruleset runtimeinfo
datastorebrowser/ get_service_status   set_hostid
firmware/          hostconfig          standby_mode_enter
net/              hosthardware        standby_mode_exit
rsrc/             hostsummary         start_esx_shell
storage/           maintenance_mode_enter start_service
summary/           maintenance_mode_exit start_ssh
vmotion/          pci_add            stop_esx_shell
cpuinfo/          pci_remove          stop_service
disable_esx_shell queryconnectioninfo stop_ssh
disable_ssh       querydisabledmethods task_list
enable_esx_shell  refresh_firewall updateSSLThumbprintsInfo
[root@esxiv7:~] vim-cmd hostsvc/datastore
Commands available under hostsvc/datastore/:
capabilities      remove
destroy           rename
info               summary
listsummary        vmfs_create
listvm            vmfs_extend
localds_create    vmfs_queryAvailableDisks
nas_create         vmfs_query_create_options
refresh           vmfs_query_extend_options
[root@esxiv7:~] vim-cmd hostsvc/datastore/summary
Insufficient arguments.
Usage: summary name

Retrieve the summary of a datastore by name.

[root@esxiv7:~] vim-cmd hostsvc/datastore/summary STORAGE01
(vim.Datastore.Summary) {
    datastore = 'vim.Datastore:5f283775-aa4d3f34-ebc3-000c29923bf1',
    name = "STORAGE01",
    url = "/vmfs/volumes/5f283775-aa4d3f34-ebc3-000c29923bf1",
    capacity = 10468982784,
    freeSpace = 2968518656,
    uncommitted = 0,
    accessible = true,
    multipleHostAccess = <unset>,
    type = "VMFS",
    maintenanceMode = <unset>
}

```

VIM-CMD Y EJECUCIONES ENCADENADAS

Con vim-cmd también podemos utilizar el encadenamiento de comandos Unix para ayudarnos a extraer la información de las salidas de los comandos de vim-cmd, ya que según que comandos puede mostrarnos demasiada información, resultando muy difícil buscar el dato que realmente queremos y difícil de interpretar.

Aquí dependerá de tus conocimientos en Unix / Linux, donde si tienes grandes conocimientos podrás ejecutar unos buenos comandos encadenados, yo como tengo unos conocimientos básicos, hago lo que puedo ;)

Os dejo este enlace interesante explicando el binario busybox usado por VMware para poder utilizar algunos comandos Unix/Linux

<https://www.sysadmit.com/2015/03/VMware-esxi-busybox.html>

En la siguiente imagen si ejecutamos el comando **vim-cmd hostsvc/net/vnic_info** para mostrarnos la información de las virtual nics, dependiendo de la cantidad de vrnics que tenga el host ESXi configuradas, el resultado de la salida del comando será un largo churro de información.

```
[root@esxiv7:~] vim-cmd hostsvc/net/vnic_info
(vim.host.VirtualNic) [
  (vim.host.VirtualNic) {
    device = "vmk0",
    key = "key-vim.host.VirtualNic-vmk0",
    portgroup = "Management Network",
    spec = (vim.host.VirtualNic.Specification) {
      dynamicProperty = <unset>
      ip = (vim.host.IpConfig) {
        dhcp = true,
        ipAddress = "192.168.0.149",
        subnetMask = "255.255.255.0",
        ipV6Config = (vim.host.IpConfig.IpV6AddressConfiguration) [
          ipV6Address = (vim.host.IpConfig.IpV6Address) [
            (vim.host.IpConfig.IpV6Address) {
              ipAddress = "fe80::20c:29ff:fe92:3bd3",
              prefixLength = 64,
              origin = "other",
              dadState = "preferred",
              lifetime = <unset>,
              operation = <unset>
            }
          ],
          autoConfigurationEnabled = true,
          dhcpV6Enabled = false
        ]
      },
      mac = "00:0c:29:92:3b:d3",
      distributedVirtualPort = (vim.dvs.PortConnection) null,
      portgroup = "Management Network",
      mtu = 1500,
      tsoEnabled = true,
      netStackInstanceKey = "defaultTcipipStack",
      opaqueNetwork = (vim.host.VirtualNic.OpaqueNetworkSpec) null,
      externalId = <unset>,
      pinnedPnic = <unset>,
      ipRouteSpec = (vim.host.VirtualNic.IpRouteSpec) null,
      systemOwned = <unset>
    },
    port = <vim.host.PortGroup.Port:key-vim.host.PortGroup.Port-67108873>
  }
]
```

Para evitarnos toda esa información de poco interés, podremos encadenar la salida del anterior comando con grep, de esta manera la salida del comando estará con la información que realmente necesitamos.

En este ejemplo lo que veremos encadenando con el comando grep es ver los dispositivos y sus correspondientes Mac Address.

```
[root@esxiv7:~] vim-cmd hostsvc/net/vnic_info | grep -e device -e mac
  device = "vmk0",
  mac = "00:0c:29:92:3b:d3",
  device = "vmk1",
  mac = "00:50:56:6e:1e:70",
[root@esxiv7:~] ■
```

Incluso podremos sacar por poner un ejemplo las 10 primeras líneas a un fichero con el comando **vim-cmd hostsvc/net/vnic_info | head -10 >> /tmp/info.txt**

```
[root@esxiv7:~] vim-cmd hostsvc/net/vnic_info | head -10 >> /tmp/info.txt
[root@esxiv7:~] cat /tmp/info.txt
(vim.host.VirtualNic) [
  (vim.host.VirtualNic) {
    device = "vmk0",
    key = "key-vim.host.VirtualNic-vmk0",
    portgroup = "Management Network",
    spec = (vim.host.VirtualNic.Specification) {
      dynamicProperty = <unset>
      ip = (vim.host.IpConfig) {
        dhcp = true,
        ipAddress = "192.168.0.149",
```

COMANDOS MÁS UTILIZADOS CON VIM-CMD

A continuación, me gustaría mostrarte cuáles son los comandos que más utilizo con vim-cmd, muchos no he llegado a utilizar, pero en cambio otros he tenido que utilizarlos en bastantes ocasiones. También he de decir que hay otros comandos que realmente no aportan mucho y que están un poco de adorno.

1. Encender Máquinas virtuales con vim-cmd

Este comando lo he tenido que usar para apagar o encender máquinas virtuales por no poder acceder vía web o a través del vSphere Client.

Para ver que Máquinas Virtuales tenemos en el host ESXi ejecutaremos el siguiente comando, donde nos quedaremos con el dato del campo vmid que usaremos más adelante.

Vim-cmd vmsvc/getallvms

```
[root@esxiv7:~] vim-cmd vmsvc/getallvms
Vmid      Name           File                Guest OS    Version   Annotation
1        VMwareporvExperts [STORAGE01] VMWareporvExperts/VMwareporvExperts.vmx centos8_64Guest  vmx-17  https://www.vmwareporvexperts.org/
[root@esxiv7:~] ■
```

Como en el comando anterior no nos informa el estado de la máquina virtual, lo comprobaremos con el siguiente comando:

Vim-cmd vmsvc/power.getstate + vmid

```
[root@esxiv7:~] vim-cmd vmsvc/power.getstate 1
Retrieved runtime info
Powered off
```

Como está apagada, ejecutaremos el siguiente comando para encenderla:

Vim-cmd vmsvc/power.on + vmid

```
[root@esxiv7:~] vim-cmd vmsvc/power.on 1
Powering on VM:
```

Y lo comprobaremos con:

Vim-cmd vmsvc/power.getstate + vmid

```
[root@esxiv7:~] vim-cmd vmsvc/power.getstate 1
Retrieved runtime info
Powered on
```

2. Realizar Backup de la configuración del host ESXi

Como la configuración de ESXi tiene un autoguardado (creo que cada hora) en el fichero /bootbank/state.tgz, ejecutaremos **vim-cmd hostsvc/firmware/sync_config**, de esta manera nos aseguraremos de que esté guardada la configuración más actual.

```
[root@esxiv7:~] vim-cmd hostsvc/firmware/sync_config
[root@esxiv7:~] ■
```

Seguido ejecutamos **vim-cmd hostsvc/firmware/backup_config**, que como resultado recibiremos un link de descarga de un fichero **configbundle-xxx.tgz**, donde sustituiremos el asterisco por la ip del ESXi.

```
[root@esxiv7:~] vim-cmd hostsvc/firmware/backup_config
Bundle can be downloaded at : http://*/downloads/52e5191d-b8f9-665a-722b-b3c8a8a79a04/configBundle-esxiv7.tgz
```

Este fichero es el que tendremos que guardar en el caso de tener que realizar una restauración de la configuración.

Podemos usar wget para descargarlo y guardarlo en un Datastore, aunque lo recomendable es sacar fuera esta copia de seguridad usando aplicaciones tales como WinSCP o MobaXterm.

Para descargar este fichero en el Datastore haremos

```
wget http://ip_esxi /download/xxxxxxxxxxxxxxxxxxxxxxx/xxx.tgz -P /vmfs/volumes/datastore
```

```
[root@esxiv7:~] wget http://192.168.0.149/downloads/52e5191d-b8f9-665a-722b-b3c8a8a79a04/configBundle-esxiv7.tgz -P /vmfs/volumes/STORAGE01/configBundle-esxiv7. 100% |*****| 59418 0:00:00 ETA
```

3. Restauración de la copia de seguridad de ESXi

Para que podamos restaurar desde el fichero de copia de seguridad del ESXi, tendremos que detener o mover las VMs a otro ESXi y ponerlo en modo mantenimiento.

Para eso, usaremos:

```
vim-cmd /hostsvc/maintenance
```

```
[root@esxiv7:~] vim-cmd /hostsvc/maintenance_mode_enter  
[root@esxiv7:~] █
```

NOTAS: para poder realizar la restauración de la configuración con éxito, tenemos que renombrar el fichero como *configBundle.tgz* y tiene que ser copiado o movido al directorio /tmp del ESXi.

También es importante, que el ESXi al que vayamos a restaurar la configuración, tenga la misma versión y build que en el momento en que se hizo la copia de seguridad.

```
[root@esxiv7:~] mv /vmfs/volumes/STORAGE01/configBundle-esxiv7.tgz /tmp/configBundle.tgz  
[root@esxiv7:~]
```

```
[root@esxiv7:~] vim-cmd /hostsvc/firmware/restore_config /vmfs/volumes/STORAGE01/configBundle-esxiv7.tgz  
(vim.fault.FileNotFound) {  
    faultCause = (vmddl.MethodFault) null,  
    faultMessage = <unset>,  
    file = "/tmp/configBundle.tgz"  
    msg = "Received SOAP response fault from [<cs p:000000dc0022d760, TCP:localhost:8307>]: restoreConfiguration  
file /tmp/configBundle.tgz was not found"  
}  
[root@esxiv7:~] cp /vmfs/volumes/STORAGE01/configBundle-esxiv7.tgz /tmp/  
[root@esxiv7:~] █
```

Así que movemos el fichero al directorio /tmp/ y lo renombramos como configBundle.tgz

Una vez que ejecutemos el comando de restauración de la configuración del ESXi con **vim-cmd /hostsvc/firmware/restore_config /ruta/configBundle.tgz**, si todo va bien, el ESXi hará un reinicio para aplicar los cambios.

```
[root@esxiv7:~] vim-cmd /hostsvc/firmware/restore_config /tmp/configBundle.tgz  
Remote side unexpectedly closed network connection
```

En el siguiente enlace podremos ver varios casos de uso con vim-cmd

<https://aprendiendoavirtualizar.com/category/vim-cmd/>

ESXCLI

QUE ES ESXCLI

ESXCLI es otra de las grandes herramientas de interfaz de línea de comandos única y exclusivamente para administrar el hypervisor ESXi.

En cada nueva versión del hypervisor, aparecen nuevos comandos / namespaces, en este link podéis encontrar los nuevos namespaces de la versión ESXi v7.0.

<https://www.virten.net/2020/04/new-esxcli-commands-in-vsphere-7-0/>

Esxcli es una de las herramientas más usadas; es más intuitiva y fácil de usar que vim-cmd,

Al igual que vim-cmd, escribiendo esxcli mostrará todos los namespaces, las opciones disponibles y las descripciones de cada una, algo muy importante para tener un mínimo de información de para qué sirve.

```
[root@esxiv7:] esxcli
Usage: esxcli [options] {namespace}+ {cmd} {cmd options}

Options:
--formatter=FORMATTER          Override the formatter to use for a given command. Available formatter: keyvalue, csv, xml
--screen-width=SCREENWIDTH     Use the specified screen width when formatting text
--debug                         Enable debug or internal use options
--version                        Display version information for the script
--?, --help                       Display usage information for the script

Available Namespaces:
device                         Device manager commands
esxcli                         Commands that operate on the esxcli system itself allowing users to get additional information.
fcoe                           VMware FCoE commands
graphics                        VMKernel graphics commands.
hardware                        VMKernel hardware properties and commands for configuring hardware.
iscsi                           VMware iSCSI commands.
network                          Operate on the system to the maintenance of networking on an ESX host. This includes a wide variety of commands to manipulate virtual networking components (vswitch, portgroup, etc) as well as local host IP, DNS and general host networking settings.
nvme                           VMware NVMe driver operations.
rdma                            Operations that pertain to remote direct memory access (RDMA) protocol stack on an ESX host.
sched                           VMKernel system properties and commands for configuring scheduling related functionality.
software                         Manage the ESXi software image and packages
storage                          VMKernel storage commands.
system                           VMKernel system properties and commands for configuring properties of the kernel core system and related system services.
vm                             A small number of operations that allow a user to Control Virtual Machine operations.
vsan                            VMware vSAN commands
```

En el siguiente enlace está la guía de toda la referencia de comandos con esxcli de ESXi 7.0

<https://code.VMware.com/docs/11743/esxi-7-0-esxcli-command-reference//reference.html>

Con el comando **esxcli esxcli command list** podemos ver el uso de todos los namespaces de esxcli

Namespace	Command	Description
device	add	Add a device to enable a software device driver.
device.alias	get	Display hardware location info for a device alias.
device.alias	list	List device aliases.
device.driver	list	Show driver status for specific devices.
device.software	add	Add a device to enable a software device driver.
device.software	list	List software devices.
device.software.device	remove	Remove a software device.
esxcli.command	list	List all of the esxcli commands.
fcoe.adapter	list	List FCoE-capable CNA devices.
fcoe.adapter	remove	Initiate FCoE adapter removal.
fcoe.nic	disable	Disable configuration of FCoE storage on behalf of an FCoE-capable CNA upon next boot.
fcoe.nic	discover	Initiate FCoE adapter discovery on behalf of an FCoE-capable CNA.
fcoe.nic	enable	Enable an FCoE-capable NIC if it is disabled .
fcoe.nic	list	List FCoE-capable CNA devices.
fcoe.nic	remove	Initiate FCoE device destroy on behalf of an FCoE-capable PNIC.
graphics.device	set	Set options on FCoE-capable CNA.
graphics.device.stats	list	List all of the graphic devices on this host.
graphics.host	get	List graphics device statistics.
graphics.host	refresh	Get host graphics properties.
graphics.vm	set	Refresh host graphics properties.
hardware.bootdevice	list	List active VMs associated with graphics devices.
hardware.clock	get	List the boot device order, if available, for this host.
hardware.cpu.cpuid	set	Display the current hardware clock time.
hardware.cpu.cpuid.raw	get	Set the hardware clock time. Any missing parameters will default to the current time.
hardware.cpu.global	list	Get short of CPUINN needed for a CPU (deprecated , use: esxcli hardware cpu cpuid raw list).
hardware.cpu.global	get	Get all CPUID fields for a CPU.
hardware.cpu	set	Get properties that are global to all CPUs.
hardware.ipmi.bmc	list	Set properties that are global to all CPUs.
hardware.ipmi.bmc	get	Get IPMI Baseboard Management Controller (BMC) properties.
hardware.ipmi.fru	set	Set IPMI Baseboard Management Controller (BMC) properties. Changes take effect immediately.
hardware.ipmi.fru	get	Get IPMI Field Replaceable Unit (FRU) device details.
hardware.ipmi.sdr	list	List IPMI Field Replaceable Unit (FRU) inventory.
hardware.ipmi.sdr	get	Get IPMI Sensor Data Repository (SDR) properties.
hardware.ipmi.sdr	list	List IPMI Sensor Data Repository.
hardware.ipmi.sel	clear	Clear IPMI System Event Log.

Para usar esxcli también podemos hacer uso de la ayuda, por ejemplo, queremos ver qué opciones nos da a la hora de apagar un ESXi

esxcli system shutdown poweroff –help

```
[root@esxivi7:~] esxcli system shutdown poweroff --help
Usage: esxcli system shutdown poweroff [cmd options]

Description:
  poweroff      Power off the system. The host must be in maintenance mode.

Cmd options:
  -d|-delay=<long>  Delay interval in seconds
  -r|-reason=<str>   Reason for performing the operation (required)
[root@esxivi7:~] ■
```

COMO FUNCIONA ESXCLI

Como he comentado anteriormente, esxcli tiene un uso más intuitivo y la nomenclatura sería la siguiente:

Esxcli [options] {namespace} + {cmd} [cmd options]

Por debajo de los namespaces podemos encontrar más namespaces por lo que otro ejemplo sería

Esxcli [options] {namespace 1} + {namespace 2} + {cmd} [cmd options]

Ejemplo:

Esxcli –formatter=csv network nic list

```
[root@esxivi7:~] esxcli --formatter=csv network nic list
AdminStatus,Description,Driver,Duplex,Link,LinkStatus,MACAddress,MTU,Name,PCIDevice,Speed,
Up,VMware Inc. vmxnet3 Virtual Ethernet Controller,nvmmxnet3,Full,Up,Up,00:0c:29:92:3b:d3,1500,vmnic0,0000:0b:00.0,10000,
Up,VMware Inc. vmxnet3 Virtual Ethernet Controller,nvmmxnet3,Full,Up,Up,00:0c:29:92:3b:dd,1500,vmnic1,0000:13:00.0,10000,
Up,VMware Inc. vmxnet3 Virtual Ethernet Controller,nvmmxnet3,Full,Up,Up,00:0c:29:92:3b:e7,1500,vmnic2,0000:1b:00.0,10000,
Up,VMware Inc. vmxnet3 Virtual Ethernet Controller,nvmmxnet3,Full,Up,Up,00:0c:29:92:3b:f1,1500,vmnic3,0000:04:00.0,10000,
[root@esxivi7:~] ■
```

Donde le diríamos que nos liste todas las nic físicas del host ESXi en formato csv.

Hacemos otro ejemplo con más opciones y desgranando cada parte del comando.

Esxcli: empezamos el comando.

```
[root@esxivi7:~] esxcli
Usage: esxcli [options] {namespace}+ {cmd} [cmd options]

Options:
  --formatter=FORMATTER          Override the formatter to use for a given command. Available formatter: keyvalue, csv, xml
  --screen-width=SCREENWIDTH    Use the specified screen width when formatting text
  --debug                         Enable debug or internal use options
  --version                       Display version information for the script
  -, --help                        Display usage information for the script

Available Namespaces:
  device                         Device manager commands
  esxcli                         Commands that operate on the esxcli system itself allowing users to get additional information.
  fcoe                           VMware FCoE commands.
  graphics                        VMware graphics commands.
  hardware                        VMKernel hardware properties and commands for configuring hardware.
  iscsi                           VMware iSCSI commands.
  network                          Operations that pertain to the maintenance of networking on an ESX host. This includes a wide variety of commands to manipulate virtual networking components (vswitch, portgroup, etc) as well as local host IP, DNS and general host networking settings.
  nvme                            VMware NVMe driver operations.
  rdma                            Operations that pertain to remote direct memory access (RDMA) protocol stack on an ESX host.
  sched                           VMKernel system properties and commands for configuring scheduling related functionality.
  software                         Manage the ESXi software image and packages
  storage                          VMware storage commands.
  system                           VMKernel system properties and commands for configuring properties of the kernel core system and related system services.
  vm                             A small number of operations that allow a user to control Virtual Machine operations.
  vsan                            VMware vSAN commands
```

Options: --formatter=csv para que el resultado del comando completo sea en formato csv, de momento la salida es idéntica a la anterior al no estar el comando completo.

```
[root@esxiv7:~] esxcli --formatter=csv
Usage: esxcli [options] {namespace}+ {cmd} [cmd options]

Options:
  --formatter=FORMATTER          Override the formatter to use for a given command. Available formatter: keyvalue, csv, xml
  --screen-width=SCREENWIDTH     Use the specified screen width when formatting text
  --debug                         Enable debug or internal use options
  --version                        Display version information for the script
  -?, --help                        Display usage information for the script

Available Namespaces:
  device                         Device manager commands
  esxcli                         Commands that operate on the esxcli system itself allowing users to get additional information.
  fcoe                           VMware FCOE commands.
  graphics                        VMware graphics commands.
  hardware                        VMKernel hardware properties and commands for configuring hardware.
  iscsi                           VMware iSCSI commands.
  network                         Operations that pertain to the maintenance of networking on an ESX host. This includes a wide variety of commands to manipulate virtual networking components (vswitch, portgroup, etc) as well as local host IP, DNS and general host networking settings.
  nvme                            VMware NVMe driver operations.
  rdma                            Operations that pertain to remote direct memory access (RDMA) protocol stack on an ESX host.
  sched                           VMKernel system properties and commands for configuring scheduling related functionality.
  software                         Manage the ESXi software image and packages
  storage                          VMware storage commands.
  system                           VMKernel system properties and commands for configuring properties of the kernel core system and related system services.
  vm                             A small number of operations that allow a user to Control Virtual Machine operations.
  vsan                           VMware vSAN commands
```

Namespace 1: network

```
[root@esxiv7:~] esxcli --formatter=csv
Usage: esxcli [options] {namespace}+ {cmd} [cmd options]

Options:
  --formatter=FORMATTER          Override the formatter to use for a given command. Available formatter: csv, keyvalue, xml
  --screen-width=SCREENWIDTH     Use the specified screen width when formatting text
  --debug                         Enable debug or internal use options
  --version                        Display version information for the script
  -?, --help                        Display usage information for the script

Available Namespaces:
  device                         Device manager commands
  esxcli                         Commands that operate on the esxcli system itself allowing users to get additional information.
  fcoe                           VMware FCOE commands.
  graphics                        VMware graphics commands.
  hardware                        VMKernel hardware properties and commands for configuring hardware.
  iscsi                           VMware iSCSI commands.
  network                         Operations that pertain to the maintenance of networking on an ESX host. This includes a wide variety of commands to manipulate virtual networking components (vswitch, portgroup, etc) as well as local host IP, DNS and general host networking settings.
  nvme                            VMware NVMe driver operations.
  rdma                            Operations that pertain to remote direct memory access (RDMA) protocol stack on an ESX host.
  sched                           VMKernel system properties and commands for configuring scheduling related functionality.
  software                         Manage the ESXi software image and packages
  storage                          VMware storage commands.
  system                           VMKernel system properties and commands for configuring properties of the kernel core system and related system services.
  vm                             A small number of operations that allow a user to Control Virtual Machine operations.
  vsan                           VMware vSAN commands
```

Namespace 2: nic, nivel inferior del namespace network

```
[root@esxiv7:~] esxcli --formatter=csv network
Usage: esxcli network {cmd} [cmd options]

Available Namespaces:
  ens                            Commands to list and manipulate Enhanced Networking Stack (ENS) feature on virtual switch.
  firewall                       A set of commands for firewall related operations
  ip                             Operations that can be performed on vmknics
  multicast                      Operations having to do with multicast
  nic                          Operations having to do with the configuration of Network Interface Card and getting and updating the NIC settings.
  port                           Commands to get information about a port
  sriovnic                      Operations having to do with the configuration of SRIOV enabled Network Interface Card and getting and updating the NIC settings.
  vm                            A set of commands for VM related operations
  vswitch                        Commands to list and manipulate Virtual Switches on an ESX host.
  diag                           Operations pertaining to network diagnostics
```

Cmd: comandos disponible down, get, list, set, up

```
[root@esxiv7:~] esxcli --formatter=csv network nic
Usage: esxcli network nic {cmd} [cmd options]

Available Namespaces:
coalesce      Commands to access coalesce parameters for a NIC
dcb           Commands regarding DCB (Data Center Bridging) protocol.
queue         Commands to access RX/TX netqueue features on a NIC
ring          Commands to access NIC RX/TX ring buffer parameters
vlan          Get VLAN information for a NIC
attachment    Commands to access attachment info for a NIC.
cso           Commands to access checksum offload settings for a NIC
eeprom        Commands to access EEPROM for a NIC
negotiate     Commands to access negotiation feature on a NIC
pauseParams   Commands to access pause parameters for a NIC
register      Commands to access registers for a NIC
selftest      Commands to access self test feature on a NIC
sg             Commands to access scatter-gather settings for a NIC
software      Commands to access NIC feature software simulation settings
stats         Get packet statistics for a NIC
tso            Commands to access TCP segmentation offload settings for a NIC

Available Commands:
down          Bring down the specified network device.
get           Get the generic configuration of a network device
list          This command will list the Physical NICs currently installed and loaded on the system.
set           Set the general options for the specified ethernet device.
up           Bring up the specified network device.
```

Cmd options -n| --nic-name=<str>

```
[root@esxiv7:~] esxcli --formatter=csv network nic down --help
Usage: esxcli network nic down [cmd options]

Description:
  down          Bring down the specified network device.

Cmd options:
  -n|-nic-name=<str>  The name of the NIC to configured. This must be one of the cards listed in the nic list command. (required)
[root@esxiv7:~] ■
```

Resultado

```
[root@esxiv7:~] esxcli --formatter=csv network nic down -n vmnic2
[root@esxiv7:~] esxcli --formatter=csv network nic list
AdminStatus,Description,Driver,Duplex,Link,LinkStatus,MACAddress,MIU,Name,PCIDevice,Speed,
Up,VMware Inc. vmxnet3 Virtual Ethernet Controller,nvmxnet3,Full,Up,Up,00:0c:29:92:3b:d3,1500,vmnic0,0000:0b:00.0,10000,
Up,VMware Inc. vmxnet3 Virtual Ethernet Controller,nvmxnet3,Full,Up,Up,00:0c:29:92:3b:dd,1500,vmnic1,0000:13:00.0,10000,
Down,VMware Inc. vmxnet3 Virtual Ethernet Controller,nvmxnet3,Half,Down,Down,00:0c:29:92:3b:e7,1500,vmnic2,0000:1b:00.0,
Up,VMware Inc. vmxnet3 Virtual Ethernet Controller,nvmxnet3,Full,Up,Up,00:0c:29:92:3b:f1,1500,vmnic3,0000:04:00.0,10000,
```

Una vista más ordenada quitando la opción **-formatter**

Name	PCI Device	Driver	Admin Status	Link Status	Speed	Duplex	MAC Address	MTU	Description
vmnic0	0000:0b:00.0	nvmxnet3	Up	Up	10000	Full	00:0c:29:92:3b:d3	1500	VMware Inc. vmxnet3 Virtual Ethernet Controller
vmnic1	0000:13:00.0	nvmxnet3	Up	Up	10000	Full	00:0c:29:92:3b:dd	1500	VMware Inc. vmxnet3 Virtual Ethernet Controller
vmnic2	0000:1b:00.0	nvmxnet3	Down	Down	0	Half	00:0c:29:92:3b:e7	1500	VMware Inc. vmxnet3 Virtual Ethernet Controller
vmnic3	0000:04:00.0	nvmxnet3	Up	Up	10000	Full	00:0c:29:92:3b:f1	1500	VMware Inc. vmxnet3 Virtual Ethernet Controller

ESXCLI Y EJECUCIONES ENCADENADAS

Al igual que vim-cmd, podemos combinar esxcli con los clásicos comandos Unix. Un buen ejemplo es cuando quieras buscar los comandos y opciones con determinado namespace.

Os dejo este enlace interesante explicando el binario busybox usado por VMware para poder utilizar algunos comandos Unix/Linux

<https://www.sysadmit.com/2015/03/VMware-esxi-busybox.html>

Un ejemplo sería ver los usos del namespace **vsan** y todas sus opciones con

esxcli esxcli command list | grep vsan | less

```
[root@esxiv7:~] esxcli esxcli command list | grep vsan | less
vsan.cluster                           get      Get information about the vSAN cluster that this host is joined to.
vsan.cluster                           join     Join the host to a vSAN cluster.
vsan.cluster                           leave    Leave the vSAN cluster the host is currently joined to.
vsan.cluster                           new     Create a vSAN cluster with current host joined. A random sub-cluster UUID will be generated.
vsan.cluster.preferredfaultdomain     get      Get the preferred fault domain for a stretched cluster.
vsan.cluster.preferredfaultdomain     set      Set the preferred fault domain for a stretched cluster.
vsan.cluster                           restore   Restore the persisted vSAN cluster configuration.
vsan.cluster.unicastagent            add     Add a unicast agent to the vSAN cluster configuration.
vsan.cluster.unicastagent            clear    Removes all unicast agents in the vSAN cluster configuration.
vsan.cluster.unicastagent            list     List all unicast agents in the vSAN cluster configuration.
vsan.cluster.unicastagent            remove   Remove a unicast agent from the vSAN cluster configuration.
vsan.cmmds.timemachine              get      Get vSAN CMMDs time machine configuration.
vsan.cmmds.timemachine              set      Configure vSAN CMMDs time machine.
vsan.datastore                          add     Add a new datastore to the vSAN cluster. This operation may be used to add either a new local datastore or a remote datastore. Adding local datastore is only allowed if vSAN is enabled on the host. In general, add should be done at cluster level. Across a vSAN cluster vSAN datastores should be in sync.
vsan.datastore                          clear    Remove all but the default datastore from the vSAN cluster. This operation is only allowed if vSAN is enabled on the host. In general, clear should be done at cluster level. Across a vSAN cluster vSAN datastores should be in sync.
vsan.datastore                          list     List datastores in the vSAN cluster.
vsan.datastore.name                    get      Get vSAN datastore name.
vsan.datastore.name                    set      Configure vSAN datastore name. In general, rename should always be done at cluster level. Across a vSAN cluster vSAN datastore name should be in sync.
vsan.datastore                          remove   Remove a datastore from the vSAN cluster. This operation is only allowed if vSAN is enabled on the host. In general, remove should be done at cluster level. Across a vSAN cluster vSAN datastores should be in sync.
vsan.debug.advcfg                     list     List all advanced configuration options with non-default values.
vsan.debug.controller                 list     Print detailed information about all vSAN disk controllers (output may change between releases)
vsan.debug.disk                       list     Print detailed information about all vSAN disks in the cluster.
vsan.debug.disk                       overview Print overview information about all vSAN disks in the cluster.
vsan.debug.disk.summary               get      Print summary information about all vSAN disks in the cluster.
vsan.debug.evacuation                precheck Examine what it takes if an entity (disk group or host) is evacuated in various modes (Action). The result is accurate when all hosts in the vSAN cluster are of the same version and have the same disk format.
vsan.debug.limit                     get      Print summary information about vSAN limits (output may change between releases)
vsan.debug.memory                    list     Print both userworld and kernel memory consumptions of vSAN.
vsan.debug.mob                       start    Start vSAN Managed Object Browser Service.
vsan.debug.mob                       status   Query vSAN Managed Object Browser Service is running or not.
vsan.debug.mob                     stop     Stop vSAN Managed Object Browser Service.
vsan.debug.object.health.summary    get      Print health summary information about all vSAN objects in the cluster (output may change between releases)
vsan.debug.object                   list     Print detailed information about vSAN objects in the cluster. This command would only show 100 objects at most by default.
```

En el siguiente enlace podremos ver varios casos de uso con esxcli

<https://aprendiendoavirtualizar.com/category/esxcli/>

COMANDOS MÁS UTILIZADOS CON ESXCLI

Como con otros sistemas operativos, en determinadas ocasiones tenemos más costumbre de utilizar la línea de comandos que la interfaz gráfica por facilidad o por rapidez, aunque la mayoría de las veces usamos la línea de comando para temas de troubleshooting, ya que es más fácil que deje de funcionar la interfaz gráfica que la Shell.

Vamos a ver unos ejemplos que nos pueden venir bien.

1. Modo mantenimiento y reiniciar host ESXi con esxcli

Este es uno de los comandos más utilizados, seguramente porque la interfaz web no funciona correctamente y no podemos reiniciarlo por esta vía.

Para reiniciar el ESXi podemos acceder por SSH (si está habilitada) o si no de forma local directamente con la Shell.

También tener en cuenta que mientras haya máquinas virtuales ejecutándose en el ESXi no podremos ponerlo en modo mantenimiento, así que o moveremos las máquinas virtuales o las apagaremos.

Pasamos a modo mantenimiento con **esxcli system maintenanceMode set -e true** y comprobamos el estado con **esxcli system maintenanceMode get**

```
[root@esxiv7:~] esxcli system maintenanceMode set -e true
[root@esxiv7:~] esxcli system maintenanceMode get
Enabled
[root@esxiv7:~] ■
```

Con el siguiente comando reiniciaremos el ESXi:

esxcli system shutdown reboot -d 10 -r "razon"

NOTAS: el tiempo de retraso es mínimo 10 segundos y la razón del reinicio va entre comillas en caso de ser dos o más palabras.

```
[root@esxiv7:~] esxcli system shutdown reboot -d 10 -r "Mal funcionamiento"
[root@esxiv7:~]
Remote side unexpectedly closed network connection
```

2. Apagar máquinas virtuales con esxcli

Otro comando muy utilizado, sobre todo en situaciones en que la máquina virtual se ha quedado en un estado “colgado / congelado”, para comprobar la lista de máquinas virtuales y su World ID es el siguiente:

esxcli vm process list

Donde anotaremos el World ID que en este caso es 68118

```
[root@esxiv7:~] esxcli vm process list
VMwareporvExperts
    World ID: 68118
    Process ID: 0
    VMX Cartel ID: 68117
    UUID: 56 4d 1b b0 68 ea 98 a9-f8 50 02 a4 c7 7a 36 2d
    Display Name: VMwareporvExperts
    Config File: /vmfs/volumes/5f283775-aa4d3f34-ebc3-000c29923bf1/VMwareporvExperts/VMwareporvExperts.vmx
[root@esxiv7:~] ■
```

Con **esxcli vm process kill –help** veremos las opciones que nos da este comando, ya que tenemos 3 tipos de **kill (soft, hard y force)**, dependiendo de la gravedad usaremos una u otra.

```
[root@esxiv7:~] esxcli vm process kill --help
Usage: esxcli vm process kill [cmd options]
Description:
  Kill           Used to forcibly kill Virtual Machines that are stuck and not responding to normal stop operations.

Cmd options:
  -t<--type=<str>  The type of kill operation to attempt. There are three types of VM kills that can be attempted: [soft, hard, force]. Users should always attempt 'soft' kills first, which will give the VMX process a chance to shutdown cleanly (like kill or kill -SIGTERM). If that does not work move to 'hard' kills which will shutdown the process immediately (like kill -9 or kill -SIGKILL). 'force' should be used as a last resort attempt to kill the VM. If all three fail then a reboot is required. (required)
  -w<--world-id=<long>  The World ID of the Virtual Machine to kill. This can be obtained from the 'vm process list' command (required)
[root@esxiv7:~] ■
```

Ejecutamos **esxcli vm process kill -t soft -w 68118** para un apagado normal, para comprobar con esxcli si la máquina virtual está apagada, ejecutaremos **esxcli vm process list**, si al ejecutar este comando no aparece la máquina virtual, significa que está apagada.

```
[root@esxiv7:~] esxcli vm process kill -t soft -w 68118
[root@esxiv7:~]
[root@esxiv7:~] esxcli vm process list
[root@esxiv7:~]
```

3. Comprobación y aplicación de revisiones con esxcli

Con esxcli también podemos instalar, actualizar, aplicar parches y revisiones a nuestros host ESXi.

Con **esxcli software vib list**, listaremos todos los vib instalados en nuestro ESXi.

Por poner un ejemplo vamos a actualizar el de **cpu-microcode** que tiene una versión **7.0.0-1.0.15843807**

Lo normal en estos casos es que nos descarguemos el fichero .zip de [My VMware](#) y lo aplicaremos a todos, pero en este caso solo voy a actualizar un vib.

Name	Version	Vendor	Acceptance Level	Install Date
bnxtnet	216.0.50.0-4vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
bnxtroce	216.0.58.0-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
brcmfcoe	12.0.1.0.1500.0-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
brcmnmvmefc	12.4.293.2-3vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
elxiscsi	12.0.0.1200.0-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
elxnet	12.0.0.1250.0-5vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
i40en	1.8.1.16-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
i40iwn	1.1.2.5-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
iavmd	2.0.0.0.1055-3vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
igbn	0.1.1.0-6vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
iser	1.1.0.0-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
ixgben	1.7.1.26-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
lpfc	12.4.293.3-5vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
lpnic	11.4.62.0-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
lsi-mr3	7.712.50.00-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
lsi-msgpt2	20.00.06.00-2vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
lsi-msgpt35	13.00.00.12-00-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
lsi-msgpt3	17.00.10.00-10vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
mtip32xx-native	3.9.8-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
ne1000	0.8.4-10vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
nenic	1.0.29.0-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
nfnic	4.0.0.44-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
nhpsa	2.0.50-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
nmlx4-core	3.19.16.7-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
nmlx4-en	3.19.16.7-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
nmlx4-rdma	3.19.16.7-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
nmlx5-core	4.19.16.7-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
nmlx5-rdma	4.19.16.7-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
ntg3	4.1.4.1-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
nvme-pcie	1.2.2.13-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
nvmerdma	1.0.0.0-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
nvmxnet3-ens	2.0.0.0.22-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
nvmxnet3	2.0.0.0.30-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
pvscsi	0.1-2vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
qcnic	1.0.15.0-8vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
qedentv	3.12.1.0-23vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
qedrndt	3.12.1.2-12vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
qfle3	1.0.66.0-5vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
qfle3f	1.0.51.0-12vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
qfle3i	1.0.15.0-6vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
qflege	1.1.0.11-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
rste	2.0.2.0088-7vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
sfvmk	2.0.0.0.1004-3vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
smartpqi	1.0.4.3011-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
vmkata	0.1-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
vmkfcoe	1.0.0.0.2-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
vmkusb	0.1-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
vmm-ahci	1.3.9-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
cpu-microcode	7.0.0.0-1.0.15843807	VMware	VMwareCertified	2020-07-16
crx	7.0.0.0-1.0.15843807	VMWare	VMwareCertified	2020-07-16
elx-esx-libelxima.so	12.0.0.1200.0-2vmw.700.1.0.15843807	VMware	VMwareCertified	2020-07-16
esx-base	7.0.0.0-1.0.15843807	VMware	VMwareCertified	2020-07-16
esx-dvfilter-generic-fastpath	7.0.0.0-1.0.15843807	VMware	VMwareCertified	2020-07-16
esx-ui	1.34.0-15603211	VMware	VMwareCertified	2020-07-16
esx-update	7.0.0.0-1.0.15843807	VMware	VMwareCertified	2020-07-16
esx-xserver	7.0.0.0-1.0.15843807	VMware	VMwareCertified	2020-07-16
loadesx	7.0.0.0-1.0.15843807	VMware	VMwareCertified	2020-07-16
lsuv2-hpv2-hpsa-plugin	1.0.0.0-2vmw.700.1.0.15843807	VMware	VMwareCertified	2020-07-16

Ponemos el host ESXi en modo mantenimiento con **esxcli system maintenanceMode set -e true**

```
[root@esxiv7:~] esxcli system maintenanceMode set -e true
```

Actualizamos el vib con:

esxcli software vib update -v /ruta_donde_hemos_descargado_el_fichero_.vib

y reiniciamos el host ESXi según nos indique la salida del comando.

```
[root@esxiv7:~] esxcli software vib update -v /tmp/VMware_bootbank_cpu-microcode_7.0.0-1.25.16324942.vib
Installation Result
Message: The update completed successfully, but the system needs to be rebooted for the changes to be effective.
Reboot Required: true
VIBs Installed: VMware_bootbank_cpu-microcode_7.0.0-1.25.16324942
VIBs Removed: VMware_bootbank_cpu-microcode_7.0.0-1.0.15843807
VIBs Skipped:
[root@esxiv7:~] ■
```

Una vez reiniciado el ESXi comprobaremos si se ha actualizado correctamente el fichero vib, viendo que ha pasado a la versión **7.0.0-1.25.16324942**

Name	Version	Vendor	Acceptance Level	Install Date
bnxtnet	216.0.50.0-4vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
bnxtroce	216.0.58.0-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
brcmfcoe	12.0.1500.0-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
brcmnmvmefc	12.4.293.2-3vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
elxiscsi	12.0.1200.0-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
elxnet	12.0.1250.0-5vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
i40en	1.8.1.16-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
i40iwn	1.1.2.5-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
iavmd	2.0.0.1055-3vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
igbn	0.1.1.0-6vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
iser	1.1.0.0-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
ixgben	1.7.1.26-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
lpfc	12.4.293.3-5vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
lpnic	11.4.62.0-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
lsi-mr3	7.712.50.00-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
lsi-msgpt2	20.00.06.00-2vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
lsi-msgpt35	13.00.12.00-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
lsi-msgpt3	17.00.10.00-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
mtip32xx-native	3.9.8-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
ne1000	0.8.4-10vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
nenic	1.0.29.0-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
nfnic	4.0.0.44-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
nhpssa	2.0.50-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
nmlx4-core	3.19.16.7-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
nmlx4-en	3.19.16.7-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
nmlx4-rdma	3.19.16.7-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
nmlx5-core	4.19.16.7-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
nmlx5-rdma	4.19.16.7-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
ntg3	4.1.4.1-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
nvme-pcie	1.2.2.13-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
nvmerdma	1.0.0.0-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
nvmxnet3-ens	2.0.0.22-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
nvmxnet3	2.0.0.30-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
pvscsi	0.1-2vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
qcnic	1.0.15.0-8vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
qedentv	3.12.1.0-23vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
qedrntv	3.12.1.2-12vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
qfle3	1.0.66.0-5vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
qflef	1.0.51.0-12vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
qfle3i	1.0.15.0-6vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
qflege	1.1.0.11-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
rste	2.0.0.2.0088-7vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
sfvmk	2.0.0.1004-3vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
smartpqci	1.0.0.4.3011-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
vmkata	0.1-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
vmkfcue	1.0.0.0.2-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
vmkusb	0.1-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
vmm-ahci	1.3.9-1vmw.700.1.0.15843807	VMW	VMwareCertified	2020-07-16
cpu-microcode	7.0.0-1.25.16324942	VMware	VMwareCertified	2020-08-10
crx	7.0.0-1.0.15843807	VMware	VMwareCertified	2020-07-16
elx-esx-libelxima.so	12.0.1200.0-2vmw.700.1.0.15843807	VMware	VMwareCertified	2020-07-16
esx-base	7.0.0-1.0.15843807	VMware	VMwareCertified	2020-07-16
esx-dvfilter-generic-fastpath	7.0.0-1.0.15843807	VMware	VMwareCertified	2020-07-16
esx-ui	1.34.0-15603211	VMware	VMwareCertified	2020-07-16
esx-update	7.0.0-1.0.15843807	VMware	VMwareCertified	2020-07-16
esx-xserver	7.0.0-1.0.15843807	VMware	VMwareCertified	2020-07-16
loadesx	7.0.0-1.0.15843807	VMware	VMwareCertified	2020-07-16
lsuv2-hpv2-hpsa-plugin	1.0.0-2vmw.700.1.0.15843807	VMware	VMwareCertified	2020-07-16

VMKFSTOOLS

QUE ES VMKFSTOOLS

vmkfstools es otra de las herramientas de línea de comandos usada por ESXi para administrar discos virtuales, dispositivos de almacenamiento y volúmenes VMFS.

Con este comando podemos administrar, crear, clonar discos virtuales, manipular el file system de un almacenamiento VMFS.

Este comando es muy usado para temas de troubleshooting, consolidación de disco virtual, para convertir discos virtuales, renombrar discos virtuales etc

Como cualquier herramienta o comando disponemos de una ayuda, en el caso de esta la ayuda la ejecutaríamos con **vmkfstools -H** o **vmkfstools –help**, podemos usar cualquiera de las dos formas, la larga y la de letras únicas.

```
[root@esxiv7:~] vmkfstools -H

OPTIONS FOR FILE SYSTEMS:

vmkfstools -C --createfs [vmfs5|vmfs6|vfat]
  -S --setfsname fsName
  -Y --unmapGranularity #[bBsSkKmMgGtT]
  -O --unmapPriority <none|low|medium|high>
  -Z --spanfs span-partition
  -G --growfs grown-partition
deviceName

  -P --queryfs -h --humanreadable
  -T --upgradevmfs
vmfsPath
  -y --reclaimBlocks vmfsPath [--reclaimBlocksUnit #blocks]

OPTIONS FOR VIRTUAL DISKS:

vmkfstools -c --createvirtualdisk #[bBsSkKmMgGtT]
  -d --diskformat [zeroedthick
                  |thin
                  |eagerzeroedthick
                  ]
  -a --adaptertype [deprecated]
  -W --objecttype [file|vsan|vvol|pmem|upit]
  --policyFile <fileName>
-w --writerozeros
-j --inflatedisk
-k --eagerzero
-K --punchzero
-U --deletevirtualdisk
-E --renamevirtualdisk srcDisk
-i --clonevirtualdisk srcDisk
  -d --diskformat [zeroedthick
                  |thin
                  |eagerzeroedthick
                  |rdm:<device>|rdmp:<device>
                  |2gbsparse]
  -W --object [file|vsan|vvol]
  --policyFile <fileName>
  -N --avoidnativeclone
-X --extendvirtualdisk #[bBsSkKmMgGtT]
  [-d --diskformat eagerzeroedthick]
-M --migratevirtualdisk
-r --createrdm /vmfs/devices/disks/...
--sectorSize [512n|4kn]
-q --queryrdm
-z --createrdmppassthru /vmfs/devices/disks/...
-v --verbose #
-g --geometry
-x --fix [check|repair]
-e --chainConsistent
-Q --objecttype name/value pair
--uniqueblocks childDisk
--dry-run [-K]
vmfsPath
```

```

OPTIONS FOR DEVICES:
  -L --lock [reserve|release|lunreset|targetreset|busreset|readkeys|readresv|
             |registerkey|clearallkeys
             ] /vmfs/devices/disks/...
  -B --breaklock /vmfs/devices/disks/...

OPTIONS FOR VMFS MODULE:
  --traceConfig [0|1]
  --dataTracing [0|1]
  --traceSize <x> (MB)
vmkfstools -H --help

```

COMO FUNCIONA VMKFSTOOLS

El funcionamiento es bastante sencillo, veamos un ejemplo para la creación de un disco virtual de un tamaño y un formato.

Comando [options1] + [options2] + [cmd options]

Vmkfstools -c 2G -d thin /ruta_donde_almacenaremos/el_disco_virtual

```
[root@esxiv7:~] vmkfstools -c 2G -d thin /vmfs/volumes/STORAGE01/VMwareporvExperts/VMwareporExperts_1.vmdk
Create: 100% done.
[root@esxiv7:~] █
```

El tamaño del disco lo podremos indicar en:

- K (kilobytes, tamaño mínimo 1024 KB)
- M (Megabytes)
- G (Gigabyte)
- T (Terabyte)

Y el formato de disco en:

- Thin
- Zeroedthick
- Eagerzeroedthick

Seguido indicaremos la ruta completa en caso de no estar posicionado sobre la carpeta de la máquina virtual donde queremos ubicar el nuevo disco virtual.

EJEMPLOS CON VMKFSTOOLS

Como en temas anteriores vamos a ver una serie de ejemplos más usados con comando, desde mi experiencia, los casos más usados han sido sobre discos virtuales.

1. Clonar disco virtual

Este seguramente lo usaremos en bastantes ocasiones, ya que es una buena forma de hacer una “copia-plantilla” de una máquina virtual si no disponemos de vCenter.

La máquina virtual tiene que estar apagada.

La ejecución del comando es sencilla

```
vmkfstools -i nombredeldiscooriginal.vmdk -d formatodisco  
nombredeldiscoclon.vmdk
```

de no estar posicionados en la carpeta de la máquina virtual, tendremos que indicar la ruta completa de origen y destino.

```
[root@esxiv7:/vmfs/volumes/5f283775-aa4d3f34-ebc3-000c29923bf1/VMwareporvExperts] vmkfstools -i VMwareporvExperts.vmdk -d thin VMwareporvExpertsClon.vmdk  
Destination disk format: VMFS thin-provisioned  
Cloning disk 'VMwareporvExperts.vmdk'...  
Clone: 100% done.
```

Comprobamos que aparecen los discos dentro de la carpeta de la máquina virtual.

```
[root@esxiv7:/vmfs/volumes/5f283775-aa4d3f34-ebc3-000c29923bf1/VMwareporvExperts] ls -l  
total 5244928  
-rw----- 1 root root 5368709120 Aug 11 17:23 VMwareporvExperts-flat.vmdk  
-rw----- 1 root root 270840 Aug 11 17:27 VMwareporvExperts.nvram  
-rw----- 1 root root 456 Aug 11 17:23 VMwareporvExperts.vmdk  
-rw-r--r-- 1 root root 0 Aug 11 17:23 VMwareporvExperts.vmsd  
-rwxr-xr-x 1 root root 3326 Aug 11 17:27 VMwareporvExperts.vmx  
-rw----- 1 root root 5368709120 Aug 11 17:27 VMwareporvExpertsclon-flat.vmdk  
-rw----- 1 root root 509 Aug 11 17:27 VMwareporvExpertsClon.vmdk  
-rw-r--r-- 1 root root 108418 Aug 11 17:27 vmware.log  
[root@esxiv7:/vmfs/volumes/5f283775-aa4d3f34-ebc3-000c29923bf1/VMwareporvExperts]
```

2. Convertir disco de Thin a Thick

Otro uso muy común es cuando tienes un disco en formato thin y quieres convertirlo en formato thick, conocido como “inflate”. Este proceso convierte el disco sin perder los datos.

Ejecutamos el siguiente comando:

```
vmkfstools -j VMwareporvExperts.vmdk
```

```
[root@esxiv7:/vmfs/volumes/5f283775-aa4d3f34-ebc3-000c29923bf1/VMwareporvExperts] vmkfstools -j VMwareporvExperts.vmdk  
vmfsDisk: 1, rdmDisk: 0, blockSize: 1048576  
 Inflate: 100% done.
```

3. Renombrar disco virtual

Si no renombramos correctamente el disco virtual, tendremos problemas para iniciar la máquina virtual.

Utilizando el siguiente comando nos aseguraremos de que este procedimiento lo haremos de la forma correcta, ya que además de renombrar el fichero vmdk también hará lo suyo con el flat.vmdk.

```
vmkfstools -E nombrediscoviejo.vmdk nombredisconuevo.vmdk
```

```
[root@esxiv7:/vmfs/volumes/5f283775-aa4d3f34-ebc3-000c29923bf1/VMwareporvExperts] vmkfstools -E VMwareporvExperts.vmdk VMwareporvExpertsorg.vmdk  
[root@esxiv7:/vmfs/volumes/5f283775-aa4d3f34-ebc3-000c29923bf1/VMwareporvExperts] ls -l  
total 5244928  
-rw----- 1 root root 270840 Aug 11 17:27 VMwareporvExperts.nvram  
-rw-r--r-- 1 root root 0 Aug 11 17:23 VMwareporvExperts.vmsd  
-rwxr-xr-x 1 root root 3107 Aug 11 17:45 VMwareporvExperts.vmx  
-rw----- 1 root root 5368709120 Aug 11 17:46 VMwareporvExpertsorg-flat.vmdk  
-rw----- 1 root root 485 Aug 11 17:46 VMwareporvExpertsorg.vmdk  
-rw-r--r-- 1 root root 108418 Aug 11 17:27 vmware.log
```

En caso de no renombrar el disco virtual de forma correcta con vmkfstools y hacerlo de forma manual, en el fichero descriptor del fichero vmdk no se aplicarán los cambios y el campo “Extend description” seguirá poniendo el nombre antiguo del disco virtual.

```
[root@esx1v7:/vmfs/volumes/5f283775-aa4d3f34-ebc3-000c29923bf1/VMwareporvExperts] cat VMwareporvExpertsorg.vmdk
# Disk DescriptorFile
version=1
encoding=UTF-8"
CID=ffffffff
parentCID=ffffffff
createType="vmfs"

# Extent description
RW 10485760 VMFS "VMwareporvExpertsorg-flat.vmdk"

# The Disk Data Base
#DDB

ddb.adapterType = "lsilogic"
ddb.geometry.cylinders = "652"
ddb.geometry.heads = "255"
ddb.geometry.sectors = "63"
ddb.longContentID = "01d74f5af56510a5625ab30297a7657b"
ddb.thinProvisioned = "0"
ddb.uuid = "60 00 C2 96 d8 fe 7b a0-f1 19 c0 72 db 16 46 c7"
ddb.virtualHWVersion = "14"
[root@esx1v7:/vmfs/volumes/5f283775-aa4d3f34-ebc3-000c29923bf1/VMwareporvExperts] █
```

Os dejo este enlace con más ejemplos del uso de vmkfstools.

<https://aprendiendoavirtualizar.com/tag/vmkfstools/>



A RICOH
Company

El Valor de la Experiencia, la Pasión por el Futuro



40 años integrando soluciones críticas de IT produciendo ventajas competitivas para nuestros clientes.

Una cultura orientada a proporcionar servicios de alta calidad: **Net Promoter Score >75%** año tras año.

Un equipo de más de 25 especialistas VMware con el más alto nivel de certificaciones.

Único VMware Premier Partner durante 10 años consecutivos y primer Principal Partner en Iberia.

Top Master Services Partner de Iberia



Barcelona

C/ Av. Barcelona 115
08970 Sant Joan Despí (Barcelona)
Tel: +34 934 770 436

Madrid

Joséfa Valcárcel, 3-5
28027 Madrid
Tel: +34 917 413 633

Valencia

Av. Cortes Valencianas 39,
1º planta, D
20 46015 Valencia
Tel: +34 961 199 628

Bilbao

Juan de Ajuriaguerra, 9, 6º
48009 Bilbao
Tel: +34 944 470 466

A Coruña

Enrique Mariñas Romero, 36, Torre de Cristal 15009 A Coruña
Tel: +34 981 237 554 15

Lisboa

Edificios Mirante Estrada alfragide Lt 107 2720 Amadora - Portugal
Tel: +35 121 472 4090
totalstor@totalstor.com

Capítulo 4

NSX-T



Patricio Cerdá

@patote83

NSX-T

INTRODUCCIÓN

En este capítulo hablaremos acerca de NSX-T 3.0, incluyendo un detalle acerca de la arquitectura y cada uno de sus componentes.

Durante este capítulo intentaré explicar paso a paso y de manera sencilla como desplegar NSX-T 3.0, partiendo del plano de Management y Control, para luego preparar el plano de Datos.

Finalmente, veremos cómo desplegar Segmentos L2 y Gateways L3 con NSX-T, así como trabajar con las distintas opciones de Firewall que provee NSX-T.

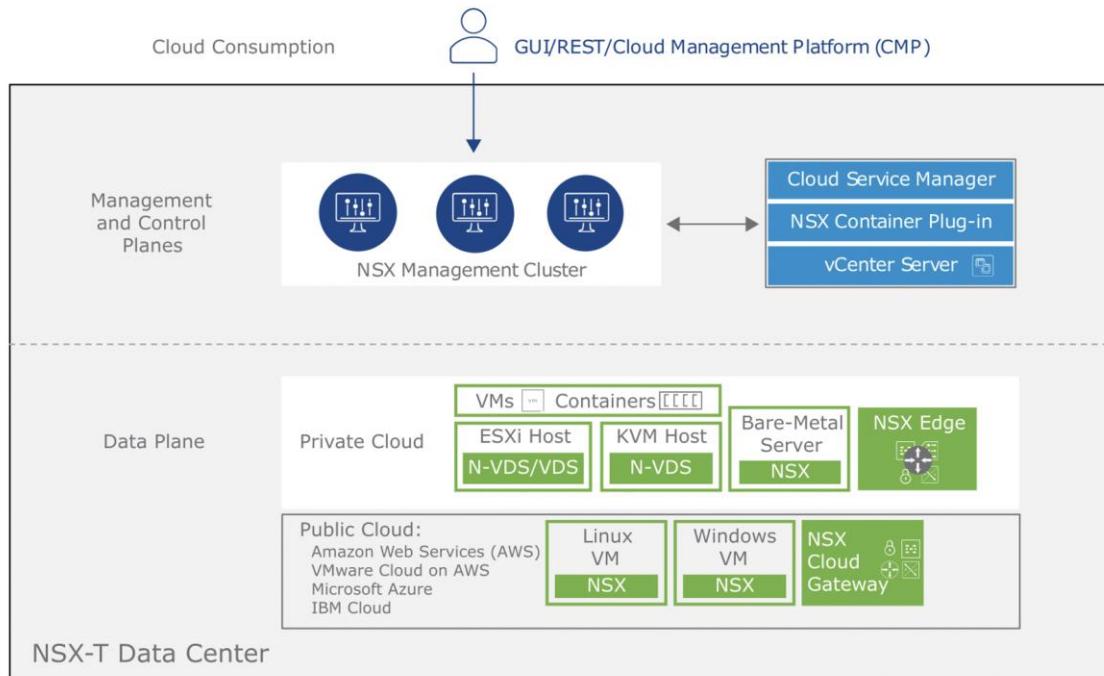
En este capítulo, lamentablemente no detallaremos cada una de las funciones de NSX-T, ya que eso inevitablemente requeriría un libro completo y dedicado para el tema, lo cual queda fuera del alcance y objetivo de este libro, el que busca entregar conocimientos sobre distintos ámbitos de las soluciones Cloud, y no sobre un único tópico.

Espero que disfruten de este capítulo y que sea de valor para ustedes.

ARQUITECTURA

La arquitectura básica de NSX-T, al igual que la arquitectura que se podía encontrar con NSX-V, consta de tres planos principales, los cuales detallaremos brevemente a continuación:

- Plano de Management
- Plano de Control
- Plano de Datos



En NSX-T Datacenter, el plano de Management y el plano de Control ahora son parte de un mismo NSX Management Cluster.

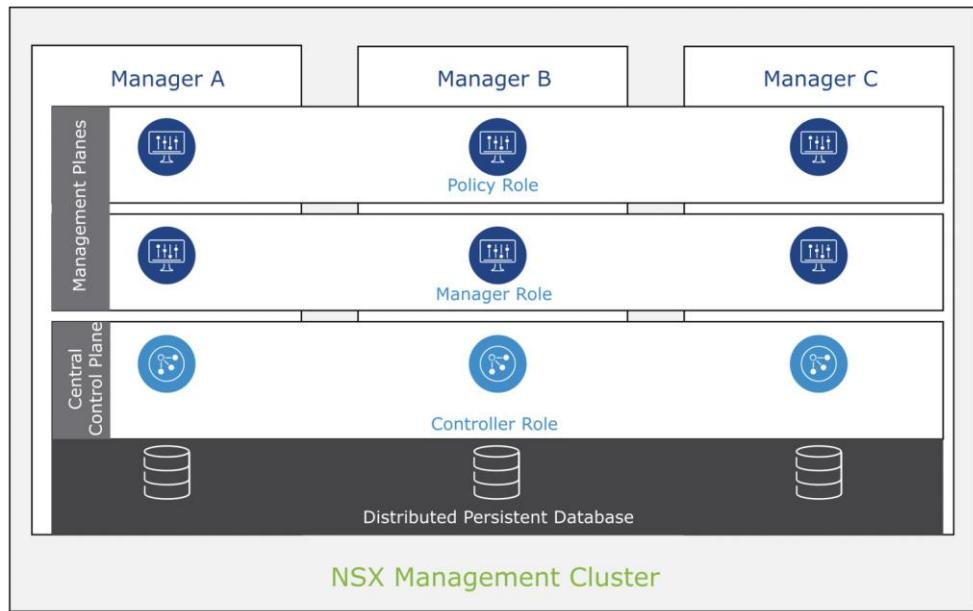
El plano de Management provee la interfaz de usuario (UI) para realizar la configuración y administración de la infraestructura NSX-T, así como el acceso via REST APIs.

El plano de Control es responsable de calcular y distribuir el estado de ejecución de la red.

NSX MANAGEMENT CLÚSTER

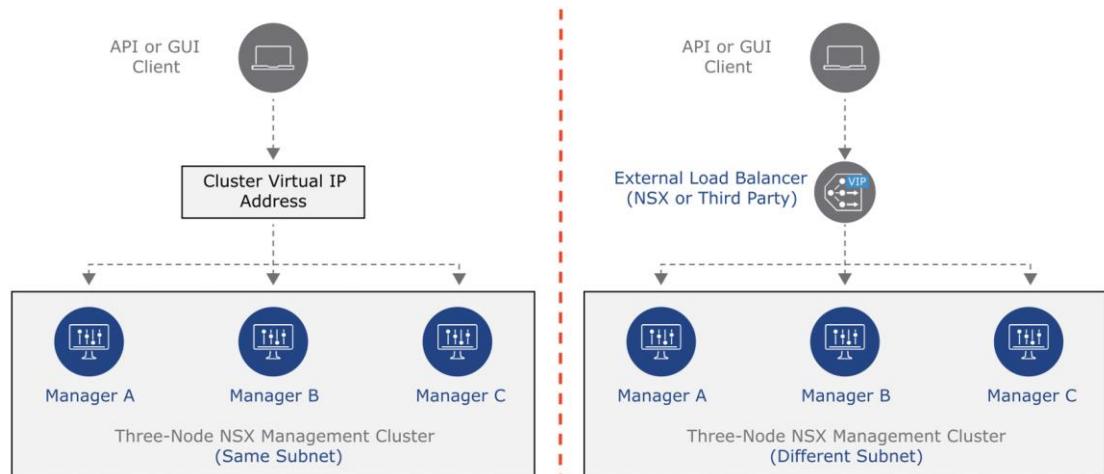
El NSX Management Cluster está formado por tres nodos NSX Manager, lo cual provee una mayor escalabilidad y disponibilidad. Cada nodo NSX Manager incluye los roles de Manager, Controller y de Policy.

El estado de la configuración de NSX-T es replicado de manera persistente en la base de datos distribuida, lo cual permite tener la misma configuración almacenada en todos los nodos del clúster.



Para acceder al NSX Management Cluster vía navegador tenemos dos alternativas principales:

- Virtual IP Address
- Load Balancer



VIRTUAL IP ADDRESS (VIP)

El acceso por **Virtual IP Address** requiere definir una dirección IP adicional que puede ser utilizada para acceder al clúster, en vez de utilizar la IP individual de uno de los nodos. El funcionamiento es bastante sencillo:

- Un nodo NSX Manager es seleccionado como líder.
- La VIP es asignada al nodo líder
- Cuando nos conectamos a la VIP, básicamente nos estamos conectado al nodo líder.
- El tráfico no es balanceado entre los nodos NSX Manager cuando se usa la VIP.
- Esta VIP puede ser usado para conectarse vía navegador o vía API.
- Si el nodo líder falla, un nuevo nodo NSX Manager es seleccionado como líder, y la VIP es asignada al nuevo líder, lo cual asegura que podamos seguir accediendo al NSX Management Cluster

LOAD BALANCER EXTERNO

El acceso vía **Load Balancer** provee alta disponibilidad al NSX Management clúster, y como es de esperar, también permite balancear las conexiones a través de todos los nodos NSX Manager:

- Todos los nodos NSX Manager funcionan de manera activa.
- El tráfico a través de la VIP es balanceado entre todos los nodos NSX Manager.

PLANO DE MANAGEMENT

Dentro del plano de Management nos encontramos con dos roles, el rol de NSX Policy y el rol de NSX Manager.



NSX POLICY

El rol de NSX Policy nos provee de un punto único de configuración para todos los componentes de redes y seguridad de la infraestructura. Nos permite definir la configuración deseada a través de la UI de NSX Manager

Nota: El rol de NSX Policy es desplegado como parte del appliance de NSX Manager.

El rol de NSX Policy permite contar con una gestión centralizada para:

- Todo tipo de máquinas: VMs, Contenedores y servidores Bare Metal
- Cloud públicas
- Múltiples hipervisores y Compute Managers (como vCenter Server)

NSX MANAGER

El rol de NSX Manager se ejecuta en cada appliance NSX Manager y está encargado de múltiples funciones:

- Recibe y valida la configuración definida a través de NSX Policy
- Publica la configuración al Plano de Control Central (CCP)

El rol de NSX Manager también estará a cargo de la instalación y preparación de los componentes del plano de datos (nodos de transporte, NSX Edge, etc.). Al mismo tiempo se encargará de obtener estadísticas del plano de datos.

PLANO DE CONTROL

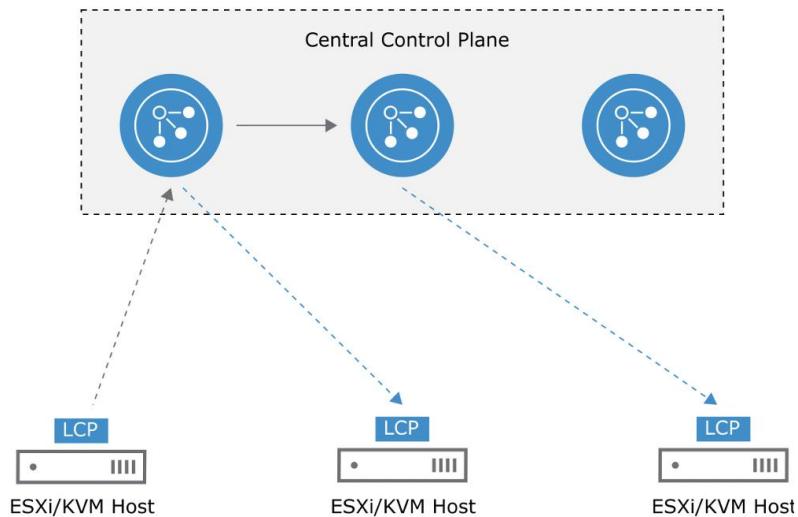
El NSX Controller mantiene el estado del sistema y configura el plano de datos. Entre las principales funciones del NSX Controller están:

- Proveer de funcionalidades de plano de control para los switches lógicos, routers lógicos, y firewall distribuido.
- Calcula todos los estados efímeros/temporales de los componentes, basado en la configuración realizada en el plano de management.
- Distribuye la información de topología reportada por el plano de datos.

Nota: Recordar que el NSX Controller y el NSX Manager comparten el mismo NSX Management Cluster con tres nodos.

El plano de control está dividido además en dos componentes principales:

- Plano de Control Central o CCP.
- Plano de Control Local o LCP.



PLANO DE CONTROL CENTRAL (CCP)

Este componente se ejecuta en el NSX Management Cluster y es parte del rol NSX Controller.

El CCP se encargará de calcular el estado de los componentes del plano de datos basado en la configuración provista por el plano de management. Esta configuración luego será distribuida a todos los elementos del plano de datos usando el LCP.

Aquí en el CCP mantendremos información requerida por switches y routers logicos:

- Tabla de TEP
- Tabla de MAC Address
- Tabla de ARP
- Tabla de Routing

PLANO DE CONTROL LOCAL (LCP)

Este componente se ejecuta en cada uno de los nodos de transporte (ESXi o KVM), así como en nodos NSX Edge.

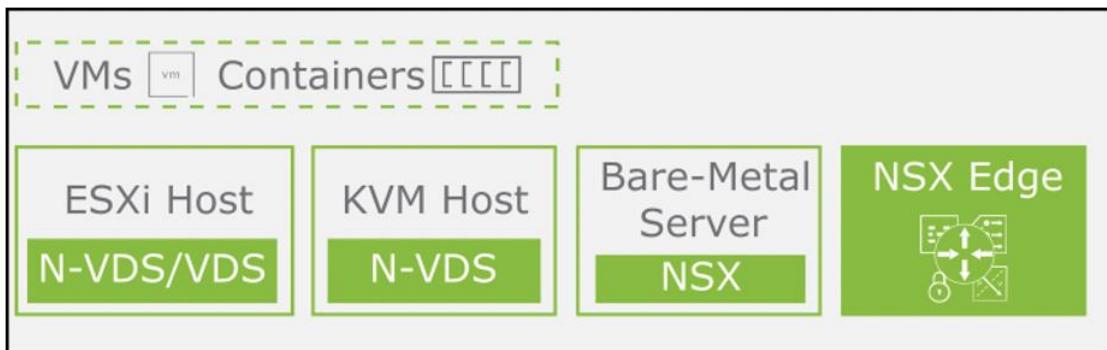
El LCP actualiza el plano de datos, incluyendo la configuración de Swithes logicos (segmentos) y Routers distribuidos, según la información enviada por el CCP.

A la vez, el LCP reporta cualquier cambio en el plano de datos al CCP, el cual luego se encargará de replicar dicho cambio a todos los nodos de transporte

PLANO DE DATOS

El Plano de Datos es responsable del reenvío (forwarding) de paquetes basado en la configuración distribuida por el plano de control (CCP), utilizando para esto las tablas y reglas provistas por el CCP, como por ejemplo las **tablas de MAC Address** y las **tablas de ARP para cada Segmento**.

El Plano de Datos también es responsable de enviar información sobre la topología actual al plano de control a través del LCP, además de mantener estadísticas de envío de paquetes.



En el plano de datos tenemos múltiples componentes también llamados Nodos de Transporte:

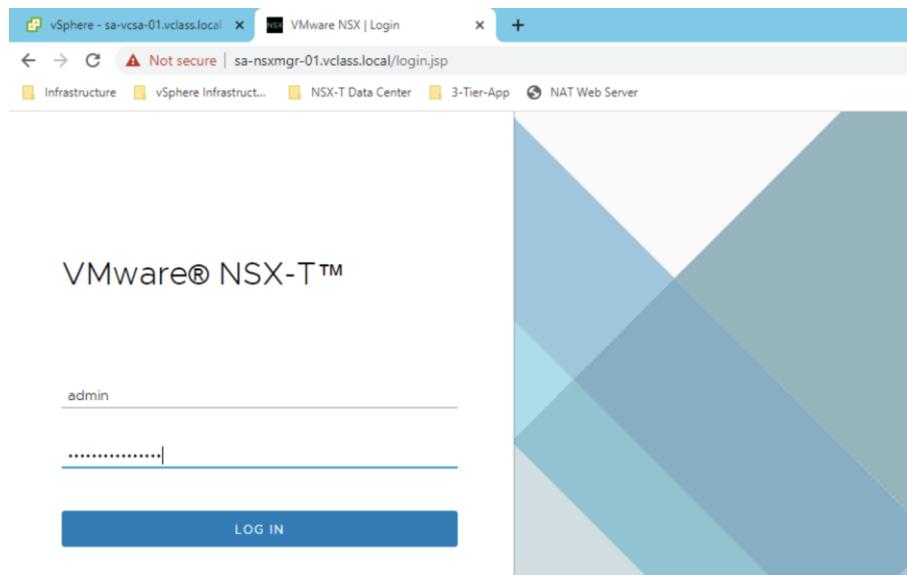
- Nodos de transporte hipervisor (ESXi o KVM): Provee el reenvío de paquetes para máquinas virtuales.
- Nodos de transporte Bare Metal (Linux o Windows): Provee el reenvío de paquetes para aplicaciones ejecutándose sobre estos nodos, así como también contenedores.
- NSX Edge Cluster: Contiene nodos de transporte Edge (Virtual o Físico), para proveer servicios de Gateway, así como también servicios Stateful (NAT, Load Balancer, entre otros)

DESPLIEGUE DE NSX MANAGEMENT CLÚSTER

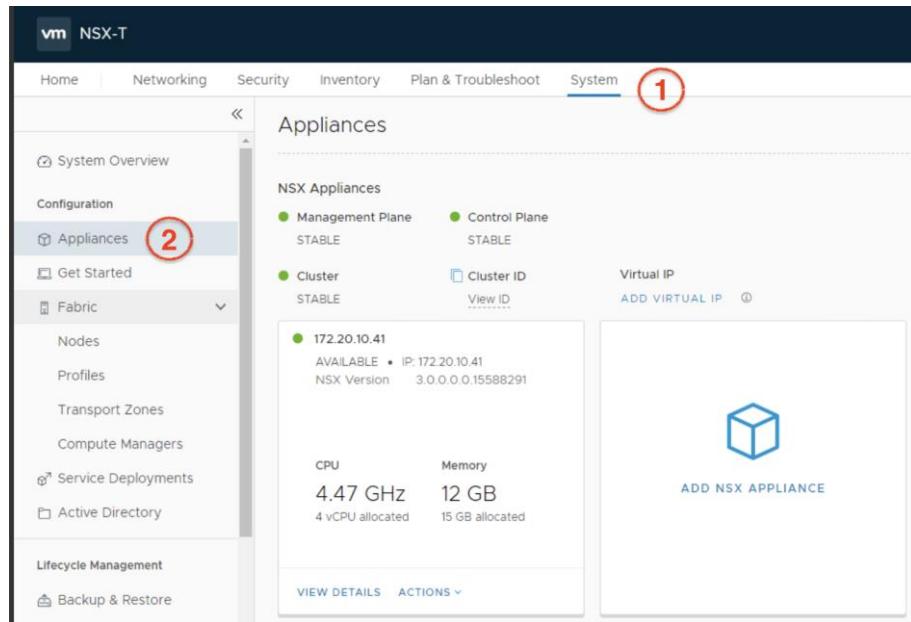
El despliegue de NSX Management Clúster comienza con el despliegue de un nodo NSX Manager importando un virtual appliance en formato OVF que descargamos desde el portal [My VMware](#).

CONECTARNOS A PRIMER NODO NSX MANAGER

Una vez desplegado el primer nodo NSX Manager, nos conectamos a este utilizando un navegador soportado, y nos autenticamos con el usuario “Admin”, cuya password fue configurada durante el despliegue del primer nodo.



1. Una vez conectados nos dirigimos a **System > Appliances**
2. Verificamos que el appliance se encuentre operando adecuadamente y de manera estable.

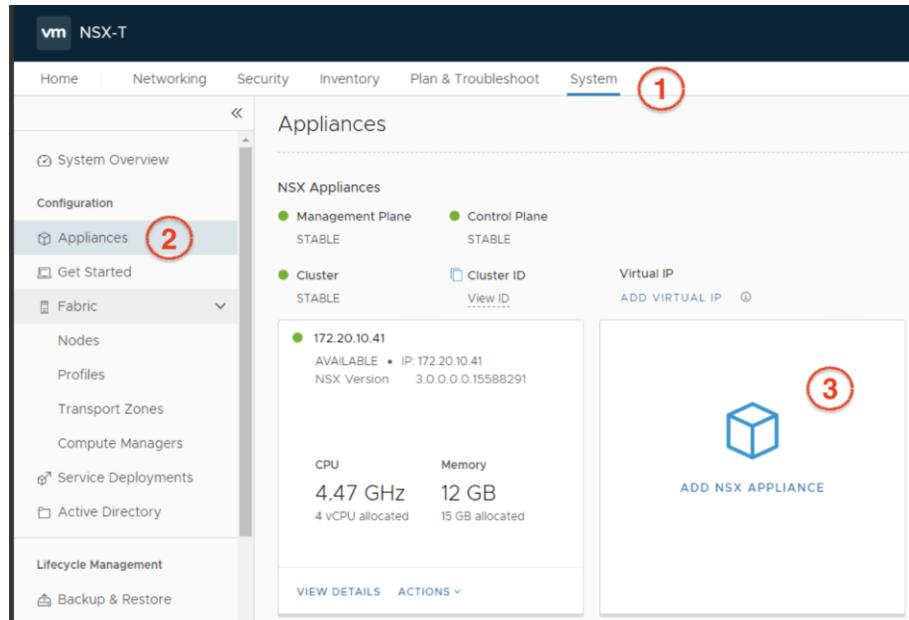


DESPLIEGUE NODOS ADICIONALES

A continuación, podremos proceder con el despliegue de los nodos adicionales de NSX Manager, simplemente haciendo click en la opción “Add NSX Appliance”.

Nota: En un ambiente en producción, el NSX Management Clúster debe ser desplegado como un cluster de tres nodos.

DESPLIEGUE DEL SEGUNDO NODO NSX MANAGER



Al iniciar el asistente, debemos ingresar los siguientes datos:

1. FQDN del nuevo nodo NSX Manager
2. IP y Mascara de subred para la red de administración
3. Gateway
4. Servidores DNS
5. Servidores NTP
6. Seleccionar el tamaño del NSX Manager appliance. Todos los nodos del clúster deben tener el mismo tamaño.

Add Appliance

Appliance Information

Host Name or FQDN* sa-nsxmgr-02.vclass.local ⓘ
Enter IP or Fully Qualified Domain Name (FQDN) e.g., subdomain.example.com

Management IP/Netmask* 172.20.10.42/24

Management Gateway* 172.20.10.10

DNS Servers* 172.20.10.10 ⓘ
Enter DNS Server IP

NTP Servers 172.20.10.10 ⓘ
Enter NTP Server IP

Node Size

Learn more about appliance selection

CANCEL NEXT

Add Appliance

Appliance Information

DNS Servers* 172.20.10.10 ⓘ
Enter DNS Server IP

NTP Servers 172.20.10.10 ⓘ
Enter NTP Server IP

Node Size

Learn more about appliance selection

Small
4 vCPU
16 GB RAM
200 GB storage

Medium
6 vCPU
24 GB RAM
200 GB storage

Large
12 vCPU
48 GB RAM
200 GB storage

CANCEL NEXT

A continuación, debemos ingresar como mínimo los siguientes datos:

1. Instancia de vCenter Server donde desplegaremos el appliance
2. Clúster donde desplegaremos el appliance
3. Datastore
4. Virtual Port Group que permita al appliance conectarse con la red de administración.

Add Appliance

Configuration

Compute Manager * sa-vcsa-01

Compute Cluster * SA-Management-Edge (do...)

Resource Pool Select resource pool

Host Select host

Datastore * SA-Shared-02-Remote (dat...)

Virtual Disk Format Thin Provision

Network * Pg-SA-Management

CANCEL BACK NEXT

Como último paso, podemos completar los siguientes datos:

1. Habilitar SSH
2. Habilitar root Access
3. Proveer credenciales de root
4. Proveer credenciales para usuario Admin CLI (puede utilizarse la misma password de root)
5. Proveer credenciales para usuario Audit CLI (puede utilizarse la misma password de root)
6. Hacer click en Install Appliance

Add Appliance

Credentials

Enable SSH Yes

Enable Root Access Yes

System Root Credentials

System Username root

Root Password

Confirm Root Password

Admin CLI Credentials

• 12 characters min
• 1 lower case
• 1 upper case
• 1 number
• At least 5 different characters
• No dictionary words
• No palindromes

CANCEL BACK INSTALL APPLIANCE

Add Appliance

1 Appliance Information

2 Configuration

3 Credentials

Credentials

System Username: root

Root Password: [Reset](#)

Confirm Root Password: [Reset](#)

Admin CLI Credentials

CLI Username: admin

CLI password: Same as root password

Audit CLI Credentials

Audit CLI Username: audit

Audit CLI password: Same as root password

[CANCEL](#) [BACK](#) [INSTALL APPLIANCE](#)

DESPLIEGUE DEL TERCER NODO NSX MANAGER

Luego, simplemente repetimos el mismo proceso para desplegar un tercer nodo NSX Manager, con lo que veremos el clúster completamente formado como vemos a continuación:

Appliances					
NSX Appliances					
Management Plane	Control Plane				
STABLE	STABLE				
Cluster	Cluster ID	Virtual IP			
STABLE	View ID	ADD VIRTUAL IP ?			
172.20.10.42	172.20.10.43	172.20.10.41			
AVAILABLE • IP: 172.20.10.42	AVAILABLE • IP: 172.20.10.43	AVAILABLE • IP: 172.20.10.41			
NSX Version 3.0.0.0.0.15588291	NSX Version 3.0.0.0.0.15588291	NSX Version 3.0.0.0.0.15588291			
CPU	Memory	CPU			
7.61 GHz	12 GB	11.13 GHz	12 GB		
4 vCPU allocated	15 GB allocated	4 vCPU allocated	15 GB allocated		
VIEW DETAILS	ACTIONS ▾	VIEW DETAILS	ACTIONS ▾	VIEW DETAILS	ACTIONS ▾

CONFIGURACION DE IP VIRTUAL

Como mencionamos anteriormente, es posible definir una IP Virtual para poder acceder al NSX Management Clúster utilizando una única IP que representa al cluster, en vez de conectarnos a nodos individuales de NSX Manager.

El proceso es muy sencillo, nos dirigimos a **System > Appliances**, donde simplemente debemos hacer click en “**Add Virtual IP**” como vemos en la siguiente imagen.

The screenshot shows the NSX Appliances interface. At the top, there are tabs for 'Cluster' (selected), 'UNAVAILABLE', 'Cluster ID', 'View ID', and a 'Virtual IP' section with a 'SET VIRTUAL IP' button. Below this, two host cards are displayed:

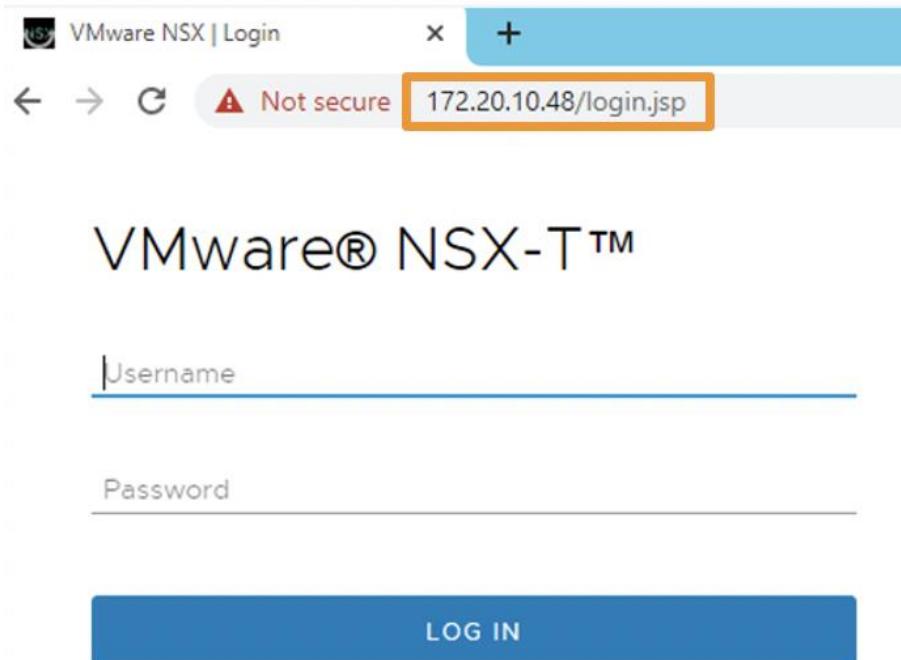
- Host 1:** IP 172.20.10.41, Available, Version: 3.0.0.0.15946739. It shows System Load (18.08), Memory (14 GB, 15 GB allocated), and 4 vCPU allocated. Buttons for 'VIEW DETAILS' and 'ACTIONS ▾' are at the bottom.
- Host 2:** IP 172.20.10.42, Available, Version: 3.0.0.0.15946739. It shows System Load (22.69), Memory (13 GB, 15 GB allocated), and 4 vCPU allocated. Buttons for 'VIEW DETAILS' and 'ACTIONS ▾' are at the bottom.

Set Virtual IP

NSX-T Managers Cluster offer a built-in VIP for high-availability, but the usage of an external load balancer offers the following benefits: 1) Load spread across all NSX-T Managers; 2) NSX-T Managers can be in different subnets and 3) Faster failover.

A modal dialog box titled 'Set Virtual IP'. It contains a single input field labeled 'Virtual IP Address*' with the value '172.20.10.48'. Below the input field are two buttons: 'CANCEL' and 'SAVE'.

Una vez configurada la IP Virtual, podremos conectarnos al NSX Management Clúster simplemente utilizando esta IP en un navegador soportado.



INTEGRACIÓN CON VCENTER SERVER

Como último paso en la configuración inicial del NSX Management Clúster, completaremos la integración entre NSX Manager y vCenter Server, la cual es totalmente opcional y solamente requerida si es que los nodos de transporte que vayamos a utilizar sean hosts ESXi gestionados por vCenter Server.

Nos dirigimos a **System > Fabric > Compute Managers** y hacemos click en “Add”. A continuación, ingresaremos los siguientes datos:

1. Nombre o alias con el que quedará registrado en NSX Manager
2. Tipo: vCenter
3. FQDN del vCenter Server
4. Puerto de conexión con vCenter Server, usualmente puerto 443.
5. Username: Usuario con privilegios de administración sobre vCenter Server
6. Password
7. Click en “Add”

New Compute Manager

Name*	sa-vcsa-01.vclass.local
Description	vCenter Server Appliance
Type*	vCenter
FQDN or IP Address*	sa-vcsa-01.vclass.local
HTTPS Port of Reverse Proxy* ⓘ	443
Username*	administrator@vsphere.local
Password*	*****
SHA-256 Thumbprint	
Enable Trust ⓘ	<input checked="" type="checkbox"/> Yes Supported for vCenter Server 7.0 or later
<input type="button" value="CANCEL"/> <input type="button" value="ADD"/>	

Una vez completado el registro, lo veremos de la siguiente forma:

Compute Manager	ID	FQDN or IP Address	Type	Registration Status	Version
sa-vcsa-01.vclass.local	b991_34ff	172.20.10.04	vCenter	Registered	7.0.0

PREPARANDO EL PLANO DE DATOS

En esta sección describiremos el proceso de preparación del plano de datos incluyendo:

1. Configuración de Zonas de Transporte

2. Configuración de IP Pools
3. Configuracion de Node Profiles
4. Switches N-VDS
5. Preparación de los Nodos de Transporte

ZONAS DE TRANSPORTE

Uno de los primeros pasos en la configuración del plano de datos es la creación de las Zonas de Transporte.

Una zona de transporte define un conjunto de nodos de transporte (ESXi, KVM y nodos NSX Edge), que se pueden comunicar entre si a través de una red de transporte utilizando para ello una o más interfaces llamadas TEP.

Los TEPs son interfaces en cada nodo de transporte que permite la comunicación entre ellos, formando una red de transporte, donde el tráfico es encapsulado y desencapsulado utilizando Geneve como protocolo.

En vSphere, los TEPs son creados utilizando puertos VMkernel, lo que permite la comunicación con otros nodos de transporte. En cierta forma, esta comunicación es similar a la que podríamos tener entre los hosts ESXi utilizando una red de vMotion, donde se utilizan puertos VMkernel en el host de origen y destino para realizar una migración con vMotion. En el caso de NSX-T, los puertos VMkernel son utilizados como TEPs, para formar un túnel entre múltiples hosts ESXi, KVM y nodos NSX Edge, donde el tráfico es encapsulado utilizando Geneve.

Existen dos tipos de Zonas de Transporte:

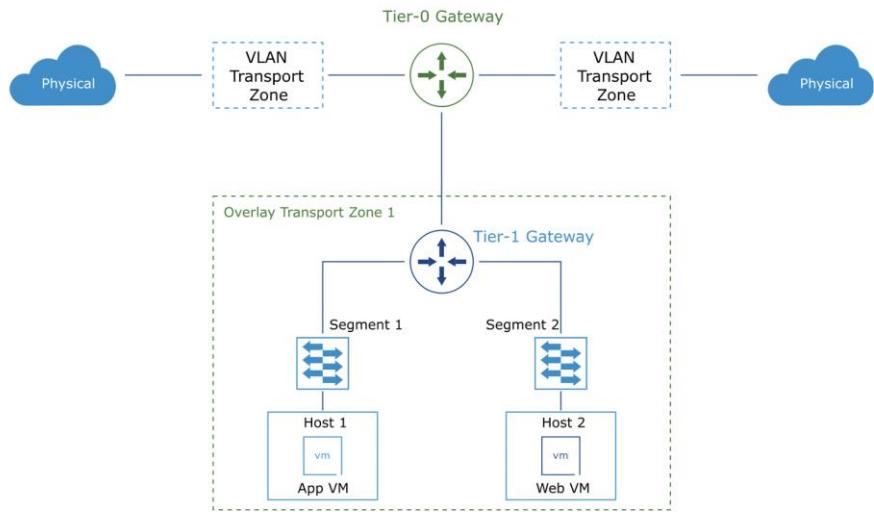
1. **Overlay**: Usado como túnel interno entre los nodos de transporte. Permite enviar el tráfico encapsulado utilizando Geneve.
2. **VLAN**: Utilizado por los uplinks de los nodos NSX Edge para establecer conectividad norte-sur (conectividad entre redes físicas y las redes lógicas en NSX-T).

Las zonas de transporte definirán luego en qué nodos de transporte (ESXi, KVM, nodos Edge) se encontrarán disponibles ciertos Segmentos lógicos, así como los Gateways Tier-1 y Tier-0.

Puntos importantes a tener en cuenta:

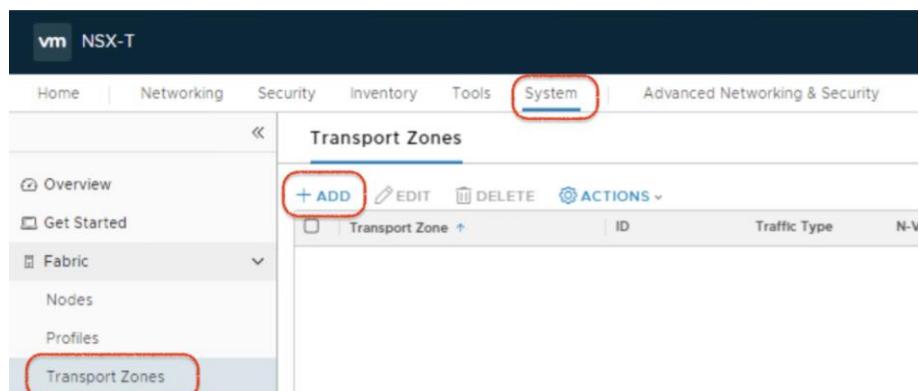
- Una única zona de transporte puede incluir múltiples tipos de nodos de transporte. Por ejemplo, en una misma zona de transporte se pueden incluir hosts ESXi y KVM.
- La zona de transporte define el tipo de tráfico: VLAN o Overlay.

- Se pueden crear múltiples Zonas de Transporte dependiendo del diseño que se desee implementar.
- Una Zona de Transporte no representa un límite de seguridad. VMs que se encuentren en distintas Zonas de Transporte podrían comunicarse entre sí si existe una ruta de conexión entre ellos.



CREAR UNA ZONA DE TRANSPORTE

La creación de una Zona de Transporte es un proceso bastante simple. Nos dirigimos a **System > Fabric > Transport Zones** y hacemos click en Add.



A continuación, añadimos dos zonas de transporte:

- Zona Overlay: Que será utilizada por los nodos de transporte (ESXi, KVM y nodos NSX Edge).
- Zona VLAN: Que será utilizada por los uplinks de los NSX Edge para la comunicación norte-sur.

Ingresamos los siguientes datos:

- Nombre de la Zona de Transporte
- Nombre del switch N-VDS asociados a la Zona de Transporte
- Tipo de tráfico: Overlay o VLAN

New Transport Zone

Name*	PROD-Overlay-TZ
Description	<input type="text"/>
N-VDS Name*	PROD-Overlay-NVDS
Host Membership Criteria	<input checked="" type="radio"/> Standard (For all hosts) <input type="radio"/> Enhanced Datapath (For ESXi hosts with version 6.7 or above)
Traffic Type	<input checked="" type="radio"/> Overlay <input type="radio"/> VLAN
Uplink Teaming Policy Names	<input type="text"/>
<input type="button" value="CANCEL"/> <input type="button" value="ADD"/>	

Finalmente vemos como quedarán creadas ambas Zonas de Transporte:

Transport Zones

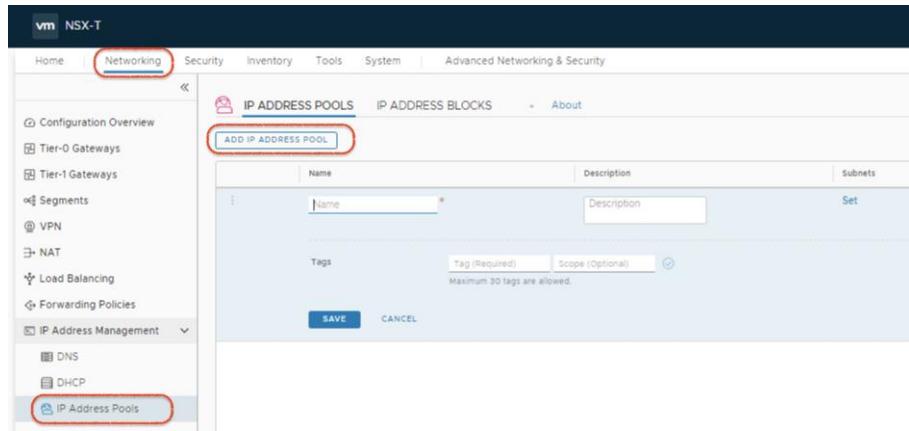
	Transport Zone ↑	ID	Traffic Type	N-VDS Name
<input type="checkbox"/>	PROD-Overlay-TZ	792f...ccd6	Overlay	PROD-Overlay-NVDS
<input type="checkbox"/>	PROD-VLAN-TZ	ce7c...edf3	VLAN	PROD-VLAN-NVDS

IP ADDRESS POOL

Los IP Pools son grupos de direcciones IP's que utilizaremos posteriormente durante el proceso de preparación de los hosts ESXi y los nodos NSX Edge, para proveer de direccionamiento IP para los Tunnel Endpoints (TEP).

El proceso de creación es muy sencillo.

1. Nos dirigimos a **Networking > IP Management > IP Address Pools**
2. Hacemos click en “**Add IP Address Pool**”



A continuación, ingresamos los datos para crear el Pool:

1. Nombre: **VTEP-IP-Pool**
2. Descripción (opcional)
3. Hacemos click en Set para configurar el pool de direcciones IP
4. Hacemos click en **Add Subnet > IP Ranges**
 - a. Ingresamos un rango de direcciones IP
 - b. CIDR para representar la máscara de subred
 - c. Default Gateway
 - d. Click en Add
 - e. Click en Apply

Set Subnets

The screenshot shows the 'Set Subnets' interface for an IP Address Pool. At the top, there is a header with tabs for 'IP Address Pool' and '#IP Address Pool Subnets (1)'. Below the header, there is a search bar and a 'COLLAPSE ALL' button. The main area is divided into two sections: 'Source' and 'IP Ranges / Block'. In the 'Source' section, there is a dropdown menu labeled 'ADD SUBNET' with a red box around it. In the 'IP Ranges / Block' section, there is a table with one row. The first column 'Source' has a dropdown menu with 'IP Ranges' selected. The second column 'IP Ranges' contains the range '172.20.11.151-172.20.11.170' with a red box around it. The third column 'CIDR' contains '172.20.11.0/24' with a red box around it. The fourth column 'Gateway IP' contains '172.20.11.10' with a red box around it. At the bottom of the table are 'ADD' and 'CANCEL' buttons. Below the table, there are 'CANCEL' and 'APPLY' buttons.

- Finalmente hacemos click en Save para guardar los cambios y finalizar la creación del IP Pool.

The screenshot shows the 'IP ADDRESS POOLS' interface. At the top, there are tabs for 'IP ADDRESS POOLS' (selected), 'IP ADDRESS BLOCKS', and 'About'. Below the tabs, there is a 'ADD IP ADDRESS POOL' button. The main area is a table with columns for 'Name', 'Description', and 'Subnets'. A new row is being added, with 'Name' set to 'VTEP-IP-Pool' (highlighted with a red box), 'Description' empty, and 'Subnets' set to '1' (highlighted with a red box). Below the table, there are 'Tags' fields with 'Tag (Required)' and 'Scope (Optional)', and a note stating 'Maximum 30 tags are allowed.' At the bottom of the table are 'SAVE' and 'CANCEL' buttons. The 'SAVE' button is highlighted with a red box.

NSX VIRTUAL SWITCH (N-VDS)

Todo nodo de transporte necesita un tipo de Virtual Switch para proveer el servicio de reenvío de paquetes en un nodo de transporte. En NSX-T 3.0 tenemos dos alternativas:

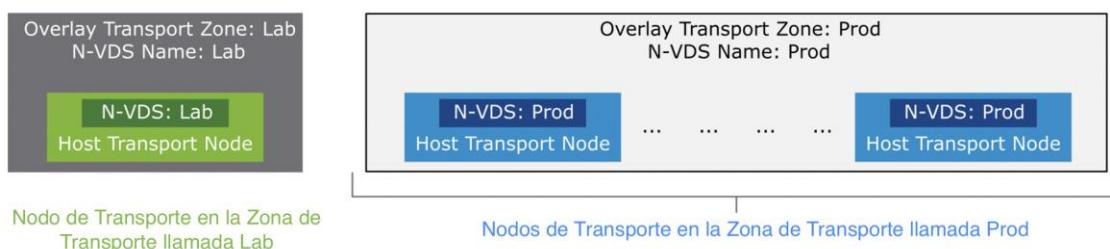
- N-VDS:** Switch lógico creado y gestionado por NSX Manager, compatible con múltiples hypervisores (ESXi y KVM), así como también con los NSX Edge.
- VDS:** Virtual Distributed Switch creado y gestionado por vSphere. Compatible solo con hypervisores ESXi 7.0.

N-VDS nos permite crear un switch lógico agnóstico del hypervisor, sobre el cual luego crearemos múltiples Segmentos capa 2 para permitir la conectividad de las Máquinas Virtuales. Los switches **N-VDS contienen los TEPs** en los nodos de transporte, y permiten que estos nodos se conecten a la red de transporte definida en la Zona de Transporte.

En cierta forma un N-VDS es similar a un Switch Distribuido en vSphere, donde creamos un switch N-VDS de manera centralizada, el cual luego estará disponible para múltiples nodos de transporte (ESXi, KVM y nodos NSX Edge). El switch N-VDS tiene las siguientes características:

- N-VDS puede coexistir en un mismo host ESXi con Standard Switches y Distributed Switches (VDS).
- N-VDS es totalmente independiente de vCenter Server
- Soporte de múltiples políticas de NIC-Teaming incluyendo LACP.
- Cada zona de transporte tiene un switch N-VDS asignado
- Los nombres de los Switches N-VDS y de las Zonas de Transporte debieran ser asignados de manera consistente, para facilitar la identificación de la Zona de Transporte a la que pertenece cada N-VDS.

En la siguiente imagen podemos ver dos Zonas de Transporte, Prod y Lab, cada una con un Switch N-VDS que se despliega a través de todos los nodos de transporte que pertenecen a la misma Zona de Transporte.

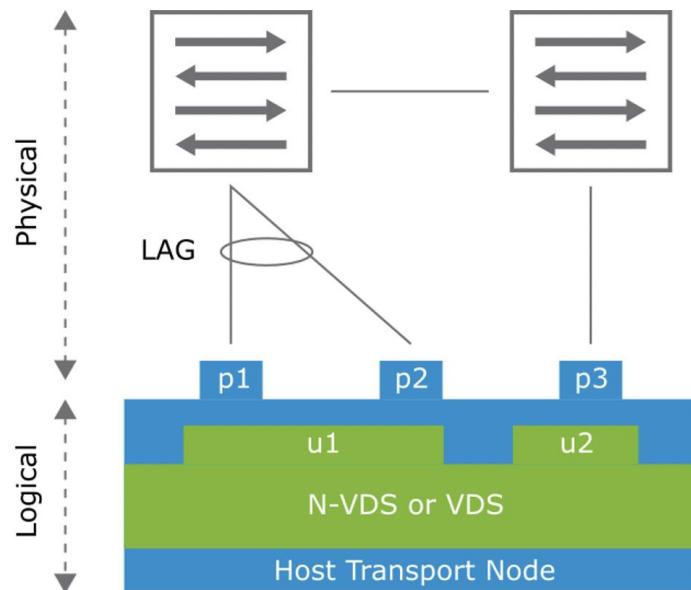


Nota: La creación de los switches N-VDS lo veremos más adelante cuando hablamos de la preparación de los nodos de transporte (ESXi o KVM).

N-VDS UPLINKS

Como todo Switch Virtual, un switch N-VDS requiere uno o más uplinks que le permitan conectarse con la red física. Cada Uplink puede ir conectado a una NIC física o a un LAG del nodo de transporte (ESXi o KVM), como lo vemos en la siguiente imagen.

La recomendación siempre es utilizar al menos dos Uplinks por cada switch N-VDS para proveer de redundancia y escalabilidad.



Como podemos ver en la imagen, este switch N-VDS tiene dos uplinks, uno de ellos conectados a un LAG con dos interfaces físicas, y el otro conectado a una única interfaz física del host.

N-VDS UPLINK PROFILES

La configuración de los Uplinks para un switch N-VDS está definida por un Uplink Profile. Un Uplink Profile define como un switch N-VDS en un nodo de transporte se conectará a las NIC físicas del host, incluyendo los siguientes parámetros:

- Teaming policy
- Active and standby uplinks
- Transport VLAN ID
- Maximum transmission unit (MTU)

Existen una serie de Uplink Profiles por defecto que pueden utilizar durante la creación de un switch N-VDS, con distintas configuraciones de Uplinks y NIC Teaming:

- Profiles con múltiples uplinks
- Profiles con un único uplink
- Profiles con uso de LAGs

Estos profiles los podemos ver en **System > Fabric > Profiles > Uplink Profiles**:

Uplink Profile	ID	Teaming Policy	Active Uplinks	Standby Uplinks
nsx-default-loadbalance-uplink-ho...	fb38...2e0d	Load Balance Source	uplink-1,uplink-2,uplink-...	
nsx-default-uplink-hostswitch-profil...	0a26...dc9f	Failover Order	uplink-1	uplink-2
nsx-edge-tag-uplink-profile	c352...5f3f	Failover Order	lag	
nsx-edge-multiple-vtep-uplink-pr...	ce82...1107	Load Balance Source	uplink-1,uplink-2	
nsx-edge-single-nic-uplink-profile	cf32...e3bc	Failover Order	uplink-1	

Si lo deseamos, podemos crear un nuevo Uplink Profile ingresando los siguientes datos:

- Nombre del Profile
- Teaming Policy
 - **Failover Order**: Al menos 1 uplink activo y opcionalmente 1 uplink standby
 - **Load Balanced Source**: Balanceo según la Puerta del switch a la cual la máquina virtual de origen está conectada.
 - **Load Balanced Source Mac**: Balanceo según la MAC Address de la máquina virtual de origen.
- Uplinks activos
- Opcional: Uplinks Standby
- VLAN
- MTU: 1600 o superior para redes Overlay

New Uplink Profile

Name*	Labs-Uplink-Profile-1													
Description														
LAGs														
<input type="button" value="+ ADD"/> <input type="button" value="DELETE"/> <table border="1"> <thead> <tr> <th>Name*</th> <th>LACP Mode</th> <th>LACP Load Balancing*</th> <th>Uplinks*</th> <th>LACP Time Out</th> </tr> </thead> <tbody> <tr> <td colspan="5">No LAGs found</td> </tr> </tbody> </table>					Name*	LACP Mode	LACP Load Balancing*	Uplinks*	LACP Time Out	No LAGs found				
Name*	LACP Mode	LACP Load Balancing*	Uplinks*	LACP Time Out										
No LAGs found														
Teamings														
<input type="button" value="+ ADD"/> <input type="button" value="CLONE"/> <input type="button" value="DELETE"/> <table border="1"> <thead> <tr> <th>Name*</th> <th>Teaming Policy*</th> <th>Active Uplinks*</th> <th>Standby Uplinks</th> </tr> </thead> <tbody> <tr> <td><input checked="" type="checkbox"/> [Default Teaming]</td> <td>Failover Order</td> <td colspan="2">Uplink1, Uplink2</td> </tr> </tbody> </table>					Name*	Teaming Policy*	Active Uplinks*	Standby Uplinks	<input checked="" type="checkbox"/> [Default Teaming]	Failover Order	Uplink1, Uplink2			
Name*	Teaming Policy*	Active Uplinks*	Standby Uplinks											
<input checked="" type="checkbox"/> [Default Teaming]	Failover Order	Uplink1, Uplink2												
Active uplinks and Standby uplinks are user defined labels. These labels will be used to associate with the Physical NICs while adding Transport Nodes.														
Transport VLAN	0													
MTU	1600													

Del mismo modo, podemos configurar un Uplink Profile usando LAGs para una configuración con LACP

- Nombre de Profile
- Añadimos un nuevo LAG
 - Nombre
 - Modo LACP
 - Política de balanceo de carga LACP
 - Numero de uplinks en el LAG
- En NIC Teaming
 - Definimos una política de NIC Teaming
 - Como uplinks Activos (opcionalmente también en Standby) ingresamos el nombre del LAG creado previamente.

New Uplink Profile

Name*	Test Profile													
Description														
LAGs														
<input type="button" value="+ ADD"/> <input type="button" value="DELETE"/> <table border="1"> <thead> <tr> <th>Name*</th> <th>LACP Mode</th> <th>LACP Load Balancing*</th> <th>Uplinks*</th> <th>LACP Time Out</th> </tr> </thead> <tbody> <tr> <td><input checked="" type="checkbox"/> lag1</td> <td>Active</td> <td>Source and destination IP ad...</td> <td>2</td> <td>Slow</td> </tr> </tbody> </table>					Name*	LACP Mode	LACP Load Balancing*	Uplinks*	LACP Time Out	<input checked="" type="checkbox"/> lag1	Active	Source and destination IP ad...	2	Slow
Name*	LACP Mode	LACP Load Balancing*	Uplinks*	LACP Time Out										
<input checked="" type="checkbox"/> lag1	Active	Source and destination IP ad...	2	Slow										

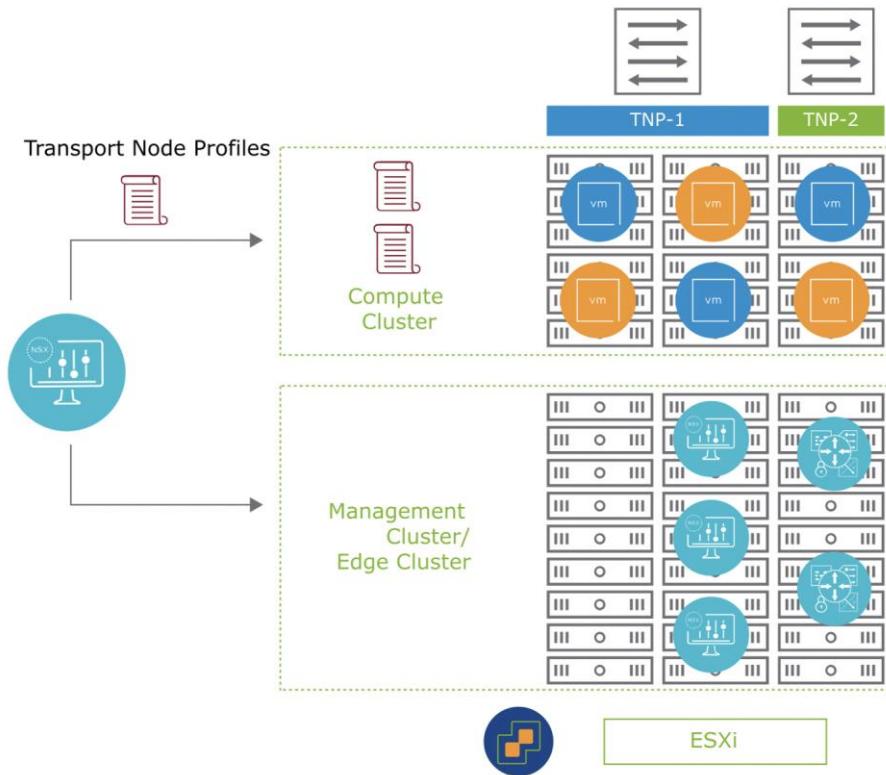
A screenshot of the vSphere Web Client interface showing the 'Teamings' configuration. The main area displays a table with a single row for '[Default Teaming]'. The columns are labeled 'Name*', 'Teaming Policy*', 'Active Uplinks*', and 'Standby Uplinks'. The 'Name*' column contains '[Default Teaming]'. The 'Teaming Policy*' column shows 'Failover Order' with 'lag1' selected. The 'Active Uplinks*' and 'Standby Uplinks' columns are empty. Below the table, a note states: 'Active uplinks and Standby uplinks are user defined labels. These labels will be used to associate with the Physical NICs while adding Transport Nodes.' At the bottom of the form, there are fields for 'Transport VLAN' (set to 0) and 'MTU' (with a dropdown menu).

TRANSPORT NODE PROFILES

Cuando contamos con una infraestructura vSphere, la manera más sencilla de preparar los Host ESXi para ser partes de la infraestructura NSX-T, es usando un Transport Node Profile.

Un Transport Node Profile es como un template que nos permite posteriormente aplicar una serie de configuraciones a host ESXi. Transport Node Profile contiene toda la configuración requerida para transformar un host ESXi en un nodo de transporte.

Una vez creado un Transport Node Profile, podemos luego aplicarlo a múltiples clústeres vSphere, de manera de asegurar que la configuración de los host ESXi se realiza de manera rápida y consistente.

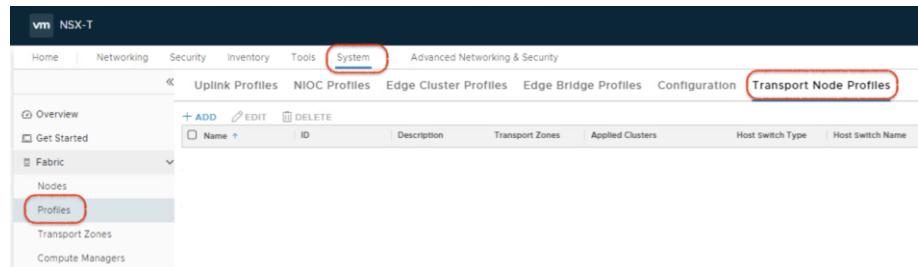


Cada Transport Node Profile define:

- Zonas de Transporte a la que pertenecerán los hosts ESXi
- Configuración de switch N-VDS
- Uplink Profile aplicable al switch N-VDS
- Método de asignación de IP para los TEPs: IP Pool o DHCP
- Asignación de NIC físicas a cada Uplink

CREAR UN TRANSPORT NODE PROFILE

Para crear un Transport Node Profile, nos dirigimos a **System > Fabric > Profiles > Transport Node Profiles** y hacemos click en **Add**.



A continuación, ingresamos los siguientes datos:

- Nombre del Transport Node Profile
- Zona de Transporte. Usualmente los host ESXi en un clúster de Producción solo necesitan una Zona de Transporte Overlay.

Add Transport Node Profile

[General *](#) [N-VDS *](#)

Name* (highlighted with a red box)

Description

Transport Zones

<input type="checkbox"/> Available (4)	<input type="checkbox"/> Selected (1)
<input type="checkbox"/> PROD-Overlay-TZ (Overlay)	<input checked="" type="checkbox"/> PROD-Overlay-TZ (Overlay) (highlighted with a red box)
<input type="checkbox"/> PROD-VLAN-TZ (VLAN)	
<input type="checkbox"/> TZ-Overlay (Overlay)	
<input type="checkbox"/> TZ-VLAN (VLAN)	

< BACK [NEXT >](#) 1 - 4 of 4 records

[Create New Transport Zone](#)

A continuación, hacemos click en N-VDS y configuraremos el switch N-VDS para los host ESXi de la zona de transporte. En este caso configuraremos un único switch N-VDS en una Zona de Transporte Overlay.

- **Nombre del switch N-VDS.** El mismo que definimos previamente al configurar la Zona de Transporte.

- **NIOC Profile.** Permite definir Reservas, Limites y Shares para distintos tipos de tráfico que pasarán a través de este switch N-VDS. Se puede crear un NIOC Profile o utilizar un NIOC Profile por defecto.
- **Uplink Profile.** Permite definir el número y tipo de Uplinks que utilizará el switch N-VDS. En este caso utilizo el perfil creado previamente, el cual utiliza un LAG con dos interfaces.
- **LLDP Profile.** Permite definir si se habilitará el envío de paquetes vía LLDP, por defecto deshabilitado.
- **IP Assignment:** Seleccionar IP Pool o DHCP. Este es el método con el que los TEP en cada host ESXi obtendrán sus direcciones IP.
- **IP Pool:** Especificamos el IP Pool que proveerá las direcciones IP para los TEP.

Add Transport Node Profile

General * N-VDS *

N-VDS Creation* NSX Created Preconfigured

+ ADD N-VDS

New Node Switch

N-VDS Name* PROD-Overlay-NVDs

Associated Transport Zones PROD-Overlay-TZ

NIOC Profile* nsx-default-nioc-hostswitch-profile

OR Create New NIOC Profile

Uplink Profile* Test Profile

OR Create New Uplink Profile

LLDP Profile* LLDP [Send Packet Disabled]

IP Assignment* Use IP Pool

IP Pool* VTEP-IP-Pool

OR Create and Use a new IP Pool

Un poco más abajo en la misma ventana, completamos la configuración indicando los uplinks que utilizaremos. En este caso, y según el Uplink Profile que hemos elegido, debemos especificar las dos interfaces físicas requeridas para el LAG. Luego simplemente hacemos click en **Add**.

Add Transport Node Profile

Associated Transport Zones: PROD-Overlay-TZ

NIOC Profile*: nsx-default-nioc-hostswitch-profile [OR Create New NIOC Profile](#)

Uplink Profile*: Test Profile [OR Create New Uplink Profile](#)

LLDP Profile*: LLDP [Send Packet Disabled]

IP Assignment*: Use IP Pool

IP Pool*: VTEP-IP-Pool [OR Create and Use a new IP Pool](#)

Physical NICs:

vmnic1	lag1-0	
vmnic4	lag1-1	

PNIC only Migration: No
Enable this option if no vms exist on PNIC selected for migration

Network Mappings for Install: [Add Mapping](#)

Network Mappings for Uninstall: [Add Mapping](#)

[CANCEL](#) [ADD](#)

PREPARAR LOS NODOS DE TRANSPORTE

Uno de los últimos pasos en el proceso de preparación del Plano de Datos, es la configuración o preparación de los nodos de transporte (ESXi o KVM). Este proceso implica:

- Instalación de los modulos de kernel necesarios para el uso de Switches lógicos, Routers distribuidos, y Firewall Distribuido.
- Creación y configuración del Switch N-VDS en cada nodo de transporte.
- Configuración de los TEPs

Todo este proceso se puede realizar en un solo paso cuando deseamos preparar un host ESXi gestionado por vCenter Server. Como vemos en la siguiente imagen, tenemos 5 hosts ESXi divididos en dos Clústeres:

The screenshot shows the vSphere Web Client interface. The top navigation bar has tabs: Home, Networking, Security, Inventory, Plan & Troubleshoot, and System (which is highlighted with a red box). Below the navigation is a secondary header with tabs: Host Transport Nodes (highlighted), Edge Transport Nodes, Edge Clusters, and ESXi Bridges. A dropdown menu 'Managed by' shows 'sa-vcsa-01.vclass.local'. On the left, a sidebar has sections: System Overview, Configuration, Appliances, Get Started, Fabric (with 'Nodes' selected and highlighted with a red box), Profiles, Transport Zones, Compute Managers, and Settings. The main content area displays a table titled 'Host Transport Nodes' with the following columns: Node, ID, IP Addresses, OS Type, and NSX Configuration. The table lists seven nodes, with the second node ('sa-esxi-01.vclass.local') having its row highlighted with a blue background. The node details are: Node: sa-esxi-01.vclass.local, ID: b991..1011, IP Addresses: 172.20.10.51, 172..., OS Type: ESXi 7.0.0, NSX Configuration: Not Configured. The node 'sa-esxi-01.vclass.local' has a checkmark next to it. Other nodes listed are: sa-esxi-03.vclass.local (b991..1020), sa-esxi-02.vclass.local (b991..1023), sa-Compute-01 (2) (selected, b991..1024), sa-esxi-04.vclass.local (c365..67fb), and sa-esxi-05.vclass.local (0906..8750).

Para preparar un vSphere Clúster, todo lo que hay que hacer es seleccionar el Clúster en **System > Fabric > Nodes > Host Transport Nodes**, como se puede ver en la imagen anterior.

1. Luego simplemente hacer click en **Configure NSX**.
2. Seleccionamos el Transport Node Profile creado en la sección anterior
3. Hacemos click en Save.

Configure NSX

X

NSX will be installed on the selected cluster with deployment configuration defined in Transport Node Profile

Select Deployment Profile* TNP_EsXi

Create New Transport Node Profile

CANCEL

SAVE

Luego de aplicar el Transport Node Profile al cluster, NSX-T instalará los módulos de kernel en cada host ESXi del clúster, configurará el o los switches N-VDS, y creará los TEPs necesarios para formar la red de transporte Overlay.

Host Transport Nodes Edge Transport Nodes Edge Clusters ESXi Bridge Clusters NCP Clusters									
Managed by sa-vcsa-01.vclasse.local									
CONFIGURE NSX REMOVE NSX ACTIONS									
	Node	ID	IP Addresses	OS Type	NSX Configuration	NSX Version	Host Switches	Tunnels	TEP IP Addresses Node Status
<input type="checkbox"/>	SA-Management-Ed...	MoRef ID: d...							3 Hosts Not Configured
<input type="checkbox"/>	sa-esxi-02.vclasse.local	b991..1023	172.20.10.52, 172...	ESXi 7.0.0	Not Configured		0	Not Available	Not Available
<input type="checkbox"/>	sa-esxi-01.vclasse.local	b991..1011	172.20.10.51, 172...	ESXi 7.0.0	Not Configured		0	Not Available	Not Available
<input type="checkbox"/>	sa-esxi-03.vclasse.local	b991..1020	172.20.10.53, 172...	ESXi 7.0.0	Not Configured		0	Not Available	Not Available
<input type="checkbox"/>	SA-Compute-01(2)	MoRef ID: d...			Applied Profile: E...				2 Hosts Up ⓘ
<input type="checkbox"/>	sa-esxi-05.vclasse.local	0906..8750	172.20.10.55, 172...	ESXi 7.0.0	Success	3.0.0.0.0.15...	1 ↑ 5	172.20.11.152	Up ⓘ
<input type="checkbox"/>	sa-esxi-04.vclasse.local	c365..67fb	172.20.10.54, 172...	ESXi 7.0.0	Success	3.0.0.0.0.15...	1 ↑ 6	172.20.11.151	Up ⓘ

En la imagen anterior podemos ver los siguientes datos:

- Versión de vSphere
- Estado de configuración NSX
- Versión NSX
- Numero de switches creados en los nodos de transporte (N-VDS)
- IP de los TEPs
- Estado del nodo.

En este momento, solo resta llevar a cabo el despliegue de los nodos NSX Edge, luego de lo cual podremos comenzar a consumir los servicios de NSX-T, creando Segmentos, Gateways, reglas de Firewall, entre otros.

NSX EDGE CLÚSTER

NODOS NSX EDGE

Un nodo NSX Edge es otro componente del Plano de Datos de NSX-T, el cual proporciona los recursos para los componentes de routing norte-sur, permitiendo la conectividad con redes externas.

Al mismo tiempo, un nodo NSX Edge puede ser utilizado para proveer múltiples servicios de red como NAT, Load Balancer, VPN, etc.

Un nodo NSX Edge es un nodo de transporte, al igual que un host ESXi, por lo que también cuenta con un TEP para conectarse a la red Overlay.

En este caso, la función del NSX Edge es finalizar el túnel Overlay, y permitir la comunicación con redes externas.

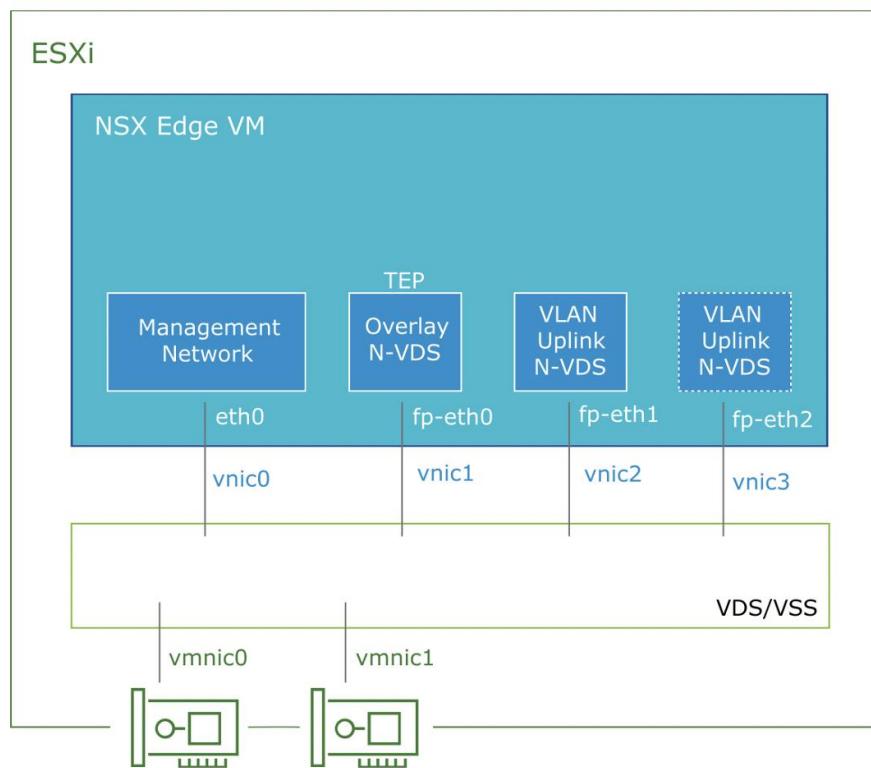
Un nodo NSX Edge puede ser desplegado como máquina virtual en un host ESXi (KVM no está soportado como host para un nodo NSX Edge), o como servidor físico/baremetal.

Tamaño	Memoria	vCPU/Cores	Disco
Small	4 GB	2	200 GB
Medium	8 GB	4	200 GB
Large	32 GB	8	200 GB
Extra Large	64 GB	16	200 GB
Bare metal (minimum)	32 GB	8	200 GB
Bare metal (recommended)	256 GB	24	500 GB

INTERFACES DE UN NODO NSX EDGE

Un nodo NSX Edge requiere de varias interfaces con distintas asignaciones:

1. La primera interfaz (eth0) será asignada a la red de administración.
 - a. Este vNIC deberá ser conectada a un Port Group en un Virtual Switch Standard o en un Virtual Distributed Switch.
 - b. El Port Group deberá proveer acceso a la VLAN de administración.
 - c. El Port Group deberá además proveer la redundancia requerida.
2. Las restantes interfaces serán configuradas para la conexión de uno o más TEPs, y para la conexión con los Uplinks como vemos en la imagen a continuación.



Es importante destacar que, como cualquier nodo de transporte, un nodo NSX Edge tendrá uno o más switches N-VDS. Una configuración bastante común es tener dos switches N-VDS en un nodo NSX-Edge

- N-VDS para la Zona de Transporte Overlay. Este switch N-VDS permite al nodo NSX-Edge contar con un TEP y conectarse con otros nodos de transporte (ESXi o KVM) a través de la red de transporte.
- N-VDS para la Zona de Transporte VLAN. Este switch N-VDS permite la conexión del NSX-Edge con una o más VLANs, las cuales luego son utilizadas como uplinks para la comunicación con las redes externas.

Un ejemplo de configuración de las restantes interfaces del nodo NSX Edge sería la siguiente, recordando que la primera interfaz siempre se utiliza para conectarse a la red de administración:

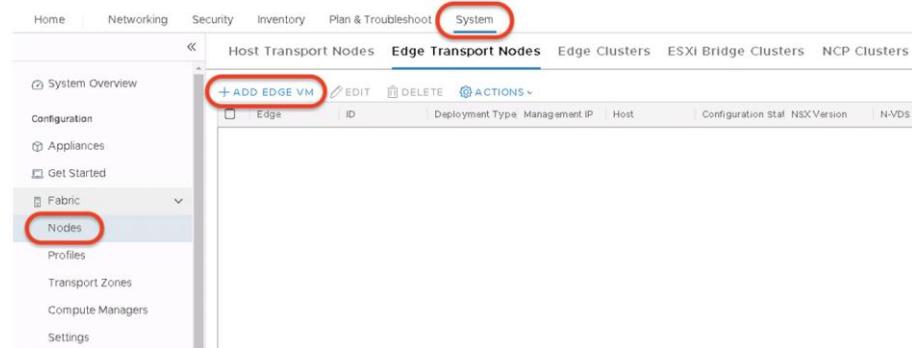
- La segunda interfaz (fp-eth0) puede ser asignada a la red Overlay, utilizando un Switch N-VDS y permitiendo la creación de un TEP.
 - Esta vNIC deberá ser conectada a un Port Group en un Virtual Switch Standard o en un Virtual Distributed Switch.
 - El Port Group deberá proveer acceso a la VLAN de la red de transporte Overlay, de manera de poder comunicarse con otros nodos de transporte (ESXi, KVM u otros nodos NSX Edge)
 - El Port Group deberá además proveer la redundancia requerida con al menos dos interfaces físicas.
- La tercera interfaz (fp-eth1) y opcionalmente también la cuarta interfaz (fp-eth2) puede ser asignada a la red VLAN, utilizando un Switch N-VDS.
 - Esta vNIC deberá ser conectada a un Port Group en un Virtual Switch Standard o en un Virtual Distributed Switch.
 - El Port Group deberá proveer acceso a las VLANs requeridas para conectarse con los switches ToR y router físicos, de manera de proveer conectividad a las redes externas, que luego será utilizado por los Gateways Tier-0. Usualmente este Port Group se configura en modo Trunk, permitiendo el paso de múltiples VLANs que luego podrán ser utilizadas por múltiples Gateways Tier-0.
 - El Port Group deberá además proveer la redundancia requerida con al menos dos interfaces físicas.

DESPLIEGUE DE UN NODO NSX EDGE

Un nodo NSX Edge puede ser desplegado de múltiples maneras:

- Utilizando la interfaz web (UI) de NSX-T (método recomendado y que detallaremos a continuación)
- Despliegue de un template OVF en vCenter Server
- Utilizar un archivo ISO. Este es el método utilizado para desplegar nodos NSX Edge Bare Metal.

Un nodo NSX Edge se puede desplegar yendo a **System > Fabric > Nodes > Edge Transport Nodes**. Desde aquí debemos hacer click en **Add Edge VM**.



En el asistente, lo primero que debemos hacer es ingresar los siguientes datos:

- Nombre del nodo NSX Edge
- FQDN del nodo NSX Edge
- Tamaño del nodo NSX Edge

1 Name and Description	Name*	sa-nxedge-01
2 Credentials	Host name/FQDN*	sa-nxedge-01.vclass.local
3 Configure Deployment	Enter Fully Qualified Domain Name (FQDN) e.g. subdomain.example.com	
4 Configure Node Settings	Description	
5 Configure NSX	Form Factor* <input type="radio"/> Small <input checked="" type="radio"/> Medium <input type="radio"/> Large <input type="radio"/> Extra Large 2 vCPU 4 vCPU 8 vCPU 16 vCPU 4 GB RAM 8 GB RAM 32 GB RAM 64 GB RAM 200 GB Storage 200 GB Storage 200 GB Storage 200 GB Storage	
	Advanced Resource Reservations CANCEL NEXT	

A continuación, lo siguiente es especificar las credenciales que serán utilizadas luego para acceder a este nodo NSX Edge:

- CLI User Name
- CLI Password

- Opcional – Permitir el acceso vía SSH al usuario CLI
- System Root Password
- Opcional – Permitir el acceso vía SSH al usuario Root
- Opcional – Proveer password para usuario Audit

Add Edge VM

1 Name and Description
2 Credentials (highlighted)
3 Configure Deployment
4 Configure Node Settings
5 Configure NSX

Credentials

CLI credentials will be set on the NSX Edge VM. These credentials can be used to login to the read only command line interface of the appliance.

CLI Credentials

CLI User Name*	admin
CLI Password*	*****
CLI Confirm Password*	*****
Allow SSH Login	<input checked="" type="checkbox"/> Yes

Root Credentials

System Root Password*	*****
System Root Confirm Password*	*****
Allow Root SSH Login	<input checked="" type="checkbox"/> Yes

Audit Credentials

El siguiente paso es especificar donde se desplegará este virtual appliance:

- Instancia de vCenter Server
- Clúster
- Datastore

Add Edge VM

1 Name and Description
2 Credentials
3 Configure Deployment (highlighted)
4 Configure Node Settings
5 Configure NSX

Configure Deployment

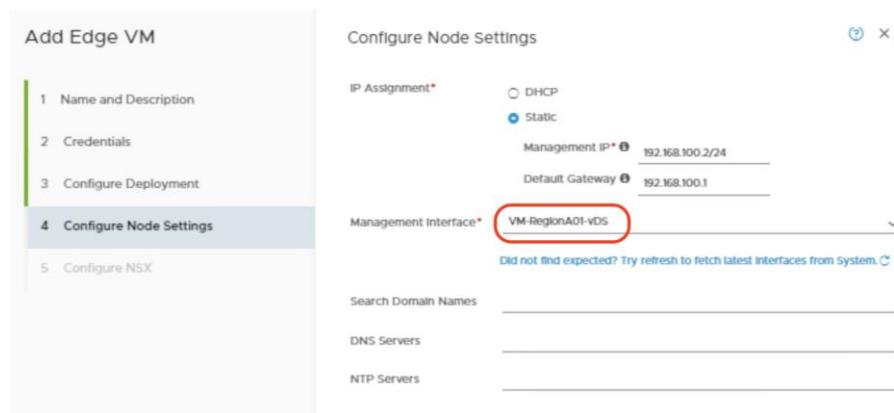
Compute Manager*	vCenter
Cluster*	RegionA01-COMP01
Resource Pool	
Host	
Datastore*	02-a

Did not find expected? Try refresh to fetch latest datastores from system.

A continuación, queda configurar la red de administración de este nodo NSX Edge:

- Método de asignación IP: DHCP o IP Estática

- IP y mascara en formato CIDR
- Default Gateway
- Port Group para conectar la vNIC a la red de administración
- DNS
- NTP



El último paso es crear los switches N-VDS asociados a la Zona de Transporte Overlay y a la Zona de Transporte VLAN. Para esto, en primer lugar, seleccionamos las Zonas de Transporte que configuraremos en este nodo NSX-Edge, en este caso “TZ-Overlay” y “TZ-VLAN”.

Veamos cómo crear el switch N-VDS para la Zona de Transporte Overlay:

- **Edge Switch Name:** Seleccionamos el nombre del switch N-VDS asociado a esta Zona de Transporte
- **Uplink Profile:** Seleccionamos un perfil que se adecue a nuestros requerimientos de redundancia, escalabilidad y balanceo de carga.
- **IP Assignment:** Método de obtención de dirección IP, en este caso usaremos IP Pool.
- **IP Pool:** Seleccionamos el IP Pool a utilizar por el TEP de este nodo NSX Edge.
- **Interfaces:** Seleccionamos un Port Group (Standard o Distribuido) para cada Uplink de este switch N-VDS. En este caso solo seleccionamos un Port Group, ya que debido al Uplink Profile que hemos elegido, solo tendremos una única interfaz para el TEP. Esto no implica que no tendremos redundancia, ya que como mencionamos previamente, la redundancia la proveemos a nivel de Port Group.

Veamos cómo crear el switch N-VDS para la Zona de Transporte VLAN:

- **Edge Switch Name:** Seleccionamos el nombre del switch N-VDS asociado a esta Zona de Transporte
- **Uplink Profile:** Seleccionamos un perfil que se aadecue a nuestros requerimientos de redundancia, escalabilidad y balanceo de carga.
- **Interfaces:** Seleccionamos un Port Group (Standard o Distribuido) para cada Uplink de este switch N-VDS. En este caso solo seleccionamos un Port Group, ya que debido al Uplink Profile que hemos elegido, solo tendremos una única interfaz como uplink, que luego será utilizado por uno o más Gateways Tier-0. Esto no implica que no tendremos redundancia, ya que como mencionamos previamente, la redundancia la proveemos a nivel de Port Group.

NSX EDGE CLUSTER

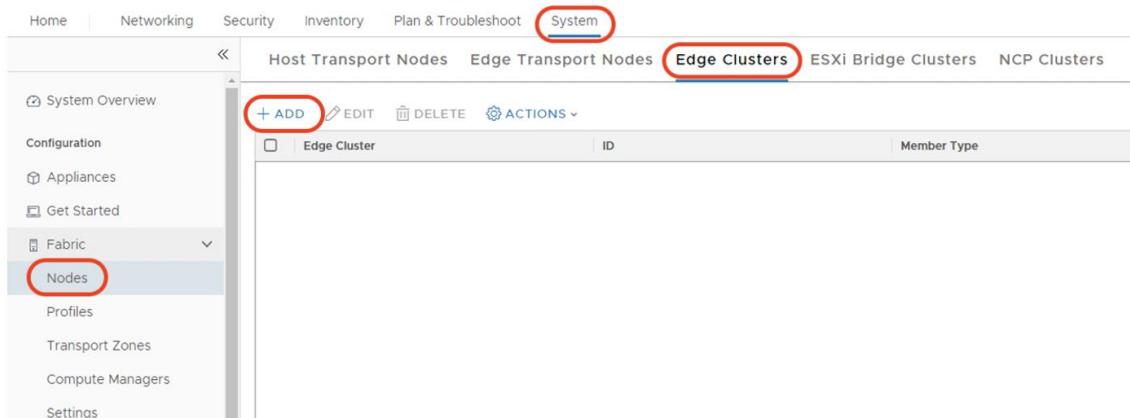
En NSX-T, los nodos NSX Edge en ningún caso proveerán servicios como componentes individuales, sino que formarán parte de un NSX Edge Clúster.

Un NSX Edge Clúster permite proveer de redundancia a los servicios de routing norte sur, así como a los demás servicios ofrecidos por los NSX Edge, al permitir configurar de 1 a 10 nodos NSX Edge en un mismo clúster. Esto permite asegurar que al menos un nodo NSX Edge está disponible para proveer servicios en caso de una falla.

Existen las siguientes consideraciones:

- Un NSX Edge Clúster puede tener de 1 a 10 nodos Edge
- Un nodo NSX Edge puede ser añadido a un único NSX Edge Clúster
- Como máximo se pueden crear 160 NSX Edge Clúster en una implementación de NSX-T.

Para poder crear un NSX Edge Clúster nos dirigimos a **System > Fabric > Nodes > Edge Clusters** y hacemos click en “Add”.



A continuación, y como vemos en la siguiente imagen, debemos ingresar los siguientes datos:

- Nombre del Edge Clúster
- Opcional – Descripción
- Edge Clúster Profile: Define el nivel de alta disponibilidad del Edge Clúster (hablaremos luego de eso)
- Tipo de Miembros: Edge Node

- Finalmente, de los Edge en la lista “Available”, seleccionamos los Edge que necesitamos y hacemos click en “>” para añadirlos a la lista “Selected” como se ve en la siguiente imagen.

Add Edge Cluster

Name *

Description

Edge Cluster Profile x v

Transport Nodes

Member Type v

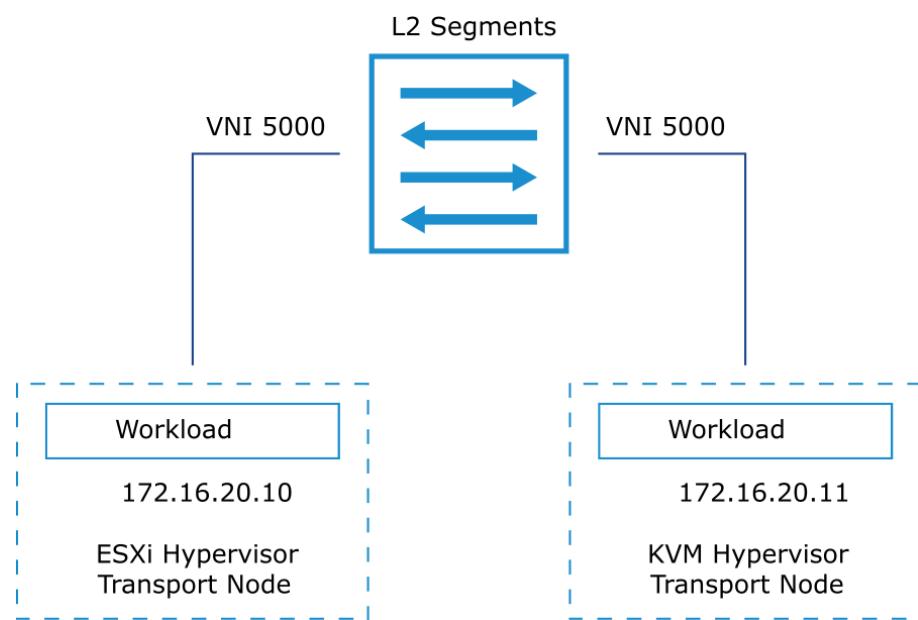
Available (2)	Selected (0)
<input checked="" type="checkbox"/> sa-nsxedge-02	
<input checked="" type="checkbox"/> sa-nsxedge-01	
> <	
BACK NEXT < > 1 - 2 of 2 records	

CANCEL
ADD

SEGMENTOS (SWITCHES LOGICOS)

INTRODUCCION

Un segmento (también conocido como Switch Lógico) es una representación de un dominio de broadcast Layer 2 a través de nodos de transporte. Al igual que con los Switches Virtuales, múltiples VMs se pueden conectar al mismo segmento para poder comunicarse entre sí, en una red layer 2, incluso cuando las VMs se encuentren en distintos nodos de transporte (ESXi/KVM).



Cada segmento tiene asignado un VNI (Virtual Network Identifier), el cual es similar a una VLAN ID. A diferencia de las VLANs, las VNI existen solo en el mundo lógico, y no requieren configuración alguna en el mundo físico, más allá de que exista comunicación entre los Nodos de Transporte de una misma Zona de Transporte, a través del uso de TEPs. Al mismo tiempo, las VNI permiten una escalabilidad mucho mayor que las que ofrecen las VLANs IDs.

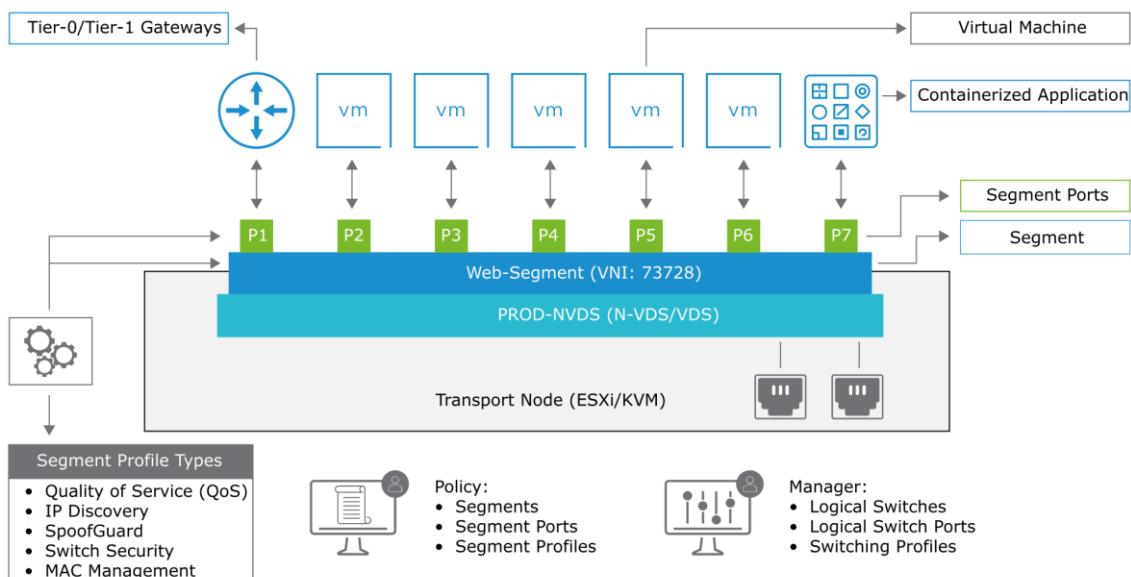
TERMINOLOGÍA

Segment Port: Un segmento, al igual que un Switch tradicional, contiene múltiples puertos. Estos puertos pueden ser utilizados luego para conectar entidades como **routers**, **VMs** o **contenedores** al segmento, y así obtener conectividad.

Segment Profiles: Segment Profiles incluyen algunos detalles de la configuración para el Segmento y los Segment Ports. Existen varios tipos de Profiles que podemos configurar:

- Quality of Service (QoS), similar al que podemos configurar en un Distributed Virtual Switch en vSphere.
- IP Discovery, que define el método utilizado por NSX para poder descubrir la IP de las VMs, pudiendo utilizar las VMware Tools, DHCP Snooping y ARP Snooping.
- SpoofGuard, que permite proteger a las VM en caso de usurpación de dirección IP.
- Switch Security, similar a las opciones de seguridad de un Switch Virtual en vSphere, incluyendo por ejemplo la opción de permitir el cambio de MAC Address en una VM.
- MAC Management, provee algunas opciones de configuración, como por ejemplo que múltiples MAC Address sean permitidas a través del mismo Segment Port.

Nota: Los Segment Profiles se pueden aplicar tanto a nivel de Segmento, como a nivel de Segment Port. Por defecto cada Segment Port hereda la configuración global del Segmento, sin embargo, a nivel de Segment Port podríamos configurar uno o más Profiles distintos, lo cual tendría prioridad por sobre los Profiles definidos globalmente en el Segmento.

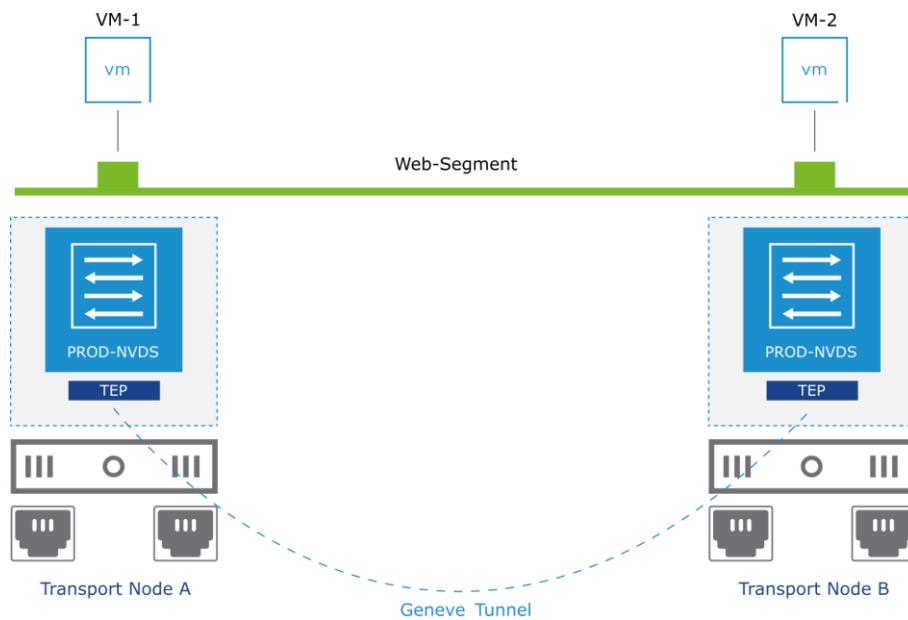


Switches N-VDS: Configurados previamente en cada Nodo de Transporte como explicamos en capítulos anteriores, proveen funcionalidades Layer 2. En vSphere 7, podemos sustituir el uso de Switches N-VDS por el uso de un Distribuyed Virtual Switch (VDS).

Zona de Transporte: Define la extensión del Segmento (en que Nodos de Transporte estará disponible), y define el tipo de Segmento (Overlay o VLAN)

TUNNELING

NSX-T Data Center utiliza redes overlay basadas en el uso de un Tunel, o Tunneling. Tunneling encapsula el tráfico de las redes virtuales y lo envía a través a de la red física. En el caso de NSX-T, se utiliza GENEVE como mecanismo para encapsular el tráfico de datos.

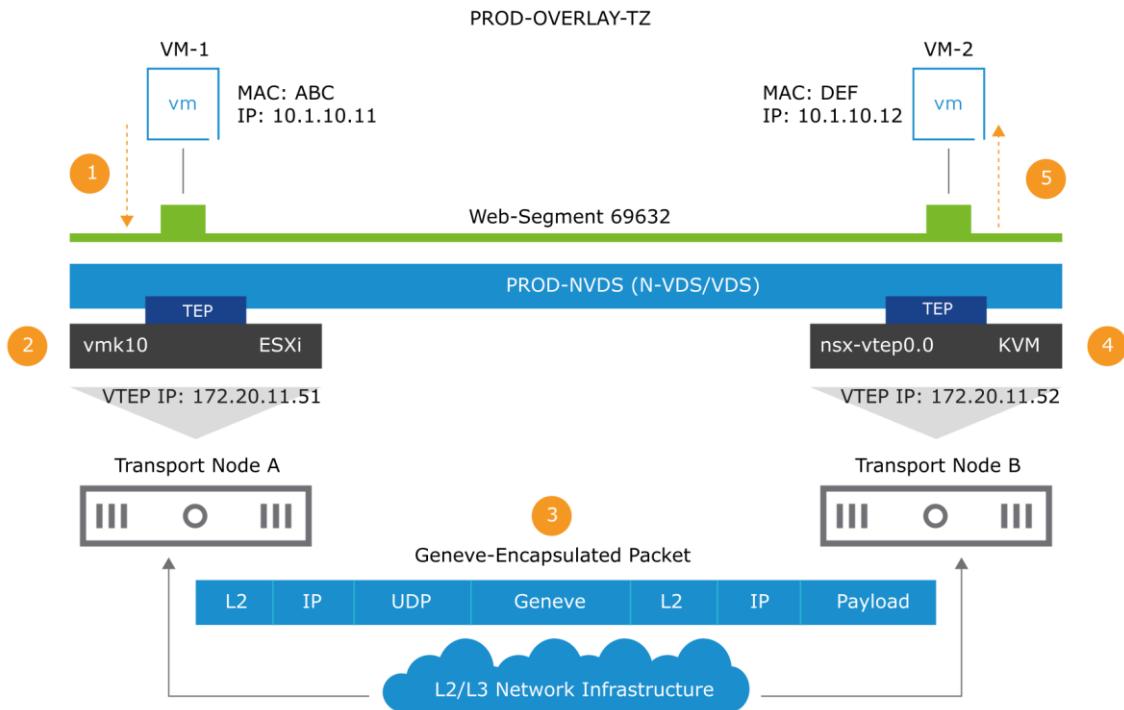


Este Túnel es creado entre los distintos Tunnel Endpoints (TEP) configurados en los Nodos de Transporte pertenecientes a la misma Zona de Transporte, como vemos en la imagen anterior.

Este Túnel también es conocido como Red de Transporte, y puede ser creado sobre redes Layer 2 o redes Layer 3, por lo que los TEPs que forman el túnel podrían estar conectados en la misma red L2, o podrían estar conectados a redes L2 distintas, comunicándose a través de una red L3.

En la imagen anterior podemos ver dos máquinas virtuales (VM-1 y VM2), ambas conectadas al mismo Segmento (Web-Segment). Las VMs se encuentran en distintos nodos de transporte que pertenecen a la misma zona de transporte.

A pesar de que el Segmento no ha sido definido en la red física, sino que existe solamente en la infraestructura lógica de NSX-T, las VMs se pueden comunicar entre si gracias a la Red de Transporte creada con los TEPs.



Geneve es un mecanismo de tunneling overlay que provee encapsulación L2 sobre L3 de los paquetes del plano de datos, lo que permite la comunicación entre VMs o Contenedores que se encuentren en distintos Nodos de Transporte, independiente de la topología de la red física.

Los paquetes son enviados de la siguiente manera como vemos en la imagen anterior:

1. La VM de origen (VM-1) envía un frame L2 a la VM de destino (VM-2)
2. El TEP en el nodo de transporte de origen (Nodo A), encapsula los frames L2 enviados por la VM de origen utilizando una cabecera Geneve en un paquete UDP.
3. El paquete UDP encapsulado es transmitido sobre la Red de Transporte al TEP de destino (Nodo B), utilizando el puerto 6081. La red de transporte puede ser una red L2 o L3.
4. El TEP de destino desencapsula la cabecera Geneve, y reenvía el frame L2 original a la VM de destino (VM-2).
5. El frame L2 es finalmente entregado a la VM de destino, de manera totalmente transparente. En este caso la VM-1 y VM-2 no son conscientes de que el frame ha viajado a través de una red física, la cual pudo incluso haber sido una red L3.

En cierta manera el mecanismo de Túnel es similar al uso de un túnel VPN, para la comunicación entre dos nodos en la red, donde el tráfico de datos puede pasar a través de múltiples dispositivos en la red, de manera totalmente transparente para los nodos de origen y destino.

CREAR UN SEGMENTO

La creación de un segmento es un proceso bastante simple como veremos a continuación. En primer lugar, debemos ir a **Networking > Segments** > y hacer click en **ADD SEGMENT**.

A continuación, debemos proveer los siguientes datos como vemos en la imagen a continuación:

1. **Segment Name:** Nombre del Segmento
2. **Connectivity:** El segmento puede ir conectado a un Gateway Tier-0 o Tier-1 para poder obtener conectividad L3, lo cual veremos en el siguiente capítulo. Ahora mismo lo podemos dejar como “None”, lo cual implica que solo se permite conectividad L2.
3. **Transport Zone:** Zona de transporte a la que pertenece este Segmento. Puede ser una Zona de Transporte Overlay o VLAN (usada para crear uplinks para NSX Edges).
4. Opcionalmente podemos configurar también los Segment Profiles que serán utilizados por el Segmento.
5. Hacemos click en Save.

subnets	Ports	Admin State
172.16.10.1/24 CIDR e.g. 10.22.12.2/23 Gateway CIDR CIDR e.g. fc7e:f206:db42::1/48	Set ⓘ	<input checked="" type="checkbox"/>

En caso de que utilicemos vSphere, los Segmentos se podrán ver de la siguiente manera como vemos en la imagen a continuación:

	Segment Name	Connectivity	Transport Zone
⋮ > ⓘ	App-Segment	None	PROD-Overlay-TZ Overlay
⋮ > ⓘ	DB-Segment	None	PROD-Overlay-TZ Overlay
⋮ > ⓘ	Web-Segment	None	PROD-Overlay-TZ Overlay

Segmentos en NSX-T

Segmentos en vSphere con N-VDS: Podemos ver que los segmentos están disponibles como redes opacas, es decir que las podemos ver y utilizar en vSphere para conectar VMs, pero no podemos modificar o eliminar estas redes.

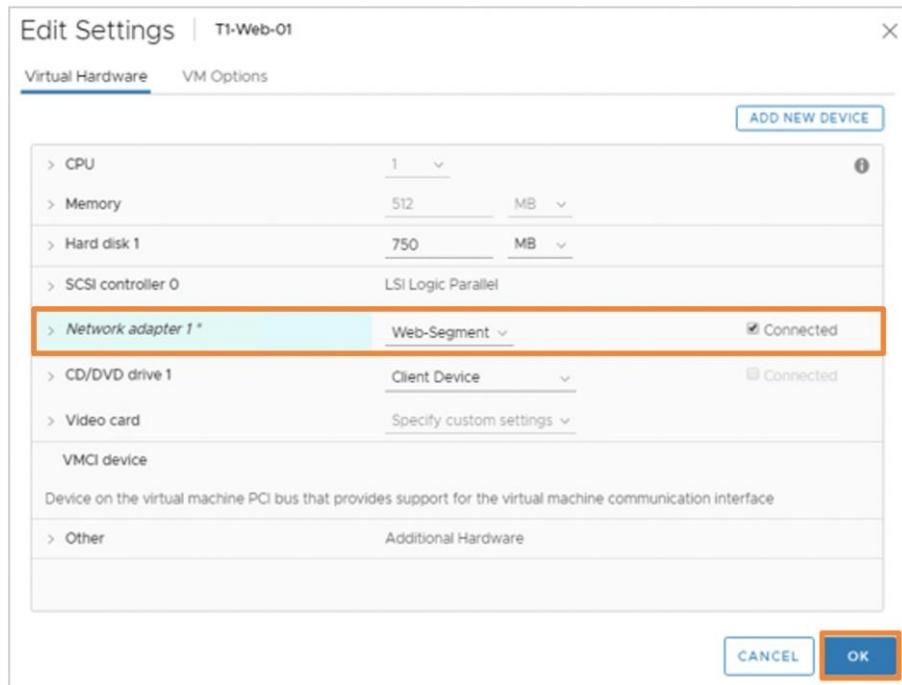
The screenshot shows the vSphere Client interface with the title bar "ESXi Hosts Configured with N-VDS". The navigation bar includes "vm", "vSphere Client", "Menu", and a search bar. The main pane displays a tree structure under "sa-vcsa-01.vclass.local" with "SA-Datacenter" expanded. Inside "SA-Datacenter", three segments are listed: "App-Segment", "DB-Segment", and "Web-Segment", with "Web-Segment" highlighted. To the right, a summary card for "Web-Segment" is shown with the status "Opaque Network".

Segmentos en vSphere con VDS: En este caso, los segmentos de NSX-T los veremos cómo Port Groups en el Switch VDS seleccionado.

The screenshot shows the vSphere Client interface with the title bar "ESX Hosts Configured with VDS". The navigation bar includes "vm", "vSphere Client", "Menu", and a search bar. The main pane displays a tree structure under "sa-vcsa-01.vclass.local" with "sb-vcsa-01.vclass.local" expanded. Inside "sb-vcsa-01.vclass.local", "SB-Datacenter" is expanded, showing "dvs-SB-Datacenter". Under "dvs-SB-Datacenter", three segments are listed: "App-Segment", "DB-Segment", and "Web-Segment", with "Web-Segment" highlighted. To the right, a summary card for "Web-Segment" is shown with the status "NSX Port Group". A green arrow points from the "NSX Port Group" label to the "Web-Segment" entry in the list.

CONECTAR UNA VM A UN SEGMENTO

Para conectar una VM a un segmento, el proceso es el mismo que haríamos con cualquier VM que conectamos a un Port Group en vSphere, es decir, editamos la VM, y seleccionamos el Port Group deseado, el cual puede ser una red opaca N-VDS o un Port Group VDS.



NSX CONTROLLER TABLES

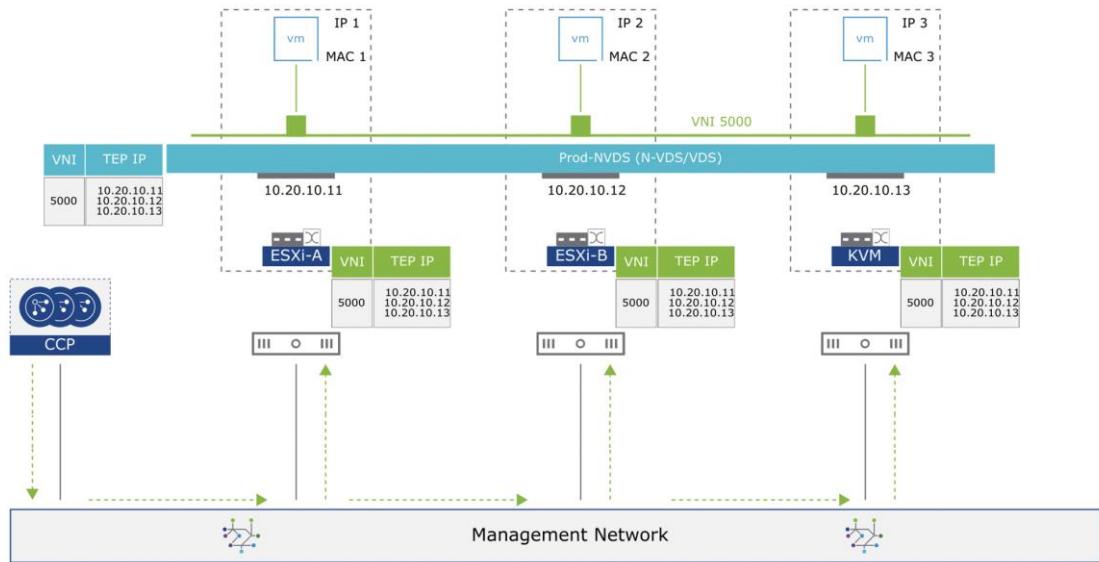
En los NSX Controllers se mantienen tres tablas de plano de control asociadas a los Segmentos, que permiten la comunicación Layer 2.

TABLAS TEP

Proveen un mapping de las IPs de los TEPs donde se encuentra visible un Segmento, como vemos en la siguiente imagen:

1. Tenemos un Segmento con la VNI 5000.
2. El segmento se encuentra disponible en tres nodos de Transporte (ESXi-A, ESXi-B y KVM).

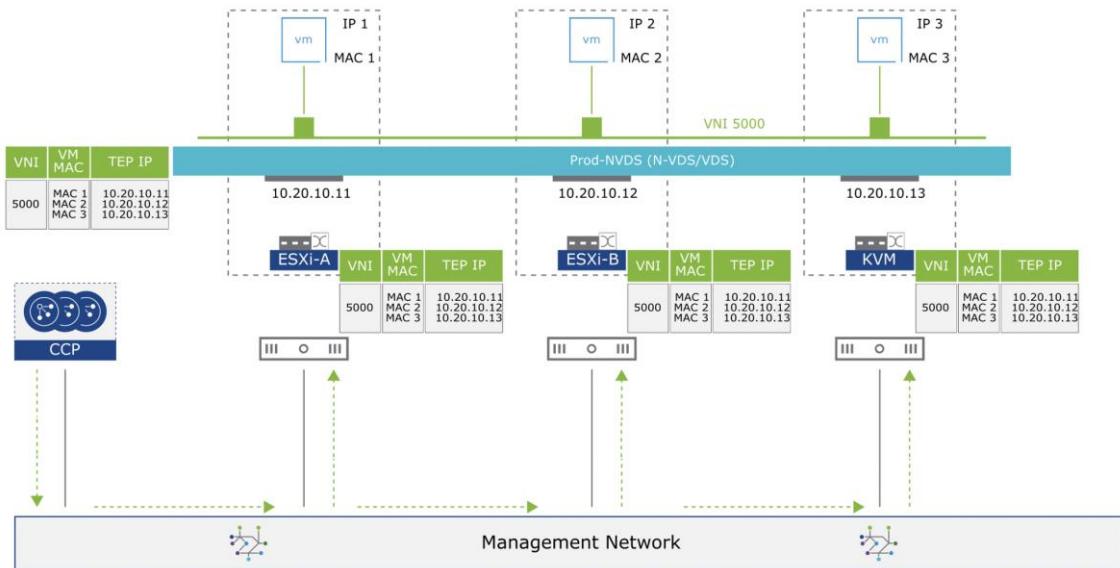
3. Cada nodo de transporte informa de esta VNI 5000, y la IP de su propio TEP al NSX Controller.
4. NSX Controller consolida esta información, con lo cual la tabla de TEP contiene información del Segmento VNI 5000, y la IP de los 3 TEPs donde se encuentra disponible.
5. Esta información consolidada es enviada a cada Nodo de Transporte que mantiene una copia local de esta tabla consolidada.



TABLAS MAC

Proveen un mapping de la MAC Address de cada VM conectada a un Segmento, con la IP del TEP a través del cual la MAC Address es alcanzable. Como vemos en la siguiente imagen:

1. Tenemos tres VMs conectadas al mismo Segmento VNI 5000.
2. Cada Nodo de Transporte registra la MAC Address de las VM que están en dicho nodo, y la asocia con la VNI (5000) y con la IP del TEP.
3. Esta información es enviada por el Nodo de Transporte al NSX Controller.
4. NSX Controller consolida esta información, por lo que ahora la Tabla MAC contiene información de la MAC Address de las tres VMs asociadas al Segmento con VNI 5000, y la IP del TEP a través del cual podemos alcanzar dichas MAC Address.
5. Esta información consolidada es enviada a cada Nodo de Transporte que mantiene una copia local de esta tabla consolidada.

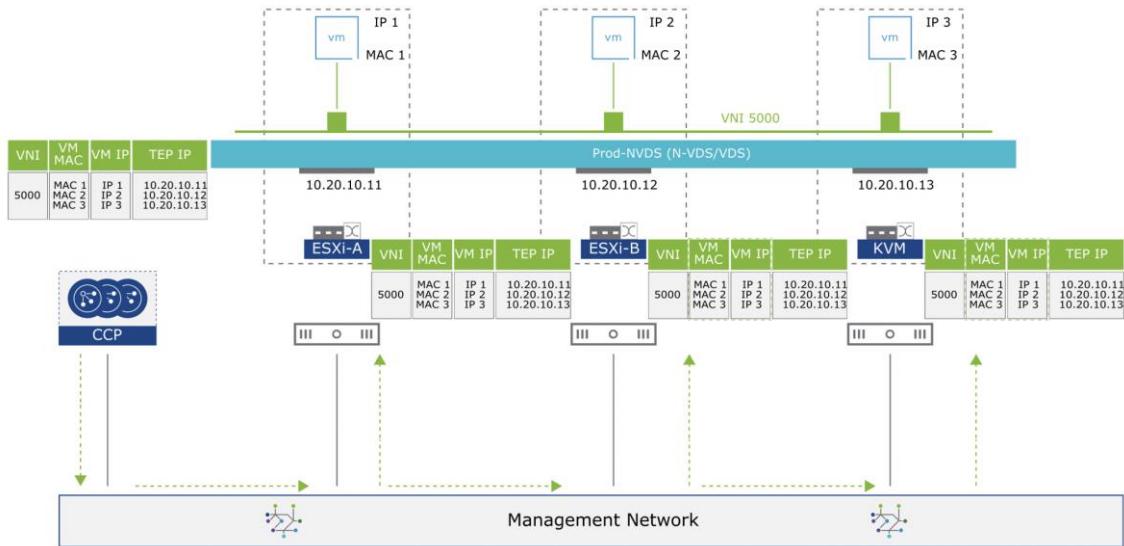


TABLAS ARP

Proveen un mapping de la IP y MAC Address de cada VM en un Segmento, con el fin de reducir o suprimir el tráfico ARP en las redes lógicas. Como vemos en la siguiente imagen:

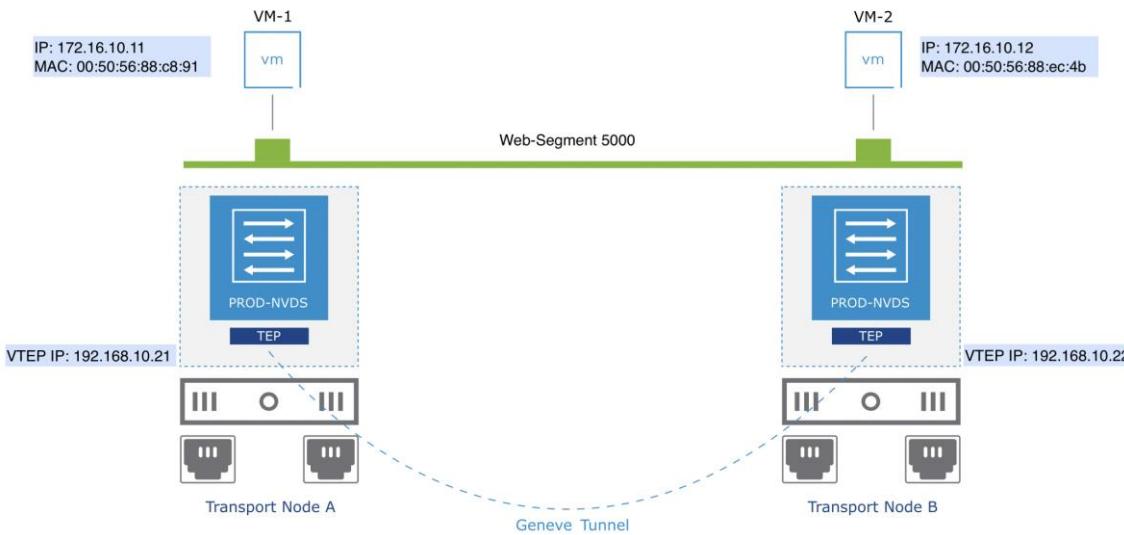
1. Tenemos tres VMs conectadas al mismo Segmento VNI 5000.
2. Cada Nodo de Transporte registra la MAC Address y la IP de las VM que están en dicho nodo, y la asocia con la VNI (5000) y con la IP del TEP.
3. Esta información es enviada por el Nodo de Transporte al NSX Controller.
4. NSX Controller consolida esta información, por lo que ahora la Tabla ARP contiene información de la MAC Address e IP de las tres VMs asociadas al Segmento con VNI 5000.

Esta información consolidada es enviada a cada Nodo de Transporte que mantiene una copia local de esta tabla consolidada.



FUNCIONAMIENTO DE LAS NSX CONTROLLER TABLES

En la sección anterior vimos que los NSX Controllers mantenían tres Tablas en el plano de Control: Tablas de TEP, MAC Address y ARP. A continuación, veremos cómo y cuándo se utilizan dichas tablas.



USO DE TABLA ARP

Imaginemos que deseamos comunicar dos VMs que se encuentran en distintos Nodos de transporte, ambas conectadas al mismo VNI 5000. Asumamos además que esta será la primera comunicación entre ambas VMs.

1. VM-1 desea hacer un ping a VM-2
2. VM-1 conoce la IP de VM-2 (172.16.10.12), pero no conoce su MAC Address (00:50:56:88:ec:4b). Debido a esto, VM-1 enviará un **ARP Request** (broadcast) a través de la red, solicitando la MAC Address asociada a la IP 172.16.10.12.
3. El TEP en el Nodo de Transporte intercepta el **ARP Request**, y en vez de enviar el Broadcast a través de la red física, primero verificará la Tabla ARP almacenada en el Nodo de Transporte, y caso de no encontrar la información, le consultará esta información al NSX Controller.
4. Una vez que el TEP obtenga la información de la MAC Address de la VM-2, reenviará esta información a la VM-1. Ahora la VM-1 conoce la IP y MAC Address de la VM-2.

USO DE TABLA MAC

Ahora que la VM-1 conoce la IP y MAC Address de la VM-2, la VM puede formar el Frame L2, con la IP y MAC Address de origen y destino. A continuación, la VM-1 intentará enviar dicho Frame L2 a través de la red:

1. Cuando el Frame L2 alcance al TEP del Nodo de Transporte de origen, el Nodo intentará encapsular el Frame L2 como un paquete UDP L3. Para esto necesita conocer la ubicación de la VM-2 según su MAC Address.
2. El Nodo de Transporte inspeccionará el Frame L2 enviado por la VM-1 para conseguir la MAC Address de destino de dicho Frame.
3. El Nodo de Transporte necesita ahora determinar la ubicación de dicha MAC Address de destino, básicamente averiguando el TEP asociado a dicha MAC Address.
4. El Nodo de Transporte consulta entonces la Tabla MAC, para conseguir esta información.
5. Con esta información, el Nodo de Transporte de origen averigua que la MAC Address de destino asociada a VM-2 se encuentra ubicada en el Nodo de Transporte con el TEP 192.168.10.22.
6. Con esta información, estamos casi listos para poder encapsular este Frame L2 y enviarlo a través de la Red de Transporte.

USO DE TABLA TEP

Como paso final para completar el proceso de encapsulamiento Geneve y enviar el Frame L2 a la VM-2, el Nodo de Transporte de origen (192.168.10.21) debe averiguar la MAC Address del TEP de destino con IP 192.168.10.22.

1. El Nodo de Transporte consulta entonces la Tabla TEP, para conseguir la MAC Address asociada con la IP del TEP de destino.
2. Una vez conseguida la MAC Address del TEP de destino, el Nodo de Transporte de origen puede construir la cabecera Ethernet del paquete UDP Geneve, y así poder reenviar este paquete UDP a través de la red de transporte.
3. El paquete UDP finalmente alcanza al TEP de destino, el cual desencapsula el paquete y lo entrega a la VM-2 según lo requerido por VM-1.

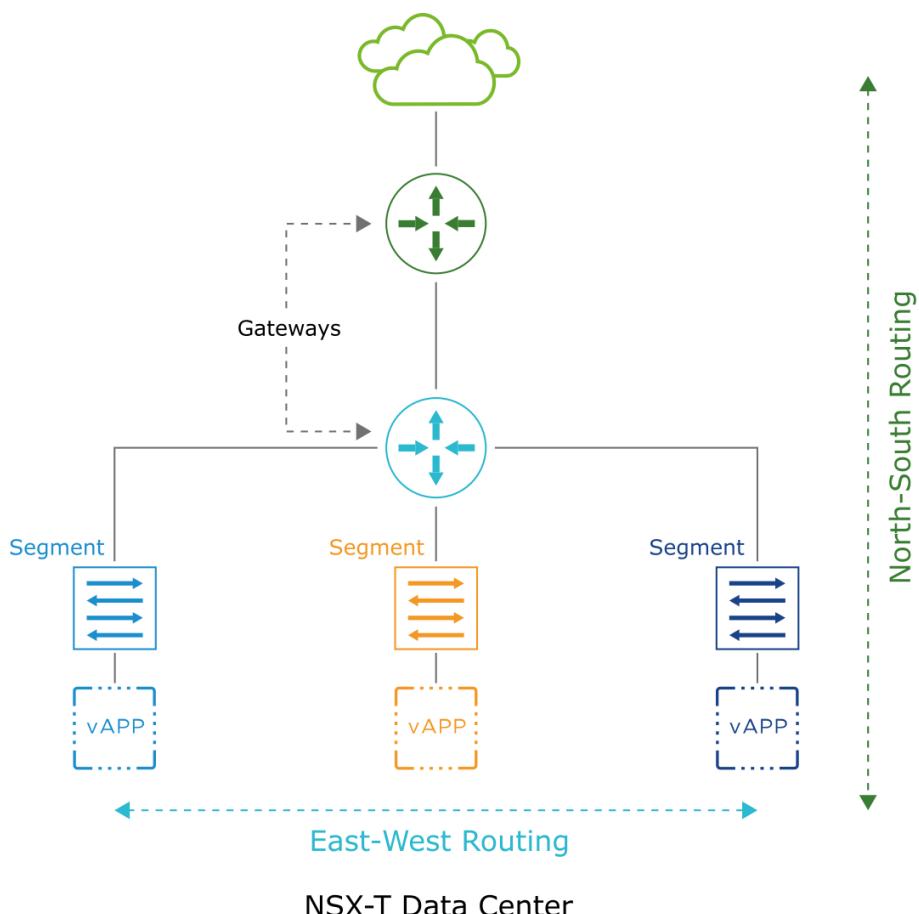
La tabla TEP también es utilizada en caso de necesitar enviar un Broadcast, Unicast o Unknown Unicast a través de un Segmento.

ROUTER LÓGICOS

Hasta ahora hemos hablado de como proveer conectividad L2 con Segmentos lógicos en NSX-T. A continuación, daremos el siguiente paso y discutiremos como proveer de conectividad Layer 3 a través del uso de Gateways Tier-0 y Tier-1 en NSX-T.

Para proveer de routing lógico, se deben cumplir los siguientes requerimientos:

- El NSX Management Clúster debe estar completamente formado y disponible.
- Se deben haber creado Zonas de Transporte y Switches N-VDS y/o VDS
- Los hypervisores deben estar preparados como nodos de transporte NSX-T y conectados a las Zonas de Transporte apropiadas.
- Se deben desplegar nodos NSX Edge y configurados de acuerdo con los requerimientos de conectividad.



TOPOLOGIAS DE ROUTING CON NSX-T

NSX-T provee dos tipos de routing lógico:

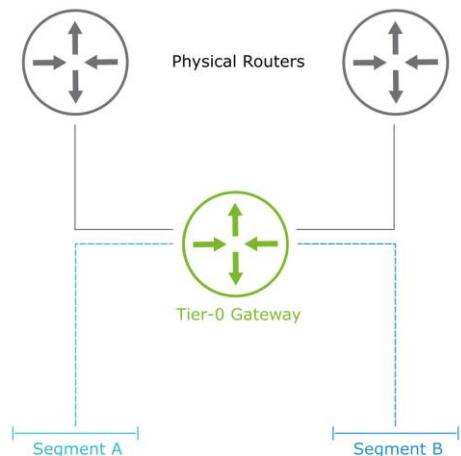
- **Routing centralizado norte-sur**, provisto por Gateways Tier-0 (T0) en conjunto con un NSX Edge Clúster.
- **Routing distribuido este-oeste**, provisto por Gateways Tier-0 así como también por Gateways Tier-1.

Adicionalmente, los Gateways Tier-0/Tier-1 proveen soporte multi-tenant, además de servicios stateful centralizados, tales como:

- NAT
- Load Balancing
- VPN

TOPOLOGIA SINGLE-TIER

En un despliegue en topología Single-Tier solo se utilizan Gateways Tier-0 (no se utilizan Tier-1) para proveer routing Este-Oeste y routing Norte-Sur. En este caso los segmentos son conectados directamente al Tier-0.

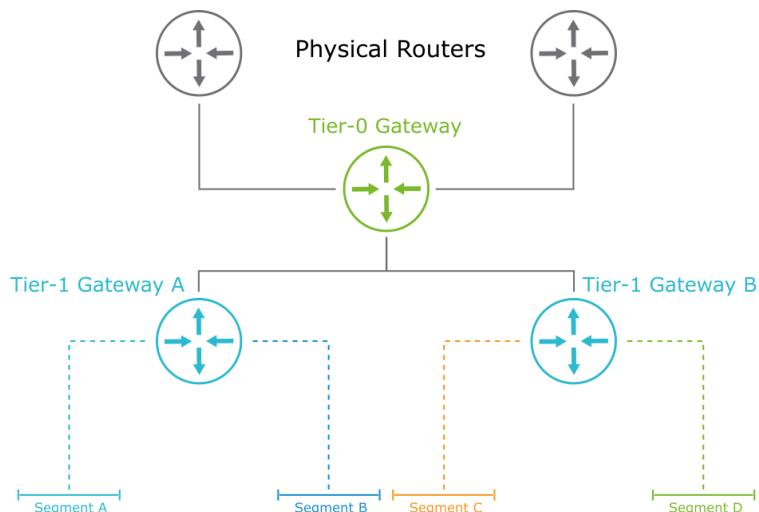


TOPOLOGIA MULTITIER

En un despliegue en topología multtier, los segmentos son conectados a Gateways Tier-1, los cuales proveen routing distribuido Este-Oeste. Del mismo modo, los Tier-1 se conectan a un Gateway Tier-0 para poder contar con routing Norte-Sur.

La topología de routing Multitier no es obligatoria, y es usualmente desplegada en ambientes multitenant:

- El proveedor de servicios será el responsable de la configuración y gestión del Gateway Tier-0
- Los tenants serán dueños de los Gateways Tier-1 y se harán cargo de su configuración.



COMPONENTES DE GATEWAYS TIER-0 Y TIER-1

DISTRIBUTED ROUTER (DR)

El componente DR (Distributed Router) se despliega cada vez que se crea un Gateway Tier-0 o un Gateway Tier-1

- Provee funcionalidades de routing distribuido este-oeste.
- Provee funcionalidades básicas de reenvío de paquetes.
- Se despliega a través de todos los Nodos de Transporte que pertenecen a una Zona de Transporte determinada, incluyendo hipervisores y nodos Edge.
- Provee el primer salto (hop) para el proceso de routing directamente en el hipervisor.

SERVICE ROUTER (SR)

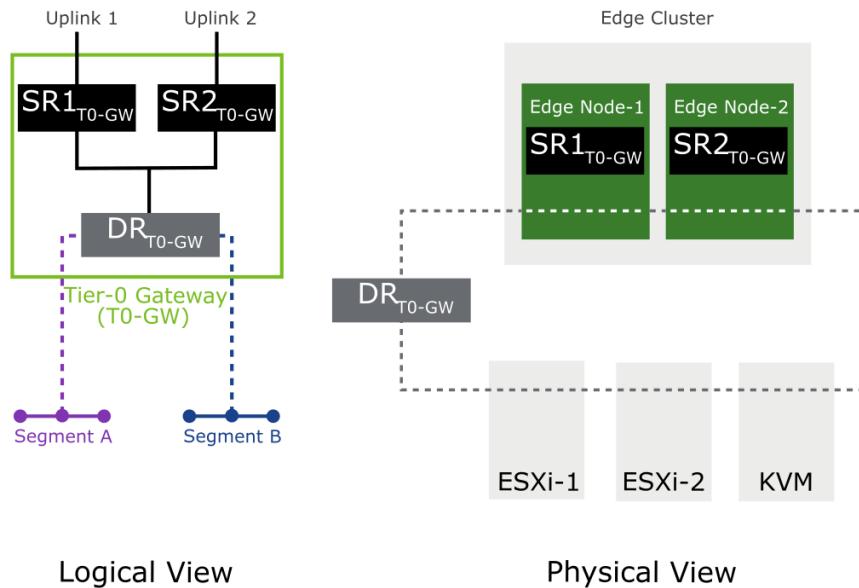
El componente SR (Service Router) se despliega obligatoriamente en un NSX Edge Clúster cuando desplegamos un Gateway Tier-0 de manera de proveer conectividad norte sur.

Al mismo tiempo, un SR se puede desplegar opcionalmente para Gateways Tier-1 cuando el Tier-1 provee servicios stateful centralizados como NAT o Load Balancing. En este caso el SR se despliega también en un NSX Edge Clúster.

COMPONENTES DR Y SR EN TOPOLOGÍA SINGLE-TIER

Como vemos en la siguiente imagen, tenemos un único Gateway Tier-0 llamado **T0-GW**, el cual despliega los siguientes componentes:

- Componente DR en cada nodo de transporte, incluyendo hosts ESXi, KVM y nodos NSX Edge.
- Componente SR desplegado en los nodos NSX Edge del NSX Edge Clúster asignado al Gateway Tier-0.
- Los segmentos de conectan directamente al Tier-0
- Los NSX Edge, a través de sus uplinks, proveen el acceso norte-sur, de manera de permitir la comunicación hacia la red física.

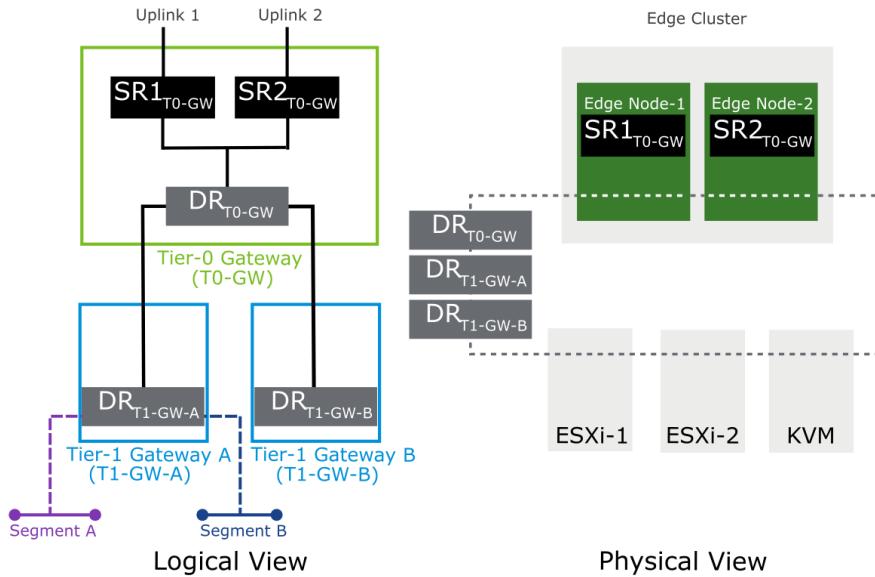


COMPONENTES DR Y SR EN TOPOLOGÍA MULTI TIER

Como vemos en la siguiente imagen, tenemos un único Gateway Tier-0 (T0-GW) y dos Gateway Tier-1 (T1-GW-A y T1-GW-B), los cuales despliegan los siguientes componentes:

- Componente DR de cada Tier-0 y Tier-1 en cada nodo de transporte, incluyendo hosts ESXi, KVM y nodos NSX Edge. En este caso, en cada Nodo de Transporte se despliegan tres DR, un DR correspondiente al Tier-0, y dos DR correspondientes a los dos Tier-1 desplegados.
- Componente SR desplegado en los nodos NSX Edge del NSX Edge Clúster asignado al Gateway Tier-0.

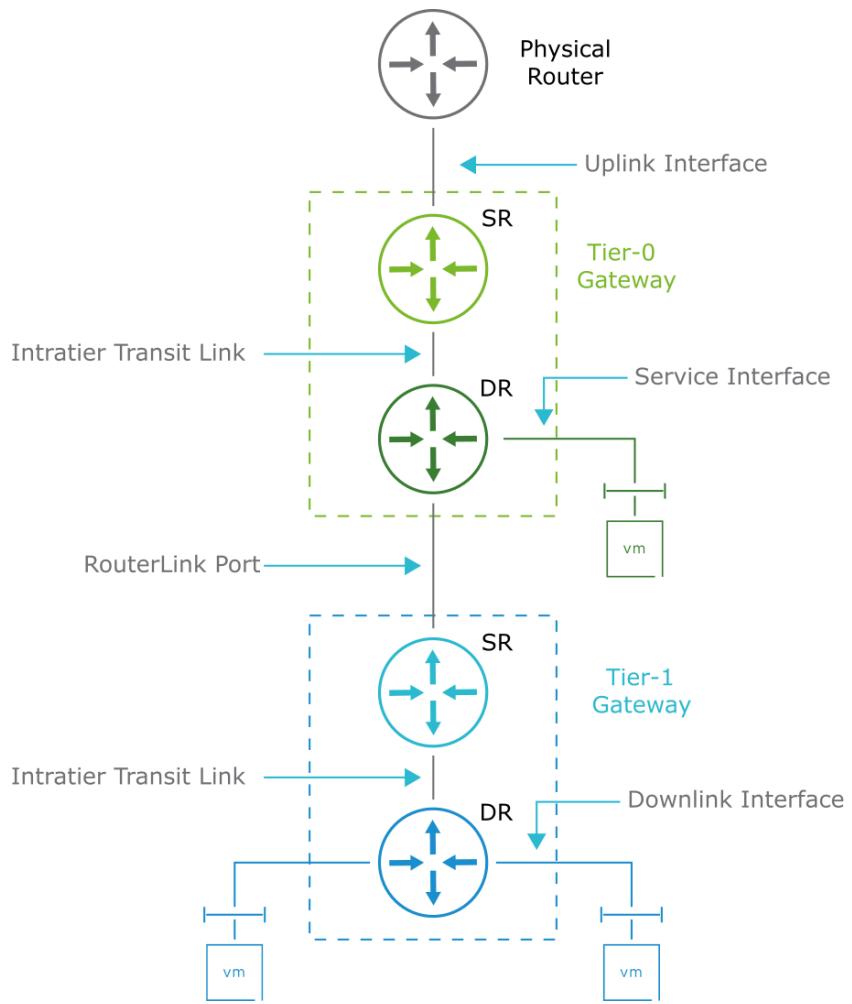
- Los segmentos de conectan a los Gateway Tier-1.
- Los Gateway Tier-1 se conectan a un Gateway Tier-0.
- Los NSX Edge, a través de sus uplinks, proveen el acceso norte-sur, de manera de permitir la comunicación hacia la red física.



INTERFACES PARA GATEWAYS TIER-0 Y TIER-1

A continuación, un detalle de las distintas interfaces utilizadas por los Gateway Tier-0 y Tier-1

- Interfaces **Uplink** conectan los Gateway Tier-0 con dispositivos físicos para conectividad norte-sur
- Interfaz **Downlink** permite conectar los Segmentos a un Gateway
- Interfaz **RouterLink** permite conectar un Gateway Tier-0 con un Gateway Tier-1. Segmento por defecto: **100.64.0.0/10**
- **Intratier Transit Link**, es una conexión interna entre el componente DR y SR del mismo Gateway (Tier-1 o Tier-0). Segmento por defecto: **169.254.0.0/28**
- **Service Interface** es una interfaz especial para servicios basados en VLAN y servicios de redirección a servicios de seguridad de terceros.



GATEWAY TIER-1

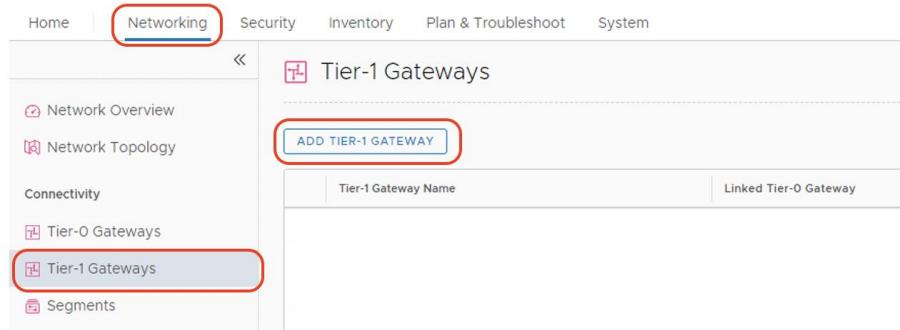
A continuación, vamos a ver como es el proceso de despliegue de un Gateway Tier-1. Este tipo de Gateway está diseñado para ambientes Multi Tenant, y tienen las siguientes características:

- El Tenant es el dueño del Tier-1 y es responsable de su configuración y administración.
- No requiere el uso de protocolos de routing dinámico. De hecho, esta opción ni siquiera se encuentra disponible.
- No soporta ECMP, y debe estar conectado a un único Gateway Tier-0 para obtener conectividad externa.

- Ofrece servicios de Default Gateway a los segmentos lógicos conectados a este Tier-1. De esta manera, las máquinas virtuales conectadas a dichos segmentos utilizarán este Tier-1 como Default Gateway y primer salto (hop) en el proceso de routing.

PROCESO DE DESPLIEGUE

El proceso de despliegue de un Gateway Tier-1 es bastante simple como veremos a continuación. En primer lugar, debemos ir a **Networking > Tier-1 Gateways** > y hacer click en **ADD TIER-1 GATEWAY**



A continuación, debemos ingresar los siguientes datos:

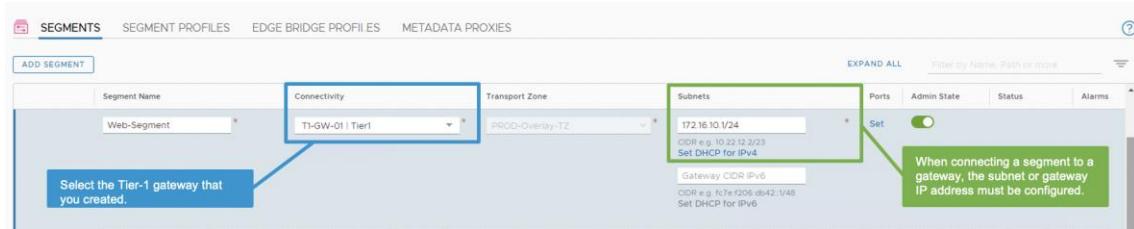
- Nombre del Tier-1 Gateway
- **Linked Tier-0 Gateway** lo dejamos en blanco por ahora, ya que aún no hemos creado ningún Gateway Tier-0.
- **Edge Clúster** también lo dejamos en blanco, ya que el Tier-1 solo requiere de un Edge Clúster cuando habilitamos servicios **stateful** como NAT o Load Balancing. En este caso solo estamos habilitando el servicio de routing, por lo que no es obligatorio.
- Hacemos click en **Save** para guardar los cambios.

Tier-1 Gateway Name	Linked Tier-0 Gateway	#Linked Segments
T1-GW-01	Select Tier-0 Gateway	
Edge Cluster	Select Edge Cluster	Edges Set ⓘ
Edges Pool Allocation Size	Select Pool Allocation Size	Enable Standby Relocation
Tags	Tag (Required) Scope (Optional)	
NOTE - Before further configurations can be done, fill out mandatory fields above (*), click 'Save' below.		
> SERVICE INTERFACES		
> STATIC ROUTES		
SAVE		CANCEL

CONECTAR SEGMENTOS A GATEWAY TIER-1

Ya tenemos creado el primer Gateway Tier-1, por lo que el siguiente paso sería conectar los segmentos requeridos a este Tier-1, de manera de utilizarlo como Default Gateway para las VMs conectadas a dichos segmentos.

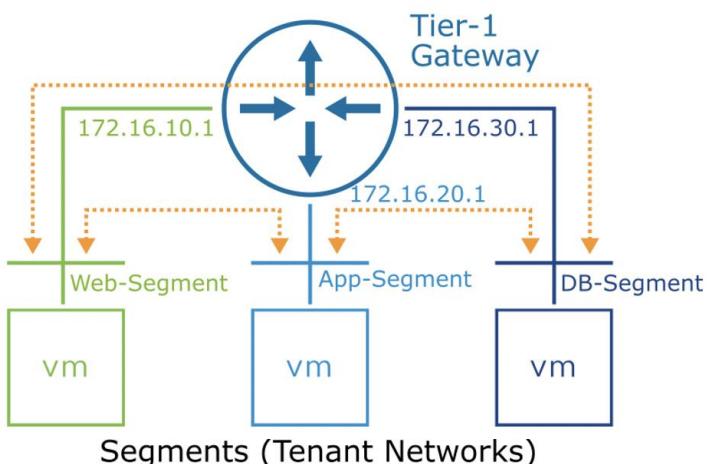
En primer lugar, debemos ir a **Networking > Segments** y editamos uno de los Segmentos creados previamente. En este ejemplo estamos editando el segmento llamado **Web-Segment**.



Al editar el Segmento, debemos ingresar los siguientes datos:

- **Connectivity:** Seleccionamos el Gateway Tier-1 al cual conectaremos este Segmento.
- **Subnets:** Especificamos la IP que utilizaremos como Default Gateway para este Segmento. Esta IP será configurada en la interfaz del Tier-1 al cual se conecta el Segmento para proveer así el servicio de Default Gateway. La Subnet la podemos definir en formato CIDR, de manera también de definir el segmento de red que será utilizado por las VMs de este segmento.

Siguiendo los mismos pasos anteriores podemos conectar múltiples segmentos a este Gateway Tier-1, permitiendo proveer de servicio de Routing distribuido Este-Oeste, lo cual permite la comunicación entre VMs conectadas a distintos Segmentos como vemos en la imagen a continuación:



GATEWAY TIER-0

A continuación, vamos a ver como es el proceso de despliegue de un Gateway Tier-0. Este tipo de Gateway está diseñado para proveer conectividad norte-sur, es decir, permite **conectar los Segmentos lógicos con las redes físicas**. Este Gateway también permite Routing distribuido este-oeste con el componente DR desplegado en cada hipervisor, como se discutió previamente.

- El Service Provider es el dueño del Tier-0 y es responsable de su configuración y administración.
- Soporta rutas estáticas y dinámicas vía BGP a través de los uplinks permitiendo compartir información de rutas con los gateways físicos mediante sesiones BGP.
- Soporta ECMP en la conexión entre el Tier-0 y los Gateways físicos.
- Ofrece servicios de Default Gateway a los segmentos lógicos conectados directamente a este Tier-0. De esta manera, las máquinas virtuales conectadas a dichos segmentos utilizarán este Tier-0 como Default Gateway y primer salto (hop) en el proceso de routing.
- Siempre requiere el uso de un NSX Edge Clúster si se desea configurar Uplinks para este Gateway Tier-0.

CONFIGURACION DE UPLINKS

Un Gateway Tier-0 puede conectarse con la red física a través de Uplinks. Estos Uplinks deben pertenecer a una Zona de Transporte VLAN y estar asociados a uno o más nodos NSX Edge, que son los que proveerán la conectividad a la red física.

Recordemos que previamente hemos desplegado nodos NSX Edge configurados para pertenecer al menos a dos Zonas de Transporte:

- **Una zona Overlay** que mediante un TEP permite comunicar el nodo NSX Edge con otros Nodos de Transporte
- **Una zona VLAN** que permite conectar el nodo NSX Edge a uno o más Port Groups en vSphere, los cuales a su vez permiten la conectividad con los Gateway físicos.

El primer paso para la configuración de estos Uplinks es la creación de Segmentos que pertenezcan a una Zona de Transporte VLAN, y que luego serán asociados con las interfaces de los nodos NSX Edge.

Se debe crear un Segmento **por cada interfaz Uplink de cada nodo NSX Edge**. Es decir, si cada nodo NSX Edge cuenta con dos Uplinks en la Zona de Transporte VLAN, y tenemos dos NSX Edge en el NSX Edge Clúster, entonces debemos crear cuatro Segmentos. En nuestro ejemplo, tenemos dos NSX Edge en el Edge Clúster, y cada uno cuenta con un único

Uplink, por lo que debemos configurar solo dos Segmentos como vemos en la siguiente imagen:

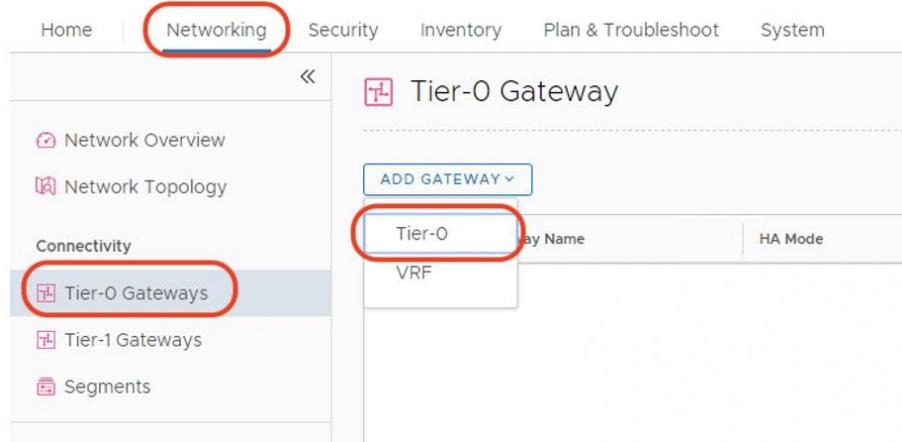
Segment Name	Connectivity	Transport Zone	Subnets	Ports	Admin State	Status	Alarms
Uplink-1	Isolated	PROD-VLAN-T2 VLAN		0	Up	Success	0
Uplink-2	Isolated	PROD-VLAN-T2 VLAN		0	Up	Success	0

Como vemos en la imagen anterior, debemos ingresar los siguientes datos:

- **Nombre del Segmento**
- **Connectivity**: Lo dejamos en blanco, es decir no conectamos el Uplink a ningún Gateway Tier-0 o Tier-1, por lo que luego el Segmento es mostrado como “Isolated” o aislado.
- **Zona de Transporte**: Seleccionamos la Zona de Transporte VLAN a la que pertenecen los nodos NSX Edge

PROCESO DE DESPLIEGUE

El proceso de despliegue de un Gateway Tier-0 es bastante simple como veremos a continuación. En primer lugar, debemos ir a **Networking > Tier-0 Gateways > Add Gateway** y hacer click en **Tier-0**

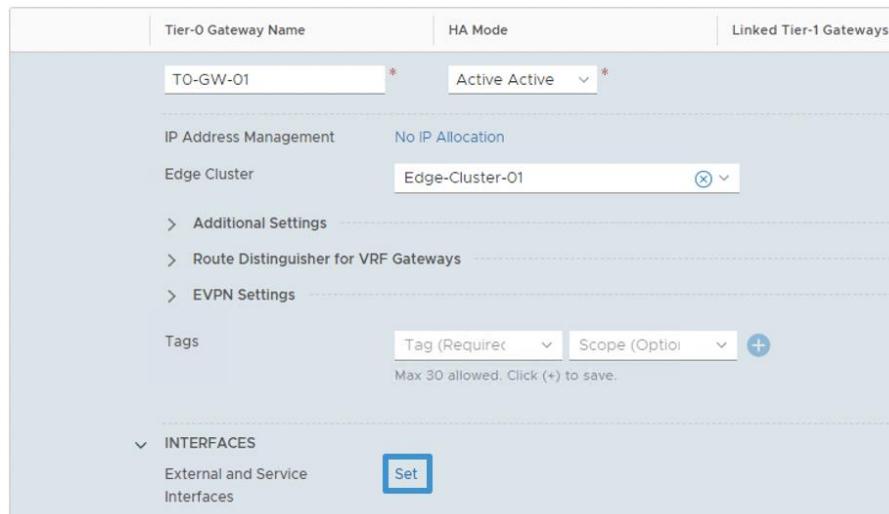


A continuación, debemos ingresar los siguientes datos:

- Nombre del Tier-0 Gateway
- **HA Mode:** Aquí tenemos dos alternativas:
 - **Active-Active:** Utilizado comúnmente por los Gateway Tier-0 cuando solo proveen servicios de Routing o Stateless NAT. Este modo no es compatible con servicios Stateful.
 - **Active-Standby:** Utilizado por Gateways Tier-0 o Tier-1 que proveerán servicios de Routing y además proveerán algún servicio Stateful como NAT o Load Balancing.
- **Edge Clúster:** Seleccionamos el NSX Edge Clúster que nos proveerá conectividad con la red física, y que opcionalmente utilizaremos para proveer servicios Stateful
- Hacemos click en **Save** para guardar los cambios. El asistente nos preguntará si deseamos seguir con la configuración del Gateway Tier-0, hacemos click en **YES**.

Tier-0 Gateway Name	HA Mode	Linked Tier-1 Gateways
TO-GW-01 *	Active Active *	
Edge Cluster	Edge-Cluster-01	(X) ✓
Additional Settings Route Distinguisher for VRF Gateways		
Tags	Tag (Required) Scope (Optional) + <small>Max 30 allowed. Click (+) to save.</small>	
<small>NOTE - Before further configurations can be done, fill out mandatory fields above (*), click 'Save' below.</small>		
INTERFACES ROUTING MULTICAST BGP ROUTE RE-DISTRIBUTION		
✓ Tier-0 Gateway TO-GW-01 is successfully created. Want to continue configuring this Tier-0 Gateway? YES (highlighted with red box) NO		
SAVE (highlighted with red box) CANCEL		

A continuación, debemos configurar las interfaces del Gateway Tier-0, para eso, como vemos en la siguiente imagen, demos expandir la sección “**INTERFACES**” y hacer click en **SET**.



En el siguiente paso debemos añadir las interfaces Uplink a este Gateway Tier-0. Se debe añadir una interfaz **por cada interfaz Uplink de cada nodo NSX Edge**. En nuestro caso tenemos dos nodos NSX Edge, cada uno con un único Uplink, por lo que debemos añadir dos interfaces.

Como vemos en la siguiente imagen, por cada interfaz añadida debemos ingresar los siguientes datos:

- **Nombre** de la interfaz
- **Dirección IP y Máscara**: Esta es la IP que utilizará el nodo NSX Edge en su interfaz Uplink para conectarse con los Gateway físicos.
- **Connected To (Segment)**: Seleccionamos el Segmento creado previamente en la Zona de Transporte VLAN. La Zona de Transporte usada en el Segmento y la asignada al nodo NSX Edge debe ser la misma.
- **Edge Node**: Seleccionamos el nodo NSX Edge al cual asociaremos esta interfaz. Como vemos en la imagen, hemos creados dos interfaces, una conectada a un nodo NSX Edge distinto (los nodos NSX Edge de este ejemplo tienen un único Uplink, pero un NSX Edge podría tener también dos Uplinks si lo consideran necesario en sus diseños).

Set Interfaces

Tier-0 Gateway TO-GW-01 #Interfaces 2

ADD INTERFACE

Name	Type	IP Address / Mask	Connected To(Segment)	Status
Uplink-1	External	192.168.100.2/24	Uplink-1	Success
Edge Node	sa-nsxedge-01			
Tags	0	PIM	Not Set	Disabled
ND Profile	default	URPF Mode	Strict	
Uplink-2	External	192.168.110.2/24	Uplink-2	Success
Edge Node	sa-nsxedge-02			
Tags	0	PIM	Not Set	Disabled
ND Profile	default	URPF Mode	Strict	

COLLAPSE ALL Search

VIEW STATISTICS

VIEW STATISTICS

REFRESH 1 - 2 of 2 Interfaces

Uplink-1 is located on sa-nsxedge-01. An SR is created on that edge node.

Uplink-2 is located on sa-nsxedge-02. Another SR is created on that edge node.

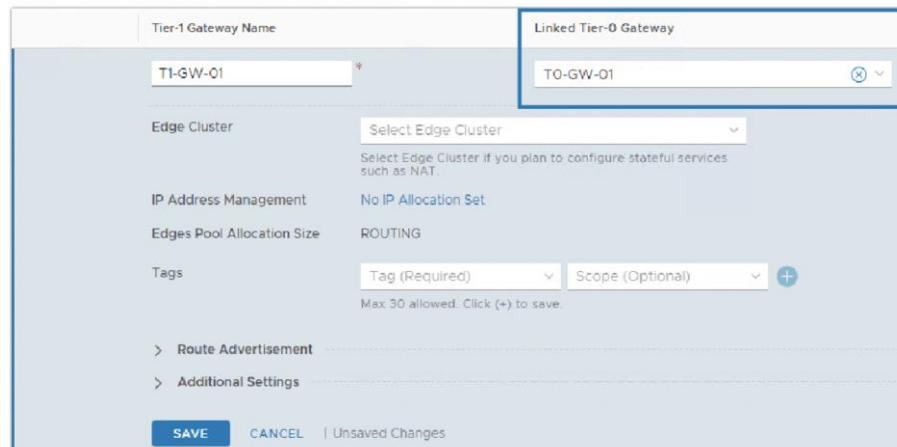
Con esto ya tenemos configurado los parámetros básicos del Gateway Tier-0.

CONECTAR GATEWAY TIER-1 Y TIER-0

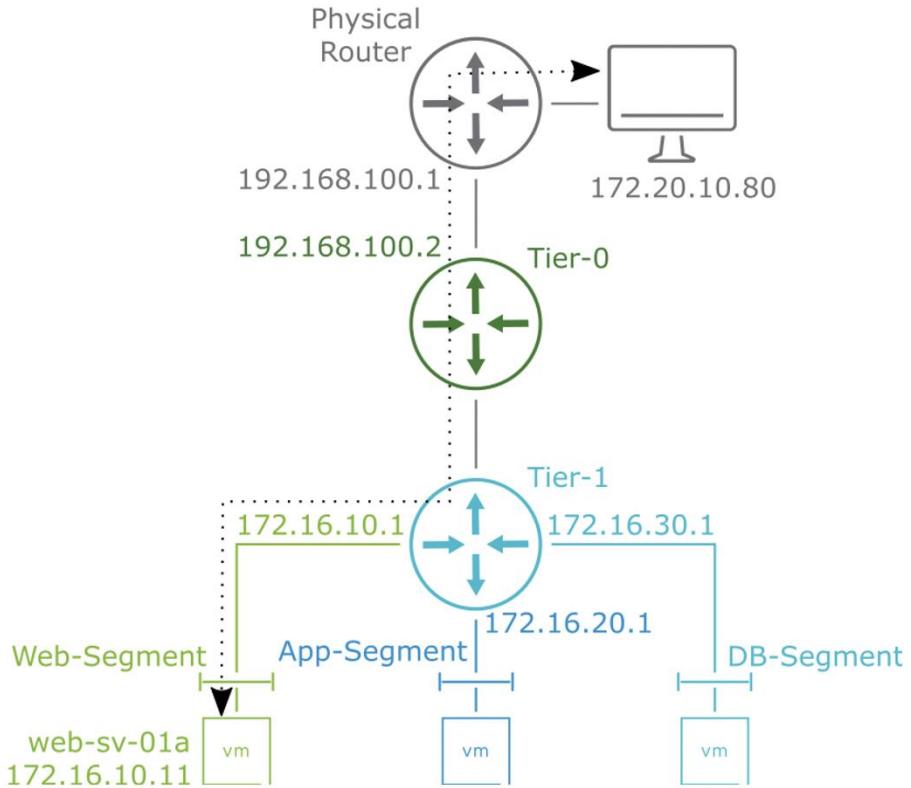
Ahora que ya hemos creado un Gateway Tier-1 y un Gateway Tier-0, los podemos conectar entre sí de manera que el Tier-1 pueda utilizar el Tier-0 como siguiente salto (hop) en la comunicación con las redes físicas.

Para poder conectar el Tier-1 con el Tier-0 debemos ir a **Networking > Tier-1 Gateways** y editar el Gateway Tier-1 que hemos creado previamente.

A continuación, y como vemos en la siguiente imagen, todo lo que necesitamos hacer es seleccionar el Gateway Tier-0 que creamos previamente, con lo cual estaremos automáticamente conectando ambos Gateways.



NSX-T automáticamente configurará rutas estáticas entre el Tier-1 y el Tier-0, de manera que todo el tráfico Norte-Sur generado por las VMs conectadas al Tier-1, será reenviado automáticamente al Gateway Tier-0 como siguiente salto (hop) en el proceso de routing.

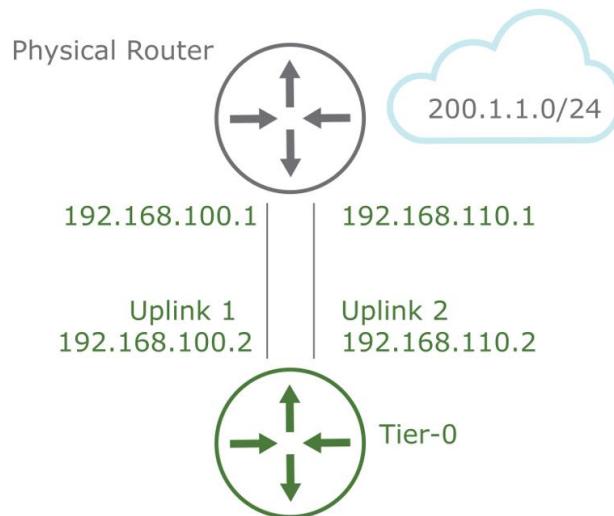


CONFIGURACION DE RUTAS

En NSX-T podemos configurar los Gateways para utilizar tanto rutas estáticas como rutas dinámicas, estas últimas solo disponibles para Gateways Tier-0. A continuación, veremos las opciones básicas de configuración de estas rutas.

RUTAS ESTÁTICAS

En NSX-T podemos configurar Rutas Estáticas tanto para Gateways Tier-0 como para Gateways Tier-1, aunque las veremos con más frecuencia configuradas en los Tier-0. En el ejemplo a continuación, lo que buscamos es crear dos Rutas estáticas que permitan al Tier-0 comunicarse con el segmento 200.1.1.0/24 según la imagen a continuación:



Para crear estas rutas, lo que debemos hacer es ir a **Networking > Tier-0 Gateways** y editar el Gateway Tier-0 que creamos previamente.

A continuación, expandimos la sección **ROUTING** y hacemos click en **SET** a la derecha de **Static Routes**.

The screenshot shows the Juniper Network Platform's 'Networking' tab selected. Under 'Tier-0 Gateways', a new gateway named 'TO-GW-01' is being configured. In the 'ROUTING' section, the 'Static Routes' table is highlighted with a blue box. It contains one entry: 'Static Routes' with a 'Set' button next to it.

Tier-0 Gateway Name	HA Mode	Linked Tier-1 Gateways
TO-GW-01	Active Active	

IP Prefix Lists	1
Static Routes	Set
Static Route BFD Peer	Set

A continuación, en la ventana **Set Static Routes** hacemos click en **ADD STATIC ROUTE** e ingresamos los siguientes datos:

- **Name:** Nombre de la ruta
- **Network:** Segmento de red que deseamos alcanzar utilizando esta ruta estática.
- **Next Hops:** Seleccionamos el o los siguientes Gateways físicos en la ruta que nos permitirán alcanzar el segmento 200.1.1.0/24. Podemos seleccionar múltiples hops/saltos por cada ruta estática.
 - **IP Address:** IP del Gateway físico que servirá como siguiente Hop/Salto en el proceso de routing a la red 200.1.1.0/24.
 - **Admin Distance:** Distancia administrativa o costo de esta ruta. Define la prioridad de la ruta cuando se cuenta con múltiples rutas para alcanzar un mismo destino. El valor va de 1 a 255, y las rutas estáticas por defecto tienen una Distancia Administrativa de 1 (la más alta después de las redes conectadas directamente al Gateway).
 - **Scope:** Nombre del Uplink del Tier-0 que utilizaremos para esta ruta estática.

Set Static Routes

Tier-0 Gateway TO-GW-01 #Static Routes 0

Name	Network	Next Hops	Status
to-200-net *	200.1.1.0/24	Set Next Hops Hop Count: 0	

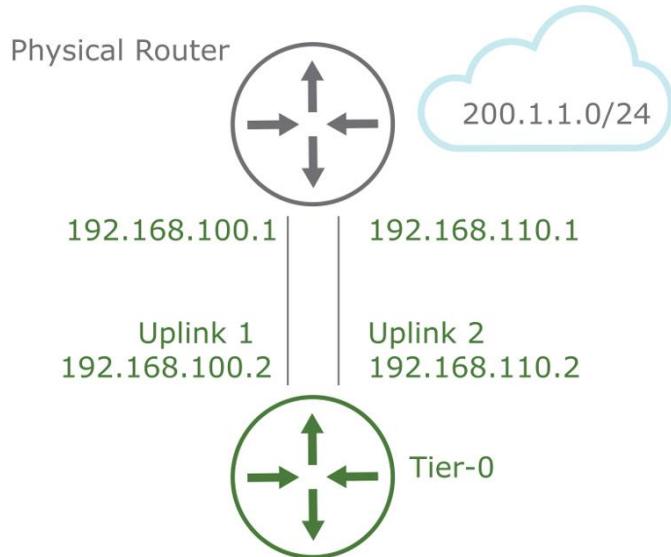
Set Next Hops

Tier-0 Gateway TO-GW-01 | Static Route to-200-net #Next Hops 1

IP Address	Admin Distance	Scope
192.168.110.1 X	1	Uplink-2

ADD STATIC ROUTE ADD NEXT HOP REFRESH CANCEL

Siguiendo el ejemplo que hemos venido detallando en este capítulo, nuestro Gateway Tier-0 cuenta con dos Uplinks, uno por cada NSX Edge en el Edge Clúster asociado al Tier-0.



Debido a esto, el Tier-0 cuenta con dos posibles caminos para alcanzar la red 200.1.1.0/24. De esta manera, si vamos a configurar rutas estáticas, debiéramos configurar dos rutas estáticas, una por cada Uplink provisto por los NSX Edge.

Ruta 1:

- Network: 200.1.1.0/24
- Next Hop: 192.168.100.1
- Scope: Uplink 1

Ruta 2:

- Network: 200.1.1.0/24
- Next Hop: 192.168.110.1
- Scope: Uplink 2

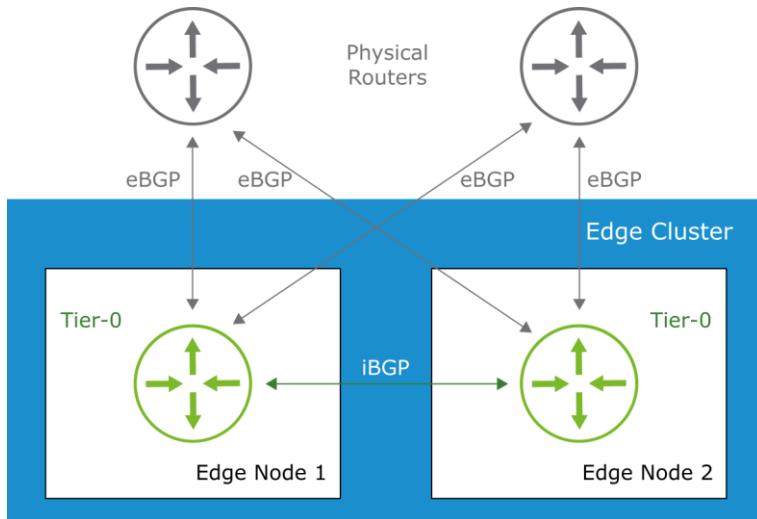
RUTAS DINÁMICAS

NSX-T también soporta el uso de Rutas Dinámicas con **BGP** en los Gateways Tier-0 como veremos a continuación.

Cuando habilitamos rutas dinámicas con BGP, lo que hacemos básicamente es establecer una sesión BGP entre el Tier-0 y un Gateway físico de manera que entre ellos puedan compartir información de rutas.

Como vemos en la siguiente imagen, un Tier-0 está asociado con un NSX Edge Clúster de dos nodos NSX Edge. Cada nodo NSX Edge puede tener 1 o 2 uplinks. En este ejemplo,

cada nodo NSX Edge cuenta con **dos Uplinks**, cada uno conectado a un Gateway físico distinto, por lo que, si deseamos habilitar BGP, se deben configurar **dos BGP Neighbors por cada nodo NSX Edge**.



En el ejemplo detallado en este capítulo, cada NSX Edge cuenta con solo un Uplink, por lo que al habilitar BGP debemos configurar solo **un BGP Neighbors por cada nodo NSX Edge**.

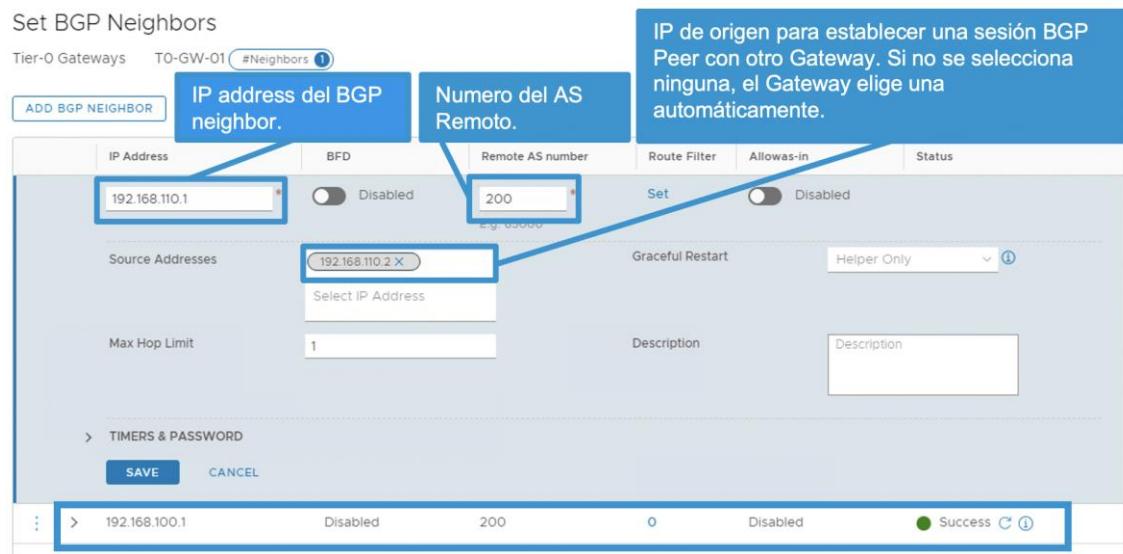
Para configurar BGP, lo que debemos hacer es ir a **Networking > Tier-0 Gateways** y editar el Gateway Tier-0 que creamos previamente, donde a continuación expandimos la sección **BGP**. En esta sección en primer lugar activamos la opción **BGP** que viene desactivada por defecto, para luego definir el **Local AS**, como vemos en la siguiente imagen.



El siguiente paso, como vemos en la imagen anterior, es ir a la sección **BGP Neighbors** y hacer click en **Set**, para poder configurar los pares BGP entre el Gateway Tier-0 y los Gateway físicos.

En la ventana Set BGP Neighbors, hacemos click en ADD BGP NEIGHBOR para añadir una nueva sesión BGP entre el Tier-0 y un Gateway físico. Debemos ingresar los siguientes datos:

- **IP Address:** Dirección IP del Gateway físico con el cual configuraremos el BGP Peer.
- **Remote AS Number:** AS configurado en el Gateway físico.
- **Source IP:** IP del Uplink en el Tier-0 que usaremos para conectarnos con el Gateway físico.



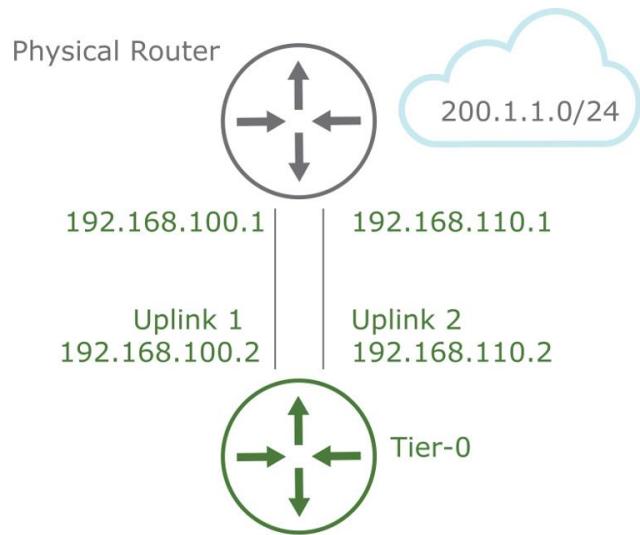
Debido a que este Tier-0 cuenta con dos Uplinks, uno por cada nodo NSX Edge en el Edge Clúster, debemos configurar dos BGP Neighbors:

BGP Neighbor 1:

- IP Address: 192.168.100.1
- Remote AS Number: 200
- Source Address: 192.168.100.2 (Uplink 1)

BGP Neighbor 2:

- IP Address: 192.168.110.1
- Remote AS Number: 200
- Source Address: 192.168.110.2 (Uplink 2)



REDISTRIBUCION DE RUTAS

Tanto en los Gateway Tier-1 como en los Tier-0 es necesario realizar algunas configuraciones adicionales para permitir la configuración ya sea mediante rutas estáticas como dinámicas.

ROUTE ADVERTISEMENT EN GATEWAYS TIER-1

En un Gateway Tier-1 debemos asegurarnos de que las redes definidas en cada segmento se encuentren disponibles para el Gateway Tier-0 al cual el Tier-1 está conectado. De esta forma, el Gateway Tier-0 también podría compartir esta información con otros gateways.

Como vemos en la imagen a continuación, podemos compartir múltiples tipos de rutas con el Gateway Tier-0, por ejemplo:

- Todas las rutas estáticas
- Rutas asociadas a las VIPs de los Load Balancers
- Todas las IP's configuradas en reglas NAT
- Todos los segmentos conectados directamente.

Tier-1 Gateway Name	Linked Tier-0 Gateway	#Linked Segments
T1-GW-01	TO-GW-01	0
Edge Cluster	Select Edge Cluster	Edges Set
IP Address Management	No IP Allocation Set	
Edges Pool Allocation Size	ROUTING	Enable Standby Relocation
Tags	Tag (Required) Scope (Optional)	+ Max 30 allowed. Click (+) to save.
Route Advertisement		
All Static Routes	<input checked="" type="checkbox"/>	All NAT IPs <input checked="" type="checkbox"/>
All DNS Forwarder Routes	<input checked="" type="checkbox"/>	All LB VIP Routes <input checked="" type="checkbox"/>
All Connected Segments & Service Ports	<input checked="" type="checkbox"/>	All LB SNAT IP Routes <input checked="" type="checkbox"/>
All IPSec Local Endpoints	<input checked="" type="checkbox"/>	Set Route Advertisement Rules

REDISTRIBUCIÓN DE RUTAS EN GATEWAYS TIER-0

Un último paso es la configuración de redistribución de rutas en el Tier-0, que permite redistribuir cualquier ruta aprendida den el Tier-0, o a través del Tier-0, hacia los Gateways físicos (upstream routers).

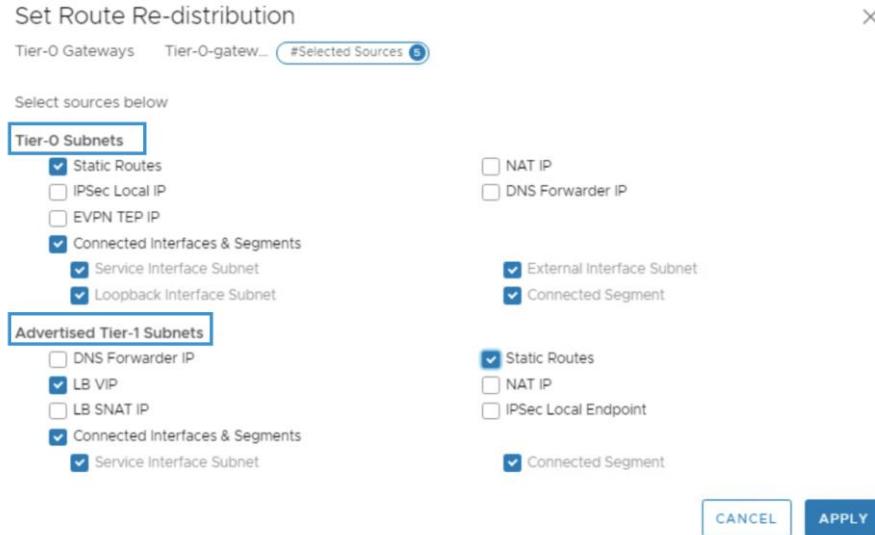
Para configurar esta redistribución, nos vamos a **Networking > Connectivity > Tier-0 Gateway** y editamos el Gateway Tier-0. Aquí habilitamos **Route Re-Distribution Status** y luego hacemos click en **Set**.



A continuación, creamos una nueva regla de redistribución de rutas haciendo click en **ADD ROUTE RE-DISTRIBUTION**. Ingresamos un nombre y hacemos click en **Set**.



A continuación, seleccionamos el tipo de rutas que deseamos compartir con los gateways/routers físicos. Aquí podemos compartir rutas que hayan sido aprendidas por el Tier-0, así como también rutas aprendidas a través del Tier-1.



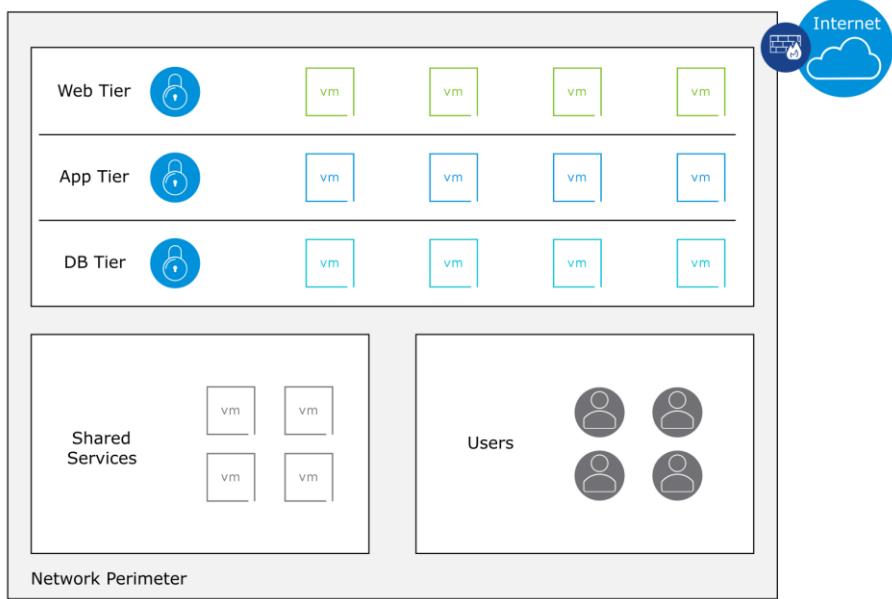
SEGURIDAD Y MICROSEGMENTACION

INTRODUCCION A LA MICROSEGMENTACION

Tradicionalmente, los datacenters enfrentan múltiples desafíos relacionados a la seguridad, muchas veces asociados a la falta de soluciones flexibles que se adapten mejor a la naturaleza dinámica de un Software Defined Data Center (SDDC).

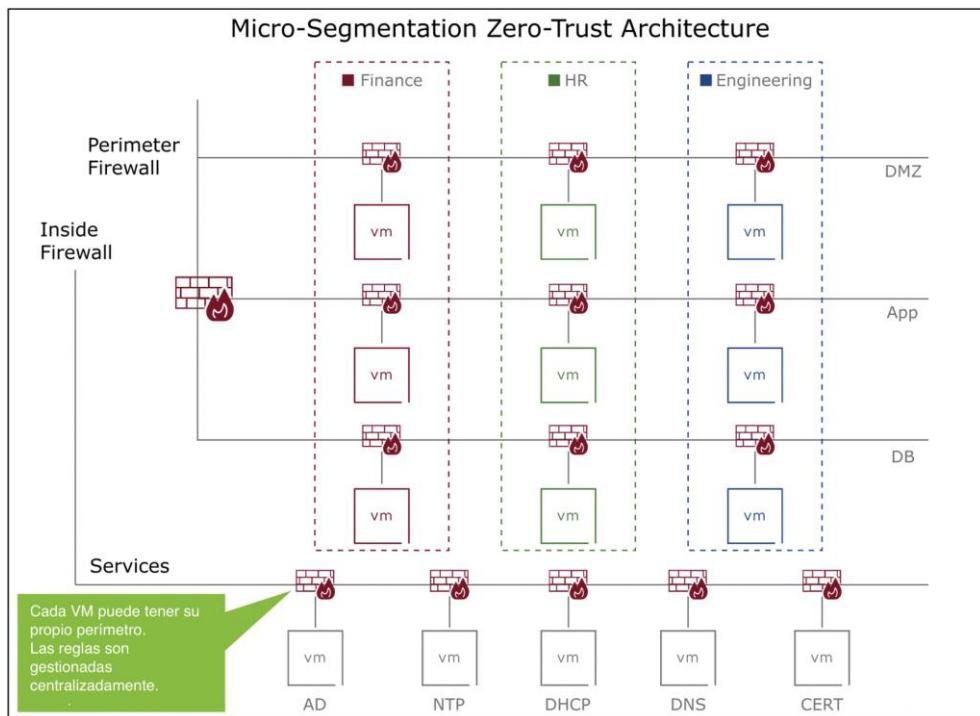
Sabemos que una protección perimetral no es suficiente en un Data Center para proveer seguridad. A nivel de seguridad debemos aplicar tantas capas como sea posible, para dificultar que un atacante tome control de nuestros servicios. E incluso, si un atacante logra penetrar en uno de nuestros servidores, un buen diseño a nivel de seguridad debiera evitar que el atacante pueda acceder a otros servicios en la red corporativa.

El objetivo de la Microsegmentación es proveer de tantas capas de seguridad como sea posible, de manera de crear microzonas de seguridad, o microsegmentos, donde podamos proveer aislación a distintos servicios.



Con la Microsegmentación podemos implementar un modelo de Cero Confianza al nivel más granular posible donde:

- Cada VM pueda tener
 - Firewall individual
 - Políticas de seguridad individuales
- Las políticas de seguridad puedan estar basadas en
 - Atributos de VMs
 - Atributos de red
 - Atributos de aplicaciones
 - Atributos de usuarios
- Los controles de seguridad puedan integrarse con soluciones de seguridad de terceros:
 - Service insertion para integración con servicios de seguridad de terceros, como Next Generation Firewall, o servicios IDS/IPS.
 - Antivirus sin agentes.

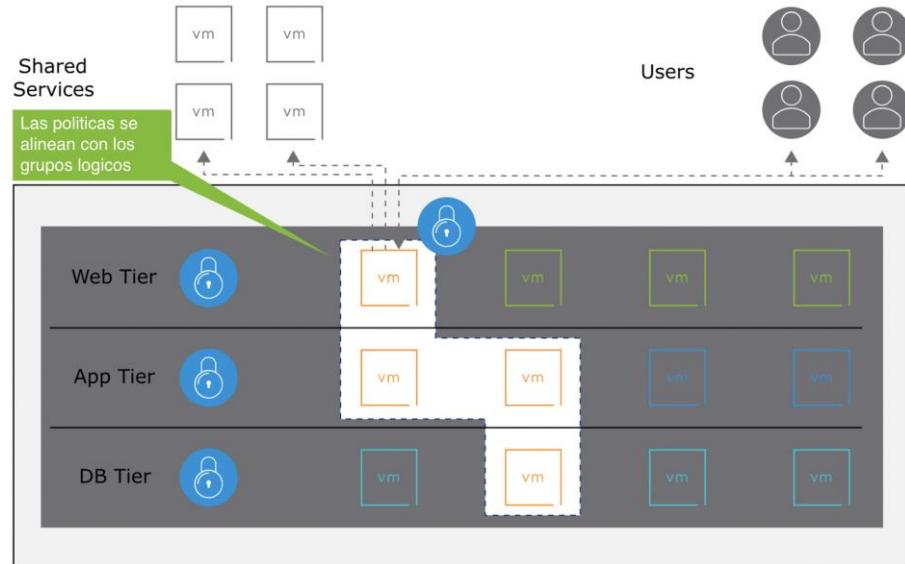


Como vemos en la imagen anterior, la Microsegmentación realiza varias funciones a nivel de seguridad:

- Divide lógicamente un Data Center en distintos segmentos de seguridad, incluso a nivel de VM individual.
- Define distintos controles y servicios de seguridad.
- Adjunta las políticas de firewall distribuido directamente a cada VM.

Por ejemplo, podríamos tener una aplicación que utiliza múltiples VMs para ofrecer un servicio específico. Esta aplicación cuenta con tres capas: Web, Aplicación y Base de Datos, y se cuenta con los siguientes requerimientos de seguridad:

- Los usuarios pueden interactuar solo con la capa Web, y nunca con la capa de Aplicación o Base de Datos.
- Las distintas VMs que componen el servicio se pueden comunicar entre sí, pero no pueden comunicarse con otras VMs, incluso con aquellas que se encuentren en el mismo segmento de red.
- La aplicación puede acceder a algunos servicios compartidos en la red, como servicios DNS, NTP, Active Directory, etc.



Como vemos en la imagen anterior, creamos una zona de seguridad donde solo las VM que pertenecen a la aplicación pueden comunicarse entre sí. Al mismo tiempo los usuarios pueden interactuar con la capa Web, y la aplicación puede interactuar con servicios compartidos como DNS, Active Directory, etc.

Las VM no pueden comunicarse con ninguna otra VM, incluso con VM que están en los mismos Segmentos.

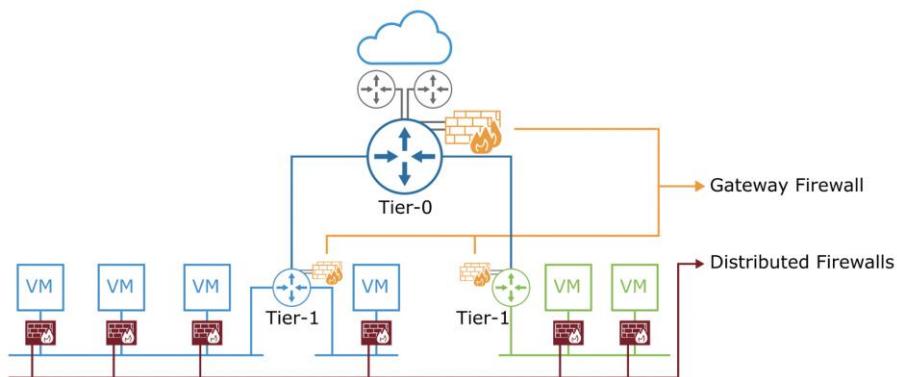
Microsegmentación permite implementar el modelo de Cero Confianza para aplicaciones en NSX-T, utilizando aislación, segmentación y servicios avanzados de seguridad. Entre los beneficios de la Microsegmentación podemos mencionar:

- Limita el movimiento lateral dentro del Data Center, es decir el movimiento entre VMs en el mismo Segmento.
- Minimiza el riesgo e impacto de las brechas de seguridad.
- Simplifica los flujos de red.
- Utiliza la infraestructura existente, siendo totalmente agnóstico e independiente de la topología de red.
- Provee de agilidad de negocio de manera segura.

INTRODUCCION AL DISTRIBUTED FIREWALL

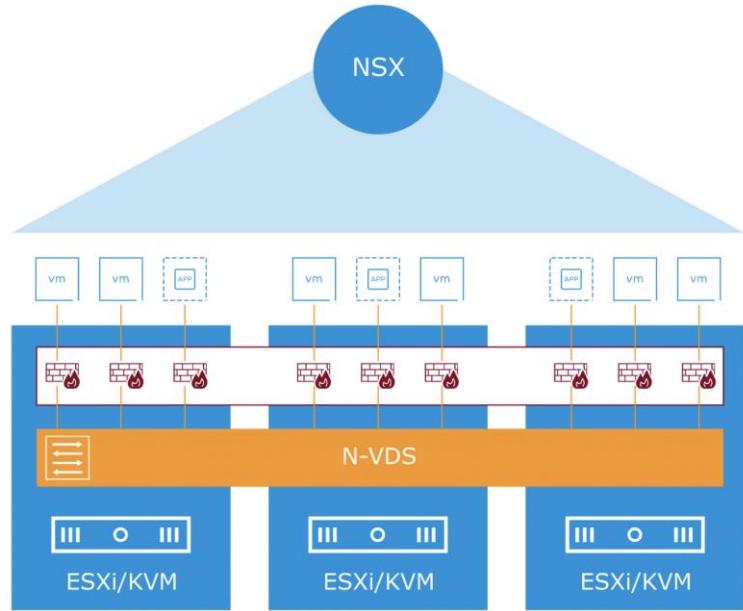
NSX-T incluye dos tipos de Firewall:

- Firewall Distribuido, optimizado para proteger el tráfico este-oeste.
- Firewall Gateway, optimizado para proteger el tráfico norte-sur.



El Firewall Distribuido (DFW) es un firewall stateful integrado en el kernel del hypervisor.

- Reside en el kernel del hypervisor, tanto en hosts ESXi y KVM, y se mantiene fuera del Sistema Operativo invitado en la VM.
- Controla el tráfico de entrada y salida en cada vNIC, con total independencia de la topología de red.



Las principales características del Firewall Distribuido son:

- Configuración centralizada a través de la interfaz de administración de NSX, o a través de APIs.
- Reglas de firewall Layer 2 Stateful
- Reglas de firewall Layer 3 Stateful y Stateless
- Reglas con contexto (Layer 7)
- Identity Firewall para VM Windows.

POLÍTICAS Y REGLAS

Una política de Firewall Distribuido es un conjunto de reglas de firewall aplicadas al tráfico este-oeste. Una política de Firewall incluye una o más reglas, que contienen instrucciones específicas para manejar distintos tipos de tráfico.

La interfaz de NSX permite agrupar las reglas de Firewall en diferentes categorías, las cuales además se van aplicando en el siguiente orden:

1. **Ethernet:** Reglas Layer 2, se aplican en primer lugar, antes que cualquier otra regla.
2. **Emergency:** Categoría con reglas de emergencia, usualmente para poner en cuarentena una o más VMs que se hayan visto expuestas a alguna amenaza de seguridad.
3. **Infrastructure:** Políticas para permitir el acceso a servicios globales y/o compartidos en el Datacenter, como Active Directory, DNS, NTP, etc.

4. **Environment:** Políticas para permitir el acceso entre distintas zonas de seguridad.

5. **Application:** Políticas granulares para permitir tráfico a nivel de aplicación.

The screenshot shows the Juniper Firewall Distributed Firewall interface. At the top, there are tabs for 'ALL RULES' and 'CATEGORY SPECIFIC RULES'. A warning message says 'Identity Firewall is disabled. Rules containing groups with identity entities (e.g. AD groups), will not be enforced.' Below this, there are several categories: 'ETHERNET (1)', 'EMERGENCY (0)', 'INFRASTRUCTURE (0)', 'ENVIRONMENT (0)', and 'APPLICATION (1)'. The 'APPLICATION (1)' category is expanded, showing 'L3 FW Policies' with sub-categories: 'Políticas temporales usadas para cuarentena', 'Políticas globales que aplican a AD, DNS, NTP, DHCP, backup y otros servicios de Management.', 'Políticas entre Zonas de Seguridad.', and 'Políticas granulares a nivel de aplicaciones.'. On the left, there's a sidebar with options like '+ ADD POLICY', '+ ADD RULE', 'CLONE', 'UNDO', 'DELETE', and '...'. A callout box points to the 'Add Rule' button with the text 'Cada categoría pueden tener sus propias políticas y reglas.' At the bottom right, there's a success message 'Success'.

Dentro de cada categoría, el Firewall Distribuido va inspeccionando las reglas en orden descendente, comenzando con la primera regla y avanzando con las siguientes reglas en la política hasta que existe una coincidencia con el tráfico que está siendo inspeccionado.

- Se pueden mover reglas hacia arriba o abajo dentro de una política para cambiar el orden en que estas son inspeccionadas y aplicadas.
- La primera regla que coincide con el tráfico inspeccionado es aplicada. Luego de una coincidencia, las reglas que vienen a continuación en la política son ignoradas.

The screenshot shows the Juniper Firewall Distributed Firewall interface. At the top, there are tabs for 'ALL RULES' and 'CATEGORY SPECIFIC RULES'. A warning message says 'Identity Firewall is disabled. Rules containing groups with identity entities (e.g. AD groups), will not be enforced.' Below this, there are arrows pointing from 'ETHERNET (1)', 'EMERGENCY (0)', 'INFRASTRUCTURE (0)', 'ENVIRONMENT (0)', and 'APPLICATION (3)' to a list of rules. The 'APPLICATION (3)' section is expanded, showing a table of rules. A callout box points to the '+ ADD POLICY' button with the text 'Política creada por el usuario.' Another callout box points to a rule in the table with the text 'Reglas creadas por el usuario dentro de la política'. The table columns include 'Name', 'ID', 'Sources', 'Destinations', 'Services', 'Profiles', 'Applied To', and 'Action'. Each row shows a rule like 'WEB TRAFFIC (1)' applied to 'DFW' with 'Allow Web Traffic' action.

En caso de que no exista ninguna coincidencia, es decir que ninguna regla de firewall coincide con el tráfico siendo inspeccionado, se aplica la regla “**Catch All**” o regla por defecto, la cual aplica a todo el tráfico que no ha sido explícitamente permitido o bloqueado por una regla de Firewall.



Cuando recién acabamos de instalar NSX-T, esta regla PERMITE por defecto todo el tráfico que entra o sale de las VM. La recomendación es que, una vez que se han diseñado e implementado todas las reglas de Firewall para permitir solo el tráfico requerido, la regla **Catch All** sea modificada para DENEGAR por defecto todo el tráfico.

CREAR GRUPOS DE SEGURIDAD

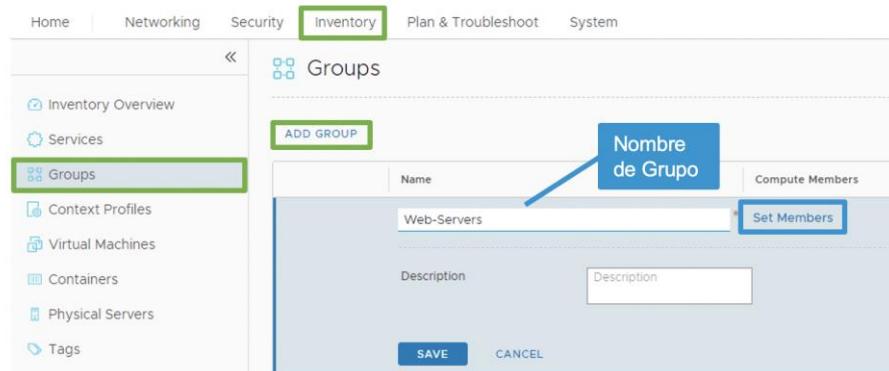
Antes de crear políticas y reglas de Firewall, podemos crear Grupos de Seguridad que luego utilizaremos como Origen o Destino en dichas reglas.

Un Grupo define un conjunto de objetos sobre los cuales aplicaremos una política de seguridad específica. Un grupo puede contener VMs, VIFs, Segmentos, Segment Ports, direcciones IP, direcciones MAC, grupos de Active Directory y servidores físicos.

Los Grupos pueden ser definidos de manera estática o dinámica:

- Con la inclusión estática, podemos definir manualmente los objetos que serán añadidos al Grupo de seguridad.
- Con la inclusión dinámica, podemos definir criterios que definen la membresía del grupo. Por ejemplo, podemos incluir todas las VM con determinado sistema operativo, o todas las VMs con un Tag específico, etc.

Para crear un Grupo nos dirigimos a **Inventory > Groups** y hacemos click en **ADD GROUP**, como vemos en la siguiente imagen:



A continuación, ingresamos **el nombre del Grupo**, y hacemos click en **Set Members** para definir los miembros del grupo.

Para añadir un criterio para inclusión dinámica, nos dirigimos a **Membership Criteria** como se ve en la siguiente imagen, y hacemos click en **ADD CRITERIA** para especificar la condición para añadir objetos al grupo. En este ejemplo, añadiremos cualquier VM cuyo nombre contenga la palabra “**web**”

Select Members | Web-Servers

Add Compute Members either by creating or by directly adding them. You can also add Identity members separately. Identity members intersect with the Compute members to define effective membership of the group.

Membership Criteria (1) Members (0) IP Addresses (0) MAC Addresses (0) AD Groups (0)

+ ADD CRITERIA Maximum: 5 Criteria

Criteria 1

Virtual Machine Name Contains web

Para añadir objetos de manera estática, hacemos click en **Members** y luego seleccionamos una categoría de objetos. A continuación, simplemente seleccionamos los objetos que añadiremos al Grupo.

Select Members | Web-Servers

Add Compute Members either by creating or by directly adding them. You can also add identity members separately. Identity members intersect with the Compute members to define effective membership of the group.

Membership Criteria (0) Members (0) IP Addresses (0) MAC Addresses (0) AD Groups (0)

Filter by Object Name

Category Virtual Machines (selected: 0) ▾

- Groups (selected: 0)
- Segments (selected: 0)
- Segment Ports (selected: 0)
- VIFs (selected: 0)
- Virtual Machines (selected: 0)**
- Physical Servers (selected: 0)

T1-DB-01 T1-L2VPN-01 T1-Web-01 T1-Web-02

View Details

Selección de miembros puede estar basada en Grupos, Segmentos, Puertos, VIFs, VMs y Servidores físicos.

REFRESH 1 - 9 of 9 Objects

CREAR REGLAS DE FIREWALL DISTRIBUIDO

A continuación, veremos cómo crear reglas en el Firewall Distribuido, el cual es un proceso bastante sencillo.

Las reglas son simplemente un conjunto de criterios que el Firewall utiliza para evaluar los flujos de tráfico. Estas reglas contienen instrucciones que determinan si un paquete debe ser permitido o bloqueado.

Como vemos en la siguiente imagen, en cada regla tenemos una serie de campos que podemos configurar.

The screenshot shows the 'DISTRIBUTED FIREWALL' interface. At the top, there are tabs for 'ALL RULES' and 'CATEGORY SPECIFIC RULES'. A warning message says 'Identity Firewall is disabled. Rules containing groups with identity entities (e.g. AD groups), will not be enforced.' On the right, a blue callout box says 'Parámetros configurables por el usuario' (User-configurable parameters). Below the tabs are buttons for '+ ADD POLICY', '+ ADD RULE', 'CLONE', 'UNDO', 'DELETE', and '...'. The main table has columns: Name, ID, Sources, Destinations, Services, Profiles, Applied To, and Action. A row for 'WEB TRAFFIC' is selected, showing 'Allow Web Traffic' with ID 1005, applied to DFW, using 'Web-Servers' and 'App-Servers' profiles, and the 'HTTP' service. The 'Action' column shows 'Allow' with a toggle switch set to 'On'.

Source/Destination: Cuando se define el origen y destino de una regla de Firewall, se puede utilizar una dirección IP, una MAC Address o un objeto (Grupo de seguridad, Segmento, etc). Si no se especifica ningún origen o destino, entonces se utiliza “Any” y la regla será aplicada a cualquier origen y/o destino.

Services: Al crear una regla de Firewall se debe especificar qué servicios serán permitidos o bloqueados. Se pueden especificar uno o más servicios en una regla. Un servicio es una combinación de puerto y protocolo. Existen múltiples servicios por defecto en NSX-T, básicamente para los servicios más conocidos como SSH, DNS, HTTP, etc. Adicionalmente, un usuario podría crear un **Servicio personalizado**, simplemente definiendo un nombre de servicio, protocolo y puerto.

The screenshot shows the 'Set Services' dialog for the rule 'Allow Web Traffic'. It lists 'Services (1)' and 'Raw Port-Protocols (0)'. Under 'Services', there is a list of entries: 'HTTP X' (selected), 'Active Directory Server', 'Active Directory Server UDP', 'AD Server', and 'CIM-HTTP'. A green callout box says 'Usuario puede definir nuevos servicios.' (User can define new services.) A blue callout box says 'Un servicio clasifica el tráfico basado en la combinación de puerto y protocolo.' (A service classifies traffic based on port and protocol combination.)

Profiles: Opcionalmente se puede aplicar un Context Profile a una regla de Firewall, de manera de crear una regla de Firewall Layer 7.

Applied To: Permite definir el alcance de aplicación de la regla de Firewall. Este atributo permite optimizar la utilización de recursos en los hosts ESXi y KVM cuando se procesan las reglas de Firewall.

Por defecto, cada regla se aplica a cada vNIC en NSX-T, lo cual es poco óptimo y genera un mayor uso de recursos en el hipervisor. Como buena práctica, se recomienda definir el campo **Applied To** para aplicar la regla solo a determinadas VMs, utilizando un Grupo o un Segmento.

The screenshot shows the NSX-T Firewall Rules interface. At the top, there are tabs for INFRASTRUCTURE (0), ENVIRONMENT (0), and APPLICATION (3). Below these are tabs for DO, DELETE, and three dots. Underneath are tabs for Sources, Destinations, Services, Profiles, and Applied To (which is highlighted with a green border). The main table has columns for Action, Success, and several icons. Two rows are visible: one for 'DFW' with 'Allow' action and another for 'App-Servers' and 'Web-Servers' also with 'Allow' action. A blue callout box points to the 'DFW' row with the text: 'Se pueden aplicar reglas al Distributed Firewall o a grupos de objetos.' A green callout box points to the 'Applied To' tab with the text: 'The Applied To define el alcance de la aplicación de cada regla.'

Action: Este campo permite definir la acción que realizará la regla de Firewall

- **Allow:** El tráfico especificado en los campos Source, Destination y Service será permitido.
- **Drop:** El tráfico especificado en los campos Source, Destination y Service será denegado. Hacer Drop a un paquete es una acción silenciosa, donde no se notifica al Origen o Destino que el tráfico ha sido bloqueado.
- **Reject:** El tráfico especificado en los campos Source, Destination y Service es rechazado. Esta acción deniega un paquete, pero envía adicionalmente un mensaje al emisor del paquete, indicando que el tráfico ha sido denegado.

The screenshot shows the NSX-T Firewall Rules interface. At the top, it says 'APPLICATION (2)'. Below are tabs for Services, Profiles, and Applied To. The main table has columns for Action, Success, and icons. One row is visible for 'HTTP' with 'Allow' action. A blue callout box points to the 'Allow' action with the text: 'Allow', 'Drop', 'Reject'. A green callout box points to the 'Action' column with the text: 'The Action defines the operation to be performed on the traffic matching the rule conditions.'

Como último paso, simplemente hacemos click en PUBLISH, para publicar las reglas recientemente creadas/modificadas.

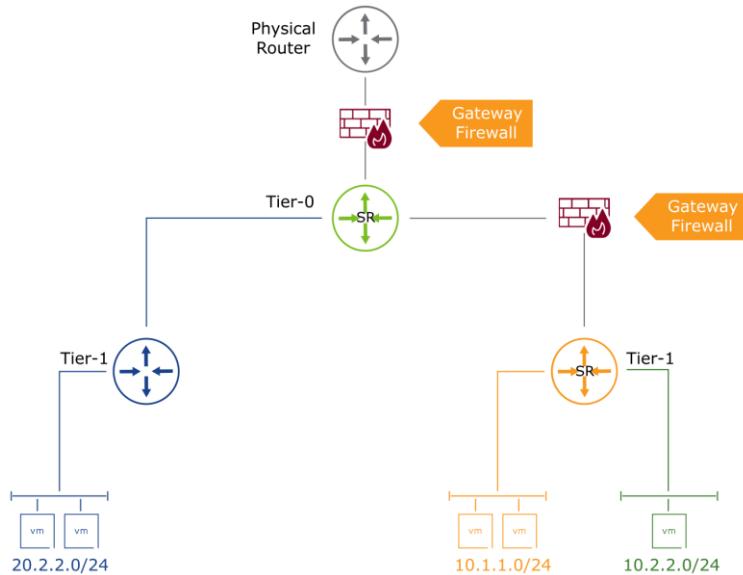
INTRODUCCION AL GATEWAY FIREWALL

El Gateway Firewall es también conocido como Firewall Perimetral, y protege el tráfico desde y hacia ambientes físicos:

- Se aplica en los uplinks de los Gateway Tier-0 o Tier-1
- Un NSX Edge Clúster debe ser asociado al Tier-0 o Tier-0 de manera obligatoria, de manera de permitir el uso del servicio de NSX Gateway Firewall.

El Gateway Firewall tiene las siguientes características:

- Firewall stateful para proteger el tráfico norte-sur, generalmente usado como Firewall perimetral
- Totalmente independiente del Firewall Distribuido, tanto desde el punto de vista de la configuración como de la aplicación de políticas.
- Está soportado tanto en gateways Tier-0 o Tier-1
- Es un servicio centralizado, por lo que requiere un componente SR del router. Por eso, el uso de un NSX Edge Clúster es obligatorio, el cual además será utilizado en configuración Activo-Standby.



CREAR REGLAS DE GATEWAY FIREWALL

El Gateway Firewall, al igual que el Firewall Distribuido, incluye categorías predefinidas para organizar las reglas, las cuales se pueden encontrar bajo **ALL SHARED RULES** donde las reglas a través de todos los Gateway Firewall estan visibles.



Las categorías definen además el orden en que las reglas serán aplicadas:

1. **Emergency**: Usadas para crear reglas de cuarentena.
2. **System**: Reglas creadas automáticamente por NSX-T y que son específicas para el tráfico interno del plano de control, como regla BFD o VPN.
3. **Pre-Rules**: Reglas aplicadas globalmente a todos los nodos Gateway Firewall.
4. **Local Gateway**: Reglas aplicadas específicamente a un nodo Gateway Firewall.
5. **Auto Service Rules**: Reglas aplicadas al plano de datos.
6. **Default**: Reglas que definen el comportamiento por defecto del Gateway Firewall, equivalente a la regla **Catch-All** del Firewall Distribuido.

Dentro de cada categoría, el Gateway Firewall va inspeccionando las reglas en orden descendente, comenzando con la primera regla y avanzando con las siguientes reglas en la política hasta que existe una coincidencia con el tráfico que está siendo inspeccionado.

- Se pueden mover reglas hacia arriba o abajo dentro de una política para cambiar el orden en que estas son inspeccionadas y aplicadas.
- La primera regla que coincide con el tráfico inspeccionado es aplicada. Luego de una coincidencia, las reglas que vienen a continuación en la política son ignoradas.

Si deseamos crear una regla específica para un Gateway Firewall, nos dirigimos a **GATEWAY SPECIFIC RULES** y seleccionamos el Gateway sobre el cual crearemos la regla.

Selección el Tier-0 o Tier-1 donde aplican las políticas de Gateway Firewall

+ ADD POLICY

Una política por defecto existe para permitir todo el tráfico

A continuación, como vemos en la imagen anterior, podemos crear una nueva Política, que recordemos es básicamente un grupo de reglas. Existe además una política por defecto que permite todo el tráfico, y que luego se recomienda modificar para denegar todo el tráfico que no haya sido específicamente permitido por otras reglas.

Finalmente, y al igual que con las reglas del Firewall Distribuido, podemos crear nuevas reglas en el Gateway Firewall ingresando los siguientes datos:

- Name
- Sources
- Destinations
- Services
- Profiles
- Applied To
- Action

Enable Logging For All Rules
Disable Logging For All Rules
Enable All Rules
Disable All Rules
Delete Policy
Add Rule
Add Policy Above
Add Policy Below

Añadir nueva regla a la política.

Reglas son aplicadas en orden descendente desde la primera regla hacia abajo

Como último paso, simplemente hacemos click en PUBLISH, para publicar las reglas recientemente creadas/modificadas.

CONCLUSION

Con esto completamos los pasos básicos en la implementación de una solución NSX-T 3.0 plenamente funcional. En este capítulo hemos visto como:

- Desplegar los componentes de la arquitectura NSX-T 3.0
- Como preparar los nodos de transporte
- Como desplegar segmentos Layer 2
- Como desplegar Gateways Tier-1 y Tier-0
- Como desplegar políticas de Firewall.

Hosted Private Cloud

Un Software-Defined Datacenter en el que poder confiar



Alta seguridad

Servicios de seguridad extendidos con NSX, incluido firewall, balanceador de carga y reglas de seguridad a través de vSphere



Recursos escalables

vSphere + Licencia Enterprise Plus incluidos para todas sus VMs + hosts en entorno totalmente dedicado



Almacenamiento seguro

Diseñado con hardware Intel de alta gama, permite la integración con vSAN para un rendimiento óptimo

Lo mejor de la tecnología VMware

- Construido sobre la misma plataforma que utiliza actualmente en sus instalaciones on-premises, incluyendo vSphere, vCenter, NSX y vSAN
- Basado en el Software-Define Datacenter (SDDC) de VMware



Datacenter «as a service» con outsourcing de toda la infraestructura



Extensión de su infraestructura on-premises al cloud



Implementación de un plan de «disaster recovery»



Consolidación de un perímetro multi-datacenter mundial



Transferencia de cargas en una infraestructura hiperescalable

Conectividad

Interconecte sus recursos cloud y on-premises en una red privada segura gracias a OVHcloud Connect, con una red mundial con ancho de banda garantizado y tráfico ilimitado.

Un ecosistema integrado

Una amplia gama de opciones compatibles, que incluyen backup, replicación y tecnología de «disaster recovery» con los servicios de nuestros partners Zerto, Veeam y VMware.

Para más información, visite ovhcloud.com

@OVHcloud_ES

Capítulo 5

HORIZON 7: ACCESO EXTERNO Y CONSOLA HTML5



Ricard Ibáñez

@ricardibanez

HORIZON 7: ACCESO EXTERNO Y CONSOLA HTML5

INTRODUCCIÓN

Este capítulo es una extensión del capítulo 12, *VDI con Horizon View* del libro escrito por la comunidad de vExperts en Español *VMware por vExperts*, donde se hablaba de la plataforma de escritorios virtuales VMware Horizon.

En el momento de redactar el libro, para poder configurar y operar la plataforma de VMware Horizon se usaba una consola web (Flex) basada en Flash, esto no sería un problema si no fuese que es una consola lenta y poco amigable. En tan solo 1 año, VMware ya ha completado la versión beta de su consola en HTML5 donde podemos encontrar todas las funciones de la consola Flash y algunas nuevas que veremos en este capítulo.

Una de las partes básicas, que quedó pendiente comentar sobre la infraestructura de VMware Horizon es la conectividad externa a la plataforma. Este punto de la configuración se puede implementar de varias maneras, desde facilitar el acceso mediante VPN a nuestra infraestructura, hasta montar servidores de acceso publicados a internet.

CONSOLA HTML5

A finales de 2018, VMware lanzaba la versión 7.7 de VMware Horizon, la cual incorporaba la primera fase beta del portal de administración basado en HTML5. Como cabía esperar, al llamarse beta, no nos permitía substituir por completo la administración, sino que se incorporaría funciones progresivamente en cada nueva versión de VMware Horizon 7.

En la última versión, a fecha de la escritura de este capítulo, VMware Horizon 7.11, ya disponemos de una versión completa de la nueva consola con una interfaz moderna y ágil para operar nuestra plataforma.

La distribución de opciones de gestión y configuración respecto a la consola Flex es muy similar por lo que no perderemos la costumbre si venimos trabajando con la antigua consola. Cierto es, que encontraremos cosas nuevas como el apartado de JMP (Just-in-time Management Platform) o la consola de soporte técnico.

Vamos a repasar cada uno de los aspectos fundamentales de esta nueva consola para poder sacar el máximo partido a nuestra plataforma VMware Horizon.

CONFIGURACIÓN

El apartado de configuración es uno de los más importantes en el momento de arrancar nuestra plataforma, ya que nos permitirá controlar las configuraciones de nuestros Connection Servers, conectar el vCenter Server, credenciales de Instant Clone, etc. En este apartado veremos pocas novedades respecto a la anterior salvo el ya mencionado JMP.

Servidores

vCenter Server	Tipo de Horizon Composer	Recuperación de espacio	Acelerador de almacenamiento	Aproximamiento
vcenter01.vsan.infra.es	Horizon Composer Server / Independiente			

El primer apartado, encontramos la gestión de vCenter Server donde añadimos y gestionamos la autenticación y las conexiones entre VMware Horizon y vCenter Server.

También podemos gestionar tanto los Connection Server como los Security Gateway, que los veremos con más detalle en el otro apartado del capítulo.

Cuentas de dominio de Instant Clone

Podemos dar de alta las credenciales con las que nuestro despliegue de Instant Clones se conecta a nuestro Active Directory.

Licencia y uso del producto

Configuración de la licencia del producto para poder operar con las distintas funciones de nuestra plataforma de VMware Horizon. También podemos ver las estadísticas de uso, tanto las conexiones simultáneas en tiempo real como los máximos alcanzados en la plataforma.

Configuración global

The screenshot shows the 'Configuración global' (Global Configuration) page. On the left is a sidebar with navigation links: Supervisor, Usuarios y grupos, Asignaciones RDP, Inventario, Configuración (selected), Servidores, Cuentas de dominio de Instant Clone, Licencia y uso del producto, Configuración global (selected), Máquinas registradas, Administradores, Arquitectura cloud AD, Configuración de eventos, Directivas globales, and Configuración de Java. The main content area has tabs: Configuración general (selected), Configuración de seguridad, and Configuración de restricciones de cliente. Under 'Configuración general', there are sections for: Tiempo de espera de la sesión de View Administrator (360 minutos), Desconectar usuarios de forma forzada (1.440 minutos), Configurar Single Sign-On (SSO), Activación automática (Desactivado), Conexión dependiente del cliente, Para clientes que admiten aplicaciones, Desconectar al dejar de usar el teclado y el mouse, desconecte las aplicaciones y desactive las credenciales SSO, Nombre, Otras clientes, Desectar credenciales SSO (15 minutos), Mensaje previo al fin de sesión (Si), Mostrar una advertencia antes del cierre de sesión forzada (No), Monitorear monitores Windows Server (No), Bloquear credencial al cerrar la pestaña para HTML Access (Si), Ocultar la lista de dominios en la interfaz de usuario del cliente (Si), and Ocultar lista de dominios (No). At the bottom are 'Cancelar' and 'Aceptar' buttons.

Permite establecer la configuración general de nuestra plataforma como del mismo modo que en el portal Flex.

The screenshot shows the 'Configuración de restricciones de cliente' (Client Restrictions Configuration) dialog box. It contains a message: 'Introduzca la versión de Horizon Client con el formato X.Y.Z (por ejemplo, 4.5.0). Se debe usar la versión 4.5.0 o una posterior de Horizon Clients, o bien la versión 4.8.0 o una posterior de Horizon Client para Chrome. Las versiones anteriores de Horizon Clients no podrán conectarse a escritorios ni aplicaciones.' Below this are input fields for: Horizon Client para Windows, Horizon Client para Linux, Horizon Client para Mac, Horizon Client para iOS, Horizon Client para Android, Horizon Client para UWP, Horizon Client para Chrome, and Horizon Client para HTML Access. There is also a checkbox 'Bloquear clientes adicionales'. At the bottom are 'Cancelar' and 'Aceptar' buttons.

Un apartado nuevo es la configuración de restricciones de cliente, donde podemos establecer versiones mínimas u obligatorias para que nuestros usuarios se conecten a la plataforma.

Máquinas registradas

The screenshot shows the 'Máquinas registradas' (Registered Machines) page. On the left is a sidebar with navigation links: Supervisor, Usuarios y grupos, Asignaciones RDP, Inventario, Configuración (selected), Servidores, Cuentas de dominio de Instant Clone, Licencia y uso del producto, Configuración global, Máquinas registradas (selected), Administradores, and Configuración de Java. The main content area has tabs: Hosts de RDS (selected) and Otros. It includes buttons: Editar, Eliminar, and Nuev... (with a dropdown menu). A search bar and filter button are at the top right. The table below has columns: Nombre DNS, Tipo, Grilla de RDS, Número máxim..., Sesiones, Versión del age..., Habilitado, and Estado. A note says 'No hay registros disponibles'. At the bottom are 'Cancelar' and 'Aceptar' buttons.

Disponemos de la gestión tanto de los hosts RDS registrados en nuestra plataforma como equipos dedicados para conexiones remotas.

Administradores

Configuración de Administradores, privilegios de gestión sobre la plataforma y la gestión de los grupos de acceso.

Arquitectura Cloud Pod

Nos permitirá activar y configurar nuestra arquitectura de Cloud Pods. Esta funcionalidad ya existía en la anterior consola y permite conectar distintas infraestructuras de VMware Horizon, permitiendo a un usuario conectarse a tu escritorio virtual independientemente de si la conexión la realizar en una infraestructura o en otra.

Configuración de eventos

Al igual que en la consola Flex, este apartado permite configurar el sistema de eventos y Logs.

Directivas globales

The screenshot shows the 'Global Directives' configuration page. It includes sections for 'Redirección multimedia (MRR)', 'Acceso USB', and 'Aceleración de hardware PCoIP'. Each section has a 'Denegar' (Deny) button and a 'Permitir' (Allow) button. A note at the bottom states: 'Permitir - Prioridad media' (Allow - Medium priority).

Idéntico a la antigua consola, podemos establecer unas directivas globales de permisos sobre Acceso USB, Redirección Multimedia y PCoIP.

Configuración de JMP

The screenshot shows the 'JMP Configuration' interface. It features a sidebar with 'Supervisar', 'Usuarios y grupos', 'Asignaciones JMP' (selected), 'Inventario', and 'Configuración'. The main area displays a welcome message: 'Le damos la bienvenida a la Just-in-Time Management Platform. Esta plataforma le permite asignar y controlar de forma adecuada los escritorios desde una única ubicación.' Below this is a note: 'Para empezar, tendrá que proporcionar la URL del servidor en el que se encontrará JMP. Solo se puede usar una URL de JMP cada vez.' At the bottom is a blue 'Agregar JMP Server' button.

Este apartado es completamente nuevo, JMP por sí solo, no es una tecnología, sino que es una manera de trabajar con 3 tecnologías de VMware Horizon que son Instant Clone, App Volumes y DEM (Dynamic Environment Manager), aunando el aprovisionamiento de escritorios virtuales de una manera sencilla desde un único punto.

Si estamos acostumbrados a trabajar con cada una de estas tecnologías, nos damos cuenta de que, para aprovisionar escritorios, hay que crear la asignación en la consola de Horizon, si debemos aprovisionar aplicaciones hay que entrar en la consola de App Volumes y si además tenemos que configurar el perfil del usuario, debemos entrar en la consola de DEM. Todo este proceso se simplifica con JMP, realizando la asignación de escritorios, aplicaciones y las configuraciones de usuario desde un solo asistente de asignación.

Proceso de configuración - <https://techzone.VMware.com/quick-start-tutorial-VMware-horizon-jmp-integrated-workflow#926454>

INVENTARIO

Este menú tiene las mismas opciones que veníamos viendo en el antiguo portal y es unos de los paneles más usados para cualquier administrador de VMware Horizon.

Desde este panel veremos que se pueden definir y asignar Pools de escritorios, crear granjas de servidores RDS y asignar las aplicaciones RDS.

Escritorios

The screenshot shows a list of desktop groups under the 'Escritorios' (Desktops) section. The columns include 'Nombre para mostrar' (Name to display), 'Tipo' (Type), 'Origen' (Origin), and 'Asignación de usuarios' (User assignment). The groups listed are:

- Escritorio
- Escritorio
- Escritorio
- Escritorio Oficina
- Escritorio Virtual
- Escritorio Centro
- Escritorio Biología Molecular

Disponemos de una consola idéntica, en este nuevo portal, para la gestión de los Pools de escritorios virtuales, podemos definir los Pools en modo automático o dedicado, generados con View Composer, Instant clone o VM completa, en definitiva, nos permite parametrizar nuestro Pool mediante el mismo asistente de creación que en la anterior consola.

This screenshot shows the details for the 'Oficina' desktop group. It includes sections for General settings (Type: Grupo de escritorios automatizado, Origin: vCenter (con vinculado)), Machine assignments (Nombre de máquina: Escritorio Oficina, Status: Habilitado, Sesiones: 32, Número de máquinas: 38), Machine status (Estado de máquinas: Disponible 32), and vCenter Server (Nombre del servidor: refalinovis-refalin, IP: 192.168.1.20, Tamaño de disco persistente: 16.240 MB).

Si accedemos dentro de uno de los Pool podemos observar toda la información disponible, así como un listado de los escritorios generados, las sesiones de los usuarios, autorizaciones, eventos generados en este Pool, las directivas asignadas o modificadas para este Pool en concreto y un control de las tareas que se ejecutan en el Pool.

This screenshot shows the list of machines within the 'Oficina' desktop group. It displays columns for Nombre DNS (Name), Usuario (User), Host, and Acciones (Actions). The machines listed are all named 'refalinoficina' followed by a number (e.g., refalinoficina1, refalinoficina2, etc.), with their respective host names and user assignments.

Las tareas que podemos lanzar desde esta consola son las de Recomponer todo el Pool o un escritorio en concreto, reiniciar un escritorio o un grupo de escritorios de este Pool, cambiar las autorizaciones o incluso desvincular discos persistentes de los escritorios virtuales.

Aplicaciones

The screenshot shows a list of application groups under the 'Aplicaciones' (Applications) section. The columns include 'Nombre para m.', 'Grupo o grupo', 'Versión', 'Editor', 'Acceso directo', 'Preinicio', and 'Modo de si'. The applications listed are:

- Calculator
- Internet Explorer
- NotePad
- Paint
- WordPad

En el apartado de aplicaciones, crearemos las asignaciones de aplicaciones previamente instaladas sobre granjas de RDS o de grupo de escritorios, permitiendo a los usuarios asignados hacer uso de ellas y poder revocar ese acceso cuando fuese necesario.

Granjas

Para poder asignar aplicaciones primero debemos crear las granjas de RDS, por lo que podremos crear nuestras granjas mediante el mismo asistente que veníamos viendo en la antigua consola Flex.

Máquinas

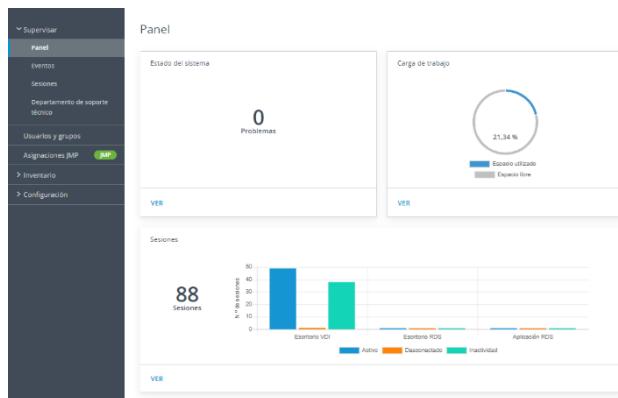
Si necesitamos tener a mano un inventario de todos los escritorios virtuales, Hosts de RDS de nuestras granjas incluso otros equipos que disponen del agente de Horizon, lo podemos ver desde esta pantalla.

Podremos realizar tareas básicas sobre los escritorios virtuales como quitar una asignación, reiniciarlo, poner en mantenimiento, desconectar a un usuario o incluso eliminar un escritorio virtual.

SUPERVISAR

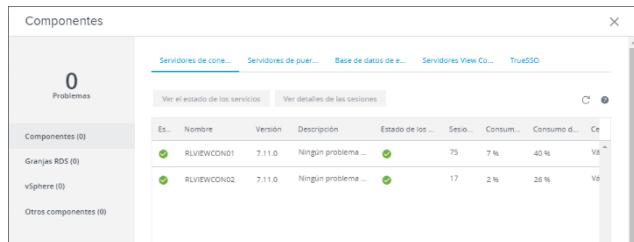
Cuando nuestra plataforma ya está completamente operativa, es necesario conocer el estado de esta misma y al igual que el portal Flex tendremos datos operativos básicos para saber si todo funciona como esperamos.

Panel



Este es un panel muy sencillo para que visualmente veamos donde tenemos que centrar nuestra atención. Está dividido en tres apartados.

La parte de las Sesiones nos da una visión general de las conexiones activas que tenemos en nuestra plataforma mediante un gráfico. Si accedemos a más información nos dirige a otro submenú de Supervisar.



El Estado del sistema que nos advierte de los fallos de los servicios de nuestra plataforma, como por ejemplo comunicación con la BBDD, estado de los Connection Server, certificados TLS, etc.

Almacenes de datos						
Nombre	vCenter Server	Ruta	Capacidad (GB)	Espacio utilizado (GB)	Libre (GB)	Opciones
vsan01-local	vcse01-vsan.reflab.int	/RefLab-CPD/vsan01-local	104	14	90	
vsan03-local	vcse01-vsan.reflab.int	/RefLab-CPD/vsan03-local	104	14	90	
vsan04-local	vcse01-vsan.reflab.int	/RefLab-CPD/vsan04-local	104	14	90	
vsanDatastore	vcse01-vsan.reflab.int	/RefLab-CPD/vsanDatastore	28.615	6.141	22.474	
vsan02-local	vcse01-vsan.reflab.int	/RefLab-CPD/vsan02-local	104	14	90	

La Carga de trabajo, nos informa de la capacidad de los Datastores que tenemos disponibles para almacenar los escritorios virtuales.

Sesiones

Usuario	Tipo	Grupo o grane	Nombre DNS	ID de cliente	Puerto de enlace...
reflab.int\REFLAB	Escrítorio	Escrítorio Oficina	vdi-oficina021.reflab.int	544810EDE640	vdi.reflab.es
reflab.int\REFLAB	Escrítorio	Escrítorio Oficina	vdi-oficina020.reflab.int	544810EDE2CF	vdi.reflab.es
reflab.int\REFLAB	Escrítorio	Escrítorio Centro	vdi-centro020.reflab.int	544810EDE57F	vdi.reflab.es
reflab.int\REFLAB	Escrítorio	Escrítorio Biología	vdi-lm003.reflab.int	544810EDE508	vdi.reflab.es

Nos permite revisar el estado de las sesiones activas, inactivas e interactuar con estas sesiones, con opciones como Desconectar, Cerrar sesión, Reiniciar un escritorio o incluso Enviar un mensaje de mantenimiento a un usuario concreto.

Eventos

Usuario	Gravedad	Hora	Módulo	Mensaje
REFLAB_BCN\jcorral	Información	16/03/2020 9:50	Agente	Perfil de tiempo de reconexión del escritorio biología...
REFLAB_BCN\jcorral	Información	16/03/2020 9:45	Agente	Perfil de tiempo de ejecución del escritorio biología...
REFLAB_BCN\jcorral	Información	16/03/2020 9:43	Agente	Obtener perfil de tiempo de la configuración en RLV
REFLAB_BCN\jcorral	Información	16/03/2020 9:43	Agente	El usuario REFLAB_BCN\jcorral se ha vuelto a conectar.
REFLAB_BCN\jcorral	Información	16/03/2020 9:43	Agente	El usuario REFLAB_BCN\jcorral solicitó el grupo biología...
REFLAB_BCN\jcorral	Información	16/03/2020 9:43	Agente	El usuario REFLAB_BCN\jcorral solicitó el grupo biología...

Los eventos registrados en este portal son los mismo que disponíamos en la consola Flex, para ver todo lo que sucede en nuestra plataforma.

Departamento de soporte técnico

Herramienta del departamento de soporte técnico

Haga clic en el icono de búsqueda situado en la parte superior de la página e introduzca un nombre de usuario para ver el estado de la sesión de usuario y realizar tareas de mantenimiento y solución de problemas en tiempo real.

Este apartado es una de las grandes novedades del nuevo portal en HTML5. Nos permite buscar por usuario y obtener información en tiempo real de la sesión y equipo virtual que usa ese usuario en concreto.

Si realizamos la búsqueda de un usuario, encontraremos las sesiones abiertas, los escritorios en uso, las aplicaciones a las que se ha conectado y los Actividades generadas.

Cliente

- Nombre de usuario: user
- Versión del cliente: 5.2.0
- Protocolo: VMware Blast

Máquina virtual

- Nombre del equipo: vdi-oficina027
- Grupo: Escritorio Oficina
- Duración de la sesión: 4 horas 59 minutos
- Hora de inicio de sesión: 23/3/20 7:59
- Estado de la sesión: Conectado
- Duración de inicio de sesión: 0 s
- Servidor de conexión: vCenter
- Nombre de proxy/puerta de enlace: [vdi-oficina027](#)

Indicadores de la experiencia del usuario

- Frecuencia de trama: 1 FPS
- Estado de Skype: Optimizado
- Contadores de sesiones de BLAST

 - Ancho de banda estimado (enlace ascendente): 955 Mbps
 - Pérdida de paquetes (enlace ascendente): 0 %
 - Bytes transmitidos: 143 MB
 - Bytes recibidos: 4,42 MB

- Contadores de CDR de BLAST

 - Bytes transmitidos: 884 B
 - Bytes recibidos: 582 B
 - Bytes transmitidos: 3,39 MB
 - Bytes recibidos: 32 B

Botones: Enviar mensaje, Asistencia remota, Reiniciar, Más...

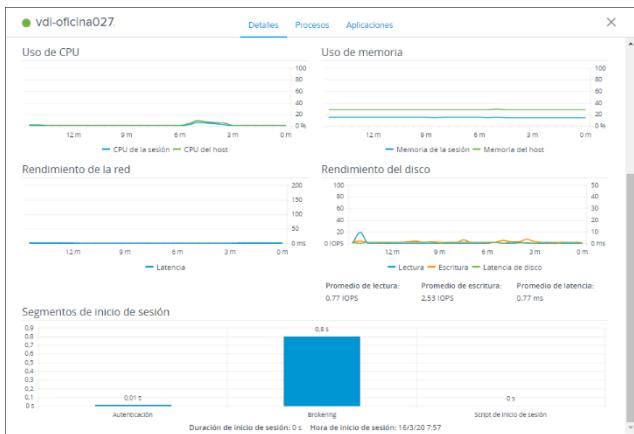
Accediendo a la información del escritorio virtual, dispondremos de varios apartados con mucha información que nos puede ser muy útiles para realizar un seguimiento de problemas.

Cliente: Información sobre el cliente de conexión que ha abierto esta sesión.

Máquina virtual: Información sobre el escritorio virtual, como el agente que usa, SO, servidor de conexión donde está conectado, duración de la sesión, etc.

Indicadores de la experiencia del usuario: Información sobre el rendimiento del protocolo de comunicación BLAST, como Ancho de banda estimado, Bytes transmitidos en IMAGEN, AUDIO.

Este apartado nos permite realizar acciones típicas de soporte como Enviar un mensaje al usuario, conectarnos mediante Asistencia remota de Windows, Reiniciar el equipo virtual, además de Desconectar, Cerrar la sesión o incluso restablecer el escritorio virtual.



Otra información muy útil es la de contadores de CPU, RAM, Red y Disco de los últimos 15 minutos de uso, además del control de tiempos del último inicio de sesión del usuario.

vdi-oficina027.					Finalizar proceso
CPU del host: 7 %	Memoria del host: 28 %	Procesos: 138			
Nombre del proceso	CPU	Memoria	Disco	Nombre de usuario	
AdobeCollabSync.exe	0 %	3.016 KB	0 KB/s	j30n9m0s	
AdobeCollabSync.exe	0 %	7.524 KB	20 KB/s	j30n9m0s	
AdobeUpdateService.exe	0 %	1.984 KB	0 KB/s	SVSTEM	
AGSService.exe	0 %	3.056 KB	0 KB/s	SVSTEM	
AGSService.exe	0 %	2.036 KB	0 KB/s	SVSTEM	
ApplicationFrameHost.exe	0 %	4.456 KB	0 KB/s	j30n9m0s	
armvcs.exe	0 %	1.412 KB	0 KB/s	SVSTEM	
audiogd.exe	0 %	7.720 KB	0 KB/s	SERVICIO LOCAL	
conhost.exe	0 %	5.712 KB	0 KB/s	SVSTEM	
conhost.exe	0 %	5.380 KB	0 KB/s	SVSTEM	
ctrss.exe	0 %	1.740 KB	0 KB/s	-	
ctrss.exe	0 %	1.980 KB	0 KB/s	-	
ctfmon.exe	0 %	12.236 KB	0 KB/s	j30n9m0s	

Dispondremos de la posibilidad de ver los procesos de Windows que está ejecutando el escritorio virtual y ver la carga de CPU, RAM o Disco de tal manera que podamos saber que está pasando.

vdi-oficina027.			Finalizar aplicación
CPU del host: 7 %	Memoria del host: 28 %	Aplicaciones: 3	
Aplicación	Descripción	Estado	
Elementos enviados	Microsoft Outlook	En ejecución	
Herramienta Recortes	Herramienta Recortes	En ejecución	
Microsoft Edge	Microsoft Edge	En ejecución	

Por último, también tendremos la posibilidad de ver las aplicaciones que corren dentro del escritorio virtual.

USUARIOS Y GRUPOS

Disponemos de un apartado central, del mismo modo que ya teníamos en la anterior consola Flex, donde podemos ver todos los usuarios autorizados a escritorios o aplicaciones, además nos permite autorizar a nuevos o desautorizarlos.

Nombre de usuario	Nombre	Apellido	Dominios	Autorizaciones de escritorios	Autorizaciones de aplicaciones
user1@reflexion.es	reflexion	user1	1	0	0
user2@reflexion.es	reflexion	user2	1	0	0
user3@reflexion.es	reflexion	user3	1	0	0
user4@reflexion.es	reflexion	user4	1	0	0
user5@reflexion.es	reflexion	user5	1	0	0
user6@reflexion.es	reflexion	user6	1	0	0
user7@reflexion.es	reflexion	user7	1	0	0
user8@reflexion.es	reflexion	user8	1	0	0
user9@reflexion.es	reflexion	user9	1	0	0
user10@reflexion.es	reflexion	user10	1	0	0

ASIGNACIONES JMP

Como ya hemos visto antes, una vez configurado nuestro JMP, desde este apartado podremos asignar y controlar los escritorios.

Le damos la bienvenida a la Just-in-Time Management Platform. Esta plataforma le permite asignar y controlar de forma adecuada los escritorios desde una única ubicación.

Para empezar, tendrá que proporcionar la URL del servidor en el que se encontrará JMP. Solo se puede usar una URL de JMP cada vez.

[Agregar JMP Server](#)

Proceso de configuración - <https://techzone.VMware.com/quick-start-tutorial-VMware-horizon-jmp-integrated-workflow#926454>

CONEXIONES REMOTAS

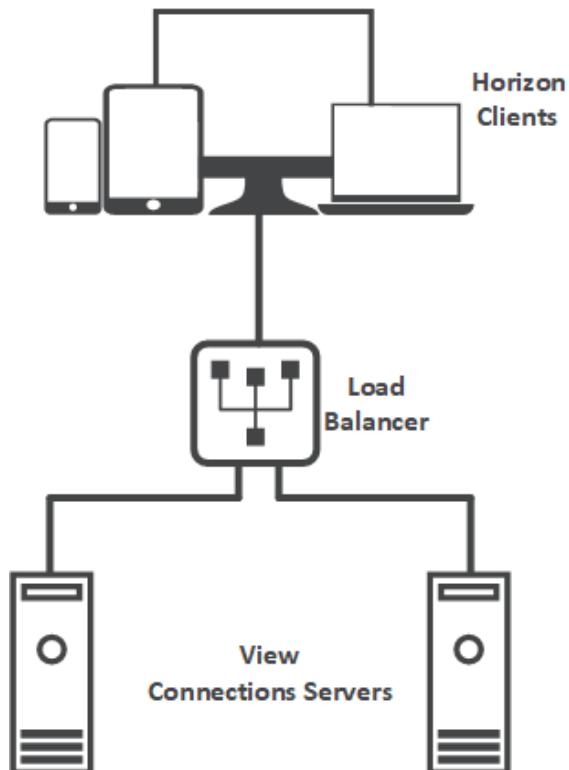
En todas las infraestructuras de escritorios virtuales consideramos una gran ventaja poder disponer de un escritorio con independencia del dispositivo que estemos usando, un portátil, un ThinClient o un Smartphone.

Con una plataforma que nos permite tanta flexibilidad, debemos diferenciar entre conectarnos desde cualquier dispositivo y conectarnos desde cualquier lugar. El primero es muy simple, siempre que dispongamos de un navegador web o un dispositivo con Horizon Client, con acceso a nuestro Connection Server, ya podemos conectarnos, pero en el segundo escenario no basta con redirigir el tráfico de nuestro acceso de internet al Connection Server para disponer de los escritorios y aplicaciones desde cualquier parte del mundo.

Es momento de abordar cuáles son algunas de las posibilidades de disponer de nuestra plataforma VMware Horizon desde cualquier lugar.

Como en la mayoría de los diseños de infraestructuras, las posibilidades son infinitas y mejorables, este capítulo pretende enseñar dos opciones básicas con las que podemos sostener un plan inicial, y según nuestras necesidades, complicar el diseño para ajustarlo.

Para facilitar la comprensión de las dos opciones, el diseño base de nuestra infraestructura de VMware Horizon se compone de dos servidores Connection Server y un balanceador de conexiones. Omitimos el hecho de si disponemos de Composer Servers, App Volumes u otros servicios, ya que, para el caso de conexiones remotas, no es relevante.



VPN

Virtual Private Network, o más conocido como VPN, es una manera de extender nuestra red empresarial, más allá del alcance del cableado físico, usando un acceso a Internet público. Mediante una VPN, podemos conseguir conectarnos desde cualquier parte del mundo, permitiendo el acceso a todos los recursos de nuestra red empresarial.

Esta tecnología no es algo novedoso, lleva con nosotros muchísimo tiempo, y es por ello que es una tecnología de fácil acceso, podemos encontrar multitud de soluciones para implementar VPN en nuestra empresa, y lo más habitual es realizarlo con el mismo dispositivo de control perimetral del que disponemos, el Firewall.

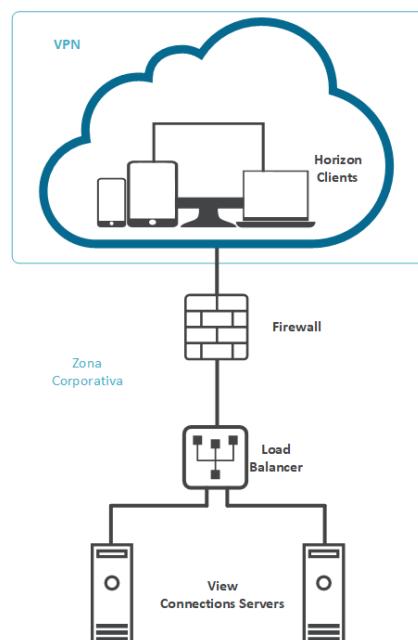
Cuando planteamos el despliegue de una VPN podemos encontrar dos grandes opciones: LAN-to-LAN o CLIENT-to-LAN.

Las redes VPN LAN-to-LAN son un tipo de diseño pensado para unir sedes empresariales que están geográficamente separadas, y entre ellas solo disponen de conexión a Internet público.

Con este tipo de conexión podemos conseguir que ambas sedes comparten recursos como si de una red LAN se tratase. Pero tiene un requisito, y es que ambas sedes deberán disponer de un dispositivo físico para realizar esta conexión VPN.

En el caso de las redes CLIENT-to-LAN, el concepto es idéntico pero el objetivo es conectar un solo ordenador a nuestra red LAN, por lo que, a través de software, sin necesidad de dispositivo dedicado, configuraremos un ordenador para permitir el acceso a nuestra red y que tenga acceso a todos los recursos de nuestra LAN como si estuviese trabajando desde un ordenador conectado físicamente a ella.

Si llevamos estos conceptos a nuestra infraestructura de VMware Horizon, descubriremos que no debemos retocar nuestro diseño de la plataforma, tan solo adaptar nuestra topología de red para permitir el acceso de la VPN a los servidores de VMware Horizon.



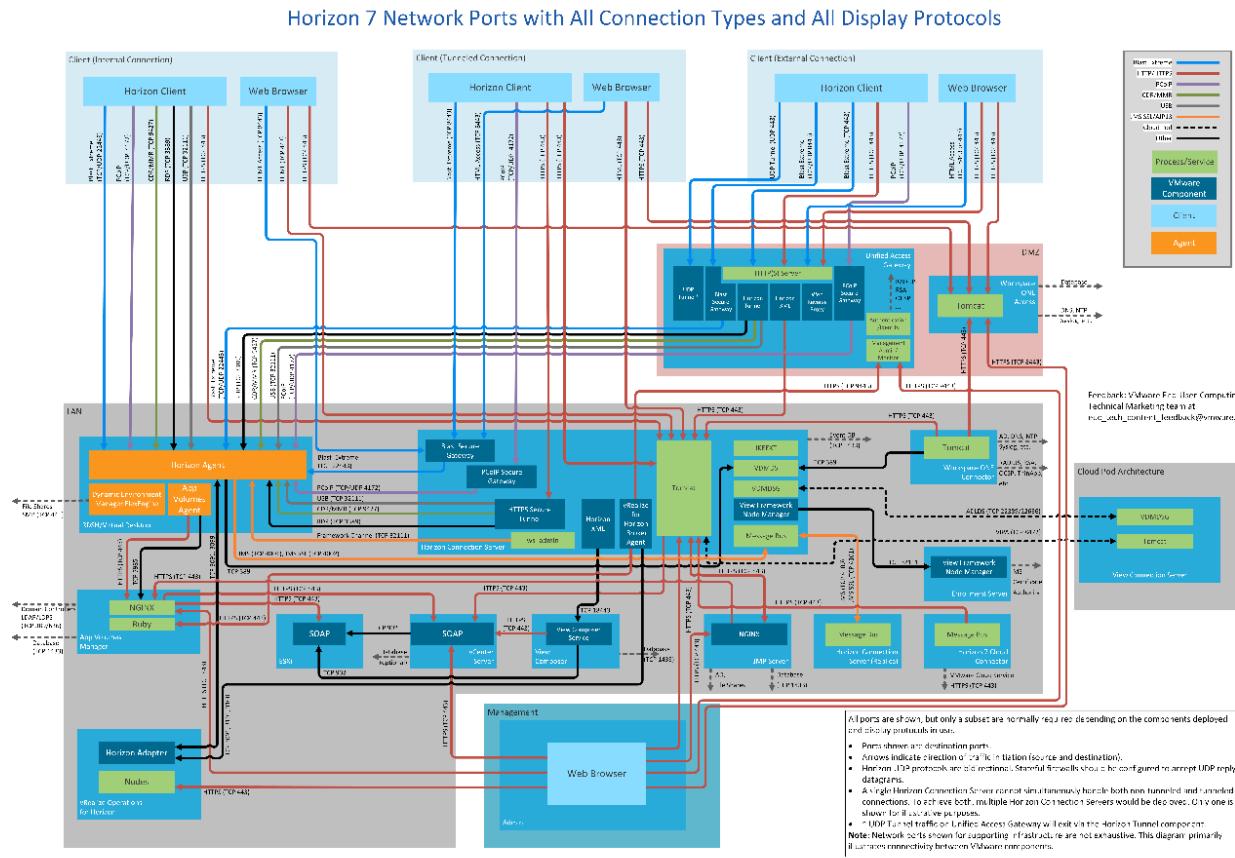
El apartado más importante de esta implementación, mediante VPN, es la seguridad. Como hemos comentado anteriormente, el hecho de habilitar un acceso a nuestra LAN a ordenadores que habitualmente no controlamos expone nuestra red a amenazas que hasta ahora no teníamos en cuenta.

Conseguir que esta implementación sea segura, no depende de nuestra plataforma VMware Horizon, sino que depende de la seguridad que podamos implementar en nuestra topología de red, protegiendo el acceso VPN hacia nuestra LAN.

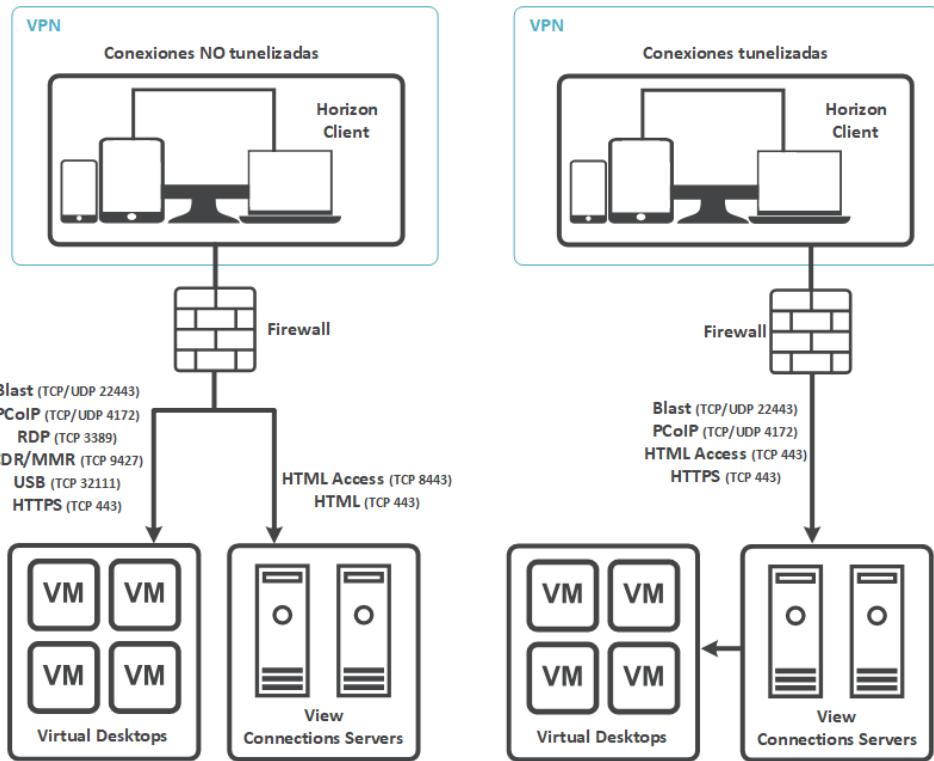
Si pensamos en un empleado que quiere disponer de VPN en el ordenador de casa para conectarse a su puesto de trabajo, debemos prever que ese ordenador no será seguro, no dispondrá de antivirus, posiblemente tenga algún malware o pueda llegar a ser un “usuario curioso” con herramientas poco recomendables para ver hasta dónde llega, es por ello que debemos controlar la VPN para que no redirija todo el tráfico de ese PC a nuestros sistemas, sino que debemos establecer que la VPN solo autorice el acceso a nuestro balanceador.

Dicho esto, es importante saber qué puertos debemos abrir para que el usuario pueda conectarse a su escritorio de manera completa, por lo que debemos darle un vistazo al mapa de red de VMware Horizon.

<https://techzone.VMware.com/sites/default/files/Horizon%207%20Network%20Ports%201%20-%20All%20Connection%20Types%20and%20All%20Display%20Protocols.png>



Como ya comentamos en el capítulo 12 del libro anterior, VMware por vExperts, existen varias maneras de tunelizar las conexiones entre clientes y escritorios, por lo que debemos abrir los puertos correspondientes en cada caso, como podemos observar en el siguiente esquema.



Para concluir esta implementación, me gustaría añadir una reflexión. Cuando se plantea un escenario como el que hemos descrito, estamos ante una solución muy simple de implementar, pues en la gran mayoría de empresas es relativamente sencillo configurar una VPN para nuestros usuarios con los recursos que ya disponemos en la empresa, y permitir el acceso a la plataforma de VMware Horizon, pero esta facilidad nos pone en riesgo porque estamos abriendo la puerta a nuevas amenazas, y estas amenazas vendrán desde redes públicas.

Esta implementación puede valer a corto plazo, pero en el caso de tener una solución definitiva, es muy importante, además de proteger el acceso desde la red VPN a nuestra LAN, implementar medidas sobre las contraseñas de acceso de nuestros usuarios a la empresa, por ejemplo, aumentando la seguridad del acceso mediante un sistema de doble factor de autenticación en la VPN, o estableciendo políticas de control del dispositivo cuando se conecta a nuestra plataforma, como por ejemplo, que no se permita conectar sin antivirus.

UNIFIED ACCESS GATEWAY (UAG)

<https://docs.VMware.com/es/Unified-Access-Gateway/index.html>

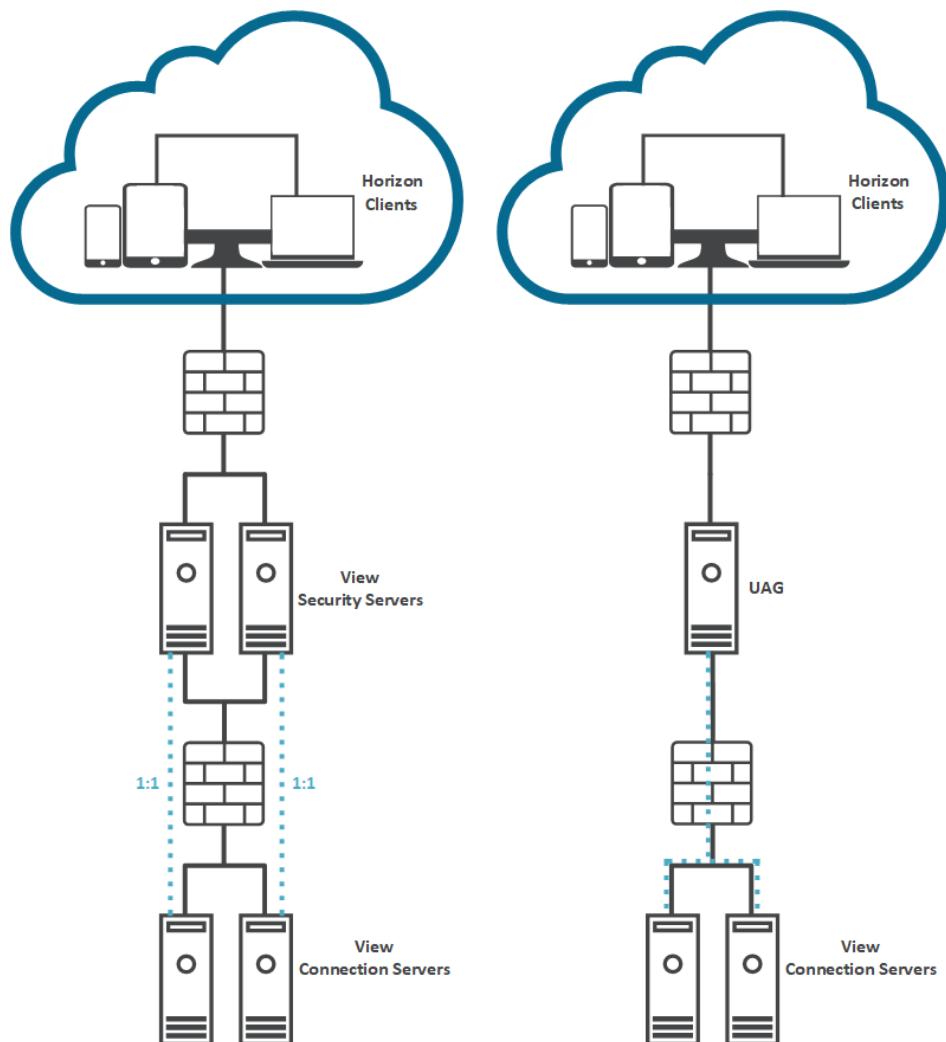
Unified Access Gateway o más conocido como UAG permite publicar en internet el acceso a las aplicaciones y escritorios de VMware Horizon de una manera segura. Además de permitir

publicar los servicios de la plataforma de Horizon, también sirve como proxy para los servicios de Workspace ONE Access y Workspace ONE UEM.

UAG funciona como host proxy de las solicitudes de autenticación hacia la plataforma de Horizon, permitiendo trabajar con distintos sistemas como SAML, RADIUS, RSA SecurID, Smart Card o Certificados de usuario.

Si miramos atrás, veremos que UAG es relativamente nuevo, con este nombre empieza en septiembre de 2017. Anterior a UAG el producto, se denominaba VMware Access Point hasta la versión 2.8 y compatible desde VMware Horizon 6.2; posteriormente pasó a llamarse UAG empezando en la versión 3.1 y compatible con VMware Horizon 6.2.3.

A fecha de la escritura de este libro, todavía existe el servicio de Security Server, el cual se incluye en todas las versiones de VMware Horizon como un servicio más, al igual que el Connection Server. Este servicio ya permitía, y permite, publicar de manera segura las aplicaciones y escritorios de VMware Horizon.



El servicio Security Server es un servicio que está destinado a desaparecer y dispone de desventajas frente a UAG que vamos a comentar.

Security Server	UAG
Requiere la instalación sobre Windows Server.	Se ejecuta sobre el Photon OS propietario de VMware y se instala desplegando una OVA.
La instalación es simple, pero requiere de Windows Server preparado con los requisitos.	Permite exportar la configuración e importarla en la instalación ante cualquier problema.
Requiere emparejar cada Security Server con un Connection Server, relación 1:1.	Un solo UAG puede trabajar con múltiples Connection Servers conectando a un balanceador.
La configuración es individual por cada instalación.	Podemos realizar scripts para automatizar la instalación.
La configuración se realizar desde la consola de Horizon	Tiene un portal dedicado que simplifica la instalación, configuración y mantenimiento.
Si queremos usar dos NICs para separar el tráfico, la parametrización se complica a nivel de Windows.	Podemos usar 3 NICs para separar el tráfico de Internet, Interno y Gestión de una manera sencilla.
Requiere puertos IPSec para emparejar con el Connection Server.	Tan solo requiere el puerto 443 hacia los Connection Servers o balanceador de carga.
La actualización debe coordinarse con el Connection Server para mantener las mismas versiones.	Las actualizaciones se realizan sin depender de los Connection Servers.
No soporta la nueva funcionalidad Blast Extreme.	
Es un servicio que no recibirá actualizaciones y desaparecerá en la versión Horizon 8.	

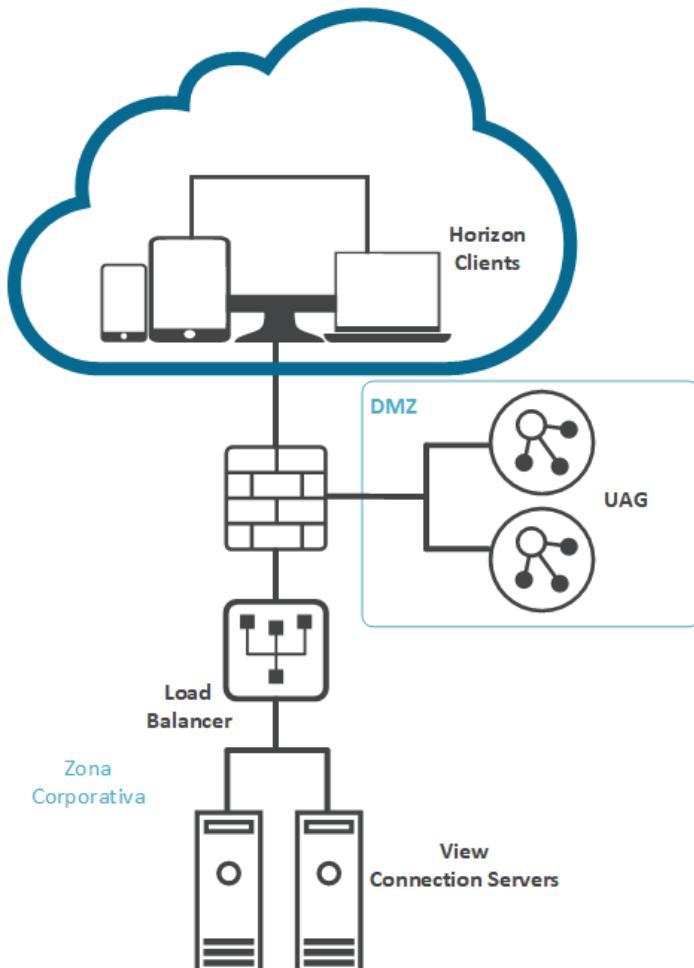
DISEÑO

Antes de poner en marcha UAG, es importante pensar cómo vamos a implementar la solución de acceso externo. Indiscutiblemente, UAG deberá alojarse en una DMZ para conseguir aislamiento de nuestra red interna.

Cuando disponemos de una DMZ en nuestra infraestructura podemos encontrarnos dos escenarios, el primero una empresa con un Firewall y distintas zonas, entre ellas las internas, la DMZ y el acceso a Internet, o un segundo escenario donde existe un Firewall perimetral, que dispone de las redes de Internet y DMZ externa, y otro Firewall que dispone de la DMZ interna y las redes internas de nuestra organización.

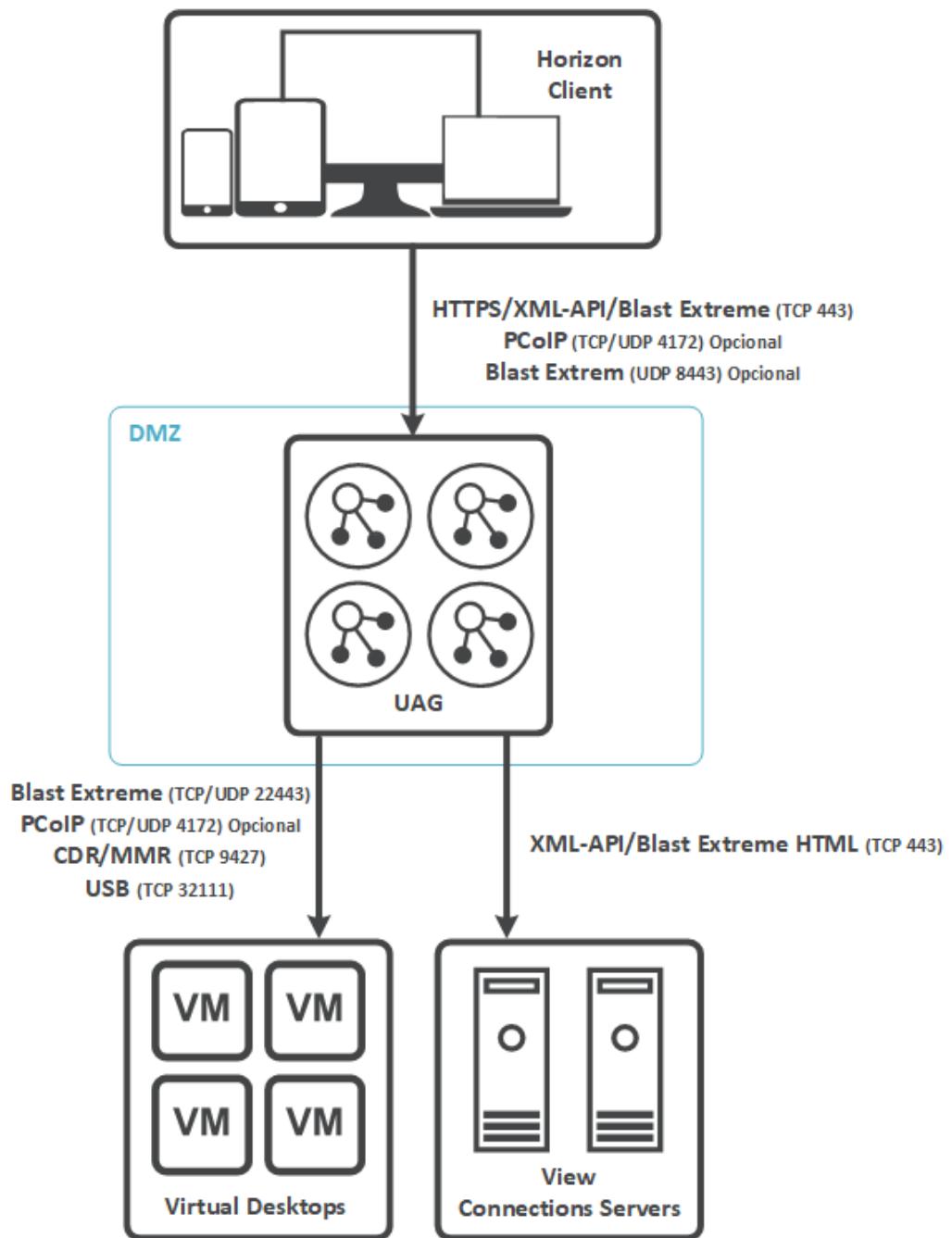
Ambos escenarios son válidos para implementar UAG, el despliegue del aplicativo nos permite seleccionar desde 1 hasta 3 NICs para nuestra implementación, por lo que se adapta a cualquier escenario de seguridad que ya tengamos en la organización.

En el diseño que planteamos en el libro, usaremos un solo Firewall donde nuestro UAG estará conectado a la zona DMZ.



Es importante prestar atención a los puertos que debemos abrir y redireccionar desde la parte de Internet a los UAG, y de los UAG a nuestra red interna, tanto a los Connection Server como a los escritorios.

<https://docs.VMware.com/es/Unified-Access-Gateway/3.4/com.VMware.uag-34-deploy-config.doc/GUID-F197EB60-3A0C-41DF-8E3E-C99CCBA6A06E.html>



DESPLIEGUE

VMware nos proporciona el OVA de la imagen del UAG, y es tan sencillo como desplegarlo con la metodología habitual de vCenter.

https://docs.VMware.com/es/VMware-vSphere/7.0/com.VMware.vsphere.vm_admin.doc/GUID-17BEDA21-43F6-41F4-8FB2-E01D275FE9B4.html

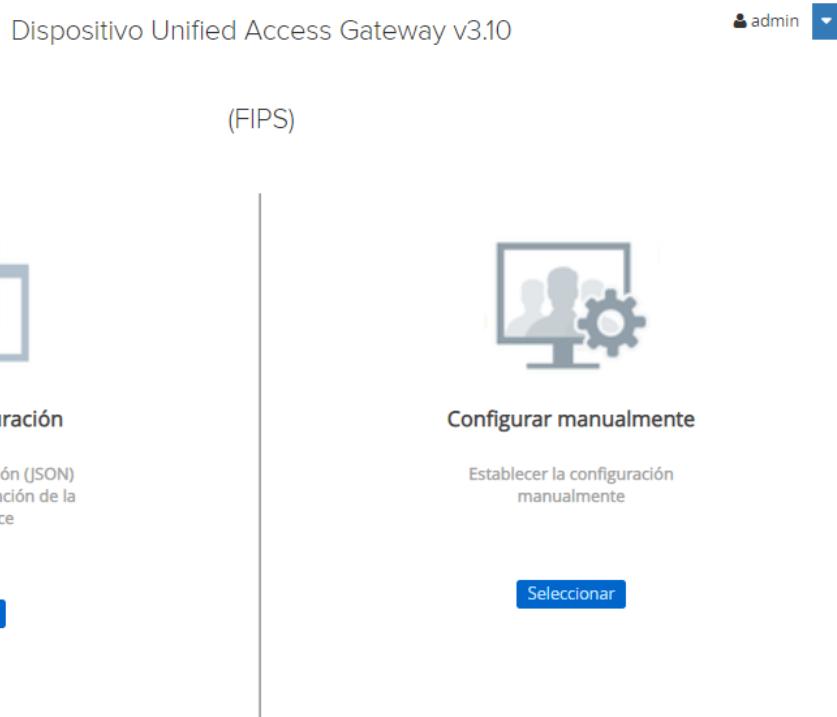
Para simplificar este proceso, o como mínimo hacerlo más amigable, se ha creado una herramienta llamada Unified Access Gateway Appliance Installer, el cual permite hacer un despliegue sin acceder al vCenter; si instalamos esta herramienta veremos que es muy parecida al asistente que usamos para desplegar nuestro vCenter.

Podéis ver el proceso de despliegue en este post.

<https://www.cenabit.com/2020/08/desplegar-uag-con-uag-deployment-utility/>

CONFIGURACIÓN

Ya disponemos de el primer servidor UAG aprovisionado por lo que es momento de configurarlo y conectarlo con nuestra infraestructura de VMware Horizon.



CONFIGURACIÓN GENERAL

Cuando accedemos a la configuración manual de nuestro UAG lo primero que vemos es la configuración del servicio perimetral, es decir, qué servicios tiene activos nuestro UAG y todas las ventanas de configuración para configurarlo, además del servicio de Autenticación para establecer la que más nos convenga para proteger la identificación de nuestros escritorios y aplicaciones.

En este caso, veremos por encima los detalles de la configuración para VMware Horizon.

Configuración general

Configuración del servicio perimetral

OCULTAR

- Configuración de Horizon
- Configuración del proxy inverso
- Configuración de Tunnel
- Configuración de Secure Email Gateway
- Configuración de Content Gateway

Configuración de autenticación

OCULTAR

Configuración de autenticación	Estado	
RSA SecurID	Deshabilitado	
RADIUS	Deshabilitado	
Certificado X.509	Deshabilitado	
Autenticación adaptativa RSA	Deshabilitado	

Cuando activamos la configuración de Horizon, podemos ver que nos solicita URL del servidor de conexión y la huella digital de URL. En este momento tenemos posibilidad de conectar nuestro UAG contra un Connection Server o contra un balanceador.

En un escenario de producción es importante disponer de un balanceador para no depender exclusivamente de un solo servidor. En el siguiente enlace podéis encontrar más información de como balancear conexiones en VMware Horizon.

<https://www.cenabit.com/2018/06/balancear-las-conexiones-en-horizon-view/>

Si nuestra infraestructura dispone de un certificado público, tan solo será necesario establecer la URL del balanceador o Connection Server, pero en el caso que usemos certificados autofirmados, debemos buscar la Huella Digital de nuestro certificado y añadirla en el campo de Huella Digital.

Configuración de Horizon

Habilitar Horizon	<input checked="" type="button"/> Sí	?
URL del servidor de conexión *	<input type="text" value="https://vdi.vmwareporexperts.org:443"/>	?
Huella digital de URL del servidor de conexión	<input type="text" value="SHA256=8d 2a 2a 30 7d 26 88 f3 c7 89 b0 68 45 24"/>	?

Especificaremos qué protocolos queremos usar a través de nuestro UAG, PCoIP o Blast Extreme, ambos soportados. Es importante cuando tratamos con acceso externo no activar más cosas de las necesarias, por lo que, si tu empresa no usa el protocolo PCoIP, no es necesario activarlo.

Con Blast Extreme tenemos la posibilidad de usar el protocolo UDP en la transmisión a través del puerto 8443 para conexiones lentas, desde esta misma ventana de configuración también podemos permitirlo activando el túnel UDP.

Configuración de Horizon

Habilitar Horizon	<input checked="" type="button"/> Sí	?
URL del servidor de conexión *	<input type="text" value="https://vdi.vmwareporexperts.org:443"/>	?
Huella digital de URL del servidor de conexión	<input type="text" value="sha256=8d 2a 2a 30 7d 26 88 f3 c7 89 b0 68 45 24"/>	?
Modo IP del servidor de conexión	<input type="button" value="IPv4"/>	?
Encabezado de origen de reescritura	<input checked="" type="button"/> NO	?
Habilitar PCoIP	<input checked="" type="button"/> NO	?
Habilitar Blast	<input checked="" type="button"/> Sí	?
URL externa de Blast	<input type="text" value="https://vdi.vmwareporexperts.org"/>	?
Habilitar servidor del túnel UDP	<input checked="" type="button"/> NO	?
Certificado de proxy de Blast	Seleccionar	?
Habilitar túnel	<input checked="" type="button"/> NO	?
Más	+	
Guardar Cancelar		

A parte de las configuraciones básicas para conectar con nuestros Connection Server y poder acceder a los escritorios y aplicaciones, tenemos más configuraciones para el tema de Comprobación de Endpoints, que se detalla en las Configuraciones Avanzadas, así como las configuraciones para SAML en el caso de autenticación con terceros.

Si usamos un balanceador de carga para nuestros UAG de manera que las solicitudes vayan hacia un nombre concreto, y queremos desviarlas a un UAG concreto, podemos usar el apartado de Asignaciones de redirecciónamiento de host.

Incluso podemos bloquear el acceso a nuestros escritorios y aplicaciones mediante HTML Access, obligando a los usuarios a acceder a través de Horizon Client.

Proveedor para comprobación de conformidad de endpoints

(i)

(i)

Patrón de proxy

(i)

(i)

SP de SAML

(i)

(i)

Coincidir con el nombre de usuario de Windows

NO
(i)

(i)

Ubicación de la puerta de enlace *

(i)

(i)

Certificados de confianza

No se agregó ningún certificado de confianza.

(i)
(i)

(i)

Encabezados de seguridad de respuesta

Nombre	Valor	(i)
Strict-Transport-Security: max-age=31536000		(i)
X-XSS-Protection: 1; mode=block		(i)
X-Content-Type-Options: nosniff		(i)
Content-Security-Policy: default-src 'self'; font-src 'self' data; script-src 'self' 'unsafe-inline' 'unsafe-eval' data; style-src 'self' 'unsafe-inline'; img-src 'self' blob: data;		(i)
X-Frame-Options: SAMEORIGIN		(i)

(i)

Asignaciones de redirecciónamiento de host

Host de origen	Host de redirecciónamiento	(i)
		(i)

(i)

Entradas de host

(i)

(i)

Deshabilitar HTML Access

NO
(i)

(i)

Menos ↲

Guardar
Cancelar

Una parte importante es que tenemos que preparar nuestro Horizon para entenderse con el acceso del UAG y es por ello, por lo que debemos realizar dos pasos muy simples.

Primero, debemos configurar todos los Connection Server de nuestra plataforma, que vayan a aceptar los accesos externos a través de UAG, de manera que no tunelizan las conexiones.

Editar configuración del servidor de conexión

General **Autenticación** **Copia de seguridad**

Etiquetas
Las etiquetas se pueden usar para restringir los grupos de escritorios que pueden acceder a través de este servidor de conexión.

Etiquetas

 Separe las etiquetas con ; o ,

Túnel seguro HTTP(s)
 Usar conexión de túnel seguro con máquina ⓘ
 * URL externa
 ⓘ
 Ejemplo: https://myserver.com:443

Puerta de enlace segura PCoIP
 Usar la puerta de enlace segura PCoIP para las conexiones PCoIP de la máquina
 * URL externa de PCoIP
 ⓘ
 Ejemplo: 10.0.0.1:4172

Puerta de enlace segura de Blast
 Usar la puerta de enlace segura de Blast en todas las conexiones Blast de la máquina ⓘ
 Usar la puerta de enlace segura de Blast solo en las conexiones HTML Access de la máquina ⓘ
 No usar puerta de enlace segura de Blast ⓘ
 * URL externa de Blast
 ⓘ
 Ejemplo: https://myserver.com:8443

Cancelar **Aceptar**

El segundo punto es, registrar nuestro UAG o todos los que tengamos para que queden en nuestra plataforma de Horizon.

Servidores

vCenter Server Puertas de enlace Servidores de cone...

Registrar **Eliminar del registro**

Filtrar ⌂ ⌄

Puerta de enlace	Versión	Dirección IP	Ubicación
RLUAG1	3.10	192.168.105.21	Externo

CONFIGURACIÓN AVANZADA

Este apartado hace referencia a todas las configuraciones generales, que podemos realizar sobre el UAG a nivel de dispositivo, como gestión de la red, usuarios, certificados o incluso HA.

Configuración del sistema

Configuración del sistema

Nombre de UAG	RIUAG1
Configuración regional *	en_US
Vigencia de la contraseña	90
Conjuntos de cifrado *	TLS_AES_128_GCM_SHA256,TLS_AES_256_GCM_SHA256
Habilitar TLS 1.0	<input type="radio"/> NO
Habilitar TLS 1.1	<input type="radio"/> NO
Habilitar TLS 1.2	<input checked="" type="radio"/> SI
Habilitar TLS 1.3	<input checked="" type="radio"/> SI
Tipo de syslog	UDP
URL de Syslog	
URL de auditoría de Syslog	
URL de comprobación de estado *	/favicon.ico
Cookies que se deben almacenar en caché *	none
Modo inactivo	<input type="radio"/> NO
Intervalo de supervisión *	60
Tiempo de expiración de autenticación	300000
Tiempo de espera de recepción de cuerpo	15000
Máximo de conexiones por sesión	16
Tiempo de espera de inactividad de conexión de cliente	360
Máximo de conexiones por sesión	16
Tiempo de espera de inactividad de conexión de cliente	360
Tiempo de espera de solicitud	10000
Tolerancia de sesgo del reloj	600
Máximo de CPU del sistema permitido	100
Tiempo de expiración de sesión *	36000000
Unirse a CEIP	<input type="radio"/> NO
Habilitar SNMP	<input type="radio"/> NO
Texto de descargo de responsabilidad de administrador	
DNS	Agregar nueva entrada de DNS 192.168.1.199
Búsqueda de DNS	Agregar nueva entrada de búsqueda de DNS reflab.int
Servidores NTP	Agregar nuevo servidor NTP 192.168.1.199
Servidores NTP de reserva	Agregar nuevo servidor NTP
<input type="button" value="Guardar"/> <input type="button" value="Cancelar"/>	

Los apartados más importantes son el de renombrar el UAG, cambiar la configuración regional, por defecto se establece en en_US, controla la vigencia de la contraseña de admin, permite activar o desactivar tipos de cifrado, se puede activar SNMP, aragar o cambiar servidores DNS, NTP, dominio de búsqueda.

Configuración de red

The screenshot shows the 'Configuración de red' (Network Configuration) screen. It displays basic network information and settings for NIC 1 (Internet Interface). Key details include:

- Puerta de enlace IPv4 predeterminada: 192.168.105.254
- Puerta de enlace IPv6 predeterminada: No disponible
- NIC 1: Interfaz de Internet
- Modo de IP: IPv4
- Modo de asignación de IP: STATIC/V4
- Dirección IPv4: 192.168.105.21
- Máscara de red: 255.255.255.0
- IPv4: Rutas estáticas
- IPv4: Dirección IPv6
- IPv4: Prefijo IPv6

A 'Cerrar' (Close) button is at the bottom right.

Nos permitirá modificar la IP de nuestro UAG. En ningún caso podremos cambiar de 1 NIC a 2 o 3 NICs, es por ello que debemos pensar el diseño con antelación.

En el caso de disponer de 2 o 3 NICs hay un campo que es el de rutas que nos permitirá establecer por donde ha de dirigirse el tráfico de cada segmento de red que nos interese.

Configuración de Alta disponibilidad

The screenshot shows the 'Configuración de Alta disponibilidad' (High Availability Configuration) screen. It includes fields for Mode (set to HABILITADO), Virtual IP Address (192.168.105.20), and Group ID (10). Buttons for Guardar (Save) and Cancelar (Cancel) are at the bottom.

Si disponemos de más de un UAG, debemos activar la opción de Alta disponibilidad y establecer una IP virtual y un ID de grupo para todos los miembros.

Si disponemos de 2 o más UAG, siempre tendremos uno que será el MASTER y los otros estarán a la espera por si este cae, en ese momento uno de los dos cogerá el rol de MASTER y de este modo no perderemos el servicio.

Configuración del certificado del servidor TLS

The screenshot shows the 'Configuración del certificado del servidor TLS' (Server Certificate Configuration) screen. It allows selecting where to apply the certificate (Administrator Interface or Internet Interface) and uploading PEM or PFX files. A message at the bottom indicates the certificate was successfully applied to the Internet interface.

Aplicar certificado a* Interfaz de administrador Interfaz de Internet

Tipo de certificado: PEM

Clave privada*: mycaservercertkeyrsa.pem

Cadena de certificados*: mycaservercert.pem

Guardar Cancelar

El certificado de la interfaz de Internet se ha cambiado correctamente.

Se ha cargado el certificado de la interfaz de administrador. Cierre esta ventana y vuelva a abrir la interfaz de usuario de administrador en una ventana nueva.

Seleccionamos donde aplicaremos el certificado (Interfaz del administrador o Interfaz de Internet).

Nos permite subir el certificado en formato PEM o PFX, seleccionamos los archivos y guardamos la configuración.

Configuración SAML

The screenshot shows the 'Configuración de SAML' page. It has two main sections: 'Configuración del proveedor de identidades SAML' and 'Configuración del proveedor del servicio SAML'. In the first section, there are two radio button options: 'Metadatos autofirmados generados' (selected) and 'Proporcionar certificado'. Below these are 'Guardar', 'Descargar la configuración del proveedor de identidades', and a download dialog box. In the second section, there are fields for 'Nombre del proveedor del servicio' and 'XML de metadatos', along with 'Guardar' and 'Cancelar' buttons. At the bottom is a 'Cerrar' button.

Esta configuración nos permite conectar con servicios de proveedor de identidad de terceros y de este modo poder usar otros tipos de autenticación, como por ejemplo, tarjeta inteligente.

Configuración del proveedor para comprobación de conformidad de endpoints

The screenshot shows the 'Configuración de proveedor para comprobación de conformidad de endpoints' page. It includes fields for 'Proveedor para comprobación de conformidad de endpoints' (OPSWAT), 'Clave de cliente', 'Secreto de cliente', 'Nombre de host' (gears.opswat.com), and two interval fields ('Intervalo de comprobación de conformidad (min)' and 'Intervalo rápido de comprobación de conformidad (min)'). There are also links for 'Mostrar códigos de estado permitidos' and 'Mostrar configuración del agente a petición de OPSWAT'. At the bottom are 'Guardar' and 'Cancelar' buttons.

Podemos conectar con el proveedor OPSWAT que permite establecer una comprobación de los dispositivos que conectan con nuestra infraestructura y que estos, cumplen los criterios definidos en el portal de OPSWAT, como por ejemplo que dispongan de Antivirus, Firewall activado, etc.

Configuración de JWT

Si disponemos de Workspace One Access podemos configurar JSON Web Token (JWT) para configurar la función de acceso único. Si se activa JWT, UAG requerirá que el dispositivo que intenta autenticar esté registrado y de este modo descartará cualquier otro intento de conexión.

Configuración de la cuenta

Nos permite agregar una nueva cuenta con el rol de Monitor y cambiar la clave de la cuenta de Admin.

También nos indica cuánto tiempo queda para la caducidad de las contraseñas de las cuentas.

Configuración de puente de identidades

Con estas opciones podremos establecer la transmisión de identidades entre nuestro Workspace ONE de manera transparente con el inicio de sesión único.

CONFIGURACIÓN DE ASISTENCIA

Este apartado dispone de información que nos será muy útil para revisar y controlar el estado del UAG.

Estadísticas de sesiones de servicios Edge

Edge Service	Total Sessions	Active (Logged In) Sessions	Inactive Sessions	Failed Login Attempts	Session High Water Mark	PCoIP Sessions	BLAST Sessions	Tunnel Sessions
Horizon	11	0	11	8	11	0	0	0
Reverse Proxy (Ira)	10	0	10	10	10	-	-	-
Reverse Proxy (sp_Jsr)	11	0	11	11	11	-	-	-
Reverse Proxy (sp_Ira)	4	0	4	0	5	-	-	-
Reverse Proxy (sp_mlt_domain)	8	0	8	8	8	-	-	-
Vmware Tunnel	1	1	0	0	1	-	-	-
Total	45	1	44	37	-	-	-	-

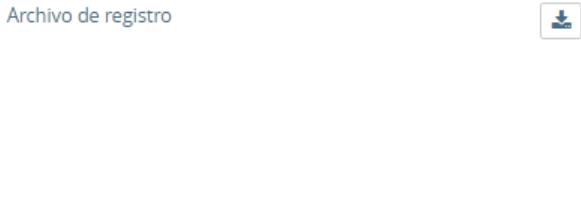
[Close](#)

Fuente de la imagen.

<https://docs.VMware.com/es/Unified-Access-Gateway/3.10/com.VMware.uag-310-deploy-config.doc/GUID-3ECAC00E-6D11-4573-8252-88A748B048CC.html>

Nos proporciona información de las sesiones activas e implementadas de nuestro UAG, tanto para Horizon como para Reverse Proxy como VMware Tunnel.

Archivo de registro



Permite descargar todos los logs contenidos en UAG y poder revisar en caso de necesidad. Una vez pulsamos su descarga, tarda unos minutos en empezar, ya que reúne todos los logs de UAG.

Configuración a nivel de registro

Establecemos el nivel de log que queremos que guarde nuestro UAG, por defecto lo tenemos en INFO, pero si necesitamos hacer un troubleshooting del sistema, podemos cambiarlo a ERROR o DEBUG para filtrar o aumentar el nivel del log.

Exportar la configuración de Unified Access Gateway

Exportar la configuración de
Unified Access Gateway



Una vez terminada la configuración del UAG, es importante exportar la configuración para poder tener un respaldo de nuestra configuración, y en el peor de los casos, poder desplegar un nuevo UAG con la configuración ya realizada mediante una importación.



**Adistec
Enterprise
Cloud**

OUR SERVICES ARE:
**VIRTUAL PRIVATE CLOUD and
BAAS** (Backup as a Service)

Virtual Private Cloud:

- A Line of Multitenant Servers operating from our four Datacenters located in Latin America (USA, Argentina, Peru & Brazil)
- Pre-Configured packages to simplify cloud adoption and growth
- Transparently priced (no hidden fees!)
- Based on VMware Cloud Director, providing a Web Portal for administration purposes
- Load Balancing and Security based on VMware NSX-Edge
- VPN Access (software) and one Public IP Address per VPC

BaaS benefits:

- Smart small and scale based on your company's needs
- Transparently priced (No hidden fees!)
- Centralized management from your Veeam Backup & Replication Console

- Create, Modify or Delete any Backup Job
- Fulfill any retention policy your business requires such as those long term & historical backups requirements (Grandfather-father-son backups)
- Recover complete VMs, VMs' configuration files, Complete VMDKs or VHDs,
- Guest files or granular item recovery such as a deleted email, SQL row, deleted Active Directory User Accounts, etc.
- Adistec Total Support with a 1-800 number to contact directly our technicians.
- VMs running on top of the following hypervisors are supported: VMware®
- ESXi (4, 5 & 6) and Hyper-V (Windows Server 2008, 2012, 2012 R2nd 2016)

Ask for more information



aec@adistec.com



A D I S T E C L A B S

Adistec Education presents our **Virtual Labs Services** which offers the possibility of renting laboratories, which deliver one vPOD per student per day, ensuring that the user experience is unique. Fully adaptable and designed according to the manufacturer's official specifications.

veeAM VMware® NUTANIX

FORTINET

kaspersky

**"Your Skills,
Our Infrastructure"**

Our benefits

- Free vPOD for the instructor
- Guarantee for service
- Try our labs for free
- vPod 24 hs available
- vPod available less than 72hs
- 7x24 support coverage
- 100% compliant with official manuals
- Trilingual Support (Spanish, English and Portuguese)

Ask for your trial for free

edu_labs@adistec.com

Adistec

Capítulo 6

MONITORIZACIÓN DE NUEVA GENERACIÓN CON WAVEFRONT... MÁS ALLÁ DE VSPHERE



Jorge De La Cruz

@jorgedlcruz

INTRODUCCIÓN A VMWARE WAVEFRONT

VMware anunció en 2017 que iba a comprar una empresa conocida como Wavefront, un startup especializado en la monitorización de métricas para aplicaciones basadas en la nube. El anuncio no podía llegar en mejor momento, ya que VMware se intentaba posicionar con servicios basados en Cloud, como es VMware Cloud on AWS, etc.

Aunque vamos a conocer mucho más sobre Wavefront más abajo, solamente os quería resumir aquí en pocas palabras lo que para mí es Wavefront. Wavefront es libertad absoluta para agrupar métricas que recolectamos de cien maneras diferentes, visualizadas de manera sencilla en un único servicio que, además, no tenemos que preocuparnos por mantener, o ampliar CPU, RAM, etc.

¿QUÉ ES WAVEFRONT?

Wavefront es una herramienta de monitorización de métricas y análisis en tiempo real, que se ejecuta en la nube teniendo en cuenta las cargas de trabajo, y es por ello por lo que ningún otro lugar es mejor que la nube para almacenar cientos de miles de métricas de diferentes orígenes.

Los datos se pueden ingerir desde millones de endpoints por cliente, y se pueden analizar en cuestión de segundos. Esta escala, y la capacidad de buscar todos los datos recogidos en tiempo real, han llevado a Wavefront a estar en el corazón de algunas de las principales aplicaciones de la nube en el mundo; tenemos varios ejemplos que conocemos, tales como son: Reddit, Workday, Doordash, OKTA, o incluso Box.

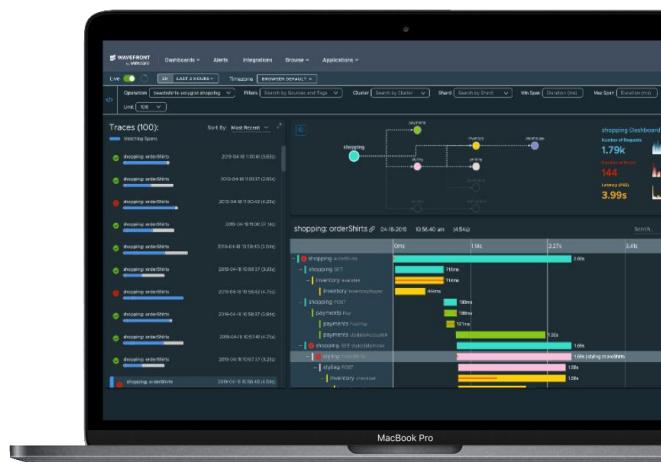


Ilustración 1 - Obtenida de la web oficial <https://www.wavefront.com/>

La recolección de datos de más de un millón de puntos finales en un momento dado muestra la escala a la que está diseñada para funcionar. La integración es posible con muchos productos más allá de los típicos de VMware. Está respaldada por muchos proveedores de servicios tales como son AWS, Azure y GCP, Linux, Ansible, Chef, Prometheus, y tecnologías

de bases de datos como MySQL. Todas las integraciones vienen con dashboards preconstruidos para reducir el tiempo de adopción.

Wavefront está diseñado para monitorizar aplicaciones "nativas de la nube" y con ello, una gran comprensión de las tecnologías de containers. Wavefront ofrece una visión en tiempo real de los containers Docker y de los principales sistemas de orquestación como es Kubernetes. Para Kubernetes, por ejemplo, Wavefront puede recopilar métricas de recursos sobre los containers, los namespaces, los nodos, las pod y el propio clúster.

La recolección de métricas puede hacerse de varias maneras, dependiendo del entorno. Las métricas pueden ser recolectadas y cargadas a Wavefront a través de un proxy usando agentes de Wavefront, biblioteca de métricas y código de aplicación, registros vía TCP o directamente de proveedores de nubes como, por ejemplo, AWS.

AWS puede integrarse de forma sencilla proporcionando a Wavefront las credenciales de AWS. Utilizando permisos de sólo lectura, Wavefront puede entonces extraer métricas del entorno AWS y cotejarlas en tableros, ejecutar búsquedas en tiempo real sobre esos datos, y ver la información de facturación.

Los datos se recogen recuperando las métricas y datos desde Amazon CloudWatch, la información de eventos de CloudTrail, y las métricas de AWS+ para métricas adicionales, utilizando APIs como los datos de volumen de EBS y los datos de facturación.

Los datos recogidos a lo largo del tiempo pueden ser buscados instantáneamente, con una respuesta muy rápida. Por ejemplo, los datos recopilados durante un período de 12 meses pueden ser fácilmente acotados y analizados en segundos.

Para Azure, es necesario crear una relación de confianza entre Azure y Wavefront con dashboards preconstruidos que cubren desde las máquinas virtuales Azure, las cuentas de almacenamiento Azure, las instancias de contenedores Azure, y las bases de datos SQL Azure.

Wavefront recopila datos métricos de series temporales en los que podemos realizar operaciones matemáticas arbitrarias para generar gráficos, para ver las anomalías o crear dashboards de indicadores clave de rendimiento (KPI). Se pueden configurar alertas inteligentes para vigilar de forma proactiva la pila de aplicaciones utilizando estos datos recopilados.

Las alertas pueden configurarse utilizando los datos recopilados para poder detectar anomalías dentro de la infraestructura o la pila de aplicaciones para detectar problemas a medida que se producen, o incluso antes de que se produzcan.

VMWARE VREALIZE OPERATIONS MANAGER

VMware vRealize Operations (vROps) es la herramienta de operaciones de VMware para la infraestructura de cloud privada. Wavefront puede integrarse directamente con vROps para salvar la brecha entre las operaciones de la infraestructura y las operaciones de las aplicaciones, lo que proporciona una visibilidad completa de toda la Infraestructura.

En un entorno híbrido, vROps puede utilizarse para supervisar toda la infraestructura dentro de la cloud privada, mientras recopilan los datos recibidos a través del servicio de cloud

Wavefront para las aplicaciones empresariales críticas que se están ejecutando en Cloud. Esto puede ayudar a facilitar la operación de los gastos generales, y minimizar las diferentes herramientas utilizadas para cada entorno.

Las máquinas virtuales en las instalaciones pueden ser monitoreadas de la misma manera usando agentes de Wavefront, pero pueden ser administradas a través de vROps. El agente entonces descubre aplicaciones que pueden ser monitorizadas, aplicaciones como Apache, MySQL, Microsoft SQL y Exchange, MongoDB y RabbitMQ.

MODELO DE PRECIO DE WAVEFRONT

Como Wavefront es una aplicación basada en la nube, y el modelo de fijación de precios lo refleja, se puede utilizar en un modelo de fijación de precios basado en el consumo impulsado por la tasa de ingestión de datos. Las métricas que puede recoger y la frecuencia con la que es controlado por el cliente, y cobrado en consecuencia, pagando efectivamente por lo que se necesita.

El punto de precio, actualmente, está valorado en 1,50 dólares por punto de datos por segundo sobre una base mensual. Esto incluye métricas estándar y personalizadas, soporte de clase empresarial, y todas las capacidades avanzadas de monitoreo. Un ejemplo dado es, un host que genera 100 puntos de datos por 10 segundos costará 15 dólares por mes, basado en el uso del agente Wavefront Telegraf.

No se cobra nada por la cantidad de agentes o integraciones configuradas, no se cobra por los datos almacenados – sólo se cobra por los datos métricos entregados a Wavefront.

ARQUITECTURA DE VMWARE WAVEFRONT

El servicio de Wavefront es el que ejecuta el motor de recolección de métricas. El servicio funciona en la nube y acepta datos desde el proxy, o proxies, de Wavefront o por ingestión directa.

- Con los servicios en la nube como son AWS, Azure, GCP, etc. Wavefront extrae los datos desde el proveedor de nubes (después de una configuración mínima). Apoyamos a todos los principales proveedores de nubes.
- Para extraer métricas y datos desde on-prem tenemos varias opciones:
 - Configurar un agente colector como, por ejemplo, el gran Telegraf, que recogerá los datos de nuestro host o infraestructura, y envía esos datos al Wavefront proxy.
 - Enviar los datos de nuestro código de aplicación al Wavefront proxy usando una biblioteca de métricas. Esto funciona bien tanto para las métricas como para las trazas y los spans.
- Si tenemos una aplicación personalizada, podemos enviar sus métricas al Wavefront proxy o directamente al servicio, siempre que los datos estén en uno de los formatos de datos admitidos. Por ejemplo, si nuestro entorno ya incluye una infraestructura de recopilación de métricas, podemos realizar un preprocesamiento de los datos y enviarlos al Wavefront proxy.
- El proxy también puede ingerir las métricas de sus archivos de registro.

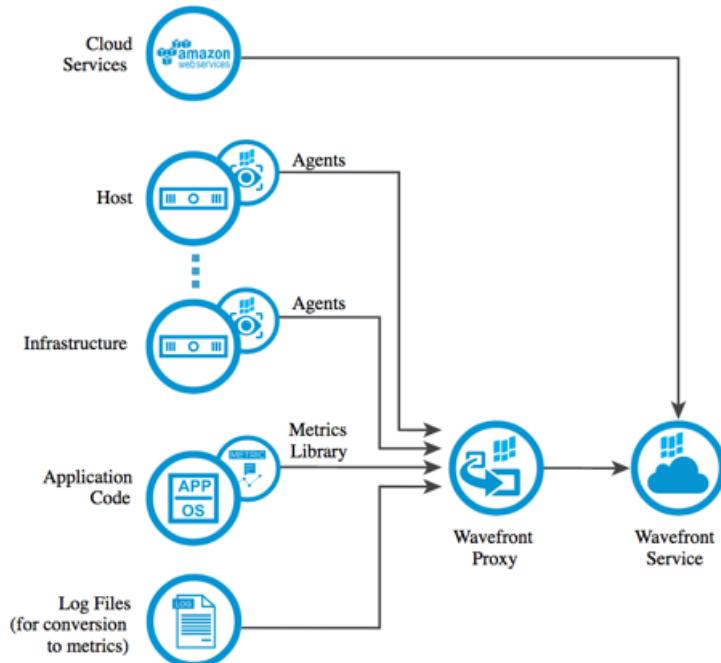
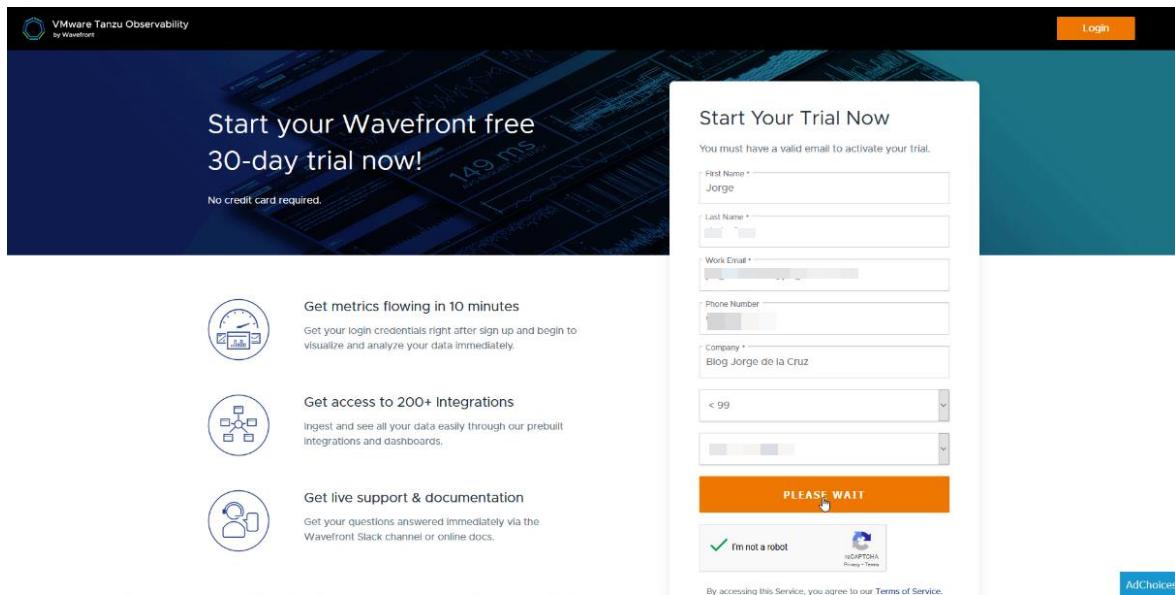


Ilustración 2 - Obtenida de la web oficial https://docs.wavefront.com/wavefront_introduction.html

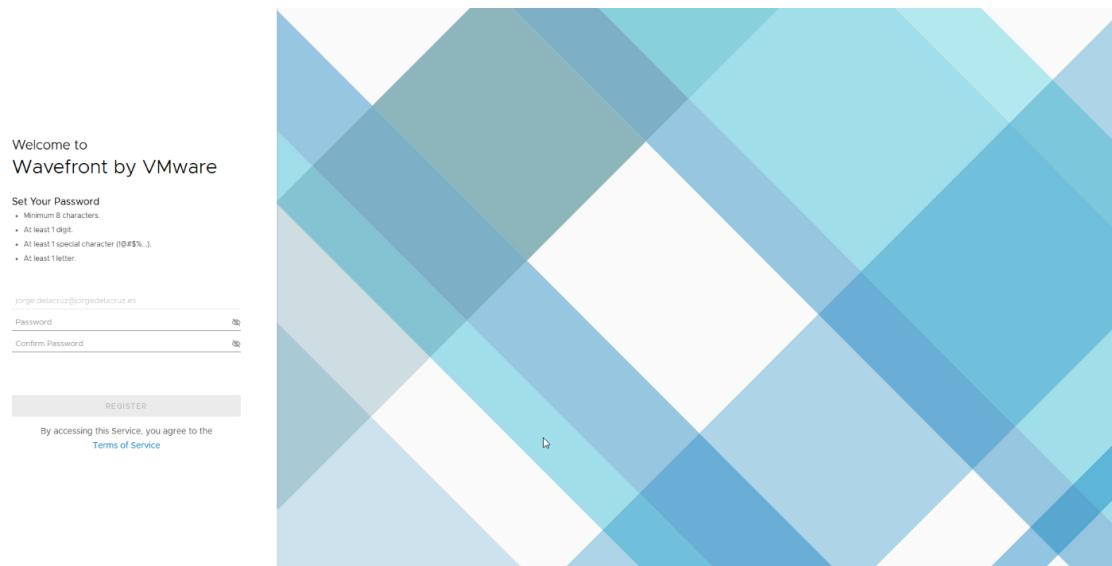
COMENZAR CON WAVEFRONT – TRIAL GRATUITA DE 30 DÍAS

Podemos comenzar una prueba de VMware Wavefront sin ningún tipo de compromiso, por 30 días, con soporte para todas las integraciones que tiene Wavefront, sin introducir ningún número de tarjeta de crédito. Como podéis imaginar, esto hace muy atractivo el servicio, y espero que os quite al miedo a probar esta brillante tecnología.

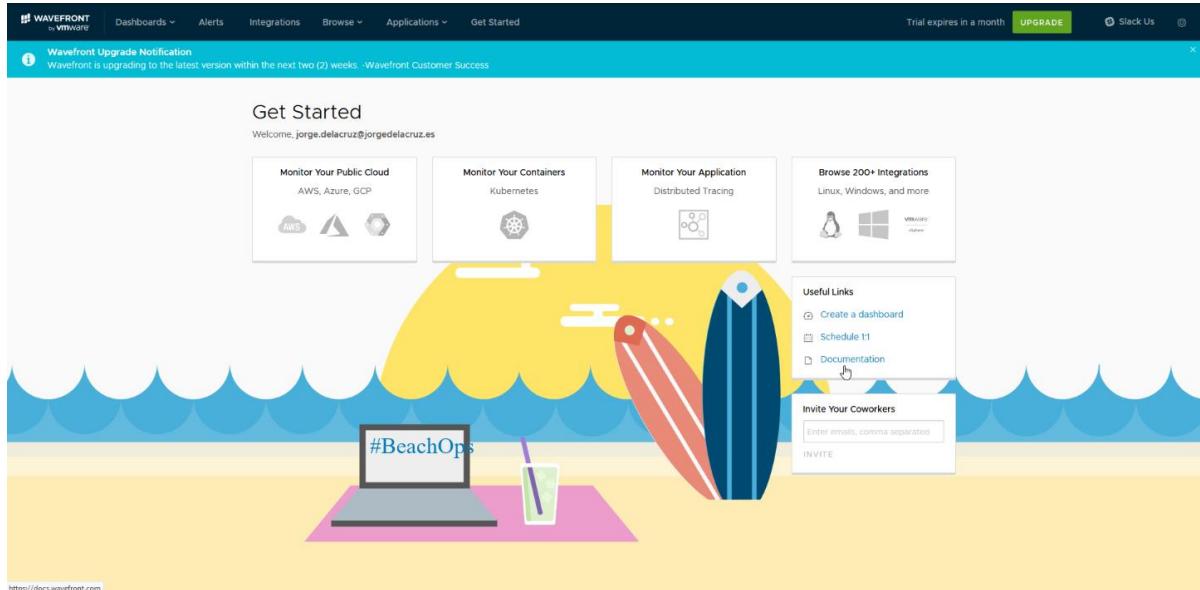
Para ello nos iremos hasta el siguiente link - <https://www.wavefront.com/sign-up/> e introduciremos nuestros datos, al menos el email tiene que ser válido para poder crear una cuenta y loguearnos, el resto de los datos lo dejo a vuestra discreción:



Crearemos ahora una contraseña para nuestra cuenta de Wavefront, con los requerimientos de complejidad adecuados:



Y ya estaremos listos, y dentro de nuestro panel de control de Wavefront. Cabe destacar que hace uso de Clarity (el framework de UI en el que se basan todos los productos de VMware), como todos los productos modernos de VMware, con lo que navegar entre los menús y opciones no se nos hace extraño.



INSTALACIÓN DE WAVEFRONT PROXY SOBRE UBUNTU 18.04 LTS

Si nos vamos a Browse – Proxies, podremos ver los Wavefront Proxies que tenemos en nuestro entorno, o en Cloud, en entornos de nuestros clientes, etc. En mi caso no tengo ninguno, con lo que haré click en añadir un nuevo Proxy:

En mi caso he optado por desplegarlo sobre una VM Linux que ejecuta Ubuntu Server 18.04 LTS, pero se podría también desplegar sobre Microsoft, Docker o Mac, además podemos escalar y desplegar tantos como necesitemos, ya que VMware no nos cobra por ello.

El comando que tendremos que ejecutar es tan sencillo como:

```
sudo bash -c "$(curl -sL https://wavefront.com/install)" --  
install \  
--proxy \  
--wavefront-url https://longboard.wavefront.com \  
--api-token TUAPIDETOKEN
```

Podremos ver algo similar a esta imagen en nuestra terminal:

```

Debian/Ubuntu
Checking installation privileges
Python detected in /usr/bin/python3
Installing python3-distutils using apt-get
Pip is not installed, installing Pip.
Wavefront CLI detected in /usr/local/bin/wave


```

Validating API Token using Wavefront URL: https://longboard.wavefront.com
Successfully validated token.
Successfully validated token.
Starting Wavefront Proxy Installation!

Y cuando todo acabe, si no ha habido ningún error o warning, veremos un output similar a éste:

```

Finished Wavefront Proxy Installation!
Starting Wavefront Proxy Configuration!
https://longboard.wavefront.com/api/
e4409c24-1e3d-41f7-89a5-ea352193b875
Restarting wavefront-proxy
Finished Wavefront Proxy Configuration!
The Proxy's configuration file can be found at /etc/wavefront/wavefront-proxy/wavefront.conf
```

De vuelta a nuestra consola de Wavefront, en Proxies, ya podremos ver que tenemos uno nuevo, listo para comenzar a enviar información de nuestro entorno, espero que haya sido sencillo ¿verdad?:

Hostname	Last Check-in	Status	Space Available	Clock Drift	Queued Items	Version	Ingestion Policy
wavefront-001	Today at 6:28 PM	Active	Gathering data ...	<1s	-	9.2	

Bien, ahora que tenemos un Wavefront Proxy en nuestro entorno, nos queda comenzar a configurarlo para que recolecte información, podemos pasar a la siguiente sección, donde veremos en detalle esta parte.

INTEGRACIÓN CON VMWARE VSphere

Ya os he contado que Wavefront hace un uso extenso de Telegraf, tanto como para VMware, Linux, Windows, o las decenas de aplicaciones que soporta, por lo tanto, para monitorizar nuestro entorno de VMware vSphere on-prem, tendremos que desplegar primero un Wavefront Proxy, que puede ser sobre Windows o Linux, y sobre este Proxy, o por separado, desplegaremos Telegraf para monitorizar VMware vSphere, vamos a ello.

Dentro de Integrations – VMware vSphere, podremos ver la pestaña de Setup, además de ver el nuevo Proxy que acabamos de desplegar, haremos un poco de scroll para ver cómo configurar esta parte:

The screenshot shows the VMware vSphere integration setup page. At the top, there's a navigation bar with links for Dashboards, Alerts, Integrations, Browse, Applications, and Get Started. On the right, it says "Trial expires in a month" and has an "UPGRADE" button. Below the navigation, there's a sidebar with the "vmware" logo and "vSphere". The main content area is titled "VMware vSphere" and "Monitor vSphere environment". It shows "METRICS" and "CONTENT". Below this, there are tabs for "Overview", "Setup" (which is selected), "Metrics", and "Dashboards". The "Setup" tab contains sections for "vSphere Setup" and "Step 1. Install the Telegraf Agent". The "vSphere Setup" section includes notes about vSphere metrics being extensive and recommends using a dedicated VM for Telegraf. It also notes that network connectivity to vCenter Server is required. The "Step 1" section provides instructions for installing the Telegraf Agent, mentioning the use of the vSphere input plugin. A "Select Wavefront Proxy" step follows, showing a dropdown menu with "wavefront-001" and an "ADD NEW PROXY" button. Finally, there's an "Install Telegraf Agent" section with a command-line instruction: "Run the following command on the host that you want to install the agent on." The command is: `curl -sL https://wavefront.com/install | bash`

INSTALACIÓN Y CONFIGURACIÓN DE TELEGRAF EN WAVEFRONT PROXY

Para los que hayáis leído el primer libro de VMware por vExperts, o a los que sigáis mi serie sobre Dashboards de Grafana, no os resultará extraño este paso. Lo que VMware Wavefront nos requiere es instalar Telegraf dentro de este Wavefront Proxy, de la siguiente manera:

```
sudo bash -c "$(curl -sL https://wavefront.com/install)" --  
install \  
--agent \  
--proxy-address ELNOMBREDETUPROXY \  
--proxy-port 2878
```

Una vez que tenemos Telegraf instalado, crearemos un fichero llamado vsphere.conf dentro de /etc/telegraf/telegraf.d/ con el siguiente contenido, cambiando la IP o FQDN de vuestro VCSA, así como el user y password (solamente se requiere read-only al nivel más alto del VCSA y sus children)

```
## Monitorizacion en tiempo real, ESXi y VMs
[[inputs.vsphere]]

## List of vCenter URLs to be monitored. These three lines must
be uncommented

## and edited for the plugin to work.

interval = "20s"
vccenters = [ "https://someaddress/sdk"]
username = "someuser@vsphere.local"
password = "secret"
vm_metric_include = []
host_metric_include = []
datastore_metric_exclude = ["*"]
max_query_metrics = 256
timeout = "60s"
insecure_skip_verify = true

## Monitorizacion historica, Datacenter, Cluster y Datastore
[[inputs.vsphere]]
interval = "300s"
vccenters = [ "https://someaddress/sdk"]
username = "someuser@vsphere.local"
password = "secret"
datastore_metric_include      =      [      "disk.capacity.latest",
"disk.used.latest", "disk.provisioned.latest" ]
cluster_metric_include = []
datacenter_metric_include = []
insecure_skip_verify = true
force_discover_on_init = true
host_metric_exclude = ["*"] # Exclude realtime metrics
```

```

vm_metric_exclude = ["*"] # Exclude realtime metrics

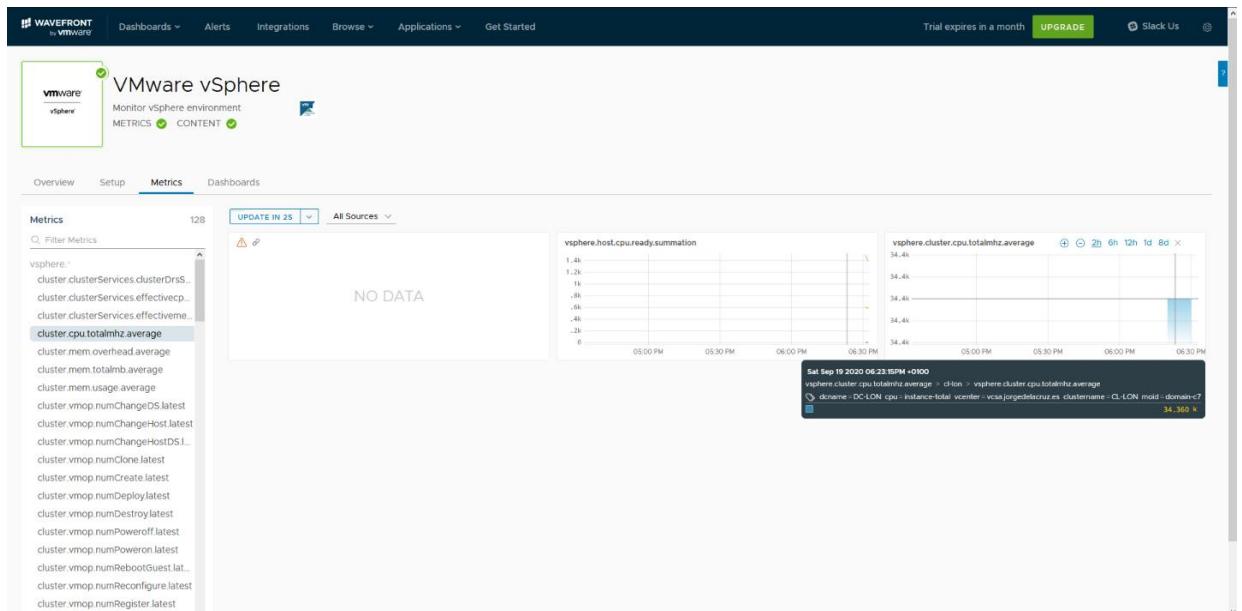
max_query_metrics = 256
collect_concurrency = 3

```

Reiniciamos el servicio de Telegraf y ya estaríamos listos para recolectar información directamente desde vCenter a este Telegraf, que enviará las métricas a Wavefront:

```
sudo service telegraf restart
```

Pasados apenas unos segundos, podremos ya ver información dentro de nuestro Wavefront, en la pestaña de Metrics:



VISUALIZACIÓN EN DETALLE DE VMWARE VSPPHERE CON DASHBOARDS DE WAVEFRONT

He dejado recolectando información suficiente por unas 24 horas para mostráros estos fantásticos Dashboards con mucha más información.

En la sección de Integrations - vSphere, podremos encontrar una pestaña llamada Dashboards, haremos click en ella, encontramos todos los siguientes Dashboards ya listos para ser consumidos:

The screenshot shows the Wavefront interface with the VMware vSphere integration installed. The top navigation bar includes links for Dashboards, Alerts, Integrations, Browse, Applications, and Get Started. A trial expiration notice and upgrade buttons are also present. The main content area is titled "VMware vSphere Dashboards" and lists five available dashboards:

- vSphere: Summary
- vSphere: Cluster
- vSphere: ESXi Host Summary
- vSphere: ESXi Host Details
- vSphere: VM Summary
- vSphere: VM Details
- vSphere: Datastore

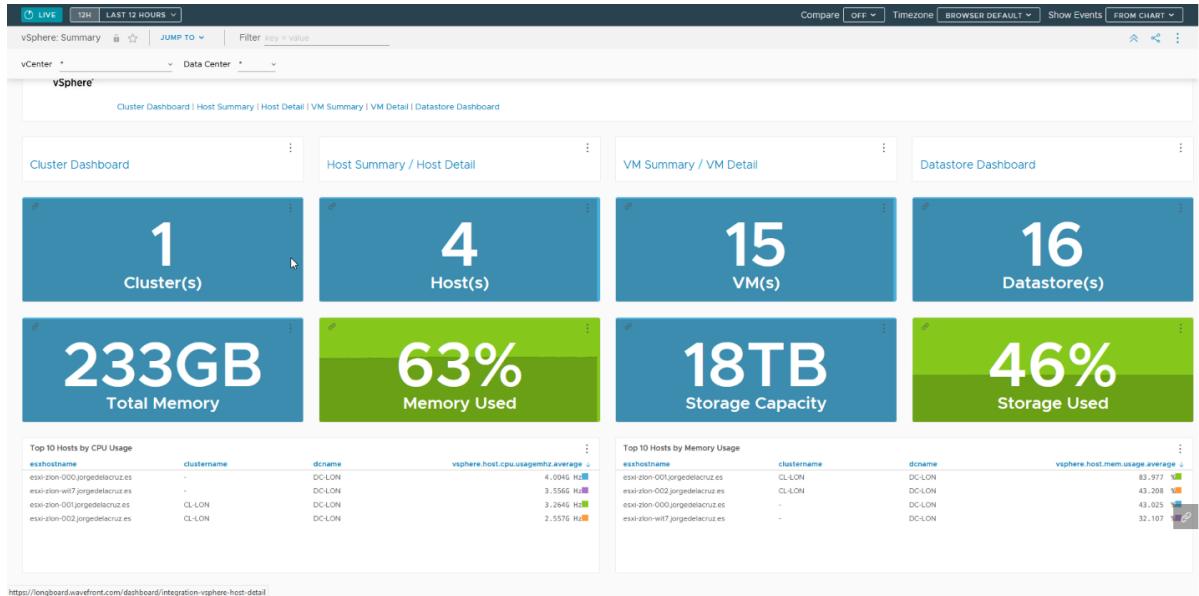
Below the dashboards, a message states "Dashboards are installed." and a "UNINSTALL DASHBOARDS" button is visible. The URL <https://longboard.wavefront.com/dashboards/integration-vsphere-summary> is shown at the bottom.

Vamos a ver detalles de cada uno de ellos, con métricas de las últimas 12 horas, por poneros un ejemplo. Me gusta que Wavefront tiene todo esto ya creado y funcionando sin necesidad de crear nosotros nada adicional.

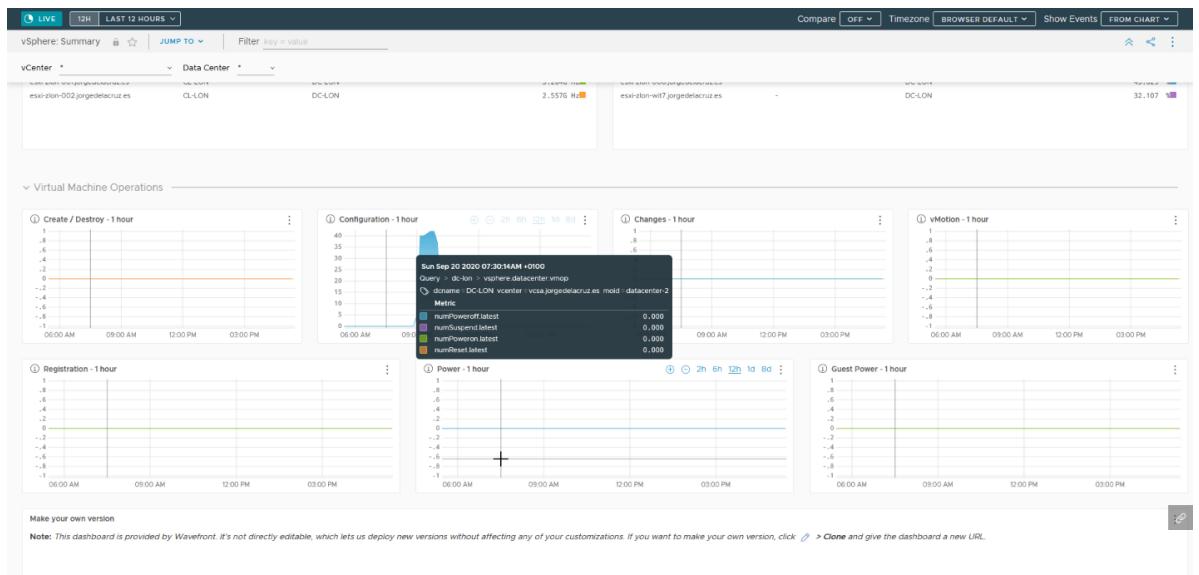
WAVEFRONT vSPHERE: SUMMARY DASHBOARD

Este Dashboard nos muestra de un vistazo rápido de todo nuestro entorno, podemos encontrar de manera sencilla; cuántos Clúster tenemos, Hosts, VMs, Datastores, así como el detalle de RAM total, el % de memoria utilizada y Storage.

De manera muy rápida y en una cómoda tabla podremos ver el Top 10 de Hosts consumiendo CPU y consumiendo Memoria RAM:



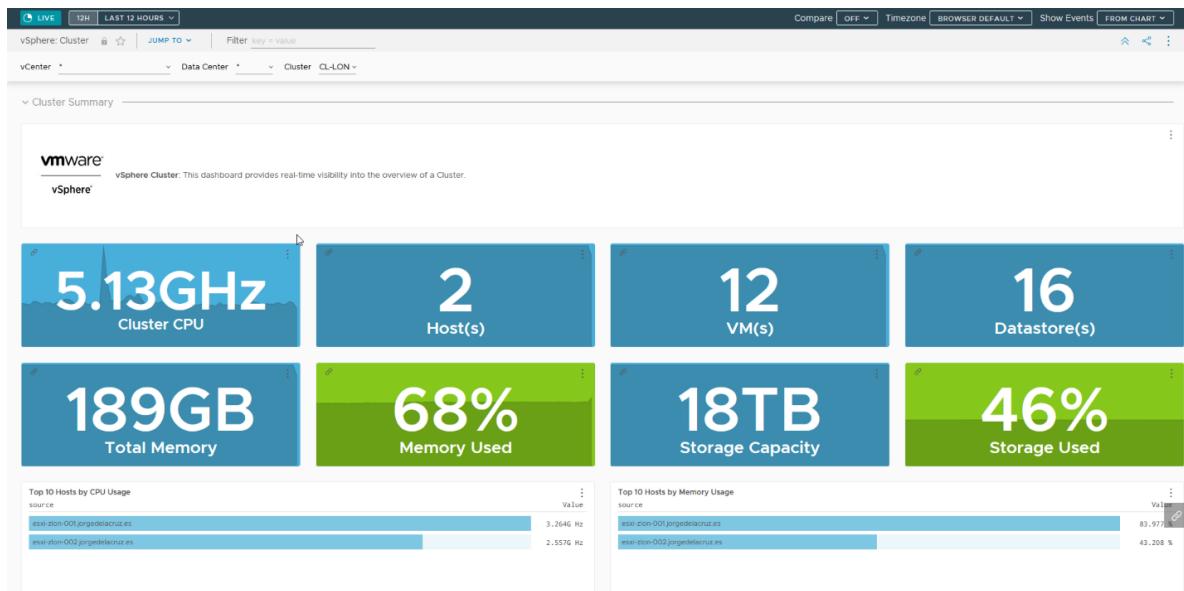
Además, nos mostrará todas las acciones que nuestro clúster ha ejecutado en el periodo de tiempo seleccionado, me refiero a vMotions, VMs creadas y destruidas, cambios en la configuración de las VMs, VMs que hemos encendido, etc.:



WAVEFRONT vSPHERE: CLÚSTER DASHBOARD

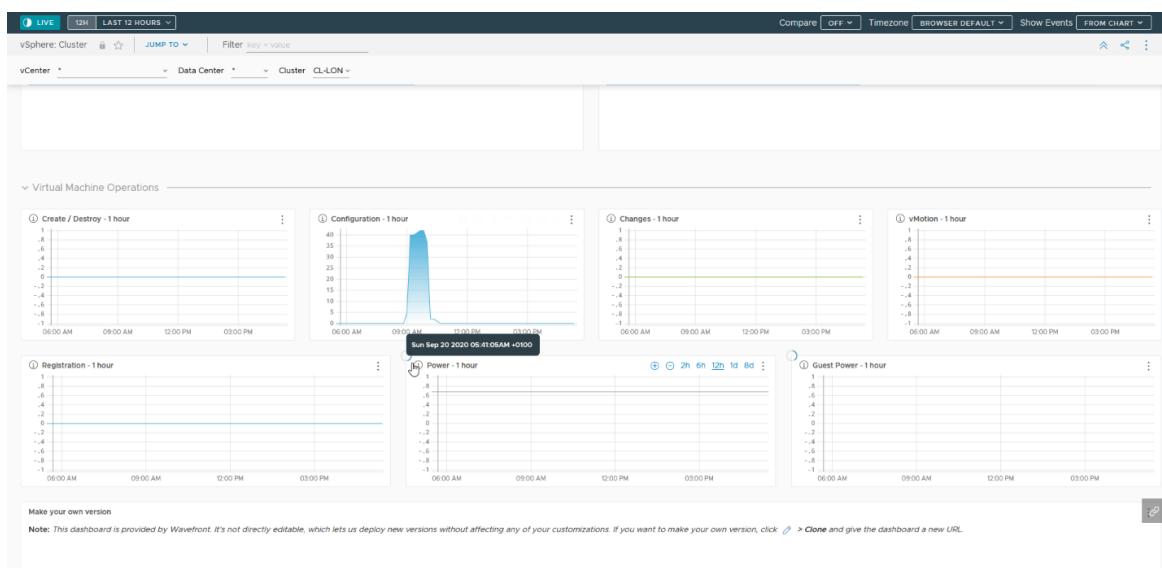
Este Dashboard es muy parecido al anterior, pero dedicado a la parte de Clúster, podremos filtrar por clúster para tener una visibilidad mucho más detallada.

Podremos observar el consumo de CPU del clúster, número total de Hosts, VMs, Datastores, y el porcentaje de RAM y Storage consumido:



Además, en este Dashboard podremos rápidamente conocer que Hosts están consumiendo más CPU y más RAM.

Si hacemos algo de scroll, podremos ver todas las acciones que nuestro clúster ha ejecutado en el periodo de tiempo seleccionado, me refiero a vMotions, VMs creadas y destruidas, cambios en la configuración de las VMs, VMs que hemos encendido, etc.:

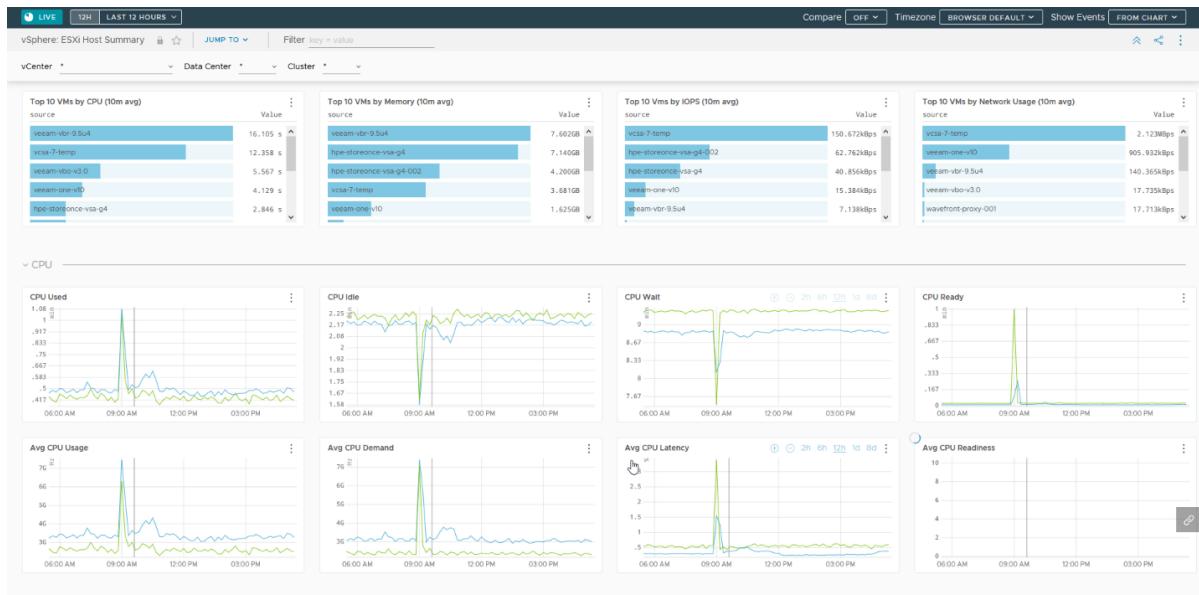


WAVEFRONT vSPHERE: ESXi Host Summary Dashboard

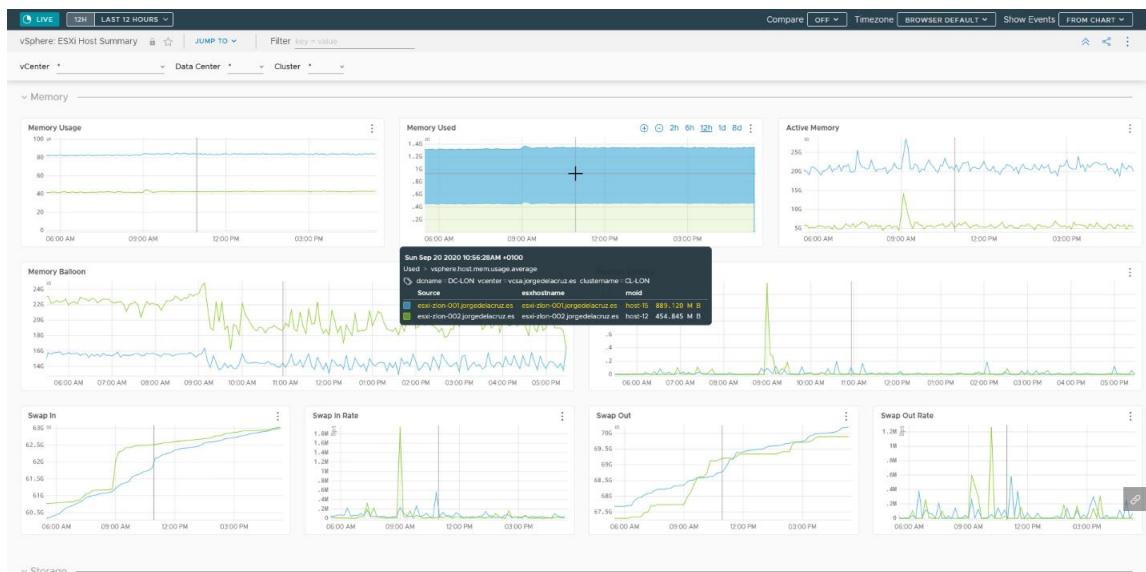
Comenzamos ahora los Dashboards que entran más en detalle, con muchísimas más métricas y detalles que nos harán la vida más sencilla. Este Dashboard sobre la visibilidad de ESXi nos mostrará de manera muy sencilla todo lo que necesitamos conocer sobre ESXi y las VMs que se ejecutan sobre los mismos.

Por ejemplo, en la primera parte del Dashboard, podemos conocer el Top 10 de VMs que consumen más CPU, RAM, IOPS y consumo de Red, tan sencillo como eso.

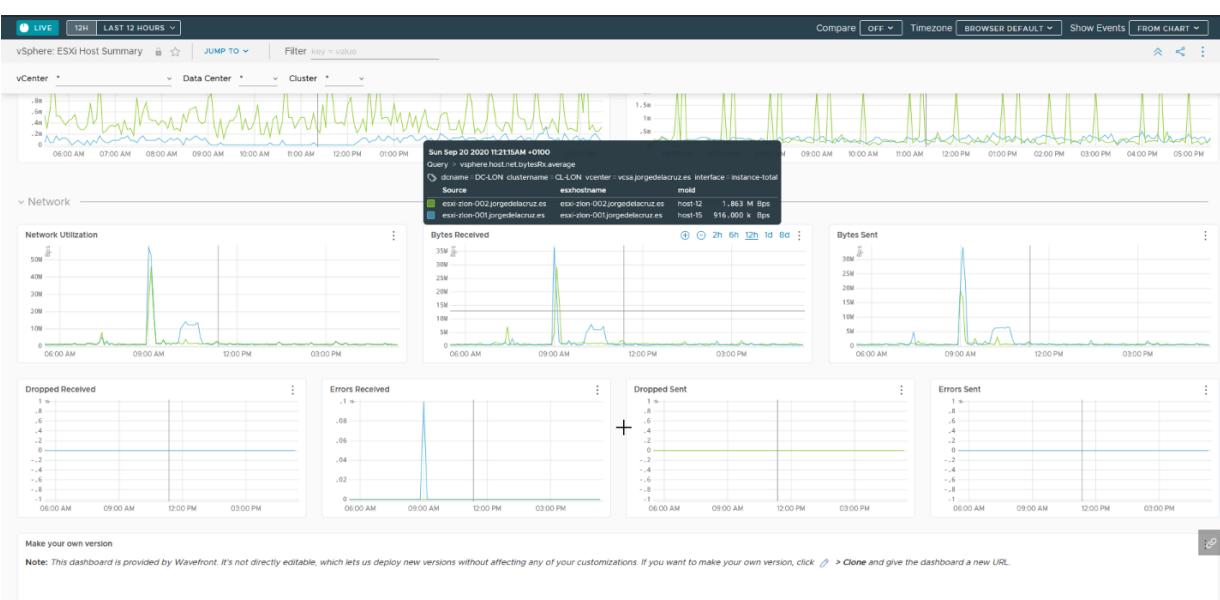
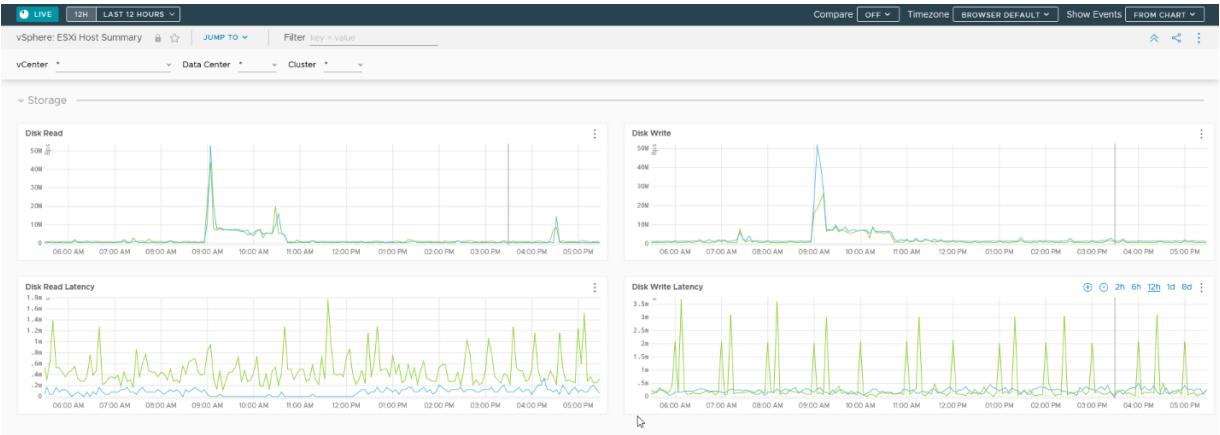
Un poco más abajo encontramos el consumo de CPU de los ESXi en sí, lo que el hipervisor está consumiendo en el rango de tiempo que hemos definido:



Encontramos, si hacemos un poco de scroll, el consumo detallado de memoria RAM de cada Host ESXi, con gráficas muy simples de interpretar y con todo lujo de detalle:

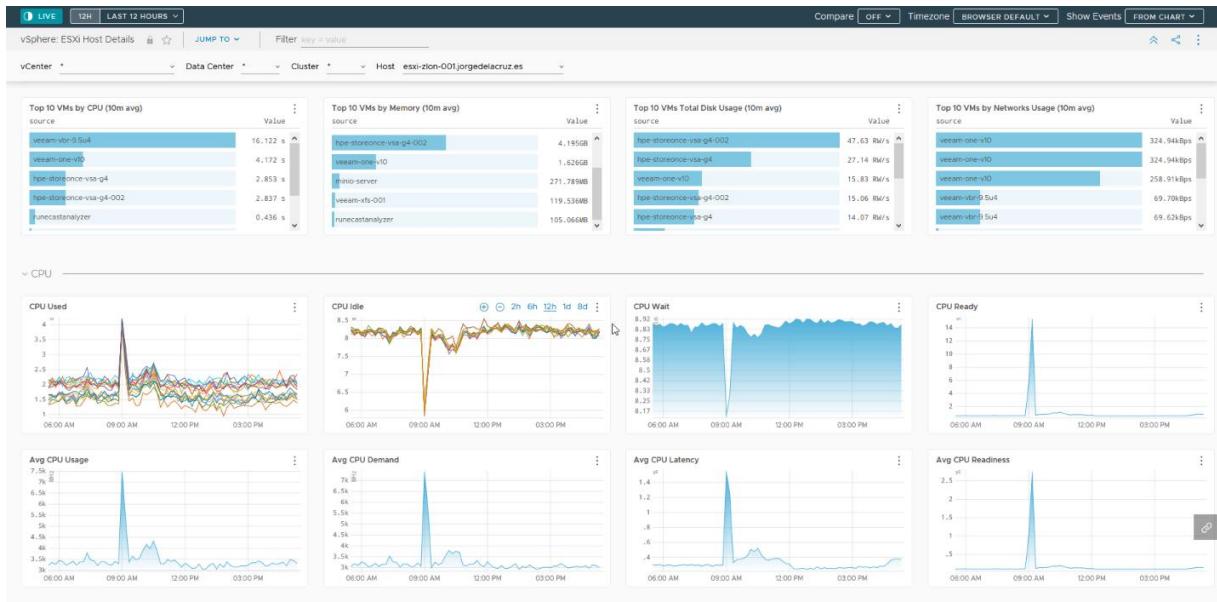


La última parte del Dashboard está dedicada al Storage y el consumo de red, ambas secciones con mucho detalle sobre cada ESXi, y que nos permitirá, con un simple vistazo, conocer en qué momento alguno de los Hosts ESXi están teniendo más consumo de Red, o de Storage IOPS, etc. Lo que nos permitirá correlacionarlo con las métricas de las VMs y ver qué VM está consumiendo esos recursos tanpreciados:

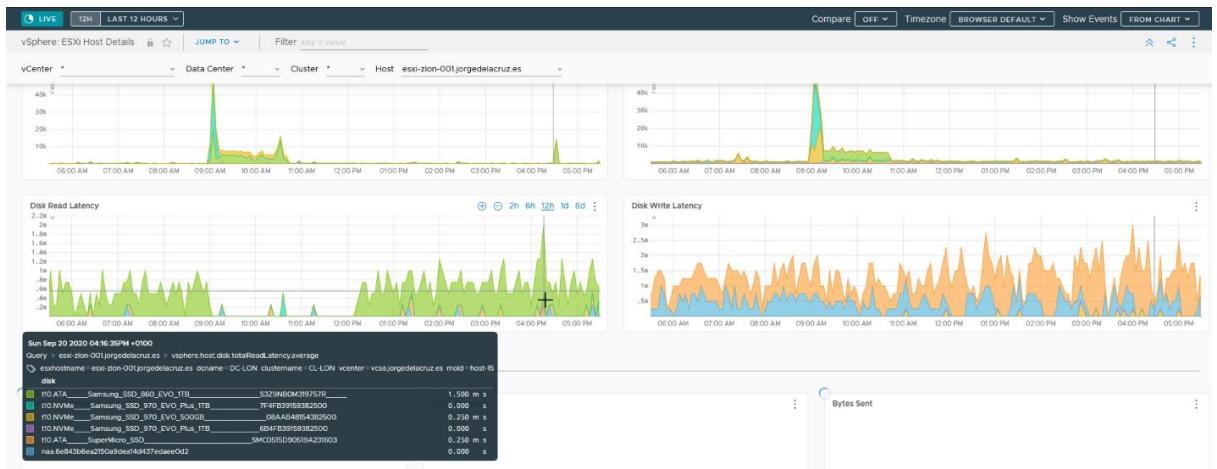


WAVEFRONT vSPHERE: ESXi Host Details Dashboard

Este Dashboard es muy parecido al anterior, pero esta vez entramos en un detalle sin precedentes a nivel de monitorización, encontramos mucho más detalle en cada una de las gráficas por componente. Además, nos muestra como siempre el Top 10 de las VMs que más CPU, RAM, Disco y Red consumen:

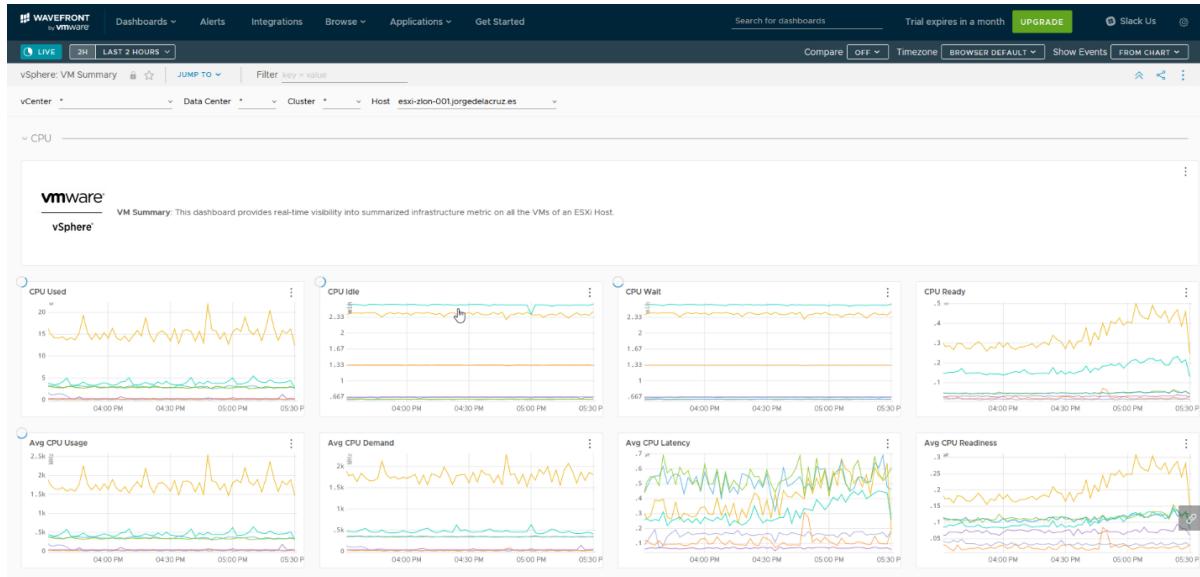


Cuando hablamos de detalle, me refiero por ejemplo a conocer el disco físico que está teniendo más READ o WRITE latency, y mucho más por supuesto, os dejo un ejemplo:

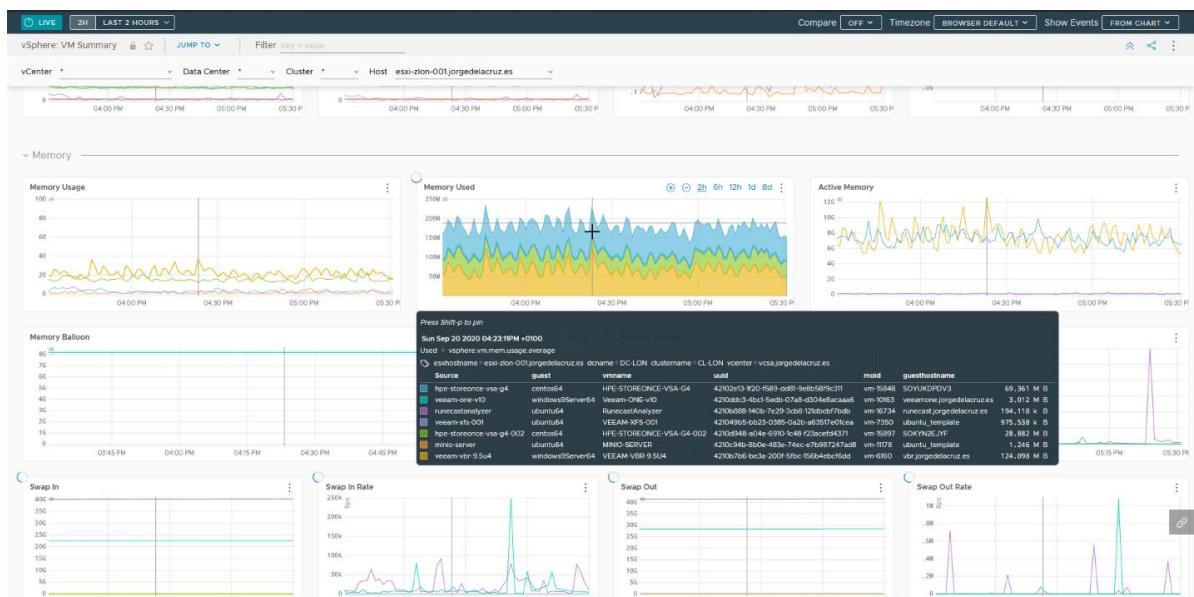


WAVEFRONT vSPHERE: VM SUMMARY DASHBOARD

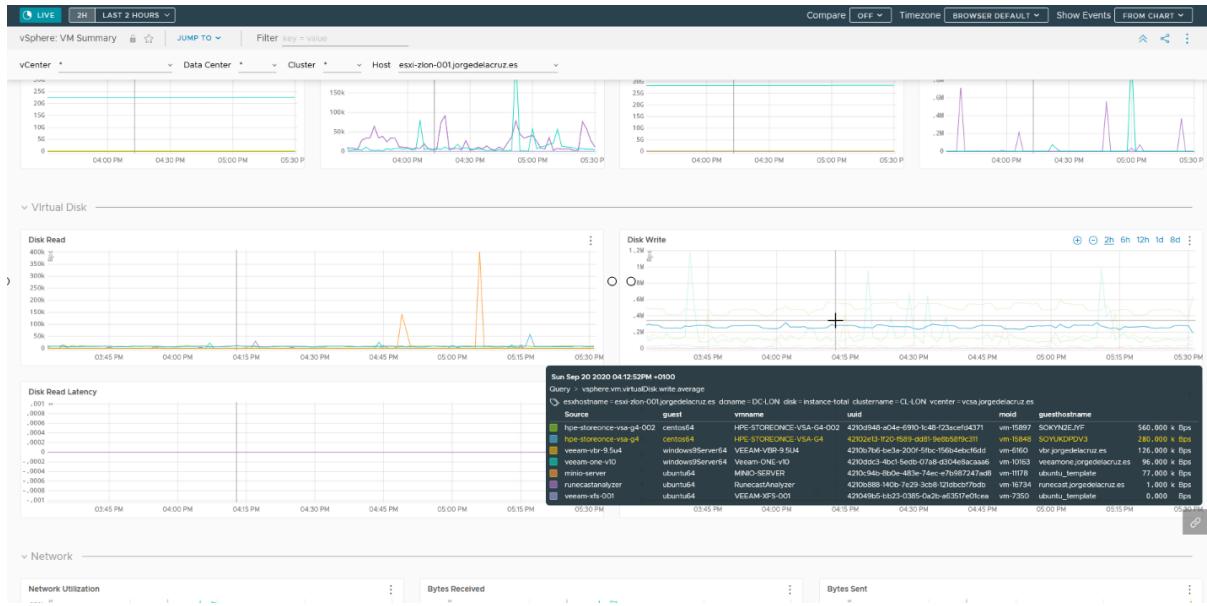
En este Dashboard podremos comprobar el consumo de recursos de las VMs, agrupado por Hosts de ESXi, con lo que tenemos una visión mucho más concreta de qué VM está consumiendo qué recurso, podremos filtrar por Host de ESXi:



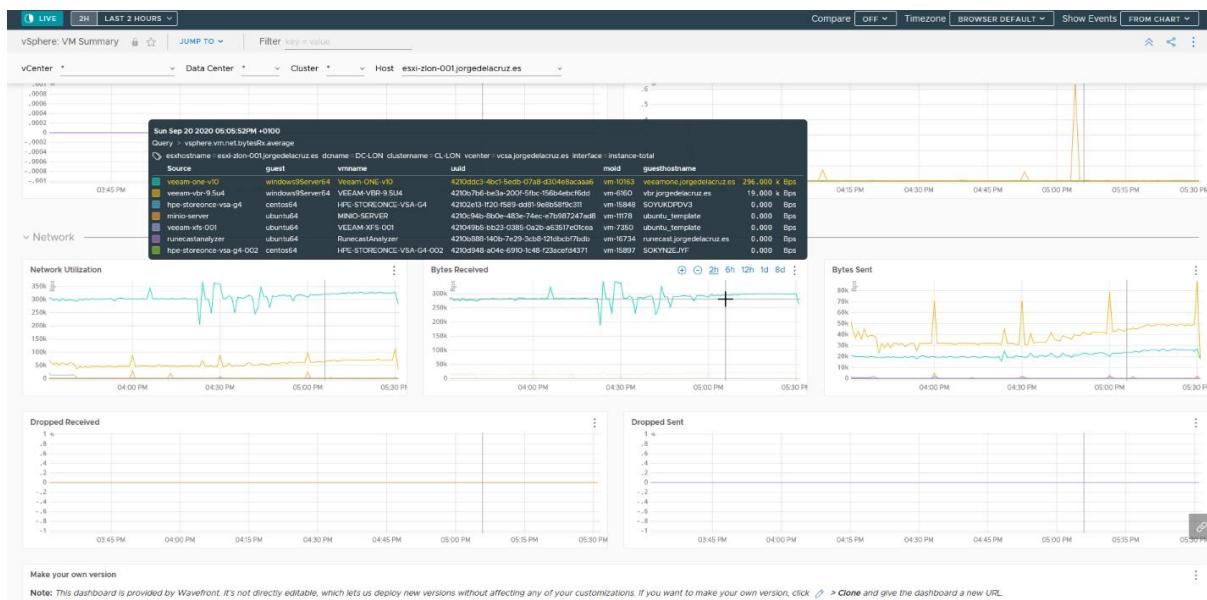
Os muestro ahora un ejemplo de la memoria RAM consumida, con detalle, por cada VM, realmente importante y sencillo de comprobar:



Podremos encontrar también muchísimo detalle de todo lo que, a Storage, y Storage latency se refiere, por VM, de manera que, si algún Ransomware entra en nuestras VMs, podremos rápidamente identificar qué VM está cifrando contenido, ya que esto aumenta la latencia de disco por supuesto:



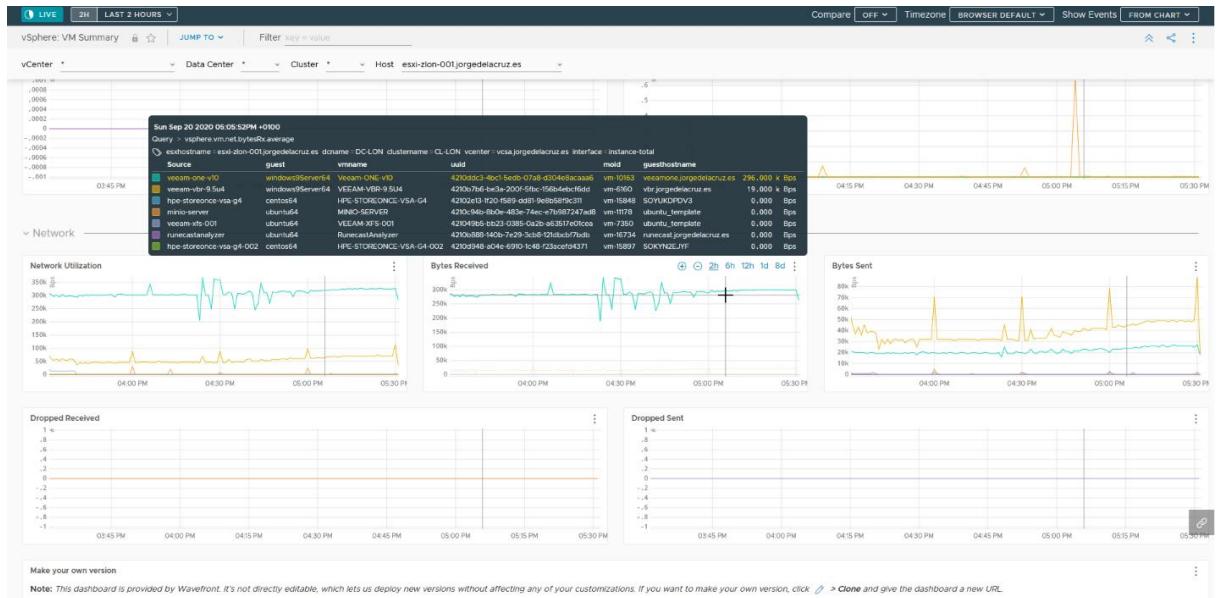
Por último, en la parte de Networking, como siempre por VM, podremos comprobar qué VM está enviando o recibiendo más paquetes, o está perdiendo paquetes de red en el envío o al recibirlos:



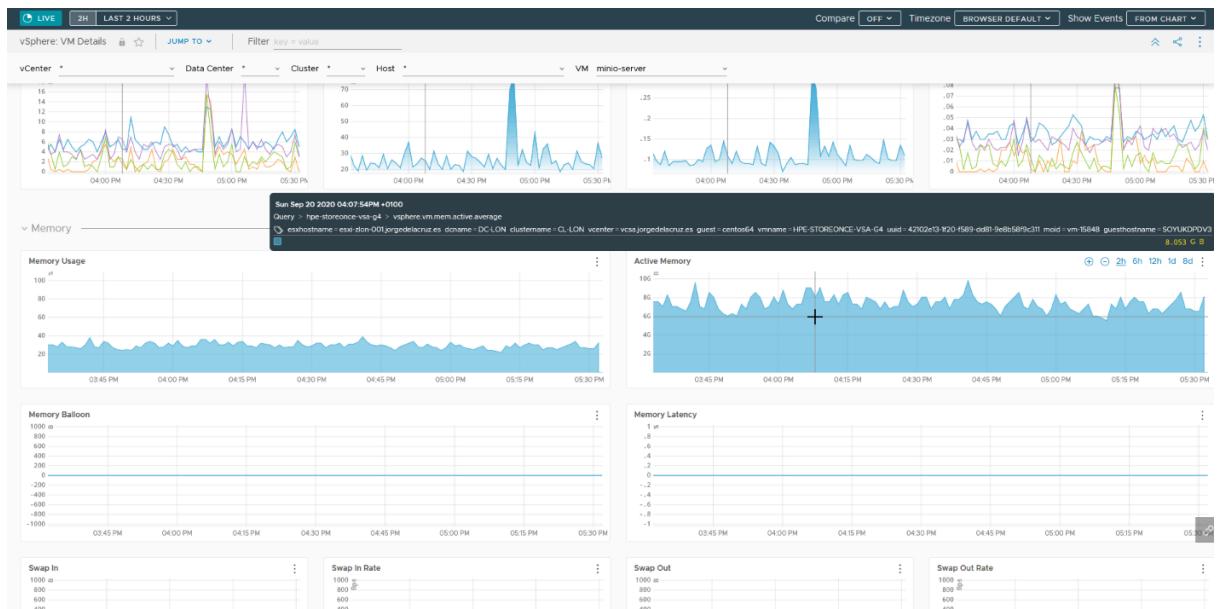
WAVEFRONT vSPHERE: VM DETAILS DASHBOARD

Llegamos a los dos últimos Dashboards, los más interesantes cuando queremos hacer troubleshooting, o si estamos facturando a clientes por recursos consumidos, etc.

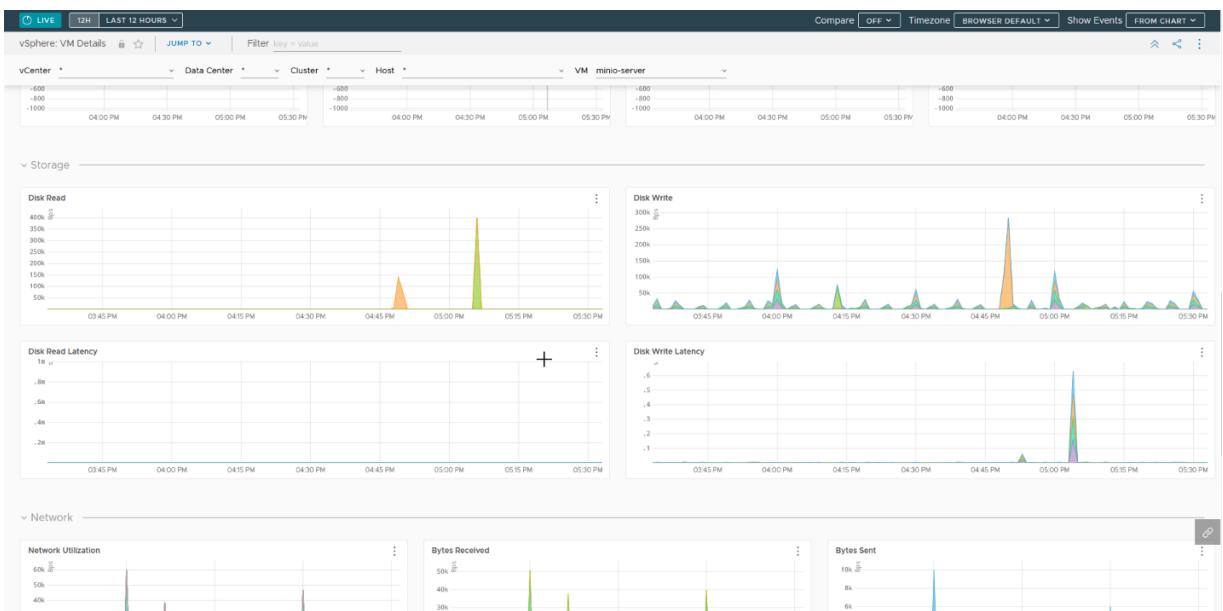
Podemos filtrar por la VM que queremos mostrar, y en la primera sección de la gráfica tenemos todo lo que a CPU se refiere, cosas tan importantes de comprobar como es el CPU Ready, o el CPU Readiness, además de latencia de CPU, o por supuesto consumo:



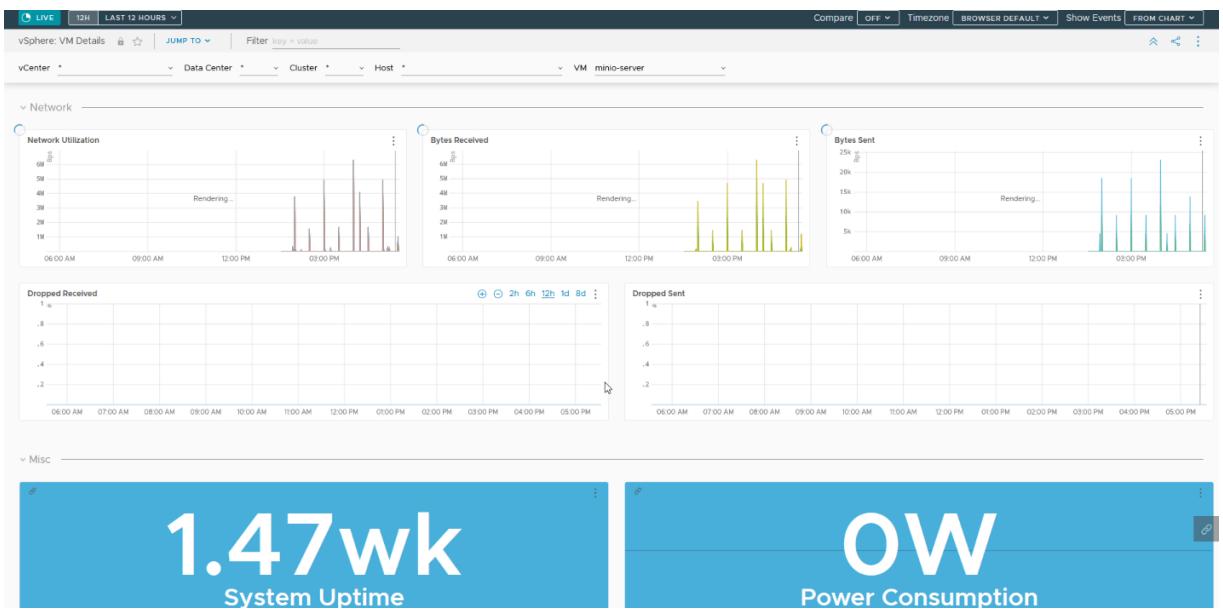
A nivel de memoria RAM, lo más importante es conocer si estamos haciendo SWAP, o qué tal se encuentra el ballooning, y por supuesto, consumo de memoria RAM:



En la parte de Storage, tenemos cuatro gráficas muy críticas cuando estamos haciendo troubleshooting, consumos y métricas sobre las lecturas y escrituras de disco, así como latencia en escrituras y lecturas:



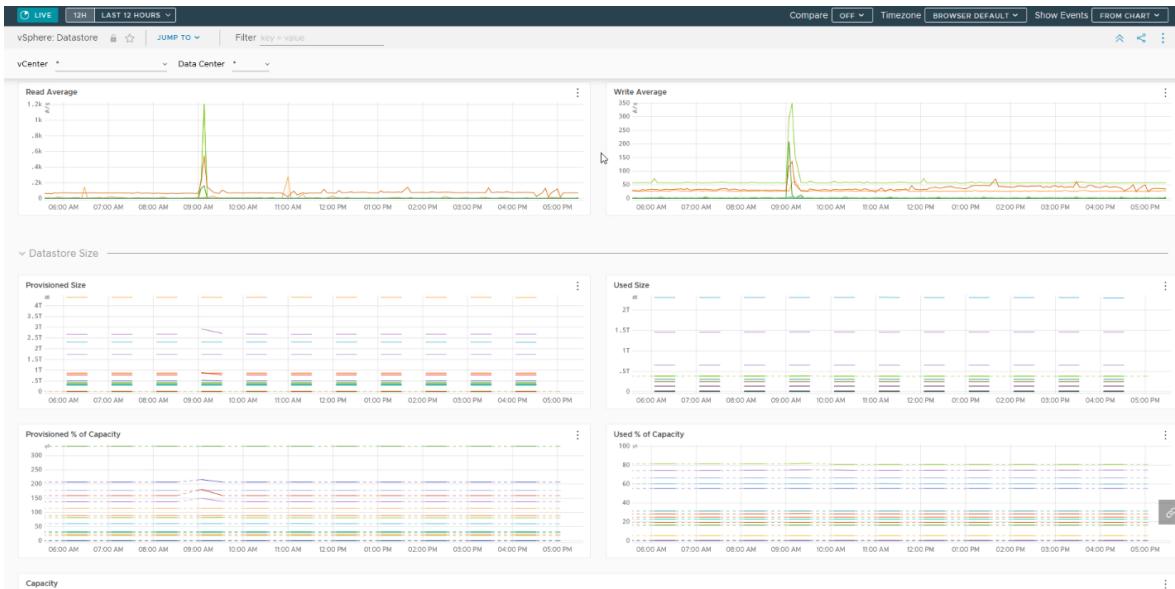
Por último, encontramos la parte de Networking, donde se puede comprobar de manera muy sencilla los paquetes que la VM ha enviado, recibido, así como paquetes perdidos, etc.:



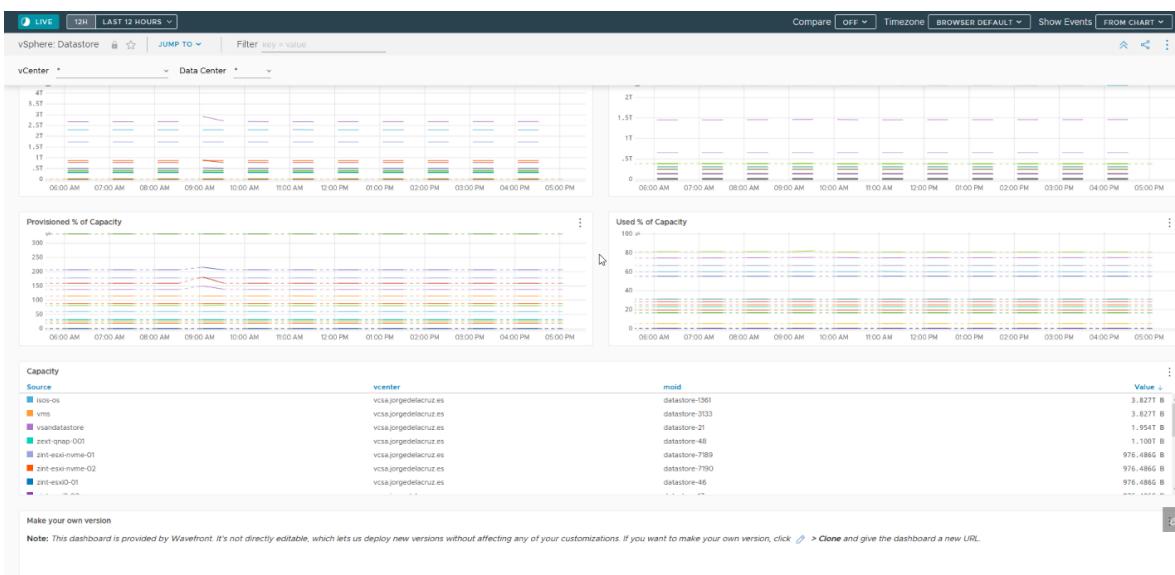
WAVEFRONT vSPHERE: DATASTORE DASHBOARD

Es cierto que a nivel de Datastore podríamos esperar un poco más, especialmente cuando estamos monitorizando VMware VSAN, pero de momento, Wavefront no incluye ningún Dashboard predeterminado para esta tarea, con lo que tendremos que conformarnos con los Dashboards existentes.

A vista de pájaro, podemos encontrar las lecturas y escrituras por cada Datastore, así como el uso de estos, tanto en GB como en porcentaje:



Encontramos también una tabla muy cómoda con la capacidad total de disco por Datastore:



INTEGRACIÓN NATIVA CON AMAZON WEB SERVICES (AWS)

Hemos visto la parte más “complicada”, desplegando Wavefront Proxies, configurando Telegraf, etc. Ahora vamos a ver cómo añadir Cloud Públicas a Wavefront, para que tengamos una visibilidad sin precedentes de estos “agujeros negros” económicamente hablando. Y no es una exageración, ya que tanto como en AWS, Azure, GCP, etc., es muy sencillo empezar a crear buckets de storage, VMs, ahora incluso PaaS o similares, y acabar con una factura que en algunos casos puede repercutir mucho en una PYME.

Con lo que tener una visualización como la que nos da Wavefront, sobre todo nuestro entorno, para mí, está más que justificada.



CONFIGURACIÓN DE INTEGRACIÓN DE AWS EN WAVEFRONT PROXY

Wavefront nos lo pone muy sencillo cuando se trata de añadir para monitorizar nuestras Cloud Públicas, son apenas unos pasos que en este capítulo vamos a ver, al menos para AWS, pero son similares para Azure.

Dentro de la integración para Amazon Web Services, haremos click en Settings – Add Integration:

A screenshot of the Wavefront interface showing the AWS integration setup page. It displays the AWS logo, service names like Lambda, S3, and CloudWatch Metrics, and a table for adding new integrations. The 'Overview' tab is selected.

Para ello, nos pide que introduzcamos un Role ARN desde Amazon IAM:

The screenshot shows the 'Amazon Web Services' configuration page within the Wavefront interface. On the left, there's a form to 'Give Wavefront read-only access to your Amazon account'. It has fields for 'Name' (set to 'AWS') and 'Role ARN' (set to 'arn:aws:iam::30121311993:role/Wavefront'). Below these are 'REGISTER' and 'BACK' buttons. To the right, a sidebar titled 'How to get Role ARN' provides a step-by-step guide:

- Step 1:** Create a Role in AWS. It says to navigate to AWS Identity and Access Management (IAM), select 'Roles' in the left menu, and click 'Create Role'. It also notes that the External ID is time-sensitive and unique per account setup.
- Step 2:** Find the role that was just created and click it. Then find the 'Role ARN' value and paste it into the form on the left.

A la derecha, nos muestra de manera muy concisa cómo crear este Rol, pero ya que estamos en un libro didáctico, vamos paso a paso, dentro de nuestra consola de AWS, haremos click en Create Role, una vez que estamos en la sección de Identity and Access Management (IAM):

The screenshot shows the 'Roles' section of the AWS IAM service. The left sidebar lists various IAM management options like Dashboard, Access management, Groups, Users, Roles (which is selected), Policies, Identity providers, Account settings, and more. The main content area is titled 'Roles' and contains a 'What are IAM roles?' section with a 'Create role' button at the bottom. A green arrow points to this 'Create role' button.

Seleccionaremos que es del tipo “Another AWS account”, introduciremos el número de cuenta de AWS que Wavefront nos da, más la opción llamada “Require external ID”, donde introduciremos el ID único que Wavefront nos da también en la parte de Settings, debe de quedar algo similar a esto, haremos click en Permissions si todo está bien:

Screenshot of the AWS IAM 'Create role' wizard Step 1: Select type of trusted entity.

The 'Another AWS account' option is selected. The 'Account ID' field contains '301213811993'. The 'External ID' field contains 'JEo243tJznbExpPz'. A note states: 'Important: The console does not support using an external ID with the Switch Role feature. If you select this option, entities in the trusted account must use the API, CLI, or a custom federation proxy to make cross-account iam AssumeRole calls. Learn more.'

Buttons at the bottom: * Required, Cancel, Next: Permissions.

En la parte de “Permissions” seleccionaremos el grupo que se llama **ReadOnlyAccess**, hay que hacer algo de scroll hasta encontrarlo, y podremos ir a Tags, donde haremos Next sin añadir nada:

Screenshot of the AWS IAM 'Create role' wizard Step 2: Attach permissions policies.

The 'ReadOnlyAccess' policy is selected and highlighted with a green arrow. Other policies listed include GlobalAcceleratorReadOnlyAccess, IAMAccessAnalyzerReadOnlyAccess, IAMReadOnlyAccess, NeptuneReadOnlyAccess, ResourceGroupandTagEditorReadOnlyAccess, ServiceQuotaReadOnlyAccess, and WellArchitectedConsoleReadOnlyAccess.

Buttons at the bottom: Create policy, Set permissions boundary.

Nos quedaría introducir un nombre para este nuevo Role, en mi caso he sido bastante descriptivo:

Screenshot of the AWS IAM 'Create role' wizard Step 4: Review.

The role name is 'Wavefront'. The role description is empty. Policies attached are 'ReadOnlyAccess'. Trusted entities are 'The account 301213811993'. The permissions boundary is not set.

Y ya lo tendríamos, si hacemos click en el nombre del Role que acabamos de crear, podemos encontrar finalmente nuestro Role ARN, que nos servirá para configurar Wavefront, podremos copiarlo haciendo click en el icono:

Role ARN	arn:aws:iam:971119770011:role/Wavefront
Role description	Edit
Instance Profile ARNs	
Path	/
Creation time	2020-09-19 18:42 UTC+0100
Last activity	Not accessed in the tracking period
Maximum session duration	1 hour Edit
Give this link to users who can switch roles in the console	
https://signin.aws.amazon.com/switchrole?roleName=Wavefront&account=971119770011	

Y de vuelta a Wavefront, introduciremos nuestro ARN que acabamos de crear, así de fácil:

How to get Role ARN

Step 1: Create a Role in AWS
Navigate to AWS Identity and Access Management (IAM). Select **Roles** in the left menu and click **Create Role**.

In Step 1: Trust
Select **Another AWS account**. Enter `301213811983` into Account ID. Enter `30a243tUzlbExpPz` into External ID. The External ID is time-sensitive and unique per account setup.

In Step 2: Permissions
Search for **ReadOnlyAccess** in the list and check it.

Si todo ha ido bien, podremos ver el fantástico mensaje de Wavefront indicándonos que empezaremos a recibir métricas durante los próximos cinco minutos.

Name	RoleARN	Types	State
AWS	arn:aws:iam:971119770011:role/Wavefront	AWS Metrics CloudWatch	Active Active

Espero que la configuración os haya parecido sencilla, son algunos pasos, y tenemos que conocer AWS un poco, pero bien es cierto que al final es crear un nuevo Role con permisos de lectura y poco más.

VISUALIZACIÓN EN DETALLE DE TODO NUESTRO AWS CON DASHBOARDS DE WAVEFRONT

He dejado recolectando información suficiente por unas 24 horas para mostráros estos fantásticos Dashboards con mucha más información.

En la sección de Integrations – Amazon Web Services, podremos encontrar una pestaña llamada Dashboards, haremos click en ella, encontramos todos los siguientes Dashboards ya listos para ser consumidos:

The screenshot shows the Wavefront interface for AWS. At the top, there's a navigation bar with links for Dashboards, Alerts, Integrations, Browse, Applications, and Get Started. A green arrow points to the 'Dashboards' link. On the right side of the header, there's a message about a trial expiring in a month, an 'UPGRADE' button, and a 'Slack Us' button. Below the header, there's a section titled 'Amazon Web Services' with sub-links for 'Monitor AWS services', 'METRICS', and 'CONTENT'. A large grid of icons represents different AWS services. The main content area is titled 'Amazon Web Services Dashboards' and lists 15 different AWS services, each with a thumbnail, name, and a brief description. Two specific dashboards are highlighted with red arrows: 'AWS: Summary' and 'AWS: Billing'.

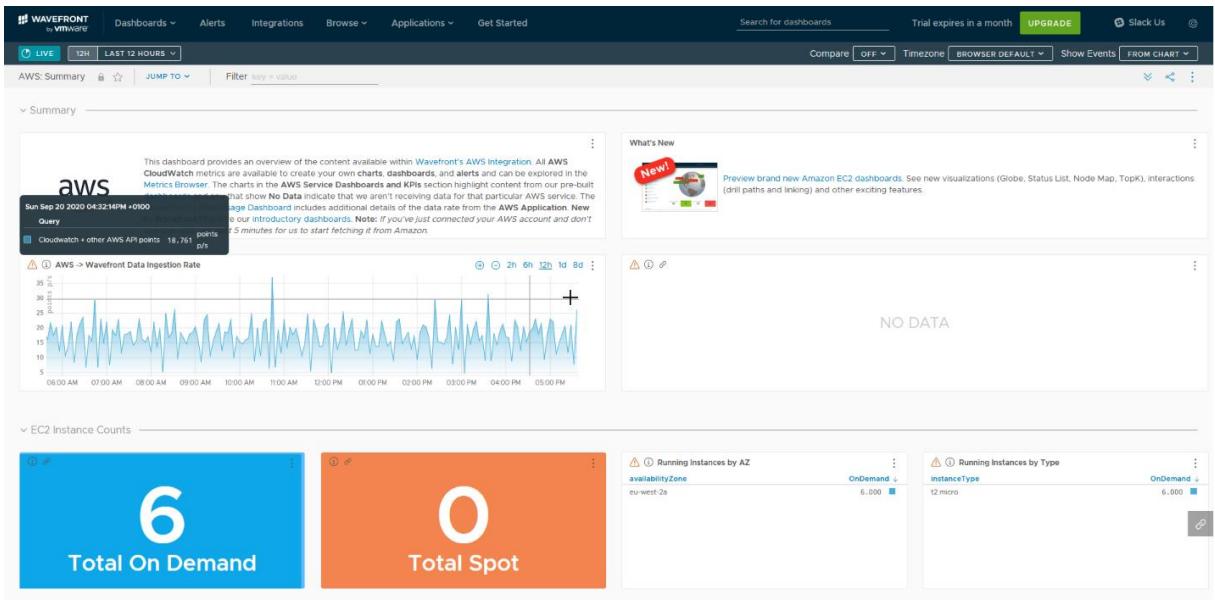
Service	Description
AWS: Summary	View AWS service KPIs in a single dashboard
AWS: EC2	View detailed metrics from your EC2 instances. Check out new Dashboard features and framework through the link in the Overview section.
AWS: Billing	View detailed billing metrics and potential cost saving opportunities
AWS: S3	View detailed metrics for your S3 services
AWS: Pricing	View the current prices for different EC2 instance types in each AWS region
AWS: ALB	View detailed metrics for your Application Load Balancer services
AWS: API Gateway	View detailed metrics for your API Gateway services
AWS: Auto Scaling	View detailed metrics for your Auto Scaling groups
AWS: CloudFront	View detailed metrics for your CloudFront services in each AWS region
AWS: CloudSearch	View detailed metrics for your CloudSearch services
AWS: CloudTrail	View detailed metrics for your CloudTrail services
AWS: DMS	View detailed metrics for your Database Migration services
AWS: Direct Connect	View detailed metrics for your Direct Connect services
AWS: DynamoDB	View detailed metrics about your DynamoDB services
AWS: EBS	View detailed metrics for your EBS services
AWS: ECS	View detailed metrics for your ECS services
AWS: EC2 Reservations	View detailed metrics for your EC2 reservations
AWS: ECS (Fargate)	View detailed metrics for your Fargate services

Vamos a ver detalles de cada uno de ellos, con métricas de las últimas 12 horas, por poneros un ejemplo. Me gusta que Wavefront tiene todo esto ya creado y funcionando sin necesidad de crear nosotros nada adicional.

Además, me parece crítico que podamos obtener un detalle completo del Billing, y de todo lo que tenemos desplegado en AWS, son los dos Dashboards que he marcado con la flechita, más arriba.

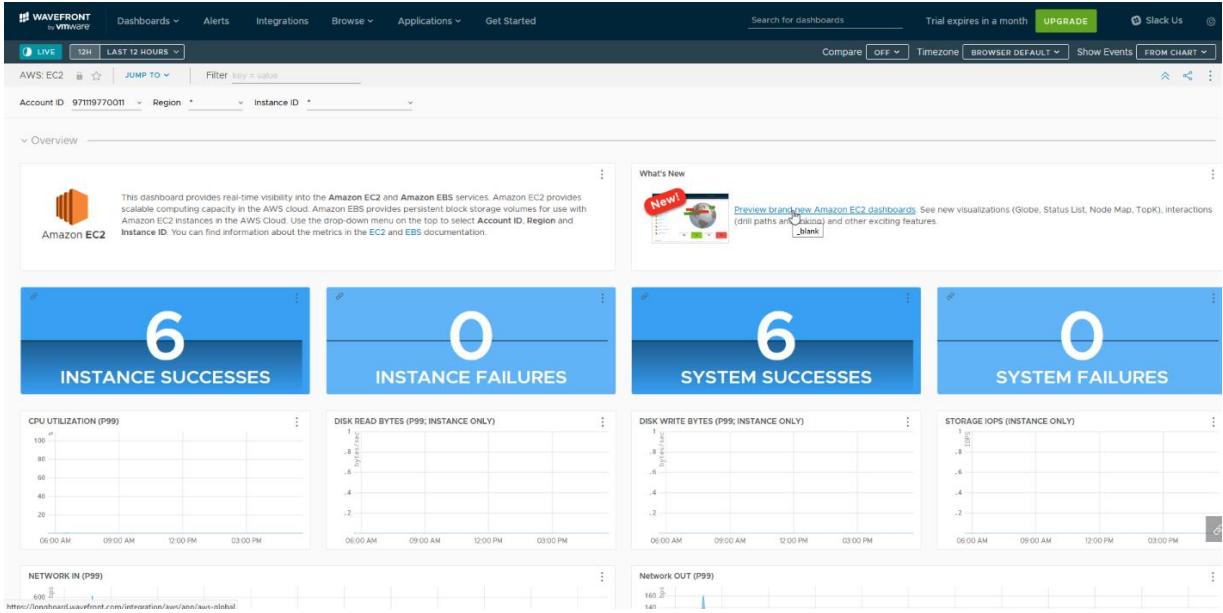
WAVEFRONT AWS: SUMMARY DASHBOARD

Este Dashboard nos muestra un vistazo rápido de todo nuestro entorno, podemos encontrar todas las instancias, CloudFront, S3, y demás recursos que estemos usando, es uno de mis Dashboards favoritos que solo por él ya merece la pena usar Wavefront:

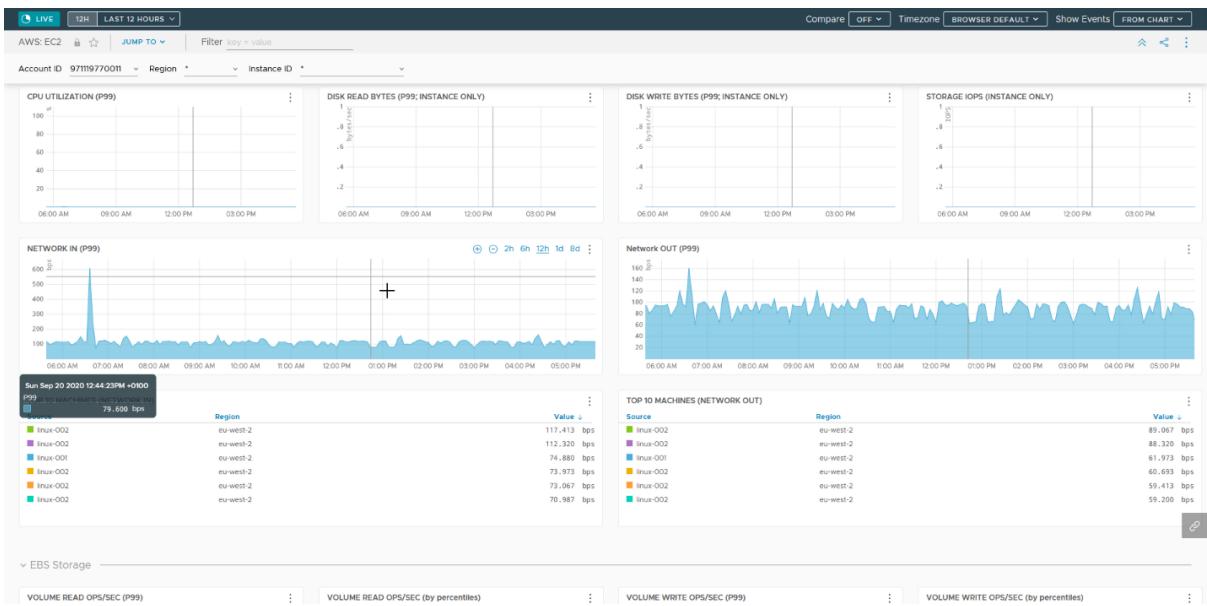


WAVEFRONT AWS: EC2 SUMMARY DASHBOARD

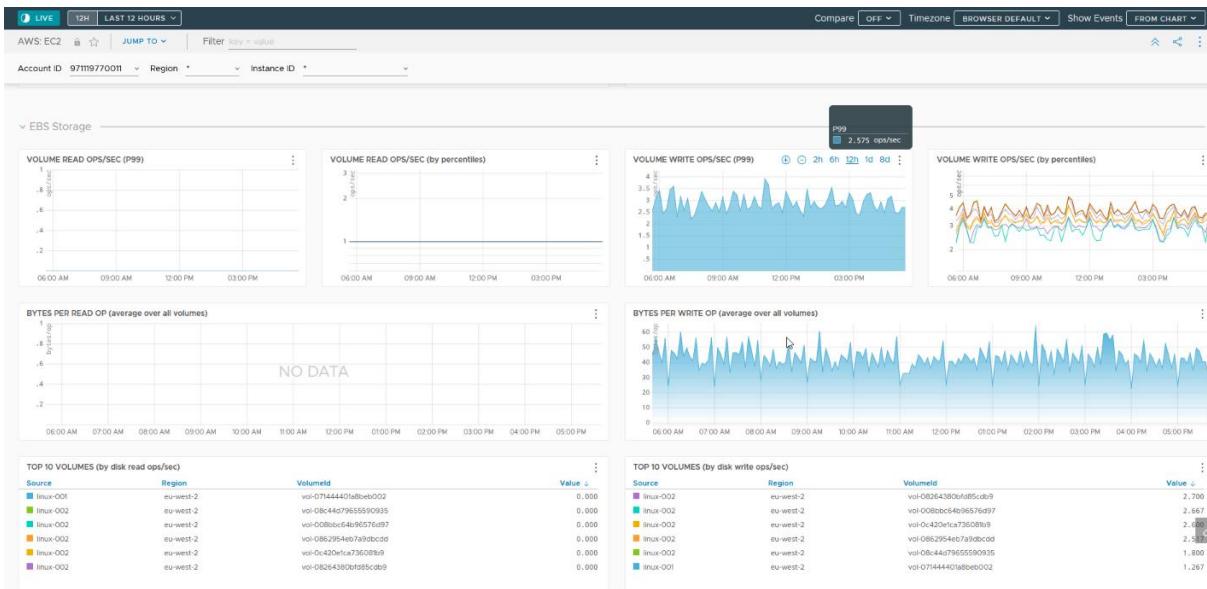
Dashboard muy completo y sencillo de comprender, podremos ver las instancias que tenemos, si están en Success, o hay system failures, etc.



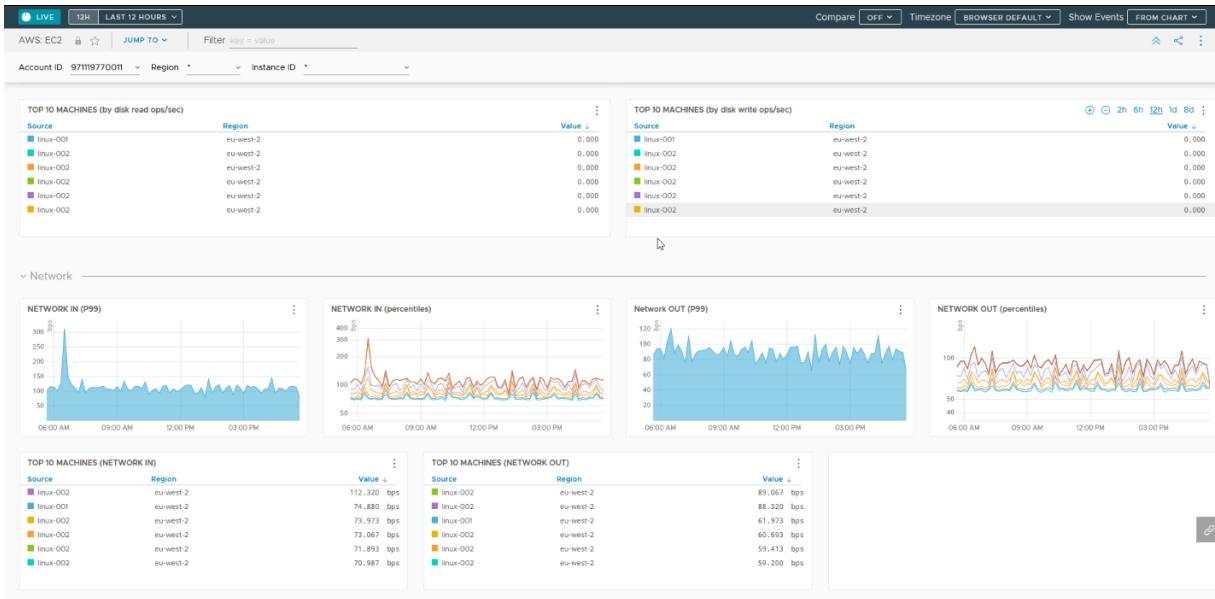
Haciendo un poquito de scroll, podremos ver el consumo de CPU por cada instancia, así como las lecturas y escrituras en disco, y los IOPS. Si seguimos en este Dashboard, podremos encontrar el tráfico de red agrupado de todas las instancias, y un poco más abajo encontramos una tabla muy cómoda con el tráfico IN y OUT de cada instancia EC2:



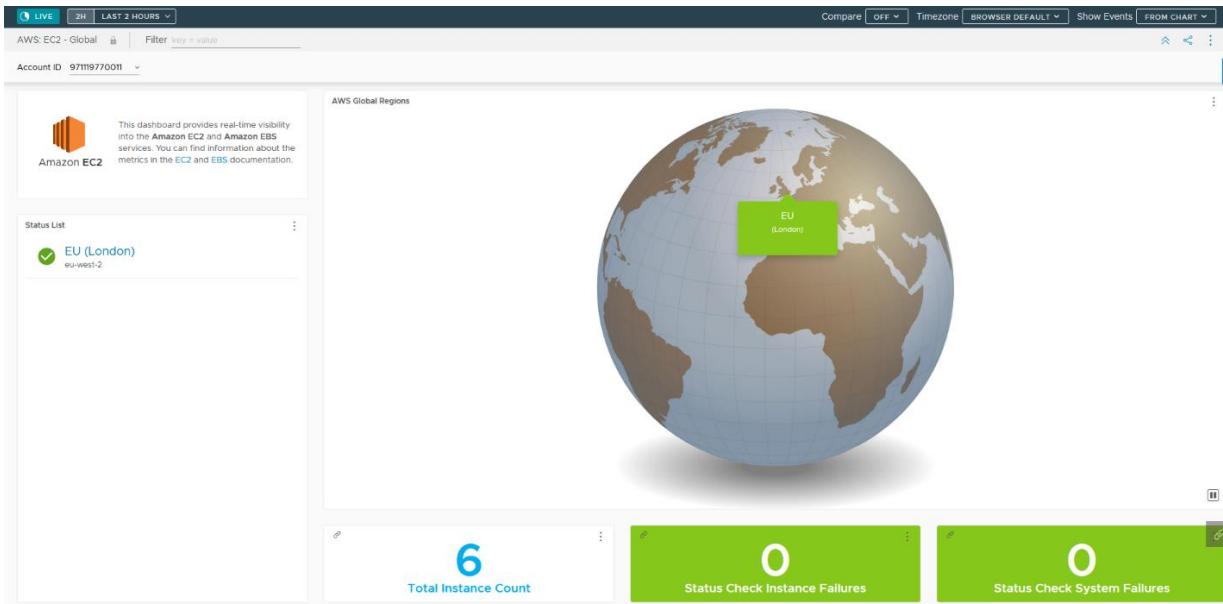
Continuamos con el almacenamiento usado, y cómo ha sido usado, en este caso hablo de los volúmenes EBS, donde encontramos detalladas gráficas de consumo y de uso de estos, esta parte del Dashboard es crítica para depurar cuellos de botella en las instancias EC2:



Finalizamos el Dashboard con la parte detallada de Networking, por cada instancia de EC2, además de resumir el Top 10, el cual es crítico cuando tenemos cientos o miles de instancias EC2, poder filtrar rápido y saber qué instancias son las más pesadas en consumo de recursos:

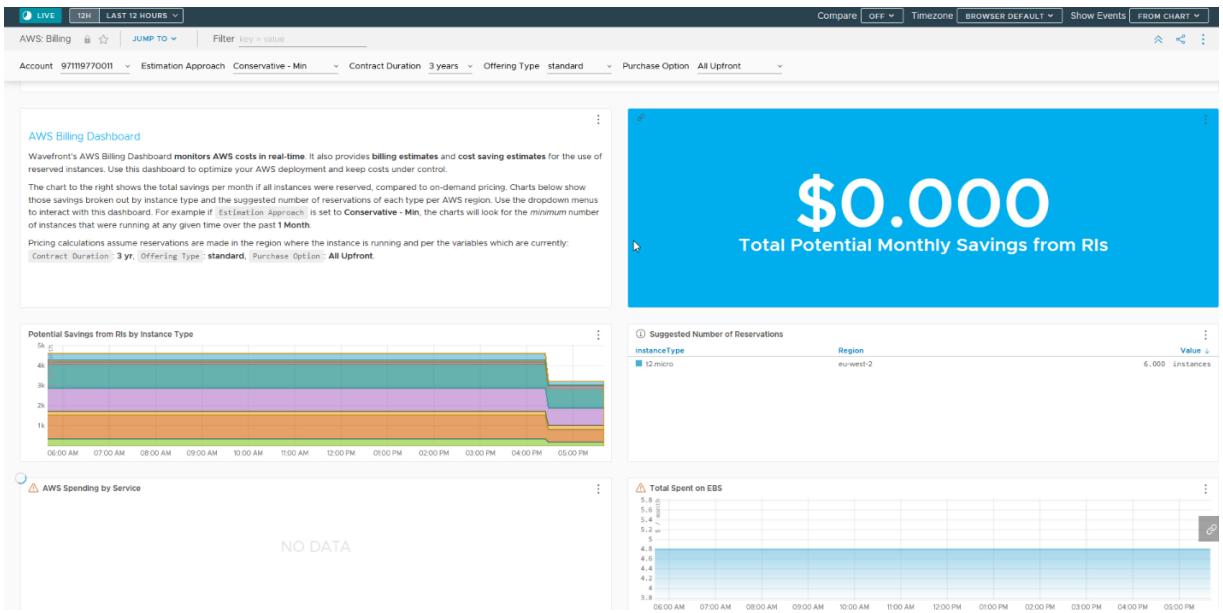


Un pequeño detalle que me ha gustado mucho es la vista de Globo Terráqueo, con los recursos que estamos consumiendo, y en qué lugar del mundo se encuentran, sencillo a la vez de práctico:

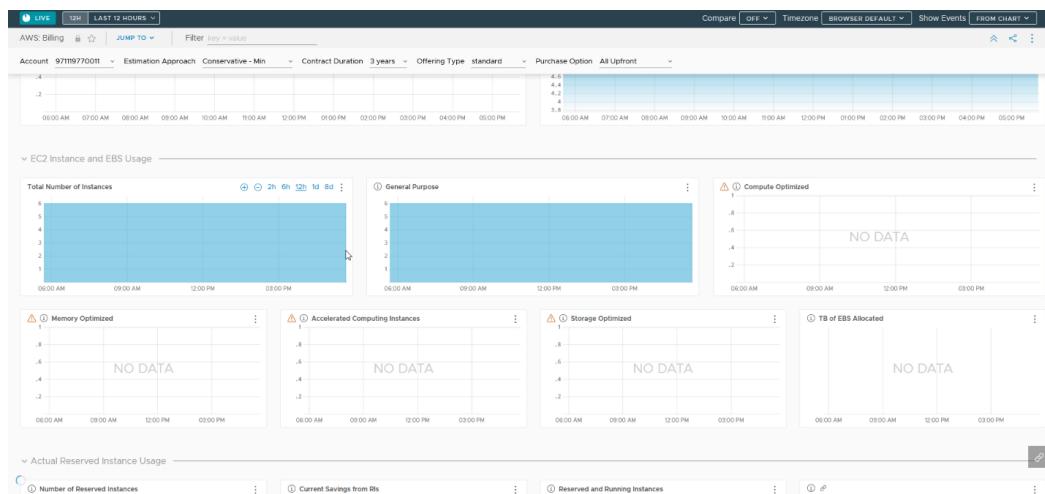


WAVEFRONT AWS: BILLING DASHBOARD

El Dashboard más importante de todo Wavefront cuando hablamos de Cloud Pública, ¡la facturación! Lamentablemente en mi caso estoy usando todo el Free-Tier que puedo, pero esto no significa que no tengamos algo de información que ojear, encontramos en azul, en grande, lo que podríamos ahorrar según Wavefront, se basa en recomendaciones, VMs que consumen poco, EC2 que podríamos ejecutar como reservadas, etc.:



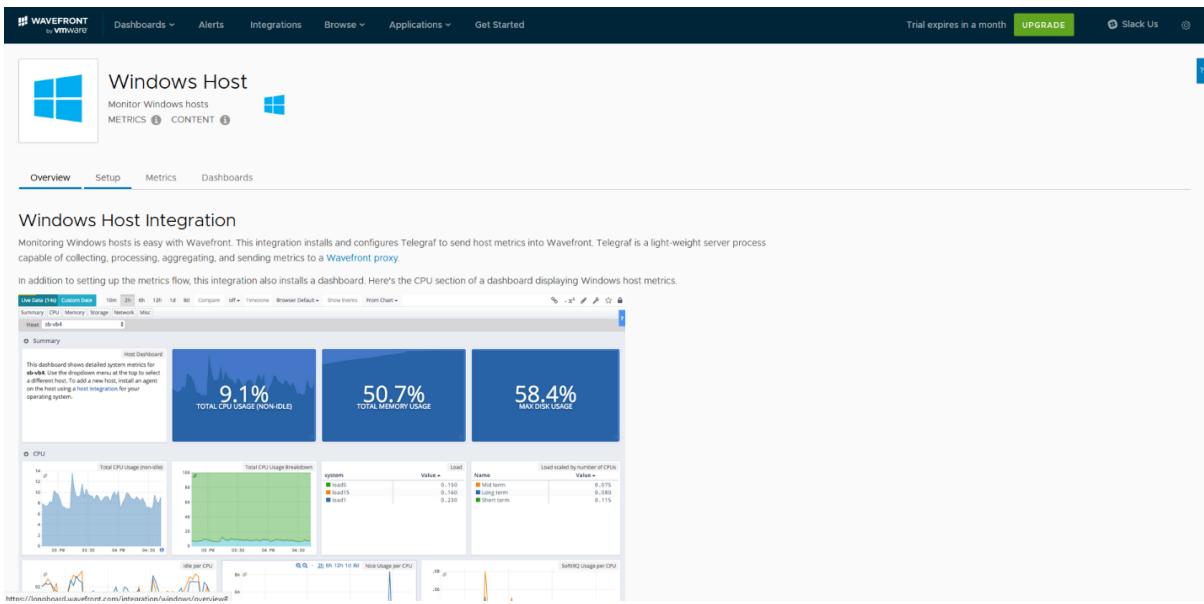
Un poquito más abajo podremos ver el consumo por recurso EC2, además de por servicio de AWS, aunque no tengo muchos, por storage, etc.:



Aunque el ejemplo no se vea en gran detalle, seguramente en cuanto añadáis vuestro entorno de AWS y le dejéis ingerir algo de información, vais a tener una visión mucho más clara sobre el consumo de AWS, también os dejo aquí un link con más información de manera detallada - <https://go.wavefront.com/aws-monitoring/>

BACK TO BASICS: MONITORIZACIÓN DE CARGAS DE TRABAJO WINDOWS

Estamos llegando a las dos últimas secciones sobre Wavefront. Hemos visto cómo monitorizar VMware vSphere, y cómo monitorizar Cloud Pública, son apenas unos minutos en ambos, pero claro, no todo es Cloud Pública y Virtualización, en muchas ocasiones querremos monitorizar en detalle el Sistema Operativo que estamos ejecutando, ya sea en Cloud, virtualizado, o físico. Bien, pues Wavefront nos da esta monitorización también, de manera muy sencilla, por ejemplo, para Windows, nos iremos a Integrations – Microsoft Windows:



Y sin más dilación haremos click en Setup para conocer los pasos que tenemos que seguir para monitorizar nuestras cargas de trabajo con Microsoft Windows.

INSTALACIÓN Y CONFIGURACIÓN DE TELEGRAF EN MICROSOFT WINDOWS

Hemos visto que Wavefront se basa en la monitorización que Telegraf nos da, con lo que no es extraño que el paso que tenemos que seguir es justamente esto, instalar el agente de Telegraf para Microsoft Windows, descargaremos Telegraf desde la web de Wavefront, normalmente lo encontraremos aquí:

<https://s3-us-west-2.amazonaws.com/wavefront-cdn/windows/wavefront-telegraf-64-setup.exe>

Ejecutaremos el asistente, que lanzará el asistente típico de una aplicación Windows, que se completará en segundos. Una vez que tenemos Telegraf instalado, tendremos que editar el fichero telegraf.conf que podemos encontrar en C:\Program Files\Telegraf y añadiremos lo siguiente:

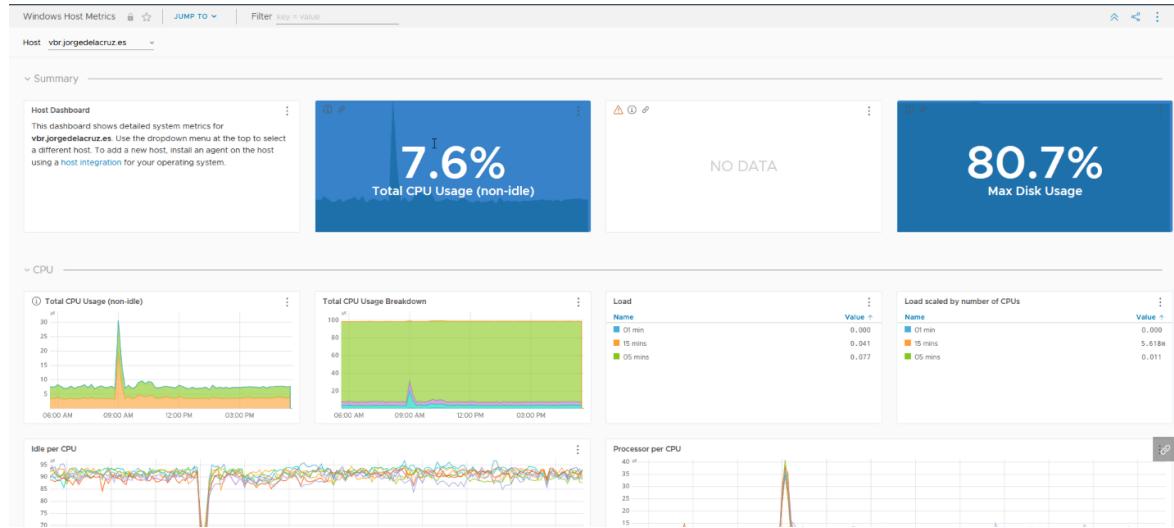
```
[ [outputs.wavefront] ]  
host = "WAVEFRONT_PROXY_HOSTNAME"  
port = 2878
```

Este paso tenemos que hacerlo si tenemos el Proxy de Wavefront en otra VM, como os comenté un poco más arriba, yo lo desplegué sobre un Ubuntu, vamos a reiniciar los servicios de Telegraf con:

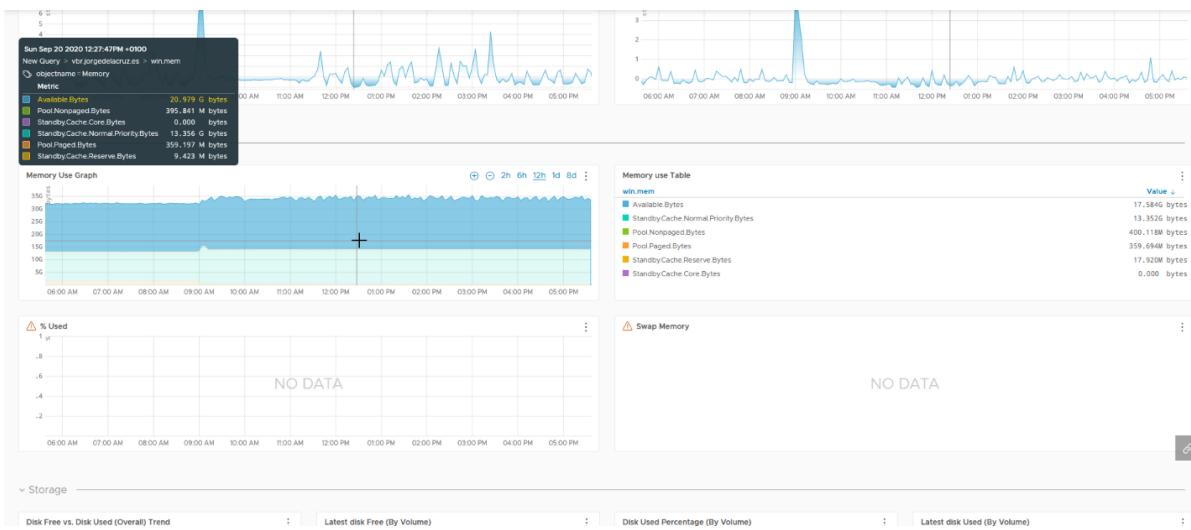
```
net stop telegraf  
net start telegraf
```

VISUALIZACIÓN DE MICROSOFT WINDOWS CON DASHBOARDS DE WAVEFRONT

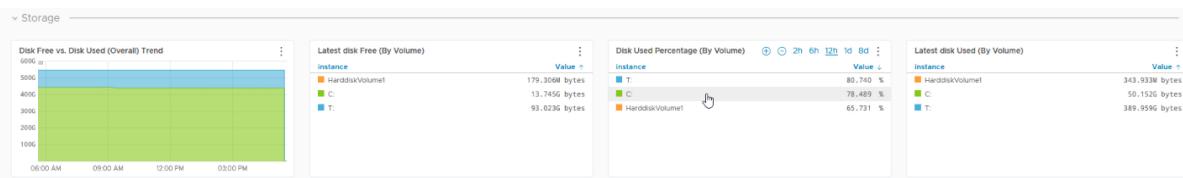
He dejado recolectando información suficiente por unas 24 horas para mostráros estos fantásticos Dashboards, listos para consumir. Podemos apreciar grande en azul el consumo de CPU, RAM y disco en paneles, así como mucho más detalle sobre el consumo de CPU en las gráficas:



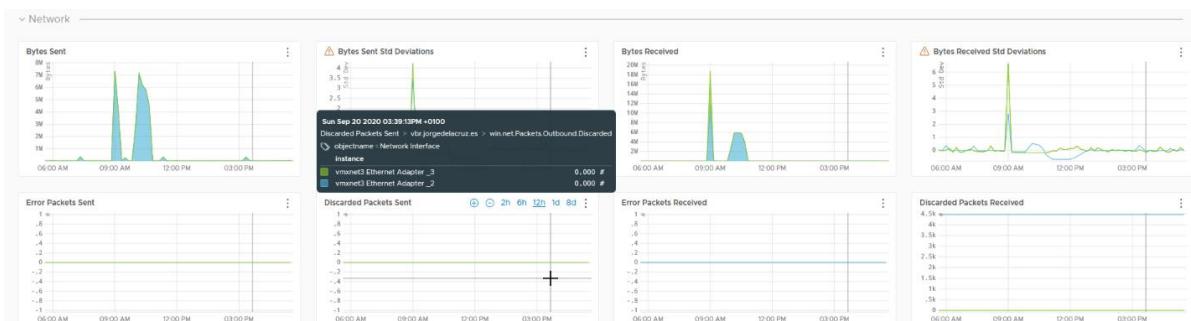
Podremos consultar también el consumo detallado de la memoria RAM de esta carga de trabajo Windows, con todo lujo de detalles:



Echar un ojo al almacenamiento, consumo de espacio en disco, que viene muy bien:

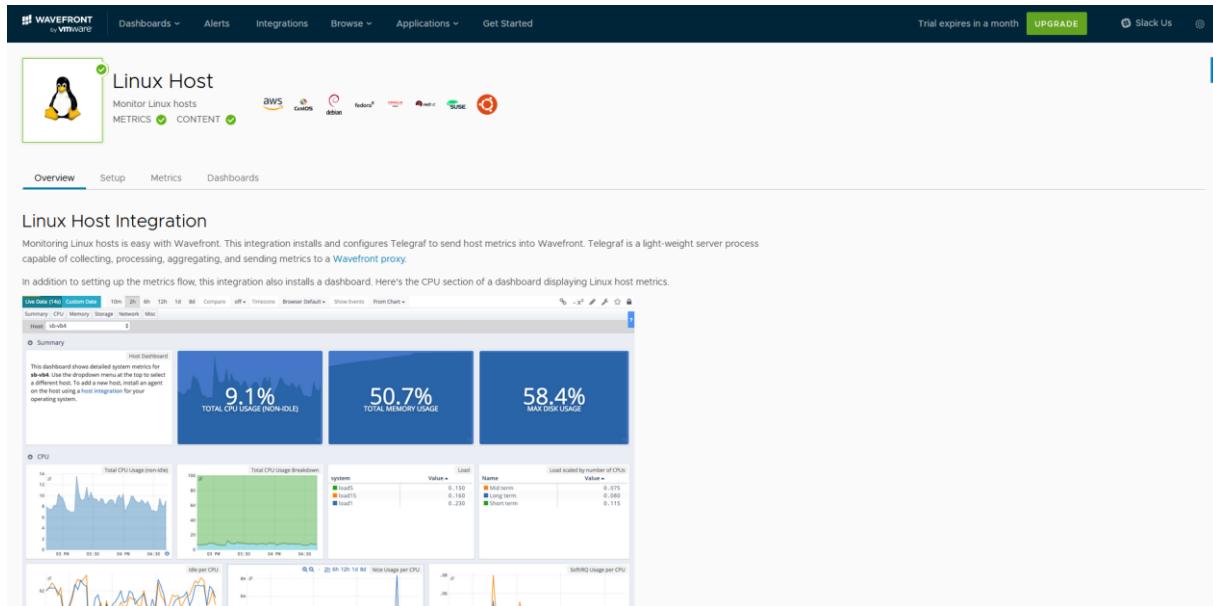


Y por supuesto podremos visualizar el consumo de Ethernet con todo lujo de detalles:



BACK TO BASICS: MONITORIZACIÓN DE CARGAS DE TRABAJO LINUX

Última sección de este capítulo del libro. Hemos visto cómo monitorizar Windows, con lo que vamos a dar el salto a Linux. En Integrations – Linux, podremos ver que tenemos todos los pasos como de costumbre:



The screenshot shows the Wavefront interface with the 'Linux Host' integration selected. The top navigation bar includes 'Dashboards', 'Alerts', 'Integrations', 'Browse', 'Applications', and 'Get Started'. A trial message 'Trial expires in a month' and an 'UPGRADE' button are visible. Below the navigation, there's a section for 'Monitor Linux hosts' with 'METRICS' and 'CONTENT' status indicators. A list of supported platforms includes AWS, CentOS, alpinelinux, fedora, openSUSE, redhat, and SUSE. The main content area is titled 'Linux Host Integration' and contains a brief description of how to set up monitoring. It also shows a preview of a dashboard with CPU usage metrics like 'TOTAL CPU USAGE (NON-IDLE)' at 9.1%, 'TOTAL MEMORY USAGE' at 50.7%, and 'MAX DISK USAGE' at 58.4%. The dashboard interface shows various charts and graphs for CPU, memory, and disk usage over time.

En la parte de Setup, podremos ver una vez más que Wavefront hace uso de Telegraf para recopilar la información, con lo que vamos a ver cómo instalarlo.

INSTALACIÓN Y CONFIGURACIÓN DE TELEGRAF EN LINUX

La instalación de Telegraf es muy sencilla, como siempre, tendremos que conocer si tenemos un Wavefront Proxy en nuestra red o no, si no lo tenemos, más arriba tenéis los pasos, si ya está instalado, en nuestro Linux instalamos Telegraf de la siguiente manera:

```
sudo bash -c "$(curl -sL https://wavefront.com/install)" --  
install \  
--agent \  
--proxy-address LAIPDETUPROXYWAVEFRONT \  
--proxy-port 2878
```

Y ya estaría, podremos comprobar que telegraf se está ejecutando con un service telegraf status, que nos mostrará algo así:

```
Redirecting to /bin/systemctl status telegraf.service
```

```
- telegraf.service - The plugin-driven server agent for reporting metrics into InfluxDB
```

```
  Loaded: loaded (/usr/lib/systemd/system/telegraf.service; enabled; vendor preset: disabled)
```

```
  Active: active (running) since Sat 2020-09-19 18:44:03 UTC; 2 days ago
```

```
    Docs: https://github.com/influxdata/telegraf
```

```
    Main PID: 25283 (telegraf)
```

```
    CGroup: /system.slice/telegraf.service
```

```
           25283           /usr/bin/telegraf           -config  
/etc/telegraf/telegraf.conf           -config-directory  
/etc/telegraf/telegraf.d
```

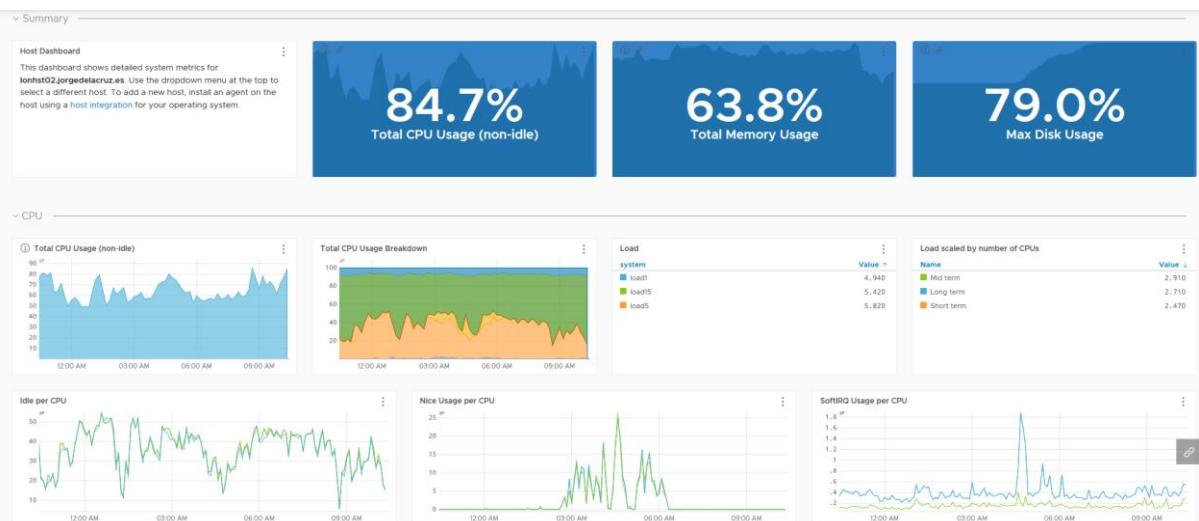
```
Sep 19 18:44:03 lonhst02.jorgedelacruz.es systemd[1]: Stopped The plugin-driven server agent for reporting metrics into InfluxDB.
```

```
Sep 19 18:44:03 lonhst02.jorgedelacruz.es systemd[1]: Started The plugin-driven server agent for reporting metrics into InfluxDB.
```

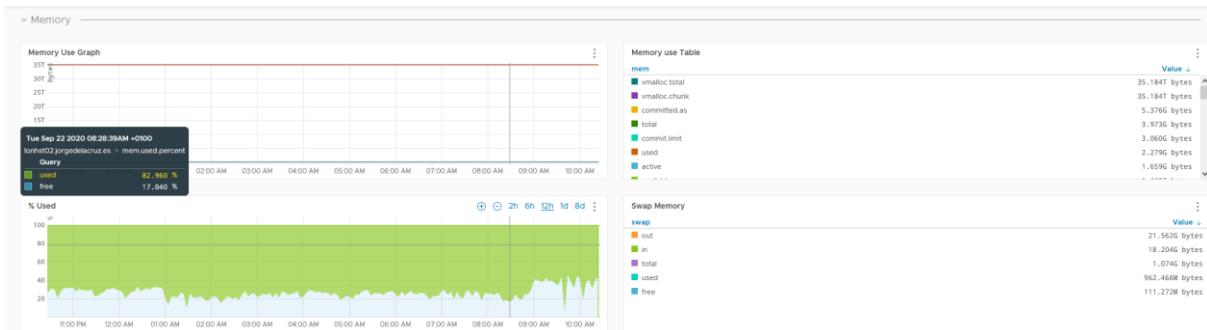
```
Sep 19 18:44:03 lonhst02.jorgedelacruz.es telegraf[25283]: 2020-09-19T18:44:03Z I! Starting Telegraf 1.15.3
```

VISUALIZACIÓN DE LINUX CON DASHBOARDS DE WAVEFRONT

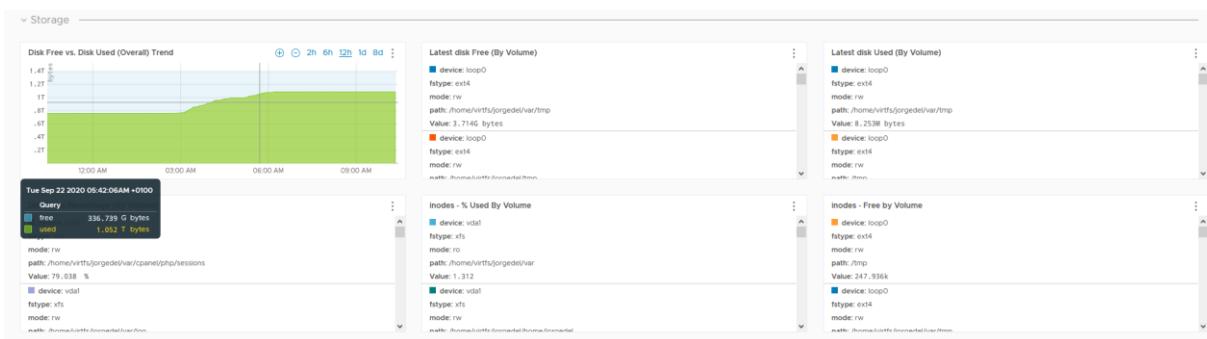
He dejado recolectando información suficiente por unas 24 horas para mostráros estos fantásticos Dashboards, listos para consumir. Podemos apreciar grande en azul el consumo de CPU, RAM y disco en paneles, así como mucho más detalle sobre el consumo de CPU en las gráficas:



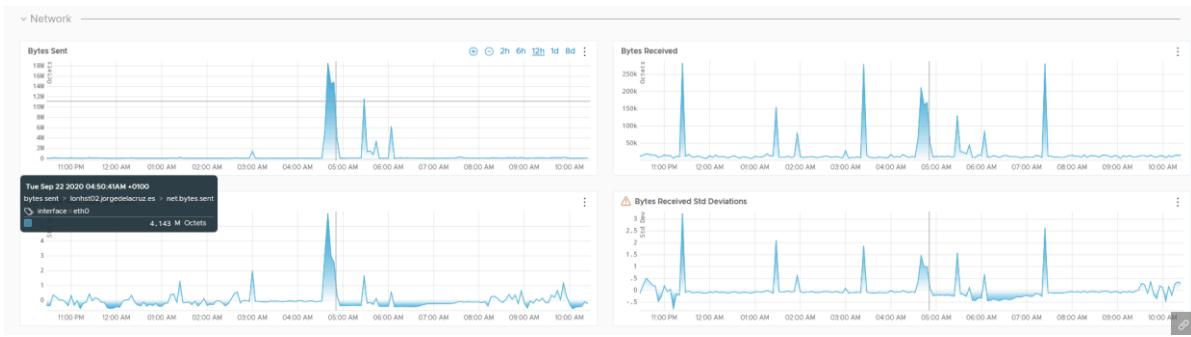
Podremos consultar también el consumo detallado de la memoria RAM de esta carga de trabajo Linux, con todo lujo de detalles:



Echar un ojo al almacenamiento, consumo de espacio en disco, que viene muy bien:



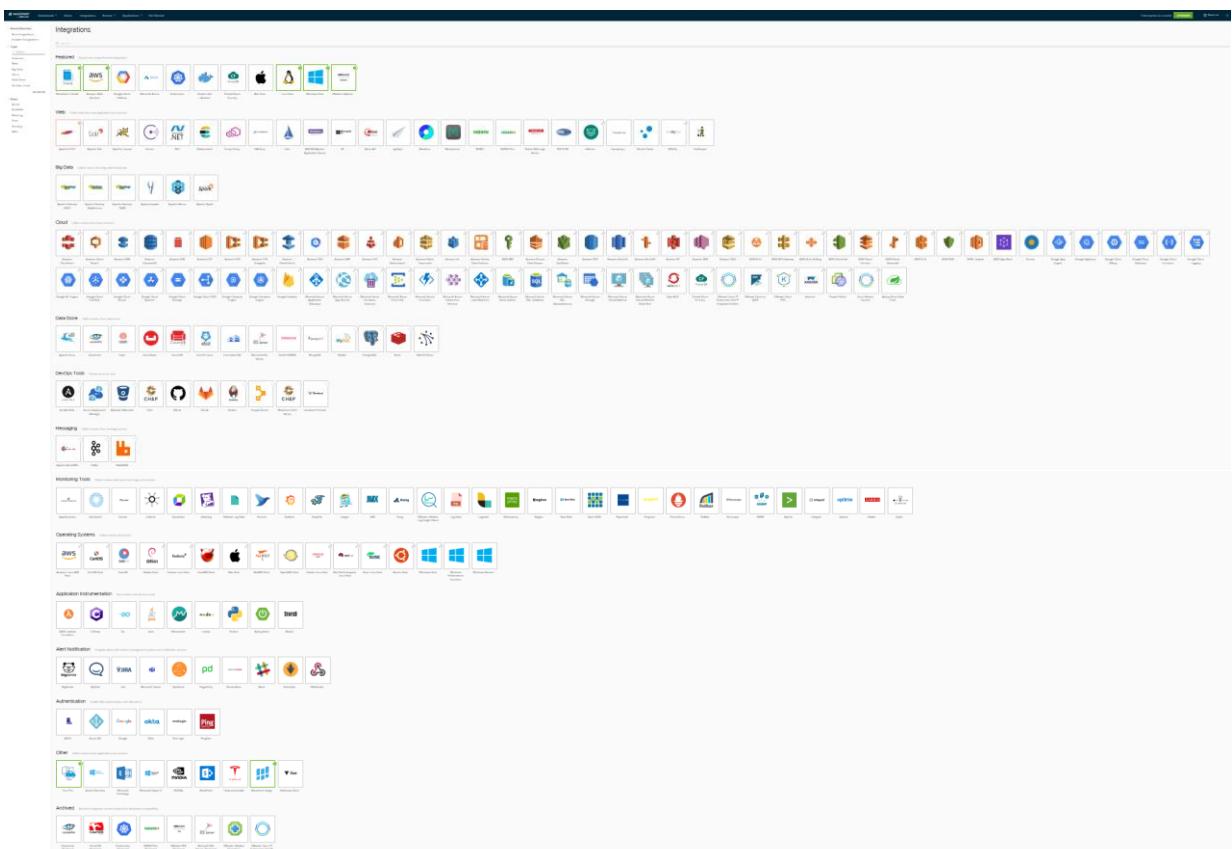
Por supuesto podremos visualizar el consumo de Ethernet con todo lujo de detalles:



Y en este caso, al ser Linux, ojear los procesos, usuarios conectados, etc. Sin duda, una información que seguro nos viene bien de vez en cuando para ojear si alguien ha entrado por SSH sin autorización, o por las noches, etc.



Eso es todo por ahora, no olvidéis que Wavefront trae decenas de servicios listos para ser monitorizados, usando Telegraf, con lo que la instalación y configuración es realmente simple, os dejo aquí todos ellos en una sola imagen ¡mirar cuántas integraciones!



Nuestro
objetivo:



Alimentar
mentes



#CAMBIA
LA
HISTORIA

La solución a
la inmigración
se encuentra
en el país de
origen



Céntrese en el desarrollo de software mientras usa una plataforma de base de datos donde se almacenan métricas de tiempo fácil de usar, escalable disponible en AWS, Azure y Google Cloud.

Fácil de empezar, fácil de escalar

Con asombrosa rapidez estarás listo y funcionando en minutos.

Constrúyelo a tu manera

Disponible para cualquier sitio, entorno o proveedor de Cloud

Construido para este propósito

Soporta millones de métricas por segundo

Acumular

- Eventos
- Métricas
- Logs
- Trazas

Analizar

InfluxDB

Actuar

- Visualización
- Alertas
- Triggers

Acumular

Obtener cualquier dato: métricas, eventos, registros, logs de cualquier lugar, sistemas, sensores, colas, bases de datos y redes, y almacenarlos en un servidor de alto rendimiento capaz de ingerir millones de puntos de datos por segundo.

Analizar

Realiza análisis en todos los conjuntos de datos con Flux - un lenguaje de consulta de cuarta generación - o InfluxQL - un lenguaje similar al SQL. Reducir la muestra de datos para un rendimiento óptimo de la consulta y proporcionar análisis en tiempo real para una mejor comprensión.

Actuar

Comienza tu viaje hacia la automatización: configura alertas con un simple clic o realiza una compleja detección de anomalías basada en algoritmos de aprendizaje de máquinas. Envía alertas a servicios populares como Slack, SMS y PagerDuty. Crea tareas personalizadas para realizar cualquier acción.

Capítulo 7

SITE RECOVERY MANAGER



Miquel Mariano
@miquelMariano

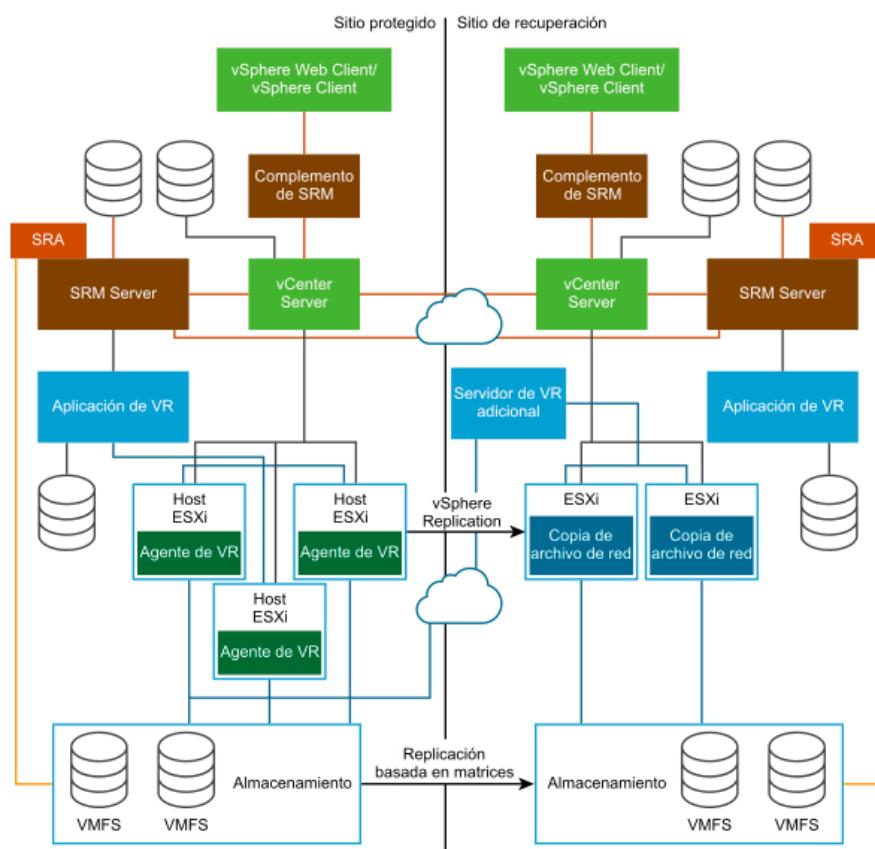
¿QUÉ ES SRM?

Site Recovery Manager, SRM para los amigos, es un producto incluido en el portfolio de VMware que nos sirve como solución de continuidad empresarial y recuperación ante desastres.

SRM está basado en replicación de VMs entre dos sites (CPDs) y nos permite planificar, probar y ejecutar VMs entre un sitio protegido de vCenter y un sitio de recuperación

Mediante “**Grupos de protección**” y “**Planes de recuperación**”, Site Recovery Manager se encarga de organizar y orquestar todas las acciones necesarias para garantizar la continuidad de nuestro negocio.

A continuación, podéis ver un esquema genérico de una arquitectura SRM en 2 sites:



- **vCenter Server:** Como sabéis es el core de cualquier arquitectura SDDC. Necesitaremos dos, uno por cada sitio a proteger. Pueden estar configurados en ELM (Enhanced Linked Mode), no hay problema.
- **SRM Server:** Es el propio servidor de Site Recovery Manager. También necesitaremos uno por sitio, y desde la versión 8.2 se proporciona en versión appliance, por lo que no es necesario ningún servidor Windows como antaño, esto facilita mucho el despliegue.
- **SRA:** Storage Replication Adapter. Es un pequeño software que se encarga de la comunicación con nuestra cabina de almacenamiento, en caso de utilizar esta

tecnología para replicar. También necesitaremos dos, y estará instalado en el propio SRM Server. Más adelante hablaremos de los tipos de replicación.

- **VR:** vSphere Replication. Appliance que nos permitirá la replicación por software de VMs entre el sitio protegido y el sitio de respaldo. Cada vCenter necesitará de su VR Appliance, pero después podremos añadir hasta 9 VR Servers para poder procesar más VMs.

REPLICACIÓN BASADA EN ARRAY VS VSOPHERE REPLICATION

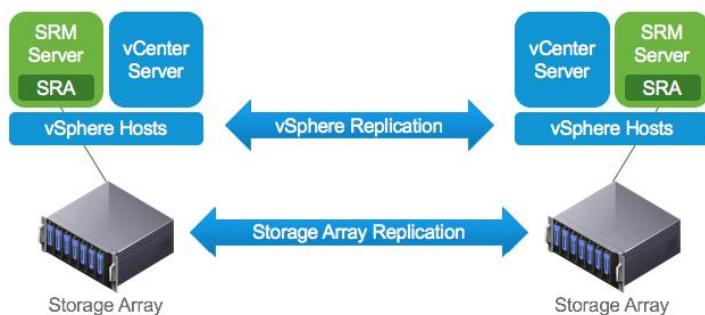
En la actualidad, SRM soporta dos tipos de replicación.

- Replicación hardware basada en cabina de almacenamiento.
- Replicación software basada en vSphere Replication

Por suerte, la mayor parte de fabricantes de storage tienen su propio SRA (Storage Replication Adapter).

El componente SRA no es más que el software que le sirve a SRM para comunicarse con nuestra cabina de almacenamiento. En la propia guía de compatibilidad de VMware podremos encontrar todos los SRA de todos los fabricantes disponibles.

<https://www.VMware.com/resources/compatibility/search.php?deviceCategory=sra>



A la hora de implementar SRM en una organización, es una decisión importante a tener en cuenta. Ambas tecnologías pueden convivir en el mismo entorno SRM, pero no todas las VMs se podrán proteger de la misma manera.

Para tomar esa decisión, os adjunto una pequeña tabla que puede ayudar a resolver muchas preguntas:

Array based

vSphere Replication

Type	Replication using the storage layer	Replication using the host/vSphere layer
RPO min/max	0 up to max supported by vendor	5 minutes to 24 hours (5 min RPO was introduced in version 6.0 for vSAN-to-vSAN and all datastores in 6.5)
Scale	Scales up to 5,000 VMs protected/2,000 simultaneously recoverable per vCenter/SRM pair	Scales up to 2,000 VMs (protected & recoverable) per vCenter/SRM pair
Write order fidelity	Supports write order fidelity within and across multiple VMs in the same consistency group	Supports write order fidelity on the disks/VMDKs that make up a VM, consistency cannot be guaranteed across multiple VMs
Replication level	Replicates at the LUN/VMFS or NFS volume level	Replicates at the VM level
Replication configuration	Replication is configured and managed on the storage array	Replication is configured and managed in the vSphere Web Client
Array/ vendor types	Requires same storage replication solution at both sites (eg. EMC RecoverPoint, NetApp vFiler, IBM SVC, etc)	Can support any storage solution at either end including local storage as long as it is covered by the vSphere HCL
Storage supported	Replication supported on FC, iSCSI or NFS storage only	Supports replicating VMs on local, attached, vSAN, FC, iSCSI or NFS storage
Cost	Replication and snapshot licensing is required	vSphere Replication is included in vSphere Essentials Plus license levels and higher of vCenter version 5.1 and higher
Deployment	Deployment is fairly involved and must include storage administration and possibly networking	Deployment requirements are minimal. Deploy an OVF at each site and start configuring replications
Application consistency	Depending on the array, application consistency may be supported with the addition of agents to the VM	Supports VSS & Linux file system application consistency
FT VMs	Can replicate UP FT protected VMs (once recovered VM is no longer FT enabled). Does not support SMP FT VMs.	Cannot replicate FT protected VMs
Powered off VMs/Templates/Linked clones/ISO's	Able to replicate powered off VMs, Templates, Linked Clones (as long as all nodes in the snapshot tree are replicated as well) and ISOs	Can only replicate powered on VMs. Cannot replicate powered off VMs, Templates, Linked Clones, ISOs or any non-VM files
RDM support	Physical and Virtual mode RDMs can be replicated	Only Virtual mode RDMs can be replicated
MSCS support	VMs that are part of a MSCS cluster can be replicated	Cannot replicate VMs that are part of a MSCS cluster. VR cannot replicate disks in multi-writer mode.
vApp support	Replicating vApps is supported	Replicating vApps is not possible. However, it is possible to replicate VMs that are part of a vApp and to create a vApp at the recovery site that they are recovered into
vSphere versions supported	Hosts running vSphere 3.5-6.5 are supported	Hosts must be running vSphere 5.0 or higher
MPIT	Multiple point in time snapshots or rollback is supported by some supported array vendors (eg. EMC RecoverPoint)	Supports up to 24 recovery points
Snapshots	Supports replicating VMs with snapshots and maintaining the snapshot tree	Supports replicating VMs with snapshots however the tree is collapsed at the target site

Response to Host failure	Replication is not impacted	Host Failure, and the VM restarting on another host triggers a full sync. For details about what a full sync involves see the vSphere Replication FAQ
vVols integration	SRM does not currently support vVols with array-based replication	vVols are supported by vSphere Replication with SRM
Interop with vRA	VMs that are managed/deployed by vRA and are using array-based replication can be easily protected either with Storage Policy-Based Protection Groups or using the vRO SRM plug-in	vRA managed VMs that need to use vSphere Replication can be protected using the vRO plug-ins for SRM and VR
Policy-based protection	Policy based protection is possible through use of SPPGs (Storage Policy Based Protection Groups)	vSphere Replication doesn't support policy-based protection

LICENCIAMIENTO

Cómo la mayoría de los productos de VMware, al instalar una o varias instancias de Site Recovery Manager se nos asignará un período de evaluación de 60 días.

Una vez vencido este período los grupos de protección creados seguirán funcionando, pero ya no se podrán agregar nuevas VMs a estos grupos o crear de nuevos.

Actualmente SRM está disponible en 2 ediciones. Standard y Enterprise.

- **Standard** > Pensada para entornos pequeños. Nos permite proteger máximo 75 VMs por sitio y por instancia SRM
- **Enterprise** > Pensada para entornos grandes y si necesitamos protección cruzada entre sites. Este plan no tiene limitación en cuanto al número de VMs a proteger

En la siguiente tabla podréis ver, a parte del nº de VMs que se pueden proteger, las funcionalidades que tiene cada versión:

Features / Particulars	SRM Standard Licensing	SRM Enterprise Licensing
Centralised Recovery Plans:	Yes	Yes
Stretched Storage Support Options	No	Yes
Nondisruptive testing	Yes	Yes
Automated Orchestration Workflow	Yes	Yes
The vSphere Replication Support	Yes	Yes
Support for array-based replication	Yes	Yes
Orchestrated cross-vCenter vMotion	No	Yes
Storage-profile protection groups	No	Yes
		Yes
VMware NSX integration	No	

También es importante comentar que si optamos por el método de replicación con vSphere Replication no tendremos que comprar licencias adicionales, ya que VR es una característica incluida en todas las versiones vSphere, **menos en vSphere Essentials**.

En caso de utilizar replicación basada en cabina de almacenamiento, tendremos que consultar con nuestro fabricante de la necesidad o no de licencias adicionales para esta característica.

BUSINESS CONTINUITY & DISASTER RECOVERY

Una vez hecha la introducción a SRM es importante profundizar en los conceptos básicos de **Business Continuity & Disaster Recovery (BCDR)**

Un **plan de continuidad del negocio** (o **BCP, Business Continuity Plan**) es un plan logístico para la práctica de cómo una organización debe recuperar y restaurar sus funciones críticas, parcial o totalmente interrumpidas dentro de un tiempo predeterminado, después de una interrupción no deseada o desastre.

SRM nos ayudará en el proceso de devolver la actividad a nuestros sistemas y servicios, pero hay otros muchos aspectos que tendremos que debatir y documentar.

El siguiente gráfico ilustra a la perfección todos los puntos para tener en cuenta ante un desastre.

Desde la detección y análisis del impacto al negocio, hasta la recuperación completa, pasando por el manejo de la crisis, respuesta de emergencia o comunicación de esta, esas son fases que tienen que estar muy bien organizadas y documentadas para poder resolver con éxito un desastre real.



Hay 3 conceptos que bajo mi punto de vista son claves a la hora de diseñar una buena solución de BCDR.

RPO (RECOVERY POINT OBJECTIVE)

- RPO se refiere al volumen de datos en riesgo de pérdida, que la organización considera tolerable. Las transacciones de cuánto tiempo estamos dispuestos a perder, o a tener que reintroducir en el sistema
- La respuesta va a depender del volumen de transacciones por unidad de tiempo, y de los mecanismos de backup, pero siempre aumenta el volumen de datos 'huérfanos' a medida que pasa el tiempo desde la última copia de seguridad.

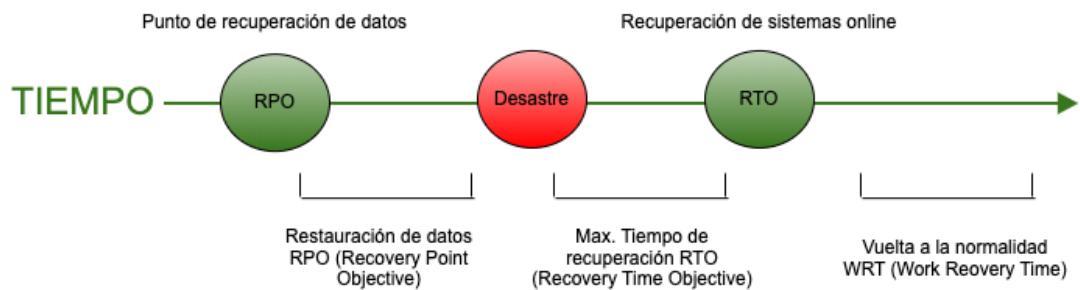
- El RPO determina el objetivo de posible pérdida máxima de datos introducidos desde el último backup, hasta la caída del sistema, y no depende del tiempo de recuperación.

RTO (RECOVERY TIME OBJECTIVE)

- Expresa el tiempo durante el cual una organización puede tolerar la falta de funcionamiento de sus aplicaciones, y la caída de nivel de servicio asociada.
- La respuesta dependerá de la criticidad de cada aplicación. No será lo mismo la aplicación que da servicio a las cajas en una gran superficie, que la aplicación para el cálculo de la nómina, que se ejecuta una vez al mes.

WRT (WORK RECOVERY TIME)

- Es el tiempo que se necesitará para, una vez que se hayan recuperado los sistemas, volver a la normalidad.
- Verificar la integridad de los sistemas y los datos
- Asegurarse de que las aplicaciones y servicios están disponibles
- Validaciones concretas de los procesos de negocio



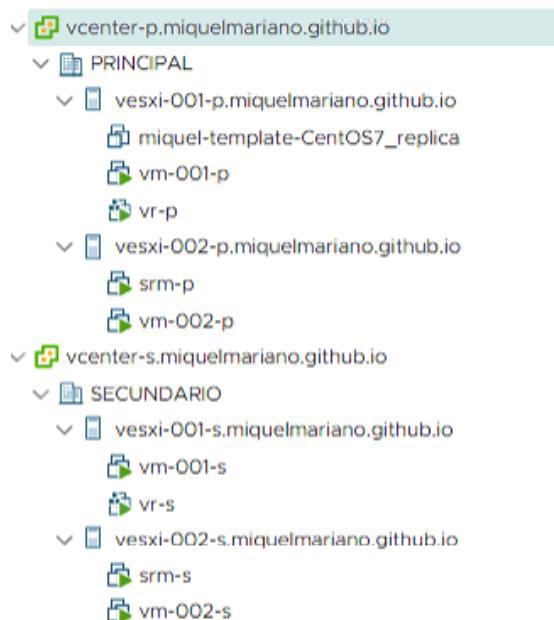
LABORATORIO

Para la redacción de este capítulo, he creado un pequeño laboratorio para tratar de explicar desde la instalación y configuración inicial, a la operativa básica de la solución.

Al ser un laboratorio completamente virtual, y no disponer de cabina de discos para la replicación por hardware, utilizaremos vSphere Replication como método de replicación.

Es obvio que, en un entorno real, cuando hablamos de sitio principal, o de protección y sitio secundario, o de recuperación, estamos hablando de ubicaciones geográficas separadas e independientes entre sí.

- 2 vCenter en modo ELM (Enhanced Linked Mode), uno por sitio
- 4 ESXi, 2 por sitio
- 2 SRM Servers, 1 por sitio
- 2 VR Appliance, 1 por sitio
- 4 VMs de prueba, 2 por sitio



Hasta la fecha de edición de este capítulo, las últimas versiones de cada producto son las siguientes:

- vSphere 7.0
- Site Recovery Manager 8.3
- vSphere Replication 8.3

No hace falta decir que es de vital importancia para cualquier proyecto que combine diferentes productos, revisar previamente la guía de compatibilidad y asegurar la completa interoperabilidad entre todos ellos.

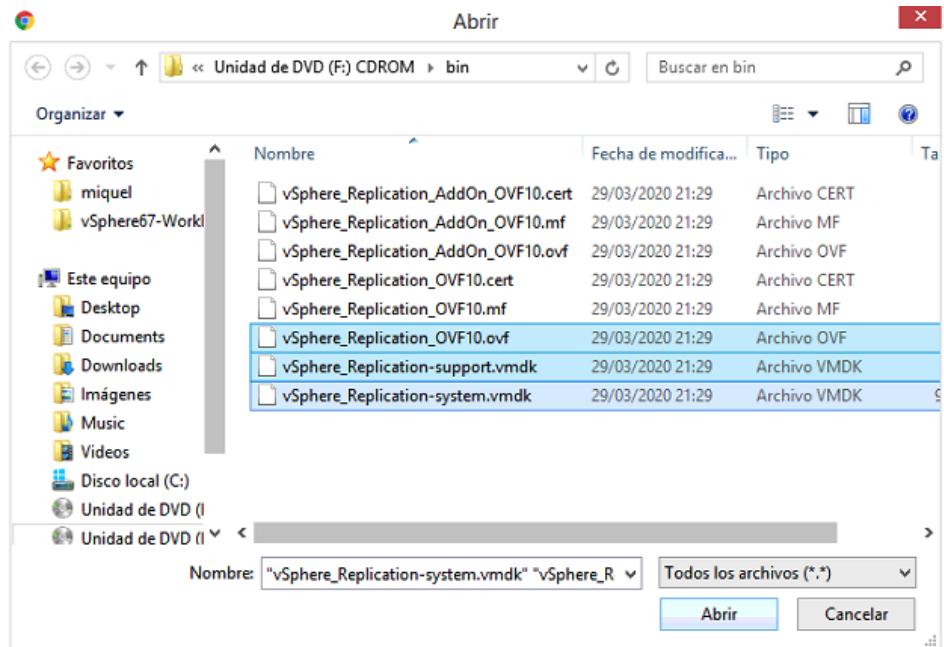
https://www.VMware.com/resources/compatibility/sim/interop_matrix.php

INSTALACIÓN vSPHERE REPLICATION 8.3

vSphere Replication se proporciona en versión appliance y lo podremos descargar desde el mismo portal my.VMware.com

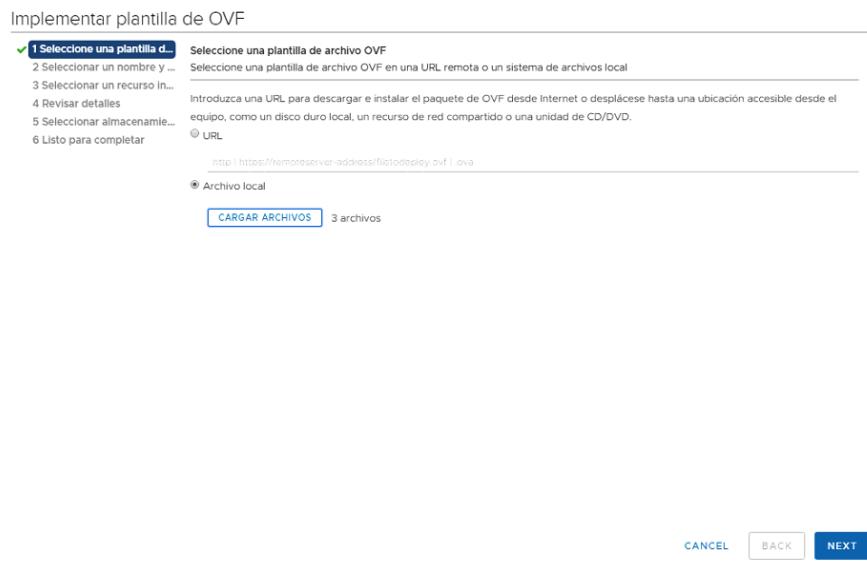
The screenshot shows the product details page for VMware vSphere Replication 8.3.0. At the top, there's a navigation bar with 'Home / VMware vSphere Replication 8.3.0'. Below it, the title 'Download VMware vSphere Replication 8.3.0' is prominently displayed. To the right, a sidebar titled 'Product Resources' lists links for 'View My Download History', 'Product Information', 'Documentation', 'vSphere Community', 'Support Resources', and a 'Get Free Trial' button. The main content area displays product details: Version 8.3.0, Description VMware vSphere Replication 8.3.0, Documentation Release Notes, Release Date 2020-04-02, and Type Product Binaries. Below this, a navigation bar includes 'Product Downloads' (which is selected), 'Drivers & Tools', 'Open Source', and 'Custom ISOs & Addons'. A large blue button labeled 'Download Now' is located on the right side of the product details section. At the bottom, there's a note about checksums: 'Information about MD5 checksums and SHA1 checksums and SHA256 checksums.'

La .iso descargada contiene los 3 ficheros necesarios para implementar la plantilla OVF.

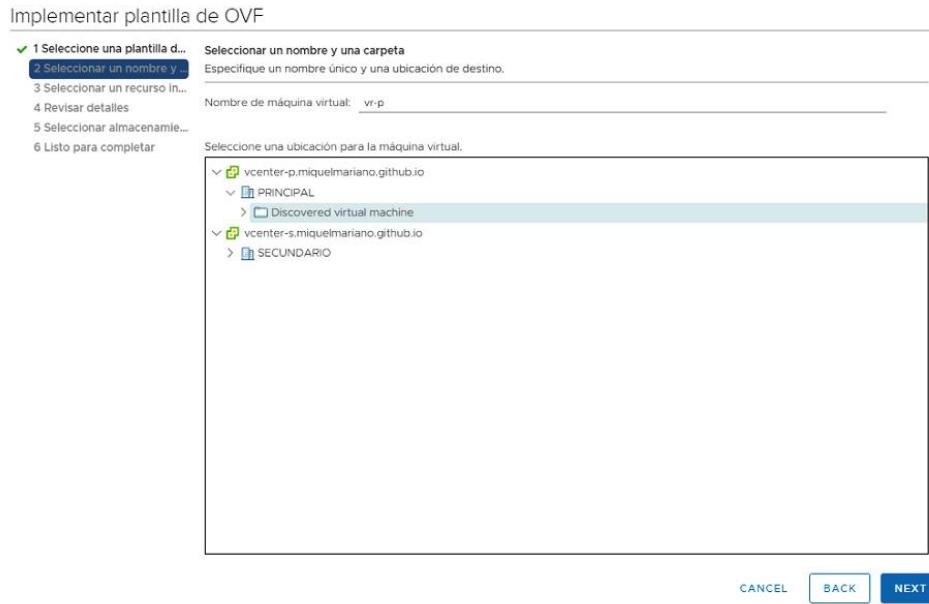


Igual que con el despliegue de SRM Server, necesitaremos 2 VR Appliances, por lo que este mismo proceso, lo haremos 2 veces.

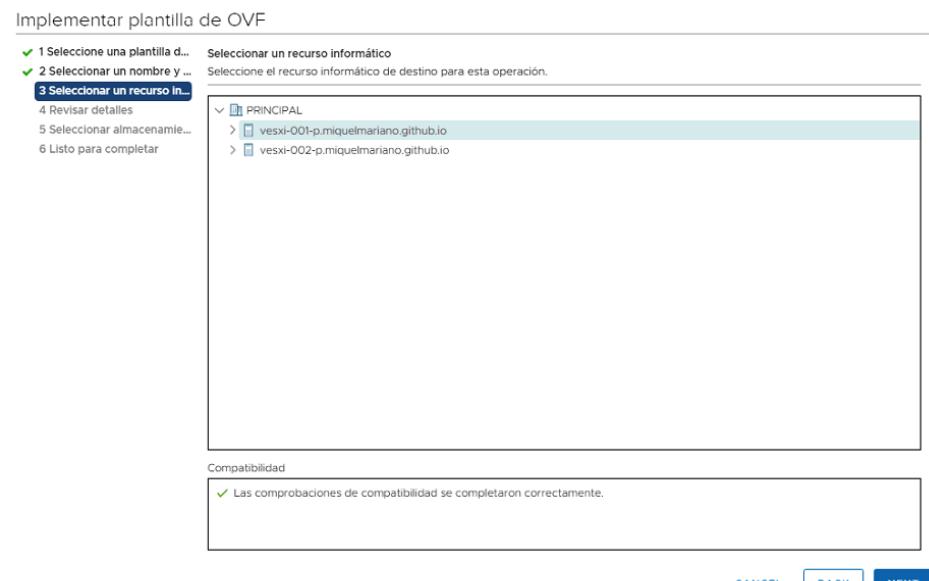
Accedemos a nuestro vCenter, y nos dirigimos a la opción de “Implementar plantilla de OVF”, seleccionaremos los ficheros que anteriormente hemos comentado.



En este caso, los nombres elegidos son **VR-P** (PRINCIPAL) y **VR-S** (SECUNDARIO)



Seleccionamos el correspondiente ESXi para desplegar la VM.



Detalles del producto y versión.

Implementar plantilla de OVF

- ✓ 1 Seleccione una plantilla d...
- ✓ 2 Seleccionar un nombre y ...
- ✓ 3 Seleccionar un recurso in...

4 Revisar detalles

- 5 Contratos de licencia
- 6 Configuración
- 7 Seleccionar almacenamiento...
- 8 Seleccionar redes
- 9 Personalizar plantilla
- 10 Enlaces de vService
- 11 Listo para completar

Revisar detalles

Compruebe los detalles de la plantilla.

Editor	No hay certificados presentes.
Producto	vSphere Replication Appliance
Versión	8.3.0.9921
Proveedor	VMware, Inc.
Descripción	vSphere Replication Appliance
Tamaño de descarga	552.9 MB
Tamaño en disco	1.1 GB (aprovisionamiento fino) 26.0 GB (aprovisionamiento grueso)

CANCEL

BACK

NEXT

Acuerdo de licencia.

Implementar plantilla de OVF

- ✓ 1 Seleccione una plantilla d...
- ✓ 2 Seleccionar un nombre y ...
- ✓ 3 Seleccionar un recurso in...
- ✓ 4 Revisar detalles

5 Contratos de licencia

- 6 Configuración
- 7 Seleccionar almacenamiento...
- 8 Seleccionar redes
- 9 Personalizar plantilla
- 10 Enlaces de vService
- 11 Listo para completar

Contratos de licencia

Se debe aceptar el contrato de licencia de usuario final.

Lea y acepte los términos del contrato de licencia.

vSphere Replication

ACUERDO DE LICENCIA DE USUARIO FINAL DE
VMWARE

TENGA EN CUENTA QUE LAS CONDICIONES DE ESTE ACUERDO DE
LICENCIA DE USUARIO FINAL REGIRÁN EL USO QUE HAGA DEL
SOFTWARE, CON INDEPENDENCIA DE LOS TÉRMINOS QUE
PUEDAN APARECER DURANTE LA INSTALACIÓN DEL SOFTWARE.

INFORMACIÓN IMPORTANTE: AL DESCARGAR, INSTALAR O
UTILIZAR EL SOFTWARE, ACEPTA LAS CONDICIONES DE ESTE
ACUERDO DE LICENCIA DE USUARIO FINAL (END USER LICENSE
ACUERDO DE LICENCIA, TANTO CUALQUIERA DE TÍTULO,

Acepto todos los contratos de licencia.

CANCEL

BACK

NEXT

Tamaño del appliance.

Implementar plantilla de OVF

- ✓ 1 Seleccione una plantilla d...
- ✓ 2 Seleccionar un nombre y ...
- ✓ 3 Seleccionar un recurso In...
- ✓ 4 Revisar detalles
- ✓ 5 Contratos de licencia
- 6 Configuración**
- 7 Seleccionar almacenamiento...
- 8 Seleccionar redes
- 9 Personalizar plantilla
- 10 Enlaces de vService
- 11 Listo para completar

Configuración		Descripción
Seleccione una configuración de implementación		
<input checked="" type="radio"/> 2 vCPU	<input type="radio"/> 4 vCPU	
2 Elementos		

CANCEL BACK NEXT

Seleccionamos el datastore dónde almacenar la VM.

Implementar plantilla de OVF

- ✓ 1 Seleccione una plantilla d...
- ✓ 2 Seleccionar un nombre y ...
- ✓ 3 Seleccionar un recurso In...
- ✓ 4 Revisar detalles
- ✓ 5 Contratos de licencia
- 6 Configuración**
- 7 Seleccionar almacenamiento...**
- 8 Seleccionar redes
- 9 Personalizar plantilla
- 10 Enlaces de vService
- 11 Listo para completar

Seleccionar almacenamiento																	
Seleccione el almacenamiento para los archivos de configuración y de disco																	
<input type="checkbox"/> Cifrar esta máquina virtual (Requiere un servidor de administración de claves)																	
Seleccione el formato de disco virtual: Puesta a cero lenta con aprovisionamiento grueso ▾																	
Directiva de almacenamiento de máquina virtual: Valor predeterminado de almacenamiento de datos ▾																	
<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Nombre</th> <th>Capacidad</th> <th>Aprovisionado</th> <th>Libre</th> <th>Tipo</th> <th>Clúster</th> </tr> </thead> <tbody> <tr> <td>Datastore-vesxi-001-p</td> <td>199.75 GB</td> <td>21.75 GB</td> <td>196.2 GB</td> <td>VMFS 6</td> <td></td> </tr> </tbody> </table>						Nombre	Capacidad	Aprovisionado	Libre	Tipo	Clúster	Datastore-vesxi-001-p	199.75 GB	21.75 GB	196.2 GB	VMFS 6	
Nombre	Capacidad	Aprovisionado	Libre	Tipo	Clúster												
Datastore-vesxi-001-p	199.75 GB	21.75 GB	196.2 GB	VMFS 6													
Compatibilidad																	
✓ Las comprobaciones de compatibilidad se completaron correctamente.																	

CANCEL BACK NEXT

Seleccionamos el VM Portgroup correspondiente.



Configuramos los parámetros propios del appliance:

- Credenciales
- Servidor de tiempo
- Hostname
- Configuración IP

Implementar plantilla de OVF

- ✓ 1 Seleccione una plantilla d...
- ✓ 2 Seleccionar un nombre y ...
- ✓ 3 Seleccionar un recurso in...
- ✓ 4 Revisar detalles
- ✓ 5 Contratos de licencia
- ✓ 6 Configuración
- ✓ 7 Seleccionar almacenamiento...
- ✓ 8 Seleccionar redes
- 9 Personalizar plantilla**

10 Enlaces de vService

11 Listo para completar

Personalizar plantilla

Personalice las propiedades de implementación de esta solución de software.

Aplicación		5 configuración
Contraseña	Contraseña de la cuenta 'raíz' del dispositivo.	
	Contraseña
	Confirmar contraseña
Servidores NTP	Lista separada por comas de los nombres de host o las direcciones IP de los servidores NTP.	
	192.168.6.100	
Hostname	The host name for this virtual machine. Provide the FQDN if you use a static IP. Leave blank to reverse look up the IP address if you use DHCP.	
	miquelmariano.github.io	
DHCP IP Version	If DHCP is selected, determines whether IPv4 or IPv6 will be enabled at first boot.	
	ipv4	
Enable file integrity	Enables file integrity monitoring of the VR appliance.	
	<input checked="" type="checkbox"/>	
Networking Properties	6 configuración	

CANCEL

BACK

NEXT

Implementar plantilla de OVF

- ✓ 1 Seleccione una plantilla d...
- ✓ 2 Seleccionar un nombre y ...
- ✓ 3 Seleccionar un recurso in...
- ✓ 4 Revisar detalles
- ✓ 5 Contratos de licencia
- ✓ 6 Configuración
- ✓ 7 Seleccionar almacenamiento...
- ✓ 8 Seleccionar redes
- 9 Personalizar plantilla**

10 Enlaces de vService

11 Listo para completar

Networking Properties

6 configuración

Default Gateway	The default gateway address for this VM. (from the IP Pool associated with the vSphere network mapped to the 'Management Network' network)
	192.168.6.1
Domain Name	The domain name of this VM. (from the IP Pool associated with the vSphere network mapped to the 'Management Network' network)
	miquelmariano.github.io
Domain Search Path	The domain search path (comma or space separated domain names) for this VM. (from the IP Pool associated with the vSphere network mapped to the 'Management Network' network)
	miquelmariano.github.io
Domain Name Servers	The domain name server IP Addresses for this VM (comma separated). (from the IP Pool associated with the vSphere network mapped to the 'Management Network' network)
	192.168.6.100,192.168.6.1C
Management Network IP Address	The IP address for this interface.
	192.168.6.64
Management Network Netmask	The netmask or prefix for this interface.
	255.255.255.0

CANCEL

BACK

NEXT

Implementar plantilla de OVF

✓ 1 Seleccione una plantilla d... Enlaces de vService
✓ 2 Seleccionar un nombre y ... Seleccione las instancias de vService que debe enlazar la plantilla de OVF implementada.

✓ 3 Seleccionar un recurso in... vCenter Extension Installation
✓ 4 Revisar detalles This appliance requires a binding to the vCenter Extension vService, which allows it to register as a vCenter Extension at runtime.
✓ 5 Contratos de licencia Proveedor: vCenter Extension vService
✓ 6 Configuración
✓ 7 Seleccionar almacenamiento...
✓ 8 Seleccionar redes
✓ 9 Personalizar plantilla Estado de enlace:
10 Enlaces de vService Mensaje de validación:
11 Listo para completar ATTENTION: This virtual machine will gain unrestricted access to the vCenter server APIs. Make sure that the virtual machine is connected to a network where it can reach the URL '<https://vcenter-p.miquelmariano.github.io/vsm/extensionService>'.

CANCEL BACK NEXT

Resumen final de la implementación.

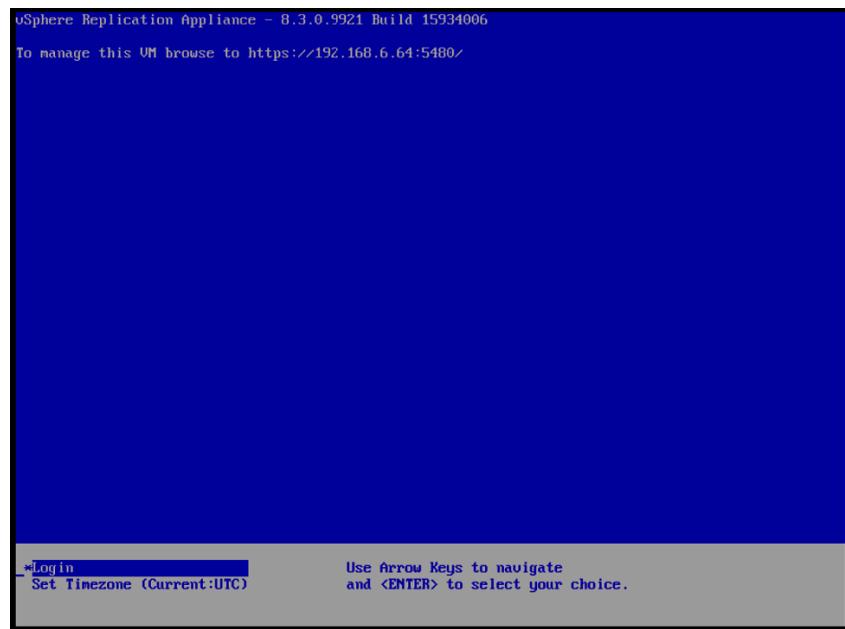
Implementar plantilla de OVF

✓ 1 Seleccione una plantilla d... Listo para completar
✓ 2 Seleccionar un nombre y ... Haga clic en Finalizar para iniciar la creación.

Nombre	vr-p
Nombre de plantilla	vSphere_Replication_OVF10
Tamaño de descarga	552,9 MB
Tamaño en disco	26,0 GB
Carpeta	Discovered virtual machine
Recurso	vesxi-001-p.miquelmariano.github.io
Asignación de almacenamiento	1
Todos los discos	Almacén de datos: Datastore-vesxi-001-p; formato: Puesta a cero lenta con aprovisionamiento grueso
Asignación de red	1
Management Network	VM Network
Configuración de asignación de IP	
Protocolo IP	IPv4
Asignación de IP	Estática - Manual

CANCEL BACK FINISH

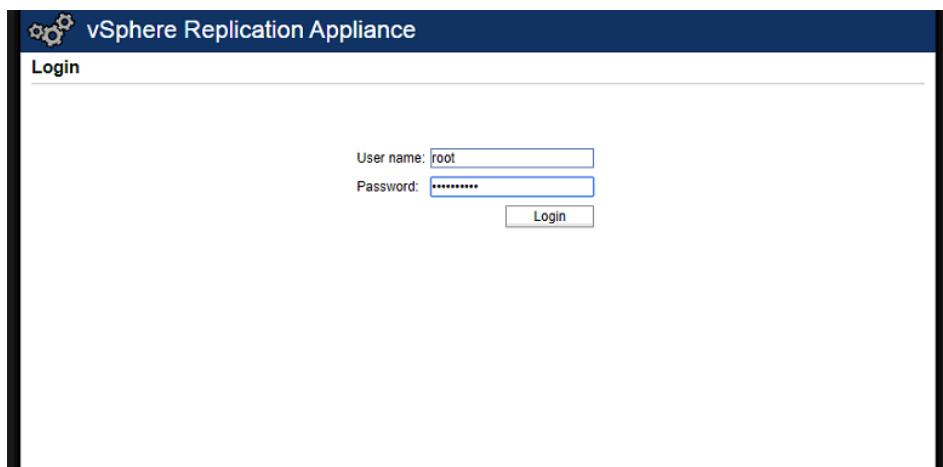
Una vez finalizado el despliegue ya estará disponible el portal de gestión en el puerto 5480



CONFIGURACIÓN INICIAL VSphere REPLICATION

Para la configuración de nuestro vSphere Replication, nos conectaremos a la IP de nuestro appliance y con el puerto de management 5480.

Recordad que las credenciales de root las hemos definido en el momento de desplegar nuestros appliances



Nos dirigiremos a la pestaña VR > Configuration

vSphere Replication Appliance

VR Network Update System Application Home | Help | Logout user root

Getting Started Configuration Security Support

Getting Started with vSphere Replication

These links will help you configure your VR appliance.

- 1. Configuration**
You can override the default configuration of the VR appliance.
[Configuration page](#)
- 2. Change Appliance Credentials**
Change the password of the VR appliance or review the SSL certificate that the appliance uses.
[Security page](#)

Replication Between Sites

Before you can replicate virtual machines between sites, you must deploy a VR appliance at each site and register it with the respective vCenter Server.

Y aquí el propio appliance ya nos propondrá la conexión al vCenter correspondiente. En mi caso, estamos configurando el site principal (hay que hacerlo en ambos) así que mi vCenter es el **vccenter-p.miquelmariano.github.io**.

Lo he comentado en otras ocasiones, no soy partidario de utilizar el usuario administrator@vsphere.local para servicios, así que me he creado uno exclusivo para esta conexión que he llamado vr@vsphere.local

vSphere Replication Appliance

VR Network Update System Application Home | Help | Logout user root

Getting Started Configuration Security Support

Startup Configuration

LookupService Address:

SSO Administrator:

Password:

VRM Host:

VRM Site Name:

vCenter Server Address:

vCenter Server Port:

vCenter Server Admin Mail:

Actions

IP Address for Incoming Storage Traffic:

SSL Certificate Policy

Accept only SSL certificates signed by a trusted Certificate Authority
(You must click the 'Save and Restart Service' button after changing this setting)

Install a new SSL Certificate

Generate a self-signed certificate

Upload PKCS#12 (*.pfx) file Ningún archivo seleccionado

Service Status

VRM service is stopped
 Tomcat service is running

Powered by VMware Studio

Aceptaremos el certificado de nuestro vCenter



Una vez finalizado, nos aparecerá el mensaje en verde de que la configuración se ha realizado correctamente.

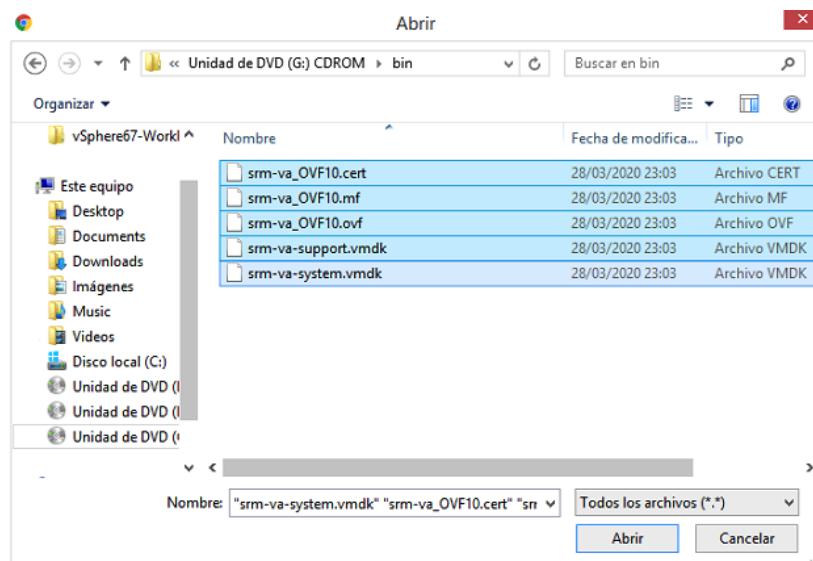
INSTALACIÓN SRM 8.3

Tal como comentaba al principio del capítulo, con la llegada de Site Recovery Manager versión 8.2 ha llegado la versión appliance del producto. Por lo tanto, ya no será necesario disponer de un servidor Windows para instalar el producto.

Desde el portal de [My.VMware](#) podremos hacer una búsqueda y descargarnos la iso con la última versión disponible.

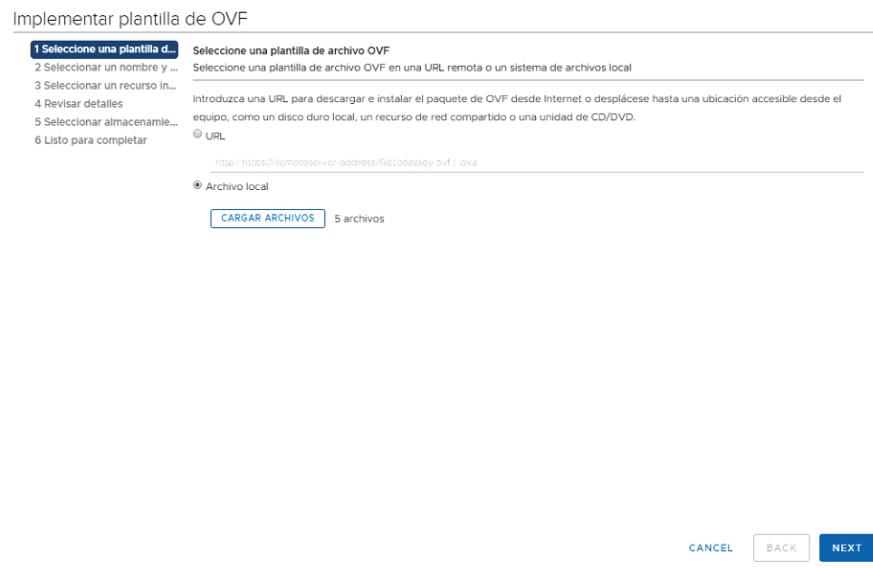
The screenshot shows the 'Download VMware Site Recovery Manager' page. A dropdown menu for 'Select Version' has '8.3' selected. To the right, a sidebar titled 'Product Resources' includes links for 'View My Download History', 'Product Info', 'Documentation', 'Knowledge Base', 'Receive Patch / Maintenance Alerts', and a 'Get Free Trial' button. Below the dropdown, a note states: 'Customers who have purchased VMware Site Recovery Manager can download their relevant installation package from the product download tab below.' A 'Read More' link is also present. At the bottom, a navigation bar offers links to 'Product Downloads', 'Drivers & Tools', 'Open Source', and 'Custom ISOs & Addons'. The main content area displays a table for 'Site Recovery Manager' with one row: 'VMware Site Recovery Manager 8.3.0' (Release Date: 2020-04-02) with a 'GO TO DOWNLOADS' button.

Dentro de la .iso descargada, en la carpeta *bin* encontraremos los ficheros necesarios para implementar la plantilla OVF

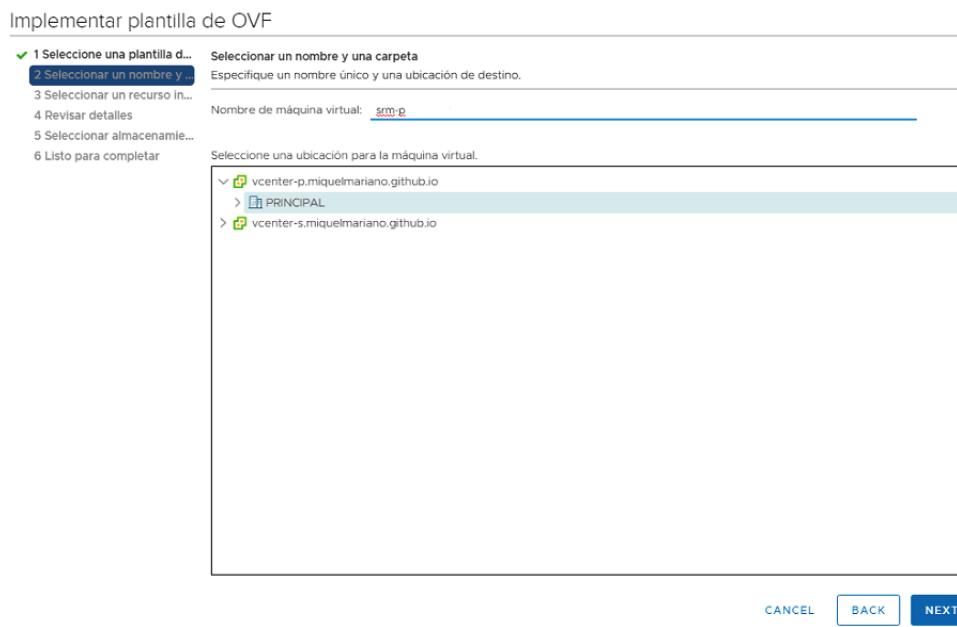


Hemos dicho que será necesario desplegar 2 SRM Servers, por lo que este proceso lo realizaremos 2 veces.

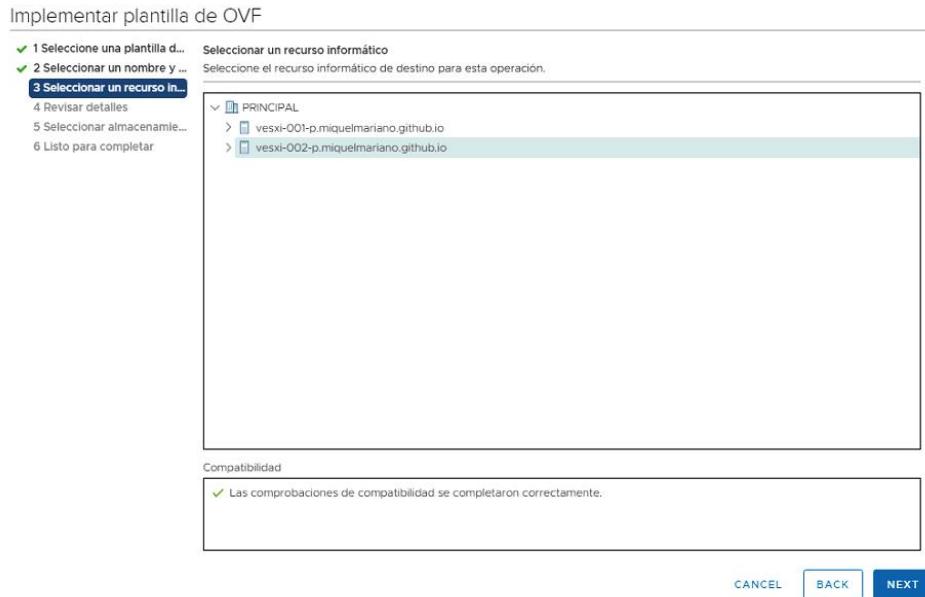
Accedemos a nuestro vCenter y nos dirigimos a la opción de “Implementar plantilla de OVF”, seleccionaremos los ficheros que anteriormente hemos comentado.



Seleccionaremos el nombre que le daremos a la VM, en mi caso y para distinguirlos fácilmente, **SRM-P** (principal) y **SRM-S** (secundario)



Seleccionamos el ESXi en el cual correrá la VM.



Revisión de detalles de producto y versión.

The screenshot shows the fourth step of the 'Implementar plantilla de OVF' (Deploy OVF Template) wizard. The title bar says 'Implementar plantilla de OVF'. The left sidebar has a checklist with steps 1, 2, 3, and 4 completed, and steps 5 through 10 listed below. Step 4 is highlighted with a dark blue background.

Revisar detalles
Compruebe los detalles de la plantilla.

Editor	VMware, Inc. (Certificado de confianza)
Producto	VMware Site Recovery Manager Appliance
Versión	8.3.0.4135
Proveedor	VMware, Inc.
Descripción	VMware Site Recovery Manager Appliance
Tamaño de descarga	939.3 MB
Tamaño en disco	1,3 GB (aprovisionamiento fino) 20,0 GB (aprovisionamiento grueso)

CANCEL **BACK** **NEXT**

Acuerdo de licencia.

Implementar plantilla de OVF

- ✓ 1 Seleccione una plantilla d...
- ✓ 2 Seleccionar un nombre y ...
- ✓ 3 Seleccionar un recurso in...
- ✓ 4 Revisar detalles

5 Contratos de licencia

- 6 Configuración
- 7 Seleccionar almacenamiento...
- 8 Seleccionar redes
- 9 Personalizar plantilla
- 10 Listo para completar

Contratos de licencia
Se debe aceptar el contrato de licencia de usuario final.

Lea y acepte los términos del contrato de licencia.

ACUERDO DE LICENCIA DE USUARIO FINAL DE
VMWARE

TENGA EN CUENTA QUE LAS CONDICIONES DE ESTE ACUERDO DE LICENCIA DE USUARIO FINAL REGIRÁN EL USO QUE HAGA DEL SOFTWARE, CON INDEPENDENCIA DE LOS TÉRMINOS QUE PUEDAN APARECER DURANTE LA INSTALACIÓN DEL SOFTWARE.

INFORMACIÓN IMPORTANTE: AL DESCARGAR, INSTALAR O UTILIZAR EL SOFTWARE, ACEPTA LAS CONDICIONES DE ESTE ACUERDO DE LICENCIA DE USUARIO FINAL (END USER LICENSE AGREEMENT) ("EULA"), TANTO SI LAS SUSCRIBE A TÍTULO PERSONAL COMO SI REPRESENTA A UNA PERSONA JURÍDICA. EN CASO DE NO ACEPTARLA, NO DEBERÁ DESCARGAR, INSTALAR NI

Acepto todos los contratos de licencia.

CANCEL BACK NEXT

Tamaño de nuestra VM. Al tratarse de un entorno de laboratorio, con la configuración más pequeña nos será suficiente.

Con la opción de 2vCPU se podrán proteger hasta un máximo de 1000 VMs.

Implementar plantilla de OVF

- ✓ 1 Seleccione una plantilla d...
- ✓ 2 Seleccionar un nombre y ...
- ✓ 3 Seleccionar un recurso in...
- ✓ 4 Revisar detalles
- ✓ 5 Contratos de licencia
- 6 Configuración**

- 7 Seleccionar almacenamiento...
- 8 Seleccionar redes
- 9 Personalizar plantilla
- 10 Listo para completar

Configuración
Seleccione una configuración de implementación

Configuración	Descripción
<input checked="" type="radio"/> 2 vCPU	Implementar la máquina virtual configurada con 2 vCPU y 8 GB de RAM
<input type="radio"/> 4 vCPU	
2 Elementos	

CANCEL BACK NEXT

Seleccionamos el datastore dónde desplegar la VM

Implementar plantilla de OVF

✓ 1 Seleccionar una plantilla d...
 ✓ 2 Seleccionar un nombre y ...
 ✓ 3 Seleccionar un recurso in...
 ✓ 4 Revisar detalles
 ✓ 5 Contratos de licencia
 ✓ 6 Configuración
7 Seleccionar almacenamiento...
 8 Seleccionar redes
 9 Personalizar plantilla
 10 Listo para completar

Seleccionar almacenamiento
Seleccione el almacenamiento para los archivos de configuración y de disco

Cifrar esta máquina virtual (Requiere un servidor de administración de claves)

Seleccione el formato de disco virtual:
Puesta a cero lenta con aprovisionamiento grueso

Directiva de almacenamiento de máquina virtual:
Valor predeterminado de almacenamiento de datos

Nombre	Capacidad	Aprovisionado	Libre	Tipo	Clúster
Datastore-vmx-002-p	199,75 GB	19,52 GB	194,1 GB	VMFS 6	

Compatibilidad

✓ Las comprobaciones de compatibilidad se completaron correctamente.

CANCEL BACK **NEXT**

Seleccionamos el VM Portgroup correspondiente a nuestra red.

Implementar plantilla de OVF

✓ 1 Seleccionar una plantilla d...
 ✓ 2 Seleccionar un nombre y ...
 ✓ 3 Seleccionar un recurso in...
 ✓ 4 Revisar detalles
 ✓ 5 Contratos de licencia
 ✓ 6 Configuración
7 Seleccionar almacenamiento...
8 Seleccionar redes
 9 Personalizar plantilla
 10 Listo para completar

Seleccionar redes
Seleccione una red de destino para cada red de origen.

Red de origen	Red de destino
Network 1	VM Network

Configuración de asignación de IP

Asignación de IP:
Estática - Manual

Protocolo IP:
IPv4

CANCEL BACK **NEXT**

A partir de aquí es dónde se configuran los parámetros del propio Appliance:

- Credenciales
- Servidor de tiempo

- Hostname
- Configuración IP

Implementar plantilla de OVF

✓ 1 Seleccione una plantilla d...
 ✓ 2 Seleccionar un nombre y ...
 ✓ 3 Seleccionar un recurso in...
 ✓ 4 Revisar detalles
 ✓ 5 Contratos de licencia
 ✓ 6 Configuración
 ✓ 7 Seleccionar almacenamie...
 ✓ 8 Seleccionar redes
9 Personalizar plantilla
 10 Listo para completar

Personalizar plantilla
 Personalice las propiedades de implementación de esta solución de software.

Aplicación		8 configuración
Habilitar SSHD		
Determina si el servicio SSHD se habilitará y se iniciará de forma predeterminada en el dispositivo. <input checked="" type="checkbox"/>		
Contraseña inicial del usuario raíz		
Se utilizará como contraseña inicial para la cuenta de usuario raíz.		
Contraseña Confirmar contraseña		
Contraseña inicial del usuario administrador		
Se utilizará como contraseña inicial para la cuenta del usuario administrador.		
Contraseña Confirmar contraseña		
Servidores NTP		
Lista separada por comas de los nombres de host o las direcciones IP de los servidores NTP. 192.168.6.100		
Nombre del host		
El nombre del host de esta máquina virtual. Deje este espacio en blanco para intentar realizar una búsqueda inversa de la dirección IP. srm-p.miquelmariano.gitf		
Contraseña inicial de la base de datos		
Se utilizará como contraseña inicial de la base de datos.		

CANCEL **BACK** **NEXT**

Implementar plantilla de OVF

✓ 1 Seleccione una plantilla d...
 ✓ 2 Seleccionar un nombre y ...
 ✓ 3 Seleccionar un recurso in...
 ✓ 4 Revisar detalles
 ✓ 5 Contratos de licencia
 ✓ 6 Configuración
 ✓ 7 Seleccionar almacenamie...
 ✓ 8 Seleccionar redes
9 Personalizar plantilla
 10 Listo para completar

Contraseña inicial de la base de datos		8 configuración
Contraseña Confirmar contraseña		
Marca de integridad del archivo		Se utilizará como marca para indicar que se debe habilitar la integridad del archivo. <input type="checkbox"/>
Marca HCX		Se utilizará como marca para indicar que se debe habilitar la compatibilidad con HCX. <input type="checkbox"/>
Propiedades de redes		8 configuración
Host Network IP Address Family Network IP address family (i.e., 'ipv4' or 'ipv6'). IPv4		
Host Network Mode Network mode (i.e., 'static', 'dhcp', or 'autoconf' (IPv6 only)). static		
Puerta de enlace predeterminada La dirección de puerta de enlace predeterminada para esta máquina virtual. (Del grupo de direcciones IP asociado a la red de vSphere asignada a la red 'Red 1') 192.168.6.1		
Nombre de dominio El nombre de dominio de esta máquina virtual. (Del grupo de direcciones IP asociado a la red de vSphere asignada a la red 'Red 1')		

CANCEL **BACK** **NEXT**

Implementar plantilla de OVF

<ul style="list-style-type: none"> ✓ 1 Seleccionar una plantilla d... ✓ 2 Seleccionar un nombre y ... ✓ 3 Seleccionar un recurso in... ✓ 4 Revisar detalles ✓ 5 Contratos de licencia ✓ 6 Configuración ✓ 7 Seleccionar almacenamie... ✓ 8 Seleccionar redes 9 Personalizar plantilla <p>10 Listo para completar</p>	<p>Puerta de enlace predeterminada La dirección de puerta de enlace predeterminada para esta máquina virtual. (Del grupo de direcciones IP asociado a la red de vSphere asignada a la red 'Red 1') <input type="text" value="192.168.6.1"/></p> <p>Nombre de dominio El nombre de dominio de esta máquina virtual. (Del grupo de direcciones IP asociado a la red de vSphere asignada a la red 'Red 1') <input type="text" value="miquelmariano.github.io"/></p> <p>Ruta de búsqueda de dominio La ruta de búsqueda de dominio (nombres de dominio separados por espacios o comas) para esta máquina virtual. (Del grupo de direcciones IP asociado a la red de vSphere asignada a la red 'Red 1') <input type="text" value="miquelmariano.github.io"/></p> <p>Servidores de nombres de dominio Direcciones IP del servidor de nombres de dominio para esta máquina virtual (separadas por comas). (Del grupo de direcciones IP asociado a la red de vSphere asignada a la red 'Red 1') <input type="text" value="192.168.6.100,192.168.6.1c"/></p> <p>Dirección IP de Red 1 La dirección IP de esta interfaz. <input type="text" value="192.168.6.63"/></p> <p>Prefijo de red de red 1 El prefijo de esta interfaz. <input type="text" value="0"/></p>
--	---

CANCEL BACK NEXT

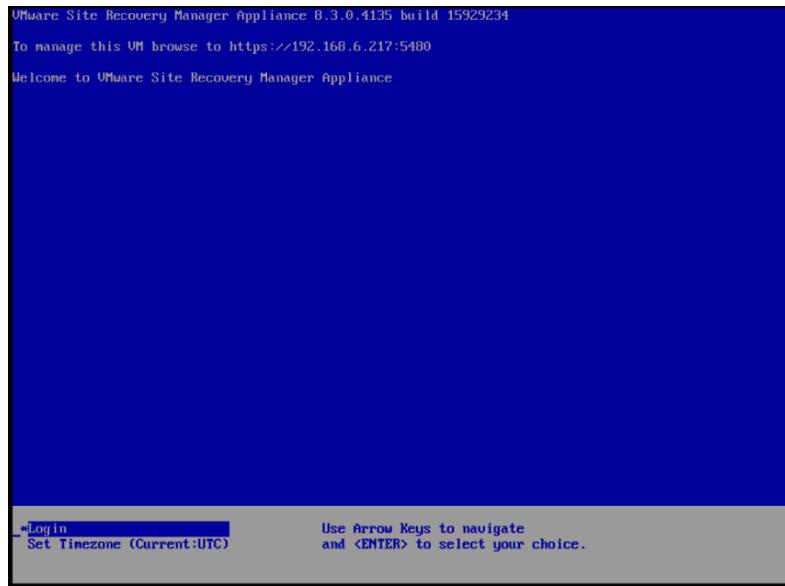
Resumen final antes de proceder a la implementación.

Implementar plantilla de OVF

<ul style="list-style-type: none"> ✓ 1 Seleccionar una plantilla d... ✓ 2 Seleccionar un nombre y ... ✓ 3 Seleccionar un recurso in... ✓ 4 Revisar detalles ✓ 5 Contratos de licencia ✓ 6 Configuración ✓ 7 Seleccionar almacenamie... ✓ 8 Seleccionar redes ✓ 9 Personalizar plantilla <p>10 Listo para completar</p>	<p>Lista para completar Haga clic en Finalizar para iniciar la creación.</p>																										
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td>Nombre</td> <td>srm-p</td> </tr> <tr> <td>Nombre de plantilla</td> <td>srm-va_OVF10</td> </tr> <tr> <td>Tamaño de descarga</td> <td>939,3 MB</td> </tr> <tr> <td>Tamaño en disco</td> <td>20,0 GB</td> </tr> <tr> <td>Carpeta</td> <td>PRINCIPAL</td> </tr> <tr> <td>Recurso</td> <td>vesxi-002-p.miquelmariano.github.io</td> </tr> <tr> <td>Asignación de almacenamiento</td> <td>1</td> </tr> <tr> <td>Todos los discos</td> <td>Almacén de datos: Datastore-vesxi-002-p; formato: Puesta a cero lenta con aprovisionamiento grueso</td> </tr> <tr> <td>Asignación de red</td> <td>1</td> </tr> <tr> <td>Network 1</td> <td>VM Network</td> </tr> <tr> <td>Configuración de asignación de IP</td> <td></td> </tr> <tr> <td>Protocolo IP</td> <td>IPV4</td> </tr> <tr> <td>Asignación de IP</td> <td>Estática - Manual</td> </tr> </table>		Nombre	srm-p	Nombre de plantilla	srm-va_OVF10	Tamaño de descarga	939,3 MB	Tamaño en disco	20,0 GB	Carpeta	PRINCIPAL	Recurso	vesxi-002-p.miquelmariano.github.io	Asignación de almacenamiento	1	Todos los discos	Almacén de datos: Datastore-vesxi-002-p; formato: Puesta a cero lenta con aprovisionamiento grueso	Asignación de red	1	Network 1	VM Network	Configuración de asignación de IP		Protocolo IP	IPV4	Asignación de IP	Estática - Manual
Nombre	srm-p																										
Nombre de plantilla	srm-va_OVF10																										
Tamaño de descarga	939,3 MB																										
Tamaño en disco	20,0 GB																										
Carpeta	PRINCIPAL																										
Recurso	vesxi-002-p.miquelmariano.github.io																										
Asignación de almacenamiento	1																										
Todos los discos	Almacén de datos: Datastore-vesxi-002-p; formato: Puesta a cero lenta con aprovisionamiento grueso																										
Asignación de red	1																										
Network 1	VM Network																										
Configuración de asignación de IP																											
Protocolo IP	IPV4																										
Asignación de IP	Estática - Manual																										

CANCEL BACK FINISH

Una vez finalizado el despliegue, ya estará disponible el portal de gestión en el puerto 5480



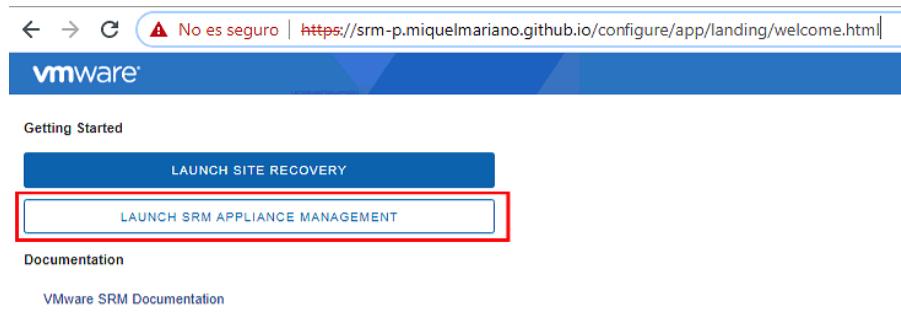
CONFIGURACIÓN INICIAL SRM

La configuración inicial de SRM es básicamente vincular la instancia que acabamos de instalar a un vCenter. Recordad que la asociación es 1:1, es decir, cada SRM Server estará asociado a un único vCenter. En nuestro caso, la asignación será:

srm-p.miquelmariano.github.io > vcenter-p.miquelmariano.github.io

srm-s.miquelmariano.github.io > vcenter-s.miquelmariano.github.io

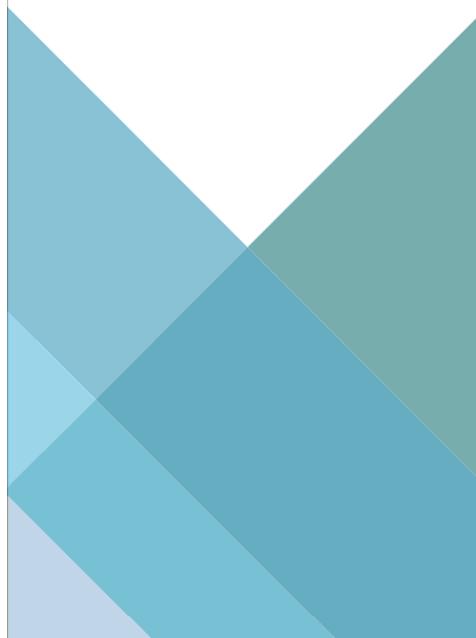
Accederemos al portal de administración de nuestro SRM en la siguiente URL y entraremos en la parte de Management en el puerto 5480



Se accede con el usuario **admin**, y la contraseña que hemos configurado en el momento del despliegue.

Bienvenido a VMware SRM Appliance Management

Nombre de usuario	<input type="text"/>
Contraseña	<input type="password"/> <small>Mostrar</small>
INICIAR SESIÓN	



A parte de la configuración standard que aparece en el menú izquierdo, lo que nos interesa es la opción “**CONFIGURAR EL DISPOSITIVO**”

Producto	VMware vCenter Site Recovery Manager
Versión	8.3.0
Compilación	15929234

Este primer SRM Server, al ser el PRINCIPAL lo vincularemos con el vCenter-P, pero recordad que habrá que volver a hacer este mismo proceso para el SECUNDARIO.

Si os fijáis, he creado un usuario en el SSO llamado srm@vsphere.local y es que no soy muy partidario de utilizar el *administrator*. Mi consejo es que siempre que podáis, creáros los usuarios de servicio necesarios para utilizar con los sistemas que se conectarán a la infraestructura vSphere.

Configurar Site Recovery Manager

Platform Services Controller

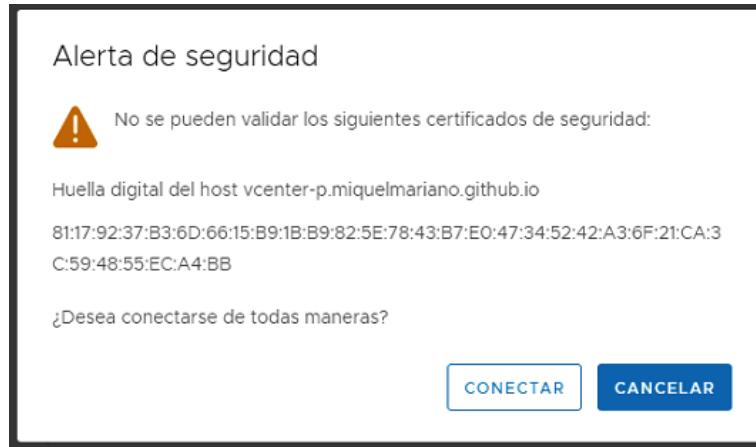
Introduzca la información de Platform Services Controller de la instancia de vCenter Server para la que desea configurar Site Recovery Manager.

Nombre del host de PSC	vcenter-p.miquelmariano.github.io
Puerto de PSC	443
Nombre de usuario	srm@vsphere.local
Contraseña

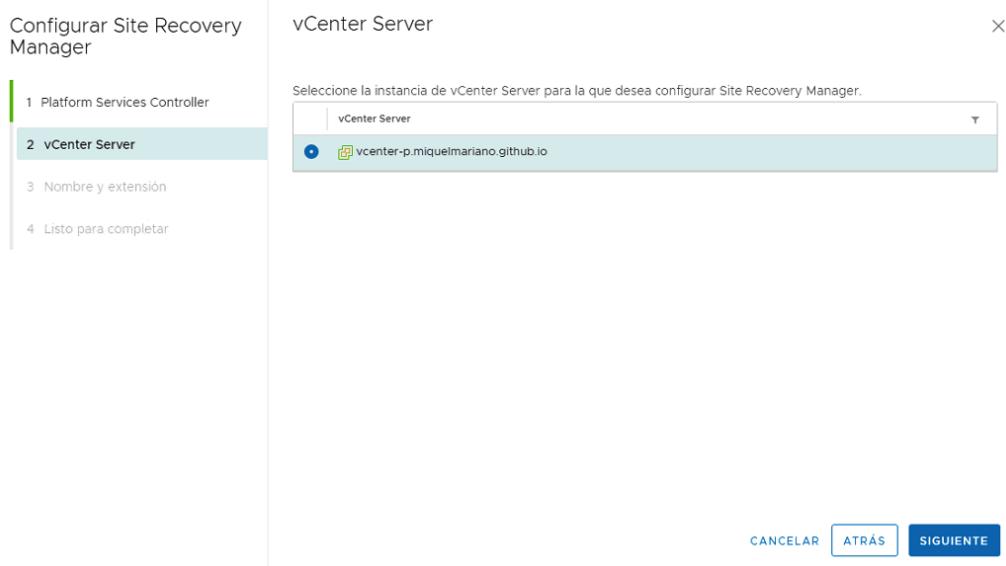
Nota: Si se lo pide el programa de configuración, deberá aceptar el certificado para que la configuración continúe.

CANCELAR **SIGUIENTE**

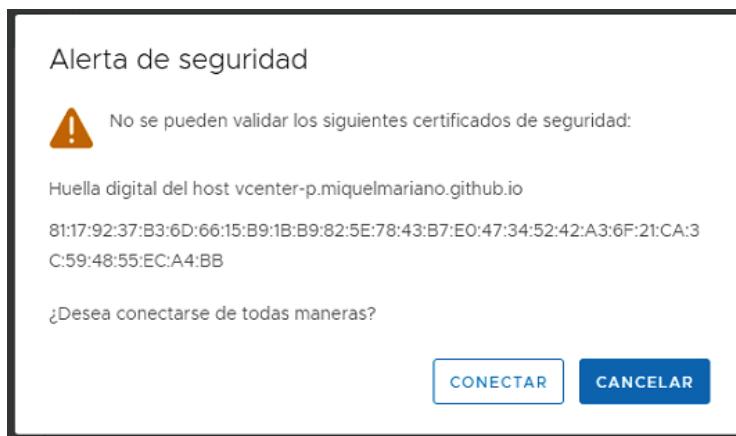
Tendremos que aceptar los certificados autofirmados del vCenter.



Seleccionamos el vCenter.

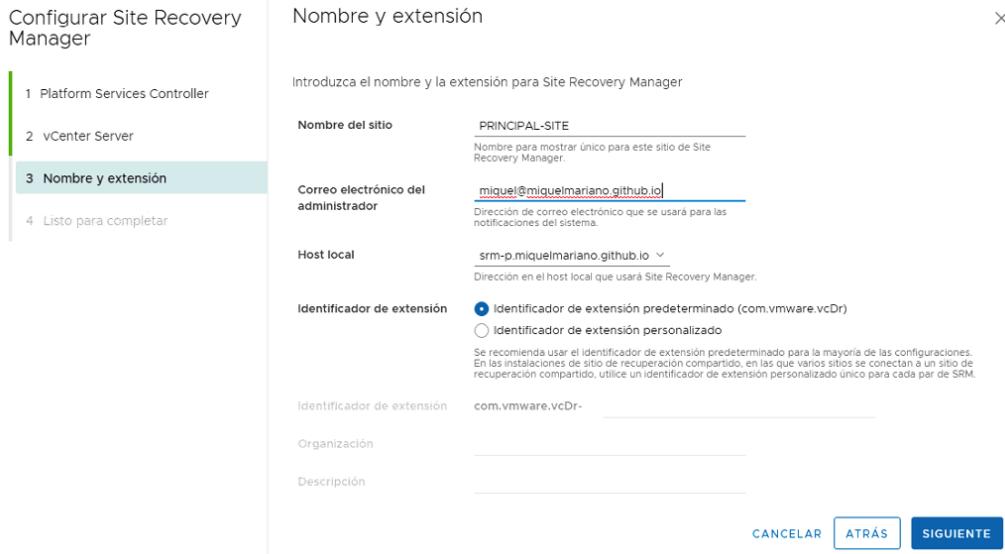


De nuevo, hay que aceptar el certificado.



Aquí es dónde definiremos algunos parámetros interesantes, no os preocupéis, se pueden cambiar a posteriori:

- Nombre del sitio > Para ser originales, en mi caso será PRINCIPAL y SECUNDARIO, pero en un entorno real, podremos el nombre del CPD o la ubicación o un nombre que nos identifique claramente cada site.
- Correo
- Servidor SRM
- Identificador de extensión > Al asociar SRM Server a un vCenter, automáticamente se nos registrará un plugin en nuestro vSphere Web Client para administrar todo el entorno. En este punto nos está preguntando por el nombre de la extensión, y mi recomendación es dejarlo con el nombre por defecto.



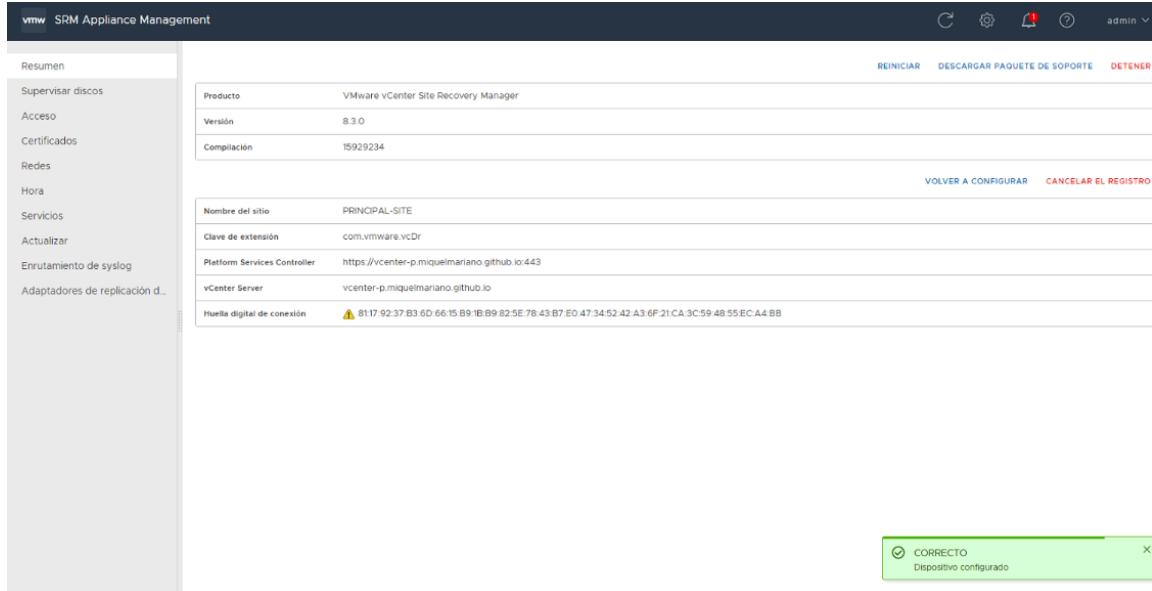
Resumen final.



Barra de progreso del proceso de configuración.



Configuración finalizada correctamente.



Insisto, este proceso lo tendrás que hacer 2 veces, para el site PRINCIPAL y para el site SECUNDARIO. Y si todo ha ido bien, en el vSphere Web Client os habrá aparecido un nuevo botón llamado “**Site Recovery**”. Desde aquí podremos acceder y ver el estado no solo de nuestro SRM, sino también de vSphere Replication, que a continuación veremos cómo se instala.

CONFIGURAR PAR DE SITIOS

El siguiente paso en la configuración de Site Recovery Manager, será configurar el par de sitios. Esta acción es simplemente vincular nuestros 2 vCenters con nuestros 2 SRM Managers, que a su vez tienen los 2 VR Appliances.

Podremos entrar en la configuración de SRM desde el propio vCenter:

The screenshot shows the vSphere Client interface with the URL `vcenter-p.miquelmaniano.github.io/ui/app/plugin/com.vmware.vrUi/com.vmware.draasclientplugin.dashboardView`. The left sidebar has a red box around the 'Site Recovery' item. The main pane is titled 'Site Recovery' and displays two sections: 'vcenter-p.miquelmaniano.github.io' and 'vcenter-s.miquelmaniano.github.io'. Each section lists 'vSphere Replication' and 'Site Recovery Manager' with 'Aceptar' (Accept) and 'CONFIGURAR' (Configure) buttons. Red arrows point from the 'Site Recovery' button in the top section to the 'Site Recovery Manager' buttons in both sections. A blue box highlights the 'ABRIR Site Recovery' (Open Site Recovery) button at the bottom of each section.

O directamente desde la propia URL del SRM Manager:

The screenshot shows the VMware Site Recovery Manager landing page with the URL `srm-p.miquelmaniano.github.io/configure/app/landing/welcome.html`. It features a 'Getting Started' section with a large blue 'LAUNCH SITE RECOVERY' button, which has a red arrow pointing to it. Below it is a smaller 'LAUNCH SRM APPLIANCE MANAGEMENT' button. The page also includes a 'Documentation' section with a link to 'VMware SRM Documentation'.

Justo al entrar, nos dará la opción de crear **NUEVO PAR DE SITIOS**

The screenshot shows the VMware Site Recovery interface. At the top, it says "vmw Site Recovery Menú". Below that is a button labeled "NUEVO PAR DE SITIOS". A section titled "Replicaciones dentro de la misma instancia de vCenter Server" contains two items: "dentro de vcenter-p.miquelmariano.github.io" and "dentro de vcenter-s.miquelmariano.github.io". At the bottom of this section is a blue "VER DETALLES" button.

Siguiendo con el asistente, nos preguntará por el sitio primario, en nuestro caso, recordad que es el **vcenter-p** y el site secundario **vcenter-s**

Utilizaremos el mismo usuario de SSO que ya teníamos creado previamente, llamado **srm@vsphere.local**

The screenshot shows the "Nuevo par de sitios" wizard. On the left, a sidebar lists steps: 1. Detalles del sitio (selected), 2. vCenter Server y servicios, 3. Listo para completar. The main area is titled "Detalles del sitio" and "Primer sitio". It asks to "Seleccione las instancias locales de vCenter Server que deseé emparejar." A dropdown menu shows "vCenter Server" with two options: "vcenter-p.miquelmariano.github.io" (selected with a blue dot) and "vcenter-s.miquelmariano.github.io". Below this is a "Segundo sitio" section with fields for "Nombre del host de PSC" (vcenter-s.miquelmariano.github.io), "Puerto de PSC" (443), "Nombre de usuario" (srm@vsphere.local), and "Contraseña" (redacted). At the bottom are "CANCELAR" and "SIGUIENTE" buttons.

Es probable que tengamos que aceptar el certificado autofirmado de nuestro vCenter

Alerta de seguridad



Site Recovery Client no puede validar los siguientes certificados de seguridad:

Huella digital del host vcenter-s.miquelmariano.github.io

82:58:88:33:CF:4B:56:7A:3E:16:45:C8:74:26:E5:E1:5E:D5:AB:16:DB:50:73:A9:3E:
9B:FC:87:CB:FF:BE:F3

¿Desea conectarse de todas maneras?

[CONECTAR](#)

[CANCELAR](#)

Una vez conectado, seleccionaremos la instancia de Site Recovery Manager, así como la instancia de vSphere Replication.

Nuevo par de sitios

1 Detalles del sitio

2 vCenter Server y servicios

3 Listo para completar

vCenter Server y servicios

Seleccione la instancia de vCenter Server que deseé emparejar.

Servicio	vcenter-p.miquelmariano.github.io	vcenter-s.miquelmariano.github.io
<input checked="" type="checkbox"/> Site Recovery Manager...	PRINCIPAL-SITE	SECUNDARIO-SITE
<input checked="" type="checkbox"/> vSphere Replication	vcenter-p.miquelmariano.github.io	vcenter-s.miquelmariano.github.io

Se han identificado los siguientes servicios en las instancias de vCenter Server.
Seleccione los que deseé emparejar:

[CANCELAR](#) [ATRÁS](#) [SIGUIENTE](#)

Otra vez habrá que aceptar certificados, esta vez del site secundario

Alerta de seguridad



Site Recovery Manager en srm-p.miquelmariano.github.io no puede validar los siguientes certificados de seguridad:

Huella digital del host vcenter-s.miquelmariano.github.io

82:58:88:33:CF:4B:56:7A:3E:16:45:C8:74:26:E5:E1:5E:D5:AB:16:DB:50:73:A9:3E:
9B:FC:87:CB:FF:BE:F3

Huella digital del host srm-s.miquelmariano.github.io

9E:D0:D6:9E:FF:B2:51:97:6D:80:26:B6:A1:D3:A5:D7:6A:12:20:DD:8E:E4:91:50:2
0:83:E6:56:2B:D0:43:3A

¿Desea conectarse de todas maneras?

[CONECTAR](#)

[CANCELAR](#)

Alerta de seguridad

⚠ Site Recovery Manager en srm-s.miquelmariano.github.io no puede validar los siguientes certificados de seguridad:

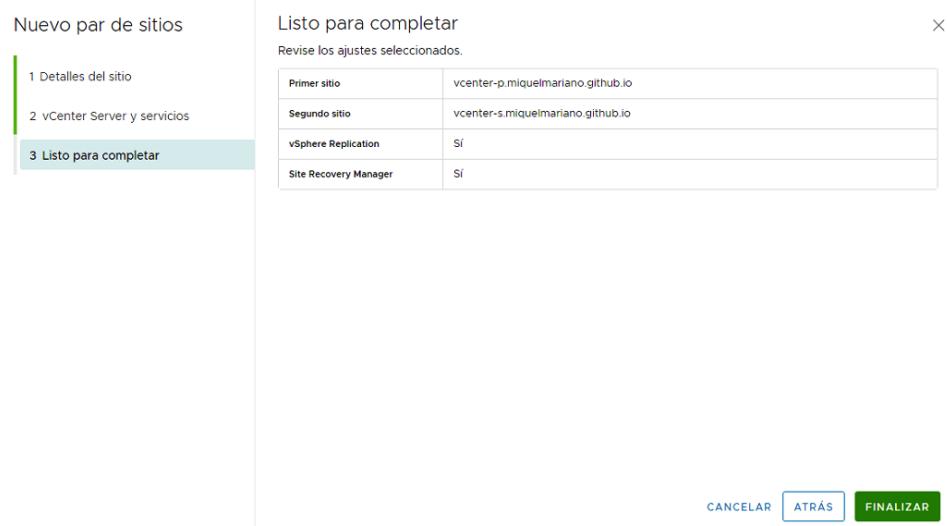
Huella digital del host vcenter-p.miquelmariano.github.io
81:17:92:37:B3:6D:66:15:B9:1B:B9:82:5E:78:43:B7:E0:47:34:52:42:A3:6F:21:CA:3C:59:48:55:EC:A4:BB

Huella digital del host srm-p.miquelmariano.github.io
9A:2E:CB:83:06:40:C5:E4:F2:AA:9D:D1:48:39:E6:16:B9:DD:C2:40:DA:AD:DF:8E:70:0D:89:AF:33:5E:16:F9

¿Desea conectarse de todas maneras?

[CONECTAR](#) [CANCELAR](#)

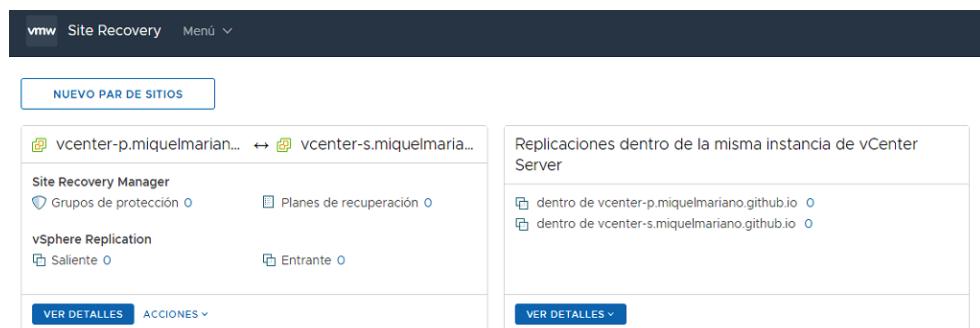
Resumen de la conexión



Listo para completar	
Revise los ajustes seleccionados.	
Primer sitio	vcenter-p.miquelmariano.github.io
Segundo sitio	vcenter-s.miquelmariano.github.io
vSphere Replication	Sí
Site Recovery Manager	Sí

[CANCELAR](#) [ATRÁS](#) [FINALIZAR](#)

Tras finalizar el asistente, nos aparecerá el recuadro con la conexión, los grupos de protección, y los planes de recuperación, así como las réplicas que se están realizando con vSphere Replication.



RESOURCE MAPPINGS

Después de cualquier implementación de Site Recovery Manager, será necesario configurar un “inventario de asignaciones” o resource mappings.

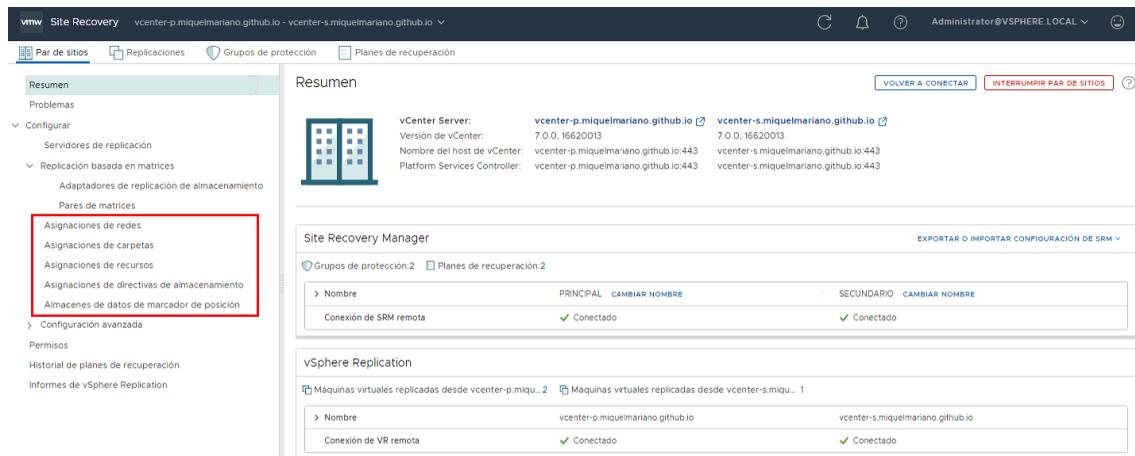
Esta configuración no es más que “decirle” al sistema qué correspondencia tendrá cada recurso del sitio primario en el sitio secundario.

Estas asignaciones incluirán los recursos de red (Virtual Machine PortGroup), carpetas, recursos de cómputo (Host, Cluster, vAPP, Resource Group) y políticas de almacenamiento.

Debemos mapear todos estos recursos desde el sitio protegido o primario, hasta un sitio de recuperación o secundario. Evidentemente, si queremos invertir los roles de primario y secundario, y “proteger” VMs en sentido inverso, también debemos realizar esta asignación en esta dirección. A continuación, veremos lo fácil que es.

En un entorno ideal, tanto el sitio principal como el secundario deberían ser simétricos, pero en la práctica, muchas veces eso no es posible. Este hecho puede hacer que debamos tener una mayor gestión de los resource mappings.

En el menú principal de la configuración del par de sitios, podremos encontrar las opciones de mapeo.



The screenshot shows the VMware Site Recovery interface. The left sidebar has a red box around the 'Asignaciones de redes' (Network Assignments) option under 'Pares de sitios' (Site Pairs). The main content area displays the 'Resumen' (Summary) page for a site pair. It shows two vCenter servers: 'vcenter-p.miquelmariano.github.io' (Principal) and 'vcenter-s.miquelmariano.github.io' (Secondary). Below this, there are sections for 'Site Recovery Manager' and 'vSphere Replication', each with tables for network assignments and replication connections. Buttons for 'EXPORTAR O IMPORTAR CONFIGURACIÓN DE SRM' and 'INTERRUMPIR PAR DE SITIOS' are visible at the top right.

A continuación, os enseño como ejemplo el mapeo de las redes. En este caso, en el vCenter principal “mapearemos” la red **P-VM_Network** con la red **S-VM_Network** en el vCenter secundario.

Asignaciones de redes

	vcenter-p.miquelmariano.github.io	vcenter-s.miquelmariano.github.io			
NUEVO	<input type="checkbox"/> P-VM_Network	Red de recuperación S-VM_Network	Asignación inversa Sí	Red de prueba Red aislada (creación automática)	Personalización de IP No

No hay ninguna asignación de red seleccionada.

Y en la otra pestaña, vemos que el mapeo es el mismo, pero en sentido contrario

Asignaciones de redes

	vcenter-p.miquelmariano.github.io	vcenter-s.miquelmariano.github.io				
NUEVO	<input type="checkbox"/> S-VM_Network	P-VM_Network	Red de recuperación P-VM_Network	Asignación inversa Sí	Red de prueba Red aislada (creación automática)	Personalización de IP No

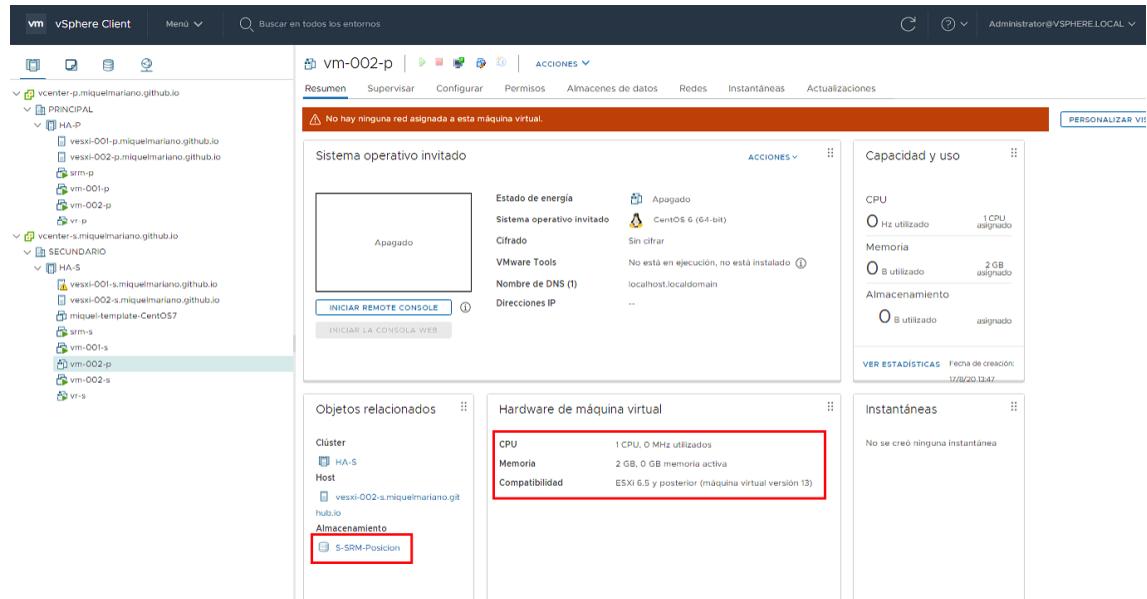
No hay ninguna asignación de red seleccionada.

ALMACENES DE MARCDOR DE POSICIÓN

Los almacenes de marcador de posición son datastores que SRM utiliza para almacenar los archivos de las VMs, cuando éstas están protegidas.

SRM reservará este espacio para VMs protegidas en el sitio de recuperación, y lo usará para registrar la máquina virtual de marcador de posición con el vCenter en el sitio secundario o de recuperación.

Básicamente, este datastore se utiliza para tener las VMs inventariadas en sitio de recuperación.



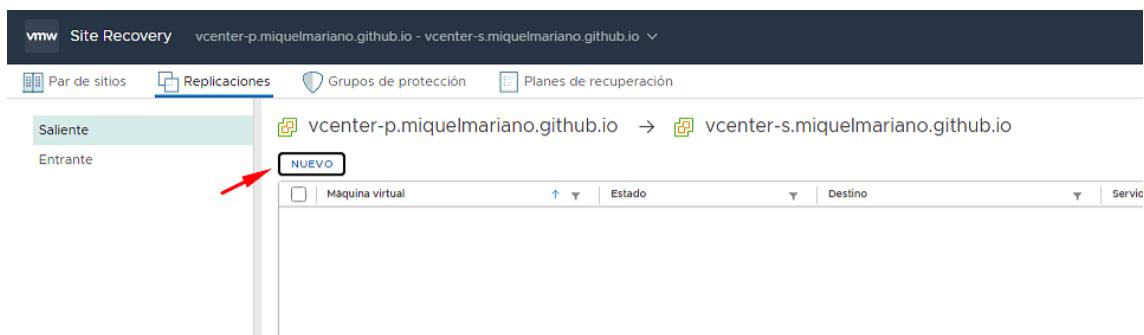
OPERACIONES CON SRM

CREAR RÉPLICA ENTRE SITES

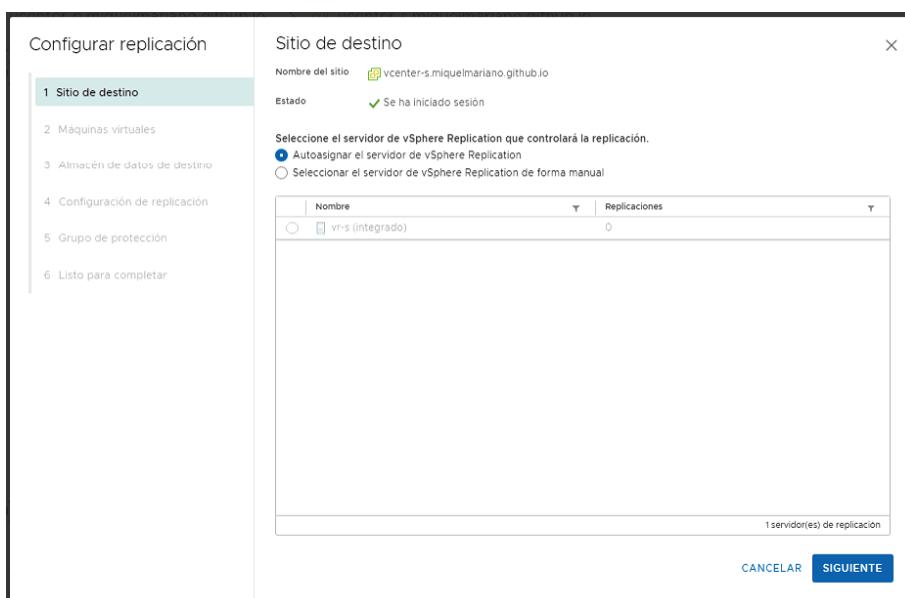
Lo primero que tendremos que crear para utilizar Site Recovery Manager es una replicación entre sites.

Ya hemos visto que podemos replicar a nivel de VM con vSphere Replication o basándonos en replicación a nivel de cabina. En nuestro laboratorio, estamos utilizando vSphere Replication.

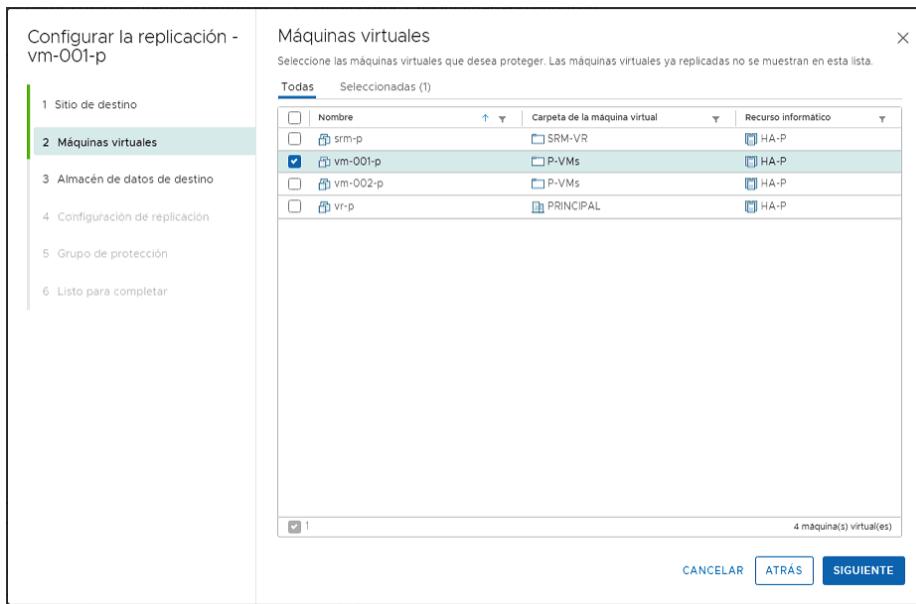
Nos dirigiremos a la pestaña de Replicaciones y añadiremos una nueva.



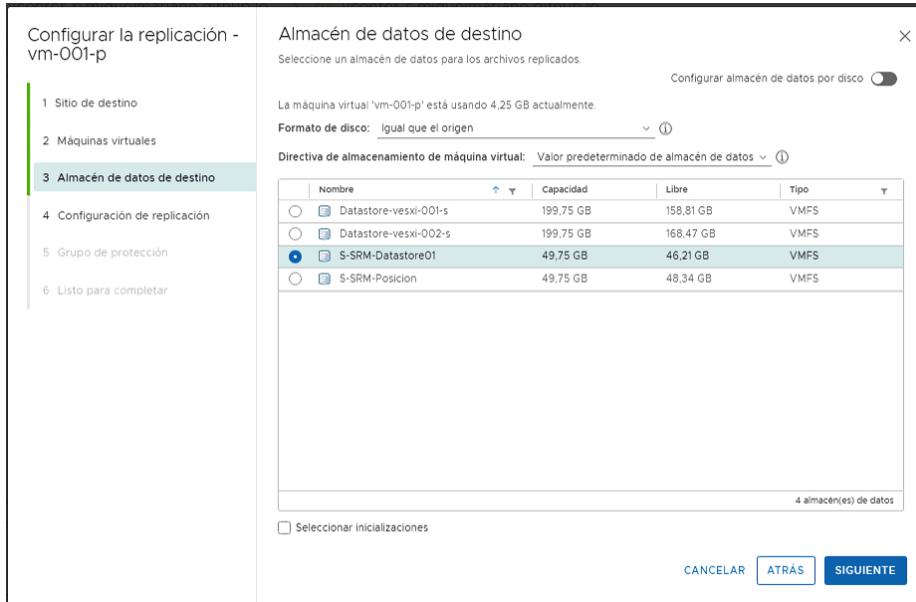
Podemos dejar que por defecto el sistema elija el servidor de vSphere Replication, o manualmente elegir el que deseemos. En nuestro caso, solo disponemos de un VR Appliance



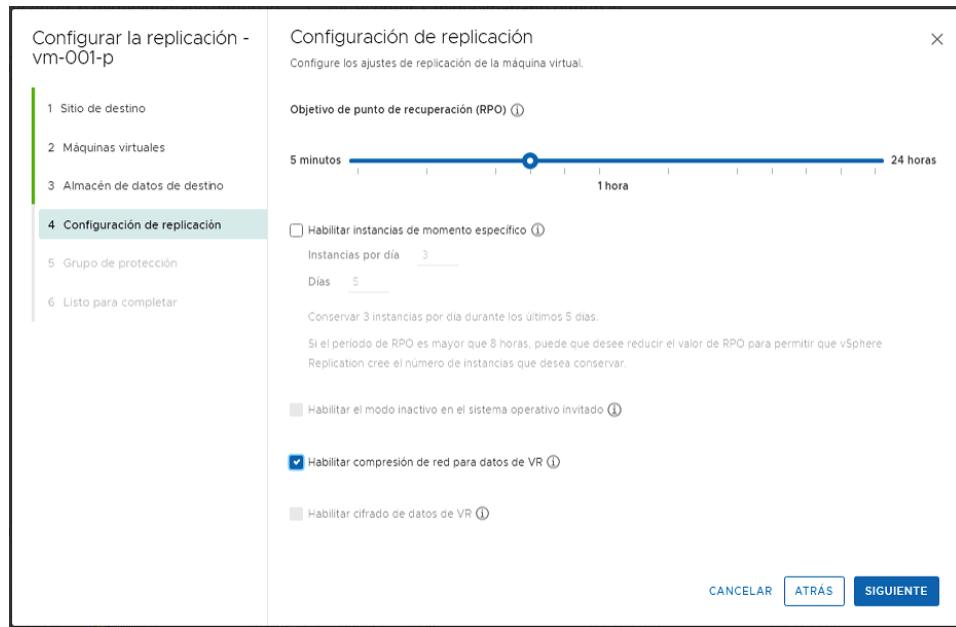
Del inventario de nuestro vCenter, seleccionaremos la VM o VMs que queramos replicar



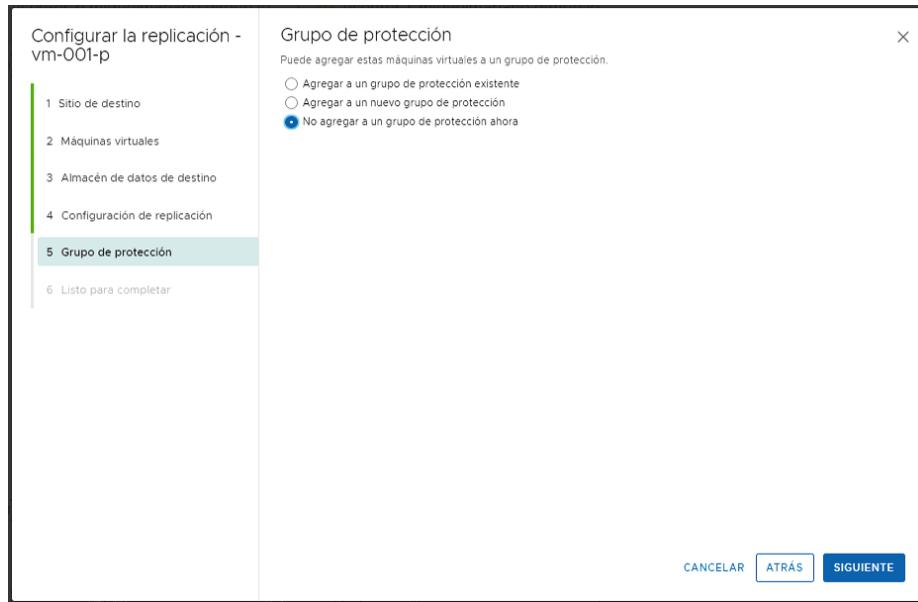
Seleccionaremos en el sitio secundario o de recuperación el datastore en el cual se almacenará esa réplica



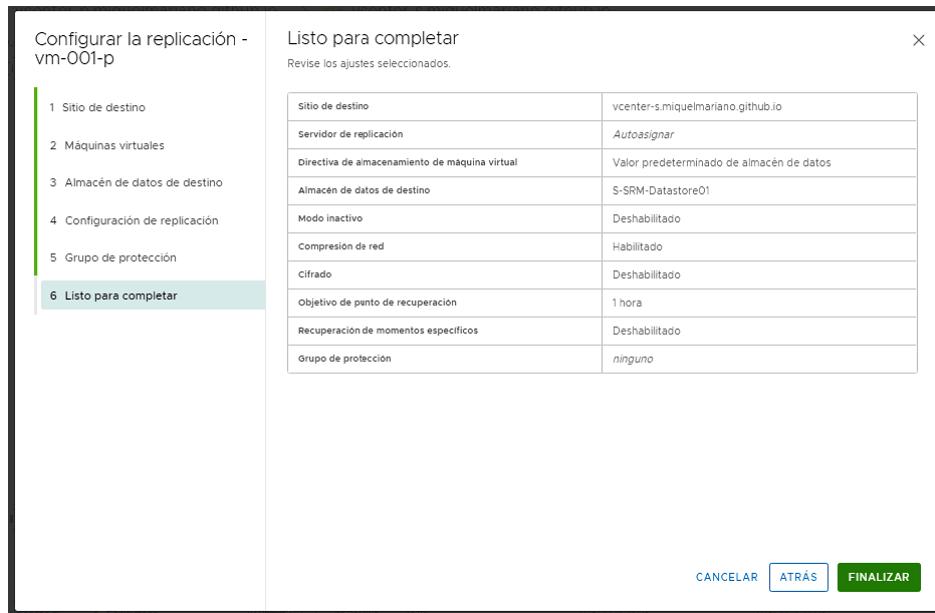
En este punto, configuraremos nuestro RPO. Con vSphere Replication, tenemos desde los 5 minutos hasta las 24 horas. Como es de suponer, cuanto menor tengamos el RPO, más tráfico en nuestra LAN se generará para la replicación.



En este punto, nos permitirá agregar esa o esas VMs a un grupo de protección. De momento no hemos hablado de ello, así que no seleccionamos ninguno. Ya lo veremos más adelante.



Y finalizaremos el wizard con un pequeño resumen de las acciones que se realizarán.



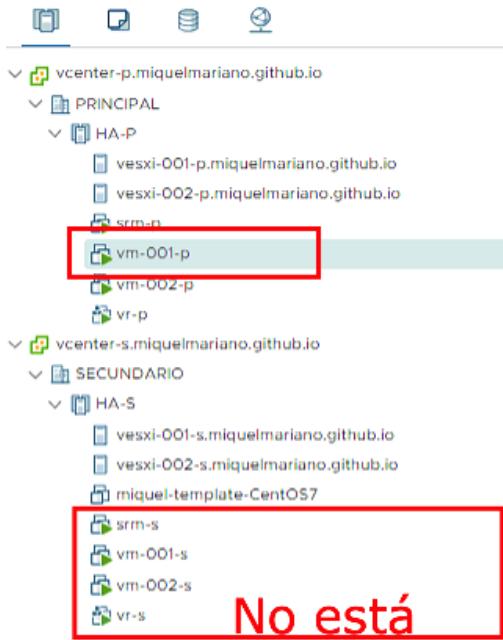
Tras finalizar la replicación inicial, nos aparecerá en la lista la VM con el estado e información correspondiente.

Máquina virtual	Estado	RPO	Destino	Servidor de replicación	Grupo de protección
vm-001-p	Correcto	1 hora	vcenter-s.miquelmariano.github.io	vr-s	
Discos configurados:	1 de 1	Último punto de sincronización de instancia:	17 ago. 2020 14:29:54		
Administrado por:	VR	Duración de la última sincronización:	2 minutos y 30 segundos		
Modo inactivo:	Deshabilitado	Tamaño de la última sincronización:	2.11 GB		
Compresión de red:	Habilitado	Tiempo de retraso:	11 segundos		
Cifrado:	Deshabilitado	RPO:	1 hora		
Almacén de datos:	S-SRM-Datastore01	Momentos específicos:	Deshabilitado		
Directiva de almacenamiento:	Valor predeterminado de almacenamiento de datos	Uso de disco de réplica:	1 KB		

Es probable que, llegados a este punto, os entre la curiosidad y os de por mirar en el inventario del vCenter.

En el inventario, veréis que está correctamente la VM que hemos replicado, pero no aparece en el sitio secundario su réplica.

Este comportamiento es normal, ya que sólo hemos hecho la mitad de la configuración. Tenemos ya los datos replicados en el datastore de destino con vSphere Replication, pero nos falta la parte de automatización que haremos con Site Recovery Manager y los Grupos de Protección.



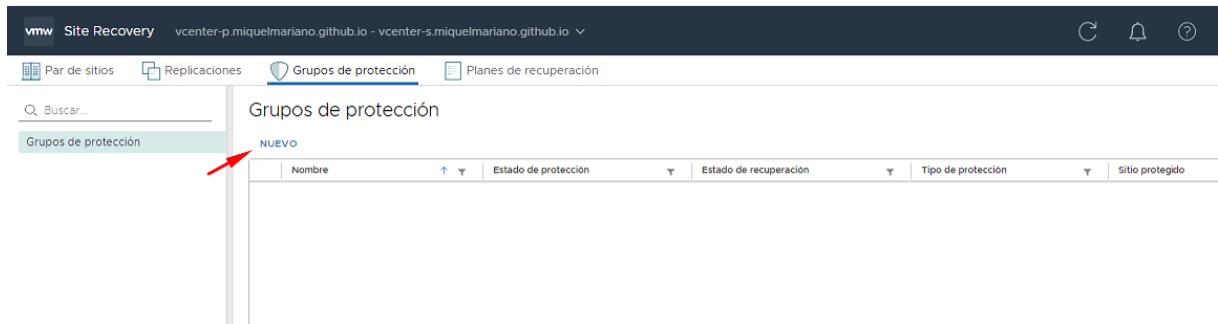
Vemos que en el datastore del sitio secundario sí está la información de la VM replicada.

Nombre	Tamaño	Modificado	Tip
Hvrcfg.GID-5f2319-7804-476c-87b3-27797d744702.490.vmx72	0,05 KB	17/08/2020 14:32:23	Ar
Hvrcfg.GID-5f2319-7804-476c-87b3-27797d744702.1st	3,56 KB	17/08/2020 14:32:24	Ar
Hvrcfg.GID-5f2319-7804-476c-87b3-27797d744702.490.vmsx71	6,11 KB	17/08/2020 14:32:23	Ar
Hvrcfg.GID-5f2319-7804-476c-87b3-27797d744702.490.vmem73	8,48 KB	17/08/2020 14:32:23	Ar
Hordik.RDID-59dd0eds-l0b5-4e9-ec37-e4ec4d9b43f50768300810494933.vmdk	1024 KB	17/08/2020 14:32:23	Di
vm-001-p_2.vmdk	2,241,536 KB	17/08/2020 14:32:23	Di

GRUPO DE PROTECCIÓN

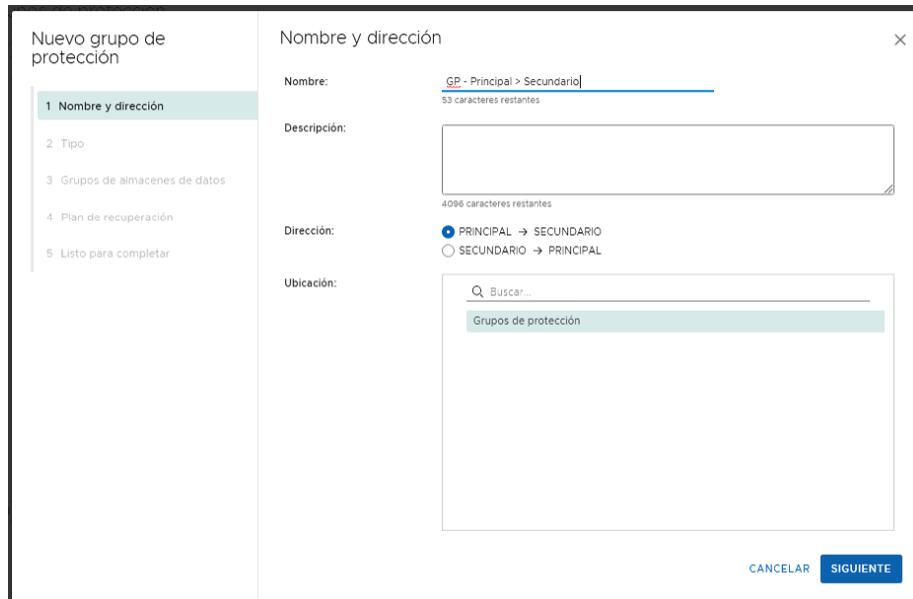
Un grupo de protección no es más que un conjunto de VMs que están siendo protegidas por SRM mediante algún método de replicación.

Para crear uno de ellos, nos dirigiremos a la pestaña correspondiente > Nuevo



The screenshot shows the VMware Site Recovery interface with the URL `vcenter-p.miquelmariano.github.io - vcenter-s.miquelmariano.github.io`. The navigation bar includes 'Par de sitios', 'Replicaciones', 'Grupos de protección' (which is highlighted in blue), and 'Planes de recuperación'. Below the navigation is a search bar labeled 'Buscar...'. The main area is titled 'Grupos de protección' and contains a table header with columns: Nombre, Estado de protección, Estado de recuperación, Tipo de protección, and Sitio protegido. A red arrow points to the 'Nuevo' (New) button located at the top left of the table area.

Al ser una prueba de laboratorio, el nombre que le asigno es GP – Principal > Secundario, en referencia a la dirección de la replicación.

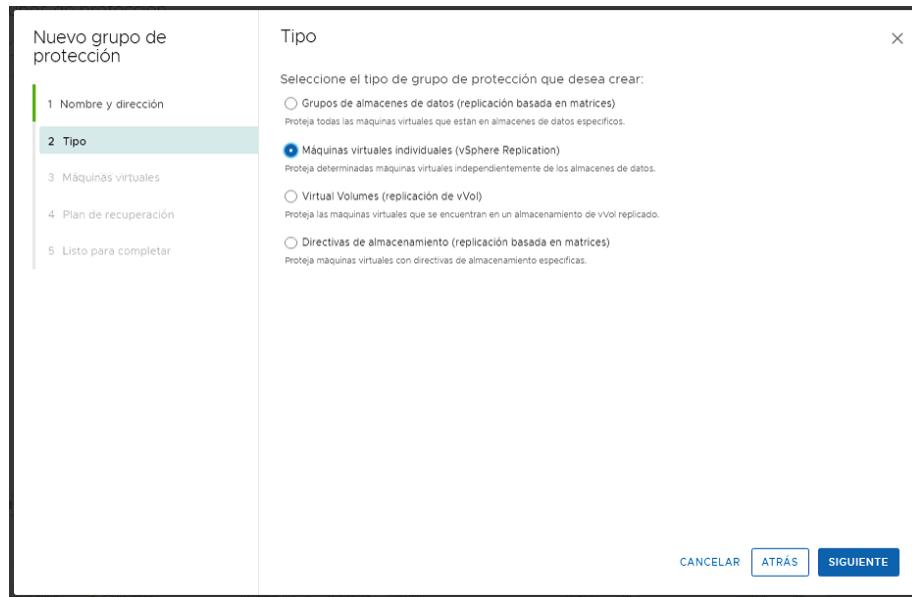


The screenshot shows the 'Nuevo grupo de protección' (New Protection Group) wizard, step 1: Nombre y dirección. On the left, a sidebar lists steps: 1. Nombre y dirección (highlighted in green), 2. Tipo, 3. Grupos de almacenes de datos, 4. Plan de recuperación, and 5. Listo para completar. The main panel shows the following fields:

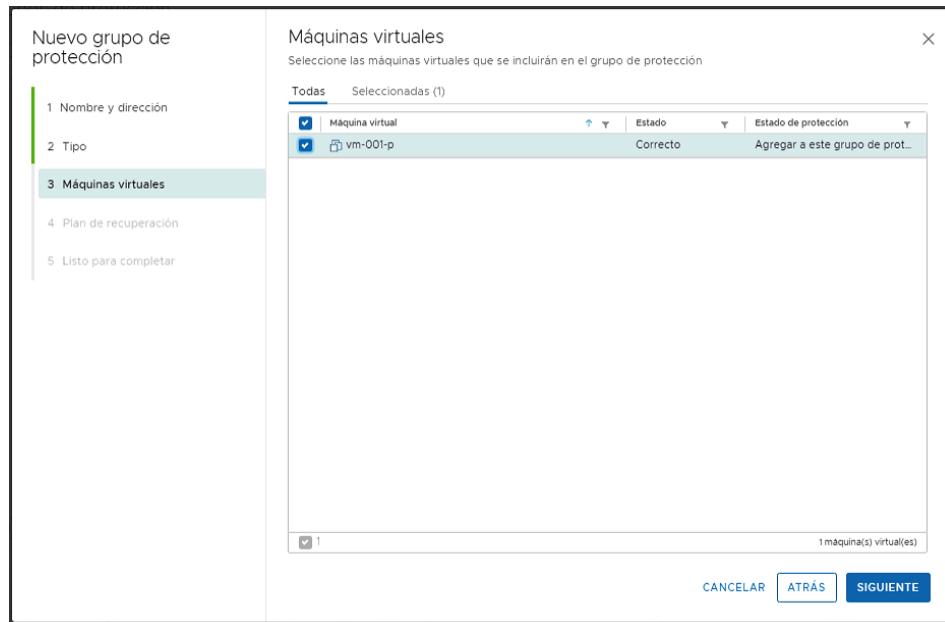
- Nombre:** GP - Principal > Secundario (53 caracteres restantes)
- Descripción:** (Empty text area, 4096 caracteres restantes)
- Dirección:** PRINCIPAL → SECUNDARIO SECUNDARIO → PRINCIPAL
- Ubicación:** Grupos de protección

At the bottom right are 'CANCELAR' and 'SIGUIENTE' buttons.

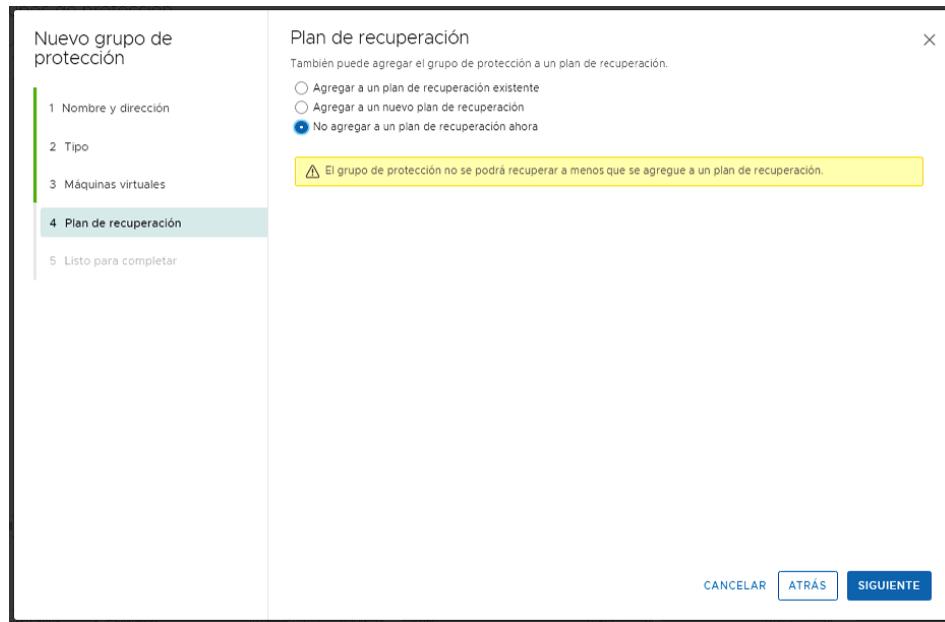
Seleccionaremos el tipo de replicación utilizado, en nuestro caso vSphere Replication.



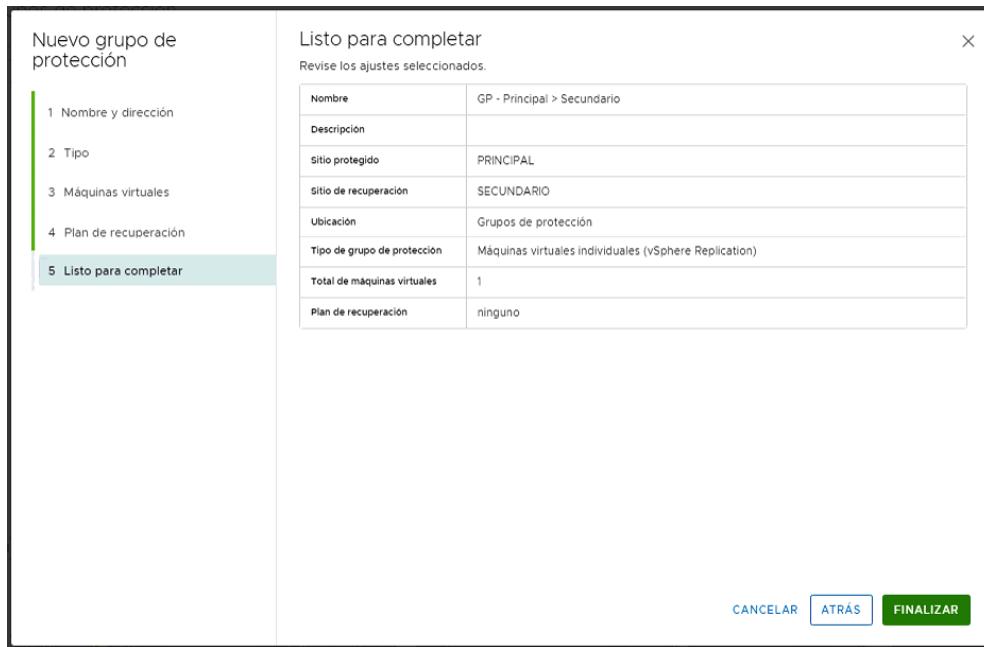
El propio wizard nos propone el inventario de VMs que actualmente se están replicando con vSphere Replication. En nuestro caso, solo hay una



No hemos hablado todavía de los planes de recuperación, así que dejaremos esta configuración para más adelante



Tras finalizar el wizard, nos aparecerá el resumen correspondiente que tendremos que finalizar



Y ya tendríamos nuestro primer Grupo de Protección creado y configurado.

Resumen

Grupo de protección: GP - Principal > Secundario
 Tipo de protección: Máquinas virtuales individuales (vSphere Replication)
 Sitio protegido: PRINCIPAL
 Sitio de recuperación: SECUNDARIO
 Descripción:

Estado:	Correcto
Máquinas virtuales:	1
Configuradas	1
Sin configurar	0

En este punto, si os fijáis en vuestro inventario de vCenter, sí que ya aparece la nueva VM.

Esta VM es sólo un “puntero” y utiliza el datastore de marcador de posición para hacer el inventariado de la VM.

En el siguiente apartado, veremos cómo podemos operar con esta “nueva” VM que acaba de aparecer en nuestro sitio secundario.

Resumen

No hay ninguna red asignada a esta máquina virtual.

Estado de energía	Apagado
Sistema operativo invitado	CentOS 7 (64-bit)
Cifrado	Sin cifrar
VMware Tools	No está en ejecución, no está instalado
Nombre de DNS	--
Direcciones IP	--

Capacidad y uso

CPU	0 Hz utilizado	1 CPU asignado
Memoria	0 B utilizado	2 GB asignado
Almacenamiento	0 B utilizado	asignado

Objetos relacionados

- Clúster: HA-S
- Host: vesxi-001-s.miquelmariano.github.io
- Almacenamiento: S-SRM-Posición

Hardware de máquina virtual

CPU	1 CPU, 0 MHz utilizados
Memoria	2 GB, 0 GB memoria activa
Compatibilidad	ESXi 6.5 y posterior (máquina virtual versión 13)

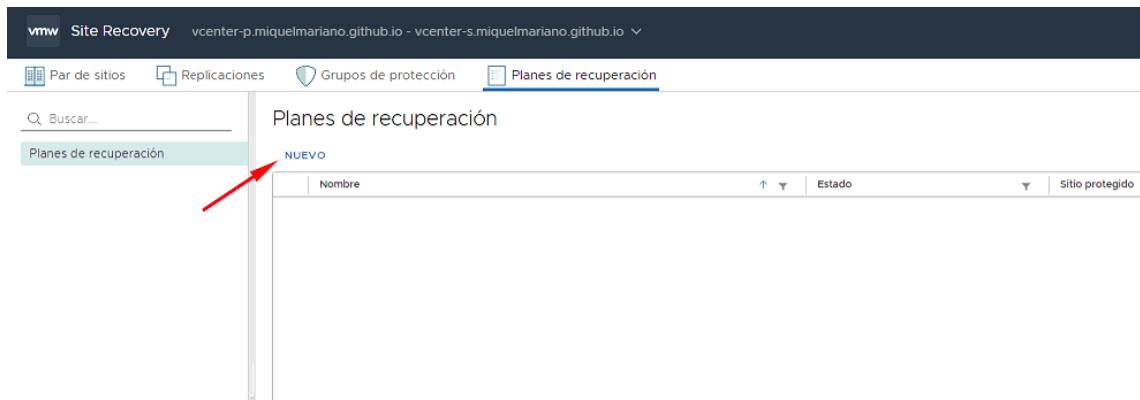
Instantáneas

No se creó ninguna instantánea

PLANES DE RECUPERACIÓN

Un plan de recuperación es un conjunto de acciones que realizará SRM para restaurar las VM del grupo de protección en el sitio secundario o de recuperación.

Para crear uno de ellos, en la pestaña “Planes de recuperación” > Nuevo



Seleccionaremos la dirección de esa recuperación y le asignaremos un nombre.

Crear plan de recuperación

1 Nombre y dirección

2 Grupos de protección

3 Redes de prueba

4 Listo para completar

Nombre y dirección

Nombre: PR - Principal > Secundario
53 caracteres restantes

Descripción:

Dirección:

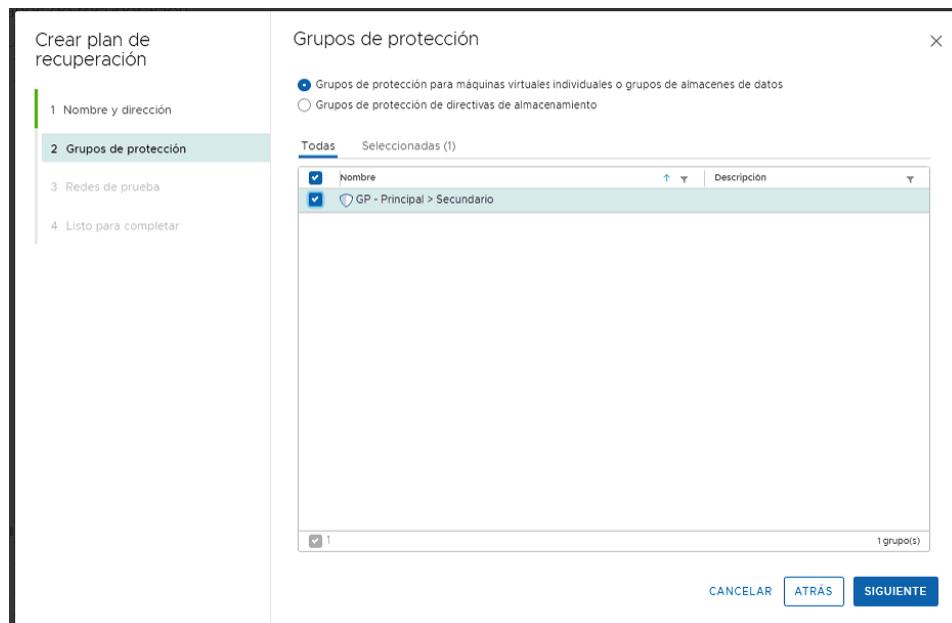
PRINCIPAL → SECUNDARIO
SECUNDARIO → PRINCIPAL

Ubicación:

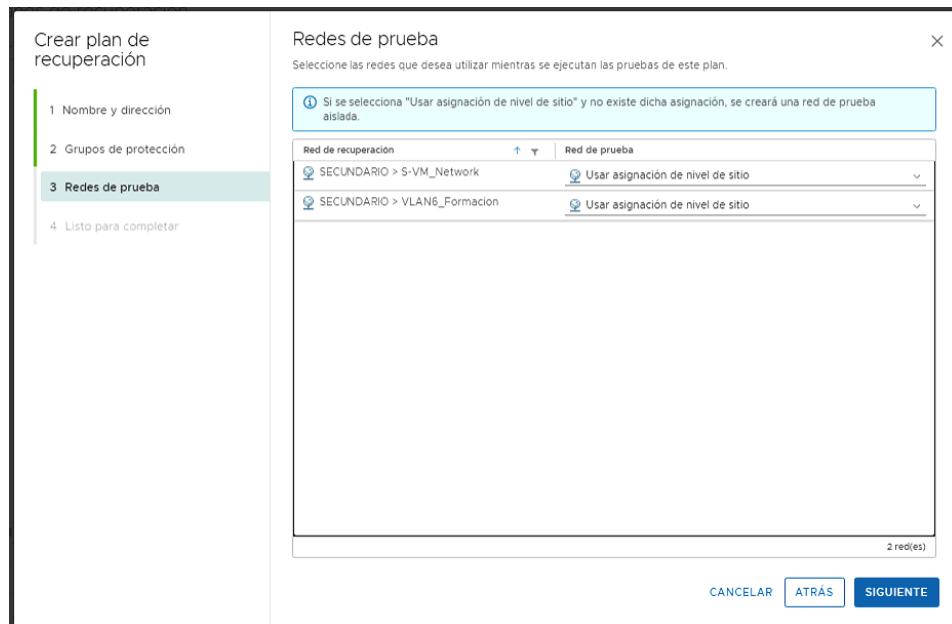
Planes de recuperación

CANCELAR SIGUIENTE

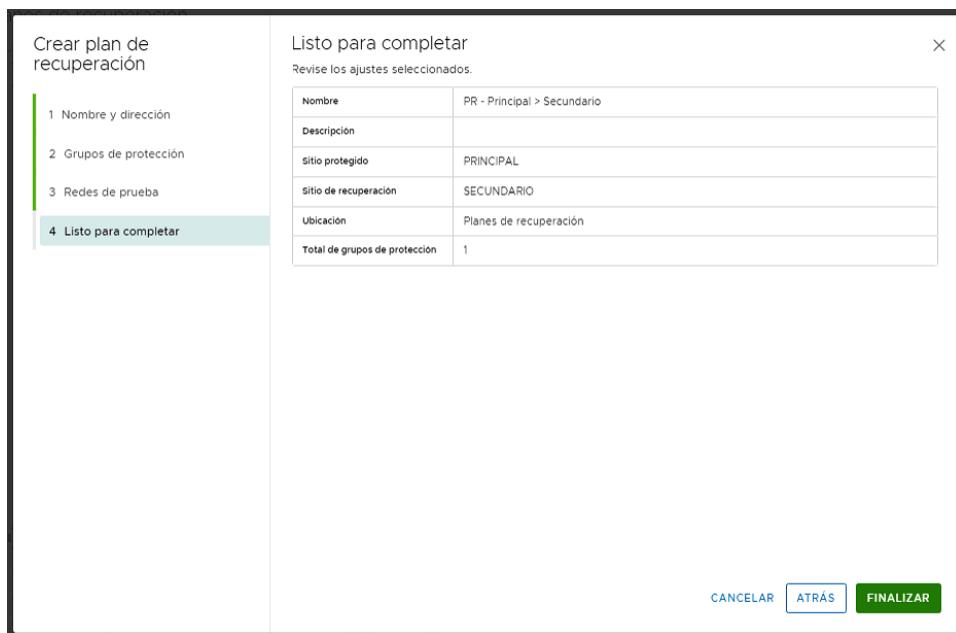
Nos aparecerán los Grupos de protección disponibles en el sentido que hemos configurado el plan, y debemos seleccionar los que deseemos. Pueden ser más de uno.



Podemos configurar en el propio plan las redes que se asociarán en el sitio secundario. Recordad que esta configuración previamente se ha definido en los “Resource Mappings”, por lo que lo podemos dejar por defecto.



Resumen de la configuración



Vista resumen del plan de recuperación que acabamos de crear.

PRUEBAS DE CONTINGENCIA

Una vez tengamos nuestro entorno SRM instalado y configurado, con sus correspondientes Grupos de protección y Planes de recuperación, estaremos en posición de empezar a “probar” o “ejecutar” nuestra instalación.

Cómo digo, los planes de recuperación, una vez configurados, tienen 2 opciones principales:

PROBAR

Esta acción montará en el sitio de recuperación una red aislada y recuperará la VM replicada (Acordaros que, aunque a la VM la veamos en el inventario, realmente es solo el “marcador de posición”). Una vez la VM esté lista, la arrancará, y podremos comprobar que es completamente funcional (sin red, evidentemente) en el sitio de recuperación.

vmm Site Recovery vcenter-p.miquelmariano.github.io - vcenter-s.miquelmariano.github.io

Par de sitios Replicaciones Grupos de protección Planes de recuperación

Buscar...

Planes de recuperación PR - Principal > Secundario

PR - Principal > Secundario

EDITAR MOVER ELIMINAR PROBAR LIMPIEZA EJECUTAR

Resumen Pasos de recuperación Problemas Historial Permisos Grupos de protección Máquinas virtuales

Plan de recuperación: PR - Principal > Secundario

Sitio protegido: PRINCIPAL

Sitio de recuperación: SECUNDARIO

Descripción:

Estado del plan: Listo

Este plan está listo para la prueba o la recuperación

Historial reciente

Limpieza	jueves, 20 de agosto de 2020 18:45:16	✓ Correcto
Prueba	jueves, 20 de agosto de 2020 18:45:16	✓ Correcto

Estado de la máquina virtual

Listo para la recuperación:	Verde
En curso:	Verde
Correcto:	Verde
Advertencia:	Verde
Error:	Verde
Incompleto:	Verde

The screenshot shows the VMware Site Recovery interface. The top navigation bar includes 'Par de sitios' (Pair of sites), 'Replicaciones' (Replications), 'Grupos de protección' (Protection groups), and 'Planes de recuperación' (Recovery plans). The current view is 'Planes de recuperación'.

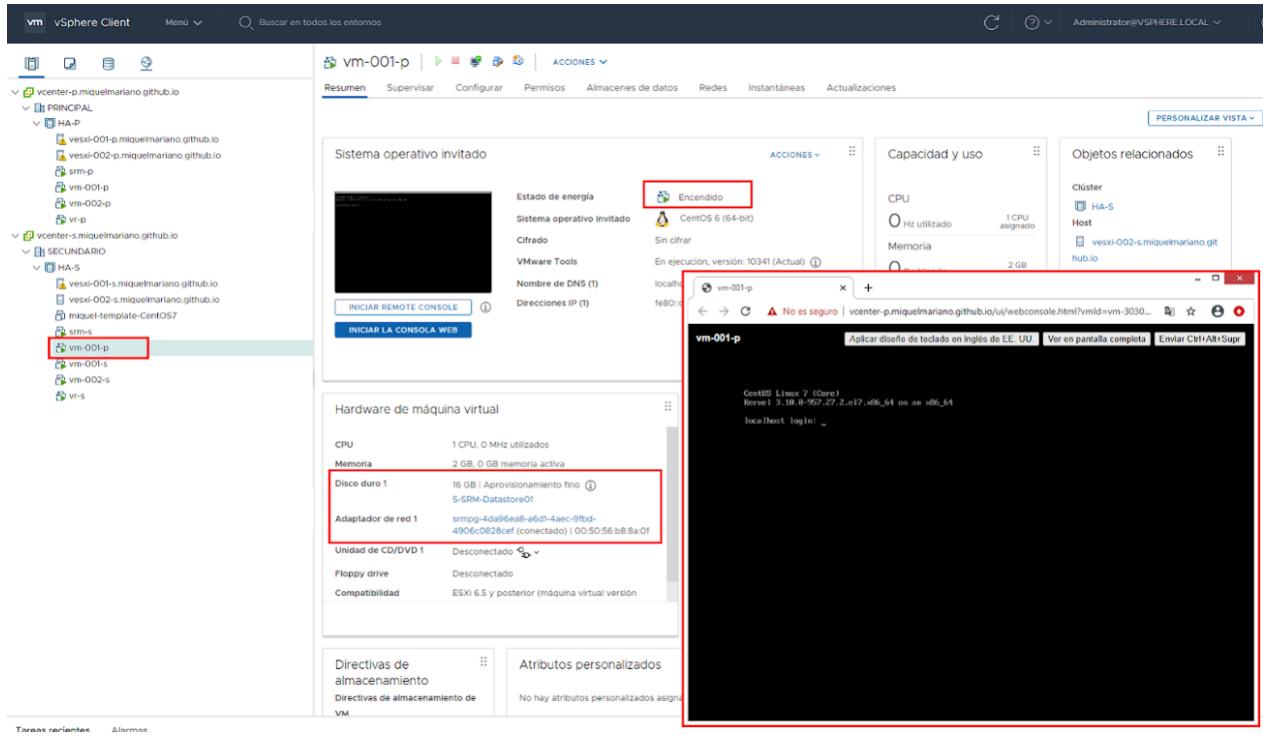
The main content area displays a recovery plan titled 'PR - Principal > Secundario'. It shows the following details:

- Plan de recuperación:** PR - Principal > Secundario
- Sitio protegido:** PRINCIPAL
- Sitio de recuperación:** SECUNDARIO
- Descripción:** [Empty]

A yellow warning box indicates that the test was completed successfully: **Prueba completada**.

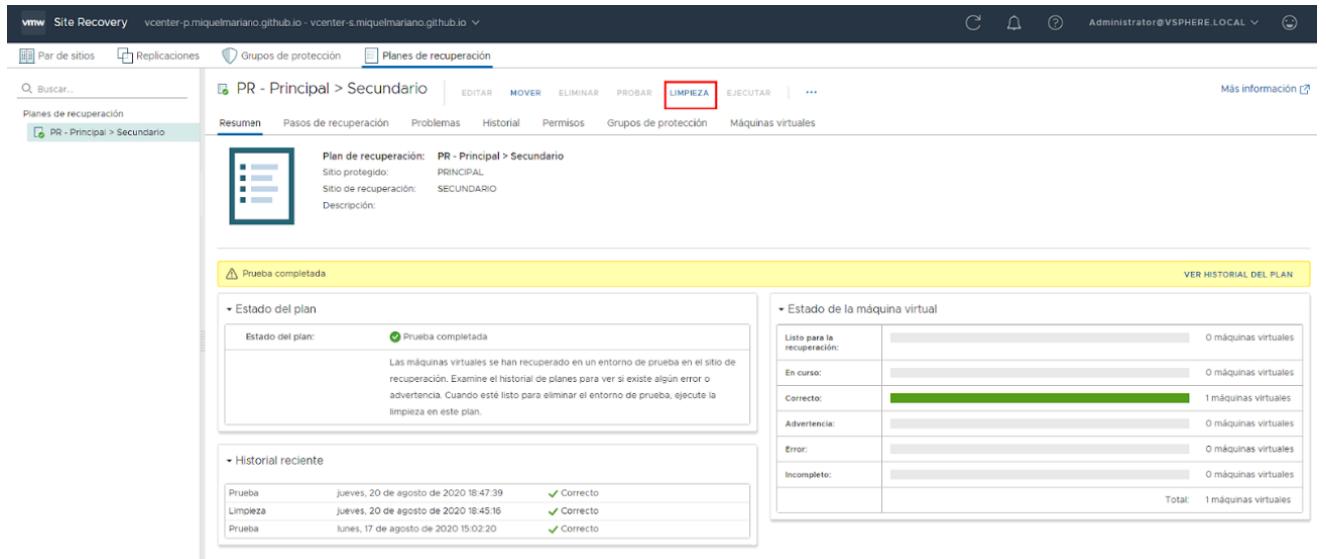
The interface includes sections for the **Estado del plan** (Plan status) and **Estado de la máquina virtual** (Virtual machine status). The **Historial reciente** (Recent history) section lists recent operations: Prueba (jueves, 20 de agosto de 2020 18:47:39), Limpieza (jueves, 20 de agosto de 2020 18:45:16), and another Prueba (lunes, 17 de agosto de 2020 15:02:20), all marked as Correcto (Correct).

Una vez finalizada la prueba, vemos que la VM está correctamente arrancada en el sitio de recuperación, almacenada en el Datastore que previamente hemos configurado en los “Resource Mappings” y en una red que el propio SRM ha creado completamente aislada.



Limpieza

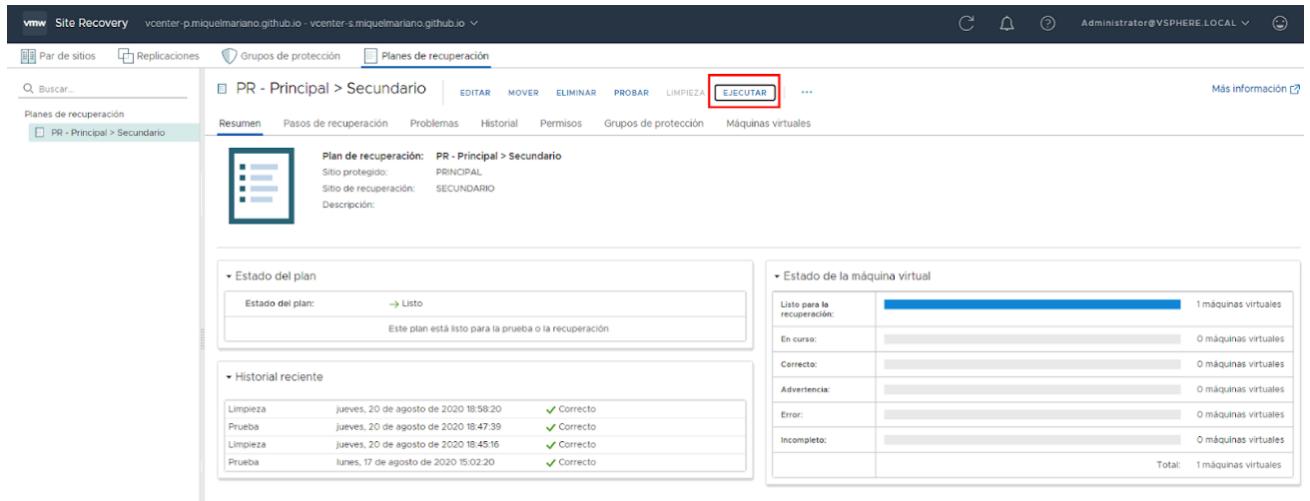
Esta acción devolverá SRM a su situación original. Apagará la VM y la dejará inventariada con el marcador de posición en su correspondiente datastore



EJECUTAR

Esta acción sí que hay que tomársela más seriamente. Lo que estamos a punto de realizar es un balanceo hacia el sitio secundario o de recuperación.

Esta acción implica que la VM original se va a apagar y va a arrancar en el sitio secundario.

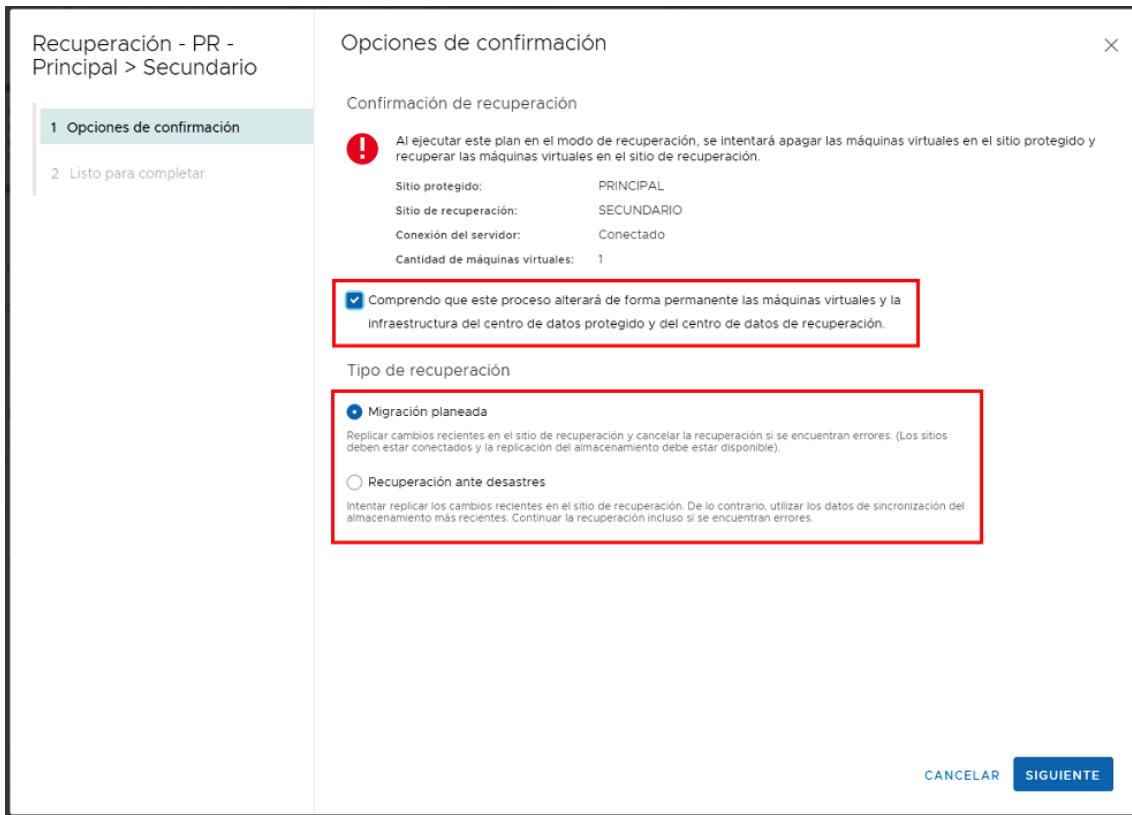


El propio wizard ya nos avisa de lo que va a pasar, y habrá que marcar el check de que comprendemos el proceso que estamos a punto de ejecutar.

También tendremos que seleccionar el tipo de recuperación:

Migración planeada: Nos sirve para balancear ordenadamente la carga entre un CPD y otro. Sincronizará los cambios antes de empezar con el balanceo.

Recuperación ante desastres: Esta opción la seleccionaremos en caso de un DR real. Si se nos ha caído el CPD principal, necesitaremos arrancar cuanto antes las VMs en el CPD secundario.



En la pestaña *Pasos de recuperación* podremos ver todo el proceso en detalle:

Paso de recuperación	Estado	Paso iniciado	Paso completado
> 1. Sincronización previa de almacenamiento	✓ Correcto	Jueves, 20 de agosto de 2020 19:04:31	Jueves, 20 de agosto de 2020 19:04:31
> 2. Apagar las máquinas virtuales en el sitio protegido	■ En ejecución	Jueves, 20 de agosto de 2020 19:04:31	0%
> 3. Reanudar las máquinas virtuales suspendidas por la recuperación anterior			
> 4. Restaurar los hosts del sitio de recuperación que están en modo de espera			
> 5. Restaurar los hosts del sitio protegido que están en modo de espera			
> 6. Preparar las máquinas virtuales del sitio protegido para la migración			
> 7. Sincronizar almacenamiento			
> 8. Suspender las máquinas virtuales no críticas en el sitio de recuperación			
> 9. Cambiar el almacenamiento del sitio de recuperación para permitir la escritura			
10. Encender las máquinas virtuales de prioridad 1			
11. Encender las máquinas virtuales de prioridad 2			
12. Encender las máquinas virtuales de prioridad 3			
13. Encender las máquinas virtuales de prioridad 4			
14. Encender las máquinas virtuales de prioridad 5			

Una vez finalizado el proceso, veremos que nuestra VM está apagada en el CPD principal y se ha encendido en el CPD secundario.

En esta ocasión, sí está conectada a una red real, y es 100% funcional y lista para dar servicio.

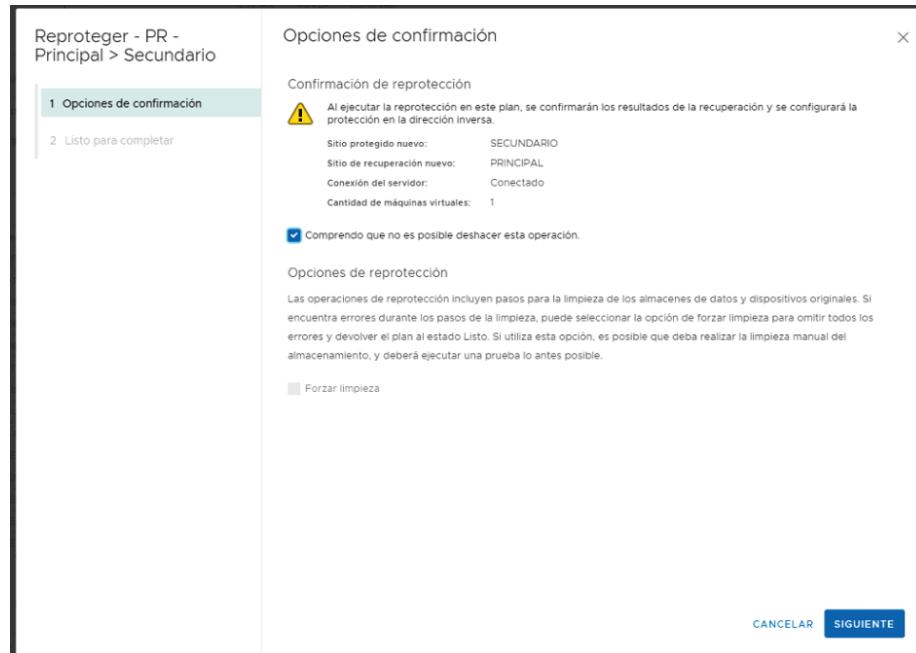
The screenshot shows the vSphere Site Recovery interface. At the top, it displays 'Site Recovery' and 'vcenter-p.miquelmariano.github.io - vcenter-s.miquelmariano.github.io'. The main pane shows a 'PR - Principal > Secundario' recovery plan. A red box highlights the 'Estado del plan:' section, which says 'Recuperación completada'. Below this, a detailed log of 14 steps is shown, all marked as 'Correcto' (Correct) and completed successfully on 'jueves, 20 de agosto de 2020 19:04:53'. The log includes steps like 'Sincronización previa de almacenamiento', 'Apagar las máquinas virtuales en el sitio protegido', and 'Encender las máquinas virtuales de prioridad 1'.

The screenshot shows the vSphere Client interface. On the left, the inventory tree shows two hosts: 'vcenter-p.miquelmariano.github.io' and 'vcenter-s.miquelmariano.github.io'. Under each host, there are 'PRINCIPAL' and 'SECUNDARIO' sections. In the 'PRINCIPAL' section of the first host, a red box highlights 'vm-001-p'. In the 'SECUNDARIO' section of the second host, another red box highlights 'vm-001-s'. The main pane displays details for 'vm-001-p'. It shows the VM is powered on ('Encendido'), running 'CentOS 6 (64-bit)', and has 'VMware Tools' installed ('En ejecución, versión: 10341 (Actual)'). It also lists its IP addresses (192.168.6.216, fe80::cfc992:2b5e:8aef), and its hardware configuration: 1 CPU, 2 GB memory, and a 16 GB disk. A terminal window at the bottom right shows a ping command to 192.168.6.216.

Reproteger

Llegados a este punto, tendremos nuestra VM dando servicio en el CPD secundario, pero ya no estará protegida por SRM. Para ello, tendremos que indicar al sistema que tiene que empezar a replicar en sentido inverso, para ello, utilizaremos la opción Reproteger

The screenshot shows the VMware Site Recovery interface. In the top navigation bar, there are tabs for 'Par de sitios' (Pair of sites), 'Replicaciones' (Replications), 'Grupos de protección' (Protection groups), and 'Planes de recuperación' (Recovery plans). The current view is under 'Planes de recuperación'. A specific plan named 'PR - Principal > Secundario' is selected. The main pane shows a summary of the plan, including its status as 'Recuperación completada' (Recovery completed). Below this, a detailed list of recovery steps is provided, each with its name, state ('Estado'), start time ('Paso iniciado'), and completion time ('Paso completado'). The steps include tasks like 'Sincronización previa de almacenamiento', 'Apagar las máquinas virtuales en el sitio protegido', and 'Restaurar los hosts del sitio de recuperación que están en modo de espera'. The 'REPROTEGER' button is highlighted with a red box at the top of the summary section.



Una vez finalizada la reprotectación, si nos fijamos, veremos que nos ha quedado nuestra VM productiva en el CPD secundario y la réplica en el Principal, es decir, a la inversa de cómo estaba inicialmente.

Para “arreglar” esta situación que la productiva vuelva a estar en el principal, no nos quedará otra que pasar de nuevo por el proceso “Ejecutar” y después “Reproteger” de nuevo.

DOCUMENTACIÓN OFICIAL

Todo, o casi todo, lo expuesto en este capítulo ha sido extraído de la documentación oficial de SRM versión 8.3 y de mi propia experiencia profesional.

<https://docs.VMware.com/es/Site-Recovery-Manager/index.html>

<https://docs.VMware.com/es/vSphere-Replication/index.html>

Espero que este capítulo sea de vuestro interés.

¡Gracias por leernos!



[PRUEBA GRATUITA](#)

Backup y recuperación n.º 1

VMware y Hyper-V

- Solución independiente de hipervisor para todas sus máquinas virtuales
- Soporte para VMware, Hyper-V y Nutanix AHV
- Recuperación rápida y confiable de VM completas a elementos individuales

vmware



[LEER INFORME](#)



Capítulo 8

VEEAM BACKUP & REPLICATION: NOVEDADES



Xavi Genestos

@sysadmit

VEEAM BACKUP & REPLICATION: NOVEDADES

INTRODUCCIÓN

Veeam Software lanzó en el año 2008 la versión 1.0 de su producto Veeam Backup, la idea era ofrecer copia de seguridad de máquinas virtuales a nivel de hipervisor para entornos VMware ESX.

El producto se diferenciaba de los sistemas de Backup tradicionales que funcionaban con agente y cinta, Veeam apostó por el Backup a disco y sin agente, ya que hacía el Backup contactando con las APIs del hipervisor y eso fue una revolución: Backups más rápidos y fiables.

De hecho, Veeam Software no incorporó la copia a cinta hasta la versión 7.0 en el año 2013.

A ese producto del año 2008 se le han añadido multitud de funcionalidades, y hoy en día, es un estándar que se utiliza en muchas empresas.

En este apartado del libro veremos la instalación básica del producto **Veeam Backup & Replication**, para aquellos que no lo habéis probado nunca, y por otro lado tenéis las novedades que incorpora la versión 10 explicadas de forma práctica.

INSTALACIÓN DE VEEAM BACKUP & REPLICATION

REQUISITOS

Antes de proceder a la instalación del producto, deberíamos repasar los requisitos.

Estos requisitos varían a cada versión y los podemos encontrar en la web de Veeam:

https://helpcenter.veeam.com/docs/backup/vsphere/system_requirements.html?ver=100

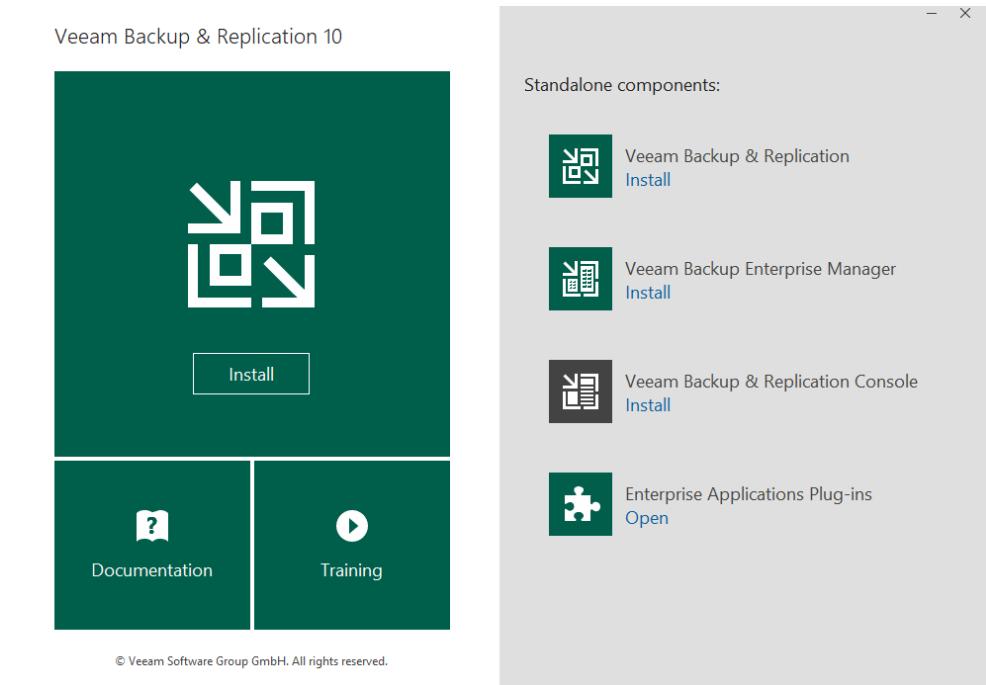
Si repasáis el enlace anterior, veréis que existen una serie de requisitos distintos para cada componente de Veeam.

A nivel de resumen, en la versión 10, si vamos a realizar una instalación de todo en uno, es decir, instalar Veeam Backup & replication en un solo equipo, necesitaremos:

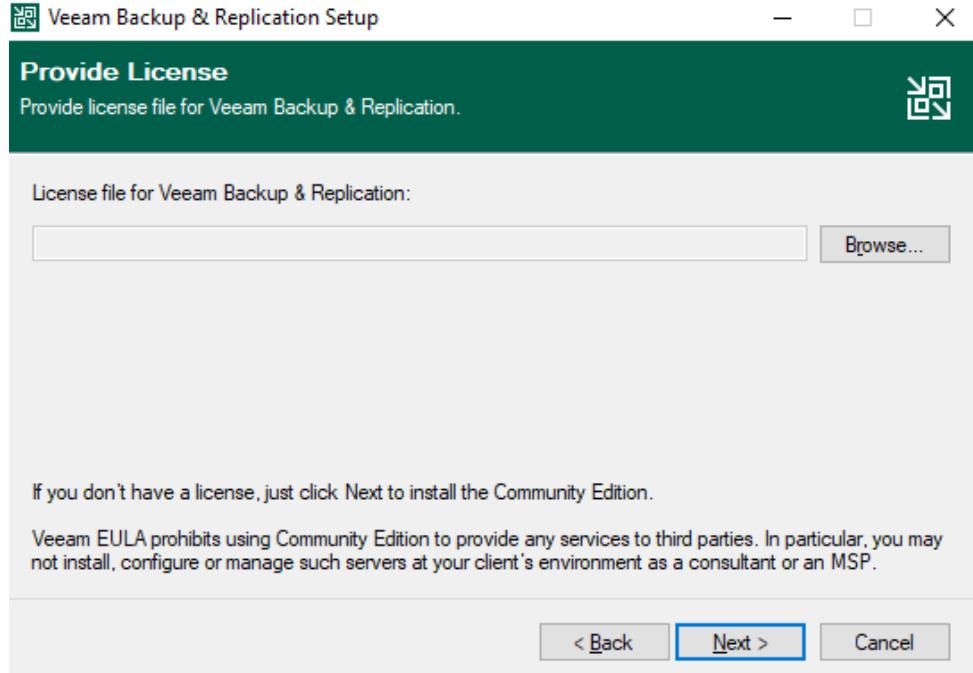
- **CPU:** x86-64 (Se recomiendan mínimo 4 cores).
- **Sistema operativo:** Windows Server, desde 2008 R2 a 2019, o cliente desde Windows 7 a Windows 10.
- **Software:** Microsoft .NET Framework 4.7.2/ Microsoft Windows Installer 4.5 / Microsoft SQL Server Management Objects / Microsoft SQL Server System CLR Types / Microsoft Report Viewer Redistributable 2015 / Microsoft Universal C Runtime
Muchos de estos componentes los instala el instalador de Veeam Backup.
- **SQL Server:** Desde Microsoft SQL Server 2008 a Microsoft SQL Server 2019
Si no disponemos de SQLServer, el instalador nos instalará un SQLServer Express.
No recomendado para ambientes productivos por la limitación de crecimiento de la base de datos hasta 10 GB.

INSTALACIÓN GUI – PASO A PASO

De los componentes que nos ofrece el asistente para instalar, procederemos a instalar: “Veeam Backup & Replication”

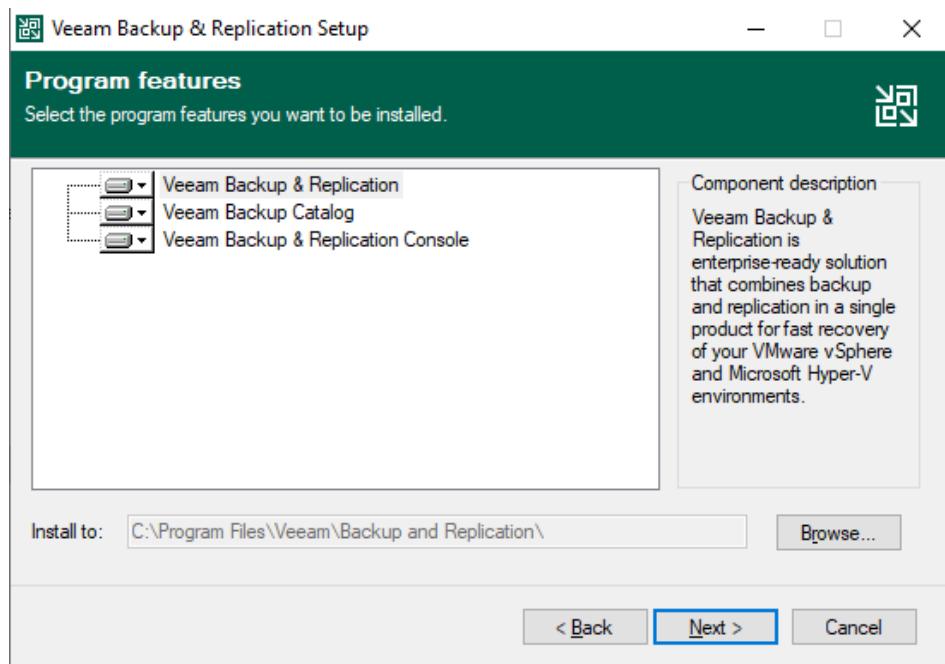


Si no especificamos una licencia, se instalará la edición: “Community”.

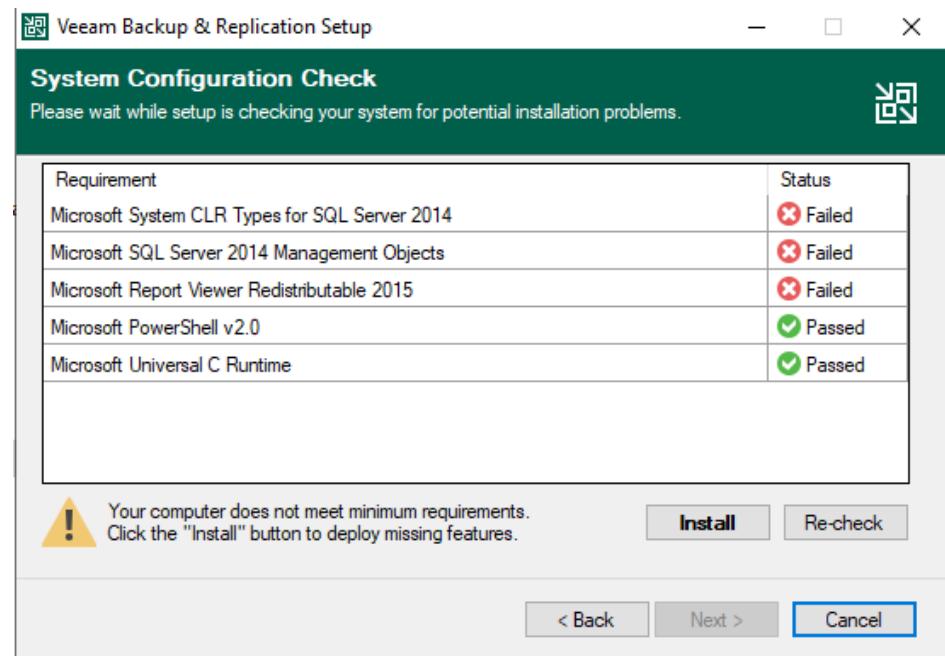


Seleccionamos los componentes a instalar.

Por defecto, se instalarán todos.



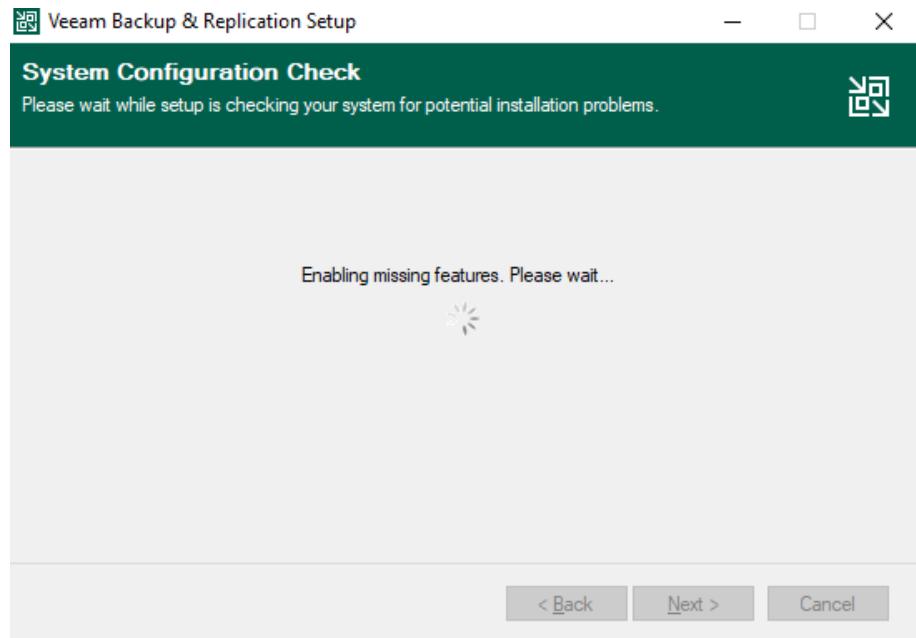
El asistente de instalación detecta los componentes instalados en el sistema y cuales le faltan.



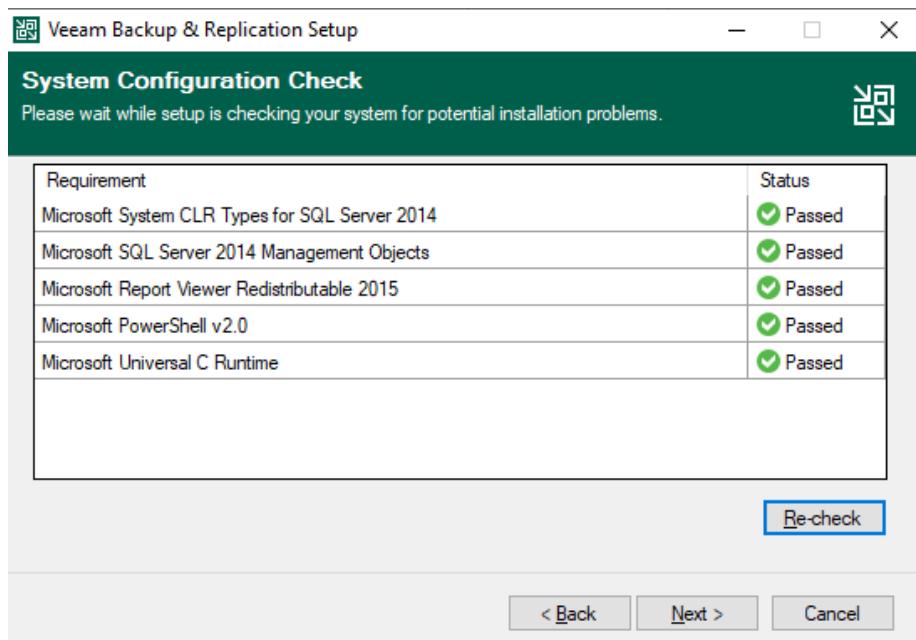
En esta captura de pantalla, podemos ver los componentes que faltan en un Windows Server 2019 de forma predeterminada.

Si pulsamos sobre el botón: "Install" se instalarán de forma automática.

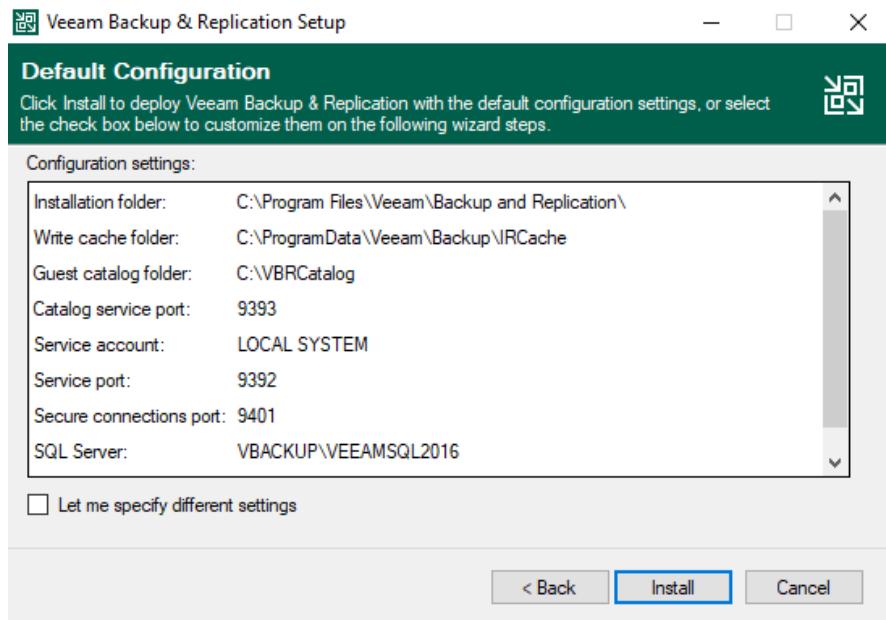
Este es el aspecto del asistente después de pulsar el botón "Install" y proceder a instalar todos los componentes que faltan.



Este es el aspecto después de instalar todos los componentes.

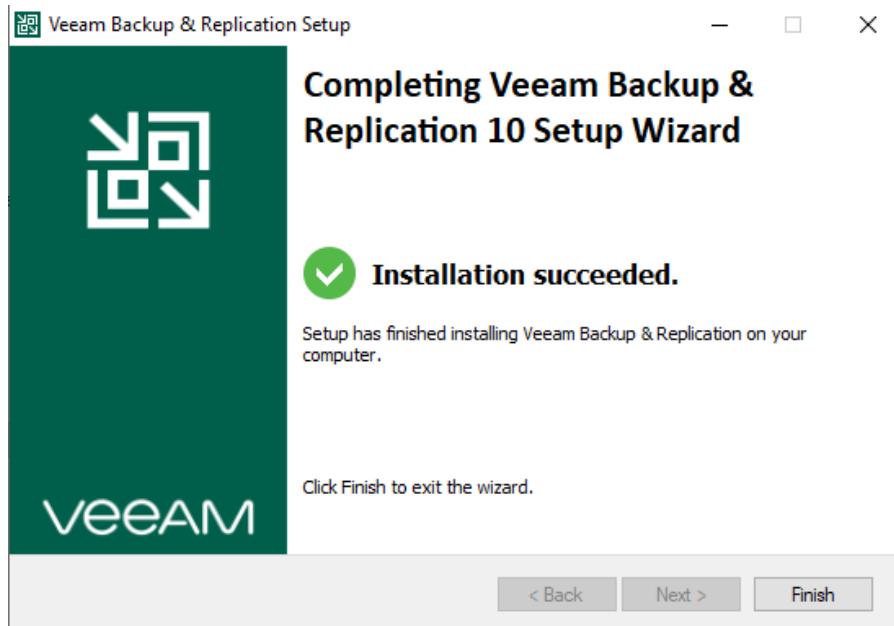


Vemos como se activa el botón de: "Next" para proseguir.



Si queremos cambiar algún parámetro de configuración, podemos marcar la opción: “**Let me specify different settings**”.

Si pulsamos directamente sobre el botón: “Install” procederemos a la instalación.

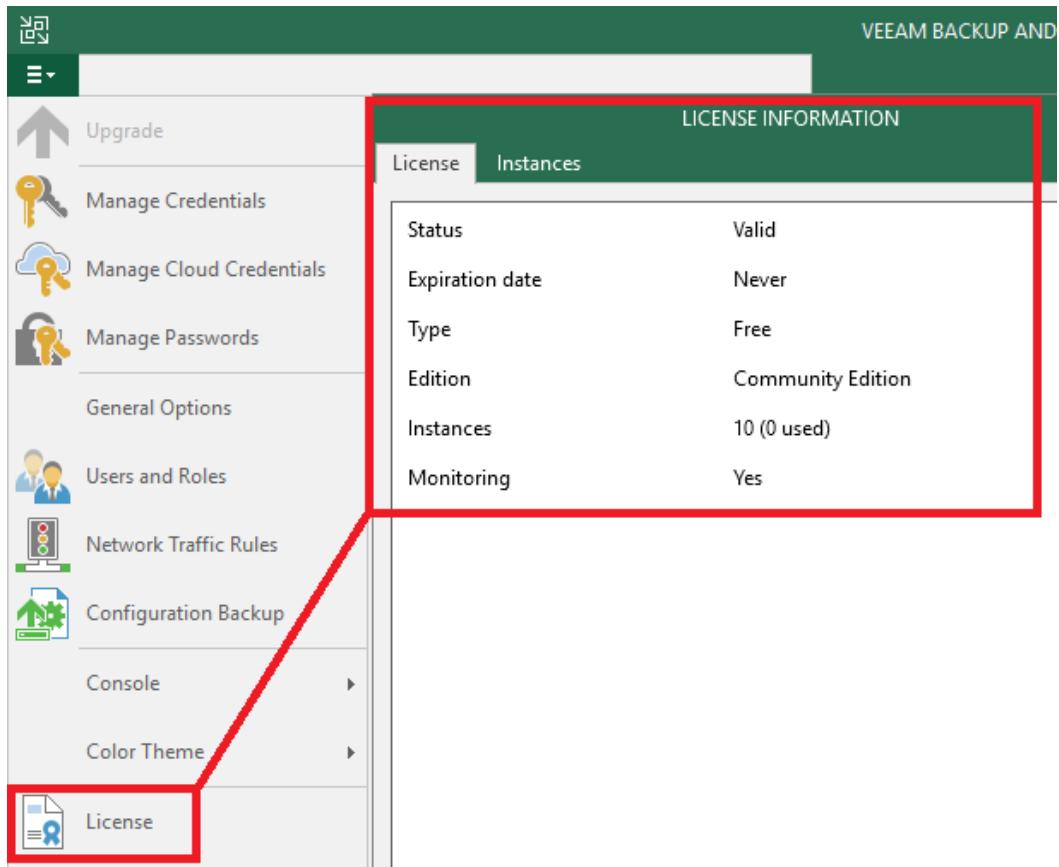


Instalación finalizada.

CONFIGURACIÓN INICIAL BÁSICA

LICENCIA

Una vez instalado el producto, dispondremos de la licencia gratuita “Community Edition”.



Si disponemos de una licencia comprada, deberíamos instalarla antes de empezar a utilizar el producto.

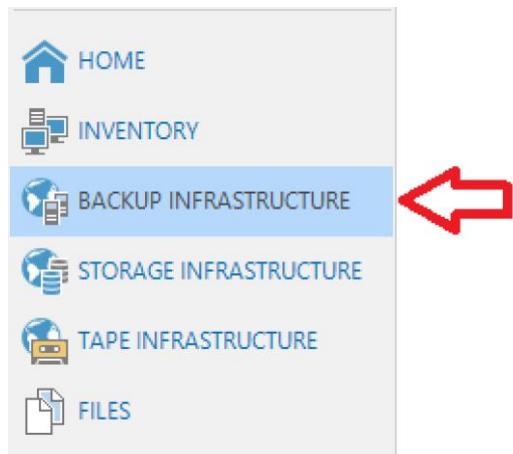
Disponéis de más información sobre la edición “Community” en este enlace:

[Veeam Backup: Community Edition \(SYSADMIT.com\)](http://SYSADMIT.com)

ORIGEN Y DESTINO

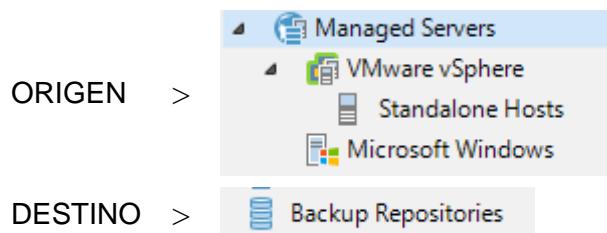
Para la puesta en marcha del producto será necesario configurar el origen y el destino.

Para configurar el origen y destino es muy sencillo y debemos hacerlo en la vista: "Backup infrastructure"



Una vez situados en esta vista configuraremos el origen dentro de "Managed Servers" y elegiremos "VMware vSphere" para hosts ESXi o Virtual Center y "Microsoft Windows" para el respaldo de equipos físicos o hosts de Hyper-V.

Para el destino, es decir, la ubicación donde se almacenarán las copias de seguridad, nos situaremos en "Backup Repositories".



NOVEDADES DE LA VERSIÓN 10

VEEAM BACKUP & REPLICATION: NAS BACKUP

Veeam NAS Backup: Una de las funcionalidades incluida en la versión 10 de Veeam Backup & Replication es la posibilidad de poder respaldar los ficheros ubicados en un NAS (Network Attached Storage).

La idea de esta funcionalidad es poder respaldar los ficheros que residen en un NAS sin instalar ningún agente en él para poder respaldarlos, dejando de depender del protocolo NDMP, poniendo foco en los protocolos SMB (CIFS), NSF, así como respaldo de File Servers Windows y Linux, logrando mejor performance en Backup y recuperación.

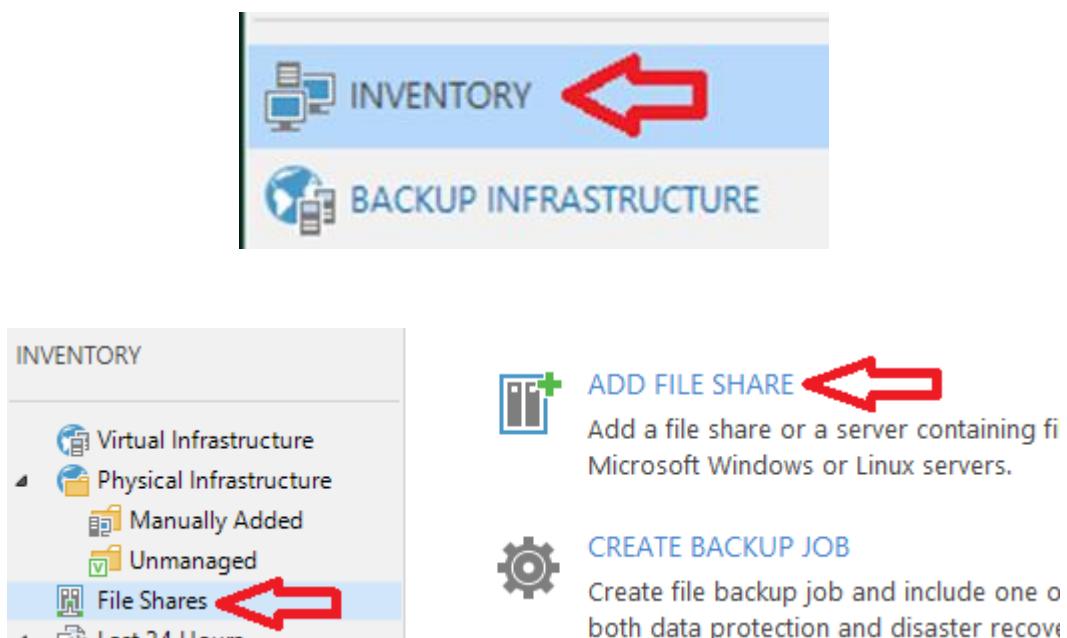
Veamos cómo funciona y cómo se configura.

Ubicación:

Para definir el recurso compartido a respaldar debemos situarnos a la vista de: "Inventory" y la opción de "File Shares" y "Add file share".

No encontraremos la opción de "File Shares" hasta la versión 10 de Veeam Backup & Replication.

Encontraremos la vista de "Inventory" en la parte inferior izquierda.



PROCEDIMIENTO

La primera opción: "File Server" es si el NAS es un servidor Windows o Linux.

La segunda opción es la de "NFS Share" por si queremos acceder al NAS utilizando el protocolo NFS.

Finalmente tenemos la opción de "SMB Share" en el caso de acceder al NAS utilizando el protocolo SMB.



NFS share

Adds an NFS file share hosted on a NAS device. Suppo



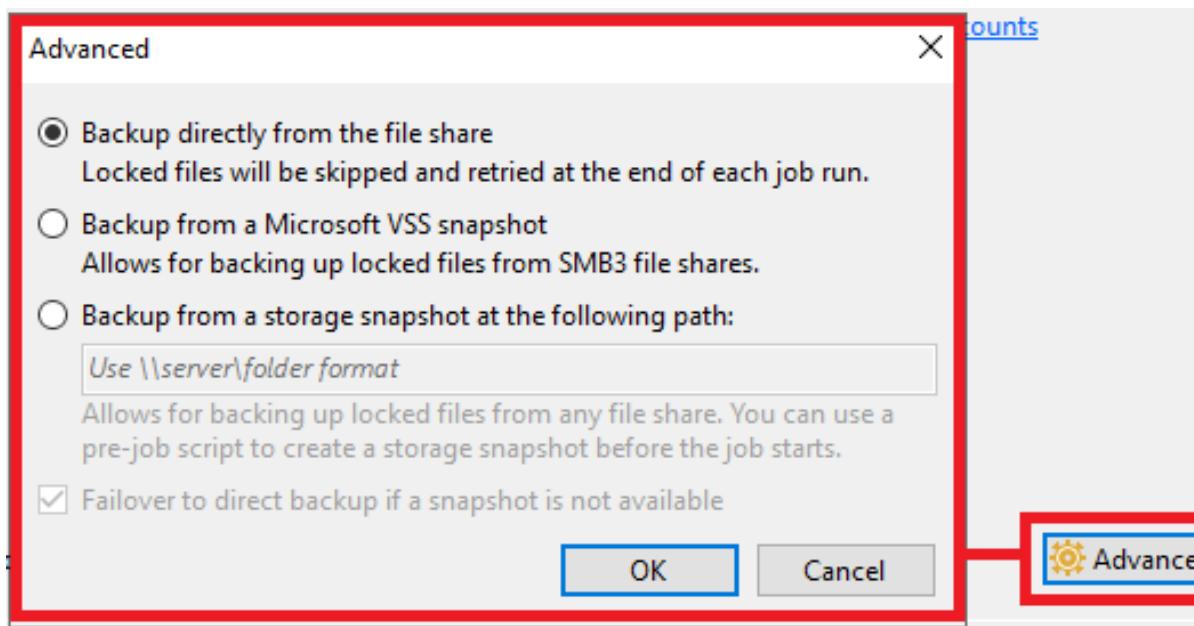
SMB share

Adds an SMB (CIFS) file share hosted on a NAS device. protocol version 3.0 or later.

A continuación, indicamos la ruta UNC (Universal Naming Convention) del recurso compartido y las credenciales de acceso:

SMB File Share	Shared folder: \\EX1\INSTALL
Processing	Use \\server\folder format
Apply	<input checked="" type="checkbox"/> This share requires access credential D1\Xavi (D1\Xavi, Last edited:
Summary	

en el botón "Advanced", disponemos de varias opciones:



Aquí podemos ver como la opción por defecto es la de: "Backup directly from the file share".

Esto significará que los ficheros bloqueados no serán copiados.

Después tenemos la opción: "Backup from a Microsoft VSS snapshot" que permite utilizar VSS (Volume Shadow Copy Service) para poder respaldar ficheros bloqueados.

Aquí tendríamos que verificar si el NAS del cual queremos respaldar los ficheros permite SMB 3.0.

A continuación, definiremos:

File proxy: Define el servidor de Veeam que queremos que se utilice para el procesamiento, esta es la opción de: "File proxy".

Caché repository: Repositorio a utilizar como caché de ficheros respaldados.

Backup I/O control: Indica el número de procesos que se lanzarán para respaldar los ficheros. Contra más procesos, más rápido irá al Backup, pero más sufrirá el NAS a respaldar. El parámetro por defecto es el nivel: "Medio".

New File Share

Processing
Define the list of file proxies to be used for this file share processing and performance.

SMB File Share

Processing

Apply

Summary

File proxy:
All proxies

Cache repository:
REPO (Created by VBACKUP\Administrator at 2)

Backup I/O control:

Lower impact
Controls how aggressively backup jobs can fetch pacing read requests of a single thread, while fas

A continuación, debemos configurar el job de respaldo:

Nos situaremos en la vista de: "Inventory" y la opción de "File Shares" y "Create backup job".

INVENTORY

- Virtual Infrastructure
- Physical Infrastructure
 - Manually Added
 - Unmanaged
- File Shares
- SMB Shares

ADD FILE SHARE

Add a file share or a server containing files Microsoft Windows or Linux servers.

CREATE BACKUP JOB

Create file backup job and include one or more file shares in it for both data protection and disaster recovery.

Definimos un nombre para el job:

New File Backup Job

The screenshot shows the 'Name' step of the 'New File Backup Job' wizard. On the left, there is a sidebar with three tabs: 'Name' (selected), 'Files and Folders', and 'Storage'. On the right, the main area has fields for 'Name' and 'Description'. The 'Name' field contains 'Backup NAS' and has a red arrow pointing to it. The 'Description' field contains 'Created by VBACKUP\Administrato'.

Name	Name: Backup NAS
Files and Folders	Description: Created by VBACKUP\Administrato
Storage	

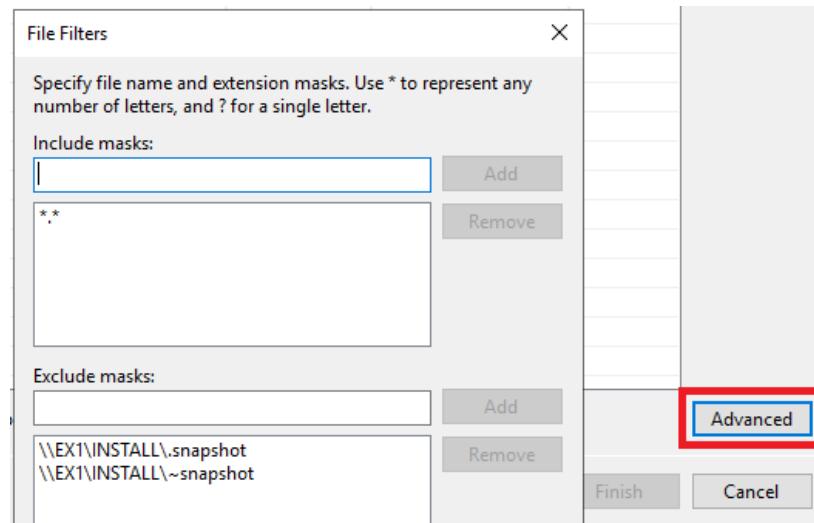
Indicamos los directorios que queremos respaldar:

Si solo indicamos el recurso compartido, se respaldarán todos los directorios que cuelgan del mismo.

The screenshot shows the 'Files and Folders' step of the backup job configuration. On the left, there is a sidebar with 'Name' (selected) and 'Storage'. On the right, the main area shows a table for specifying files and folders to be backed up. The table has columns for 'File or folder', 'Server', and 'File mask'. One row is present, showing '\\EX1\INSTALL' as the file or folder, '\\EX1\INSTALL' as the server, and 'Excluding \\EX1*' as the file mask.

Name	File or folder	Server	File mask
Files and Folders	\\EX1\INSTALL	\\EX1\INSTALL	Excluding \\EX1*
Storage			

y con el botón "Advanced", podemos definir exclusiones:



Indicamos repositorio donde almacenar el backup, así como la retención a utilizar:

New File Backup Job

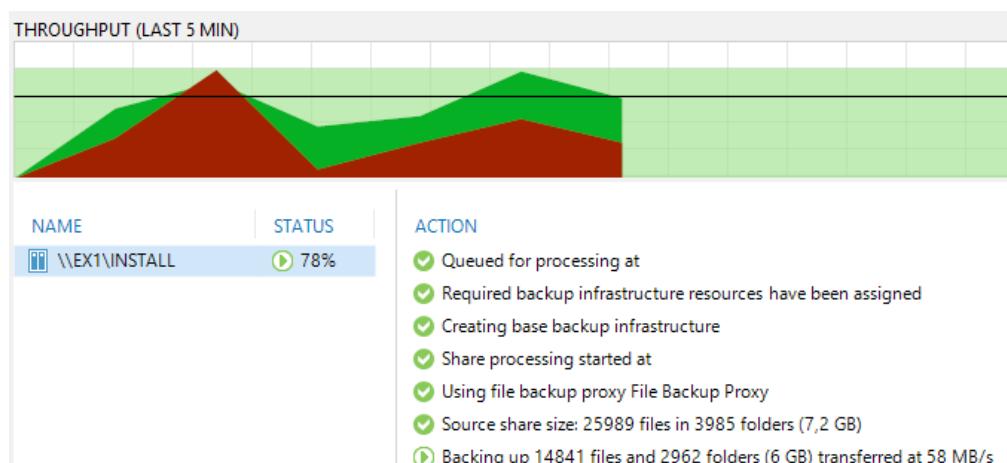
Storage
Specify target backup repository and file retention policy for this job.

Name	Backup repository:
Files and Folders	REPO (Created by VBACKUP\Administrator a 399 GB free of 399 GB)
Storage	Keep all file versions for the last: 28 Retains recent versions of each file for the spe

Finalmente, en el asistente configuraremos cuando lanzar la ejecución del job.

Resultado

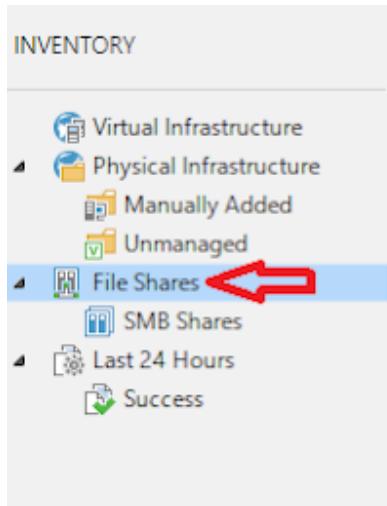
Aquí podemos ver el aspecto de la ejecución del job:



RESTORE

Para realizar el restore de los contenidos del NAS respaldado, disponemos de varias opciones.

Nos situaremos en la vista de: "Inventory" y la opción de "File Shares" y "Restore".



- [ADD FILE SHARE](#)
Add a file share or a server containing files you want to Microsoft Windows or Linux servers.
- [CREATE BACKUP JOB](#)
Create file backup job and include one or more data so both data protection and disaster recovery, as well as f
- [RESTORE](#) 
Perform the entire file share restore (in case of a compl
restore individual accidentally deleted files or older file

Aquí podemos ver las distintas opciones para el restore:

- [Restore entire share](#)
Restores the latest version of all files to the selected location service, or major storage-level corruption impacting unknown
- [Rollback to a point in time](#)
Reverts all files modified since the specific date and time to deleted. Use this option to recover from ransomware, virus c
- [Restore individual files and folders](#)
Restores the required file version, or point-in-time state of a find and restore missing files or folders, or fetch previous file

Aquí podemos elegir si rescatar todo el recurso compartido, realizar una recuperación granular de ficheros o la opción de "Rollback to a point in time" que nos permite recuperar ficheros modificados indicando una fecha y hora específicas, opción muy interesante para recuperarse de ataques de ransomware.

ESTRUCTURA DE FICHEROS EN EL REPOSITORIO

El resultado de la copia de seguridad de los ficheros de un NAS no queda guardado en ficheros VBK.

Veremos una estructura de directorios con ficheros de extensión vblob, vslice, vindex

Vista parcial de los ficheros resultantes de un job de copia al que le hemos llamado: "Backup NAS".

```
D:\REPO\Backup_NAS>tree /f
Folder PATH listing for volume DATOS
Volume serial number is 88A2-8952
D:.
    Backup_NAS.vstore
        e4ca80fded364773b4754d5780ea0c00
            e4ca80fded364773b4754d5780ea0
                data
                    408b5cf10f3940a9a118683ab
                        0000.vblob
                        0001.vblob
                        0002.vblob
                        0003.vblob
                        0004.vblob
                        0005.vblob
                        0006.vblob
                        0007.vblob
                        0008.vblob
                        0009.vblob
                        000a.vblob
                        000b.vblob
                        000c.vblob
                        000d.vblob
                        000e.vblob
                        000f.vblob
                        000g.vblob
                        000h.vblob
                        000i.vblob
                        000j.vblob
                        000k.vblob
                        000l.vblob
                        000m.vblob
                        000n.vblob
                        000o.vblob
                        000p.vblob
                        000q.vblob
                        000r.vblob
                        000s.vblob
                        000t.vblob
                        000u.vblob
                        000v.vblob
                        000w.vblob
                        000x.vblob
                        000y.vblob
                        000z.vblob
                meta
                    00f.vir
                    00f.vsl
                    01f.vir
                    01f.vsl
                    02f.vir
                    02f.vsl
                    03f.vir
                    03f.vsl
                    04f.vir
                    04f.vsl
                    05f.vir
                    05f.vsl
                    06f.vir
                    06f.vsl
                    07f.vir
                    07f.vsl
                    08f.vir
                    08f.vsl
                    09f.vir
                    09f.vsl
                    0af.vir
                    0af.vsl
                    0bf.vir
                    0bf.vsl
                    0cf.vir
                    0cf.vsl
                    0df.vir
                    0df.vsl
                    0ef.vir
                    0ef.vsl
                    0ff.vir
                    0ff.vsl
                    0000.vir
                    0001.vir
                    0002.vir
                    0003.vir
                    0004.vir
                    0005.vir
                    0006.vir
                    0007.vir
                    0008.vir
                    0009.vir
                    000a.vir
                    000b.vir
                    000c.vir
                    000d.vir
                    000e.vir
                    000f.vir
                    000g.vir
                    000h.vir
                    000i.vir
                    000j.vir
                    000k.vir
                    000l.vir
                    000m.vir
                    000n.vir
                    000o.vir
                    000p.vir
                    000q.vir
                    000r.vir
                    000s.vir
                    000t.vir
                    000u.vir
                    000v.vir
                    000w.vir
                    000x.vir
                    000y.vir
                    000z.vir
                    0000.vsl
                    0001.vsl
                    0002.vsl
                    0003.vsl
                    0004.vsl
                    0005.vsl
                    0006.vsl
                    0007.vsl
                    0008.vsl
                    0009.vsl
                    000a.vsl
                    000b.vsl
                    000c.vsl
                    000d.vsl
                    000e.vsl
                    000f.vsl
                    000g.vsl
                    000h.vsl
                    000i.vsl
                    000j.vsl
                    000k.vsl
                    000l.vsl
                    000m.vsl
                    000n.vsl
                    000o.vsl
                    000p.vsl
                    000q.vsl
                    000r.vsl
                    000s.vsl
                    000t.vsl
                    000u.vsl
                    000v.vsl
                    000w.vsl
                    000x.vsl
                    000y.vsl
                    000z.vsl
```

BACKUPS INCREMENTALES

Si hacemos backup de ficheros en un NAS:

¿Podemos hacer copias de seguridad incrementales al no haber CBT?

La respuesta es que sí.

Veeam Backup & Replication utiliza la funcionalidad CFT: Changed File Tracking

Con CFT, Veeam Backup & Replication puede determinar qué archivos han cambiado dentro de un sistema de archivos, de forma que cuando se realiza una copia de seguridad de un sistema NAS y se realiza una copia de seguridad incremental, el proceso no necesita recorrer todo el sistema de archivos para descubrir qué ha cambiado.

Con CFT cuando hay un cambio en una subcarpeta, solo reconoce los cambios de la carpeta que contiene los objetos alterados y todas las carpetas anteriores, a la carpeta principal.

Es por este motivo, el punto anterior, el resultado del Backup es una estructura de ficheros y directorios.

Podéis encontrar la versión web de este apartado en la siguiente URL:

<https://www.sysadmit.com/2020/05/veeam-backup-nas-backup.html>

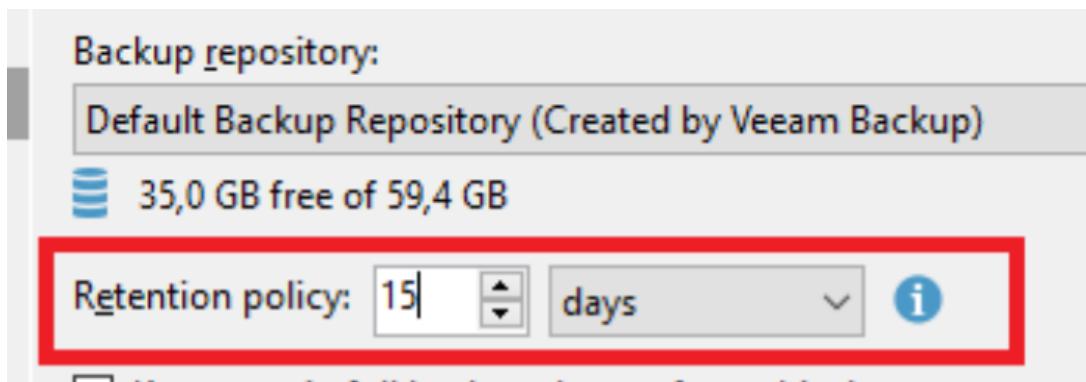
VEEAM BACKUP & REPLICATION: GFS

A partir de la versión 10 de Veeam Backup & Replication se incluye la opción GFS: "Grandfather-Father-Son" en el mismo job de respaldo primario.

¿QUÉ ES?

La opción GFS sirve para que sobre una copia full realizada no se le aplique la retención configurada a nivel general.

Imaginemos que tenemos un job de Backup cuya retención es de 15 días:



Con la opción de GFS podemos configurar que dentro de esta cadena de 15 días haya una copia full que permanezca más allá de los 15 días.

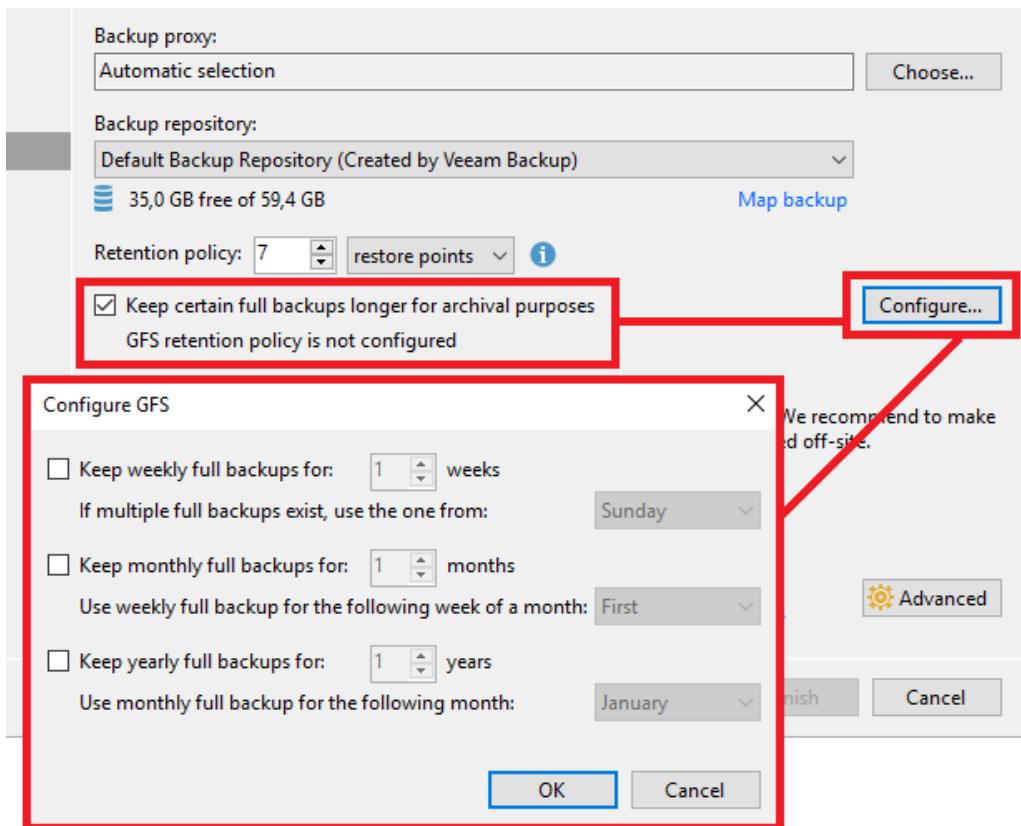
¿DÓNDE SE CONFIGURA?

La configuración de GFS se configura en el apartado de: "Storage" de un "Backup job".

Para activar GFS, debemos marcar la opción: "**Keep certain full backup longer for archival purposes**"

A continuación, si pulsamos el botón: "Configure" podemos configurar cuando queremos que se aplique la política GFS.

Tenemos: Semanal, mensual o anual.



VEEAM BACKUP GFS: "BACKUP COPY" VS "GFS"

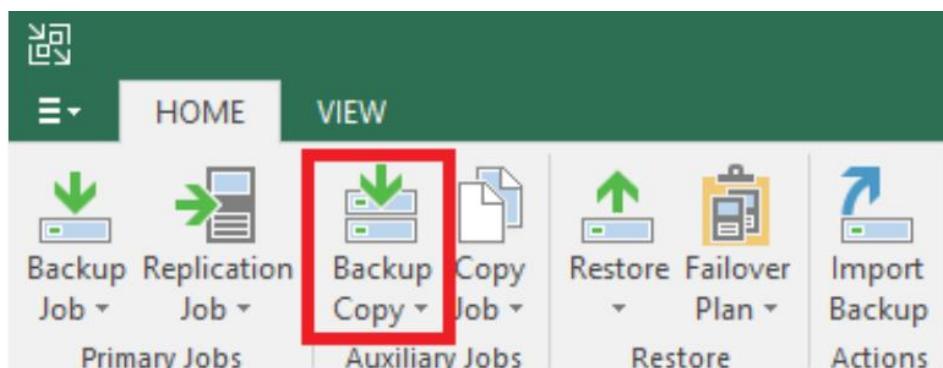
Muchos se estarán haciendo la siguiente pregunta: ¿Cómo podía realizar un archivado de un Backup y saltándose la retención configurada antes de la versión 10 de Veeam Backup & Replication?

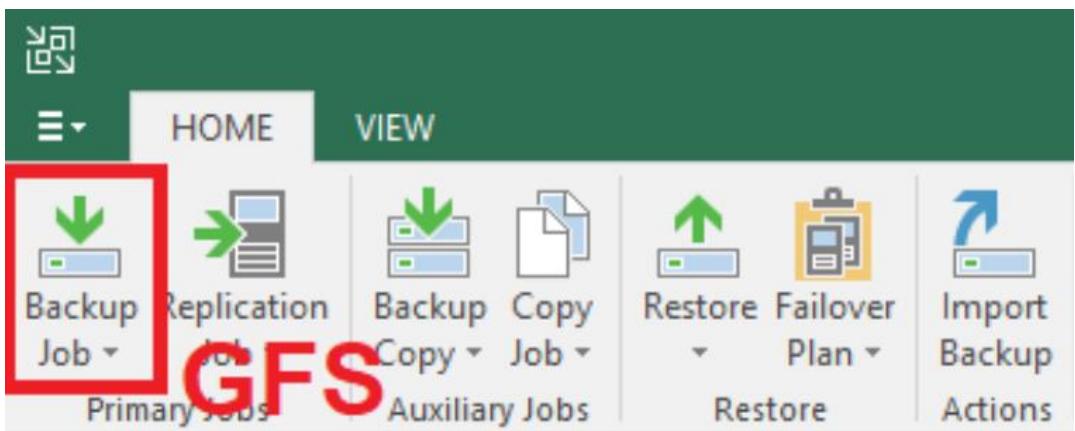
La respuesta es utilizando "Backup copy".

"Backup copy" es un tipo de job y allí configuraremos que una copia del respaldo primario sea enviada a otro lugar.

Fijaos donde se configura cada una de las opciones:

"Backup copy":





Como podemos ver en las imágenes:

- GFS es una opción de un "Backup job" y por tanto realizará la retención distinta (retención GFS) configurada dentro de la cadena de ficheros del Backup job normal.
- "Backup copy" es otro tipo de job y por tanto no alterará la cadena de ficheros del "Backup job".

Ambos tienen en común que permiten el archivado del Backup independientemente de la retención general configurada.

La diferencia está en que GFS lo realizará en la propia cadena de ficheros del Backup job, mientras que "Backup copy" no.

Con "Backup copy" podremos situar la copia en otro repositorio, con GFS no.

GFS marca con un flag una copia full que no puede ser alterada por la retención general del job de Backup.

GFS no crea un full Backup, marca un full Backup para que no sea alterado por la retención general.

¿Y SI NO HAY FULL BACKUP?

Un ejemplo:

Marcamos GFS para el sábado y este día no se realiza ninguna copia full, se realiza incremental.

¿Qué ocurre?

El backup job del sábado al tratarse de un backup incremental no se le puede marcar el flag de GFS, por tanto, pasará al siguiente día hasta que el backup job realice una copia full.

Cuando esto suceda, imaginemos el lunes, automáticamente se le asignará el flag de GFS.

Recordemos que GFS no genera copias full, solo marca copias full para que no sean eliminadas por la retención general del job.

VEEAM BACKUP: LINUX BACKUP PROXY

Otra de las novedades que incorpora la versión 10 de Veeam Backup & Replication es la posibilidad de utilizar un Backup Proxy virtual Linux sobre VMware.

Hasta la versión anterior a Veeam Backup & Replication 10 solo se podía utilizar un Backup Proxy sobre sistemas operativos Windows.

Recordemos que el componente Proxy de Veeam Backup & Replication es el encargado de realizar las tareas de compresión y deduplicación, es el encargado de procesar los datos.

Si desplegamos más de un Backup Proxy, podremos realizar ese trabajo de forma paralela en varios equipos.

LINUX VS WINDOWS

Utilizar Backup Proxy basados en sistemas operativos Linux nos permitirán:

- No usar licenciamiento de Windows Server para alojar los Backup Proxy.
- Proxy más ligero: Como el sistema operativo es menos pesado, habrá más recursos para las tareas del proxy.

Requisitos

Los requisitos para utilizar un Linux Backup Proxy son los siguientes:

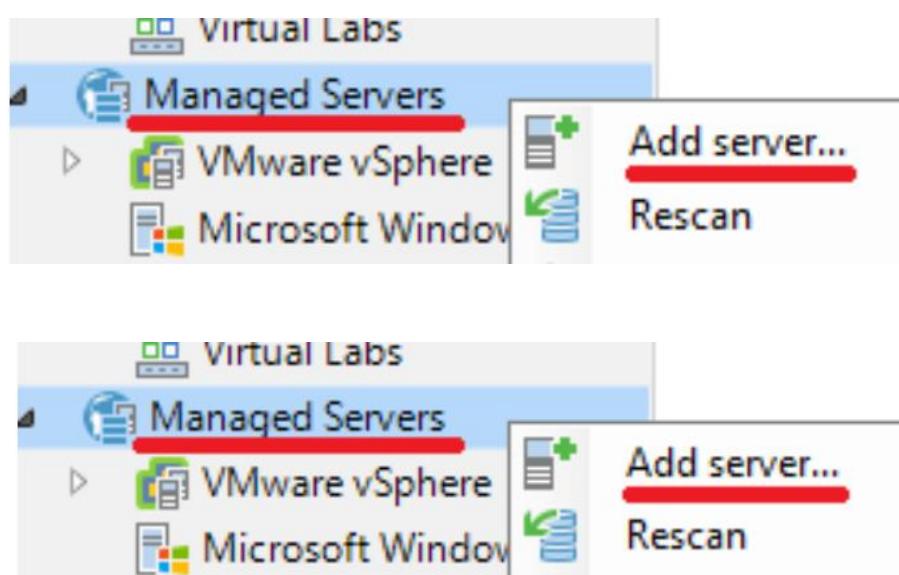
- Se requieren credenciales de *root*.
- Se requiere conexión SSH vía autenticación con password. En el fichero de configuración del servicio SSH **/etc/ssh/sshd_config**, el parámetro **PasswordAuthentication** tiene que estar configurado en yes.
- El parámetro **disk.EnableUUID** de la VM Linux donde queremos instalar el Backup Proxy debe estar en *TRUE*.
- Solo se permite el modo de transporte **Virtual appliance**, por tanto, no podremos montar un Linux Backup Proxy en un equipo físico.
- No se permite instalar un Linux Backup Proxy en AWS ya que se requiere poder acceder a los parámetros VDDK.
- No se puede utilizar Linux backup proxy en los siguientes escenarios: Réplica sobre aceleradores WAN, VM copy, Backup de file share e integración con sistemas de storage.

¿CÓMO SE CONFIGURA?

Para configurar un Linux Backup Proxy seguiremos los siguientes pasos:

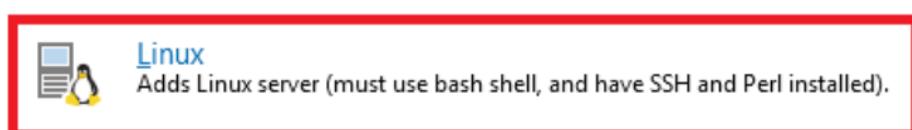
Paso 1: Añadir la VM Linux que queremos utilizar como Linux BackupProxy

Dentro de la vista: "Backup infrastructure", nos situaremos en: "Managed Servers", una vez allí, seleccionaremos: "Add server".

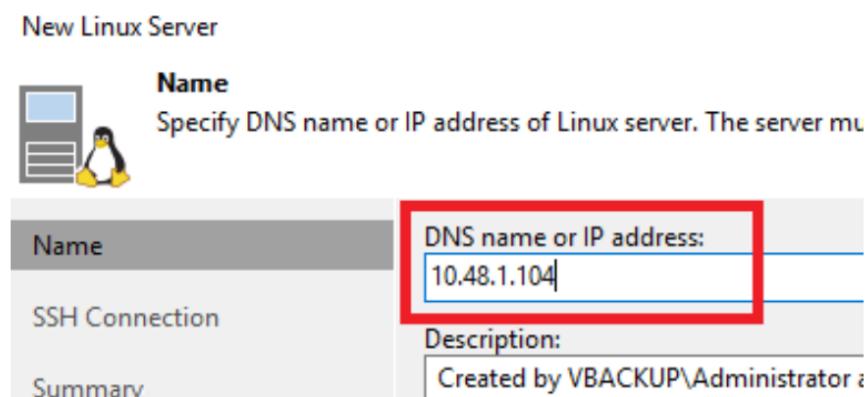


A continuación, veremos el asistente para añadir el servidor administrado.

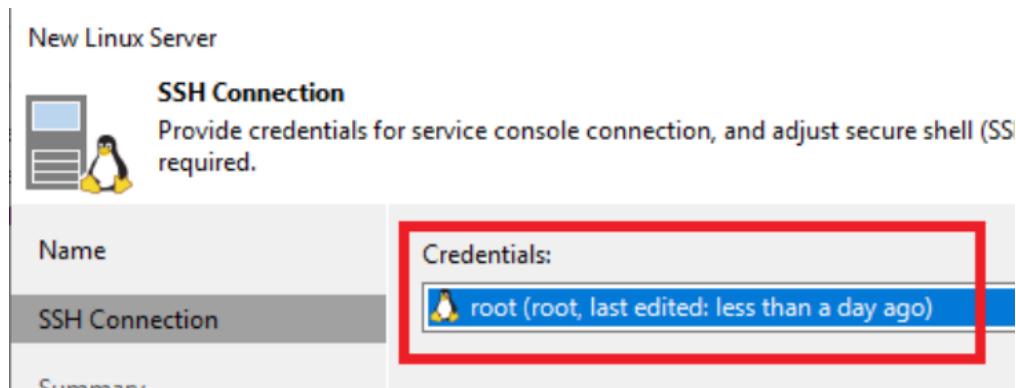
Como tipo de servidor, seleccionaremos: Linux



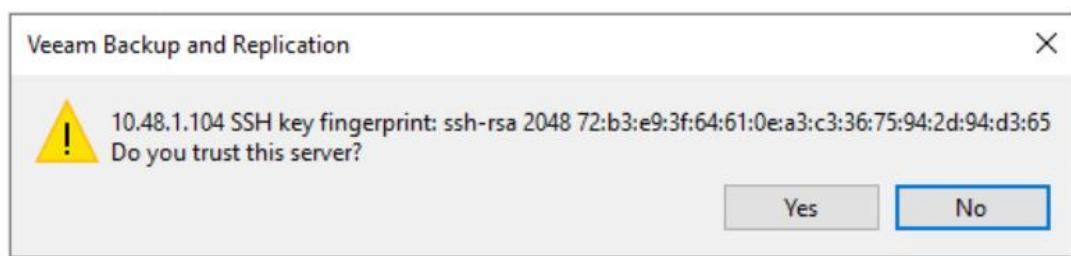
Indicamos la dirección IP o hostname del equipo Linux a añadir:



Introducimos las credenciales:



y ya se establece la primera conexión SSH al equipo:



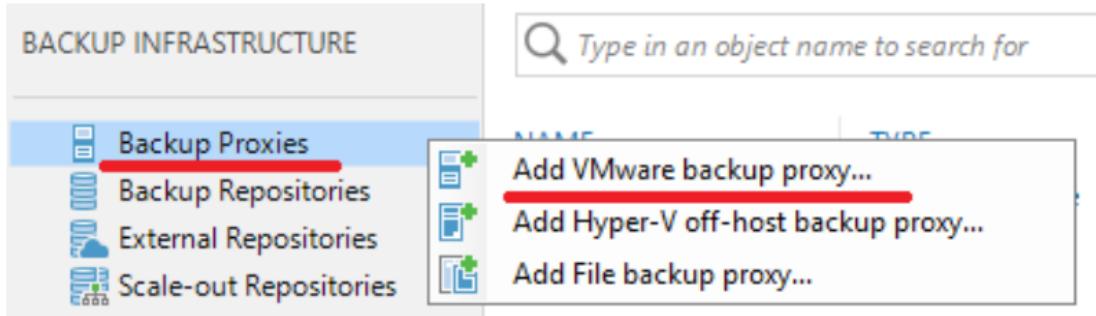
Si no podemos realizar una conexión vía SSH como root, podemos repasar este post que explica cómo hacerlo:

[Linux: Habilitar SSH root \(SYSADMIT.com\)](#)

Paso 2: Añadir el Linux backup proxy como proxy

El siguiente paso es añadir el Linux Backup Proxy.

Para ello, nos dirigiremos a: Backup Infrastructure - Backup Proxies - Add Proxy - VMware vSphere



Seleccionamos el Linux backup proxy:

New VMware Proxy

Server

Select a server to use for backup proxy. You can choose between any Microsoft server in your inventory and not assigned a backup proxy role already.

Choose server:
10.48.1.104 (Created by VBACKUP\Administrator)

Server

Traffic Rules

Proxy description:

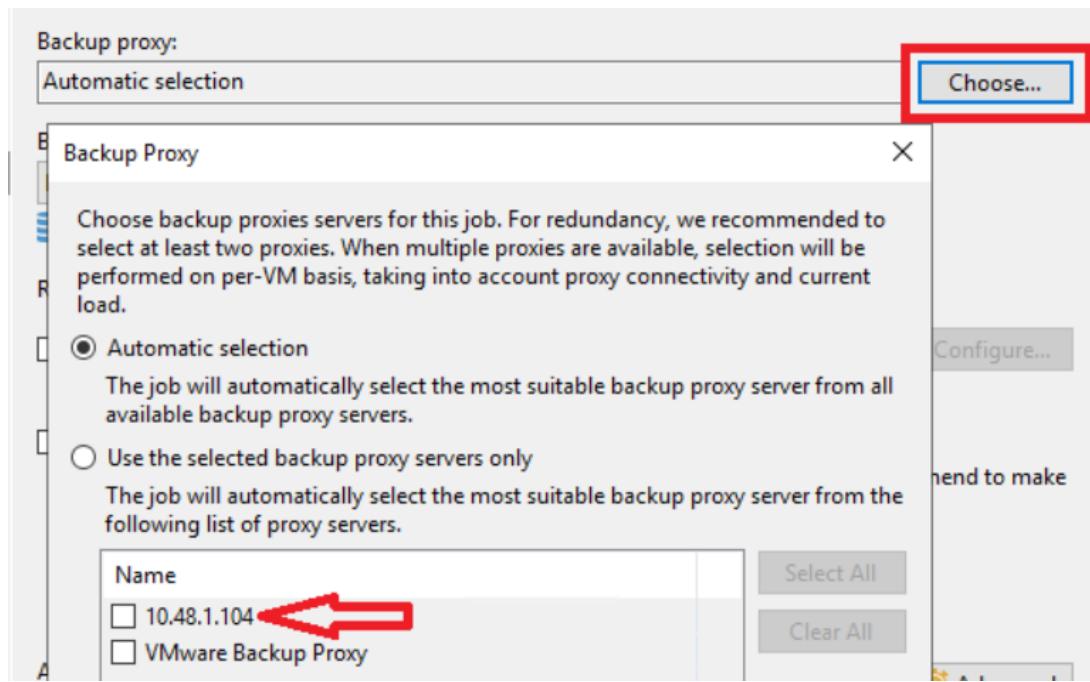
y completamos el asistente.

Paso 3: Prueba. Configuramos un job para utilizar el Linux Backup Proxy

Finalmente, para probar su funcionamiento, bastará con crear o editar un job de backup

En el apartado de "Storage", podemos seleccionar el "Backup Proxy"

Siempre tendremos el "VMware Backup Proxy" que es el proxy que se crea automáticamente al instalar Veeam y después tenemos el proxy que hemos instalado:



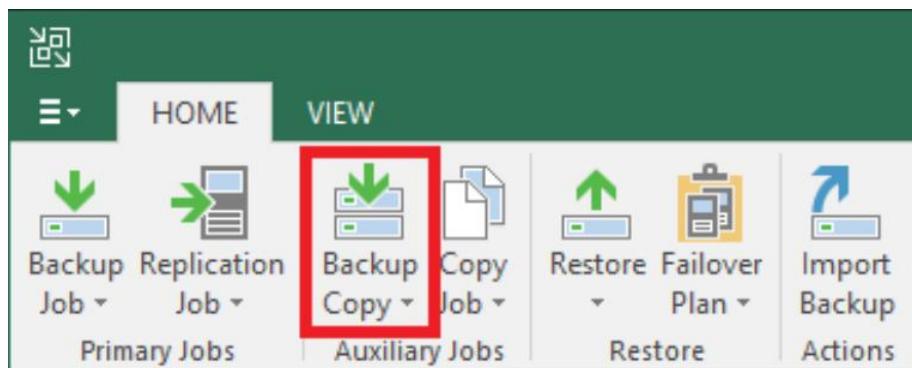
VEEAM BACKUP: IMMEDIATE COPY (MIRRORING)

Una de las novedades introducidas en la versión 10 de Veeam Backup & Replication es el modo de copia: Immediate copy (mirroring) en los backup copy job.

¿Qué es un Backup copy job?

Antes de entender que es el modo Immediate copy (mirroring), debemos entender que es un Backup copy job.

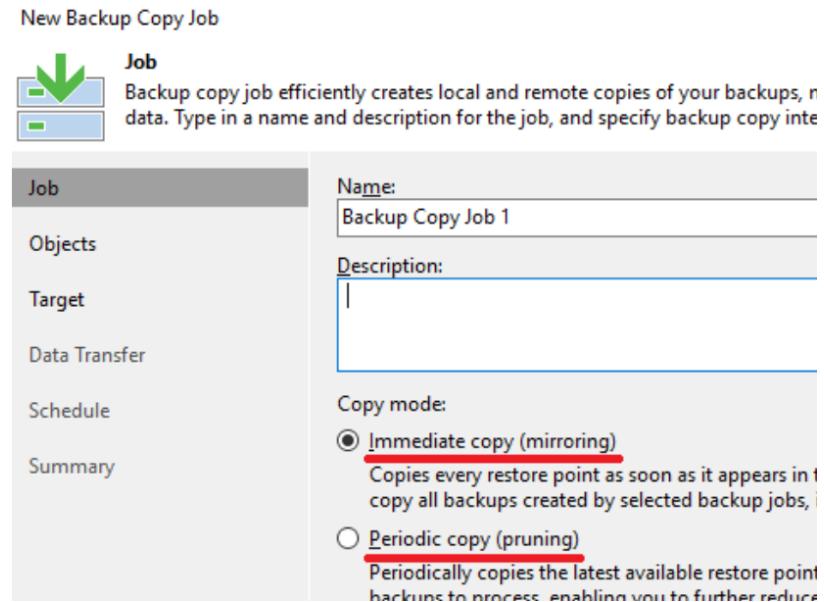
La opción de backup copy job la encontraremos aquí:



La idea de un Backup copy job es realizar una copia de seguridad de un job existente a una ubicación alternativa, es decir a otro repositorio.

¿QUÉ ES IMMEDIATE COPY (MIRRORING)?

Si repasamos las opciones del Backup copy job a partir de la versión 10 de Veeam Backup & Replication, veremos que existen dos modos de copia:



Immediate copy (mirroring):

Nueva opción a partir de la versión 10 de Veeam Backup & Replication.

Esta opción copia todos los puntos de restauración que hay en el repositorio primario del job en el momento que aparecen.

Para entenderlo de forma fácil, es capaz de copiar una cadena de Backups de forma instantánea.

Es capaz de copiar logs de transacciones (si existen) de SQL y Oracle.

Requiere ficheros de Backup "Per-VM", esto significa que necesitamos que cada VM respaldada corresponda a un fichero.

Aquí tenemos la explicación y donde se ubica la opción "Per-VM":

[Veeam Backup: Per-VM backup files \(SYSADMIT.com\)](#)

Periodic copy (pruning):

Esta opción solo copia el último punto de restauración disponible.

No es capaz de copiar logs de transacciones de SQL y Oracle.

Como podemos ver, la opción de Periodic copy (pruning) consumirá menos recursos ya que solo copiará el último punto de restauración disponible.

PARÁMETROS COMUNES

Además de elegir Immediate copy (mirroring) o Periodic copy (pruning), configuraremos:

Objects:

Seleccionamos los jobs a copiar.

New Backup Copy Job

Objects
Add backup jobs which backups should be mirrored to the target repository and transaction log backups.

Job	Objects to process:
Name	Type
Objects	WWWA
Target	Backup Job

Target:

Configuramos el destino: El repositorio, los puntos de restauración a mantener en el destino y si queremos una retención a nivel de archivado.

New Backup Copy Job

Target
Specify the target backup repository, number of recent restore points to keep and use map backup functionality to seed backup files.

Job	Backup repository:
Objects	Default Backup Repository (Created by Veeam Back)
Target	34,9 GB free of 59,4 GB
Data Transfer	Restore points to keep: 7
	<input type="checkbox"/> Keep the following restore points as full backups

Data Transfer:

Configuramos el tipo de transferencia, si utilizamos transferencia directa o bien utilizando un acelerador WAN.

New Backup Copy Job

Data Transfer
Choose how object data should be transferred from source to target backup job.

Job **Direct**
Objects Object data will be sent directly from source to target backup job, suitable for copying backups on-site, and off-site over a fast network.
Target **Through built-in WAN accelerators**
Data Transfer Object data will be sent to target repository through both source and target sites. This mode provides better performance for WAN transfers.
Source WAN accelerator:

Schedule:

Configuramos cuando queremos que se haga la copia.

Podemos configurar una copia continua o definir los periodos de tiempo en que se ejecutará. Este aspecto es muy importante de cara al consumo de recursos.

New Backup Copy Job

Schedule
Specify when this job is allowed to transfer data over the network. Backup copy jobs run continuously according to copy interval and/or as the new object restore points appear.

Job This job can transfer data:
 Any time (continuously)
 During the following time periods only:
Objects
Target
Data Transfer
Schedule
Summary

12	2	4	6	8	10	12	2	4	6	8	10	12
All												
Sunday												
Monday												
Tuesday												
Wednesday												
Thursday												
Friday												
Saturday												

VEEAM INSTANT RECOVERY P2V – VEEAM AGENT A ESXI

Una de las novedades que presenta la versión 10 de Veeam Backup & Replication es la posibilidad de recuperar un Backup de un equipo físico haciendo P2V (Physical-to-Virtual) a VMware ESXi.

Explicado de otra forma: Veeam Backup & Replication permite respaldar equipos físicos utilizando "Veeam Agent", con la versión 10 podemos restaurar una copia de seguridad realizada con "Veeam Agent" a un VMware ESXi.

Es importante tener en cuenta que para realizar conversiones P2V deberíamos utilizar la herramienta gratuita VMware Converter, aquí tenéis algunos temas para tener en cuenta para acelerar el proceso:

[VMware: Converter acelerar proceso \(SYSADMIT.com\)](#)

La funcionalidad de P2V de "Veeam Agent" a "VMware ESXi" solo debe ser utilizada con fines de recuperación de desastres.

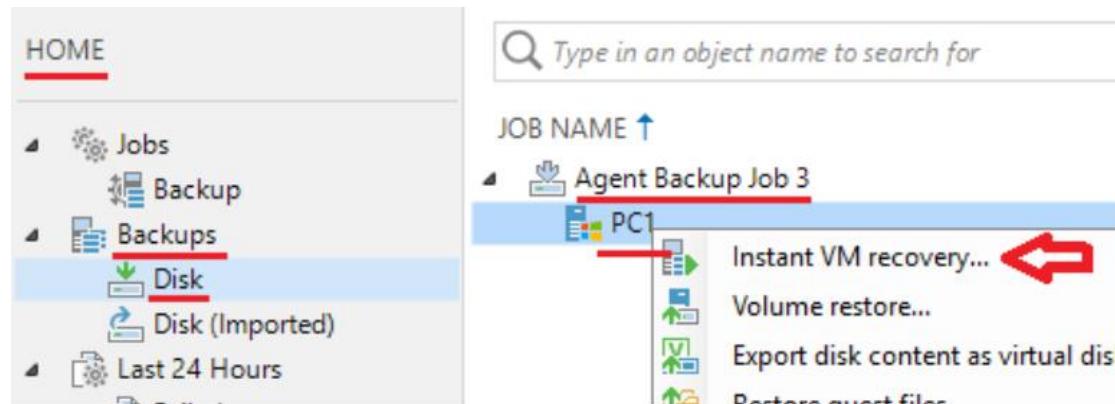
Veamos cómo funciona.

PROCEDIMIENTO DE RESTORE A VM.

En primer lugar, buscamos la copia de seguridad realizada con "Veeam Agent" de un equipo físico, para ello, nos situaremos en la vista "Home", "Backups" y "Disk".

Allí seleccionaremos un Backup realizado por "Veeam Agent".

Una vez seleccionado, haremos botón derecho, "Instant VM recovery".



Seleccionamos la máquina a restaurar:

El job podría englobar varias máquinas y tendríamos que elegir la que quisiéramos recuperar.

También podríamos seleccionar el punto de restauración a recuperar.

The screenshot shows the 'Instant Recovery to VMware' interface. On the left, there's a sidebar with icons for 'Machines' (selected), 'Destination', and 'Datastore'. The main area is titled 'Machines' and contains a sub-section 'Machines to restore:' with a search bar and a table. The table has two columns: 'Name' and 'Size'. A single row is selected, showing 'PC1' and '11 GB'. A green arrow icon is positioned above the 'Machines' section.

Indicamos los parámetros de destino:

This screenshot shows the 'Restored VM name' configuration screen. It includes fields for 'Restored VM name' (containing 'PC1', step 1), 'Host' (containing '10.48.0.71', step 2), 'VM folder' (containing 'vm'), 'Resource pool' (containing 'Resources'), and 'Networks'. The 'Networks' section contains a table with 'Source' and 'Target' columns. In the 'Source' column, there's a list with 'PC1' and its network adapter 'Intel(R) 82574L Gigabit Netw...'. In the 'Target' column, there's a list with 'VM Network' (step 3). At the bottom, there's a note about customizing BIOS UUID and an 'Advanced' button (step 4).

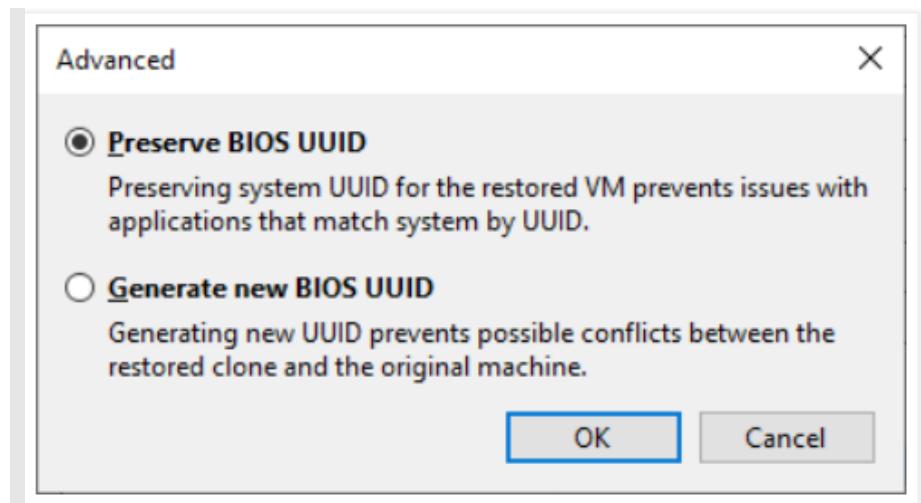
1. Nombre que queremos asignar a la VM.
2. Nombre del host VMware ESXi al que queremos restaurar el Backup en forma de VM.
3. Elegimos el "Virtual Switch" al que queremos conectar la VM.

Veremos cómo nos aparece:

Source: NIC física que tiene el equipo que hemos respaldado con Veeam Agent.

Target: "Virtual Switch" de VMware ESXi.

4. Con el botón "Advanced" podemos cambiar el UUID (Universally Unique Identifier) de BIOS.



En la mayoría de los casos, no será necesario generar un nuevo BIOS UUID.

Podemos ver el BIOS UUID de un equipo ejecutando:

En sistemas operativos Windows

wmic bios get serialnumber

En sistemas operativos Linux

dmidecode -t system | grep Serial

A continuación, seleccionamos si queremos "Redirect write cache" seleccionando un Datastore del VMware ESXi.

Seleccionando esta opción, conseguiremos mayor rendimiento si el datastore seleccionado es rápido.

The screenshot shows the 'Instant Recovery to VMware' interface. On the left, there's a sidebar with icons for 'Machines', 'Destination', and 'Datastore'. The 'Datastore' icon is highlighted with a grey background. The main panel has a title 'Datastore' with a small icon of a server with a green arrow pointing right. Below it, a descriptive text says: 'By default, changed virtual disk blocks are stored in the datastore. If you want to store them in a different location desired for performance or capacity reasons, you can redirect the write cache to another datastore.' To the right of this text is a checked checkbox labeled 'Redirect write cache'. Underneath it is a dropdown menu labeled 'Datastore:' containing the option 'DATASTORE1'. At the bottom right of the main panel, there's a status bar showing '4,1 TB free of 6,5 TB'.

Seleccionamos si queremos o no "Secure restore"

Si marcamos la opción "Secure Restore", la VM será escaneada en busca de virus antes de ser encendida en el entorno de VMware ESXi.

Para que esta opción funcione, es necesario tener instalado un antivirus compatible en el equipo donde está instalado Veeam Backup & Replication.

Por ejemplo, Windows Defender, el antivirus incluido por defecto en Windows Server 2016.

De hecho, si el antivirus puede ser llamado desde línea de comandos, podríamos integrarlo con "Secure Restore" editando el fichero XML **AntivirusInfos.xml**

The screenshot shows the 'Instant Recovery to VMware' interface with the 'Secure Restore' tab selected. On the left, there's a sidebar with icons for 'Machines', 'Destination', 'Datastore', and 'Secure Restore'. The 'Secure Restore' icon is highlighted with a grey background. The main panel has a title 'Secure Restore' with a small icon of a server with a green arrow pointing right. Below it, a descriptive text says: 'Scan the selected backup for malware, such as computer viruses or ransomware, prior to recovery. This option is available if a compatible antivirus installed on the mount server specified for the corresponding backup.' To the right of this text is a checkbox labeled 'Scan machine for virus threats prior performing recovery'. Below this checkbox, there's a note: 'Machine you are about to restore will be scanned by antivirus to ensure resulting machine will not cause virus spread after recovery.' Further down, there's a section titled 'When virus threat is detected:' with two radio button options: 'Proceed to recovery but disable VM network adapters' (selected) and 'Abort VM recovery'. At the bottom right of the main panel, there's a checkbox labeled 'Scan entire VM for virus threats'.

y finalizamos el asistente.

En el apartado de resumen, tenemos dos opciones muy interesantes que nos permiten controlar que la VM convertida sea iniciada y conectada a la red o no:

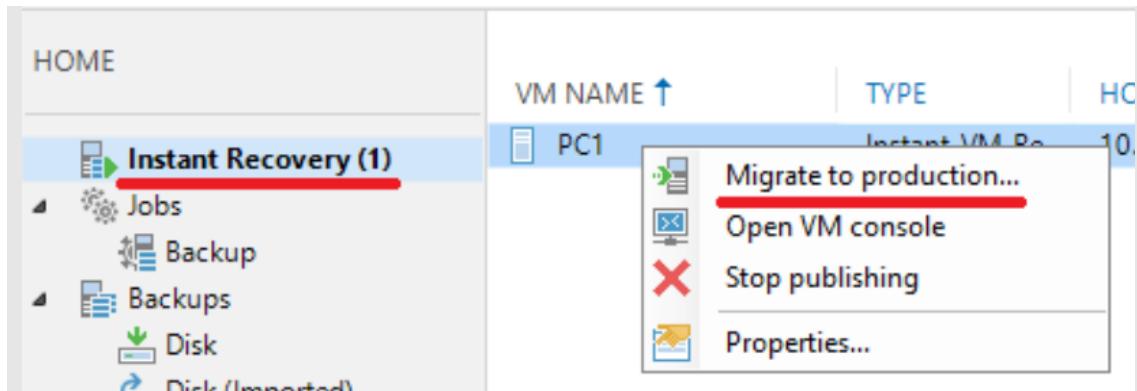
Summary	<p>storage. Alternatively, you can perform cold VM migration during your next</p> <p>If you are performing manual recovery testing, remember to change VM net before powering on the VM.</p> <p><input type="checkbox"/> Connect VM to network</p> <p><input type="checkbox"/> Power on target VM after restoring</p>
---------	---

Aquí tenemos el detalle de la ejecución del job y en el cuadro rojo vemos la conversión P2V (Physical-to-Virtual):

Reason	Parameters	Log
Message <ul style="list-style-type: none"> ✓ Starting VM PC1 recovery ✓ Connecting to host 10.48.0.71 ✓ Restoring from Default Backup Repository ✓ Checking if vPower NFS datastore is mounted on host ✓ Locking backup file ✓ Publishing VM ✓ Preparing change storage ✓ Updating VM configuration ✓ Checking free disk space available to vPower NFS server. ✓ Registering VM ✓ Creating VM snapshot ✓ Performing P2V conversion ✓ Conversion completed, first VM boot may take longer time than usual ✓ Consider installing VMware Tools in the guest OS at your convenience ✓ Updating session history ✓ PC1 has been recovered successfully ✓ Waiting for user to start migration 		

Al finalizar la conversión, podemos migrar la VM convertida al entorno de producción.

Para ello, nos situamos sobre "Instant recovery" y pulsamos sobre: "Migrate to production":



Podéis encontrar la versión web de este aparado en la siguiente URL:

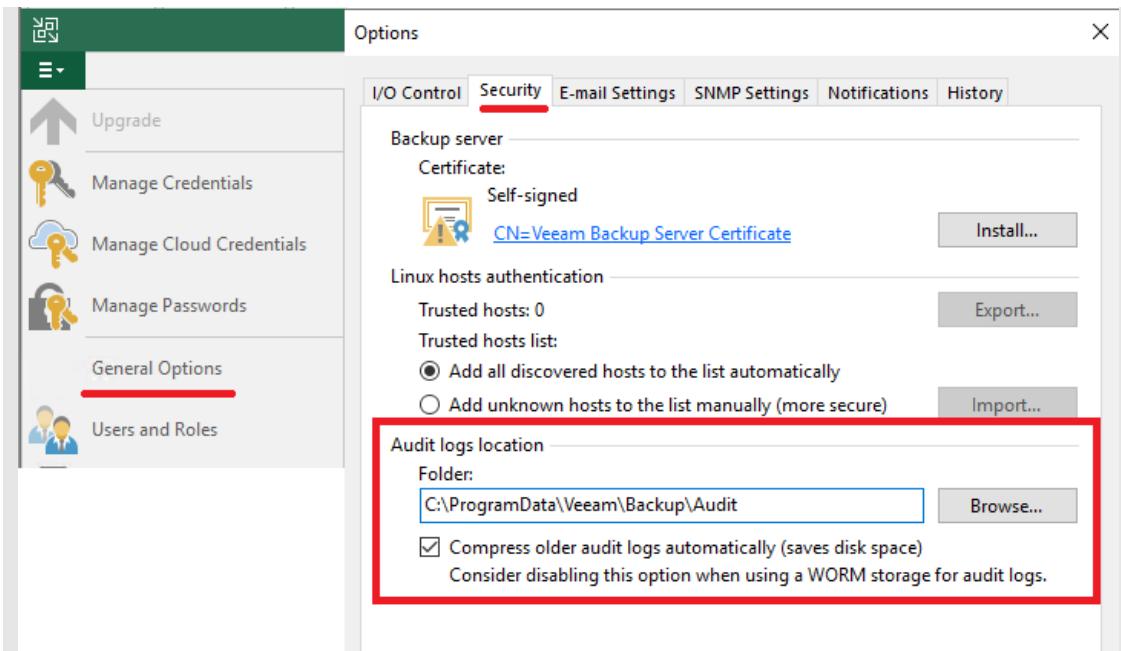
<https://www.sysadmit.com/2020/05/veeam-backup-p2v-veeam-agent-esxi.html>

VEEAM BACKUP: AUDIT LOGS – RUTA

Otra de las novedades que incorpora la versión 10 es la posibilidad de definir la ruta de los logs de auditoría.

¿DÓNDE ESTÁ LA OPCIÓN?

Si nos situamos donde residen las opciones generales de Veeam Backup, veremos que en la pestaña "Security" existe un apartado con el nombre: "Audit logs location":



Como podemos ver, la ruta por defecto apunta a: C:\ProgramData\Veeam\Backup\audit y dentro de esta carpeta veremos ficheros comprimidos en ZIP y en su interior los logs.

VEEAM BACKUP: AUDIT LOGS RUTA ¿QUÉ CONTIENEN LOS LOGS?

Estos logs contienen información de auditoría en formato CSV de las sesiones realizadas de: "File Level Restore".

Es importante entender que, si vía el explorador de ficheros que ha abierto la copia hacemos un copiar-pegar, esta acción no quedará registrada en este log.

Si examinamos el log generado, dentro del CSV podemos ver: Hora, Usuario, SID, Operación, Resultado, Objeto.

Ejemplo:

07.06.2020 17:05:00Z, SYSADMIT\Administrador, S-1-5-21-XXXXXXX-XXXXXXX-XXXXXXX-500, Restore, Success, E:\DATOS\fichero.xlsx



Soluciones Cloud Profesionales



NUBE
PÚBLICA



NUBE
PRIVADA



NUBE
HÍBRIDA

EXPERTOS EN



Microsoft Azure

CARACTERÍSTICAS

- Disponible en **54 regiones, 140 países, 1'6 Pbps de ancho de banda por región**
- Nube **fácilmente escalable**
- Reduce **costes de infraestructura y servicio**
- Transforma **tu negocio**
- Segura y robusta
- Aumento de la **productividad**
- Alta disponibilidad

APLICACIONES

- SISTEMAS DE DISASTER RECOVERY
- COPIAS DE SEGURIDAD
- SAP, ERP
- ENTORNOS DESARROLLO
- BBDD
- ENTORNOS DE PRUEBAS
- ESCRITORIOS VIRTUALES

ISO 9001
BUREAU VERITAS
Certification



Microsoft

vmware®

COHE_SITY veeAM

A CORUÑA:

C/Fontaíña 56 Bajo
15.404 – Ferrol (A Coruña)

OURENSE:

Rúa Carriarico, 21
32.002 - Ourense

www.2ksystems.com

info@2ksystems.com

981 369 519

Capítulo 9

CLOUD NATIVE APPS PARA ADMINISTRADORES DE VSPHERE



Federico Cinalli

@FCinalliP

CLOUD NATIVE APPS PARA ADMINISTRADORES DE VSphere

INTRODUCCIÓN

Para los que trabajamos en sistemas desde hace varios años y que nunca desarrollamos software, la aparición de las Cloud Native Apps y la forma en que está impactando, tanto en las operaciones como en la infraestructura misma de los sistemas, hizo que llegue un punto en que ya no podemos mirar hacia otro lado y terminemos claudicando para, finalmente, salir de nuestra zona de confort y aprender la base de estas nuevas tecnologías.

Existen múltiples conceptos asociados a lo que se conoce como “Cloud Native” con los que deberíamos estar familiarizados, incluso tal vez en un orden determinado, para poder comprender primero el motivo del cambio, la evolución y el cambio de paradigma que trajo consigo tecnologías como Contenedores, Kubernetes y recientemente Tanzu.

El objetivo de este capítulo es facilitar la transición desde los conocimientos de base que actualmente disponemos los administradores de vSphere, y los sysadmins en general, hacia las tecnologías anteriormente mencionadas como Kubernetes y Tanzu, de las cuales encontrarás en este mismo libro un excelente material y mayor profundidad en conceptos escrita por maravillosos autores.

Comenzaremos el capítulo con un poco de historia, los motivos que generaron los cambios y cómo poco a poco se fueron alineando los caminos del Software Defined Datacenter con las bases de las Cloud Native Apps.

Seguiremos con los principales “nuevos” conceptos a incorporar, siempre desde el punto de vista de un sysadmin, y los iremos encajando en el nuevo puzzle a la vez que iremos sumando más acrónimos a la lista.

Por último, terminaremos el capítulo con la base de la nueva infraestructura de vSphere 7 con el add-on de Kubernetes soportando, de forma nativa, Contenedores y Kubernetes en los Clusters y Hosts de ESXi.

¿Empezamos?

DEL DATACENTER FÍSICO A LA TRANSFORMACIÓN DIGITAL

Es muy difícil imaginarse hoy en día un centro de datos que opere sin una base de virtualización. Tanto en tamaño, por costos operativos, gestión, mantenimiento y eficiencia en general.

Hace muchos años ya que la virtualización es un commodity que obligó a empresas como VMware, con una cuota de mercado enorme, a renovarse y reinventarse a sí misma.

Pero no solo empresas como VMware tuvieron que reinventarse y redefinirse.

Antes de comenzar a analizar los cambios en las tecnologías de Infraestructura y la modernización del desarrollo de software veamos cómo la Transformación Digital es, en gran medida, una gran impulsora de semejantes cambios.

Debo confesar que las primeras veces que comencé a escuchar el término “Transformación Digital” lo asociaba automáticamente a un término de marketing que se sumaba a la lista de palabras y frases de moda como disruptivo, paradigma, time-to-market, etc, etc.

Pero un día me encontré en París en una sala repleta de colegas de profesión y nos iban a hacer una presentación de Digital Transformation en las siguientes dos horas. No había escapatoria ni café que aguante semejante tortura.

Como si fuera poco al terminar la sesión me llegó el bonus track: “Federico, mañana te toca exponer un resumen de esta presentación”. ¿Algo podría mejorar eso?

No hay nada mejor que tener que explicar algo a los demás para obligarte a entenderlo.

Lo cierto es que las circunstancias hicieron no solo que lo entienda, sino que además hasta que me creyera esto del Digital Transformation. Ahí va mi resumen.

Si de repente tuviésemos que nombrar una de las empresas tecnológicas más exitosa de los últimos 10 años que fue capaz de superar en crecimiento a Amazon, Google, Facebook y Microsoft, ¿cuál sería? La número 1 es Netflix. ¿Y la número 2?

¿Cómo se nos quedaría la cara si nos dijeran que la empresa tecnológica de la que estamos hablando es Domino's Pizza?



Domino's Pizza, empresa tecnológica

En 2010 Domino's Pizza venía acumulando pérdidas de cuota de mercado con la competencia, así como también se reducían los ingresos y el valor de sus acciones en bolsa cayeron hasta U\$D 13,08.-

En ese mismo año la compañía reposiciona a uno de sus ejecutivos como el nuevo CEO y éste giró el timón hacia una dirección basada en la tecnología. En poco tiempo la mitad del personal que trabajaba en la sede central, en Michigan, era de perfil tecnológico.

Entre los principales cambios que llevaron a cabo fue potenciar las plataformas para pedidos on-line y diversificar la logística y los medios de entrega.

Domino's fue la primera cadena de pizzas en ofrecer un sistema de tracking a los pedidos online.

Los métodos para poder hacer un pedido se multiplicaron ofreciendo plataformas diversas como Smartwatch, Alexa, Facebook, Smart TVs, vía SMS, ¡Google home y hasta vía Twitter!



App de Domino's en Smartwatch

La apuesta e inversiones en redes sociales, las aplicaciones, la logística en general y proyectos de I+D como entrega con drones y coches 100% eléctricos y autónomos posicionaron a Domino's Pizza como número 1 en los Estados Unidos y no precisamente porque sea la pizza más sabrosa.

Como resultado el valor de las acciones pasó de U\$D 13,08.- el 21 de agosto de 2010 a unos increíbles U\$D 419,51.- exactamente 10 años después.



Ahora podemos preguntarnos ¿qué tienen en común empresas como Domino's Pizza, Netflix, Uber, Airbnb y Alibaba? **Son empresas tecnológicas que venden servicios y/o productos.**



Comenzaron ofreciendo alquiler de DVD's con envío postal.

En 2007 se adelantaron a Blockbuster y movieron su negocio a Cloud.

El valor de las acciones subió desde U\$D 3,54.- en 2007 hasta U\$D 494.-

Netflix representa el 15% del tráfico mundial de Internet.



Empresa creada en el año 2009.
Es la empresa de transporte más grande del mundo.
No es propietaria de ningún vehículo.
Su software es el que hace la magia de conectar Conductores con Viajeros.

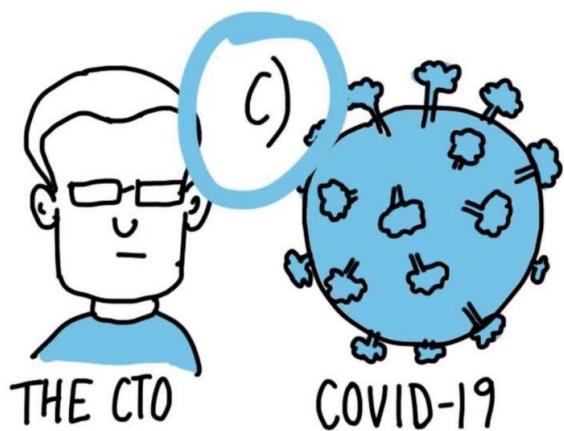
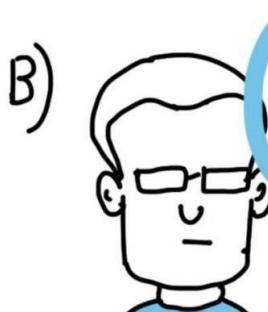
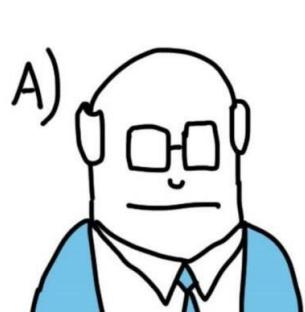


La mayor empresa de retail del mundo.
No dispone de inventario.
Su plataforma de software lo hace todo.



La empresa más grande en oferta de alojamiento privado.
No es propietaria de ningún hotel o casa para alquiler.
El Software que da servicio a su web y el marketing online hacen el trabajo.

WHO LED THE DIGITAL TRANSFORMATION OF YOUR COMPANY ?

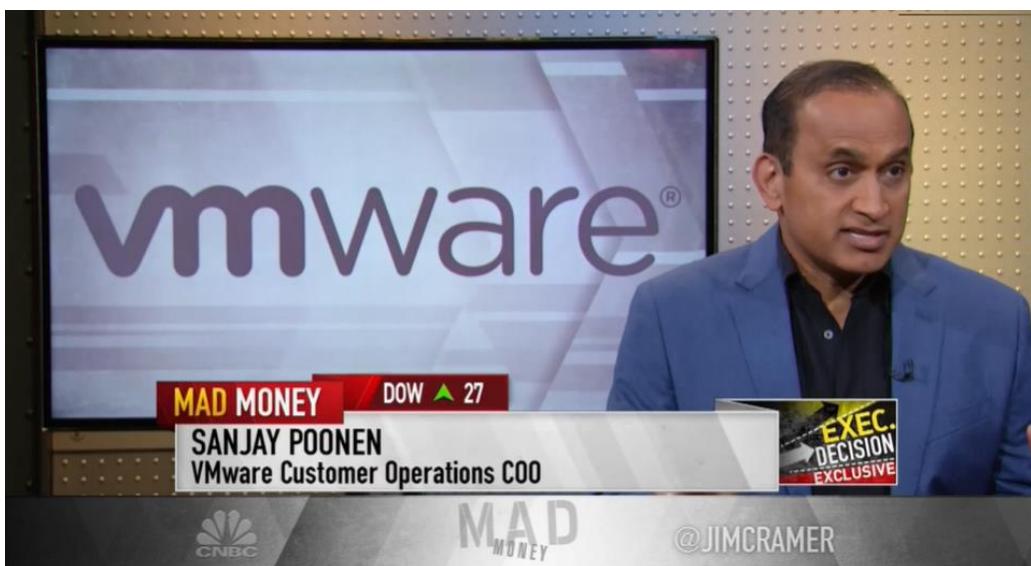


LA TRANSFORMACIÓN DIGITAL Y EL TIME TO MARKET

Como hemos visto en los ejemplos anteriores el denominador común de estos casos de éxito es mejorar la experiencia del usuario a través de la tecnología.

Ese cambio tecnológico tiene como uno de sus principales pilares el software y la forma de llegar a los consumidores.

La aceleración tecnológica en las empresas requiere no solo un cambio cultural para adaptarse a los nuevos desafíos sino también, además de la incorporación de nueva tecnología y modernización del software, recalcular su velocidad de crucero a los tiempos que demanda el mercado para ser competitivos y ofrecer algo realmente innovador.



Sanjay Poonen (VMware COO): “Estamos en la edad dorada del Software”

Canal Youtube de CNBC

Se espera que en los próximos 5 años se desarrollarán más aplicaciones que en los últimos 40 años.

El **Time to Market** define el tiempo que una empresa necesita para poner en el mercado un nuevo producto o servicio y poder ser de los primeros en ofrecerlo para ganar cuota de mercado.

En otras palabras, la Transformación Digital y el Time to Market llegaron para patear el tablero y redefinir las formas, los medios, las tecnologías y los tiempos que las empresas necesitan para ofrecer algo diferente a la experiencia del usuario y convertir ese factor diferenciador en resultados.

Esta situación generó una necesidad de cambio en la forma en que se desarrolla, entrega, mantiene y renueva el software.

A continuación, veremos cómo evolucionaron las tecnologías para llegar a lo que hoy en día conocemos como Cloud Native.

DE LA VIRTUALIZACIÓN AL CLOUD NATIVE

Pasaron muchos años desde que en 1999 VMware encendió su primera Máquina Virtual y que, desde ese momento, se revolucionaron los Centros de Datos Físicos.

Si bien hoy en día las tecnologías de virtualización están presentes en todos los Centros de Datos, somos testigos directos de la Modernización de Aplicaciones.

Esas aplicaciones modernas requieren de un ecosistema determinado para poder ser desplegadas, actualizadas, monitorizadas, automatizadas, securizadas y eliminadas. Estamos hablando de un entorno nativo Cloud o, como suele llamarse, Cloud Native.

Veamos a continuación la evolución de las tecnologías para luego centrarnos en comprenderlas desde un punto de vista de un administrador de vSphere o Sysadmin.



A partir de 1999 VMware tuvo un crecimiento y una evolución tecnológica impresionante.

Su primer producto fue VMware Workstation y las primeras ventas llegaron principalmente de universidades.

En 2002 publicaron el primer Hipervisor apuntando ya al mercado Enterprise y ese mismo año se migró la primera Máquina Virtual de un Host a otro. Hoy en día todavía sigue siendo increíble ver cómo una VM puede moverse sin dejar de dar servicio.

Para 2003 ya disponíamos de la versión 2.0 de ESX y se presentó en sociedad vCenter 1.0 para comenzar a gestionar los recursos de forma centralizada.

En ese mismo año 2003, mientras el móvil o teléfono celular más popular era el Nokia 1100, dos empleados de Amazon (Chris Pinkham y Benjamin Black) propusieron a la compañía comercializar determinados servicios basados en la misma plataforma que utilizaban para dar servicios a amazon.com.



Nokia 1100 - El móvil más popular de 2003

Si bien fue en 2006 cuando se lanzó oficialmente AWS, realmente sucedió a partir de 2004 cuando comenzaron a comercializar el primer servicio llamado Simple Queue Service mientras en Sudáfrica se desarrollaba EC2.



El año 2007 nos traía una versión mejorada de vCenter la cual ya era capaz de aplicar las recomendaciones de DRS para seguir las reglas de afinidad y anti-afinidad además de hacer uso de vMotion para balancear la carga de CPU y RAM.

A partir de ese año los teléfonos móviles ya no serían los mismos después de la grandiosa presentación que hizo Steve Jobs del iPhone 1.0 con sus increíbles 4 GB de capacidad, su cámara de 2 Megapíxeles y la inédita hasta la fecha funcionalidad Multi-touch.

No era el primer teléfono considerado inteligente, pero sí que abría una nueva página en cuanto a la siguiente generación de smartphones.

El 29 de junio de 2007 fue el lanzamiento oficial del iPhone y también en cierta forma la semilla de lo que pronto sería la revolución de las Apps, pilar fundamental hoy en día en la Transformación Digital.



El iPhone 1.0

En menos de un año, más precisamente el 7 de abril de 2008, el gigante Google presenta su plataforma Cloud que hoy conocemos como GCP o Google Cloud Platform.

Si bien eran años en donde la virtualización se hacía poco a poco más fuerte en las infraestructuras On-premise, las plataformas Cloud públicas comenzaban a aparecer ofreciendo un puñado de servicios.

VMware se hacía cada vez más fuerte en el Centro de Datos con los comienzos del VDI para virtualizar el puesto de trabajo. Hoy en día en épocas de pandemias podemos trabajar desde cualquier sitio gracias a la evolución de esas tecnologías de VDI de diferentes fabricantes.

Si bien Microsoft no se hizo lo suficientemente fuerte en el mercado de la virtualización, sí que tuvo la visión de apostar al mercado Cloud Público lanzando de forma oficial su plataforma Azure en febrero de 2010.

Poco a poco se fueron cosechando los que serían los ingredientes de la Transformación Digital y las Cloud Native Apps.



Tan solo dos años después de su creación en septiembre de 2009, Nutanix presentaría su primer producto basado en lo que hoy conocemos como Hipervconvergencia (HCI).

Los fundadores de Nutanix se basaron en la forma en que Google y Facebook gestionaban sus infraestructuras de cómputo y almacenamiento para poder escalar de forma granular utilizando recursos de almacenamiento locales.

Nutanix se centró al principio en su solución de almacenamiento distribuido para entornos HCI y claramente fué el pionero en esta solución que hoy en día tiene un enorme protagonismo en Infraestructuras On-premise.

El VMworld de 2011 fue muy interesante por varios motivos que tendrán un impacto considerable en los años siguientes.

Se comienza a entender al Hipervisor como un commodity y esto obliga a VMware a tener que reinventarse, ya mirando de reojo a las plataformas Clouds Públicas y entornos Multi-Cloud.

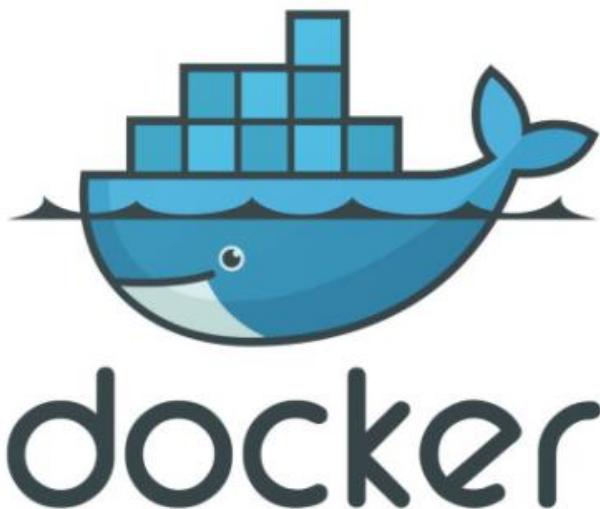
Por primera vez se acuña el término SDDC o Centro de Datos Definido por Software que tendrá una relevancia fundamental en los lanzamientos de productos de VMware en un futuro muy cercano.

Mientras que VMware premiaba a soluciones como Nutanix y Veeam, comenzaba a alimentar su portfolio de soluciones para el SDDC como vCOPS (hoy en día vRealize Operations Manager) y miraba con cierto cariño el producto de Nutanix vislumbrando un buen futuro.

En 2012 la compra de NICIRA por parte de VMware sería la crónica de un divorcio anunciado entre Cisco y VMware al estar ambos buscando ser líder de mercado de lo que hoy conocemos como SDN para la virtualización de las Redes y la Seguridad.

Solo pasaría un año desde la compra de NICIRA hasta que, producto de la fusión entre vCNS (vShield) y NICIRA, NSX for vSphere estuviera disponible en su primera versión.

Mientras tanto los equipos de desarrollo de Software que estaban muy a la vanguardia comenzaban a adaptar poco a poco nuevas prácticas alineadas con metodologías AGILE, el uso de Microservicios y la aplicación de DevOps para mejorar la eficiencia.



Tecnología de Contenedores basado en Código Abierto y Software Libre

La aparición, en Marzo de 2013, de Docker como tecnología de Contenedores hizo el resto y para esa fecha ya estaban prácticamente todos los ingredientes para la modernización de Aplicaciones.



El SDDC anunciado en el VMworld de 2012 ya tenía fundamentos tan solo 2 años pasado el anuncio con el SDN basado en NSX for vSphere y el SDS nativo en Hipervisor de vSAN preparando un cóctel ideal para Infraestructuras HCI.

El VMworld del año 2014 también presentaría lo que hoy conocemos como CMP (Cloud Management Platform) al hacer un rebranding importante con varios productos y crear la Suite vRealize.

vCloud Automation Center (vCAC) pasaría a llamarse vRealize Automation (vRA).

vCenter Orchestrator (el gran desconocido) sería ahora vRealize Orchestrator (vRO).

vCenter Operations (vCOPS) comenzaría a llamarse vRealize Operations Manager (vROps).

Por último, ITBM se renombró como vRealize Business y se sumará en un futuro vRealize Log Insight y vRealize Network Insight.

La Suite vRealize se presenta como el CMP o Cloud Management Platform para la gestión, monitorización y automatización de recursos para entornos Multi-Cloud.

Llegados a este punto podemos apreciar la visión de futuro de IT hacia entornos Multi-Cloud.

El éxito del desarrollo con tecnologías Agile y la tendencia en alza de implementación de Contenedores sumado a la creciente demanda de prácticas DevOps generó una necesidad para ayudar a la gestión del despliegue, autoescalado, gestión y automatización de Aplicaciones basadas en Contenedores.

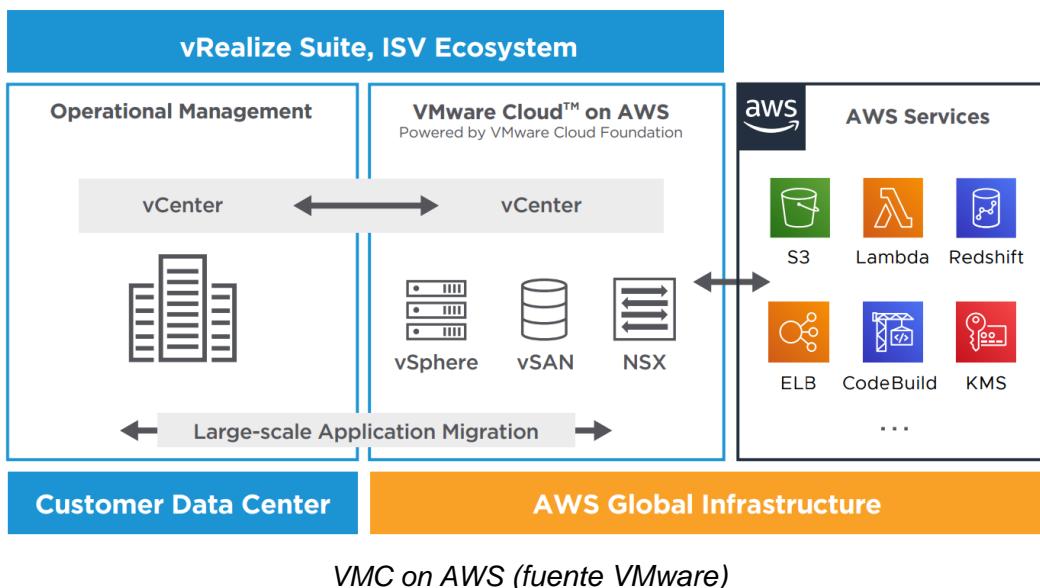
Google creó Kubernetes en 2014 para dar soporte a las nuevas necesidades generadas.

En el año 2015 se creó la Cloud Native Computing Foundation (CNCF) como una rama de la Linux Foundation para ayudar en el avance de las tecnologías de Contenedores y Google decidió donar Kubernetes a la CNCF.

Ahora sí, gracias al aporte de Google con Kubernetes y el empuje de la CNCF podemos decir que estaban listos todos los ingredientes para dar soporte a lo que hoy llamamos Cloud Native Apps. Hablaremos de esto en el siguiente punto.

Ya hemos mencionado que VMware entendió el futuro de las plataformas basadas en entornos Multi-Cloud y que, si bien continuaba teniendo una cuota de mercado enorme en Virtualización On-premise, debía adaptarse a las nuevas reglas que regirán en el mercado.

Esto supuso el desarrollo de NSX-T como solución de SDN para entornos Multi-Cloud y en 2016 presentó la versión 1.0 que sería tímidamente adaptada por los clientes al disponer actualmente de NSX para vSphere, cada vez más estable y en pleno apogeo de implementaciones.



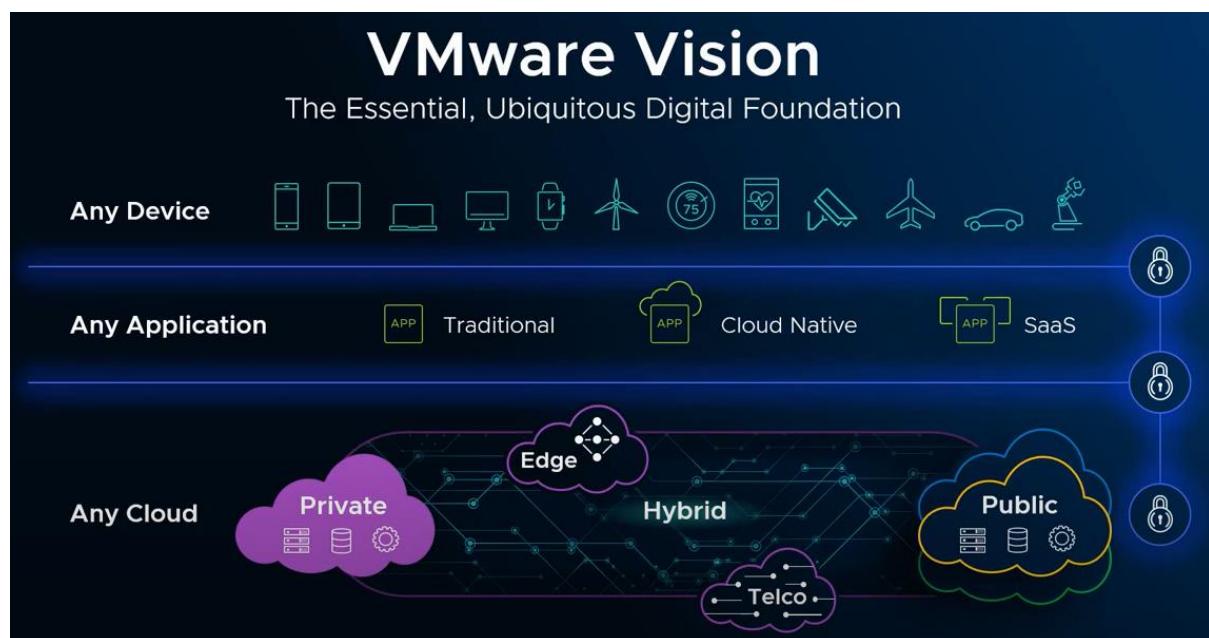
Y confirmando el camino Multi-Cloud se asoció con AWS ofreciendo lo que hoy conocemos como VMC on AWS y la posibilidad de migrar Máquinas Virtuales con Hardware virtual VMware a Datacenters de AWS pudiendo consumir tanto los servicios de VMware como así también soluciones de AWS como puede ser Almacenamiento S3, ejecución de código basado en eventos con Lambda o bien Bases de Datos RDS por mencionar algunas.



Hacía varios años ya que VMware tenía claro que el futuro pasaba por soportar una infraestructura híbrida tanto de Máquinas Virtuales (también llamadas Legacy) y Contenedores orquestados por Kubernetes.

¿Pero qué podría hacer diferente la oferta de VMware cuando plataformas de Cloud Públicas ofrecen Kubernetes de forma “nativa”?

La existencia de las Máquinas Virtuales no tiene una fecha de caducidad clara, aunque sí que es cierto que los nuevos desarrollos son Cloud Native y, sumado a eso, las soluciones IoT y Edge Computing tendrán más y más protagonismo.



La apuesta de VMware es Any App, Any Device, Any Cloud (fuente VMware)

VMware inició su viaje al Hybrid-Cloud preparando el terreno para ofrecer una Infraestructura Híbrida y Multi-Cloud capaz de dar soporte tanto a Máquinas Virtuales, Cloud Native Apps y Edge Computing para entornos On-premise y en diversas plataformas de Cloud Públicas.

Pivotal Container Service sería en cierta forma un aperitivo de Kubernetes en vSphere pero a la vez toda una declaración de intenciones. Podemos decir que todavía no estaba preparado todo el ecosistema que VMware consideraba que necesitaba para rodear a Kubernetes y hacer competitiva la oferta.

Para conseguir ese ecosistema era necesario salir de compras y, por suerte para VMware, la tesorería gozaba de muy buena salud.

Heptio fue fundada en 2016 por Joe Beda y Craig McLuckie, dos de los tres creadores de Kubernetes (Google, 2014), y su core principal era proveer de servicios profesionales a empresas que estaban adoptando o utilizando actualmente Kubernetes.

Tanto servicios de soporte como formación complementaban la oferta de Heptio.



Joe Beda (izquierda) y Craig McLuckie (derecha)

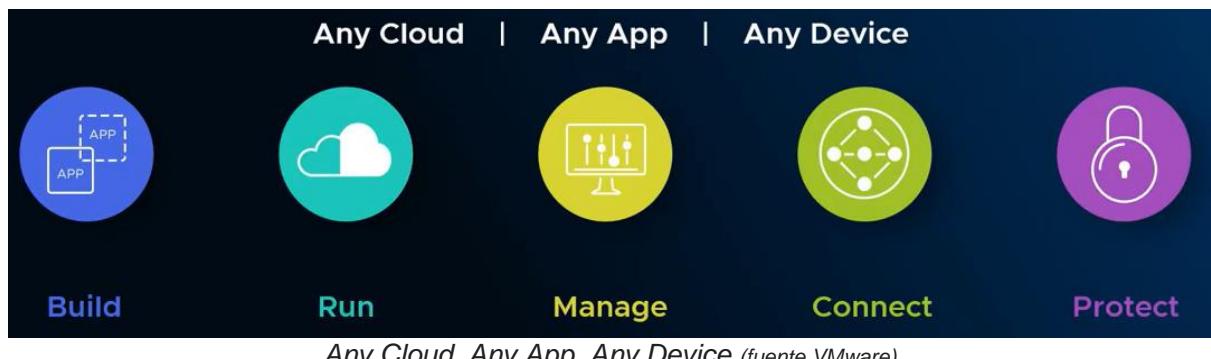
El hecho de poder contar en VMware con dos de los tres co-fundadores de Kubernetes, especialmente Joe Beda, permitía disponer de un aliado fundamental en la visión del negocio de Cloud Native Apps.

Ya con Heptio aportando el know-how, NSX-T en plena evolución y la maquinaria a pleno rendimiento, 2019 sería un gran año para la consolidación del proyecto.

No solamente estaban muy avanzados los acuerdos de ofrecer VMware en Google y Azure, sino que además sería el momento de dos adquisiciones estratégicas para lo que se venía denominando Project Pacific.

Bitnami, empresa creada en Sevilla, se posicionó como referente en empaquetado y mantenimiento de aplicaciones para Contenedores. Curiosamente ésa no era la línea principal de la compañía, pero tanto la tecnología como el destino quiso que un producto que tenían como secundario no solo se posicione como referente en el mercado, sino que además termine llamando tanto la atención a una empresa como VMware que finalmente cerraron la adquisición en febrero de 2019.

Finalmente, y también en 2019, VMware se hace con Carbon Black (Seguridad en Cloud Native) y Pivotal (Plataforma de desarrollo de Cloud Native Apps) para consolidar la oferta Desarrollar, Desplegar, Mantener, Conectar y Securizar Aplicaciones Modernas bajo la modalidad Any App, Any Device, Any Cloud.



A continuación, aprenderemos los conceptos básicos de criterios y componentes de Cloud Native Apps para finalizar el capítulo entendiendo las bases del cambio más importante en la historia de vSphere para poder ofrecer Kubernetes de forma nativa.

PRINCIPALES CONCEPTOS DE CLOUD NATIVE APPS

El objetivo de este capítulo es comprender las bases de las tecnologías que definen la modernización de las aplicaciones, pero desde el punto de vista de un administrador de vSphere y/o sysadmin en general.

Las tecnologías y conceptos que comentaremos a continuación no son ciencia espacial o matemática cuántica, pero, tal vez, para alguien que trabaja como administrador de sistemas necesita un pequeño cambio o ajuste de chip (aká firmware upgrade) para abrir un poco la mente y entender los cambios.

No olvidemos que disponemos en este mismo libro capítulos enteros escritos por verdaderos Cracks sobre estas tecnologías y que el objetivo de este simple capítulo es comprender los conceptos básicos de Cloud Native Apps para estar mejor preparados a la hora de profundizar en Kubernetes, Tanzu y el ecosistema en general, siempre desde el punto de vista del perfil y perspectiva de un sysadmin.

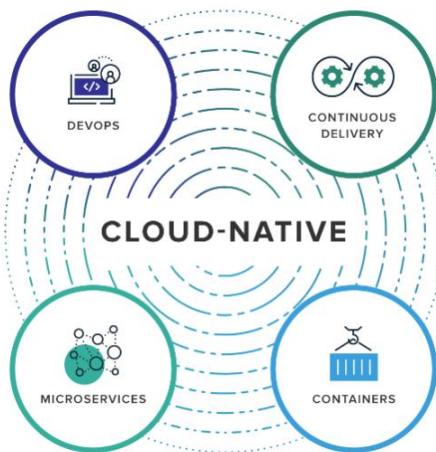
Si no te aburriste con las primeras páginas de este capítulo y las leíste completitas ya sabrás perfectamente el porqué de los cambios y la evolución de la tecnología por lo que vamos a comenzar con los conceptos fundamentales de rodean las Cloud Native Apps.

MODERNIZACIÓN DE APLICACIONES

Existen múltiples capas que forman parte de la modernización de las aplicaciones o de lo que se considera hoy que un Software sea Cloud Native.

Comencemos con la definición de *Cloud Native* según la Cloud Native Computing Foundation:

“Aplicaciones escalables ejecutándose en entornos dinámicos que utilizan tecnologías como Contenedores, Microservicios y APIs declarativas”.



También podríamos agregar otras características como que las aplicaciones sean portables y fáciles de escalar.

Veamos los principales cambios en la modernización de las aplicaciones a nivel de las capas de Desarrollo, Arquitectura e Infraestructura.

METODOLOGÍA Y DESARROLLO

Metodología / Desarrollo Waterfall Agile DevOps

Evolución de la metodología del desarrollo y la entrega

Waterfall

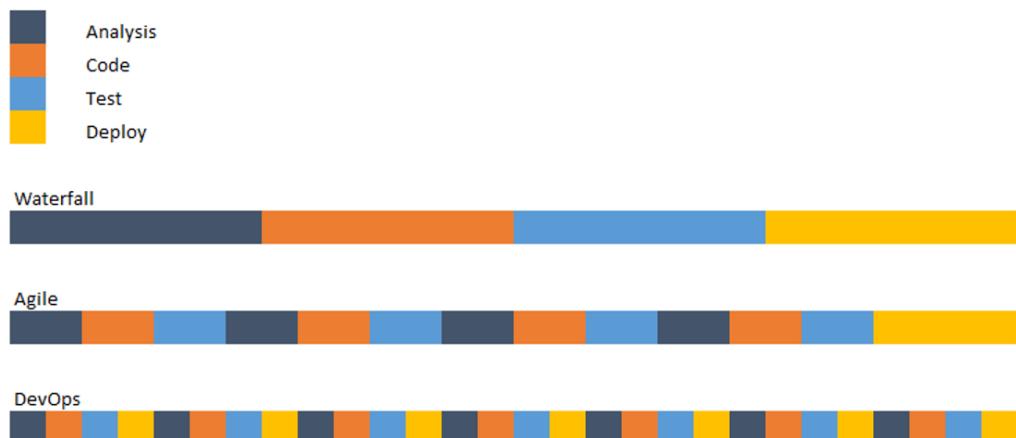
Al principio los proyectos de desarrollo eran muy extensos, así como también los tiempos que requerían una actualización. A menudo se encontraban que debido a una nueva petición del cliente debía cambiar el diseño durante el proceso. Estos procesos podían durar tranquilamente entre 6 y 12 meses. A esta metodología se la conoce como Waterfall.

Agile

La evolución requería otra dinámica y una forma diferente de gestionar los desarrollos.

La metodología Agile viene a flexibilizar y dinamizar tanto la organización como el trabajo en sí mismo.

Básicamente consiste en distribuir todo el proyecto en pequeños “pedacitos” para acelerar la entrega y gestionar tiempos más dinámicos a la vez que facilita el trabajo en equipo.



Comparativa entre Waterfall, Agile y DevOps (Fuente stevefenton.co.uk)

DevOps

Otro concepto de la modernización es lo que se conoce como integración y entrega continua (CI Continuous Integration - CD Continuous Delivery) y está asociado a DevOps.

Existen múltiples conceptos de DevOps o, tal vez, el concepto de DevOps depende de la perspectiva desde donde se lo mire.

Independientemente de la perspectiva DevOps ayuda a dinamizar las entregas tanto de Software como también de Infraestructuras y de ahí el nombre Dev (Developer) Ops (Operations).

Lo cierto es que la cultura DevOps, para los Administradores de vSphere, tiene mucho que ver con el SDDC y productos como vRealize Orchestrator, vRealize Automation y la posibilidad de integrar soluciones como Ansible, Chef, Puppet, Terraform e incluso Kubernetes!

ARQUITECTURA

Arquitectura

Monolítica

SOA

Microservicios

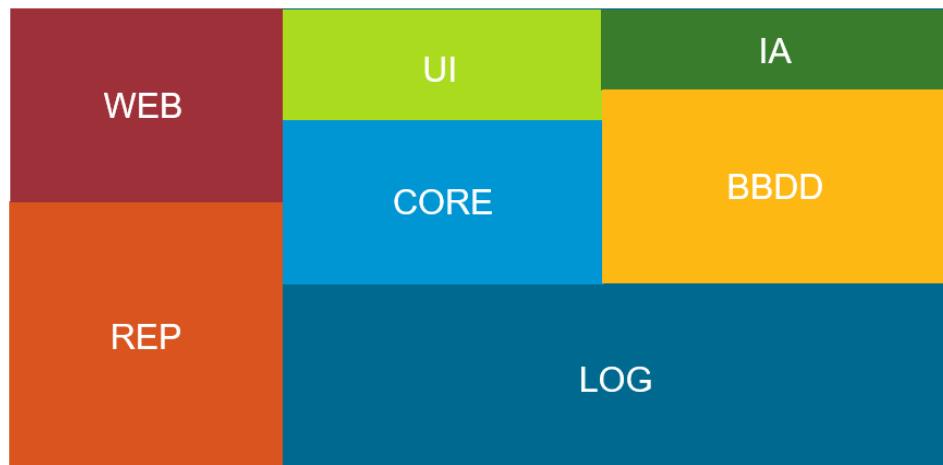
Aplicaciones Monolíticas

Es lo que hoy conocemos como aplicaciones *legacy*, algo del pasado, pero en cuestiones tecnológicas ya que todavía existen (y existirán) una enorme cantidad de aplicaciones monolíticas que dan servicio a todo tipo de software en las compañías.

Se hace referencia a que una aplicación es monolítica cuando la gran mayoría de sus componentes y dependencias opera en la misma instancia de máquina, física o virtual, que da recursos de cómputo, almacenamiento y red.

Las aplicaciones monolíticas están asociadas al método de desarrollo waterfall, es decir largos procesos de diseño, análisis, desarrollo, pruebas, actualizaciones y finalmente implementación que solían durar varios meses.

Como podemos imaginar, estos métodos están muy lejos del **time-to-market** y el dinamismo asociado que se requiere hoy en día en plena **transformación digital**.



Aplicación Monolítica con todos sus componentes en un único recurso

Estas aplicaciones monolíticas traen consigo un problema asociado que es la escalabilidad. Normalmente cuando necesitamos incrementar los recursos a una aplicación que se ejecuta en una máquina solemos incrementar los recursos de forma vertical, agregando más cpu, ram y disco a la máquina virtual, aunque todo tiene su límite.

Desde el punto de vista de la alta disponibilidad también estamos limitados a lo que pueda ofrecer la plataforma en cuanto a servicios de clúster (coste más alto) y balanceo, aunque no todas las aplicaciones de este tipo fueron desarrolladas con tecnologías que permiten trabajar en alta disponibilidad y, entre otras cosas, ser capaces de disponer de ventanas de mantenimiento en determinadas instancias o nodos.

SOA

Bajo el concepto de “divide y vencerás” se comenzó a aplicar hace muchos años un estilo de arquitectura orientada a servicios (Service Oriented Architecture) y podemos decir que fue la precuela de lo que terminaría siendo los Microservicios.

SOA introdujo hace varios años dos pilares fundamentales de los que, incluso hoy en día, se tiene muy presente: *People, Process and Technology*.

De forma muy resumida, la base de SOA define lo siguiente:

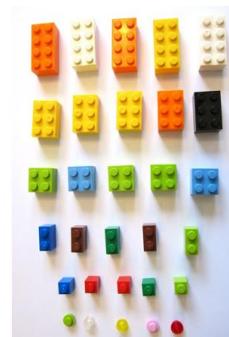
- Desarrollo de software basado en requerimientos y objetivos de negocio
- Distribución de sistemas
- Reutilización de los recursos
- Uso de estándares abiertos para permitir interoperabilidad
- Servicios basados en contextos funcionales alineados con el negocio
- Facilitar la adaptación al cambio
- Reducción de costes
- Incremento en la eficiencia de procesos



Monolítico



SOA



Microservicios

Microservicios

Una vez identificados los límites de las aplicaciones monolíticas en un contexto de negocios muy dinámico que demanda agilidad y eficiencia, más la aparición de nuevas tecnologías que permiten aplicar arquitecturas SOA (en realidad la evolución de SOA) llegamos a lo que hoy conocemos como Microservicios.

Lo que conocemos hoy en día como Cloud Native Apps está basado en las tecnologías de orquestación de contenedores y los Microservicios.

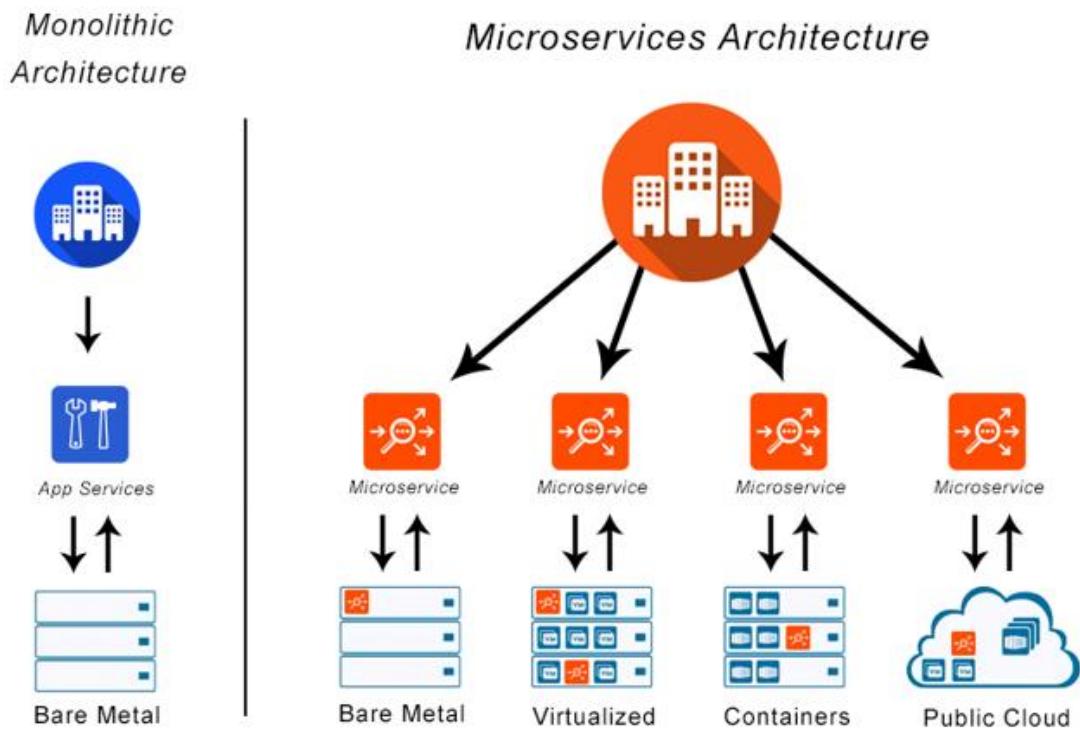
Una aplicación moderna está desarrollada por múltiples componentes individuales que dan servicio en contenedores y tienen la capacidad de ofrecer tolerancia a fallos y sistemas de auto escalamiento.

Los componentes de las aplicaciones se comunican entre sí a través de la red utilizando APIs y se complementan con los servicios de infraestructura para el balanceo de carga, la disponibilidad, la distribución y el mantenimiento.

Cada componente o proceso que forma parte de una aplicación está desarrollado con tecnologías modernas, y es el equipo de desarrollo el que define el lenguaje más apropiado a utilizar para cada servicio. Esto contrasta con las aplicaciones monolíticas que estaban desarrolladas con un único lenguaje de programación.

Cuando necesitamos aprovisionar una mayor cantidad de recursos o incrementar el número de instancias y/o réplicas de un servicio (escalamiento horizontal), simplemente lo aplicamos al servicio en sí mismo a través de reglas y políticas redefiniendo el *estado deseado* en Kubernetes.

De esta forma se consigue independencia entre procesos, dinamismo, agilidad, alta disponibilidad y escalamiento tanto manual como automático. Si a todo esto le sumamos la portabilidad que nos ofrecen las tecnologías de contenedores como Docker y una plataforma de orquestación como Kubernetes, estamos prácticamente definiendo a las Cloud Native Apps.



Llegados a este punto merece la pena que nos hagamos la siguiente pregunta:

¿ES LO MISMO CLOUD BASED QUE CLOUD NATIVE?

Una aplicación operando en Cloud puede ser algo *legacy* que fue migrado (tal vez adaptado) a una plataforma Cloud mientras que una aplicación Cloud Native fue diseñada y desarrollada desde el día 1 para trabajar bajo los principios e infraestructuras Cloud Native.

Que una máquina virtual que da soporte a una aplicación monolítica haya sido migrada a un entorno Cloud no convierte mágicamente lo *legacy* en **Cloud Native**.

Simplemente es un cambio de contexto hacia una infraestructura que se gestiona de forma más dinámica, pero lo *legacy* se mantiene al igual que las limitaciones que la definen.

Hoy en día la gran mayoría del software, actual y futuro, está siendo adaptado-rediseñado para que trabaje sobre infraestructuras Cloud Native.

INFRAESTRUCTURA

Dentro de lo que es Infraestructura poco podemos decir en este capítulo, orientado a sysadmins, sobre la evolución del Datacenter físico hacia la Virtualización. Otro cantar es el camino desde una Máquina Virtual, pasando por un Contenedor y llegando a Kubernetes en un entorno SDDC que bien podría ser Cloud Público o Privado, pero todo a su tiempo.



Evolución de la Infraestructura en la Modernización de Aplicaciones

Física y VM

El cambio del mundo físico al virtual fue una verdadera revolución consiguiendo optimizar el uso de los recursos de Cómputo, permitiendo la ejecución de múltiples instancias de máquinas virtuales con sus Sistemas Operativos y Aplicaciones sobre el mismo Host físico.

Si bien el uso de máquinas virtuales nos permite una optimización muy importante de los recursos, estas VMs están asociadas a lo que conocemos como aplicaciones Monolíticas.

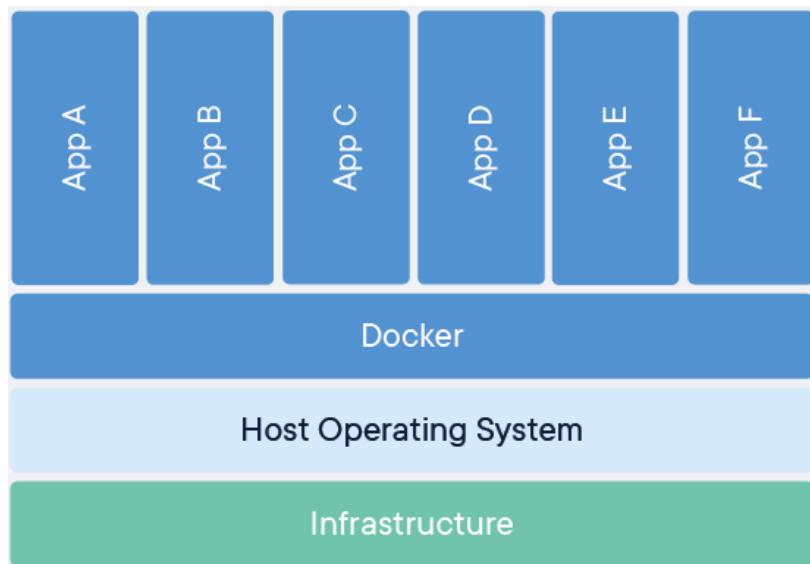
Comentaremos la evolución de las aplicaciones Monolíticas un poco más adelante.

Los contenedores hicieron su aparición hace unos cuantos años atrás, prácticamente en forma paralela a la virtualización a través de diferentes distribuciones, tal vez LXC (LinuX Containers) en 2008 ofreció la base de servicios más completa, pero definitivamente fue Docker, en 2013, la que se posicionó como “la” plataforma de Contenedores más utilizada y que ayudó a popularizar la solución consiguiendo prácticamente una metonimia al asociar automáticamente Docker con Contenedores y viceversa.

Contenedores

Los contenedores son la encapsulación de aplicaciones con sus dependencias, una especie de virtualización de aplicaciones a nivel de sistema operativo y nos permiten, además de la eficiencia y dinamismo en cuanto a consumo de recursos, ofrecer lo que se denomina ultra portabilidad.

Vendría a ser algo así como convertir una máquina virtual (Host de contenedor) en un entorno Multi-Tenant para las aplicaciones.



Múltiples Aplicaciones compartiendo dependencias bajo el mismo SO

Fuente: Docker

Para poder desplegar una aplicación en un Contenedor necesitamos primero lo que llamamos un Host Container. Ese Host Container suele ser una Máquina Virtual o también puede ser un Host Físico, además de un servicio nativo de alguna plataforma Cloud (Pública o Privada).



Diferentes plataformas de Contenedores

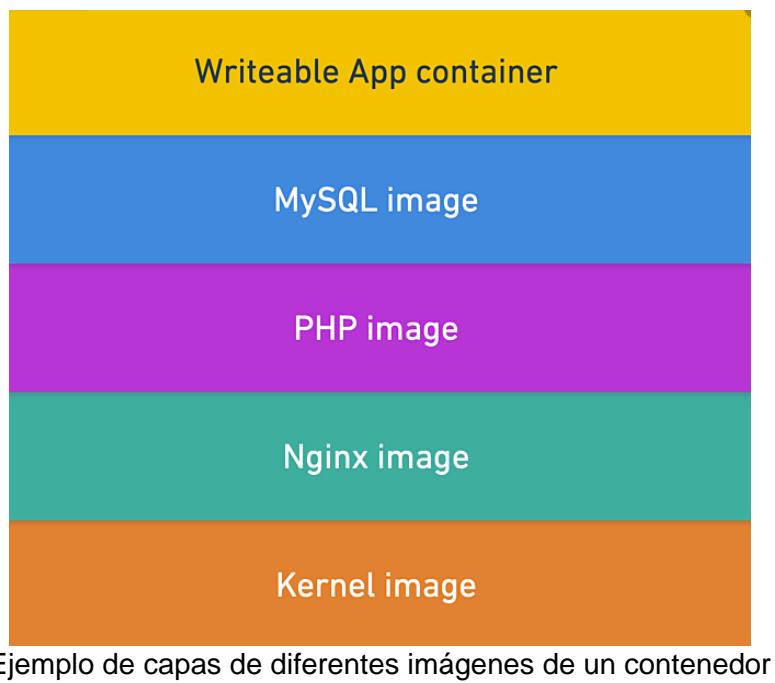
Fuente: Docker

Los Contenedores utilizan las imágenes como fuente para la ejecución de procesos y aplicaciones, más sus correspondientes dependencias.

En cierta forma podríamos asociar una imagen a un snapshot de una aplicación.

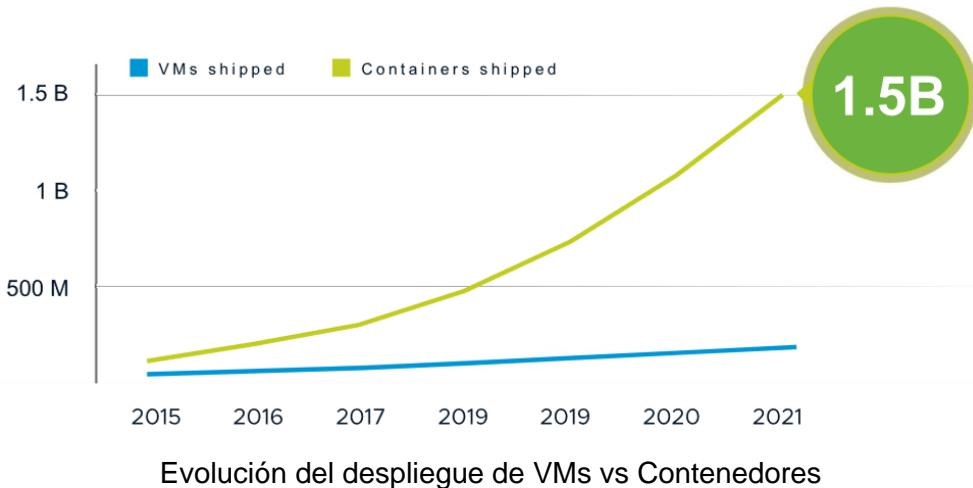
Las imágenes se cargan en cada instancia de Host Container desde un repositorio de imágenes.

Cada aplicación puede estar compuesta por múltiples capas basadas en imágenes. Por ejemplo, una capa puede incluir el SO (Debian, Ubuntu, etc), otra capa la aplicación de servicio (NGINX) y otra capa adicional que sea el código que hace funcionar a la aplicación.



Los contenedores son una excelente opción cuando lo que se busca es eficiencia, pudiendo desplegar la mayor cantidad de aplicaciones en el menor número de instancias de Host Containers.

Por otra parte, las máquinas virtuales entregan una determinada cantidad de recursos de cómputo, almacenamiento y red a una aplicación determinada dando recursos en un ratio 1:1 a las aplicaciones. Como mencionamos anteriormente, las máquinas virtuales están asociadas a aplicaciones Monolíticas y muchas veces el límite y/o problema es la escalabilidad en cuanto a recursos debido a que únicamente pueden crecer de forma vertical.



(Fuente: IDC, Matt Eastwood, IDC Directions, Diciembre 2019)

Veamos una comparativa entre una Máquina Virtual y un Contenedor.

	Máquina Virtual	Contenedor
Recursos	Pesada	Liviana
Sistema Operativo	Cada App tiene su propio SO	Las Apps comparten SO y dependencias
Plataforma	Hipervisor	Máquina Virtual / Cloud
Tiempo de encendido	Segundos - Minutos	Milisegundos - Segundos
Consumo memoria	Consumo de SO + Apps	Consumo optimizado
Aislamiento	Total	Parcial
Asignación de recursos	Configuración VM	Interna (Docker)

Si bien, la base de la modernización de las aplicaciones y la piedra angular de los microservicios son los contenedores, no todas las aplicaciones son idóneas para entornos de contenedores.



“Cuando tu única herramienta es un martillo, todo te parece un clavo”

Determinadas aplicaciones consideradas pesadas como un ERP o aplicaciones que requieran de todos los recursos del Host físico tal vez no sean los mejores candidatos para trabajar en Contenedores.

Las aplicaciones que fueron desarrolladas para escalar en forma horizontal, a través de instancias adicionales, ya sea para incrementar tanto la disponibilidad como el rendimiento son ideales para trabajar en estos entornos y mucho más si son orquestadas a través de sistemas como Kubernetes.

Kubernetes

Si bien la tecnología de contenedores supuso (y supone) un cambio radical en la forma de desplegar aplicaciones, lo cierto es que Kubernetes es la solución que marca la diferencia y que consolida de forma definitiva la distribución de aplicaciones como microservicios.



Kubernetes (también denominado K8s) es un orquestador de contenedores basado en código abierto que permite la distribución, automatización, gestión y escalado de software desplegado en contenedores.

Creado por Google bajo el proyecto en código Borg y lanzado oficialmente en 2014. En Julio de 2015 Google cedió el proyecto a la Cloud Native Computing Foundation, y fue a partir de ese momento cuando Kubernetes comenzó su crecimiento exponencial consolidándose como la plataforma de referencia para cualquier software que pretenda ser Cloud Native.

Como administradores de vSphere, sabemos que el servicio de Clúster de HA y DRS nos permite incrementar la disponibilidad de máquinas virtuales si un Host cae, si necesita balancear las VMs entre los Hosts al encendemos nuevas máquinas, y además si crece la demanda de recursos.

¿Pero qué ocurre si agotamos los recursos de una aplicación? vCenter no tiene el criterio para crear automáticamente una máquina adicional con el objetivo de auto escalar. No solamente que no tiene el criterio, sino que además una máquina virtual está asociada a una aplicación monolítica y, tanto la alta disponibilidad como el escalado, es muy diferente al software desplegado bajo el paraguas de los microservicios.

Sin embargo, Kubernetes sí que ofrece alta disponibilidad, balanceo y auto escalamiento a nivel de software, siempre basado en el fichero *manifest*, que es en donde definimos las reglas del estado deseado.

¿Qué es un estado deseado?

Un término muy utilizado en Cloud Native Apps es el estado deseado o *declarative*.

Anteriormente cuando necesitábamos desplegar o implementar algo, debíamos trabajar en modo imperativo, es decir, definir cómo hacerlo indicando todos los pasos.

Hoy en día disponemos de tecnologías como K8s, que tienen la “inteligencia” suficiente como para que nosotros le digamos cómo lo queremos, y el propio sistema se las arregla para hacer que funcione como se lo hemos pedido.

Como ejemplo, podemos mencionar un sistema básico de calefacción y aire acondicionado en el cual somos nosotros los que ajustamos la temperatura.



Ejemplo de Imperative - Calefacción y Aire manual

En el caso de una configuración de tipo *declarative*, establecemos el resultado y el sistema se las tiene que arreglar para llegar y mantener ese resultado deseado.



Ejemplo de Declarative - Climatizador bizona

Otro ejemplo es, cuando necesitamos trasladarnos a un sitio determinado. Si somos nosotros los que conducimos el vehículo, entonces debemos encargarnos de todo: la conducción, el tráfico, el camino, las señales, la velocidad, etc. Estaríamos en modo imperativo.

Si para llegar al mismo sitio simplemente llamamos a un Uber, entonces nos subimos al vehículo y nos despreocupamos porque sabemos que, de una forma u otra, el conductor se las arreglará para dejarnos en el sitio deseado.

Si bien este capítulo no pretende cubrir Kubernetes ni mucho menos, veamos no obstante algunas pinceladas de la arquitectura y componentes de K8s.

Arquitectura de Kubernetes

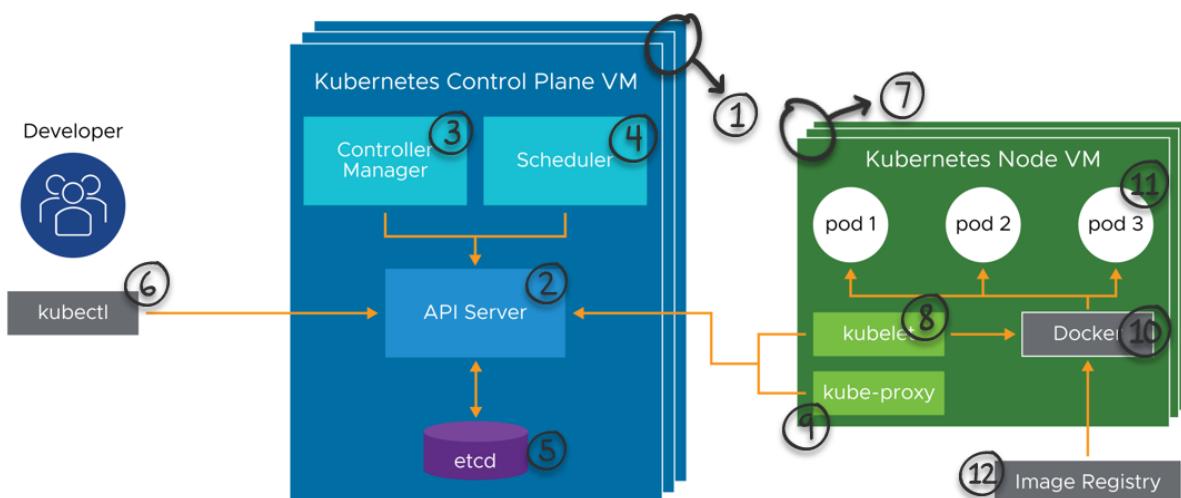
Para cualquier entorno en producción que se precie siempre buscaremos una plataforma que aporte alta disponibilidad, rendimiento, gestión centralizada y a ser posible también balanceo y auto escalamiento.

Un Clúster de Kubernetes aporta todas las funcionalidades descritas anteriormente sobre la base de la orquestación del despliegue de aplicaciones.

Las opciones de despliegue de K8s pueden ser desde un simple equipo portátil, un host bare-metal, un clúster de vSphere on premise o una solución de Kubernetes “llave en mano” en una plataforma de Cloud Pública. Como si fuera poco, estamos ante una solución estándar que hace agnóstica a la plataforma sobre la que funciona y, por lo tanto, eso hace portables las aplicaciones.

Esto supone un hábitat idílico para aplicaciones que necesiten trabajar en entornos de Cloud Híbrida y/o Multi-Cloud.

Arquitectura básica de Kubernetes:



1-Kubernetes Control Plane: El control Plane de Kubernetes puede estar formado por 1 o más instancias, aunque en producción normalmente tenemos 3. Esas instancias pueden ser máquinas virtuales o bien un servicio SaaS en Cloud.

En vSphere with Kubernetes está compuesto por 3 instancias de maquinas virtuales que da servicio al Clúster de Kubernetes.

2-API Server: servicio clave en un Clúster de Kubernetes debido a que tanto los desarrolladores, otras aplicaciones externas y el entorno de comandos kubectl gestionan todos los recursos a través del API de K8s.

3-Controller Manager: Mencionamos anteriormente que definimos el estado deseado a través de un fichero de manifiesto en donde especificamos el estado deseado como el número de réplicas, reglas de afinidad, auto-escalado, etc. El Controller Manager se encarga de hacer cumplir con todas las definiciones de estados deseados que se enviaron K8s.

4-Scheduler: El Scheduler vendría a ser una especie de símil con el VPXD de vCenter debido a que uno de los Master Node del Control Plane de Kubernetes utiliza el scheduler para comunicarse con todos los Nodos de Kubernetes, más específicamente con los Worker Nodes para la monitorización y gestión de cambios.

5-Etcd: Es la base de datos que almacena las configuraciones y los estados deseados de forma persistente.

6-Kubectl: Entorno de línea de comandos que podemos instalar en cualquier equipo para interactuar con el Cluster de Kubernetes desplegando nuevas aplicaciones, aplicando cambios, visualizando estados y demás operaciones.

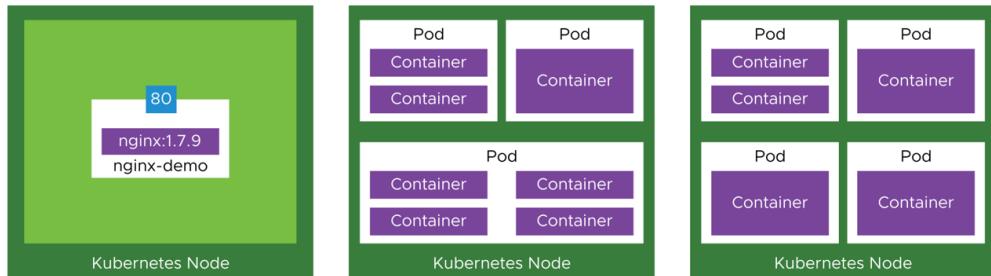
7-Worker Nodes: Son los recursos que se encargan de ejecutar los Pods de Kubernetes en donde se ejecutan las diferentes aplicaciones en contenedores. Estos nodos pueden ser desde equipos físicos, una maquina virtual o bien una instancia de un servicio en Cloud Publica.

8-Kubelet: Es lo que hace funcionar a un Worker Node. Por mencionar una semejanza con algo en vSphere podríamos mencionar al servicio hostd que es el que se encarga de ejecutar todas las operaciones en un Host de ESXi.

9-Kube-proxy: Este es otro servicio fundamental en un worker node debido a que se encarga de gestionar toda la comunicación tanto del Nodo como de todas las instancias de los contenedores con el exterior, así como también las comunicaciones con otros nodos y con el Control Plane de K8s.

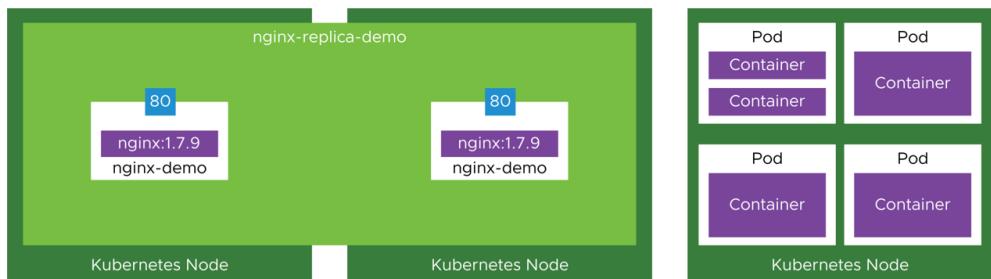
10-Docker: es el motor del servicio local de los contenedores. Si bien en la mayor cantidad de casos se utiliza Docker, Kubernetes es capaz de interactuar con otro tipo de servicios similares.

11-Pod: Objeto lógico que agrupa a uno o más contenedores. Por ejemplo podemos agrupar dos o más aplicaciones dentro del mismo Pod para que se enciendan y apaguen a la vez.

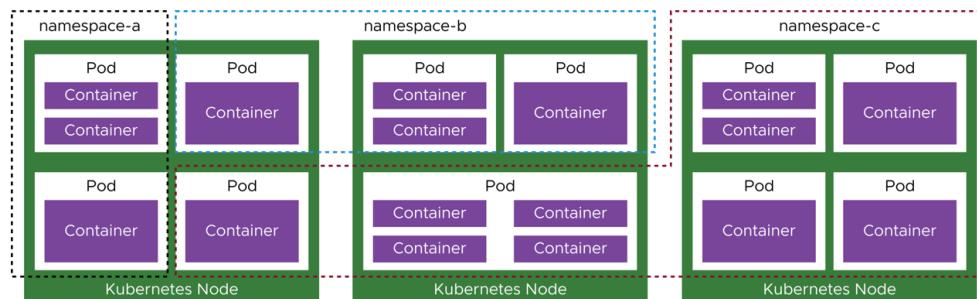


12-Image Registry: servicio externo desde donde se cargan las aplicaciones que se desplegarán en los Pods.

Réplica Set: cuando desplegamos las aplicaciones podemos especificar el número de instancias, o réplicas, que se van a distribuir a través de los diferentes nodos con el fin de escalar la cantidad de recursos e incrementar la disponibilidad.



Namespace: Contexto de recursos de computo y almacenamiento en el cual se deplegarán los diferentes Pods a través de los Nodos disponibles.



Al no ser éste un capítulo dedicado a Kubernetes no profundizaremos más en la arquitectura, ya que el objetivo es ofrecer una vista general de esta increíble solución.

Disponemos en este mismo libro de un capítulo completo dedicado única y exclusivamente a Kubernetes.

No podemos cerrar la parte de K8s sin conocer el caso de uso más curioso.



Pokémon Go fué el mayor despliegue de K8s en Google Container Engine

Más de 500 millones de descargas y más de 20 millones de usuarios activos por día

VMWARE Y LA MODERNIZACIÓN DE APLICACIONES

Ahora que esto de la modernización de aplicaciones, los microservicios, los contenedores y hasta K8s nos suena más familiar, podemos hacer un resumen del camino que tomó VMware hace unos años orientándose a ampliar su plataforma para dar soporte a Cloud Native Apps.

Hoy en día, con vSphere 7 y Tanzu, es posible desplegar Kubernetes de forma nativa en vCenter y dar soporte a los desarrolladores, los devops y los administradores de sistemas ofreciendo un entorno “nativo” tanto para aplicaciones *legacy* funcionando en máquinas virtuales como también un entorno nativo para aplicaciones modernas y todo lo que conlleva.

Los servicios de red y seguridad se pueden gestionar, escalar y automatizar con NSX-T incluso incorporando servicios de terceros.

La evolución de la monitorización con vRealize Operations Manager y la automatización con vRealize Automation incorporan a estos nuevos invitados a la fiesta que son los desarrolladores ofreciéndoles un hábitat nativo.

Lo más destacable de todo es la oferta de una plataforma completa para dar soporte al hardware virtual más utilizado en entornos privados y las cloud native apps, ya sea en infraestructuras de cloud Privadas, Públicas y también Híbridas.

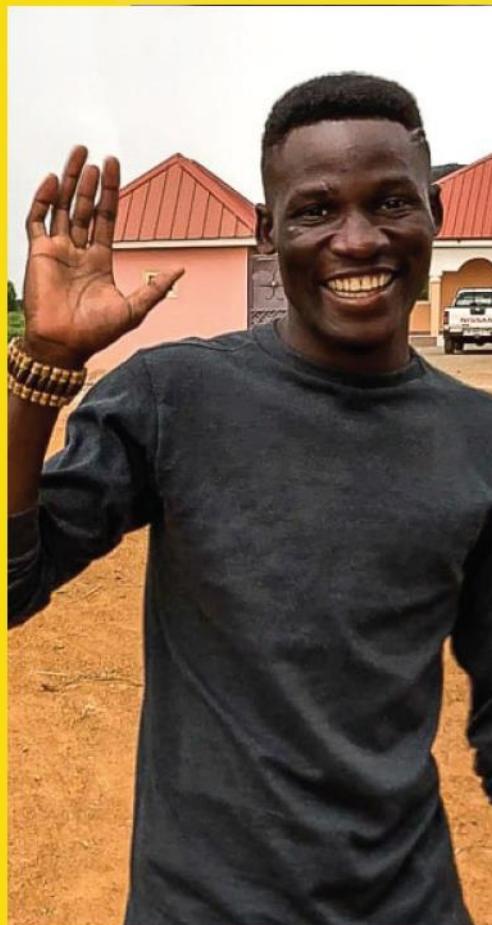
Ahora sí, Any App, Any Device, Any Cloud.

Su nombre es **Basil David** y tiene 25 años. Conoció NASCO Feeding Minds en 2012, cuando estudiaba en el colegio. Pudo **usar por primera vez un ordenador** y ver las oportunidades que ofrecía.

Cuando terminó Senior High School, y después de **aprender informática** en una de las aulas de NASCO Feeding Minds, decidió estudiar Artes Visuales para después emprender. Finalmente creó un **estudio propio**.

Su negocio consiste en realizar **diseños mediante un ordenador** y luego plasmarlos en productos.

NASCO Feeding Minds cambió muchas cosas en su vida, le dio **conocimiento informático** y le enseñó a manejar un ordenador. Así obtuvo las bases para terminar siendo un **diseñador gráfico**.



**CAMBIA
LA
HISTORIA**

NASCO
FEEDING
MINDS



VDC+
Business continuity

¿Tu negocio tiene aplicaciones críticas?



Innovador, revolucionario y exclusivo.

Kio Networks es el único Data Center con un servicio de estas características: última tecnología para ofrecer un sistema IT sin incidencias.



Pago por consumo

Ofrecemos contratación flexible por recursos, sin contratos mínimos. Contrata lo que necesites en cada momento y acelera los tiempos de entrega de tus servicios. Consume tus nuevos recursos de forma inmediata.



Sistema automatizado

Restauración de tus servicios en tiempo mínimo en caso de desastre, sin intervención humana.



Mayor rendimiento y adaptabilidad

Contratación 100% flexible. Sin contratos mínimos o permanencias, con escalabilidad sin límites.



Datos replicados, recuperación inmediata

Los datos se escriben de forma síncrona en dos centros de datos de Kio. El RTO es inferior a 5min con restauración automática en el site remoto.



Máxima seguridad y tranquilidad

Tu información y tus aplicaciones disponibles en dos ubicaciones de forma simultánea y automática.



Capítulo 10

Ecosistema CLOUD NATIVE, Impacto, Presente y Futuro



Ivan Camargo

@ivanrcamargo

ECOSISTEMA CLOUD NATIVE, IMPACTO, PRESENTE Y FUTURO.

INTRODUCCIÓN



Estoy seguro de que, si has estado inmerso en la industria de la tecnología, has escuchado palabras como Cloud Native, Microservicios, contenedores, entre muchas más. Pues bien, como es costumbre y ya durante varias décadas la tecnología está en constante cambio, es como un río que todo el tiempo se mueve, avanza y trae cosas nuevas a su paso.

Escuchamos cifras impactantes como por ejemplo que la sociedad ha generado más datos los últimos años 2 a 4 años que en toda la historia

de la humanidad y así el impacto que traerá la inteligencia artificial en los empleos actuales y profesiones, sumado a enormes desarrollos tecnológicos como la computación cuántica que se espera que va a resolver problemas de la humanidad que no imaginamos, las redes 5G con las cuales los desafíos actuales de conectividad podrían solucionarse, entre muchos otros temas de tendencia en la industria de la tecnología.

El mundo ha cambiado y ha llevado a las grandes compañías de tecnología a repensar como poder solucionar los desafíos que tiene una sociedad tan hiperconectada y exigente a la vez, la cual espera que la tecnología pueda satisfacer. Antes de entrar en materia quiero compartir algunas anécdotas personales para que puedas conocerme un poco mejor.

En los últimos 5 años de mi carrera tuve el inmenso honor de trabajar en 2 de las compañías que se han tomado muy en serio esos desafíos. Quiero arrancar por comentar que una de estas empresas fue justamente VMware. Aún recuerdo cuando pisé por primera vez las oficinas de Palo Alto, CA y sentí como un latino de un barrio de clase media de Bogotá, Colombia, pisaba ese lugar destinado a mentes capaces y a talentos de otra galaxia. Pues bueno ahí estaba yo, y durante varias ocasiones tuve la oportunidad de compartir con personas muy talentosas de todo el mundo.

Antes de entrar a VMware trabajé por años en algunos partners donde logré algunas cosas que recuerdo muy especiales, como la apertura del primer capítulo de la comunidad de VMUG en el país, ser parte de la comunidad de vExperts a nivel mundial y lo más importante para mí; las personas que conocí en ese camino. (Ariel Sanchez, Sergio Muñoz, Larry Gonzalez, Patricio Cerda, Luis Consistre, Jorge Torres, Kyle Murley, Stalin Peña, Dave Morera, Eduardo Molina, Elver Sena, Valdecir, y muchos más) Para mí fue todo un privilegio, conocer personas tan talentosas y tan buena onda.

Fueron un poco más de 2 años de mi vida en VMware llenos de desafíos muy interesantes. Voy a aprovechar este espacio de compartir con la comunidad, una anécdota muy bien guardada que tenía. Desde que entre a VMware Uno de mis objetivos profesionales fue ser parte del programa de CTOA. Los CTO Ambassadors son miembros de un pequeño grupo de tecnólogos colaboradores individuales experimentados y talentosos. Son ingenieros de sistemas de preventa (SE), gerentes de cuentas técnicas (TAM), consultores de servicios

profesionales, arquitectos e ingenieros de servicios de soporte global. Los embajadores de la oficina de CTO ayudan a garantizar una estrecha colaboración entre I + D y los clientes de VMware para abordar los problemas actuales de los clientes y las necesidades futuras de la manera más eficaz posible. algunas semanas después de salir de VMware a mi siguiente aventura (IBM), me encontraba en un evento de tecnología en San Diego, CA y me encontré con un amigo cercano el cual muy emocionado saco su teléfono celular del bolsillo para compartirme algo, "había sido seleccionado para el programa de CTOA".

Ya podrán imaginarse lo que sentí esa noche, fue una mezcla de sentimientos, rabia, alegría, frustración, definitivamente ese día aprendí que muchas veces pierdes para ganar, por mi cabeza pasaron muchas emociones para ser honesto. Soy de las personas que se arriesgan y creo que cuando más arriesgas más probabilidades tienes de ganar y de aprender.

Luego de esta pequeña historia personal y como apertura a lo que quiero compartir, ahora trabajo en IBM haciendo muchas cosas interesantes con clientes de la región y sobre todo con la misión de acelerar sus procesos de modernización y adopción de tecnologías nativas de nube.

Ahora muchos de ustedes dirán... y el preámbulo ¿por qué razón? ¿Qué tiene esto que ver con el capítulo de este libro asociado en donde esperaba que me hablaran del ecosistema cloud native? Pues bien, acá comienza esta historia.

Desde hace unos 12 o 13 años estuve trabajando en la adopción de la virtualización del cómputo en el mercado. Seguro algunos de ustedes vivieron esa época en donde a mucha gente le costaba entender que ya no necesita decenas de servidores para correr sus aplicaciones y contrario a esto la virtualización le permitía tener en unos cuantos servidores, características muy interesantes de disponibilidad, rendimiento y lo más importante disminuir el consumo inútil de energía que aumentaba la huella de carbono en el planeta. Era difícil pensar en eso y con el paso de los años la virtualización creció en números acelerados, logrando llevar el mismo principio a otras capas del centro de datos como lo fueron el almacenamiento, la seguridad y las redes.

Pues bien, guardando las proporciones y el nivel de impacto, estamos enfrentados a una realidad similar en la forma en cómo surgen formas distintas de construir y empaquetar las aplicaciones. La era de la nube, el desarrollo tecnológico y la entrada de gigantes como AWS, Azure y Google, nos dieron la oportunidad de divertirnos algunos años más.

Personalmente creo que la virtualización fue un habilitador clave en la computación en la nube que hoy conocemos. En este capítulo tratare de hacer algunas reflexiones del impacto que la computación nativa de nube ha traído al mercado y lo que viene en el futuro.

En esta capítulo vas a encontrar 4 aspectos que considero relevantes para comprender el ecosistema y la relevancia que tienen en el presente y el futuro de la tecnología nativa de nube:

- Contexto Cloud Native
- Proyecto Cloud Native OpenSource
- Plataformas Cloud Native
- Futuro Cloud Native

CONTEXTO CLOUD NATIVE

Creo que existen varias interpretaciones de lo que significa Cloud Native, muchos dirán que una base de datos en PaaS en cualquiera de los cloud providers es cloud Native, o incluso que el proceso de instalar una aplicación en una máquina virtual en un proveedor de nube es cloud native. Para este caso voy a tomar la definición textual de la Fundación de computación nativa de nube (CNCF) que lo describe como:

“Técnicas que permiten sistemas altamente desacoplados que son resistentes, manejables y observables. En combinación con una automatización robusta, permiten a los ingenieros realizar cambios de alto impacto con frecuencia y predicción con un mínimo trabajo.”

Me encanta la definición porque, en primer lugar, no describe tecnologías particulares, por el contrario, se enfoca en técnicas/patrones que al final en mi opinión son el centro de la computación nativa de nube. Mis compañeros en algunos capítulos de este libro han ido abordando algunos temas de historia muy relevantes para el ecosistema de cloud native, por lo cual por efectos de practicidad y sencillez voy a omitir en esta parte. Solo quiero resaltar algunos hitos muy relevantes.



La organización CNCF (**Cloud Native Computing Foundation**) nació en 2015 como resultado en principio de una donación realizada por Google de uno de sus principales proyectos y de mayor relevancia (Kubernetes) que impactó la manera en cómo Google empezó a replantear la manera de construir sistemas distribuidos altamente escalables, disponibles y resilientes. Desde allí se marca un hito importante en la computación nativa de nube, principalmente porque nacía la mayor asociación tecnológica de los últimos

años en el mercado de la tecnología.

Varios gigantes de la tecnología estuvieron de acuerdo en la necesidad de iniciar un proceso de democratización tecnológica logrando que los esfuerzos de las distintas compañías pudieran estar disponibles para la comunidad en la modalidad de proyectos en donde una o varias Compañías ponían a disposición de la organización recursos económicos, recursos técnicos para lograr trabajar en sinergias que permitieran establecer estándares abiertos.

Ahora que hablo de esto puedo recordar en un evento de Openstack Summit en el que pude participar, entrando en una sala en donde nacían los posibles proyectos OpenSource y literalmente vi como abrían un archivo editable por todos, en donde cada uno se encarga de algunos temas específicos del proyecto o grupo de interés que iniciaba, la participación y la cooperación que sentí ese día en esa sala fue impresionante.

CNCF nació bajo la cobertura de **Linux Foundation** y fue de apoco logrando que las principales Compañías de investigación y desarrollo en tecnologías cloud native se unieran con el propósito de establecer un marco de gobierno para el mismo y así evitar que una compañía pudiera tomar el control de las tecnologías y proyectos que se desarrollaban. Por

otro lado, es super interesante que Compañías que típicamente adquieren tecnología, también tuvieran un espacio en el mismo para poder participar de las discusiones y aportar la experiencia en campo, esto es realmente apasionante.

CNCF cuenta con un marco de gobierno en mi opinión bastante bien estructurado con 5 principales actores:

- **TOC Comité de Supervisión Técnica**
- **Junta Directiva**
- **Embajadores**
- **Personal**
- **Comunidad de usuarios finales**

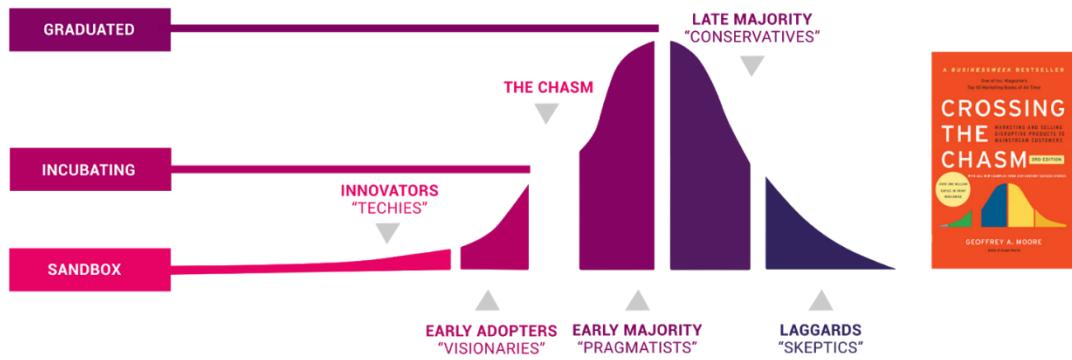
Cada uno de ellos tiene propósitos y objetivos específicos, pero lo interesante en mi opinión detrás de este enfoque más allá de ser una fundación que monopoliza, es regular y observar el desarrollo de las tecnologías cloud native, con el propósito de unir esfuerzos en pro de “proyectos” de mayor impacto en donde las Compañías ponen sus recursos de investigación e ingeniería en proyectos open source que logran luego monetizar de diferentes formas. El Open Source del siglo XXI no se basa en adolescentes rebeldes en sus habitaciones trabajando en tecnologías que nadie controla y que eventualmente funcionan. Organizaciones como CNCF logran unir con un propósito común, las Compañías más importantes y relevantes en tecnologías cloud native con una estructura, un marco de gobierno, y un objetivo superior.

CNCF tiene algunos críticos por su puesto y algunas acciones que han llamado la atención durante estos más de 5 años. Por ejemplo, recientemente un anuncio muy importante por parte de Google acerca de uno de los proyectos con un potencial en el futuro muy interesante se trata de Istio. Istio nació luego del trabajo de Google, IBM y Lyft. Es un proyecto que cubre las capacidades de malla de servicio (Service Mesh) en las aplicaciones y arquitecturas de microservicios. Si bien se esperaba que al ser un esfuerzo y proyecto OpenSource lo natural era que se hubiera donado como parte de CNCF, Google tomó una decisión un poco diferente y fue crear la fundación Open Usage Commons. Las reacciones de la industria fueron variadas, porque lo natural era haber usado a CNCF para donar el proyecto. Las especulaciones fueron muchas, sin embargo, creo que lo importante acá es que las grandes empresas de la tecnología hoy son conscientes del impacto en sus tecnologías y la importancia de un marco abierto para garantizar a sus clientes la independencia y flexibilidad, sumado a su interés de aportar en el futuro de sus negocios digitales que dependen del buen uso de la tecnología. Los invito a ver más detalles acerca del marco de gobierno, los proyectos, las empresas que hacen parte del ecosistema de CNCF, entre otros detalles muy relevantes para el entendimiento más detallado del ecosistema.

PROYECTOS CLOUD NATIVE

Como mencionaba en la sección anterior, en el ecosistema cloud native hablamos de proyectos los cuales son categorizados en diferentes tipos y dependiendo de su sostenibilidad, nivel de madurez, adopción y compromiso de múltiples organizaciones:

Este es el diagrama “Crossing the chasm” podemos entender el nivel de madurez de cada proyecto.



“Los proyectos de CNCF tienen un nivel de madurez de “sandbox”, “incubating” o “graduated”, que corresponde a los niveles de innovación, Adopción temprana de acuerdo con el diagrama “Crossing the Chasm”. El nivel de madurez es una señal de CNCF sobre qué tipo de empresas deberían adoptar diferentes proyectos. Los proyectos aumentan su madurez al demostrar su sostenibilidad en el comité de supervisión técnica de CNCF (TOC): que analizan su adopción, tasa saludable de cambios y compromisos de múltiples organizaciones, que por supuesto han adoptado el código de conducta de la CNCF; y han logrado y mantenido la insignia de mejores prácticas” <https://www.cncf.io/projects/>

Para explicar un poco mejor este punto, voy a tomar 2 ejemplos claros del marco de gobierno, nivel de compromiso de las empresas, contribución y otros aspectos relevantes en un proyecto de CNCF:

Kubernetes

Kubernetes es un proyecto donado por Google en el 2015. Kubernetes es una de las plataformas de gestión de contenedores líderes en el mercado. Desde ese momento varias Compañías empezar a aportar de manera relevante y con un interés muy particular en ayudar a Kubernetes a ser el estándar por defecto en gestión de contenedores. Proveedores de nube como el mismo Google, Azure y AWS, sumado a un extenso ecosistema de Compañías interesadas en aportar en el desarrollo tecnológico de Kubernetes como VMWare, RedHat, Cisco, Suse, entre otros.

Quiero ilustrar el nivel de involucramiento y compromiso en un proyecto de CNCF.

Lo primero a mencionar es que ustedes pueden ir en cualquier momento a revisar algunas de las cifras que voy a compartir las cuales pueden tener modificaciones por efectos de tiempo.

<https://k8s.devstats.cncf.io/d/12/dashboards?orgId=1&refresh=15m>

Contribucion Kubernetes desde sus inicios:

The screenshot shows a table titled "Kubernetes Companies statistics (Contributions, Range: Last decade), bots excluded". The table has columns for Rank, Company, and Number. The data shows the top 10 companies contributing to Kubernetes over the last decade, with Google at the top and CNCF ranked 10th.

Rank	Company	Number
	All	2263244
1	Google	802651
2	Red Hat	316588
3	VMware	138101
4	Independent	77810
5	IBM	56446
6	Microsoft	45932
7	Huawei	43543
8	Fujitsu	18251
9	Intel	14442
10	CNCF	14387

Contribucion Kubernetes ultimo año:

The screenshot shows a table titled "Kubernetes Companies statistics (Contributions, Range: Last year), bots excluded". The table has columns for Rank, Company, and Number. The data shows the top 10 companies contributing to Kubernetes in the last year, with Google at the top and NEC ranked 10th.

Rank	Company	Number
	All	516075
1	Google	120651
2	VMware	67910
3	Red Hat	29844
4	Independent	23263
5	Microsoft	19673
6	IBM	13322
7	The Scale Factory	7995
8	SUSE	6362
9	Kubernetes	5982
10	NEC	4672

Si tomamos los datos y hacemos un análisis muy sencillo el resultado es:

El 68% de las empresas que han aportado a Kubernetes son:

All	2263244		Porcentaje de Contribución
1	Google	802651	35%
2	Red Hat	316588	14%

3	VMware	138101	6%
4	Independent	77810	3%
5	IBM	56446	2%
6	Microsoft	45932	2%
7	Huawei	43543	2%
8	Fujitsu	18251	1%
9	Intel	14442	1%
10	CNCF	14387	1%
			68%

De ahí para adelante el otro 32% este combinado en más de 200 Compañías los últimos 10 años. Me parece muy relevante el aporte independiente que se estima en un 3% para el momento en que escribo este libro (agosto 2020).

Si tomamos una muestra del último año los resultados nos muestran varias cosas interesantes.

All	516075		Porcentaje de contribución
1	Google	120651	23%
2	VMware	67910	13%
3	Red Hat	29844	6%
4	Independent	23263	5%
5	Microsoft	19673	4%
6	IBM	13322	3%
7	The Scale Factory	7995	2%
8	SUSE	6362	1%
9	Kubermatic	5982	1%
10	NEC	4672	1%
			58%

Es muy interesante ver Compañías como VMware, RedHat y Microsoft tomando un liderazgo importante en contribución. Personalmente para mi ver esto me llena de interés y sobre todo ver como se unen esfuerzos en pro de un bienestar superior. Suena romántica la frase, pero realmente trabajar como un equipo, encontrando la forma de que cada uno de sus individuos sea recompensado es genial. Cada empresa que aporta en estos proyectos hace inversiones anuales de 350.000 dólares sumado a los recursos técnicos y lo que considero más importante, permitir que una organización como CNCF normalice y asegure marcos abiertos en el desarrollo de la tecnología.

Lo interesante que veremos más adelante es como cada uno de los que contribuye está tomando esa contribución para el desarrollo de sus soluciones. La monetización es la clave de este enfoque.

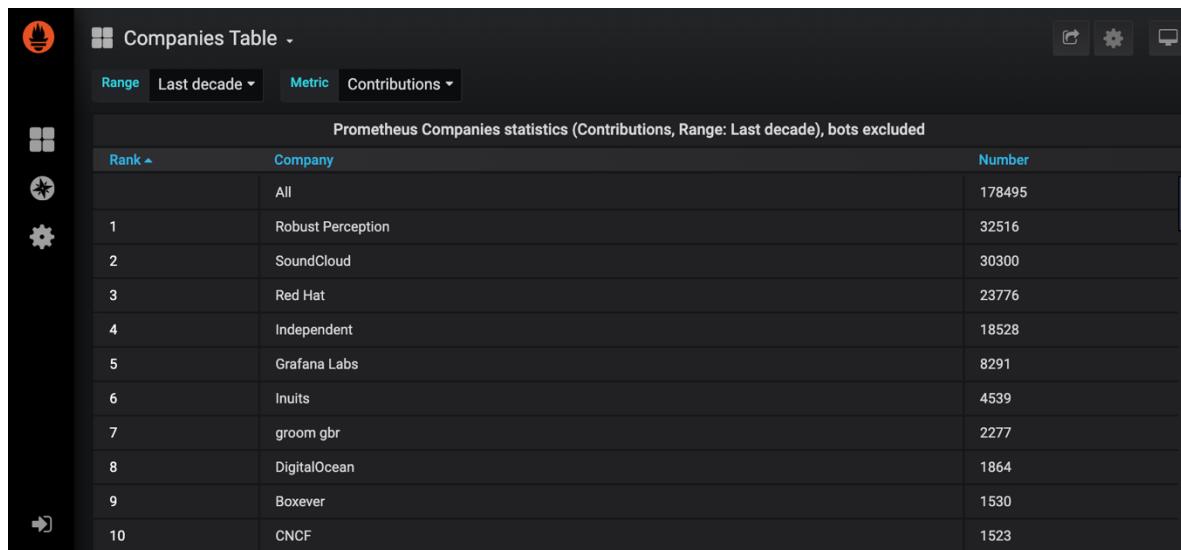
Prometheus

Prometheus es uno de los proyectos graduados en CNCF el cual tiene una definición corta pero muy clara: “De las métricas a la información”. Las Compañías han estado teniendo múltiples herramientas por las que pagaron miles de dólares las cuales ante una falla o la necesidad de encontrar la raíz de un problema son bastante ineficientes.

Prometheus es uno de los proyectos OpenSource más relevantes asociado a la práctica o disciplina de Observabilidad, el monitoreo hoy es uno de los tópicos más relevantes para garantizar la disponibilidad de las aplicaciones y sistemas distribuidos. Las nuevas disciplinas de gestión y operación de aplicaciones requieren de personas, herramientas y tecnología para tal fin.

Vamos a revisar los datos:

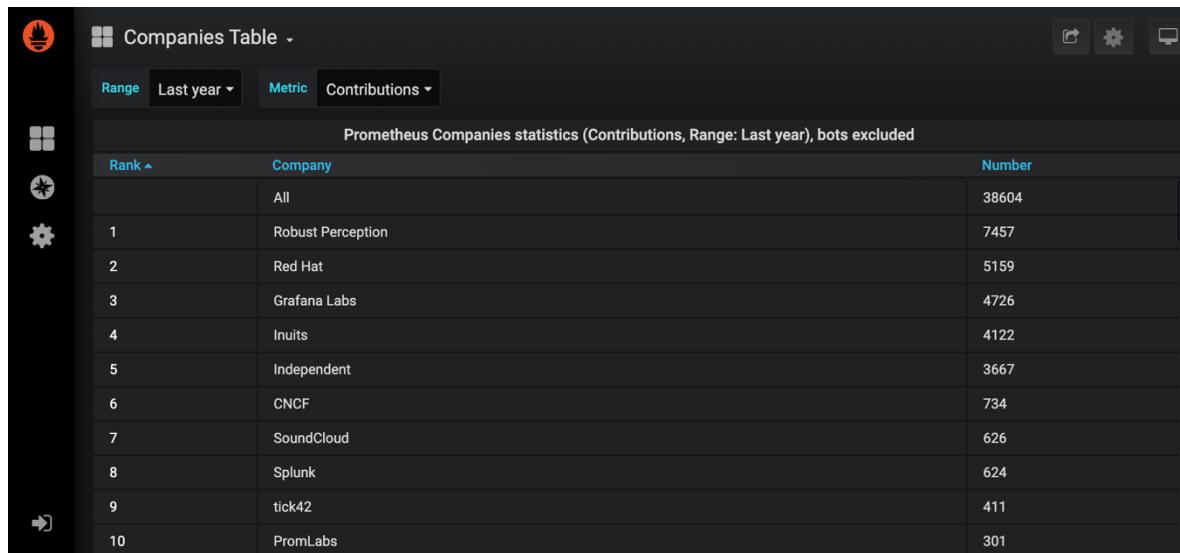
Contribución Prometheus desde sus inicios:



The screenshot shows a table titled "Prometheus Companies statistics (Contributions, Range: Last decade), bots excluded". The table has three columns: Rank, Company, and Number. The data is as follows:

Rank	Company	Number
	All	178495
1	Robust Perception	32516
2	SoundCloud	30300
3	Red Hat	23776
4	Independent	18528
5	Grafana Labs	8291
6	Inuits	4539
7	groom gbr	2277
8	DigitalOcean	1864
9	Boxever	1530
10	CNCF	1523

Contribucion Prometheus el último año:



The screenshot shows a table titled "Companies Table" with the following data:

Rank	Company	Number
	All	38604
1	Robust Perception	7457
2	Red Hat	5159
3	Grafana Labs	4726
4	Inuits	4122
5	Independent	3667
6	CNCF	734
7	SoundCloud	626
8	Splunk	624
9	tick42	411
10	PromLabs	301

Este caso es un poco diferente, porque la mayor parte de la contribución esta distribuida, en este caso Robust Perception, RedHat y Grafana Labs son las principales Compañías en contribución. Pero vemos tambien otros casos un poco desconocidos y extraños.

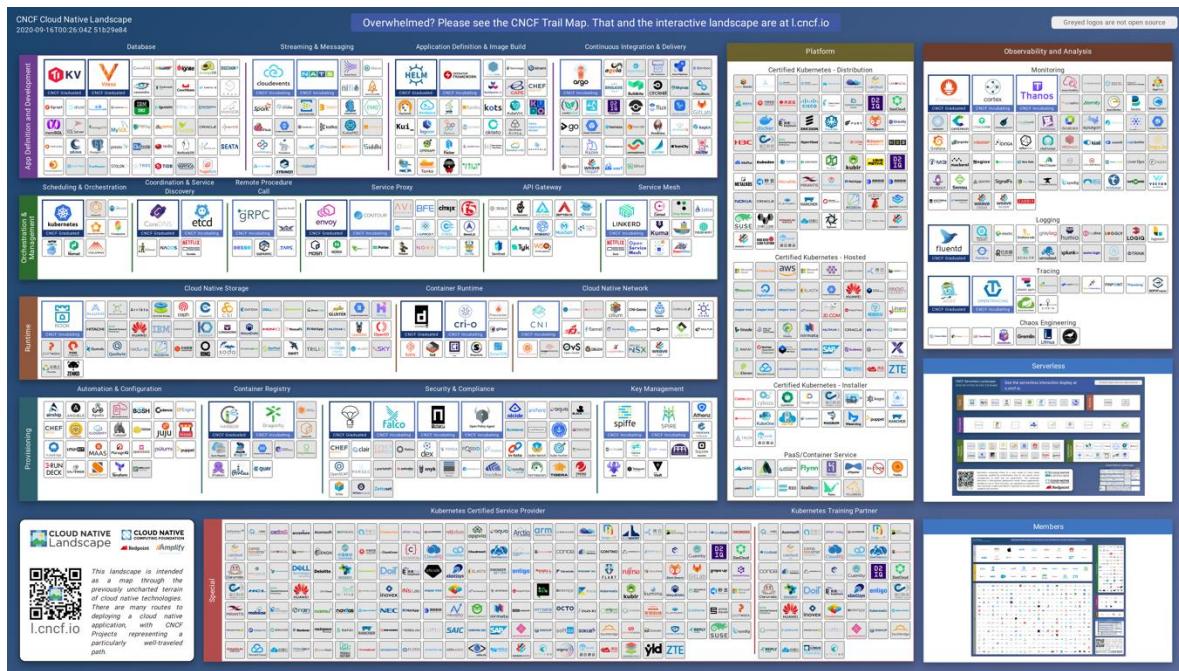
Algunos pensamientos generales de esta sección:

- El enfoque abierto en cloud native, más allá de ser una ilusión es una realidad que ha demostrado el potencial y las diferentes formas de monetización que existen en el mercado.
- La madurez y la sostenibilidad de las tecnologías y su evolución definitivamente son un pilar en el ecosistema cloud native.
- El enfoque en proyectos e iniciativas más que en productos cloud native acelera la innovación y el respaldo de la comunidad.
- El impacto que las tecnologías cloud native ha tenido los ultimos 5 años ha llevado a que enormes corporaciones tecnologicas redefinan sus estrategias de productos y soluciones.
- El ecosistema cloud native esta mucho más organizado y estructurado de lo que pensabas.

En la siguiente sección hablaremos un poco de como algunas de las empresas de tecnología toman sus esfuerzos en el ecosistema abierto para el desarrollo de sus productos y soluciones.

PLATAFORMAS CLOUD NATIVE

Es importante entender la naturaleza de los proyectos del ecosistema cloud native para entender con claridad la forma en que algunas de las Compañías en el mercado recuperar su inversión en el mercado open source.



Esta imagen ha sido usada para todo tipo de memes en la web, personas tratando de explicar la complejidad, otros la utilizan para propósitos de evasión por desconocimiento, entre otros. Sin embargo, en mi experiencia fue importante y valioso para interpretar correctamente lo que se quiere trasmitir con la misma, algunos pensamientos:

- Comprender la diversidad me permitió entender que más del 40% de las Compañías y proyectos ahí no existían hace 3 o 4 años.
- Interpretar que los proyectos tienen esfuerzos compartidos por diferentes fabricantes.
- Organizar algunos de los componentes relevantes del ecosistema en función de prácticas y patrones.
- Tener una fuente inteligente y no comercial para identificar soluciones, componentes y herramientas del ecosistema resulta valioso en estos tiempos.
- Dejar la ignorancia e ir viendo las cosas nuevas que están en etapas tempranas de desarrollo. (incubación) dan una perspectiva más enriquecida.

Impacto

Voy a enfocarme en 3 plataformas cloud native que considero son muy relevantes y de gran potencial en el presente y el futuro del ecosistema. Estas plataformas integran varios de los proyectos open source que hemos estado discutiendo hasta el momento, adicionando por supuesto capacidades y componentes adicionales, que de acuerdo con sus modelos de comercialización tienen algunas ventajas y beneficios.

Voy a tratar de dar una revisión muy general y en alto nivel de cada una de las plataformas, enfocándome en lo que considero de valor. No voy a hacer ninguna crítica al respecto, en cada uno de los casos voy a enfocarme en las cosas que considero interesantes y positivas. Lo que viene obedece a pensamientos y opiniones mías que no comprometen la opinión o el punto de vista de mi empleador actual.



Antes de entrar al detalle de cada uno veremos el reciente reporte de la consultora Forrester en donde analiza las principales plataformas de contenerización multicloud.

Las tres (3) plataformas sobre las cuales haré una introducción son **Google Anthos**, **RedHat Openshift** y **VMware Tanzu**. Las 3 plataformas hacen parte del reporte de Forrester.

La razón por la que hablare un poco de ellas es básicamente por interés personal y experiencia. Me hubiera encantado entrar a revisar en detalle otros jugadores que admiro como lo son **Rancher Labs**, **Platform9**, quien siendo bastante más pequeños que sus competidores, han logrado hacer cosas muy interesantes.

La revisión que vamos a dar está enfocada en características técnicas de arquitectura, componentes y puntos que me considero interesantes. No voy a incluir ningún tipo de información comercial ni de modelos de distribución.

Una plataforma cloud native no tiene una definición particular, sin embargo, lo que persigue es poder entregar todos los componentes y herramientas necesarias para poder desarrollar

y gestionar aplicaciones modernas (cloud native). Muchos creen que lo único que se requiere es un clúster de Kubernetes y no es así.

GOOGLE ANTHOS



Google como padre fundador de Kubernetes tiene una voz de autoridad en el mercado cloud native. Google ha planteado múltiples cambios en la historia de la humanidad y en cuestiones de tecnología definitivamente es relevante. Desde la creación de Kubernetes hasta el planteamiento de prácticas y disciplinas de operación modernas de aplicaciones (SRE), lo convierten en un jugador importante que tiene muchos ojos encima frente a sus posturas y estrategia de productos tecnológicos.

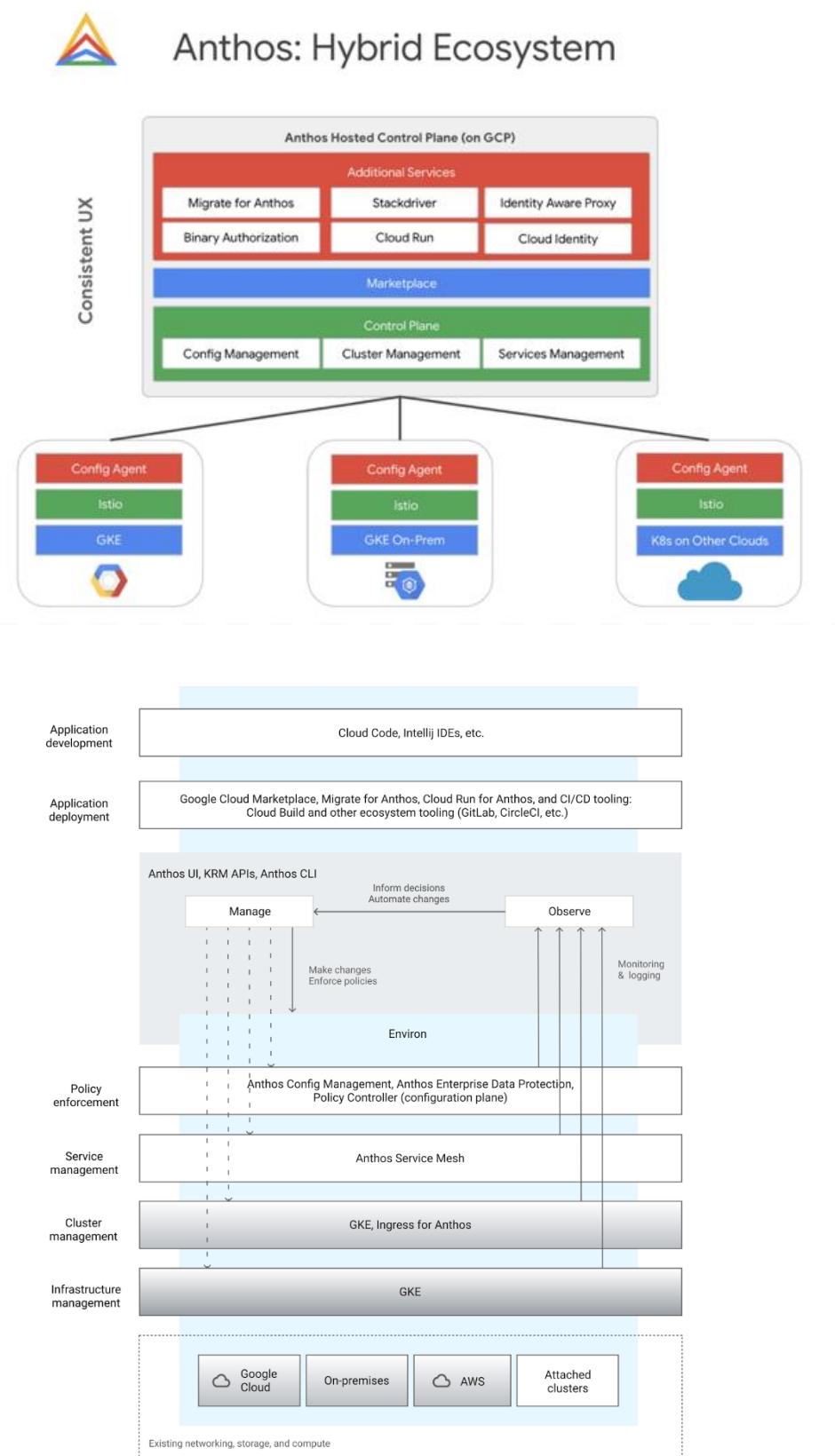
Desde la nube a la tierra

Google inicio la comercialización de Kubernetes con la oferta de GKE (Google Kubernetes Engine) la oferta de Kubernetes como un servicio administrado en Google Cloud. En mi opinión una de las mejores ofertas de Kubernetes administrado sin duda, como resultado de la fuerte adopción de Kubernetes, entendió que el mercado y las necesidades del ecosistema empresarial corporativo requería de un planteamiento distando en la arquitectura de sus aplicaciones e infraestructura.

Recuerdo compartir por redes sociales la demostración que hizo hace unos años en uno de sus eventos donde compartieron como habilitaban capacidades de migración de máquinas virtuales desde vSphere hacia GKE. Convirtieron una máquina virtual con una aplicación sencilla a un contenedor que se ejecutaba en GKE. Definitivamente era la entrada y el cambio de estrategia para acelerar la transformación del mercado empresarial.

Es por esto por lo que en el 2019 lanza su estrategia de nube híbrida conocida como **Google Anthos** en donde habilitaba en principio la posibilidad de llevar su servicio de Kubernetes GKE en un modelo de infraestructura multicloud. Es decir, dar la capacidad de poder correr el servicio administrado de GKE por fuera de las fronteras de Google Cloud, permitiendo que se pudiera ejecutar en los centros de datos en las premisas de sus clientes o en diferentes proveedores de nube. Adicional a esto agregando una serie de capacidades adicionales a GKE, como la gestión de máquinas virtuales (aplicaciones tradicionales) y otros aspectos que a continuación explicaré.

Diagrama de arquitectura de Google Anthos:

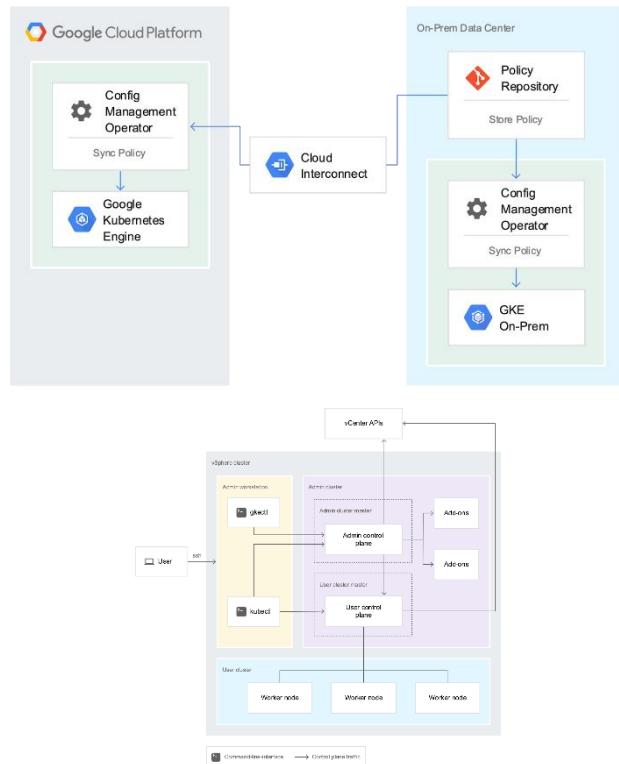


<https://cloud.google.com/anthos/docs/concepts/overview>

Desde abajo hacia arriba encontraremos ya no solo la posibilidad de ejecutar diferentes aplicaciones en Google Cloud, también vemos la posibilidad de llevar el stack de Google Anthos en las premisas del cliente, así como AWS e incluso sobre otros tipos de clústeres de Kubernetes en el modelo de Attached Clusters.

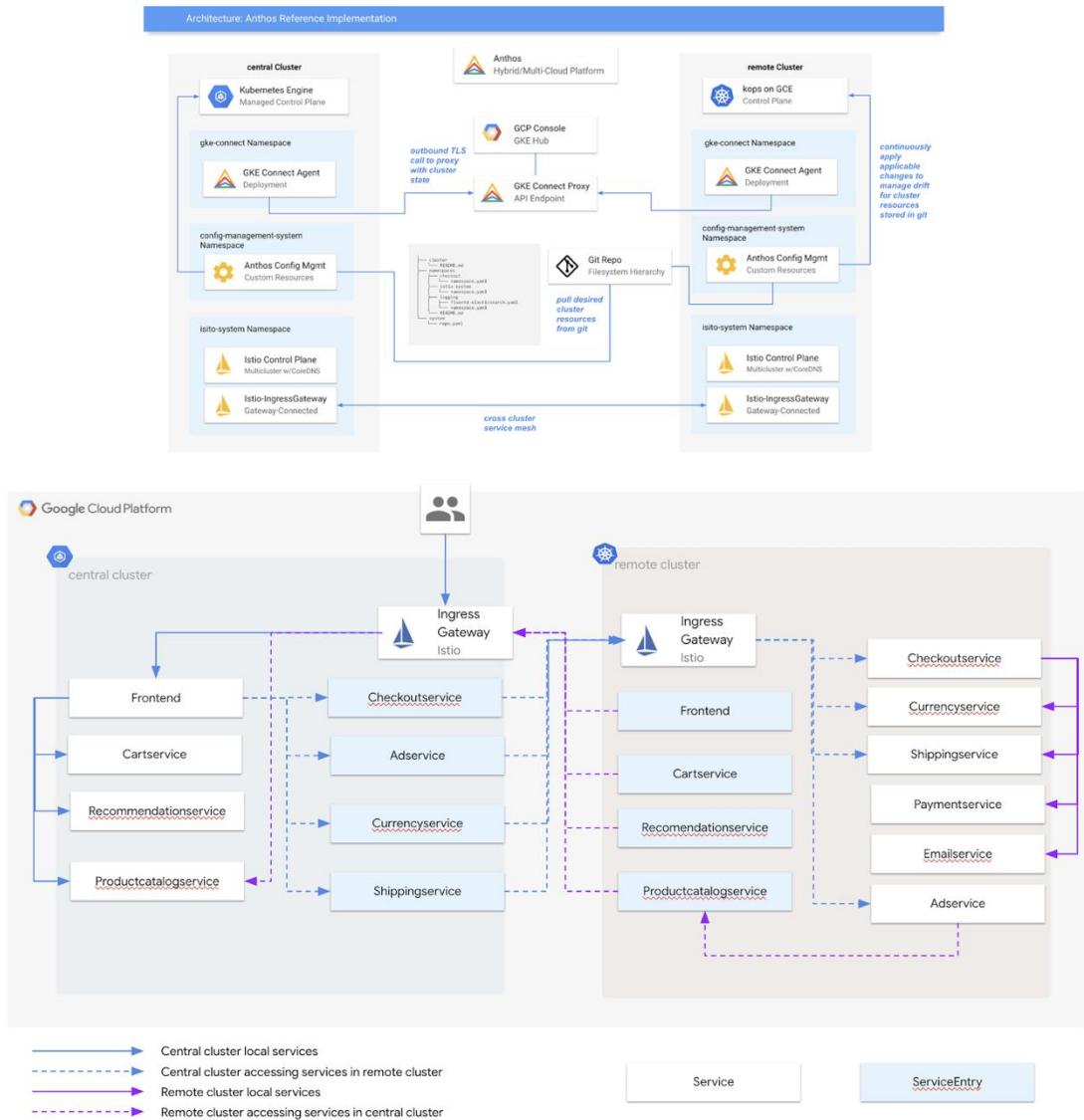
Subiendo en el stack de Google Anthos, encontramos por supuesto las capacidades para entregar GKE en modelo de infraestructura heterogénea en donde el plano de control se encuentra alojado en Google Cloud desde el cual se extienden algunas capacidades de comunicación y gestión para lograr un control centralizado con clústeres de GKE descentralizados en múltiples plataformas.

Un ejemplo de la arquitectura de Google Anthos en donde se ilustra la gestión desde el plano de control de Google Cloud y el servicio de Google Anthos corriendo en premisas de un cliente. Para este caso el clúster de GKE de Google Anthos se ejecuta en una infraestructura de vSphere:



El clúster de GKE Onprem es técnicamente el mismo que se ejecuta en Google Cloud con la diferencia en la infraestructura subyacente es VMware vSphere y algunas capacidades propias de la infraestructura en Google Cloud, por supuesto. Esta infraestructura requiere de algunos componentes en ejecución local, para asegurar la correcta administración, visibilidad y gestión desde Google Anthos.

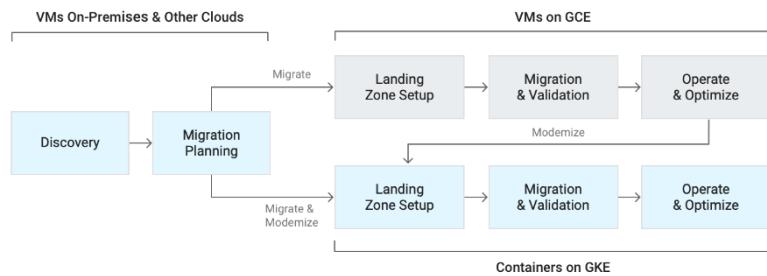
Como parte de las capacidades de plataforma de Google Anthos se encuentra el marco de gestión y control ya mencionado. Adicionalmente Google agrega el servicio de **Anthos Service Mesh** el cual basado en Istio, se encarga de todas las tareas de comunicación, seguridad y gestión de patrones modernos de acceso a las aplicaciones y administración de tráfico.



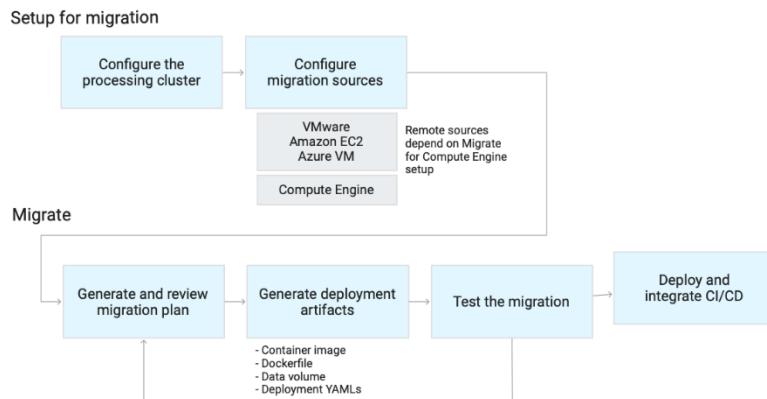
El manejo de políticas y el manejo de la configuración en Google Anthos se centraliza y se maneja desde un único punto de control, garantizando que todos los tipos de clústeres de GKE estar conforme a las políticas definidas.

Las 2 últimas capas de Google Anthos están orientadas principalmente en entregar capacidades a los desarrolladores. Algunos de los servicios del catálogo de aplicaciones que se tienen para Google Cloud, ahora están disponibles para ser usados en todo el ecosistema que gobierna Google Anthos. En particular esto hace referencia al catálogo de aplicaciones que existe en el marketplace de Google Cloud el cual permite desplegar de manera muy sencilla diferentes tipos de aplicaciones, bases de datos y servicios sobre Anthos, permitiendo también extender algunos servicios de Google Cloud como son Google Cloud Run la implementación de knative sobre Anthos, CI/CD Tools, entre otros. Por supuesto esto no es una camisa de fuerza, es posible utilizar componentes no Google sobre Anthos, soluciones open source o de otros fabricantes. Por supuesto estas capacidades van desde los IDE de desarrollo para los equipos de aplicaciones.

Finalmente, y una de las capacidades que Google ha mostrado sin ningún temor son los procesos de migración de cargas de trabajo tradicionales a contenedores.

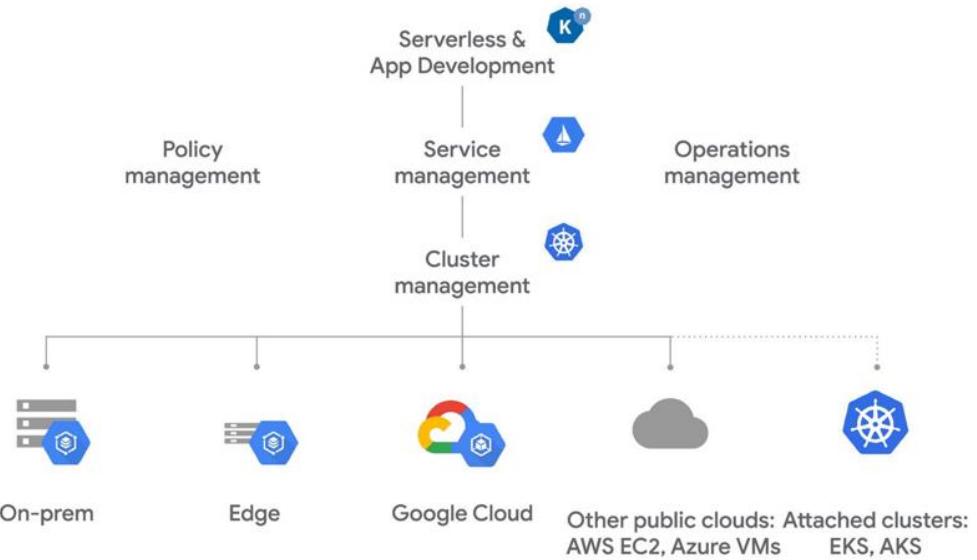


Google ha manifestado desde el principio su enfoque en acelerar el proceso de modernización de aplicaciones. Para esto ha hecho varios esfuerzos en presentar algunas soluciones que facilitan el proceso.



Esta sin duda es una de las capacidades en donde Google toma distancia de algunos de sus principales competidores, que aún no ven acá un proceso coherente técnicamente.

Finalmente, la visión de Google, así como la otras Compañías, es que un cliente en el futuro va a tener una combinación de diferentes tipos de clústeres de Kubernetes, aprovecha las capacidades de GKE en cualquier lugar, así como también máquinas virtuales, pero no ignoran que existan casos en donde un cliente quiera administrar y gestionar otros tipos de clúster como serían AKS o EKS, lo cual es un planteamiento muy interesante por parte de Google.



Bueno, acá termina mi viaje liviano sobre algunas de las capacidades de Google Anthos, lo prometido es deuda les dejo las 5 cosas que me parecen más relevantes de Google Anthos:

- **Anthos Service Mesh:** Definitivamente es una de mis favoritas, como lo dijo Joe Beda (Cofundador de Kubernetes), Kubernetes es la plataforma de plataformas y la idea de entregar la malla de servicios basada en Istio es un respaldo a la idea y necesidad de incluir una malla de servicios como parte fundamental de la arquitectura de una plataforma cloud native. Por otro lado, la idea de entregar la malla como un servicio gestionado, es maravillosa.
- **Migración de cargas de trabajo:** Considero de vital importancia este punto, porque si bien no todas las cargas de trabajo van a correr seguramente en un entorno de contenedores, hoy si tenemos una cantidad de cargas de trabajo que tomarían ventajas muy importantes al hacerlo. Esto acelera el proceso de modernización de aplicaciones y disminuye el esfuerzo considerablemente.
- **MultiNube:** Hoy en día tenemos cientos de debates con personas rasgando sus vestiduras con el tema de multicloud, nube híbrida y demás. Para mí un aspecto relevante del enfoque de Anthos es la capacidad de ver lo que el cliente puede necesitar o querer en el futuro. La idea de permitir que GKE se extienda a cualquier infraestructura me parece muy interesante sobre todo desde la perspectiva de operación y gestión en entornos altamente desacoplados y heterogéneos.
- **Enfoque Abierto:** Si bien Google ha sido y sigue siendo un gran contribuidor del ecosistema, el enfoque en soluciones open source y su apoyo y respaldo a proyectos muy relevantes para el futuro de las aplicaciones como lo son Istio, knative, Kubernetes, gVisor es una muestra de su apoyo en la democratización de la tecnología. Espero que lo siga haciendo en el futuro ya que sus contribuciones han abierto mercados y proyectos de gran impacto.
- **Ingeniería de productos:** Desde la primera vez que despliegue un clúster desde GKE, admire la simplicidad con la que Google construye sus productos. Personalmente creo que se toman muy enserio el diseño de todos los productos que liberan al mercado, en este caso particular con Anthos no es la excepción. Su enfoque siempre ha sido la experiencia de usuario.

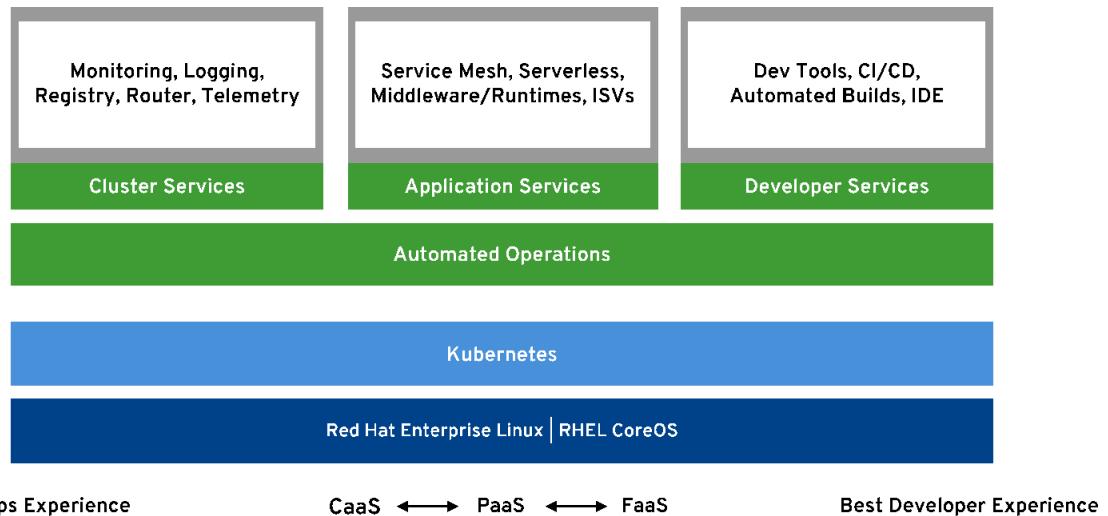
REDHAT OPENSHIFT CONTAINER PLATFORM



RedHat una de las Compañías con gran experiencia en el mercado ofreciendo soluciones open source en un modelo de suscripción, su manera de monetizar. RedHat Openshift a diferencia de algunas de las plataformas de gestión de contenedores, no nació propiamente con Kubernetes. Los inicios de Openshift estuvieron más hacia ofrecer un PaaS y durante algunos años su principal competidor fue Cloud Foundry. Un PaaS enfocado en ayudarles a las organizaciones a definir el ciclo de vida completo para la construcción de aplicaciones desde sus diferentes estados y etapas desde el desarrollo, pruebas y producción, asegurando todas sus dependencias y trabajando en su momento con la infraestructura disponible para lograrlo.

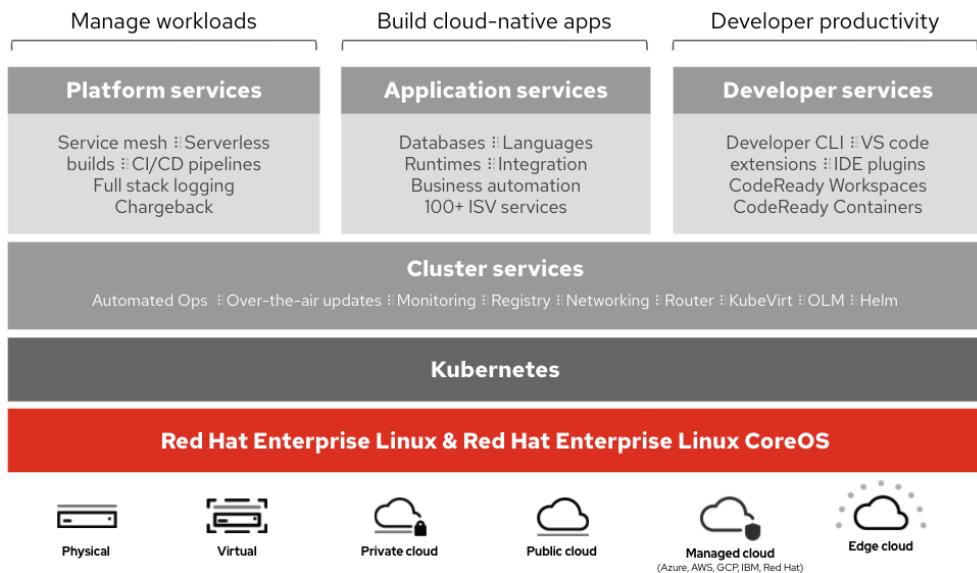
Con la llegada de Kubernetes, Openshift se transforma y pasa de tener un modelo únicamente de un PaaS hacia un modelo mixto de PaaS + CaaS + FaaS.

Este proceso de transformación tuvo impactos, sin embargo, creo que hoy ya es prueba superada para RedHat.



Openshift Container Platform también entrega un conjunto de herramientas para darles a los equipos de desarrollo y operaciones todo lo que pueden necesitar para la construcción y operación de aplicaciones modernas (cloud native).

En el siguiente diagrama funcional, podremos explorar las capacidades de Openshift 4, el cual a la fecha de liberación de este libro es la última versión.



Desde abajo hacia arriba Openshift tiene un enfoque multiplataforma y hoy cuenta con las certificaciones sobre los principales proveedores de servicios de nube. AWS, GCP, IBM, Azure así como de plataformas on-premise como lo son VMware vSphere, RedHat Virtualization, Z Series. De la misma manera tiene modelos dedicados y administrados en múltiples nubes, simplificando el modelo de consumo dependiendo de las necesidades.

Cluster Services es probablemente en mi opinión uno de los componentes poco valorados, pero que definitivamente ofrece un valor muy alto para clientes que saben lo que significa operar Kubernetes.

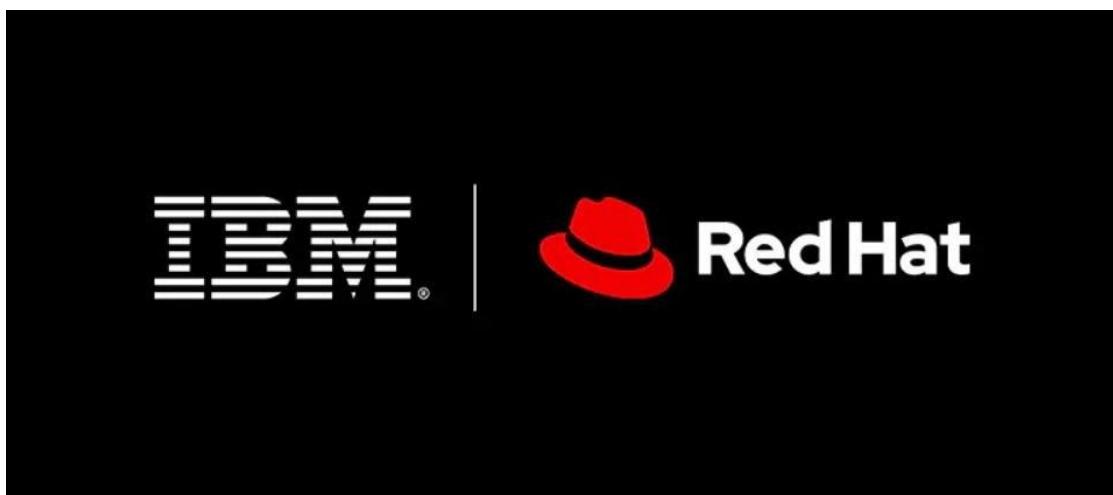
Cluster Services es el add-on que RedHat ha desarrollado para entregar todas las capacidades de gestión, operación y desarrollo adicionales a los servicios nativos de Kubernetes. En esta capa se incluyen varios proyectos opensource nativos o como operadores, correos, entre otros certificados en RedHat Openshift.



En el diagrama funcional de RedHat Openshift en sus capas de platform, application y developer services, entrega un conjunto de proyectos opensource y capacidades en donde realmente tiene capacidades muy relevantes. Por ejemplo, la posibilidad de ir desde CaaS, PaaS y FaaS es muy interesante. RedHat Openshift incluye implementaciones certificadas

de knative para ayudar en la implementación de cargas de trabajo serverless (FaaS) basado en Kubernetes asegurando capacidades de portabilidad.

Como vimos en la sección de aporte histórico a Kubernetes, RedHat ocupa el segundo lugar después de Google en contribución. Esto se traduce a su vez en aporte en diferentes proyectos del ecosistema de cloud native, lo cual han convertido a Openshift en una de las plataformas más maduras del mercado. De hecho, esa fue una de las razones por las cuales IBM realizó la adquisición de RedHat en el 2018 por 34 billones de dólares, en donde por supuesto Openshift fue una de las razones más atractivas detrás de esa inversión, sumado a la cultura de la compañía.



Hoy IBM aceleró su estrategia de nube hibrida y el path de modernización de sus aplicaciones utilizando Openshift. La capacidad de llevar una plataforma cloud native multiplataforma integrando todas las capacidades necesarias para la evolución de las aplicaciones modernas, convierten a Openshift en un jugador importante en el mercado.

Bueno, por acá termina mi viaje sobre algunas de las capacidades de Openshift, lo prometido es deuda les dejo las 5 cosas que me parecen más relevantes:

- **Cluster Services:** Definitivamente en esta capa van a tener todos los componentes para asegurar un correcto ciclo de vida de la plataforma. Los quiero invitar a revisar los detalles incluidos en la capa de servicios de clúster, con detalles muy a bajo nivel pero que definitivamente simplifican la operación de la plataforma. Amplíen en detalles como kubevirt, OLM Operator LifeCycle Manager, CoreOS y otros más.
- **MultiPlataforma:** La posibilidad de consumir Openshift en IaaS y PaaS en diferentes nubes, on premises en plataformas virtualizadas y baremetal son una opción consistente de operación para múltiples tipos de cargas de trabajo.
- **Enfoque Abierto:** El aporte en los proyectos open source en el ecosistema cloud native es sin duda el corazón de la innovación. Ubicar diferentes logos de proyectos open source en un power point es simple, asegurar y entregar a los clientes el soporte y mantenimiento NO lo es... y acá RedHat ha hecho un trabajo muy interesante en su historia como una de las empresas con mejor estrategia de monetización del open source de la industria.
- **IBM:** Creo que el futuro de Openshift es interesante y con la llegada del gigante azul se vienen muchas cosas interesantes. La capacidad de investigación y desarrollo

sumada a la capacidad comercial que tiene IBM en el mercado pueden acelerar la innovación y adopción de Openshift en el futuro.

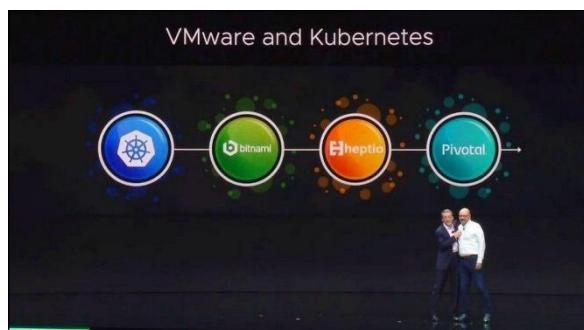
- **Personas:** La primera vez que ingresé en Openshift tuve respuesta a una de las problemáticas de las soluciones de tecnología y es construir productos para las necesidades clientes o buscar clientes para tus productos. El diseño y arquitectura de Openshift está enfocado en mejorar la interacción entre los equipos de desarrollo y operaciones. Este no es un tema menor sobre todo en el ecosistema cloud native en donde se tienen múltiples componentes con roles diferentes. De hecho, mi recomendación para las empresas que quieren simplificar esa fricción y tienen deseo de acelerar la adopción es Openshift.

VMWARE TANZU



Mi amigo Jorge Torres realizó una introducción mencionando algunos de los componentes principales de VMware Tanzu, por lo cual no voy a entrar en detalles en este capítulo. Voy a hacer algunas menciones interesantes que veo detrás de VMware Tanzu y las características interesantes a considerar.

VMware los últimos años realizó una transformación muy importante con la llegada de Kubernetes. Por su puesto VMware logró interpretar que la arquitectura de las aplicaciones ha cambiado y que a su vez la infraestructura también, razón por la cual las inversiones están direccionaladas a seguir manteniéndose como un jugador importante en la industria de la infraestructura y con la llegada de Kubernetes en las aplicaciones.



VMware adquirió Heptio, compañía fundada por 2 de los creadores de Kubernetes, Bitnami. Compañía dedica a simplificar la implementación de aplicaciones utilizando un catálogo estandarizado y por último Pivotal (de la familia) muestran la importancia que tiene para VMware Kubernetes, sumado al trabajo que viene haciendo con proyectos como Pacific, el cual consiste en llevar Kubernetes al kernel del hipervisor con capacidades para brindar a sus clientes una capa común para todos sus diferentes tipos de cargas de trabajo sobre vSphere.

Personalmente considero que tiene mucho para dar en el futuro y lo importante en el corto plazo es consolidar una estrategia simple para sus clientes. Sobre todo, en un mercado en donde muchos clientes ya no quieren comprar software, quieren consumir nube y pagar por su uso.

Bueno, por acá termina mi viaje sobre algunas de las capacidades de Tanzu, lo prometido es deuda, les dejo las 5 cosas que me parecen más relevantes:

- **vSphere:** vSphere fue y es hoy un elemento que facilitó la primera ola de transformación de la infraestructura para las aplicaciones. Con más de medio millón de clientes en el mundo, son una puerta abierta para que esos mismos clientes que confiaron en VMware, renueven votos y lleven sus cargas de trabajo de kubernetes hacia vSphere.
- **MultiPlataforma:** VMware logró llevar su infraestructura fuera del centro de datos, logrando que AWS, Azure, Oracle Cloud, Google Cloud, IBM Cloud abrieran las puertas de sus centros de datos para ofrecer un camino sencillo de migración de sus cargas de trabajo a la nube. En ese sentido Tanzu habilita capacidades multiplataforma para clientes que usan vSphere o servicios nativos en cada nube con una cartera de productos y soluciones multicloud.
- **Enfoque Abierto:** El último tiempo VMware se ha ganado un espacio de contribución muy relevante en el ecosistema cloud native, principalmente por los proyectos que venía desarrollando Heptio y que ahora toman relevancia en VMware: (Velero, Sonobuy, Cluster API, Harbor, Spring, Concord, Antrea, Octant, entre otros.) Ya veremos cuál es la disposición para la monetización del open source en el futuro.
- **Definido por Software:** Luego de transformar la infraestructura del centro de datos, VMware desarrolló un conjunto de tecnologías que permitieron llevar diferentes servicios en un modelo de virtualización. Las redes y la seguridad, el almacenamiento y su sólida adopción, convierten a VMware hoy en una compañía muy sólida para ofrecer un conjunto de soluciones de plataformas cargas de trabajo cloud native propias de VMware o de terceros con soluciones líderes en el mercado. (VMware Cloud Foundation). Por su puesto para clientes que ven valor en mantener sus aplicaciones sobre una arquitectura de soluciones y tecnologías VMware.
- **SaaS:** VMware inició un proceso de transformación de sus soluciones ofreciendo un conjunto de herramientas en un modelo SaaS. Este modelo SaaS en mi opinión tiene gran valor, ya que elimina la necesidad de gestionar soluciones en el centro de datos en donde las soluciones de VMware siempre necesitaron de alta experiencia y conocimiento. Algunas de las soluciones de Tanzu se ofrecen en un modelo SaaS para la gestión de clústeres de Kubernetes, Monitoreo y Gestión, Servicios de Service Mesh, entre otros. Esto elimina la necesidad de gestión de componentes, el desafío es que estos costos puedan pagarse en su relación costo/beneficio.

FUTURO CLOUD NATIVE

Como hemos estado revisando durante este capítulo, las tecnologías cloud native llegaron para quedarse. Las principales empresas de tecnología del mundo están trabajando en el desarrollo de este tipo de tecnologías y proyectos. La pregunta es ¿Cuál es el futuro?, pues bien, los casos de uso son varios este cambio de paradigma toca la inteligencia artificial el aprendizaje automático AI/ML, la modernización en las redes 5G, NFV, el internet de las cosas, entre muchos otros. Los invito a revisar e investigar un poco más en estos puntos y cómo el enfoque cloud native, acelera el desarrollo de estas tecnologías.

Los beneficios que trae este enfoque moderno en la construcción de aplicaciones y plataformas: brindan enormes beneficios basados en las necesidades particulares de escalabilidad y velocidad, pero que han ido evolucionando con el paso del tiempo. Proyectos como Istio, knative y tekton llamados a ser la siguiente generación cloud native, definitivamente van a ser parte de esta innovación y del futuro de las tecnologías cloud native.

Sugiero revisar algunos de los siguientes proyectos:



El proyecto Kubeflow se dedica a hacer que las implementaciones de flujos de trabajo de aprendizaje automático (ML) en Kubernetes sean simples, portátiles y escalables. Nuestro objetivo no es recrear otros servicios, sino proporcionar una manera sencilla de implementar los mejores sistemas de código abierto para ML en diversas infraestructuras. En cualquier lugar que esté ejecutando Kubernetes, debería ser capaz de ejecutar Kubeflow.

<https://www.kubeflow.org/>



La distribución certificada de Kubernetes creada para IoT & Edge computing desarrollada por Rancher Labs. K3s es una distribución de Kubernetes certificada y de alta disponibilidad diseñada para cargas de trabajo de producción en ubicaciones remotas desatendidas, con recursos limitados o dentro de dispositivos de IoT.

<https://k3s.io/>



La tecnología KubeVirt aborda las necesidades de los equipos de desarrollo que han adoptado o desean adoptar Kubernetes, pero poseen cargas de trabajo existentes basadas en máquinas virtuales que no se pueden contenerizar fácilmente. Más específicamente, la tecnología proporciona una plataforma de desarrollo unificada donde los desarrolladores pueden crear, modificar e implementar aplicaciones que residen tanto en contenedores de aplicaciones como en Máquinas virtuales en un entorno común y compartido.

<https://kubevirt.io/>



Vitess es una solución de base de datos para implementar, escalar y administrar grandes grupos de instancias de bases de datos de código abierto. Actualmente es compatible con MySQL y MariaDB. Está diseñado para ejecutarse con la misma eficacia en una arquitectura de nube pública o privada que en hardware dedicado.

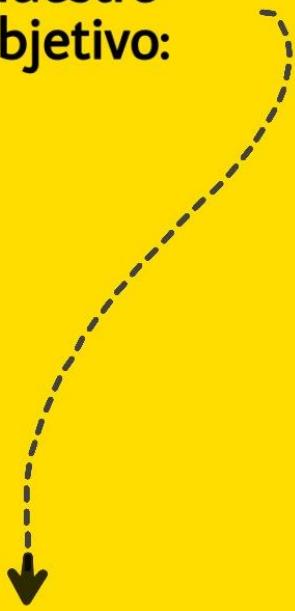
<https://vitess.io/>

REFLEXIONES FINALES

Quiero agradecerte por el tiempo que tomaste para leer mi capítulo, mi objetivo principal era dar una revisión a la evolución de las tecnologías y el impacto que han tenido en el mercado. Desde la experiencia que he podido tener trabajar en 2 Compañías que han apuntado con mucho ímpetu y liderazgo en estas tecnologías. Estoy convencido que el futuro promete. La llegada del ecosistema cloud native generó una evolución y un impacto fuerte en la forma en cómo se construyen las aplicaciones y la infraestructura. No es casualidad que Compañías del tamaño de Google, IBM, RedHat, VMware, Azure hayan transformado muchos de sus productos y soluciones para este nuevo paradigma.

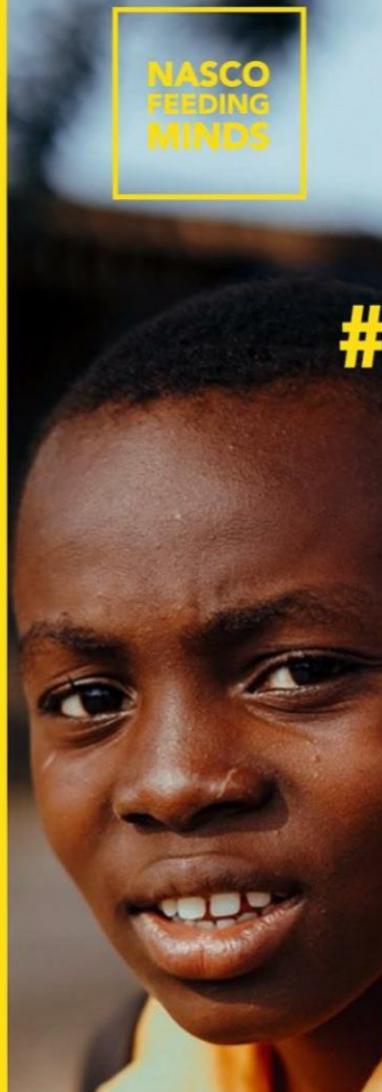
Conocer los marcos de gobierno de los proyectos, las Compañías detrás de esa contribución y la importancia del enfoque abierto es muy importante en el ecosistema. Cloud Native no es una tecnología, un estilo, es una cultura, es una alteración, es una mutación, es una nueva ola. Los que abrazan este cambio a tiempo van a tener diversión los próximos años, estamos en los primeros años de una transformación el futuro es impresionante.

Nuestro
objetivo:



Alimentar
mentes

NASCO
FEEDING
MINDS



#CAMBIA
LA
HISTORIA

La solución a
la inmigración
se encuentra
en el país de
origen

ALTARO BACKUP

HYPER-V | VMWARE | PHYSICAL | ENDPOINT | OFFICE 365

Ejecución rápida. Eficiente. Valor
imbatible. Solución de respaldo
para Hyper-V, VMware, Office365,
terminales y servidores.



Más de 50 000 clientes confían en Altaro.

Primeros 30 días gratis: www.altaro.com

Capítulo 11

KUBERNETES DESDE CERO



Raúl Unzué

@elblogdeNegu

INTRODUCCIÓN A KUBERNETES

Kubernetes, o k8s en su forma abreviada, es una plataforma de código abierto que permite la automatización de implementaciones, el escalado y la administración de aplicaciones que residen en contenedores.

El objetivo principal de Kubernetes, es reducir la carga de orquestar la infraestructura subyacente de cómputo, red y almacenamiento.

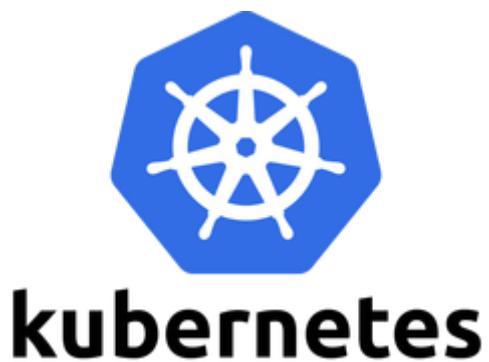
Esto permite, que los operadores y desarrolladores de aplicaciones, se centren por completo en los flujos de trabajo que tienen que ver con los contenedores y en la operación del autoservicio. Consiguiendo automatización de alto nivel y flujos de trabajo personalizados, que permiten implementar y gestionar aplicaciones compuestas de múltiples contenedores.

Kubernetes es capaz de ejecutar todas las categorías principales de cargas de trabajo, como aplicaciones monolíticas, aplicaciones con estado o sin estado, microservicios, servicios, trabajos por lotes...., aunque su uso más extendido es el de los microservicios.

Un microservicio es un modo de programar software, donde se dividen los elementos, por ejemplo, de una aplicación, en piezas mucho más pequeñas, que trabajan independientemente entre ellas, para en conjunto llevar a cabo las mismas tareas. Es un elemento fundamental de la optimización del desarrollo de aplicaciones hacia un modelo nativo en la nube.

Aunque hablaremos de la historia en otro apartado, Kubernetes fue desarrollado como una solución por parte de Google para llevar aplicaciones a producción para después ser entregado a la Cloud Native Computing Foundation en 2014.

Hoy en día es uno de los proyectos de código abierto con mayor crecimiento en los últimos años y abarca hosts tanto en nubes públicas, privadas o híbridas. Siendo una plataforma que permite ejecutar contenedores tanto en grupo de máquinas virtuales o híbridas.



La ventaja de Kubernetes, sobre otras tecnologías como Docker, es que resuelve muchos problemas relacionados con la ejecución de microservicios en una arquitectura de producción.

¿Cómo lo hace? Kubernetes nos provee de ciertas herramientas o servicios como:

- **Escalado:** pudiendo crear copias de uno o varios contenedores, facilitando escalar nuestras aplicaciones y asegurando que todas se encuentren funcionando. Podemos decirle a Kubernetes que si nuestro tráfico, o la CPU de nuestro servidor aumenta hasta cierto número, cree más réplicas de nuestros pods para satisfacer la demanda.
- **Equilibrio de carga:** nos provee automáticamente, de un balanceador de carga interno o externo para nuestros servicios. Kubernetes proporciona a los Pods sus propias direcciones IP y un solo nombre DNS para un conjunto de Pods, y puede equilibrar la carga entre ellos.
- **Actualizaciones y despliegue de código:** utiliza varios mecanismos que permiten revertir, comprobar o comprobar el historial de un despliegue
- **Autocuración** eliminando, reemplazando o reprogramando los contenedores que están fallando o mueren
- **Orquestación de almacenamiento:** es posible montar automáticamente el sistema de almacenamiento que mejor nos convenga, ya sea almacenamiento local, de un proveedor de nube pública como AWS o GCP, o sistemas de red como iSCSI, Ceph, Gluster o NFS entre otros.
- **Alta disponibilidad:** nos permite montar nodos suficientes para dar una alta disponibilidad a nuestras aplicaciones. Para ello hay que montar mínimo 3 nodos Máster y 3 Workers

Todas estas tareas, durante años han sido ejecutadas tradicionalmente de forma manual o era complicado realizarlas de forma tradicional.

Adicionalmente, como ya hemos comentado, al ser de código abierto, su diseño brinda la libertad de aprovechar la infraestructura en la nube local, híbrida o pública, pudiendo trasladar sin esfuerzo las cargas de trabajo a donde nos interese.

Como último apunte, y al hilo de la temática del libro, la diferencia fundamental entre K8s y VMware Enterprise PKS está en las diferentes configuraciones que podremos implementar:

What Enterprise PKS Adds to Kubernetes

The following table details the features that Enterprise PKS adds to the Kubernetes platform.

Feature	Included in K8s	Included in Enterprise PKS
Single tenant ingress	✓	✓
Secure multi-tenant ingress		✓
Stateful sets of pods	✓	✓
Multi-container pods	✓	✓
Rolling upgrades to pods	✓	✓
Rolling upgrades to cluster infrastructure		✓
Pod scaling and high availability	✓	✓
Cluster provisioning and scaling		✓
Monitoring and recovery of cluster VMs and processes		✓
Persistent disks	✓	✓
Secure container registry		✓
Embedded, hardened operating system		✓

HISTORIA KUBERNETES

Kubernetes comenzó como una solución de orquestación de contenedores internos de Google.

El nombre en clave original para Kubernetes dentro de Google fue "Project Seven of Nine", una referencia a un personaje que interpreta a un dron en Star Trek del mismo nombre. Ahora explicaremos qué es el sistema Borg, de donde parte Kubernetes.



Para contar la historia de cómo Kubernetes evolucionó de una solución de orquestación de contenedores internos en Google, a la herramienta que conocemos hoy, profundizamos en la historia y vamos a recopilar los hitos más significativos.

2003-2004: NACIMIENTO DEL SISTEMA BORG

<https://research.google/pubs/pub43438/>

Alrededor de 2003-2004, comienza un pequeño proyecto dentro de Google, con unas 3-4 personas, con el objetivo de crear una nueva versión del motor de búsqueda.

El sistema Borg de Google, es un sistema de gestión de clúster que ejecuta cientos de miles de trabajos, desde miles de aplicaciones diferentes, a través de varios clústeres, cada uno con hasta decenas de miles de máquinas.

Este sistema, originalmente fue escrito en el lenguaje de programación C++, aunque finalmente reescrito en Go.

2013: DE BORG A OMEGA

Después de Borg, Google presentó el sistema de gestión de clúster llamado Omega, que tenía una gran escalabilidad de grandes clústeres de cómputo.

2014: GOOGLE PRESENTA KUBERNETES

A mediados de 2014, se presenta Kubernetes como una versión de código abierto de Borg, se abre el proyecto sobre github y se unen a la comunidad de Kubernetes grandes actores como Microsoft, Red Hat, IBM o Docker.

<https://github.com/kubernetes/kubernetes/commit/2c4b3a562ce34cddc3f8218a2c4d11c7310e6d56>

2015: CLOUD NATIVE COMPUTING FOUNDATION Y V.1.0

Siguen sumándose al proyecto nuevas empresas y proyectos (Huawei, Openshift...), se presenta durante este año las versiones 1.0 y 1.1, comienzan las primeras charlas técnicas en San Francisco y el paso más importante, Google se asocia a la Linux Foundation para formar Cloud Native Computing Foundation (CNCF).

2016: DIFERENTES INTEGRACIONES Y MEJORAS

Primera versión de Helm (gestor de paquetes de Kubernetes), lanzamiento de Minikube que facilita la gestión local de Kubernetes, las nuevas versiones que surgen durante el año (1.2, 1.3, 1.4 y 1.5) traen novedades como el escalado, implementación simplificada de aplicaciones, administración automatizada de clústeres, kubeadm, compatibilidad con OpenAPI, soporte para Windows Server, ...

¡Como curiosidad, Pokemon GO! (la aplicación del año) se convierte en un caso de éxito, como el mayor despliegue de Kubernetes en Google Container Engine.

2017: ESTABILIZACIÓN Y ADOPCIÓN

La versión 1.6 trae la estabilidad a Kubernetes.

Google, IBM y Lyft anuncian la tecnología Istio, que permite la gestión del flujo de tráfico, aplicar políticas de acceso y agregar datos de telemetría entre microservicios.

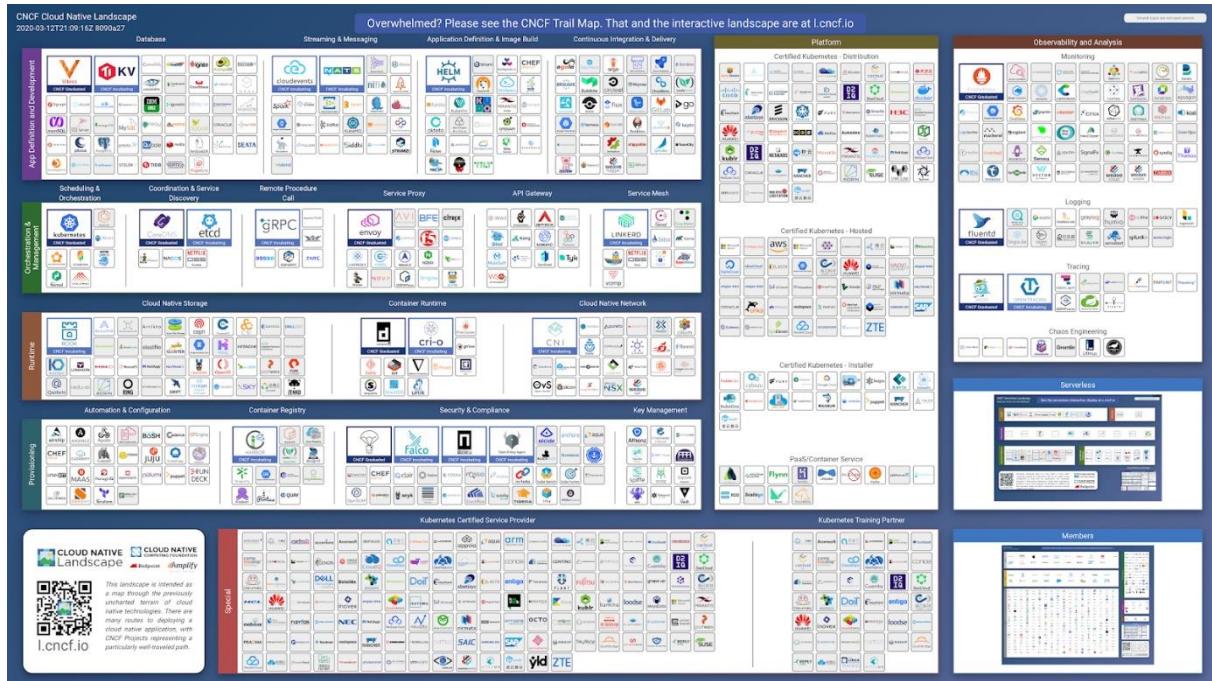
<https://developer.ibm.com/dwblog/2017/istio/>

Otros hitos importantes este año que podemos reseñar son como que Github se ejecuta en Kubernetes, Oracle se une a la CNCF, la propia CNCF anuncia los primeros proveedores de certificados para Kubernetes, se presentan las versiones 1.7, 1.8 y 1.9 (acceso basado en roles, control de acceso a la API...), Docker adopta complementariamente Kubernetes, Amazon anuncia Elastic Container Service para Kubernetes, presentación Kubeflow...

2018 - 2019: EVOLUCIÓN

Google lanza el Podcast de Kubernetes, se celebran varias KubeCon a ambas partes del charco, Istio 1.2, diferentes versiones hasta 1.17, ...

Os dejo este esquema de todos los proyectos CNCF:

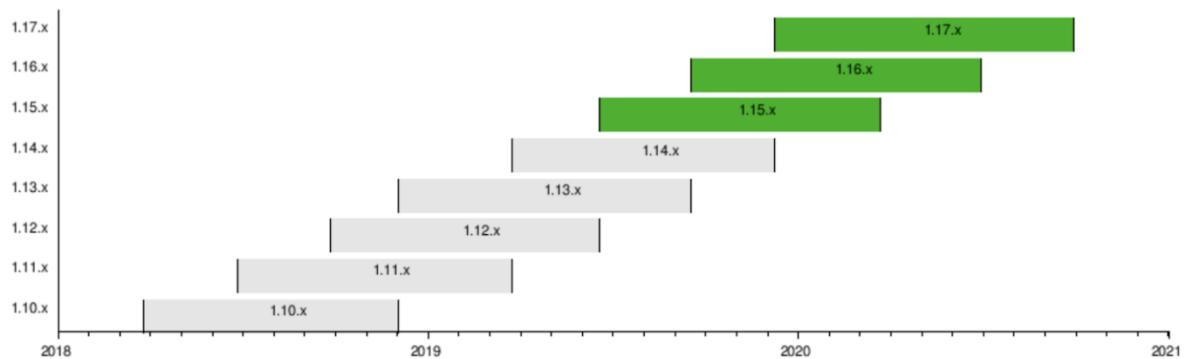


<https://www.máquinasvirtuales.eu/wp-content/uploads/2020/04/kubernetes-historia-2.png>

CICLO DE VIDA DE VERSIONES EN KUBERNETES

Si implantáis proyectos sobre Kubernetes, debéis tener en cuenta el ciclo de vida del versionado que vais a gestionar y la política de soporte que se implementa, para que no generéis aplicaciones sobre sistemas obsoletos.

En este caso, Kubernetes sigue una política de soporte N-2 (lo que significa que las 3 versiones menores más recientes reciben correcciones de seguridad y errores). Esto generalmente da como resultado una versión menor en particular compatible con ~ 9 meses; como se ilustra en la tabla a continuación:



A continuación, podéis comprobar las versiones y su soporte actual:

Versión	Fecha de lanzamiento	Notas
1.0	10 de julio de 2015	Lanzamiento original
1.1	9 de noviembre de 2015	
1.2	16 de marzo de 2016	
1.3	1 de julio de 2016	
1.4	26 de septiembre de 2016	
1.5	12 de diciembre de 2016	
1.6	28 de marzo de 2017	
1.7	30 de junio de 2017	
1.8	28 agosto 2017	
1.9	15 de diciembre de 2017	
1.10	28 marzo 2018	
1.11	3 de julio de 2018	
1.12	27 de septiembre de 2018	
1.13	3 de diciembre de 2018	
1.14	25 marzo 2019	
1.15	20 junio 2019	
1.16	22 octubre 2019	
1.17	9 de diciembre de 2019	

Leyenda: Versión antigua Versión anterior, aún mantenida
Última versión Última versión preliminar Lanzamiento futuro

FUENTE: <https://en.wikipedia.org/wiki/Kubernetes#History>

Al ser de código abierto, podéis encontrar todo el ChangeLog entre versiones en la web del proyecto:

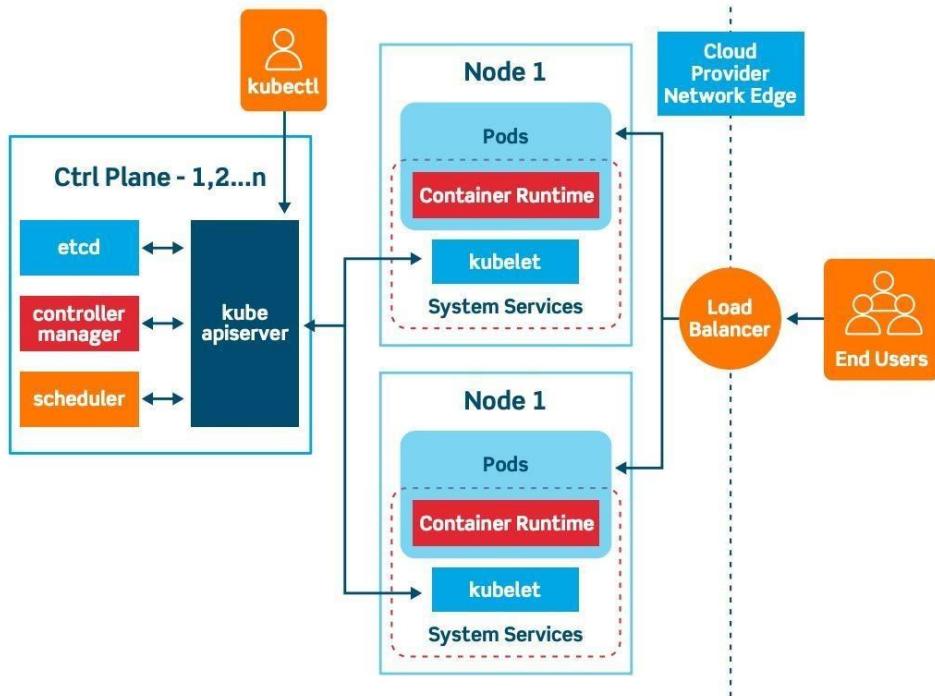
<https://kubernetes.io/docs/setup/release/notes/>

COMPONENTES PRINCIPALES EN KUBERNETES

Como se dijo anteriormente (pero vale la pena comentarlo de nuevo, para explicar la arquitectura y los componentes), Kubernetes es una plataforma de código abierto para implementar y administrar contenedores. Proporciona un tiempo de ejecución de contenedor, orquestación de contenedores, orquestación de infraestructura centrada en contenedores, mecanismos de auto curación, descubrimiento de servicios y equilibrio de carga. Se utiliza

para la implementación, el escalado, la administración y la composición de contenedores de aplicaciones en grupos de hosts.

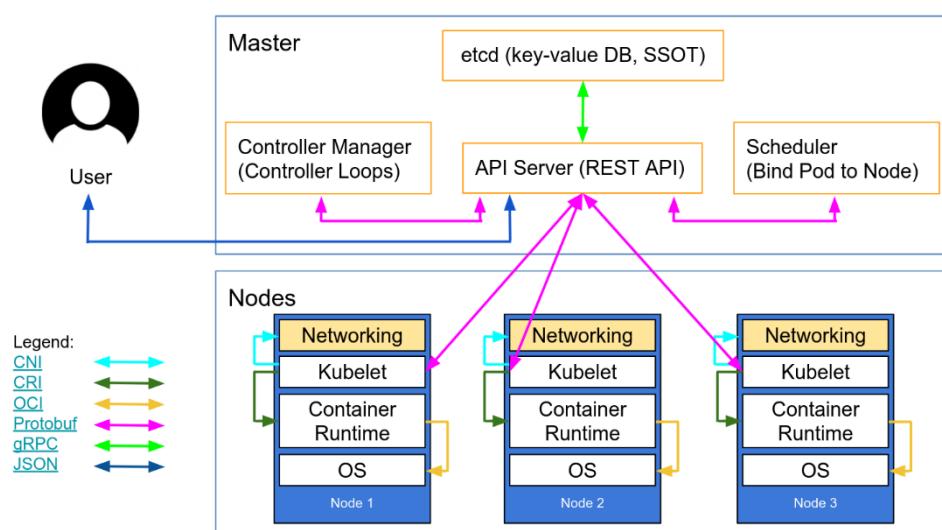
Desde un nivel alto, un entorno de Kubernetes consta de un panel de control (maestro o máster), un sistema de almacenamiento distribuido para mantener el estado del clúster coherente (etcd) y varios nodos del clúster (Kubelets).



Aunque vamos a detallar varios de ellos en un apartado propio os dejo los términos de los diferentes componentes que os vais a encontrar:

- **Clúster:** Conjunto de máquinas físicas o virtuales y otros recursos utilizados por kubernetes.
- **Máster:** el máster o maestro o padre es la máquina que gestiona los nodos de Kubernetes, asignando tareas a cada nodo.
- **Slave o Nodo o Worker:** un esclavo o nodo o worker o helpers (este año 2020 se está planteando la modificación del término por interpretarse como racista) es una máquina que realiza las tareas que le indica el máster.
- **Pod:** un grupo de uno o más contenedores implementados en un nodo único. Todos los contenedores de un pod comparten la dirección IP, la IPC Socket, el nombre del host y otros recursos. Los pods abstraen la red y el almacenamiento del contenedor subyacente. Esto le permite mover los contenedores por el clúster con mayor facilidad.
- **Controlador de replicación:** controla cuántas copias idénticas de un pod deben estar ejecutándose en algún lugar del clúster.
- **Servicio:** separa las definiciones de tareas de los pods. Los proxies de servicios de Kubernetes envían automáticamente las solicitudes de servicio al pod correspondiente, sin importar adónde se traslada en el clúster, o incluso si está siendo reemplazado.

- **Kubelet:** es un servicio que se ejecuta en cada nodo y revisa si los contenedores están iniciados y ejecutándose.
 - **kubectl:** herramienta para la configuración de la línea de comandos de Kubernetes.
 - **Etcd:** base de datos de Kubernetes, donde se guardan las configuraciones de Kubernetes, los estados de los pods, deploys... Está basada en base de datos key values por lo que es muy sencilla, rápida y escalable. K3s usa por ejemplo SQL Lite que está pensado a entornos más pequeños. Generalmente, es raro tener que mantener este tipo de servicios por los administradores
 - **API Server (REST API):** API significa interfaz de programación de aplicaciones. Desde la API se realiza la orquestación del ciclo de vida (escalado, actualizaciones, etc.) de diferentes aplicaciones. También actúa como la puerta de entrada al clúster, por lo que los clientes deben poder acceder al servidor API desde fuera del clúster. Por ejemplo, los clientes se autentican a través del Servidor API y también lo usan como proxy / túnel para nodos, pods y servicios.
 - **Scheduler:** se encarga de colocar los Pods en los diferentes nodos del clúster. Lo que hace es analizar los Pods y ver cuál es el nodo factible para su ejecución una vez que lo tiene decidido, se pone en contacto con la API mediante un proceso que se llama Binding.
 - **Container Runtime:** permite la ejecución de los contenedores dentro del clúster.



NETWORKING

El capítulo de Networking en Kubernetes es el más complejo de explicar y manejar los aspectos internos de las redes que componen el clúster. Esto se debe a que Kubernetes se compone de varios nodos y dentro de ellos corren Contenedores y Pods que normalmente tienen que interrelacionarse entre ellos. Algo que, con Docker, por ejemplo, no pasa, ya que reside todo en el mismo host y simplemente se generan redes virtuales que residen en él.

Existen varios tipos de comunicaciones de red en un clúster Kubernetes:

- Contenedor a Contenedor
- Pod a Pod
- Pod a Servicio
- Tráfico externo a Servicio

Y existen a su vez unas reglas que tenemos que entender:

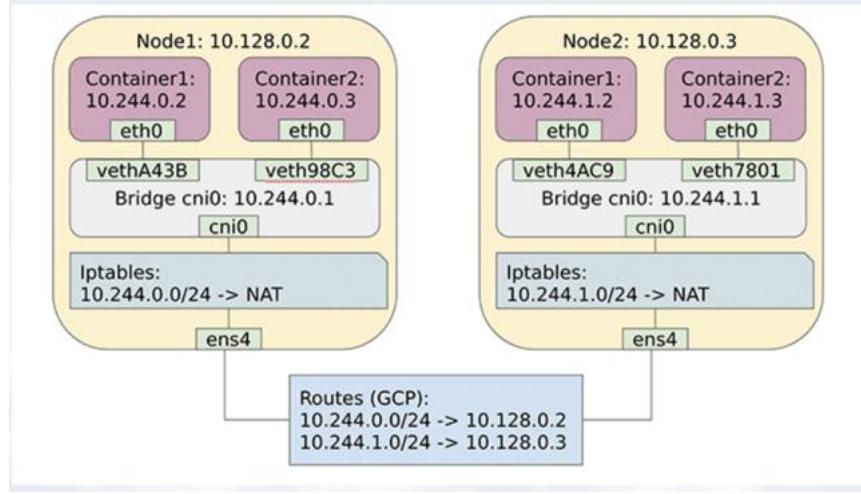
- Los Pods del clúster de Kubernetes, da igual en que nodo corran, pueden comunicarse con otros Pods de diferentes nodos sin hacer NAT
- Elementos como los demonios del sistema (Ejemplo: kubelet), pueden comunicarse con todos los Pods en ese nodo
- Todos los contenedores pueden comunicarse entre sí directamente sin NAT
- Todos los nodos pueden comunicarse con todos los contenedores (y viceversa) sin NAT
- La IP que un contenedor ve, es la misma para sí y para el resto de los componentes

Se puede decir, que cuando trabajamos con Pods de Kubernetes, se intenta alcanzar el modelo de red de la virtualización con máquinas virtuales, ya que cada Pod tiene una IP asignada de forma dinámica. Con lo que podemos usar esto para poder asignar puertos, descubrir servicios, equilibrio de carga...y poder llevar una aplicación que ya existía en una máquina virtual a Kubernetes.

Estas direcciones IP de las que hablamos de que se asignan en los Pods, residen en el mismo Namespace. Y los contenedores que corren en esos Pods, que están en el mismo Namespace, son capaces de llegar a los puertos de escucha de cada contenedor dentro de ese Namespace.

Lo hay que tener claro, es que los contenedores dentro de cada Pod tienen que gestionar el uso de los puertos de escucha.

Y a nivel de host, es posible gestionar los puertos de escucha en los Pods y reenviar a los puertos de escucha de los Pods. Siendo algo totalmente transparente para el Pod que está en ejecución.



La idea es que asignemos una subred para cada host y luego configuremos algún tipo de enrutamiento entre los hosts para reenviar el tráfico del contenedor de manera adecuada.

Para poder lograr esa comunicación entre los diferentes componentes del clúster, Kubernetes utiliza CNI (Container Network Interface) o una red superpuesta. Vamos a explicar un poco más en detalle esto...

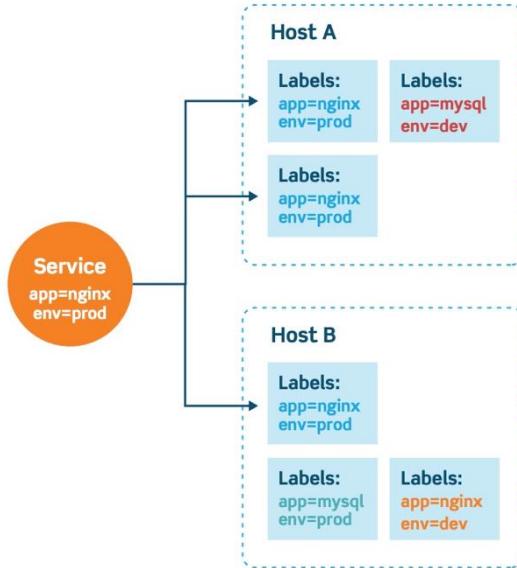
En cuanto al CNI, es un proyecto de la Cloud Native Computing Foundation, que consiste en crear unas características y bibliotecas concretas para usar plugins que permitan configurar interfaces de red en contenedores de Linux.

URL PROYECTO: <https://github.com/containernetworking/cni>

Al final, la función principal del CNI es la conectividad de los contenedores, y la creación y eliminación de los recursos asignados a los contenedores cuando son creados o eliminados.

En el propio proyecto, se pone a disposición de los diferentes desarrolladores el código Go, de tal forma que sea más fácil crear plugins.

Existen diferentes plugins que nos permitirán crear una red superpuesta, ya creados por diferentes empresas VMware (NSX), Amazon ECS, Calico, Huawei, Google... y que se pueden cargar fácilmente en un clúster Kubernetes.



Uno de los plugins más utilizado en este contexto de red es Cálico. Que es un proyecto OpenSource que gestionaría la capa de red y adicionalmente implementa seguridad en el entorno, al integrarse dinámicamente con los cambios en el firewall del propio host donde están trabajando los contenedores.

Es un plugin muy escalable y que nos permite crecer a la vez que lo haga nuestra plataforma. Existe una versión Enterprise con las siguientes características extras:

- Política de red jerárquica
- Controles de acceso de salida (políticas de DNS, puertas de enlace de salida)
- Visualización de red y resolución de problemas
- Recomendaciones de política de red
- Vista previa y puesta en escena de la política de red
- Controles de cumplimiento e informes
- Detección de intrusos (actividad sospechosa, detección de anomalías)
- Gestión de múltiples clústeres con federación de múltiples nubes

Ejemplo de instalación de Calico:

<https://docs.projectcalico.org/getting-started/kubernetes/quickstart>

- curl <https://docs.projectcalico.org/manifests/calico-typha.yaml> -o calico.yaml
- kubectl apply -f calico.yaml



COMUNICACIONES EN UN CLUSTER DE KUBERNETES

- **CLUSTER A MASTER:**

- Todos los canales de comunicación desde el clúster hacia el Máster terminan en el ApiServer (ningún otro componente del Máster está diseñado para exponer servicios remotos)
- Normalmente, se usa para la comunicación HTTPS (TCP 443), donde el ApiServer escucha
- A los nodos se les proporciona un certificado raíz para que se puedan conectar con el ApiServer
- Los Pods se conectarán al ApiServer a través de una cuenta de servicio
- Con todo esto, las conexiones por defecto dentro del clúster son totalmente seguras

- **MASTER A CLUSTER:**

- Hay dos vías de comunicación del Máster (ApiServer) al Clúster, la primera es desde el ApiServer al proceso kubelet que se ejecuta en cada nodo del Clúster:
 - Recoger entradas de registro de Pods
 - Conectar (a través de kubectl) con Pods en ejecución
 - Facilitar la funcionalidad port-forwarding del kubelet
 - Se realiza mediante conexiones HTTPS
 - Por defecto, el ApiServer no verifica el certificado del kubelet, por lo que la conexión es vulnerable a ataques del tipo man-in-the-middle, e insegura para conectar a través de redes públicas o de no confianza. Para asegurar la conexión se usa un certificado en su defecto un túnel SSH
 - Por eso, debemos habilitar la autorización y autenticación al kubelet para proteger nuestra API
- La segunda, es desde el ApiServer a cualquier nodo, Pod, o servicio a través de la funcionalidad proxy del ApiServer

- **APISERVER A NODOS, PODS Y SERVICIOS:**

- Las conexiones desde el ApiServer a un nodo, pod o servicio se realizan por defecto con HTTP y, por consiguiente, no son autenticadas o encriptadas. En consecuencia, estas conexiones no son seguras, aunque podemos implementar HTTPS

- **TÚNELES SSH:**

- Kubernetes ofrece soporte para túneles SSH (TCP 22) que protegen la comunicación entre el Máster y el Clúster
- El túnel garantiza que dicho tráfico no es expuesto fuera de la red en la que se ejecutan los nodos
- Los túneles SSH se consideran obsoletos, y no deberían utilizarse a menos que se sepá lo que se está haciendo

PUERTOS TCP/IP NECESARIOS EN KUBERNETES

A su vez, para que nuestro clúster de Kubernetes trabaje correctamente, deberemos asegurarnos de que tanto los nodos maestros como los nodos de trabajo tengan los puertos en el firewall abiertos. Os dejo a continuación un par de tablas explicativas:

Nodos maestros

Protocolo	Dirección	Rango de puertos	Propósito	Usado por
TCP	Entrada	6443*	Servidor de la API de Kubernetes	Todos
TCP	Entrada	2379-2380	etcd API de cliente de servidor	kube-apiserver, etcd
TCP	Entrada	10250	kubelet API	Auto, Plano de control
TCP	Entrada	10251	kube-scheduler	Auto
TCP	Entrada	10253	kube-controlador-gerente	Auto

Nodos de trabajo

Protocolo	Dirección	Rango de puertos	Propósito	Usado por
TCP	Entrada	10250	kubelet API	Auto, Plano de control
TCP	Entrada	30000-32767	Servicios de NodePort**	Todos

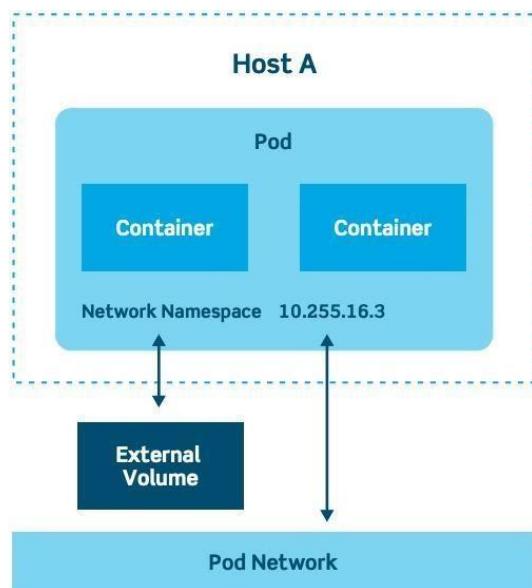
PODS

Un pod representa un proceso en ejecución en un clúster.

La unidad más pequeña de Kubernetes son los pods, con ellos podemos correr contenedores. Un pod representa un conjunto de contenedores que comparten almacenamiento y una única IP. Los pods son efímeros, cuando se destruyen se pierde toda la información que contenía. Si queremos desarrollar aplicaciones persistentes tenemos que utilizar volúmenes.

Los pods son uno de los conceptos cruciales en Kubernetes, ya que son el elemento clave con la que los desarrolladores interactúan.

Esta construcción lógica empaqueta una sola aplicación, que puede consistir en múltiples contenedores y volúmenes de almacenamiento. Por lo general, un solo contenedor (a veces con algún programa auxiliar en un contenedor adicional) se ejecuta en esta configuración, como se muestra en el diagrama a continuación.



Los pods pueden escalarse automáticamente de forma horizontal (es decir, aumentar o reducir el número de instancias) y realizar actualizaciones continuas. Y representan procesos en ejecución en un clúster.

Tipos de objetos asociados a Pods

Existen varios tipos de objetos asociados a pods:

- **ReplicaSet**, el valor predeterminado, es un tipo relativamente simple. Asegura que se ejecute el número especificado de pods
- **Deployment** es una forma declarativa de administrar pods a través de ReplicaSets. Incluye mecanismos de actualización y reversión
- **DaemonSet** es una forma de garantizar que cada nodo ejecute una instancia de un pod. Se utiliza para servicios de clúster, como monitoreo de salud y reenvío de registros

- **StatefulSet** está diseñado para administrar pods que deben persistir o mantener el estado
- **Job y CronJob** ejecutan trabajos de corta duración como únicos o en un horario.

Ejemplo de Pod

Un ejemplo de un Pod, sería una aplicación WordPress con una base de datos MySQL. Para implementarlos correctamente, lo haríamos en dos pods diferenciados, uno para cada servicio. Para ello podríamos generar dos ficheros YAML, uno para WordPress y otro para MySQL.

Pod para WordPress (`wordpress.yaml`):

```
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: wordpress-storage
  labels:
    app: wordpress
spec:
  accessModes:
    - ReadWriteMany
  resources:
    requests:
      storage: 8Gi
```

Pod para MySQL (`mariadb.yaml`):

```
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: mysql-storage
  labels:
    app: wordpress
spec:
  accessModes:
    - ReadWriteMany
  resources:
    requests:
      storage: 8Gi
```

Lanzamos la creación de los Pods:

```
kubectl create -f wordpress.yaml
pod "wordpress" created
```

```
kubectl create -f mariadb.yaml
pod "mariadb" created
```

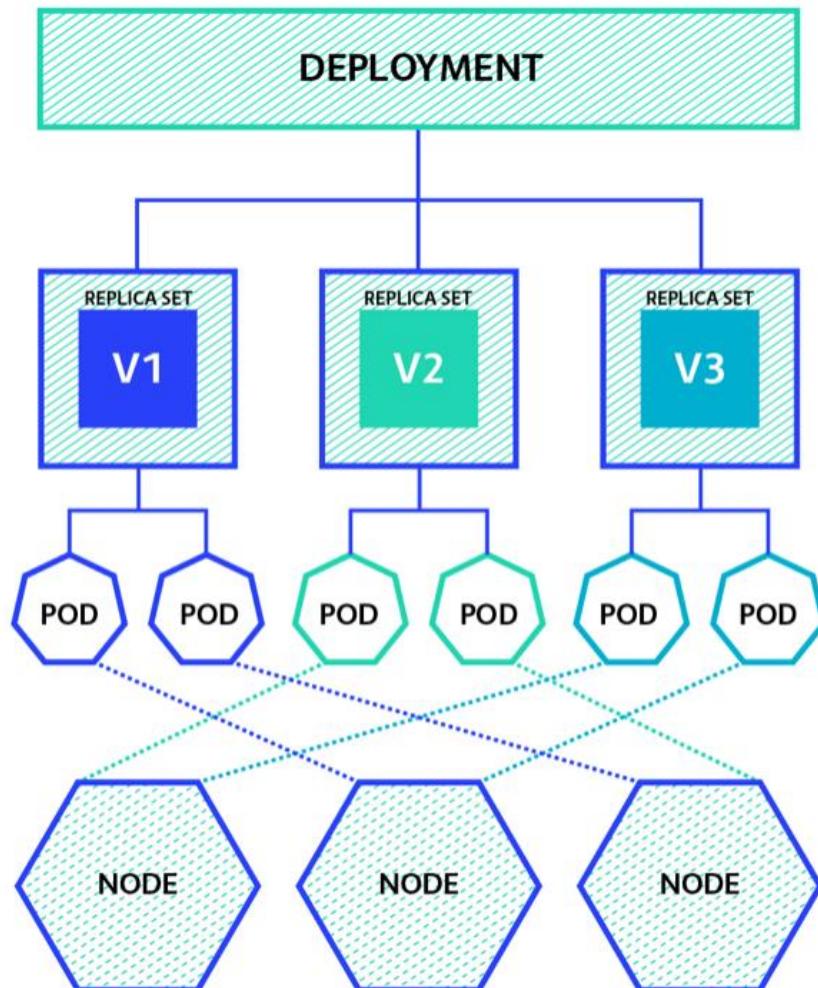
DEPLOYMENT

Podríamos definir un Deployment, como un objeto que permite administrar un conjunto de Pods idénticos.

La ventaja principal de utilizar Deployments en una implementación, es que sin ellos tendríamos que crear, actualizar o eliminar un grupo de Pods manualmente. Y esto nos da:

- Escalabilidad de los Pods
- Control total de las réplicas
- Actualizaciones continuas
- Gestión de versiones (nos facilitan el Upgrade y la vuelta atrás o Rollback)
- Despliegues automáticos

Gráficamente, se podrían decir que un Deployment es la unidad de más alto nivel que se puede gestionar en una infraestructura Kubernetes.



Creación de un Deployment

Los Deployment se crean a partir de un fichero YAML o YML, donde se definen las réplicas, pods, versión y otros múltiples parámetros. Os dejo un ejemplo básico:

```
raulunzue@KBMMASTER:~$ kubectl create -f nginx-deployment-  
elblogdenegu.yml  
  
apiVersion: apps/v1  
kind: Deployment  
metadata:  
  name: nginx-deployment-elblogdenegu  
  labels:  
    app: nginx  
spec:  
  replicas: 6  
  selector:  
    matchLabels:  
      app: nginx  
  template:  
    metadata:  
      labels:  
        app: nginx  
    spec:  
      containers:  
        - name: nginx  
          image: nginx:latest  
          ports:  
            - containerPort: 8080
```

Modificar versión de un Deployment

Podemos editar nuestro Deployment con el siguiente comando:

```
raulunzue@KBMMASTER:~$     kubectl     edit     deploy     nginx-deployment-  
elblogdenegu
```

```

# Please edit the object below. Lines beginning with a '#' will be ignored,
# and an empty file will abort the edit. If an error occurs while saving this file wil
l be
# reopened with the relevant failures.
#
apiVersion: apps/v1
kind: Deployment
metadata:
  annotations:
    deployment.kubernetes.io/revision: "1"
    kubectl.kubernetes.io/last-applied-configuration: |
      {"apiVersion":"apps/v1","kind":"Deployment","metadata":{"annotations":{},"labels":{}},"spec":{"replicas":6,"selector":{"matchLabels":{"app":"nginx"}}, "template":{"metadata":{"labels":{}},"spec":{"containers":[{"image":"nginx:latest","name":"nginx","ports":[{"containerPort":8080}]}]}}}
    creationTimestamp: "2020-03-29T19:53:24Z"
    generation: 2
  labels:
    app: nginx
  name: nginx-deployment-elblogdenegu
  namespace: default
  resourceVersion: "17705931"
  selfLink: /apis/apps/v1/namespaces/default/deployments/nginx-deployment-elblogdenegu
  uid: df6e290e-c160-40bd-9db9-5f02ef3ac50d
spec:
  progressDeadlineSeconds: 600
  replicas: 3
  revisionHistoryLimit: 10
  selector:
    matchLabels:
      app: nginx
"/tmp/kubectl-edit-he7jg.yaml" 73 lines, 2429 characters

```

O cambiar la imagen utilizada para actualizarla, por ejemplo, con el siguiente comando:

```

raulunzue@KBMMASTER:~$ kubectl set image deployment nginx-deployment-
elblogdenegu nginx=1.17.8 --all
deployment.apps/nginx-deployment-elblogdenegu image updated

```

Lo que genera una nueva réplica:

```

raulunzue@KBMMASTER:~$ kubectl get deploy,rs
NAME                                     READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/nginx-deployment-
elblogdenegu     3/3        1           3           13d
                                                 CURRENT   READY   AGE
replicaset.apps/nginx-deployment-elblogdenegu-
5499946b7f     1           1           0           2m14s
replicaset.apps/nginx-deployment-elblogdenegu-
7dd686b6bc     3           3           3           13d
                                         DESIRED

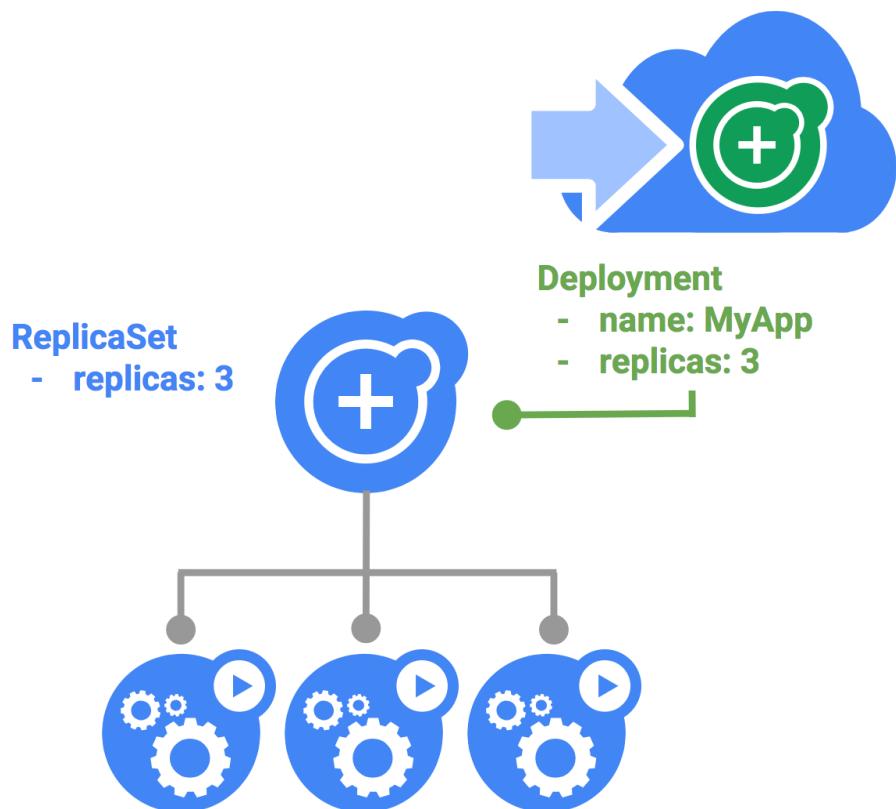
```

REPLICASET

Se trata de un tipo de objeto que tiene como tarea principal asegurarse que el número de réplicas que se establecieron para un pod sea cumplido. Esto permite a Kubernetes dar:

- Tolerancia a fallos
- Una alta escalabilidad dinámica
- Permite que la infraestructura no interrumpa el servicio

La forma más común de implementación es usando un Deployment, el cual a su vez despliega un ReplicaSet de manera transparente para nosotros. La razón principal es que los Deployments nos permiten gestionar actualizaciones de los pods y los ReplicaSet no.



Si desglosamos un fichero YAML de ejemplo que genera un Deployment, en las especificaciones marcamos el número de réplicas:

```

raulunzue@KBMMASTER:~$ nano nginx-deployment-elblogdenegu.yml

apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment-elblogdenegu
  labels:
    app: nginx
spec:
  replicas: 6
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - name: nginx
          image: nginx:latest
          ports:
            - containerPort: 8080

```

En un objeto ReplicaSet, lo haríamos de la siguiente forma:

```

apiVersion: extensions/v1beta1
kind: ReplicaSet
metadata:
  name: nginx-deployment-elblogdenegu
  namespace: default
spec:
  replicas: 6
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - name: nginx
          image: nginx:latest
          ports:
            - containerPort: 8080

```

Volvemos al Deployment, y lo generamos:

```
raulunzue@KBMMASTER:~$ kubectl apply -f nginx-deployment-
elblogdenegu.yml
deployment.apps/nginx-deployment-elblogdenegu created
```

Comprobamos como se generan las réplicas que hemos marcado en el fichero:

```
raulunzue@KBMMASTER:~$ kubectl get pods -o wide
NAME READY STATUS RESTARTS AGE IP NODE NOMINATED NODE READINESS GATES
nginx-deployment-elblogdenegu-6dbbf956cf-2rmxv 1/1 Running 0 27s
10.69.1.3 kubernetes01
nginx-deployment-elblogdenegu-6dbbf956cf-7h6sm 1/1 Running 0 27s
10.69.1.2 kubernetes01
nginx-deployment-elblogdenegu-6dbbf956cf-hs78p 1/1 Running 0 27s
10.69.1.4 kubernetes01
nginx-deployment-elblogdenegu-6dbbf956cf-hxds8 1/1 Running 5 45d
10.69.2.29 kubernetes02
nginx-deployment-elblogdenegu-6dbbf956cf-j4fs6 1/1 Running 0 27s
10.69.2.32 kubernetes02
nginx-deployment-elblogdenegu-6dbbf956cf-rg882 1/1 Running 1 59m
10.69.2.30 kubernetes02
```

Una vez desplegado el Deployment, el escalado (no siempre tiene que ser mayor número, sino que puede ser menor) es muy sencillo:

```
raulunzue@KBMMASTER:~$ kubectl scale --replicas=3 deployment nginx-
deployment-elblogdenegu
deployment.apps/hello-world-deployment-elblogdenegu scaled
```

Al escalar un número menor, veremos cómo se terminan (borran) automáticamente:

```
[raulunzue@KBMMASTER:~$ kubectl scale --replicas=3 deployment nginx-deployment-elblogdenegu
deployment.apps/nginx-deployment-elblogdenegu scaled
[raulunzue@KBMMASTER:~$ kubectl get pods -o wide
NAME READY STATUS RESTARTS AGE IP NODE NOMINATED NODE READINESS GATES
nginx-deployment-elblogdenegu-7dd686b6bc-684cl 0/1 Terminating 0 100s <none> kubernetes01 <none> <none>
nginx-deployment-elblogdenegu-7dd686b6bc-b5k82 1/1 Running 0 100s 10.69.2.53 kubernetes02 <none> <none>
nginx-deployment-elblogdenegu-7dd686b6bc-c6jms 1/1 Running 0 100s 10.69.5.227 kubernetes01 <none> <none>
nginx-deployment-elblogdenegu-7dd686b6bc-lmslj 1/1 Running 0 100s 10.69.2.52 kubernetes02 <none> <none>
nginx-deployment-elblogdenegu-7dd686b6bc-pqd2k 0/1 Terminating 0 100s <none> kubernetes02 <none> <none>
nginx-deployment-elblogdenegu-7dd686b6bc-szpqh 0/1 Terminating 0 100s <none> kubernetes01 <none> <none>
```

Los contenedores se reducen:

```
[raulunzue@KBMMASTER:~$ kubectl get pods -o wide
NAME READY STATUS RESTARTS AGE IP NODE NOMINATED NODE READINESS GATES
nginx-deployment-elblogdenegu-7dd686b6bc-b5k82 1/1 Running 0 2m19s 10.69.2.53 kubernetes02 <none> <none>
nginx-deployment-elblogdenegu-7dd686b6bc-c6jms 1/1 Running 0 2m19s 10.69.5.227 kubernetes01 <none> <none>
nginx-deployment-elblogdenegu-7dd686b6bc-lmslj 1/1 Running 0 2m19s 10.69.2.52 kubernetes02 <none> <none>
```

Podemos ver los ReplicaSet con el siguiente comando:

```
raulunzue@KBMMASTER:~$ kubectl get rs
NAME                                DESIRED   CURRENT   READY   AGE
nginx-deployment-elblogdenegu-7dd686b6bc   3         3         3      13d
```

Si os preguntáis qué une un Pod a un ReplicaSet, lo podéis observar cuando ampliáis la información que os trae el comando. Si extraemos los datos con "describe" veremos que viene definido en los Hash del Selector y las Etiquetas:

```
raulunzue@KBMMASTER:~$ kubectl describe rs nginx-deployment-elblogdenegu-7dd686b6bc
Name:           nginx-deployment-elblogdenegu-7dd686b6bc
Namespace:      default
Selector:       app=nginx, pod-template-hash=7dd686b6bc
Labels:         app=nginx
                pod-template-hash=7dd686b6bc
Annotations:    deployment.kubernetes.io/desired-replicas: 3
                  deployment.kubernetes.io/max-replicas: 4
                  deployment.kubernetes.io/revision: 1
Controlled By: Deployment/nginx-deployment-elblogdenegu
Replicas:       3 current / 3 desired
Pods Status:    3 Running / 0 Waiting / 0 Succeeded / 0 Failed
Pod Template:
  Labels:  app=nginx
          pod-template-hash=7dd686b6bc
Containers:
  nginx:
    Image:      nginx:latest
    Port:       8080/TCP
    Host Port:  0/TCP
    Environment: <none>
    Mounts:     <none>
    Volumes:    <none>
    Events:     <none>
```

JOB Y CRONJOBS

Para los que no estéis muy familiarizados con Linux, vamos a dar unas pequeñas pinceladas básicas sobre qué son los Jobs y CronJobs.

En pocas palabras los podemos definir de la siguiente forma:

- Los Jobs son trabajos secuenciales que constan de un principio y un fin. Es un objeto de ejecución única que crea una tarea y garantiza que el trabajo finalice. Son parte de la API de Kubernetes.
- Un CronJob es una utilidad de Linux, que nos permite establecer ejecuciones periódicas de forma automática, en unas horas, minutos o días concretos de esos trabajos programados (las famosas tareas programadas de Windows). También son parte de la API de Kubernetes.

Tanto los Jobs como los CronJobs nos permitirían realizar automatizaciones dentro de Kubernetes de tareas que tienen que ser programables y escalables. Ejemplos de estas tareas serían copias de seguridad, informes, envíos de emails, tareas de limpieza o modificaciones de aplicaciones en un momento concreto.

Tipos de Jobs

Existen varios tipos de Jobs:

Usaré el siguiente ejemplo para explicar los diferentes tipos de Jobs:

<https://gist.github.com/raulunzue/d6fd33c639bc71b2578816842488fce5>

- Job simple:
 - Una tarea que sólo inicia un Pod, salvo que el Pod falle
 - El Job finaliza cuando el Pod finaliza correctamente

```
apiVersion: batch/v1
kind: Job
metadata:
  name: ebdn-cuenta
spec:
  template:
    metadata:
      name: ebdn-cuenta
    spec:
      containers:
        - name: cuenta-secuencial
          image: centos:7
          command:
            - "bin/bash"
            - "-c"
            - "for i in {0..15} ; do echo $i ; sleep 1; done"
      restartPolicy: Never
```

- Job secuencial, pero con un conteo de finalización fijo:
 - Este Job iniciaría varios Pods
 - El valor *completions* marcaría cuando el Job se completa de forma exitosa según la cantidad de ejecuciones correctas de los Pods lanzados

```
apiVersion: batch/v1
kind: Job
metadata:
  name: ebdn-cuenta
spec:
```

```

completions: 3
template:
  metadata:
    name: ebdn-cuenta
  spec:
    containers:
      - name: cuenta-secuencial
        image: centos:7
        command:
          - "bin/bash"
          - "-c"
          - "for i in {0..15} ; do echo $i ; sleep 1; done"
    restartPolicy: Never

```

- Jobs en paralelo en una cola de trabajo:
 - Si añadimos la propiedad parallelism, conseguimos que el número de tareas elegido se pueda paralelizar hasta el número especificado en esta propiedad.

```

apiVersion: batch/v1
kind: Job
metadata:
  name: ebdn-cuenta
spec:
  completions: 10
  parallelism: 3
  template:
    metadata:
      name: ebdn-cuenta
    spec:
      containers:
        - name: cuenta-secuencial
          image: centos:7
          command:
            - "bin/bash"
            - "-c"
            - "for i in {0..15} ; do echo $i ; sleep 1; done"
    restartPolicy: Never

```

Por otra parte, tenemos los Cronjobs. En una infraestructura tradicional linux, implementaríamos un CronJob a través de Crontab y el demonio Cron. En Kubernetes usaríamos el tipo de objeto CronJob para implementarlo a través de un fichero YAML.

Elementos de un CronJob

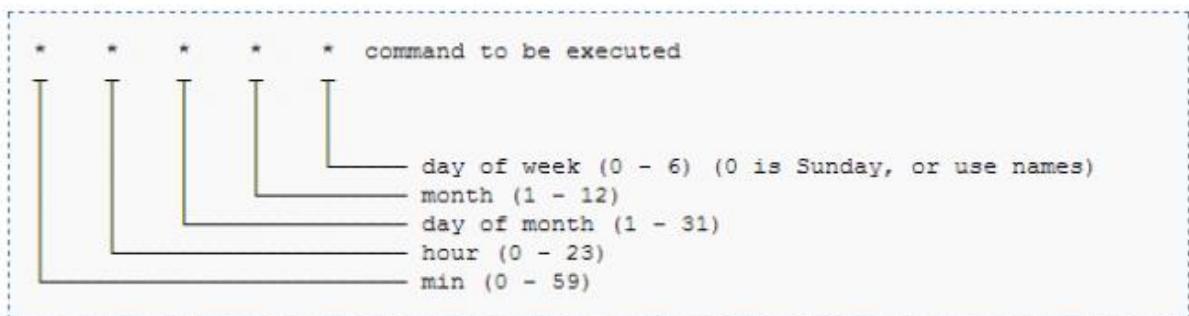
En Kubernetes, y en la mayoría de los CronJobs utilizados, se incluyen los siguientes componentes principales:

- La secuencia de comandos que será llamada a hacer o ejecutada
- El comando que ejecuta la secuencia de comandos en forma recurrente

- La acción o la salida de la secuencia de comandos (que depende de lo que la secuencia de comandos es llamada a hacer)

A nivel de fichero YAML tendremos varios campos principales:

- **schedule:** Donde se colocaría la programación y donde se pueden usar los siguientes caracteres especiales:
 - ? : Valor comodín, que coincide con un solo carácter
 - * : Es otro valor comodín que coincide con cero o más caracteres
 - / : Permite especificar un intervalo para un campo. Por ejemplo, el valor minuto, de la imagen anterior, está en */1 se interpreta que se va a ejecutar cada minuto. Si lo utilizamos en el campo día de la semana con el valor 0/4, se interpreta que se ejecuta cada cuarto domingo.



- **jobTemplate:** donde describimos lo que hace el CronJob. En este apartado incluimos las imágenes de contenedores a utilizar, los comandos que se van a ejecutar dentro de contenedor y las políticas de reinicio del CronJob.
- **startingDeadlineSeconds:** sería el valor máximo en segundos que un CronJob puede tardar en iniciarse si tiene un fallo y no se ejecuta a la hora-fecha programada. Si no le pasamos este valor, si se da un error nunca se agotará el tiempo, y puede que se ejecute nuestro CronJob varias veces de forma simultánea.
- **concurrencyPolicy:** para evitar que ante un fallo el CronJob se ejecute varias veces simultánea existe la política de simultaneidad. Los valores admitidos serían:

Valor	Significado
Allow	Se permiten trabajos simultáneos. Esta es la opción predeterminada.
Forbid	No se permiten trabajos simultáneos; los trabajos nuevos no pueden comenzar hasta que los anteriores se hayan completado o se haya agotado el tiempo de espera.
Replace	No se permiten trabajos simultáneos; los trabajos anteriores se cancelan en favor de los nuevos.

- **restartPolicy:** la política de reinicio del Pod (valores Always, OnFailure o Never)

Ejemplo CronJob

```
apiVersion: batch/v1beta1
kind: CronJob
metadata:
  name: nginx
spec:
  schedule: "*/1 * * * *"
  startingDeadlineSeconds: 100
  jobTemplate:
    spec:
      template:
        spec:
          volumes:
            - name: negu-volume
              hostPath:
                path: /mnt/web
                type: Directory
          containers:
            - name: nginx
              image: ubuntu:latest
              command: ["/bin/sh"]
              args: ["-c", "echo '.' >> /usr/share/nginx/html/index.html"]
              volumeMounts:
                - mountPath: "/usr/share/nginx/html"
                  name: negu-volume
    restartPolicy: OnFailure
```

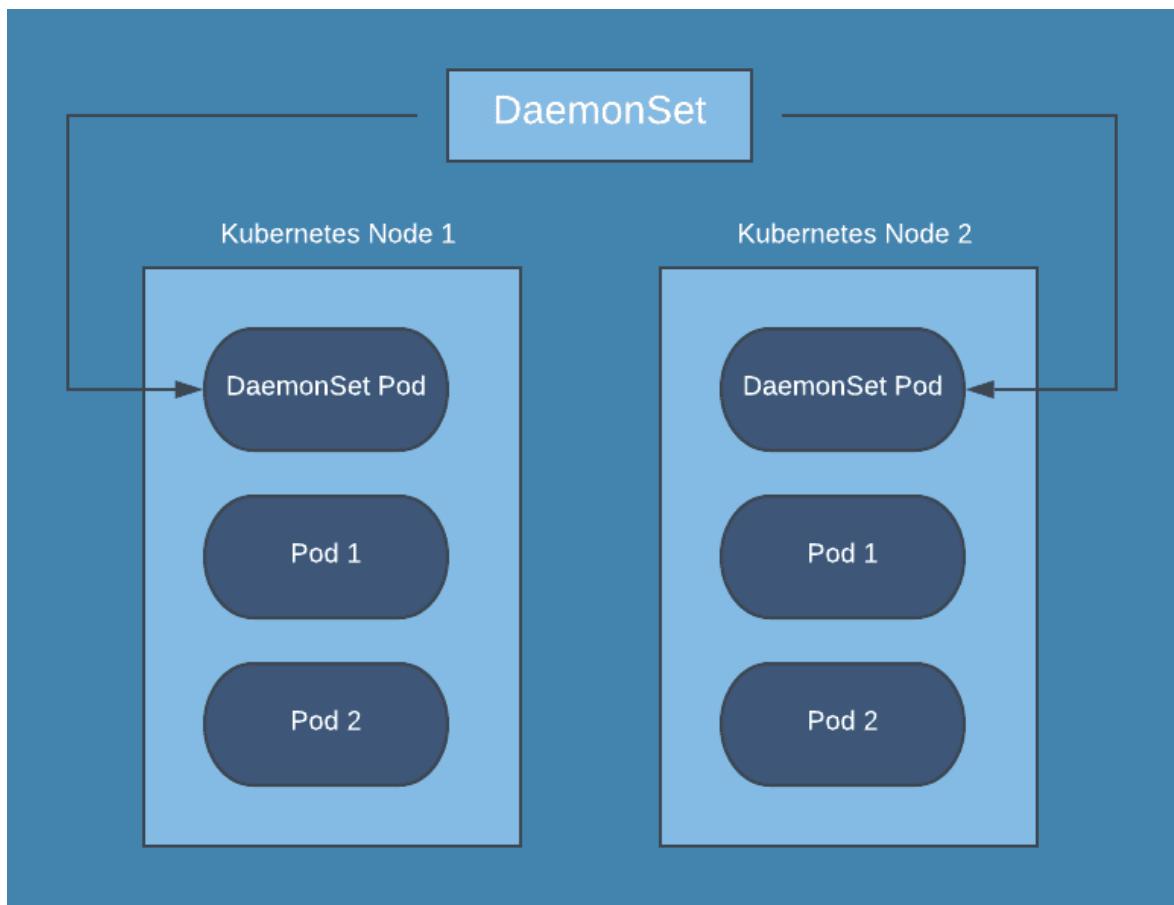
Podéis ver el ejemplo completo en el "ANEXO II. Caso práctico. **Ejemplo de Aplicación con CronJob**"

DAEMONSET

Seguimos hablando de Pods y sus objetos asociados. En este caso vamos a explicar otro tipo de objeto llamado DaemonSet.

Como ya hemos explicado brevemente, **DaemonSet** es una forma de garantizar que cada nodo ejecute una instancia de un Pod, lo que nos garantiza ejecutar un daemon en cada nodo.

Se utiliza para servicios de clúster, como monitoreo de salud, reenvío de registros, centralizar logs... Por ejemplo, si aumentamos los hosts de nuestro clúster, se generaría un Pod en cada host agregado, y si eliminamos hosts, los Pods se eliminarían. Y por consecuencia, cuando eliminamos un DaemonSet.



¿Cuándo usar DaemonSet?

Los usos más comunes para DaemonSet serían:

- Supervisión de nodos:
 - Prometheus
- Si necesitamos generar daemons para un clúster de almacenamiento como:
 - CephFS
 - GlusterFS

- Necesitamos gestionar logs centralizados:
 - Logstash
 - Fluentd
 - Loggly

Trabajar con DaemonSet

Los DaemonSet nos dan ventajas como hacer un rápido rollback rápido en todo nuestro clúster.

Si queréis listar vuestros DaemonSet podéis hacerlo mediante el comando

kubectl get daemonsets --all-namespaces

O de forma abreviada:

kubectl get ds --all-namespaces

Los ficheros YAML que lanzaremos para generar nuestros DaemonSet deben constar de los siguientes campos:

```
apiVersion: apps/v1
kind: DaemonSet
metadata:
  name: ebdn-daemonset
  namespace: ebdn-namespace
  Labels:
    key: value
spec:
  template:
    metadata:
      labels:
        name: ebdn-daemonset-container
    ...
  selector:
    matchLabels:
      name: ebdn-daemonset-container
```

- **apiVersion** (Requerido)
- **kind** (Obligatorio): debe ser DaemonSet
- **metadata** (Requerido)
- **spec.template** (Obligatorio): Haríamos una definición del Pod que deseamos ejecutar en todos los nodos

- **spec.selector** (Obligatorio): Un selector de Pods administrado por DaemonSet. Este valor debe ser una de las etiquetas especificadas en la plantilla de Pod (en este ejemplo por nombre). Este valor no se puede cambiar después de haber creado el DaemonSet sin dejar huérfanos los Pods creados por el DaemonSet.

Adicionalmente, podemos usar:

- **spec.template.spec.nodeSelector**: Se puede usar para ejecutar sólo un subconjunto de nodos que coinciden con el selector
- **spec.template.spec.affinity**: Se puede usar para ejecutar sólo un subconjunto de nodos que coincidan con la afinidad

Para aplicarlo haremos lo siguiente:

```
kubectl apply -f ecdn-daemonset.yaml
```

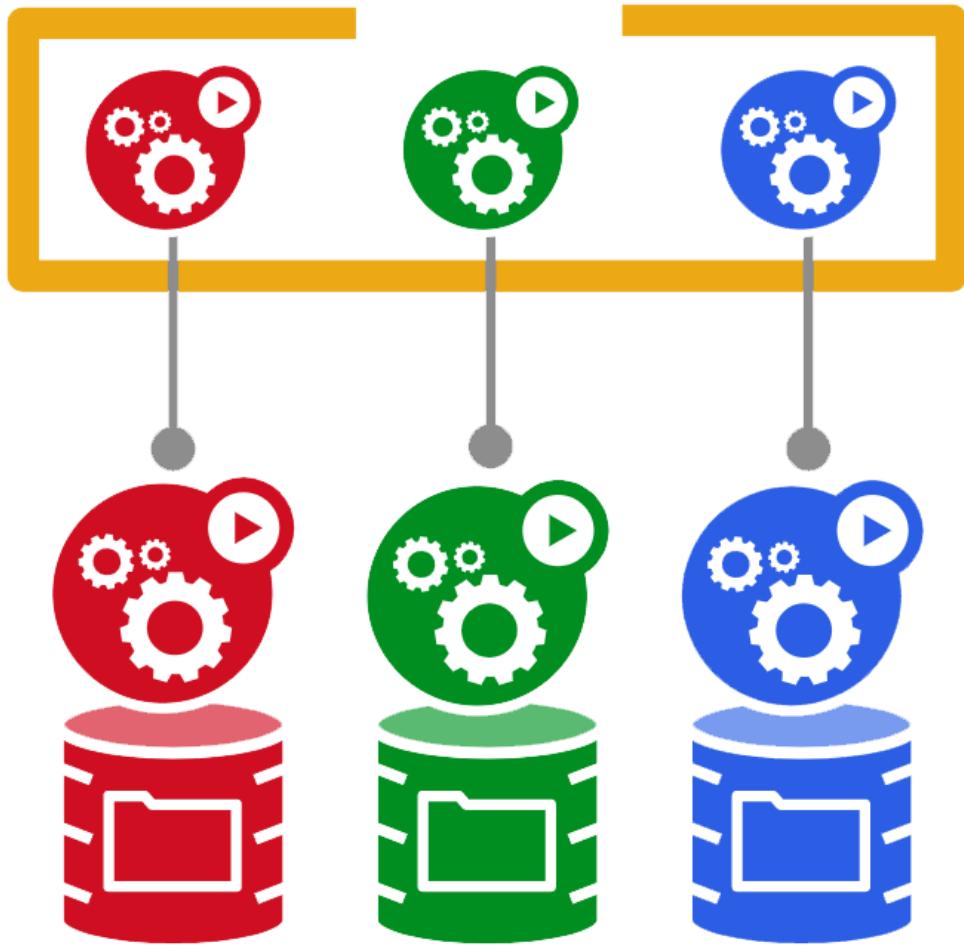
STATEFULSET

Siempre que hablamos de contenedores, lo relacionamos con la existencia efímera para la cual se desarrollaron. Eso puede ser una ventaja muchas veces, porque hace que nuestras aplicaciones sean mucho más ágiles y escalables.

Eso en el mundo real, saliendo de labs y pruebas varias, no siempre es la opción que necesitamos, sobre todo en entornos productivos.

Un ejemplo de este tipo de aplicaciones, son las bases de datos, que implementadas en un clúster Kubernetes necesitan identidades exclusivas y persistentes.

Para solucionar esto, existe **StatefulSet**, que están disponibles desde la versión 1.9 de Kubernetes. Permite implementar aplicaciones con estado, dotar a nuestras aplicaciones de persistencia, identificadores estables y un orden a la hora de gestionar las modificaciones de nuestros contenedores.



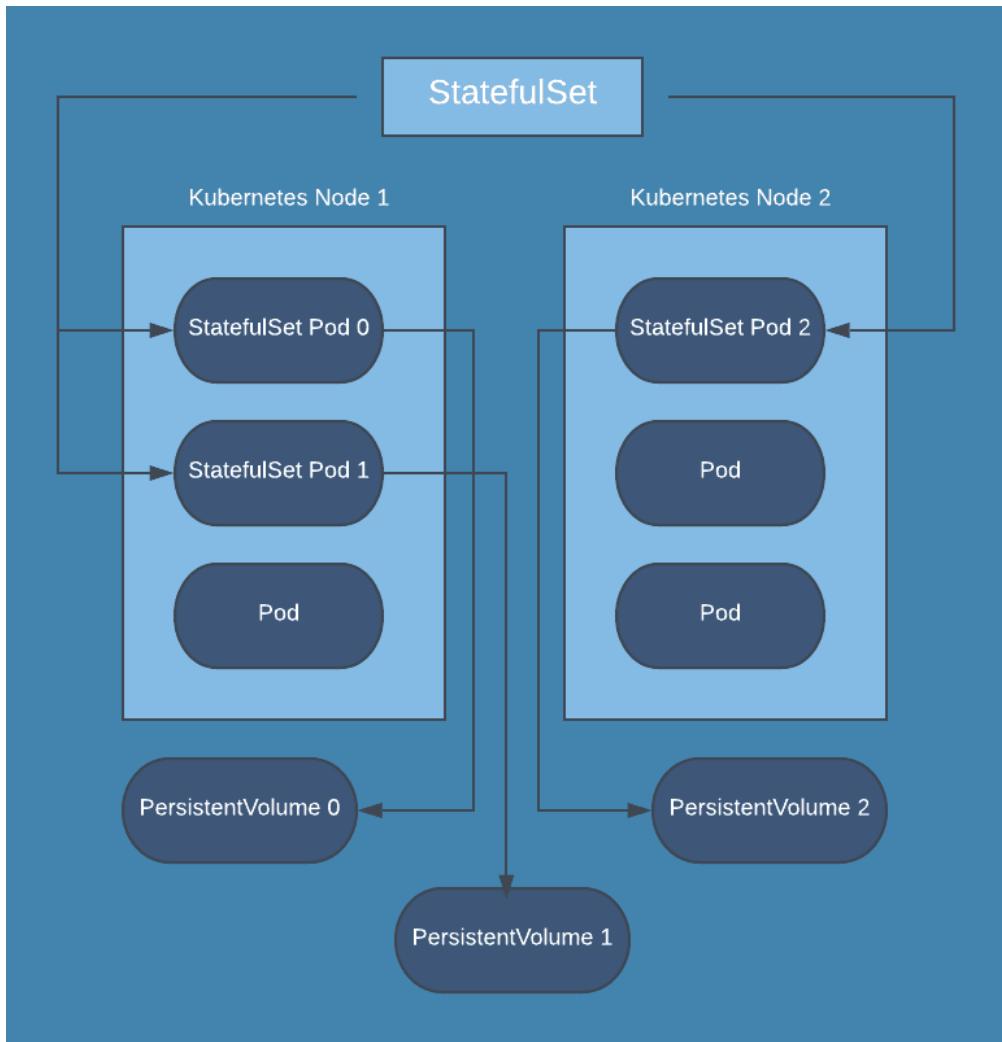
¿Cómo funciona esta persistencia? La información del estado y otros datos de cualquier **Pod** del **StatefulSet** se conservan en el almacenamiento del disco persistente asociado.

Cuando se generan **Pods** con **StatefulSet**, cada Pod tiene asignado un número ordinal (pertenece a una sucesión ordenada) y adicionalmente un ID de red.

A su vez, se puede usar **volumeClaimTemplates** para generar un volumen persistente individual para cada **Pod**.

¿En qué casos usaríamos StatefulSet? Algunos ejemplos para usar StatefulSet podrían ser:

- Un pod de base de datos (MariaDB, Redis o Apache Cassandra, por ejemplo) con acceso a un volumen. En este caso, se desea que mantenga el acceso al mismo volumen incluso si se vuelve a implementar o se reinicia
- Otro ejemplo, sería un clúster de BBDD con varios nodos y que se necesita que cada nodo mantenga el acceso a sus datos
- Una aplicación web que necesita comunicarse con sus réplicas utilizando identificadores de red conocidos y predefinidos



Actualización de un StatefulSet

StatefulSet dispone de dos formas de gestionar sus actualizaciones, que ahora vamos a explicar, **OnDelete** y **RollingUpdate**.

Esto se hace ejecutando el mismo comando que en su despliegue, con los cambios dentro.

```
kubectl apply -f statefulset.yaml
```

OnDelete

Para aplicar una actualización **OnDelete** se usa el valor **.spec.updateStrategy.type**

En este caso, los **Pods** no se reemplazarán cuando se ejecuta. En su lugar, hay que eliminarlos manualmente antes de crear la nueva versión.

RollingUpdate

En el caso de **RollingUpdate**, por el contrario, los pods asociados al **StatefulSet** se eliminarán y luego se reemplazarán en orden ordinal inverso.

Con esta estrategia de actualización, también puedes especificar **.spec.updateStrategy.rollingUpdate.partition** a un valor ordinal, y todos los **Pods** con un

valor ordinal más alto se reemplazarán con la nueva versión, mientras que las antiguas se conservarán.

De esta forma, te permite realizar implementaciones por fases.

Ejemplo de StatefulSet

Este ejemplo primero se crea un servicio que puede ser utilizado por StatefulSet, y luego crea un StatefulSet que mantiene un pod Redis conectado a un volumen. Para lanzarlo usaremos el siguiente comando:

```
kubectl apply -f statefulset.yaml
```

Que tendrá el siguiente contenido:

```
apiVersion: v1
kind: Service
metadata:
  name: redis
  namespace: default
  labels:
    app: redis
spec:
  ports:
  - port: 6379
    protocol: TCP
  selector:
    app: redis
  type: ClusterIP
  clusterIP: None
---
apiVersion: apps/v1
kind: StatefulSet
metadata:
  name: redis
spec:
  selector:
    matchLabels:
      app: redis
  serviceName: "redis"
  replicas: 1
  template:
    metadata:
      labels:
        app: redis
  spec:
    containers:
    - name: redis
      image: redis:6.0.1
      command: ["redis-server", "--appendonly", "yes"]
      ports:
      - containerPort: 6379
```

```

        name: web
      volumeMounts:
        - name: redis-aof
          mountPath: /data
    volumeClaimTemplates:
    - metadata:
        name: redis-aof
      spec:
        accessModes: [ "ReadWriteOnce" ]
        storageClassName: "gp2"
        resources:
          requests:
            storage: 1Gi

```

OTRO EJEMPLO:

```

apiVersion: v1
kind: Service
metadata:
  name: nginx
  labels:
    app: nginx
spec:
  ports:
    - port: 80
      name: web
  clusterIP: None
  selector:
    app: nginx
---
apiVersion: apps/v1
kind: StatefulSet
metadata:
  name: web
spec:
  selector:
    matchLabels:
      app: nginx # Label selector that determines which Pods belong
      to the StatefulSet
      # Must match spec: template: metadata: labels
  serviceName: "nginx"
  replicas: 3
  template:
    metadata:
      labels:
        app: nginx # Pod template's label selector
  spec:
    terminationGracePeriodSeconds: 10
    containers:
      - name: nginx
        image: gcr.io/google_containers/nginx-slim:0.8
        ports:
          - containerPort: 80
            name: web
        volumeMounts:

```

```

    - name: www
      mountPath: /usr/share/nginx/html
  volumeClaimTemplates:
  - metadata:
      name: www
    spec:
      accessModes: [ "ReadWriteOnce" ]
      resources:
        requests:
          storage: 1Gi

```

SERVICIOS

Los Servicios son objetos que permiten reenviar tráfico de red a un conjunto de Pods, lo cual nos permite acceder a nuestras aplicaciones. Los Servicios se implementan a través de iptables, cuando se genera un servicio se le asigna una IP virtual interna (ClusterIP) que permite la interconexión entre Pods y es el componente kube-proxy de Kubernetes el que se comunica con la API Server que es el que comprueba si se han generado nuevos servicios.

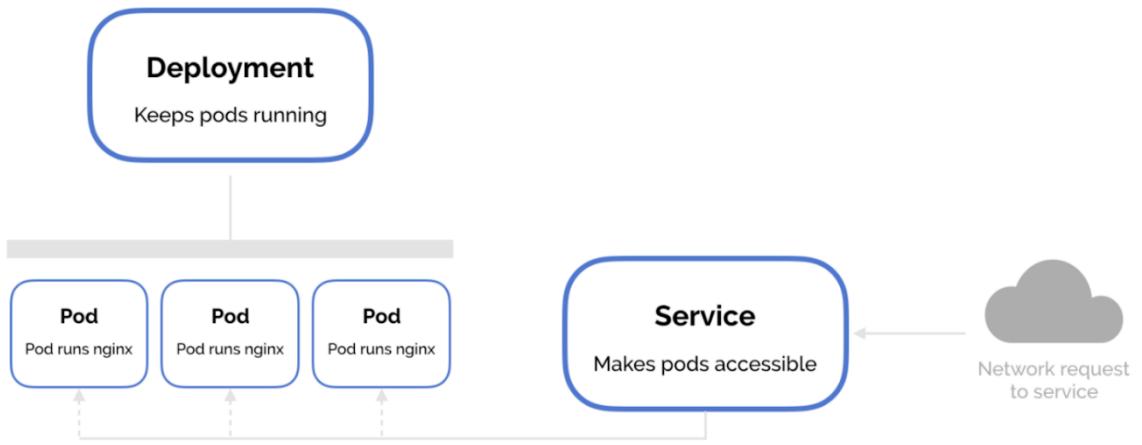
Para definir qué Pods usan qué Servicios, podéis pensar que se usan direcciones IP estáticas, pero no es así, se usan los selectores o etiquetas que definen a cada conjunto de Pods.

Son asignaciones dinámicas, y permite que las nuevas versiones o añadir nuevos Pods a un Servicio sea realmente fácil. Es decir, si activamos un Pod con las mismas etiquetas que un Servicio, se asigna automáticamente a ese Servicio.

¿Qué diferencia hay entre Deployment y Servicio?

La diferencia fundamental está en su funcionalidad. Como hemos dicho, un Servicio permite el acceso a la red a un conjunto de Pods (actuando como un proxy). Y un Deployment se utiliza para mantener un conjunto de Pods en ejecución mediante la creación de Pods a partir de una plantilla.

Ambos trabajan con etiquetas o selectores con los Pods. Y se pueden generar independientemente el uno de otro, o trabajar en conjunto, que es la forma habitual de trabajar.



Tipos de Servicios

Existen varios tipos de servicios en Kubernetes, los principales y que vamos a explicar son:

- ClusterIP
- NodePort
- LoadBalancer
- Ingress, que, aunque no es un servicio propiamente, tiene una gran relación

Un poco de networking para Kubernetes

Uno de los puntos que más nos cuesta entender en las infraestructuras con microservicios es la red y como trabajamos con nuestro clúster de contenedores y kubernetes.

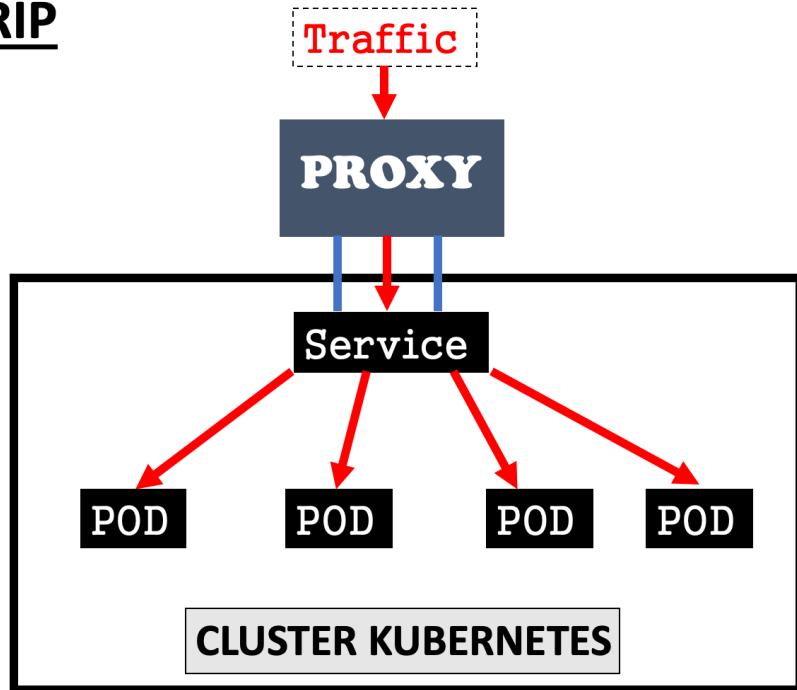
Lo que vamos a intentar explicar es la diferencia entre ClusterIP, LoadBalancer, Ingress y NodePort en Kubernetes. Y a su vez en qué casos usaríamos cada uno.

Para empezar, hablaremos de ClusterIP, que es el servicio que se genera de forma predeterminada y que nos permite acceder a los servicios dentro del clúster.

De forma predeterminada, este servicio no es accesible desde Internet. Para acceder necesitaríamos habilitar el acceso a través del proxy de Kubernetes con el siguiente comando:

```
kubectl proxy --port = 8080
```

CLUSTERIP



Lo que hay que entender es que se permite acceder a nuestra API, utilizando el siguiente esquema:

<http://localhost:8080/api/v1/proxy/namespaces/<NAMESPACE>/services/<SERVICE-NAME>:<PORT-NAME>/>

EJEMPLO:

<http://localhost:8080/api/v1/proxy/namespaces/default/services/negulab-clusterip:http/>

```
apiVersion: v1
kind:
metadata:
  name: negulab-clusterip
spec:
  selector:
    app: app
  type: ClusterIP
  ports:
  - name: http
    port: 80
    targetPort: 80
    protocol: TCP
```

Usar ClusterIP + Proxy es interesante para realizar pruebas internas, administración o para permitir tráficos internos, pero no es lo adecuado para una plataforma productiva, ya que para lanzar kubectl debemos autenticarnos en nuestra infraestructura previamente.

Así que, si ClusterIP no nos sirve para Producción, qué podemos usar...

NODEPORT

Una alternativa es NodePort, que es otro tipo de servicio como ClusterIP.

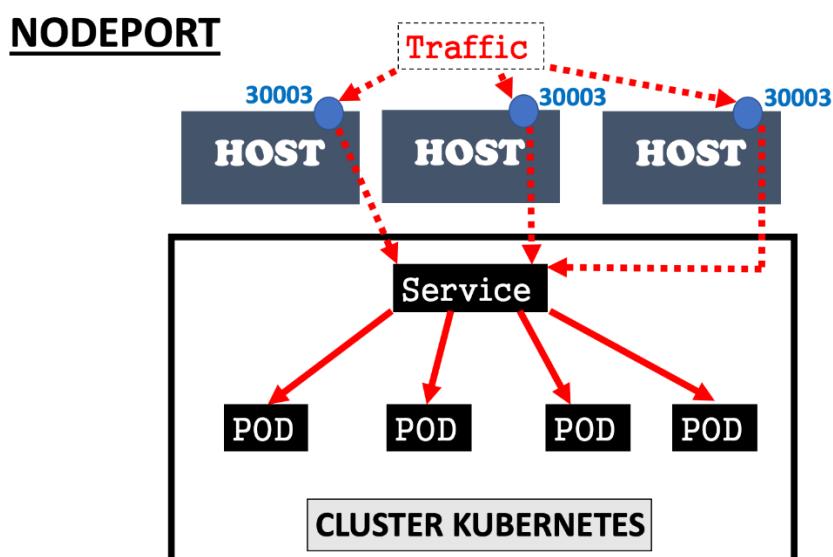
Al utilizar NodePort abrimos un puerto específico en todos los hosts, y se reenvía el tráfico directamente al servicio, y a los Pods.

El puerto se especifica en el fichero YAML que crea el servicio, y si no se especifica, se utilizará automáticamente un puerto libre.

```
apiVersion: v1
kind: Service
metadata:
  name: elblogdenegu-nodeport-service
spec:
  selector:
    app: app-ebdn
  type: NodePort
  ports:
  - name: http
    port: 80
    targetPort: 80
    nodePort: 30003
    protocol: TCP
```

Aunque parece sencillo, tiene ciertas limitaciones:

- Sólo se pueden usar puertos en el rango 30000-32767
- Sólo un servicio por puerto
- No es para ambientes en Producción, pero si para hacer una demo de una aplicación, por ejemplo
- Te puede dar problemas con las IPs de los hosts, ya que estamos exponiendo el puerto en cada nodo. Las solicitudes desde fuera serían NodeIP:NodePort, entonces para solucionar esto necesitaríamos implementar un balanceador de carga tipo HAProxy para compensarlo



LOADBALANCER

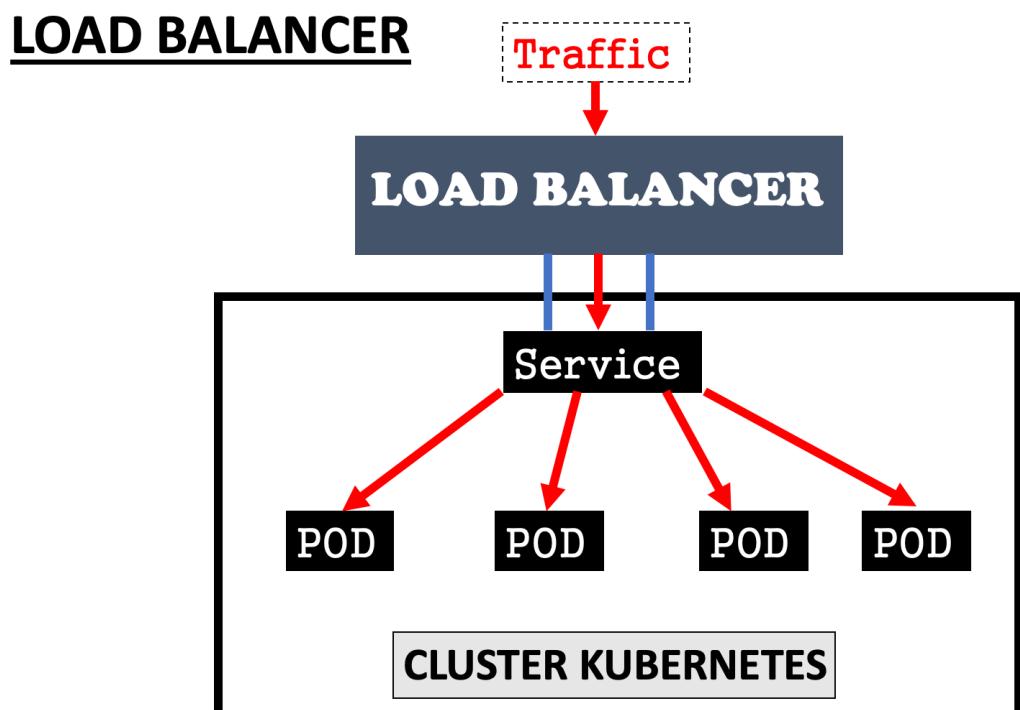
Otro tipo de servicio que podemos implementar es LoadBalancer, que es el modo más estándar de exponer un servicio en Internet.

Al implementarlo se activa un balanceador de carga que proporcionará una única dirección IP que reenvía todo el tráfico a su servicio.

Este tipo de servicio tiene ciertas ventajas y algún inconveniente:

- Es ideal para exponer un servicio concreto, pero no podremos exponer varios servicios a través de él. Con lo que, si queremos exponer varios servicios, tendremos que usar varios LoadBalancer, lo que, en un cloud, significa dinero.
- En este tipo de servicio no hay filtrado ni enrutamiento...

Se podría implementar un proyecto de página web, sabiendo que nuestro clúster va a utilizar un puerto específico HTTP o HTTPS para exponerla a Internet.



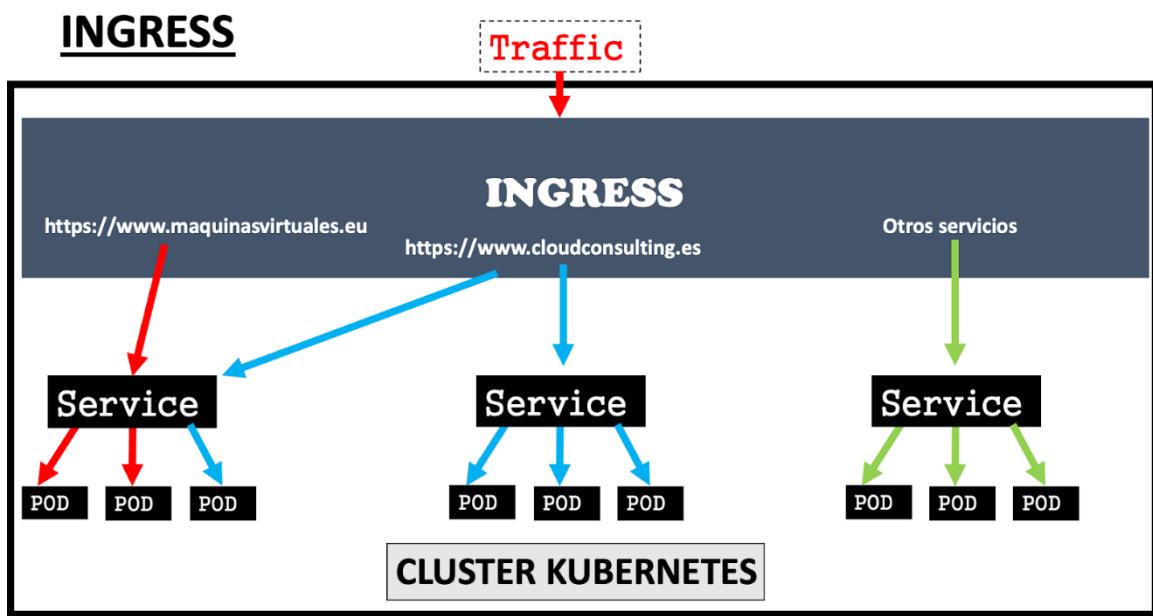
INGRESS

Por último, hablaremos de la forma más completa de exponer un servicio, Ingress.

Ingress no es un tipo de servicio como el resto, se trata más de un enrutador que permite la entrada al clúster y gestionar el acceso a múltiples servicios.

Ingress tiene ciertas ventajas:

- Dispone de características o complementos avanzados como SSL, Routing, Subdominios para servicios en Backend, ...
 - Con respecto a LoadBalancer, sólo necesitas una IP para exponer varios servicios. Esto se traduce en menos dinero al contratarlo en servicios cloud, por ejemplo
 - Dispone de soporte para平衡adores de carga nativos de la nube (de Google, Amazon y Microsoft)
 - Ingress permite configuraciones basadas en tiempos de espera o limitaciones de velocidad y enruteamientos basados en contenido, autenticación y mucho más.
- Entre los pequeños inconvenientes el más importante:
- Es más complejo de implementar que el resto de las soluciones



EJEMPLO:

```
apiVersion: extensions/v1beta1
kind: Ingress
metadata:
  name: ebdn-ingress
spec:
  backend:
    serviceName: ebdn
    servicePort: 8082
  rules:
  - host: www.máquinasvirtuales.eu
    http:
      paths:
```

```

    - backend:
        serviceName: negu
        servicePort: 8082
    - host: cloudconsulting.es
      http:
        paths:
        - path: /*
          backend:
            serviceName: cloud
            servicePort: 8082

```

ROLLING UPDATE Y ROLLBACKS

Una de las mayores ventajas que veo al usar Kubernetes, es la facilidad de hacer despliegues de nuevas versiones de aplicaciones (Updates) y poder volver a atrás (Rollbacks) relativamente rápido si hay un problema.

Lo vamos a explicar con un despliegue clásico a través de un Deployment de Nginx. Imaginamos que tenéis corriendo un contenedor Nginx. Generamos un contenedor mediante un Deployment que se llama "nginx":

```

raulunzue@KBMMASTER:~$ kubectl create deployment nginx --
image=nginx:1.17.9
deployment.apps/nginx created

```

Revisamos en el propio contenedor que la versión es la adecuada:

```

raulunzue@KBMMASTER:~$ kubectl exec -it nginx-5c95c56f6f-h4767 --
/bin/bash
root@nginx01-nginx-5c95c56f6f-h4767:/# nginx -v
nginx version: nginx/1.17.9

```

Ahora vamos a forzar un update a la versión 1.17.10:

```

raulunzue@KBMMASTER:~$ kubectl set image deploy nginx
nginx=nginx:1.17.10 --record
deployment.apps/nginx image updated

```

Si observamos el comportamiento, la versión antigua del contenedor, se muere y arranca un nuevo contenedor con la nueva versión:

```

raulunzue@KBMMASTER:~$ kubectl get pods -o wide

```

NAME	READY	STATUS		
RESTARTS	AGE	IP	NODE	
nginx-5c95c56f6f-h4767	1/1	Terminating	0	3m3s
10.69.2.132	kubernetes02			
nginx-5df596bbf9-rmr9x9	1/1	Running	0	
14s	10.69.5.63	kubernetes01		

Comprobamos la versión nuevamente, pero del nuevo contenedor:

```
raulunzue@KBMMASTER:~$ kubectl exec -it nginx-5df596bbf9-rmr9x9 -- /bin/bash
root@nginx01-nginx-5df596bbf9-rmr9x9:/# nginx -v
nginx version: nginx/1.17.10
```

Si queremos volver a la versión anterior, podemos consultar el histórico mediante:

```
raulunzue@KBMMASTER:~$ kubectl rollout history deployment nginx
deployment.apps/nginx
REVISION  CHANGE-CAUSE
1          <none>
2          kubectl set image deploy nginx nginx=nginx:1.17.10 --
record=true
```

Observaremos al hacer el rollback con el siguiente comando, que el proceso es a la inversa, se termina el nuevo contenedor y se ejecuta otro con la versión antigua:

```
raulunzue@KBMMASTER:~$ kubectl rollout undo deployment nginx --to-revision=1
deployment.apps/nginx rolled back
```

```
raulunzue@KBMMASTER:~$ kubectl rollout undo deployment nginx --to-revision=1
deployment.apps/nginx rolled back
raulunzue@KBMMASTER:~$ kubectl get pods -o wide
NAME           READY   STATUS      RESTARTS   AGE     IP       NODE   NOMINATED NODE   READINESS GATES
nginx-5c95c56f6f-8rd49   0/1   ContainerCreating   0        4s    <none>   kubernetes02   <none>   <none>
nginx-5df596bbf9-rmr9x9   1/1   Running    0        8m57s  10.69.5.63  kubernetes01   <none>   <none>
nginx01-b55d67df6-q19vw   1/1   Running    2        31d   10.69.2.130  kubernetes02   <none>   <none>
sealed-secrets-1591042254-7666b747cd-grs6v  0/1   CrashLoopBackOff  1684   5d22h  10.69.5.55  kubernetes01   <none>   <none>
raulunzue@KBMMASTER:~$ kubectl get pods -o wide
NAME           READY   STATUS      RESTARTS   AGE     IP       NODE   NOMINATED NODE   READINESS GATES
nginx-5c95c56f6f-8rd49   1/1   Running    0        8s    10.69.2.133  kubernetes02   <none>   <none>
nginx-5df596bbf9-rmr9x9   0/1   Terminating 0        9ms   <none>   kubernetes01   <none>   <none>
nginx01-b55d67df6-q19vw   1/1   Running    2        31d   10.69.2.130  kubernetes02   <none>   <none>
sealed-secrets-1591042254-7666b747cd-grs6v  0/1   CrashLoopBackOff  1684   5d22h  10.69.5.55  kubernetes01   <none>   <none>
```

Volvemos a comprobar la versión de Nginx:

```
raulunzue@KBMMASTER:~$ kubectl exec -it nginx-5c95c56f6f-8rd49 -- /bin/bash
root@nginx-5c95c56f6f-8rd49:/# nginx -v
nginx version: nginx/1.17.9
```

Si os parece interesante, el poder es que lo podéis hacer con otros objetos como deployments, daemonset...

EJEMPLO PRÁCTICO

Lo que haríamos es preparar dos ficheros YAML. Uno tendría la instalación de contenedores en la versión

```
raulunzue@KBMMASTER:~$ cat nginx-deployment-elblogdenegu.yml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment-elblogdenegu
  labels:
    app: nginx
spec:
  replicas: 6
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - name: nginx
          image: nginx:1.17.0
          ports:
            - containerPort: 8080
  nodeSelector:
    node: kb01
```

Y otro fichero con el Update:

```
raulunzue@KBMMASTER:~$ cat nginx-deployment-elblogdenegu-update.yml
apiVersion: apps/v1
kind: Deployment
metadata:
```

```

name: nginx-deployment-elblogdenegu
labels:
  app: nginx
spec:
  replicas: 6
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - name: nginx
          image: nginx:nginx:1.17.10
          ports:
            - containerPort: 8080
  nodeSelector:
    node: kb01

```

ETIQUETAS, ESPACIOS DE NOMBRES, CONTEXTO Y ANOTACIONES

ETIQUETAS

Las Etiquetas o Labels se usan habitualmente en todo tipo de ámbitos de programación, virtualización...

Si has gestionado, o lo pretendes, plataformas de Producción con múltiples Pods, Servicios, Deployments...utilizar etiquetas te ayudarán a gestionarlos más fácilmente.

Unas etiquetas son pares de clave/valor que se asocian a los objetos. Se asocia una etiqueta a un objeto, y luego puedes referenciarlo fácilmente, ya que es un nombre que has elegido tú y no algo automático que genera el sistema, que será casi imposible asociar cuando tienes muchos objetos.

No hay limitación de etiquetas, y casi tenéis total libertad para colocar el nombre de los valores. Os pongo los caracteres admitidos:

- Consta de dos partes, un prefijo opcional y un nombre separados por /
- Nombre: menor o igual a 63 caracteres, tiene que terminar con un carácter alfanumérico ([a-z0-9A-Z]) y en medio del nombre soporta cualquier carácter alfanumérico (incluidos guiones altos-bajos o puntos).

Podremos gestionar subconjunto de objetos o clústeres. Os pongo unos ejemplos tomando de referencia un objeto Pod. Las podéis gestionar a través de la sección metadata al generar el Pod:

```

apiVersion: v1
kind: Pod
metadata:
...
  labels:
    pod-template-hash: "6783451234"
    run: mongo

```

Se pueden añadir etiquetas al aire, aunque ésta no es la mejor práctica, porque si queréis redeployar el Pod, tendréis que volver a asignar la etiqueta (Acordaros de la naturaleza efímera de los contenedores):

```

kubectl label pods mongo app=elblogdenegu
kubectl get pods mongo --show-labels
NAME      READY   STATUS    RESTARTS   AGE   LABELS
mongo     1/1     Running   0          66m   app=elblogdenegu

```

Y una vez asignadas, podemos gestionar nuestros Pods a través de ellas:

```

kubectl get pods -l app=elblogdenegu
NAME      READY   STATUS    RESTARTS   AGE
mongo     1/1     Running   0          66m

kubectl get pods -Lrun
NAME      READY   STATUS    RESTARTS   AGE   APP
mongo     1/1     Running   0          66m   elblogdenegu

```

Podemos también asignar etiquetas a un nodo, porque queremos forzar que los contenedores se ejecuten en uno en concreto. Simplemente, le añadimos una etiqueta al nodo:

```

raulunzue@KBMMASTER:~$ kubectl label nodes kubernetes01 node=kb01
node/kubernetes01 labeled

```

Revisamos que la etiqueta efectivamente existe:

```

raulunzue@KBMMASTER:~$ kubectl describe node kubernetes01
Name:           kubernetes01
Roles:          <none>
Labels:         beta.kubernetes.io/arch=amd64
                beta.kubernetes.io/os=linux
                kubernetes.io/arch=amd64
                kubernetes.io/hostname=kubernetes01
                kubernetes.io/os=linux
Annotations:   node=kb01
                flannel.alpha.coreos.com/backend-data: {"VtepMAC":"ce:5e:9b:d6:4d:00"}
                flannel.alpha.coreos.com/backend-type: vxlan
                flannel.alpha.coreos.com/kube-subnet-manager: true
                flannel.alpha.coreos.com/public-ip: 192.168.2.194
                kubeadm.alpha.kubernetes.io/cri-socket: /var/run/docker-shim.sock

```

Y al generar el Deployment utilizamos nodeSelector para el nodo concreto, con su etiqueta:

```

raulunzue@KBMMASTER:~$ cat nginx-deployment-elblogdenegu.yml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment-elblogdenegu
  labels:
    app: nginx
spec:
  replicas: 6
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - name: nginx
          image: nginx:latest
          ports:
            - containerPort: 8080
  nodeSelector:
    node: kb01

```

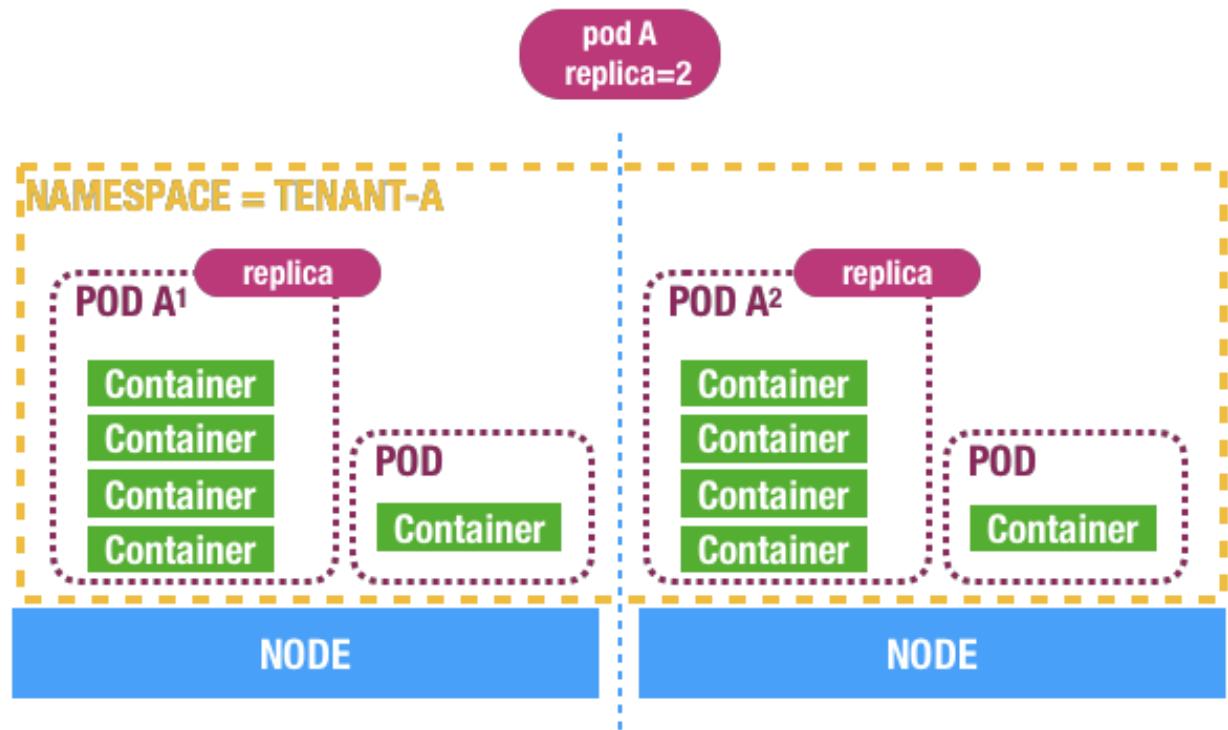
ESPACIOS DE NOMBRES O NAMESPACE

En general, en informática, un *Namespace* o espacio de nombres es un medio para organizar clases dentro de un entorno, agrupándolas de un modo más lógico y jerárquico.

Los Namespace en Kubernetes, por su parte, son clústeres virtuales dentro de un clúster físico. Su objetivo principal es el de proporcionar a múltiples equipos, usuarios y proyectos un

entorno virtualmente separado para trabajar, y evitar que los equipos se interpongan entre sí al limitar lo que los equipos de Kubernetes pueden ver y acceder.

Para aislar los recursos, podemos asignar políticas de acceso y cuotas a nuestro Namespace.



A nivel de operación, para ver los Namespace de los que dispone nuestro clúster a los que nuestro usuario tiene acceso, podemos verlos de la siguiente forma:

```
raulunzue@KBMMASTER:~$ kubectl get namespaces
NAME          STATUS   AGE
default       Active   151d
haproxy-controller   Active   79d
ingress-basic    Active   79d
kube-node-lease  Active   151d
kube-public     Active   151d
kube-system     Active   151d
kubernetes-dashboard Active   151d
```

Existen varias formas de gestionar Namespace, lo podemos hacer vía comando:

```
raulunzue@KBMMASTER:~$ kubectl create ns elblogdenegu01
namespace/elblogdenegu01 created
```

O mediante un fichero YAML:

```
apiVersion: v1
kind: Namespace
metadata:
  name: elblogdenegu01
```

Y una vez generado podemos asignar recursos al definirlos, por ejemplo, con un Deployment (lo mismo pasaría con servicios u otro tipo de objeto):

```
apiVersion: apps/v1beta1
kind: Deployment
metadata:
  name: nginx
  namespace: elblogdenegu01
  ...
  
```

O con un Pod:

```
raulunzue@KBMMASTER:~$ kubectl run nginx --image=nginx -n elblogdenegu01
pod/nginx created
```

Para eliminarlo podemos utilizar:

```
kubectl delete ns elblogdenegu01
```

CONTEXTO

Un contexto determina el clúster y el usuario que podemos utilizar por defecto. Podemos revisar el contexto que nos corresponde de la siguiente forma:

```
raulunzue@KBMMASTER:~$ kubectl config current-context
kubernetes-admin@kubernetes
```

Con el ejemplo del Namespace, lo que podemos hacer es determinar el Namespace antes generado como nuestro contexto por defecto. De tal forma, que el usuario se evita estar introduciendo el -n o la referencia continua al Namespace. Lo haríamos de la siguiente forma:

```
raulunzue@KBMMASTER:~$ kubectl config set-context kubernetes-admin@kubernetes --namespace=elblogdenegu01
Context "kubernetes-admin@kubernetes" modified.
```

ANOTACIONES

Por último, vamos a hablar de las anotaciones, que son pares de clave/valor que se asocian a los objetos como las etiquetas.

Podemos usar las etiquetas o anotaciones para adjuntar metadatos a los objetos de Kubernetes y que nos sea más cómoda su gestión. La diferencia, es que las etiquetas pueden utilizarse para seleccionar objetos y para encontrar colecciones de objetos que satisfacen ciertas condiciones.

Por el contrario, las anotaciones no se utilizan para identificar y seleccionar objetos. Los metadatos de una anotación pueden ser pequeños o grandes, estructurados o no estructurados, y pueden incluir caracteres no permitidos en las etiquetas.

La sintaxis de las anotaciones es prácticamente que la misma que las etiquetas, pero son interesantes para guardar datos en el despliegue de versiones, información autor, insertar números de teléfono del responsable del despliegue...

Todos estos datos, si no los pasas con anotaciones, los tienes que guardar en un recurso externo. Se definen de la siguiente forma:

```
"metadata": {
    "annotations": {
        "key1" : "value1",
        "key2" : "value2"
    }
}
```

VOLÚMENES

Si habéis llegado hasta este capítulo, tendréis claro que los Pods tienen una naturaleza efímera, por eso necesitamos usar volúmenes para que los datos persistan en el tiempo.

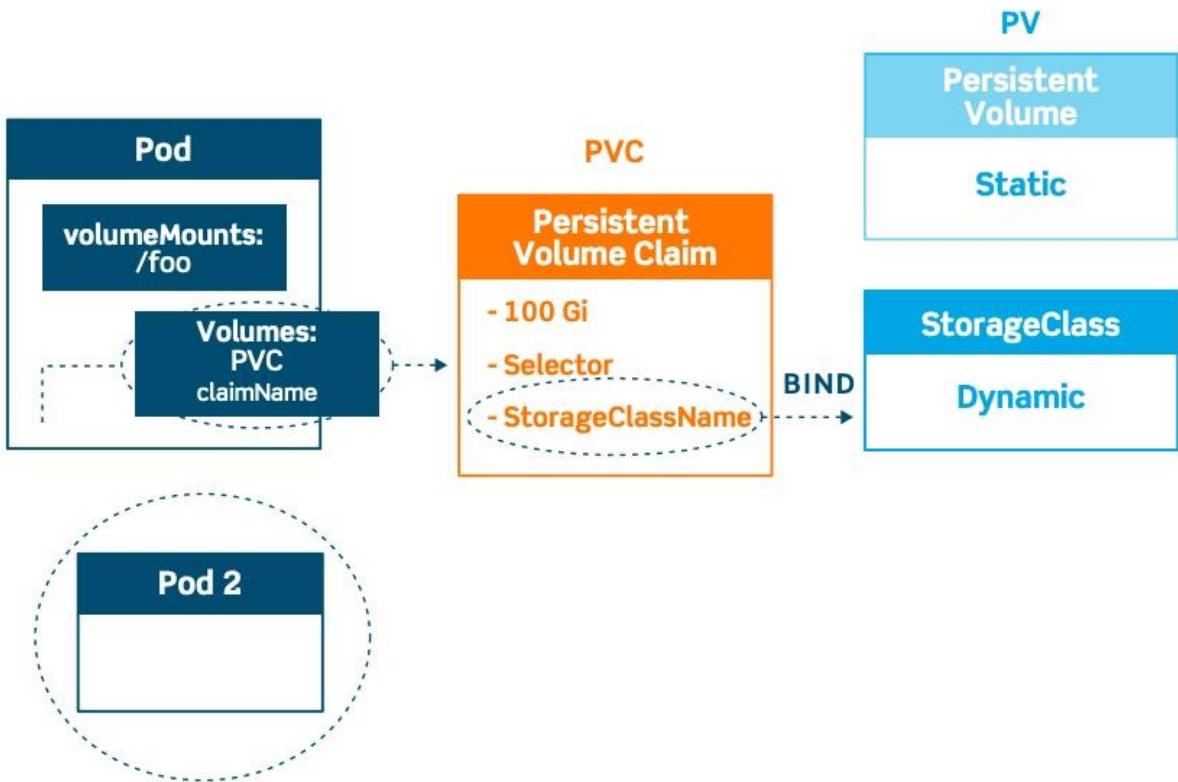
Kubernetes usa el concepto de volúmenes para referirse a los tipos de almacenamientos que podemos implementar.

A nivel funcional, podríamos decir que un volumen no es más que un directorio al que es capaz de acceder un Pod. Estos directorios se definen a la hora de generar los Pods.

Podemos generar muchos tipos de volúmenes, pero os pongo los ejemplos más básicos, para más información podéis revisar (<https://kubernetes.io/docs/concepts/storage/volumes/#types-of-volumes>):

- **hostPath:** Este tipo de volumen se refiere a una carpeta específica del host. Sólo sería interesante montarlo si es un recurso compartido por todos los hosts del clúster, montado de la misma forma en el sistema. Ejemplo: un share de un NAS (NFS, Samba,...), montado en un directorio concreto como /mnt/NAS. Si no se hace así, sólo será accesible en el host donde esté corriendo el Pod, lo que hará que tengamos un problema si el Pod corre en otro host que no tiene la misma información en el recurso.
- **gitRepo:** Podemos usar nuestro repositorio online de GitHub para montar el volumen sobre él directamente.
- **emptyDir:** Cuando montamos un tipo de volumen emptyDir, cuando eliminemos el Pod se borrará automáticamente el volumen con él. Es interesante para compartir información entre contenedores.
- **configMap:** Por último, ya que esto es sólo una aproximación a los volúmenes, vamos a hablar de configMap. Que nos permite injectar datos de configuración e incluso archivos a los Pods. Por ejemplo, la ruta al log del Pod o ficheros .conf para diferentes aplicaciones que no nos dejarían de otra forma.
- **persistentVolumenClaim (PVC) / persistentVolumen (PV):** si lo que necesitamos son volúmenes persistentes usaríamos PV o PVC. En el fondo se parece mucho a un recurso iSCSI, que tú usas, teniendo una capa de abstracción del sistema que lo facilita (Windows, Linux, NAS, Cabina,...). Este tipo de volumen no están asignados por defecto a un Pod, sino que simplemente es un objeto más que podemos usar. La diferencia entre PV y PVC, es que el primero deberemos asociarlo a un Pod mediante un PVC. El PVC se podría comparar más a un Deployment, que en vez de generar Pods, genera volúmenes. Nosotros llamaremos al PVC para que nos genere un volumen de xGB. Esto lo hace a través de la librería de Kubernetes **storageClassName**, que permite conectarse a la API y generar los storages.

En el enlace anterior podéis ver todas las posibilidades...



Ejemplo de YAML con diferentes volúmenes

Creamos en el host el recurso para el volumen:

```
mkdir /mnt/data
```

Y montamos un fichero html con el siguiente contenido:

```
sudo sh -c "echo 'Hola Kubernetes, prueba storage El Blog de Negu' > /mnt/data/index.html"
```

Primero generaremos el PersistentVolume:

```
apiVersion: v1
kind: PersistentVolume
metadata:
  name: ebdn-pv-volume
  labels:
    type: local
spec:
  storageClassName: manual
```

```

capacity:
  storage: 10Gi
accessModes:
  - ReadWriteOnce
hostPath:
  path: "/mnt/data"

```

Generamos el PersistentVolumeClaim:

```

apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: nginx-pvc
spec:
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 5Gi
  storageClassName: do-block-storage

```

Y luego lanzamos la ejecución del Pod:

```

apiVersion: v1
kind: Pod
metadata:
  name: www
spec:
  containers:
    - name: nginx
      image: nginx
      volumeMounts:
        - mountPath: /home
          name: home
        - mountPath: /git
          name: git
          readOnly: true
        - mountPath: /temp
          name: temp
  volumes:
    - name: home
      hostPath:
        path: /home/debian
    - name: git
      gitRepo:
        repository: https://github.com/elblogdenegu/kubernetes.git
    - name: temp
      emptyDir: {}
    - name: www-volume
      persistentVolumeClaim:
        claimName: nginx-pvc

```

HELM

En este capítulo vamos a hablar de Helm ("Timón"). Se podría definir como un administrador de paquetes para Kubernetes, lo que serían apt o yum en la mayoría de las distribuciones Linux.

Helm implementa Charts, que es una aplicación empaquetada, que nos permite instalar, definir y actualizar aplicaciones complejas en Kubernetes.

Podría decirse que es un gran repositorio de aplicaciones complejas de implementar.

¿Pero quién mantiene Helm? Lo hace la CNCF o Cloud Native Computing Foundation, de la cual son miembros, actores tan relevantes como Google, Microsoft, Bitnami o su propia comunidad.

Utilizando Helm Charts podremos crear, versionar y publicar aplicaciones en Kubernetes.



INSTALAR HELM EN DEBIAN

Instalamos snap:

```
raulunzue@KBMMASTER:~$ sudo apt update  
raulunzue@KBMMASTER:~# sudo apt-get install snap
```

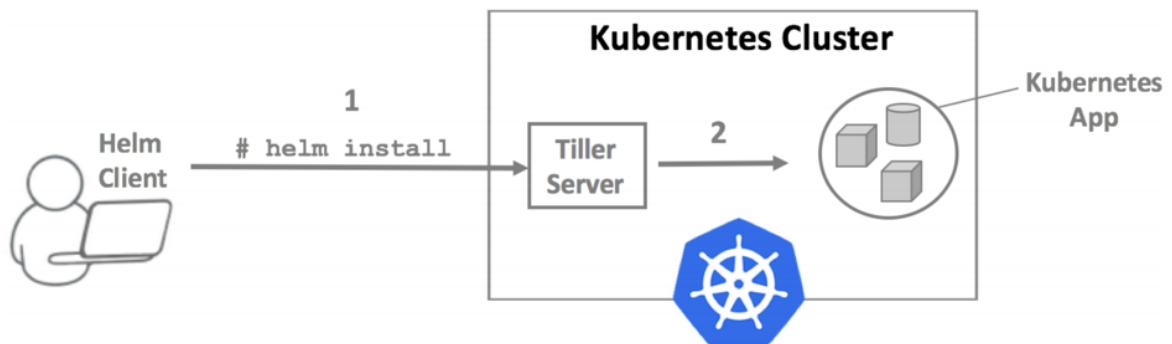
Instalamos Helm:

```
raulunzue@KBMMASTER:~$ sudo snap install helm --classic  
helm 3.1.2 from Snapcrafters installed
```

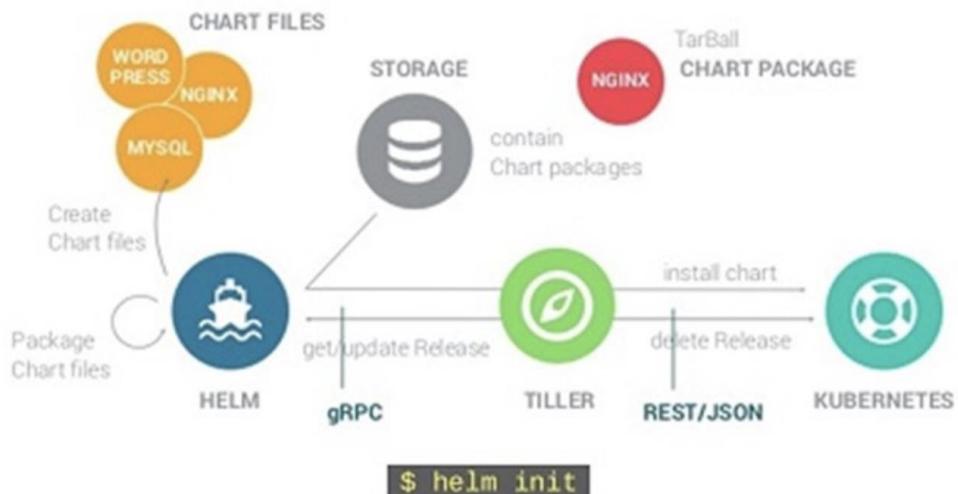
¿QUÉ COMPONENTES TIENE HELM?

Helm es una aplicación cliente-servidor.

- CLIENTE: Helm sería el nombre del cliente
- SERVIDOR: Tiller es un componente que gestiona los Charts, interactuando con la API. Esto nos permite actualizar, consultar, eliminar o instalar los paquetes en Kubernetes.



Helm Architecture



REPOSITORIOS HELM

A continuación, os dejo varios enlaces interesantes de Helm. Los que tienen la marca stable como dice su nombre son los estables, y los que marcan como incubator están en fase de desarrollo:

<https://hub.kubeapps.com/>

Buenas prácticas:

https://helm.sh/docs/chart_best_practices/

Para crear tus propios Helm Charts podéis basaros en la información de GitHub:

<https://github.com/helm/charts>

Ejemplos Helm

Ejemplo de instalación Wordpress con Helm:

```
raulunzue@KBMMASTER:~$ helm search hub wordpress

          URL                                CHART VERSION
APP VERSION DESCRIPTION

  https://hub.helm.sh/charts/bitnami/wordpress      9.3.6      5.
4.1           Web publishing platform for building blogs and ...

  https://hub.helm.sh/charts/presslabs/wordpress-...
v0.8.4        v0.8.4      Presslabs WordPress Operator Helm
Chart

  https://hub.helm.sh/charts/presslabs/wordpress-...
v0.8.5        v0.8.5      A Helm chart for deploying a WordPress site
on ...
```

```
raulunzue@KBMMASTER:~$ helm repo add bitnami
https://charts.bitnami.com/bitnami

"bitnami" has been added to your repositories
```

```
raulunzue@KBMMASTER:~$ helm install mywordpress bitnami/wordpress
```

NAME: mywordpress

LAST DEPLOYED: Sun May 31 09:57:05 2020

NAMESPACE: default

STATUS: deployed

REVISION: 1

NOTES:

** Please be patient while the chart is being deployed **

To access your WordPress site from outside the cluster follow the steps below:

1. Get the WordPress URL by running these commands:

NOTE: It may take a few minutes for the LoadBalancer IP to be available.

Watch the status with: 'kubectl get svc --namespace default -w mywordpress'

```
export SERVICE_IP=$(kubectl get svc --namespace default mywordpress --template "{{ range (index .status.loadBalancer.ingress 0) }}{{.}}{{ end }}")
```

```
echo "WordPress URL: http://$SERVICE_IP/"
```

```
echo "WordPress Admin URL: http://$SERVICE_IP/admin"
```

2. Open a browser and access WordPress using the obtained URL.

3. Login with the following credentials below to see your blog:

```
echo Username: user
```

```
echo Password: $(kubectl get secret --namespace default mywordpress -o jsonpath=".data.wordpress-password" | base64 --decode)
```

Desinstalación:

```
raulunzue@KMASTER:~$ helm delete mywordpress
```

```
release "mywordpress" uninstalled
```

Ejemplo de instalación OpenVPN con Helm:

<https://www.máquinasvirtuales.eu/instalar-openvpn-server-en-kubernetes/>

COMANDOS BÁSICOS EN KUBERNETES

En este apartado, vamos a listar unos cuantos comandos que os serán útiles para administrar vuestra infraestructura Kubernetes:

<https://kubernetes.io/docs/reference/generated/kubectl/kubectl-commands>

Si queremos extraer la información de nuestro clúster

```
kubectl cluster-info
```

```
raulunzue@KBMMASTER:~$ kubectl cluster-info
Kubernetes master is running at https://192.168.2.193:6443
KubeDNS is running at https://192.168.2.193:6443/api/v1/namespaces/kube-system/services/kube-dns:dns/
proxy

To further debug and diagnose cluster problems, use 'kubectl cluster-info dump'.
```

COMANDO CON GET

```
$ kubectl get [(-o|--output=)json|yaml|wide|custom-columns=...|custom-columns-file=...|go-template=...|go-template-
file=...|jsonpath=...|jsonpath-file=...] (TYPE[.VERSION][.GROUP] [NAME | -l label] | TYPE[.VERSION][.GROUP]/NAME ...)
[flags]
```

COMANDO CON RUN

```
$ kubectl run NAME --image=image [--env="key=value"] [--port=port] [--dry-run=server|client] [--overrides=inline-json]
[--command] -- [COMMAND] [args...]
```

Listar de los nodos del clúster

```
kubectl get nodes
```

```
raulunzue@KBMMASTER:~$ kubectl get nodes
NAME        STATUS   ROLES      AGE    VERSION
kbmaster    Ready    master    88d    v1.17.4
kubernetes01  Ready    <none>    16d    v1.17.4
kubernetes02  Ready    <none>    88d    v1.17.4
```

Listado de Servicios

```
kubectl get service -A
```

NAMESPACE	NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
default	kubernetes	ClusterIP	10.96.0.1	<none>	443/TCP	88d
haproxy-controller	haproxy-ingress	NodePort	10.96.189.146	<none>	80:30086/TCP,443:31806/TCP,1024:32668/TCP	16d
haproxy-controller	ingress-default-backend	ClusterIP	10.96.32.39	<none>	8080/TCP	16d
kube-system	kube-dns	ClusterIP	10.96.0.10	<none>	53/UDP,53/TCP,9153/TCP	88d
kube-system	kubernetes-dashboard	ClusterIP	10.96.218.232	<none>	80/TCP	27m
kubernetes-dashboard	dashboard-metrics-scraper	ClusterIP	10.96.200.140	<none>	8000/TCP	15m
kubernetes-dashboard	kubernetes-dashboard	ClusterIP	10.96.204.237	<none>	443/TCP	15m

Listar Pods

```
kubectl get pods -A
```

NAMESPACE	NAME	READY	STATUS	RESTARTS	AGE
default	nginx-deployment-elblogdenegu-7dd686b6bc-b5k82	1/1	Running	0	6d14h
default	nginx-deployment-elblogdenegu-7dd686b6bc-c6jms	1/1	Running	0	6d14h
default	nginx-deployment-elblogdenegu-7dd686b6bc-lmslj	1/1	Running	0	6d14h
haproxy-controller	haproxy-ingress-596fb4b4f4-kn8qh	0/1	CrashLoopBackOff	4650	16d
haproxy-controller	ingress-default-backend-558fbcb9b46-j5wfp	1/1	Running	1	16d
kube-system	coredns-6955765f44-gwgk5	0/1	Running	5	20d
kube-system	coredns-6955765f44-kddw7	1/1	Running	6	88d
kube-system	etcd-kbmaster	1/1	Running	6	88d
kube-system	kube-apiserver-kbmaster	1/1	Running	7	88d
kube-system	kube-controller-manager-kbmaster	1/1	Running	8	88d
kube-system	kube-flannel-ds-amd64-ckhtc	1/1	Running	7	88d
kube-system	kube-flannel-ds-amd64-mwtm4	1/1	Running	1	16d
kube-system	kube-flannel-ds-amd64-xcppv	1/1	Running	6	88d
kube-system	kube-proxy-f4kl5	1/1	Running	6	88d
kube-system	kube-proxy-r6nv9	1/1	Running	7	88d
kube-system	kube-proxy-w48xm	1/1	Running	1	16d
kube-system	kube-scheduler-kbmaster	1/1	Running	8	88d
kube-system	kubernetes-dashboard-6bf999dbcc-8f2c4	0/1	CrashLoopBackOff	10	29m
kubernetes-dashboard	dashboard-metrics-scraper-76585494d8-npsx7	1/1	Running	0	16m
kubernetes-dashboard	kubernetes-dashboard-5996555fd8-466vv	0/1	CrashLoopBackOff	7	16m

Listar Pods con más información, como en qué node del clúster están corriendo

```
kubectl get pods -o wide
```

NAME	READY	STATUS	RESTARTS	AGE	IP	NODE	NOMINATED NODE	READINESS GATES
nginx-deployment-elblogdenegu-7dd686b6bc-b5k82	1/1	Running	0	6d14h	10.69.2.53	kubernetes02	<none>	<none>
nginx-deployment-elblogdenegu-7dd686b6bc-c6jms	1/1	Running	0	6d14h	10.69.5.227	kubernetes01	<none>	<none>
nginx-deployment-elblogdenegu-7dd686b6bc-lmslj	1/1	Running	0	6d14h	10.69.2.52	kubernetes02	<none>	<none>

Listar Deployments

```
kubectl get deployments -A
```

NAMESPACE	NAME	READY	UP-TO-DATE	AVAILABLE	AGE
default	nginx-deployment-elblogdenegu	3/3	3	3	6d14h
haproxy-controller	haproxy-ingress	0/1	1	0	16d
haproxy-controller	ingress-default-backend	1/1	1	1	16d
kube-system	coredns	1/2	2	1	88d
kube-system	kubernetes-dashboard	0/1	1	0	28m
kubernetes-dashboard	dashboard-metrics-scraper	1/1	1	1	16m
kubernetes-dashboard	kubernetes-dashboard	0/1	1	0	16m

```
# Listar Namespaces
```

```
kubectl get namespaces
```

```
raulunzue@KBMMASTER:~$ kubectl get namespaces
NAME          STATUS  AGE
default       Active  88d
haproxy-controller  Active  16d
ingress-basic   Active  16d
kube-node-lease  Active  88d
kube-public     Active  88d
kube-system      Active  88d
kubernetes-dashboard  Active  88d
```

```
# Listar Pods del Namespace "default"
```

```
kubectl get pods -n default
```

```
raulunzue@KBMMASTER:~$ kubectl get pods -n default
NAME                               READY   STATUS    RESTARTS   AGE
nginx-deployment-elblogdenegu-7dd686b6bc-b5k82  1/1    Running   0          6d13h
nginx-deployment-elblogdenegu-7dd686b6bc-c6jms  1/1    Running   0          6d13h
nginx-deployment-elblogdenegu-7dd686b6bc-lmslj   1/1    Running   0          6d13h
```

```
#Listar varios componentes en un solo comando
```

```
kubectl get deployments,pods,services,namespaces -o wide
```

```
raulunzue@KBMMASTER:~$ kubectl get deployments,pods,services,namespaces -o wide
NAME                                         READY   UP-TO-DATE   AVAILABLE   AGE   CONTAINERS   IMAGES   SELECTOR
deployment.apps/nginx-deployment-elblogdenegu  3/3    3           3           6d13h  nginx        nginx:latest  app=nginx
NAME          READY   STATUS    RESTARTS   AGE   IP           NODE   NOMINATED NODE   READINESS
GATE5
pod/nginx-deployment-elblogdenegu-7dd686b6bc-b5k82  1/1    Running   0          6d13h  10.69.2.53  kubernetes02  <none>   <none>
pod/nginx-deployment-elblogdenegu-7dd686b6bc-c6jms  1/1    Running   0          6d13h  10.69.5.227  kubernetes01  <none>   <none>
pod/nginx-deployment-elblogdenegu-7dd686b6bc-lmslj   1/1    Running   0          6d13h  10.69.2.52   kubernetes02  <none>   <none>
NAME      TYPE    CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE   SELECTOR
service/kubernetes  ClusterIP  10.96.0.1   <none>     443/TCP  88d   <none>
NAME          STATUS  AGE
namespace/default  Active  88d
namespace/haproxy-controller  Active  16d
namespace/ingress-basic   Active  16d
namespace/kube-node-lease  Active  88d
namespace/kube-public     Active  88d
namespace/kube-system      Active  88d
namespace/kubernetes-dashboard  Active  88d
```

```
# Listar con abreviaturas
```

```
kubectl get deploy,pod,svc,ns
```

```

raulunzue@KMASTER:~$ kubectl get deploy,pod,svc,ns
NAME                                         READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/nginx-deployment-elblogdenegu   3/3     3           3          6d14h

NAME                                                 READY   STATUS    RESTARTS   AGE
pod/nginx-deployment-elblogdenegu-7dd686b6bc-b5k82   1/1     Running   0          6d14h
pod/nginx-deployment-elblogdenegu-7dd686b6bc-c6jms   1/1     Running   0          6d14h
pod/nginx-deployment-elblogdenegu-7dd686b6bc-lmslj   1/1     Running   0          6d14h

NAME            TYPE      CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
service/kubernetes   ClusterIP   10.96.0.1    <none>        443/TCP     88d

NAME          STATUS  AGE
namespace/default   Active  88d
namespace/haproxy-controller   Active  16d
namespace/ingress-basic   Active  16d
namespace/kube-node-lease   Active  88d
namespace/kube-public   Active  88d
namespace/kube-system   Active  88d
namespace/kubernetes-dashboard   Active  88d

```

Exponer un deployment

```
kubectl expose deployment nombre-deployment --port=80 --type=NodePort
```

Información detallada del Pod

```
kubectl describe pod nombre-servicio
```

```

raulunzue@KMASTER:~$ kubectl describe pods nginx-deployment-elblogdenegu-7dd686b6bc-b5k82
Name:           nginx-deployment-elblogdenegu-7dd686b6bc-b5k82
Namespace:      default
Priority:       0
Node:          kubernetes02/192.168.2.192
Start Time:    Sun, 29 Mar 2020 21:53:24 +0200
Labels:         app=nginx
                pod-template-hash=7dd686b6bc
Annotations:   <none>
Status:        Running
IP:            10.69.2.53
IPs:
  IP:          10.69.2.53
Controlled By: ReplicaSet/nginx-deployment-elblogdenegu-7dd686b6bc
Containers:
  nginx:
    Container ID:  docker://8492980d5d504a91b3020437ee946e31bb64e56976f4ba365d37e0d008efdaf3
    Image:          nginx:latest
    Image ID:      docker-pullable://nginx@sha256:2539d4344dd18e1df02be842ffc435f8elf699cfcc55516e2cf2cb16b7a9aea0b
    Port:          8080/TCP
    Host Port:    0/TCP
    State:        Running
      Started:   Sun, 29 Mar 2020 21:53:35 +0200
    Ready:        True
    Restart Count: 0
    Environment:  <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from default-token-zqphn (ro)
Conditions:
  Type     Status
  Initialized  True
  Ready      True
  ContainersReady  True
  PodScheduled  True
Volumes:
  default-token-zqphn:
    Type:      Secret (a volume populated by a Secret)
    SecretName: default-token-zqphn
    Optional:   false
    QoS Class:  BestEffort
    Node-Selectors: <none>
    Tolerations: node.kubernetes.io/not-ready:NoExecute for 300s
                  node.kubernetes.io/unreachable:NoExecute for 300s
Events:        <none>

```

Eliminar Servicio

```
kubectl delete service nombre-servicio
```

Podemos agregar namespace asociado también

```
kubectl delete service kubernetes-dashboard --namespace=kubernetes-
dashboard
```

```

raulunzue@KMASTER:~$ kubectl delete service kubernetes-dashboard --namespace=kubernetes-dashboard
service "kubernetes-dashboard" deleted

```

Eliminar deployment

```
kubectl delete deployment nombre-deployment
```

Eliminar deployment de un namespace

```
kubectl delete deployment kubernetes-dashboard --namespace=kubernetes-dashboard
```

```
raulunzue@KMASTER:~$ kubectl delete deployment kubernetes-dashboard --namespace=kubernetes-dashboard
deployment.apps "kubernetes-dashboard" deleted
```

Eliminar namespace

```
kubectl delete ns kubernetes-dashboard
```

```
raulunzue@KMASTER:~$ kubectl delete ns kubernetes-dashboard
namespace "kubernetes-dashboard" deleted
```

Eliminar cuenta de servicio

```
kubectl delete sa kubernetes-dashboard --namespace=kubernetes-dashboard
```

```
raulunzue@KMASTER:~$ kubectl delete sa kubernetes-dashboard --namespace=kubernetes-dashboard
serviceaccount "kubernetes-dashboard" deleted
```

Escalar a 3 replicas un deployment

```
kubectl scale --replicas=3 deployment nginx-deployment-elblogdenegu
```

o

```
kubectl scale --replicas=3 deployment kubectl scale --replicas=6 deployment
deployment nginx-deployment-elblogdenegu -n nginx-namespace
```

```
raulunzue@KMASTER:~$ kubectl scale --replicas=6 deployment nginx-deployment-elblogdenegu
deployment.apps/nginx-deployment-elblogdenegu scaled
```

Listar ReplicaSet

```
kubectl get rs
```

```
# Acceder al Pod ubuntu
```

```
kubectl --namespace=elblogdenegu exec -it ubuntu bash
```

```
# Crear un secreto
```

```
kubectl create secret generic mysql-pass --from-literal=password=mypassword
```

```
# Crear el contenido definido en el fichero deployment.yaml
```

```
kubectl create -f deployment.yaml
```

```
# Listar los tokens
```

```
kubeadm token list
```

```
# Agregar nodo al clúster
```

```
kubeadm join 192.168.2.190:6443 --token yssgk3.fuq017u179rjybht --discovery-token-ca-cert-hash sha256:a0fdd8e375b4593ca3cb7c38509c5ea75f0b132064c51f421daa67d66bcb28f7
```

```
# Consumo nodos
```

```
kubectl top  
kubectl top node  
kubectl top node NODE_NAME
```

```
# Obtener Jobs
```

```
kubectl get jobs
```

```
# Ver logs de un Job
```

```
kubectl logs NOMBRE_JOB
```

```
#Listar todos los DaemonSet
```

```
kubectl get daemonsets --all-namespaces
```

HARDENING

Hardening en informática, es el proceso de asegurar un sistema mediante la reducción de vulnerabilidades.

Esas vulnerabilidades, pueden ser usuarios, servicios, configuraciones o software innecesarios en el sistema, para el objetivo que ha sido generado, es decir, que no aportan realmente nada a la experiencia de usuario.

El objetivo principal del hardening, es el de entorpecer la labor del atacante y ganar tiempo para poder minimizar los problemas ante un incidente de seguridad.

Realmente, los que vivimos de la informática, tenemos claro que ningún sistema es invulnerable, pero trabajar el Hardening aportará capas adicionales de seguridad sobre vuestras aplicaciones e infraestructuras IT, que complementando a dispositivos como firewalls y sistemas SEM/SIEM o de monitorización de la plataforma, dificultan enormemente la labor de los atacantes.

Por suerte, también podemos trabajar en Kubernetes el Hardening para lograr una infraestructura segura.

Os vamos a dar, en los siguientes apartados, unas pocas pautas que podéis utilizar. Como anticipo, es importante saber que un clúster de Kubernetes tiene unos requerimientos a nivel de comunicaciones y deberemos abrir los siguientes puertos de comunicaciones para que trabajen entre ellos los diferentes roles:

Master node(s):

TCP	6443*	Kubernetes API Server
TCP	2379-2380	etcd server client API
TCP	10250	Kubelet API
TCP	10251	kube-scheduler
TCP	10252	kube-controller-manager
TCP	10255	Read-Only Kubelet API

Worker nodes (minions):

TCP	10250	Kubelet API
TCP	10255	Read-Only Kubelet API
TCP	30000-32767	NodePort Services

SECRETOS

Si queremos empezar a securizar nuestros proyectos de Kubernetes, podemos empezar con buenas prácticas no almacenando en claro objetos con datos sensibles, como contraseñas, llaves SSH o tokens de OAuth.

El uso de Secretos te permite controlar la manera en que se usan los datos sensibles, y reduce notablemente el riesgo de exposición de esos datos sensibles a usuarios no autorizados.

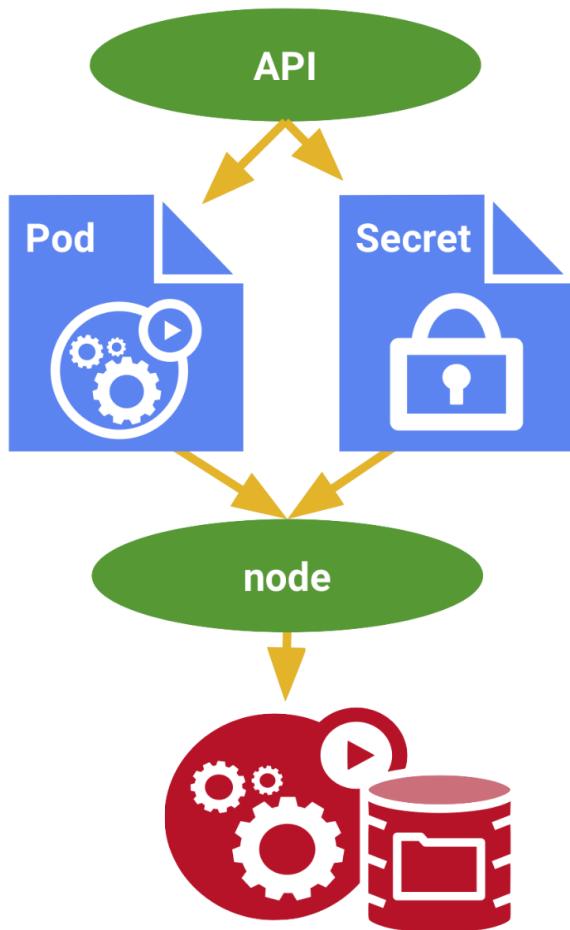
Esta información se suele colocar en las especificaciones de los Pods o en imágenes de contenedores.

Un secreto puede ser generado tanto por un usuario como por el propio sistema. Cuando lo hace el sistema, se generan automáticamente mediante cuentas de servicios con credenciales de API.

Kubernetes crea automáticamente secretos que contienen credenciales para acceder a la API y modifica automáticamente sus Pods para usar este tipo de secreto.

Otros datos interesantes sobre los secretos:

- Los secretos son objetos con espacios de nombres, es decir, existen en el contexto de un espacio de nombres
- Puede acceder a ellos a través de un volumen o una variable de entorno desde un contenedor que se ejecuta en un pod
- Los datos secretos en los nodos se almacenan en volúmenes tmpfs
- Existe un límite de tamaño por secreto de 1 MB
- El servidor API almacena secretos como texto sin formato en etcd
- La creación y el uso automático de credenciales de API se puede deshabilitar o anular si se ve necesario. Sin embargo, si todo lo que se necesita es acceder de forma segura al servidor API, este es el flujo de trabajo recomendado.



El problema de los secretos es que se cifran en base64. Con lo que tú puedes crear un secreto de la siguiente forma:

```
raulunzue@KMASTER:~$ echo -n 'elblogdenegu' | base64
ZWxibG9nZGVuZWd1
```

Por ejemplo, creamos un usuario y contraseña:

```
echo -n 'admin' | base64
YWRTaw4=
```

```
echo -n '1f2d1e2e67df' | base64
MWYyZDFlMmU2N2Rm
```

E implementarlos con un fichero YAML de la siguiente forma:

```
apiVersion: v1
kind: Secret
metadata:
  name: mysecret
type: Opaque
data:
```

```
username: YWRtaW4=
password: MWYyZDFlMmU2N2Rm
```

Pero también descifrarlos fácilmente con otro comando:

```
echo 'YWRtaW4' | base64 --decode
echo 'MWYyZDFlMmU2N2Rm' | base64 --decode
```

Y esto es un gran problema, si almacenamos todos nuestros usuarios y contraseñas, por ejemplo, en un repositorio.

Para paliar estas cosas, deberíamos ir un paso adelante y utilizar herramientas tipo KubeSealed. Que es una herramienta que te permite cifrar secretos mediante un recurso que se llama SealedSecret.

PROYECTO: [HTTPS://GITHUB.COM/BITNAMI-LABS/SEALED-SECRETS](https://github.com/bitnami-labs/sealed-secrets)

Se instala mediante el binario de KubeSealed, lo que hace es cifrar mediante una key específica para el clúster, utilizando el certificado del clúster de Kubernetes donde se aplica. Para trabajar con la herramienta deberemos llevar los secretos a JSON.

En la web del proyecto os explican su instalación con Helm:

```
raulunzue@KMASTER:/usr/local/bin$ brew install kubeseal
/home/linuxbrew/.linuxbrew/Homebrew/Library/Homebrew/brew.sh:
line 4: warning: setlocale: LC_ALL: cannot change locale (en_US.UTF-
8): No such file or directory
/bin/bash: warning: setlocale: LC_ALL: cannot change locale
(en_US.UTF-8)
/home/linuxbrew/.linuxbrew/Homebrew/Library/Homebrew/brew.sh:
line 4: warning: setlocale: LC_ALL: cannot change locale (en_US.UTF-
8): No such file or directory
/bin/bash: warning: setlocale: LC_ALL: cannot change locale
(en_US.UTF-8)
/home/linuxbrew/.linuxbrew/Homebrew/Library/Homebrew/brew.sh:
line 4: warning: setlocale: LC_ALL: cannot change locale (en_US.UTF-
8): No such file or directory
==> Downloading https://linuxbrew.bintray.com/bottles/patchelf-
0.10.x86_64_linux.bottle.1.tar.gz
==> Downloading from
https://akamai.bintray.com/7f/7f19eacef4e3d18d9c82a4f4060bd86abeb912
a4a76fa37c29f0bb104a38b2a2?gda=exp=1591045240~hmac=7815af
#####
##### 100.0%
==> Downloading https://linuxbrew.bintray.com/bottles/kubeseal-
0.12.4.x86_64_linux.bottle.tar.gz
==> Downloading from
https://akamai.bintray.com/83/83a5abe17baa28f13d5ad6241c161a97243b0c
cdb0dcfa58172a1ce63837ca34?gda=exp=1591045240~hmac=05bb19
```

```

#####
##### 100.0%
==> Installing dependencies for kubeseal: patchelf
==> Installing kubeseal dependency: patchelf
/bin/bash: warning: setlocale: LC_ALL: cannot change locale
(en_US.UTF-8)
==> Pouring patchelf-0.10.x86_64_linux.bottle.1.tar.gz
  ↳ /home/linuxbrew/.linuxbrew/Cellar/patchelf/0.10: 8 files,
921.5KB
==> Installing kubeseal
==> Pouring kubeseal-0.12.4.x86_64_linux.bottle.tar.gz
  ↳ /home/linuxbrew/.linuxbrew/Cellar/kubeseal/0.12.4: 5 files,
33.6MB

```

También se puede instalar con Helm:

VERSION	DESCRIPTION	URL	CHART VERSION	APP
1.10.1	A Helm chart for Sealed Secrets	https://hub.helm.sh/charts/stable/sealed-secrets	0.12.1	

```

raulunzue@KMASTER:~$ helm search hub sealed
          URL
          CHART VERSION APP

```

VERSION	DESCRIPTION	URL	CHART VERSION	APP
1.10.1	A Helm chart for Sealed Secrets	https://hub.helm.sh/charts/stable/sealed-secrets	0.12.1	

```

raulunzue@KMASTER:~$ helm install stable/sealed-secrets --
generate-name
NAME: sealed-secrets-1591042254
LAST DEPLOYED: Mon Jun 1 22:11:01 2020
NAMESPACE: default
STATUS: deployed
REVISION: 1
TEST SUITE: None
NOTES:
You should now be able to create sealed secrets.

```

1. Install client-side tool into /usr/local/bin/

```

GOOS=$(go env GOOS)
GOARCH=$(go env GOARCH)
wget https://github.com/bitnami-labs/sealed-secrets/releases/download/v0.12.1/kubeseal-\$GOOS-\$GOARCH
sudo install -m 755 kubeseal-$GOOS-$GOARCH
/usr/local/bin/kubeseal
---
```

Generaríamos un secreto:

```

raulunzue@KMASTER:~$ echo -n elblogdenegu | kubectl create
secret generic mysecret --dry-run --from-file=foo=/dev/stdin -o
json >ebdnsecret.json

```

```
W0601 22:16:05.684421      7753 helpers.go:535] --dry-run is  
deprecated and can be replaced with --dry-run=client.
```

Lo revisamos con cat ebdnsecret.json y vemos como está cifrado:

```
raulunzue@KBMMASTER:~$ cat ebdnsecret.json  
{  
    "kind": "Secret",  
    "apiVersion": "v1",  
    "metadata": {  
        "name": "mysecret",  
        "creationTimestamp": null  
    },  
    "data": {  
        "foo": "ZWxibG9nZGVuZWd1"  
    }  
}
```

Probamos a descifrarlo como hemos dicho para secretos en base64:

```
raulunzue@KBMMASTER:~$ echo 'ZWxibG9nZGVuZWd1' | base64 -d  
elblogdenegubase64: entrada inválida
```

Instalamos kubeseal controller:

```
raulunzue@KBMMASTER:~$ kubectl apply -f https://github.com/bitnami-labs/sealed-secrets/releases/download/v0.7.0/controller.yaml
```

```
raulunzue@KBMMASTER:~$ kubectl apply -f https://github.com/bitnami-labs/sealed-secrets/releases/download/v0.7.0/sealedsecret-crd.yaml
```

Para cifrarlo con Kubeseal lo hacemos con el siguiente comando:

```
raulunzue@KBMMASTER:~$ kubeseal < ebdnsecret.json >  
ebdnsecretsealed.json
```

Lo volvemos a comprobar con cat y verificamos el encriptado es totalmente más seguro, y es un recurso "SealedSecret".

```
raulunzue@KBMMASTER:~$ cat ebdnsecretsealed.json
```

Y aplicamos el nuevo secreto:

```
raulunzue@KMASTER:~$ kubectl apply -f ebdnsecretsealed.json
```

Y verificamos que disponemos de él:

```
raulunzue@KMASTER:~$ kubectl get sealedsecrets
```

Y revisamos los secretos:

```
raulunzue@KMASTER:~$ kubectl get secrets
```

Podemos ver nuestro secreto:

```
raulunzue@KMASTER:~$ kubectl get secrets ebdnsecret -o yaml
```

USUARIOS Y AUTENTICACIÓN

Cuando empiezas a trabajar con Kubernetes, puedes tener la tendencia a querer usar usuarios como con otro tipo de infraestructuras.

Kubernetes tiene dos tipos de categorías de usuarios:

- Usuarios "normales"
- Cuentas de servicio que son administrados por Kubernetes directamente

En otro tipo de plataformas los usuarios "normales" se agregarían a través de llamadas a la API del clúster. Pero ese tipo de objeto no existe en Kubernetes.

Estos usuarios son gestionados normalmente por servicios externos, ya sea por una plataforma cloud (Amazon, Google...) o por un administrador que distribuye claves privadas, por ejemplo.

A partir de esto, deberemos intentar crear una estrategia de autenticación lo más completa posible:

- Tokens para las cuentas de servicio
- Y al menos otro método para la autenticación de usuarios (LDAP, SAML, Kerberos...) mediante un proxy o webhook de autenticación

PERMISOS (RBAC)

Como ya hemos comentado la gestión de identidades y usuarios no está integrada en la plataforma y debe ser administrada por plataformas externas (Directorio Activo, Keycloak...). Sin embargo, lo que sí hace Kubernetes es manejar la autenticación y la autorización.

Cuando hablamos de dar permisos en un clúster de Kubernetes, tendremos que hablar del control de acceso basado en Roles. O lo que se denomina Role Based Access Control o RBAC, el cual maneja políticas de seguridad para usuarios, grupos o Pods, y que está implementado de una forma estable desde la versión 1.8 de Kubernetes.

Existen otras formas de autorizar usuarios como ABAC (Attribute-based access control), mediante Webhook o Node Authorization, pero explicaremos RBAC que es la más popular.

Ya hemos hablado de los usuarios en el punto anterior. En el caso de los Pods, para que un Pod pueda acceder a la API de Kubernetes necesitamos darle permisos específicos para ello. De hecho, los Roles se definen a nivel del espacio de nombres o Namespace.

Roles, Cluster Roles, Role/Cluster Bindings y Service Accounts

Vamos a definir brevemente en qué consisten los tipos de permisos que somos capaces de dar mediante RBAC.

- **Roles:** sirven para declarar servicios que afectan a Namespaces. Es decir, si necesitas darle permisos a un usuario, para acceder a un Namespace necesitas generar un role.
 - Recursos: son Pods, NameSpace, Servicios, Deployments...
 - Verbos: acciones que puedo lanzar a esos recursos (Get, List...)
- **Cluster Role:** si lo que buscamos es dar permisos a todo el Cluster y todos los Namespaces generaremos un Cluster Role.
- **Role/Cluster Binding:** define qué usuarios tienen qué roles. Sirve para asignar los usuarios a un Role o Cluster Role y poder asignar los permisos.
 - Role: definición de los permisos para cada tipo de recurso de Kubernetes
 - Subject: son los propios usuarios o grupos de usuarios
 - Service Account: usuario que se crea para asignar los permisos
- **Service Accounts:** para dar permisos a Pods, necesitamos generar Service Accounts o cuentas de servicio

Como apunte, los permisos sólo permiten acceso a recursos, porque "por defecto se deniega todo" y es posible asignar varios roles al mismo usuario

El único pre-requisito para usar RBAC que esté habilitado en nuestro clúster mediante la opción "--authorization-mode=RBAC". Eso lo podemos comprobar mediante el comando:

```
kubectl api-versions
```

Si está habilitado tendremos un valor tipo:

```
.rbac.authorization.k8s.io/v1
```

Para implementar un Role en un Namespace concreto lo definiríamos:

```
kind: Role
apiVersion: rbac.authorization.k8s.io/v1
metadata:
  namespace: ebdn-namespace
  name: example-role
rules:
- apiGroups: []
  resources: ["pods"]
  verbs: ["get", "watch", "list"]
```

Y luego generaríamos el RoleBinding a partir de él en el mismo Namespace. Le asignamos al usuario "elblogdenegu" el role de ejemplo "example-role", generando un Role binding "example-rolebinding":

```
kind: RoleBinding
apiVersion: rbac.authorization.k8s.io/v1
metadata:
  name: example-rolebinding
  namespace: ebdn-namespace
subjects:
- kind: User
  name: elblogdenegu
  apiGroup: rbac.authorization.k8s.io
roleRef:
  kind: Role
  name: example-role
  apiGroup: rbac.authorization.k8s.io
```

Si lo que queremos es generar permisos sin depender de Namespace (para dárselos, por ejemplo, a un nodo) definiríamos ClusterRole:

```
kind: ClusterRole
apiVersion: rbac.authorization.k8s.io/v1
metadata:
  name: example-clusterrole
rules:
- apiGroups: []
  resources: ["pods"]
  verbs: ["get", "watch", "list"]
```

Y definiríamos un ClusterRoleBinding de la siguiente forma:

```
kind: ClusterRoleBinding
apiVersion: rbac.authorization.k8s.io/v1
metadata:
  name: example-clusterrolebinding
subjects:
- kind: User
  name: elblogdenegu
  apiGroup: rbac.authorization.k8s.io
roleRef:
  kind: ClusterRole
  name: example-clusterrole
  apiGroup: rbac.authorization.k8s.io
```

Por último, explicamos como dar todos los permisos a un grupo de usuarios o a cuentas de servicio. Al definir el RoleBinding en el apartado subjects le definiríamos el grupo de la siguiente forma:

```
subjects:
- kind: Group (cambiaríamos esto por ServiceAccount si queremos
hacerlo con una cuenta de servicio)
  name: "administradores"
  apiGroup: rbac.authorization.k8s.io
```

Si le añadimos el Namespace, sólo le daremos totales sobre el Namespace, lo hago con una cuenta de servicio:

```
subjects:
- kind: Group
  name: system:serviceaccounts:ebdn-namespace
  apiGroup: rbac.authorization.k8s.io
```

Si os preguntáis como saber qué permisos existen, los podéis revisar en la API REFERENCE:

<https://kubernetes.io/docs/reference/generated/kubernetes-api/v1.18/>

GESTIÓN DE RECURSOS Y LÍMITES

Cuando creas contenedor en una infraestructura Kubernetes, sobre todo en Producción, es importante gestionar los recursos y límites que vamos a asignar a nuestras aplicaciones. A nivel de seguridad es importante, porque un solo contenedor, al compartir un host con otros contenedores, podría generar una denegación de servicio.

En la generación del Pod lo podemos controlar fácilmente mediante las secciones Requests y Limits en el fichero de ejecución YML o YAML. A continuación, vamos a explicar cada sección por separado:

Sección Requests en Kubernetes

La mejor forma de explicar algo es con un ejemplo:

```
apiVersion: v1
kind: Pod
metadata:
  name: pod-requests
spec:
  containers:
    - name: elblogdenegu-nginx
      image: nginx
      resources:
        requests:
          cpu: "200m"
          memory: "20Mi"
```

Lo que estamos haciendo es definir por cada contenedor, la CPU y Memoria RAM que van a necesitar como mínimo para poder ejecutarse.

- CPU: el valor XXXm define en unidades de CPU. En el ejemplo, le estamos dando una quinta parte de una CPU
- MEMORIA: el valor XXMi define la memoria en MegaBytes o MB.

Como podéis imaginar aquí sólo damos el mínimo necesario, pero para limitar necesitaremos usar Limits.

Sección Limits en Kubernetes

Volvemos a un ejemplo:

```
apiVersion: v1
kind: Pod
metadata:
  name: pod-requests
spec:
  containers:
    - name: elblogdenegu-nginx
      image: nginx
      resources:
        limits:
          cpu: "1"
          memory: "40Mi"
```

En el caso de Limits lo que estamos dando son *los máximos de uso* de cada contenedor dentro del host.

Como podéis presuponer, los valores de Limits y Requests se pueden y deben usar juntos. Si no ponéis valores mínimos y sólo colocáis límites a vuestros contenedores, el valor mínimo (Requests) se tomará el de los especificados en Limits.

¿Qué pasa si un contenedor llega al límite de los recursos?

Cuando un contenedor intenta usar una cantidad mayor de RAM o CPU de los límites marcados, pasa del estado Running a Terminated. Y se reiniciará para restablecerse.

Eso pasa gracias a OOMKilled (Out Of Memory Killed), que está implementado en el kernel del sistema y genera un error de memoria.



¿Qué pasa si un Host llega al límite de los recursos?

Si la suma de los límites de los contenedores sobrepasa la capacidad del nodo, lo que pasará es que, en algún momento, ciertos contenedores podrían ser reiniciados e incluso matados para liberar recursos en el sistema.

OTRAS PAUTAS BÁSICAS

En este apartado, os voy a dar pautas que podéis seguir de una forma más genérica, con las que hemos desarrollado hasta ahora:

- **Puertos privilegiados:** Hay que evitar asignar puertos privilegiados en Pods en todo lo posible. Los puertos TCP/IP privilegiados son los menores a 1024 y son especiales en cualquier sistema operativo. Una de sus características, es que normalmente un usuario básico, no tiene privilegios para correr servicios que escuchen en ellos. Esto cambia un poco cuando trabajas con contenedores, ya que sí somos capaces de publicarlos. Por lo que sería conveniente dejar estos puertos para excepciones, como balanceadores de carga, que gestionen las conexiones.
- **SSH en contenedores:** Podemos tener la costumbre de querer implementar un server SSH en contenedores Linux. Esto no debería realizarse, ya que todas las conexiones a la infraestructura deberían realizarse a través del host. Levantar SSH en contenedores, hace más compleja la plataforma, complica la gestión de políticas de acceso y actualizaciones.
- **Chequear infraestructura periódicamente:** Existen herramientas (EJ: Kube Bench, <https://github.com/aquasecurity/kube-bench>) que nos permiten de una forma rápida chequear a nivel de seguridad nuestra infraestructura. Una buena práctica, es realizar este tipo de chequeos periódicamente.
- **Capabilities Linux:** Las capabilities son una herramienta útil para asegurar sistemas Linux. Son atributos especiales en el kernel, que otorgan privilegios específicos a procesos y ejecutables binarios. Lo que nos permite dar a un proceso algunos privilegios, pero no todos los privilegios, por ejemplo, del usuario root.
- **Evitar contenedores privilegiados:** si ejecutamos contenedores con privilegios, permitiría al contenedor hacer casi las mismas cosas que el host puede hacer.
- **Uso de Selinux o AppArmor:** Estas dos herramientas permiten aislar aplicaciones dentro de otras dentro de un sistema. Aunque son herramientas muy complejas de implementar, según la criticidad de nuestra infraestructura, deberíamos plantear su uso. Por una parte, SELinux, se basa en añadir etiquetas de seguridad a los objetos y AppArmor usa perfiles de programas para restringir de forma individual capacidades de ciertos programas. El uso de una u otra, dependerá del host y la distribución linux que se use en el host.
- **Actualizaciones:** revisaremos periódicamente bugs que se puedan corregir mediante actualizaciones tanto de nuestro Clúster como de nuestras aplicaciones publicadas.

AUTO-ESCALAMIENTO

Una de las virtudes de Kubernetes, es olvidarnos en cierta forma de su gestión (aunque esto no sea real en el día a día de un administrador), y esto es gracias a que son plataformas autoescalables. Aclaro esto...

No quiero decir que se mantenga sola, pero casi. Dentro de las múltiples herramientas que proporciona Kubernetes, disponemos de la posibilidad de hacer que nuestros Pods se autoescalen ellos solos ante cargas del sistema.

Imagináros que mantenéis una página web que ofrece un determinado producto, que, en un momento determinado, por causas externas al negocio tiene un pico alto de compras (ejemplo: vendes "epis" y surge una pandemia). Si tus Pods, que ofrecen una página web con su servidor nginx y su base de datos, son capaces de crecer de forma automática ante el aumento de la CPU de los servidores, tu negocio no se vería resentido, y podrías reaccionar con margen. Algo que en una infraestructura tradicional es más complicado, porque prácticamente dependes del factor humano casi por completo.

Para esto, existe un objeto llamado **HorizontalPodAutoscaler**, que simplemente nos permite generar reglas con respecto a una métrica que nosotros le pasamos a los ReplicaSet (de los cuales ya hemos hablado).

Ejemplo:

```
apiVersion: autoscaling/v1
kind: HorizontalPodAutoscaler
metadata:
  name: auto-scaler
spec:
  scaleTargetRef:
    kind: ReplicaSet
    name: nginx-deployment-ebdn-64fc5c855d
  minReplicas: 3
  maxReplicas: 10
  targetCPUUtilizationPercentage: 50
```

En este ejemplo, dependiendo de la CPU que utilizarán los Pods que hay actualmente corriendo, subirá o bajará el número de mínimo 3 y un máximo de 10.

También puedes hacer lo mismo con el siguiente comando:

```
kubectl autoscale rs nginx-deployment-ebdn-64fc5c855d --max=10 --cpu-percent=50
```

Los objetos HorizontalPodAutoscaler (hpa) se comprueban con el siguiente comando:

```
kubectl get hpa --watch
```

Se puede aprovisionar objetos como Pods, pero también un Clúster completo.

ANEXO I. CASO PRÁCTICO: CREAR UN CLÚSTER DE KUBERNETES

REFERENCIA: <https://www.máquinasvirtuales.eu/instalar-cluster-kubernetes-en-debian/>

En este caso práctico, os voy a enseñar cómo crear un cluster Debian con 3 nodos para poder usarlo con Kubernetes on-premise. Lo que haremos es darles a los 3 nodos un role. Uno de ellos será el master, los otros dos nodos completarán el cluster de Kubernetes.



Partimos de tener tres máquinas virtuales con Debian y actualizadas.

Lo primero es hacer que resuelvan el nombre de las máquinas entre ellas para hacer la gestión más rápida y que se entiendan entre los nodos. Cambiar master-node y slave-node por los nombres de vuestras máquinas virtuales:

```
hostnamectl set-hostname master-node
```

```
hostnamectl set-hostname slave-node
```

Así que adicionalmente, introducimos en cada máquina virtual, en su fichero hosts las ips y los nombres de las máquinas:

```
root@KBMMASTER:~# nano /etc/hosts
```

```
192.168.2.190 KBMASTER.NEGU.LOCAL KBMASTER  
192.168.2.202 KUBERNETES01.NEGU.LOCAL KUBERNETES01  
192.168.2.195 KUBERNETES02.NEGU.LOCAL KUBERNETES02
```

Instalación Kubernetes

Lo primero que haremos es instalar Docker en cada nodo:

```
apt install docker.io
```

Comprobamos la versión instalada:

```
root@KUBERNETES02:~# docker --version  
Docker version 18.09.1, build 4c52b90
```

Habilitamos el servicio para que arranque al iniciar el sistema operativo en cada nodo:

```
root@KUBERNETES01:~# systemctl enable docker  
Synchronizing state of docker.service with SysV service script with /lib/systemd/systemd-sysv-install.  
Executing: /lib/systemd/systemd-sysv-install enable docker
```

Agregamos la key de Kubernetes para todos los nodos:

```
root@KBMMASTER:~# apt-get install curl  
root@KBMMASTER:~# curl https://packages.cloud.google.com/apt/doc/apt-key.gpg | sudo apt-key add -s  
OK
```

Agregamos el repositorio:

```
root@KUBERNETES02:~# apt-add-repository "deb  
http://apt.kubernetes.io/ kubernetes-xenial main"
```

Actualizamos la caché:

```
root@KUBERNETES01:~# apt-get update  
  
Obj:1 http://security.debian.org/debian-security buster/updates  
InRelease  
  
Obj:2 http://deb.debian.org/debian buster InRelease  
  
Obj:3 http://deb.debian.org/debian buster-updates InRelease  
  
Des:4 https://packages.cloud.google.com/apt kubernetes-xenial  
InRelease [8.993 B] Des:5 https://packages.cloud.google.com/apt  
kubernetes-xenial/main amd64 Packages [32,2 kB] Descargados 41,2 kB  
en 1s (41,6 kB/s)  
  
Leyendo lista de paquetes... Hecho
```

E instalamos kubeadm en los tres nodos:

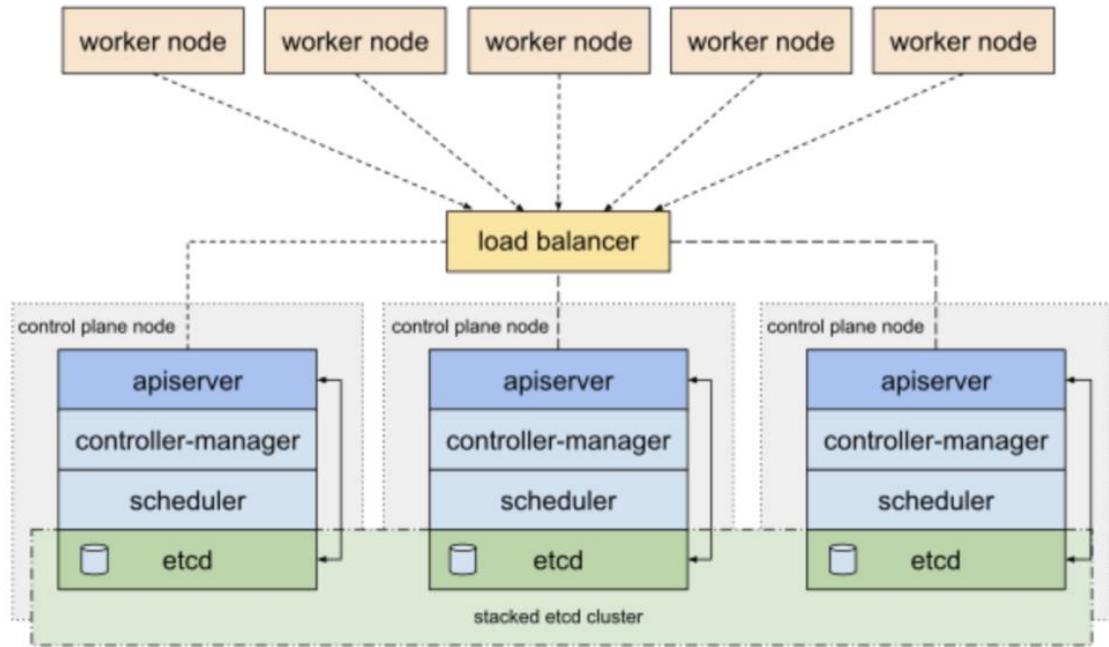
```
root@KBMMASTER:~# apt install kubeadm
```

Comprobamos que está bien instalado:

```
root@KBMMASTER:~# kubeadm version  
  
kubeadm version: &version.Info{Major:"1", Minor:"17",  
GitVersion:"v1.17.0", GitCommit:"70132b0f130acc0bed193d9ba59dd186f0e634cf",  
GitTreeState:"clean", BuildDate:"2019-12-07T21:17:50Z",  
GoVersion:"go1.13.4", Compiler:"gc", Platform:"linux/amd64"}
```

La topología que usaremos es modo Stacked. Si queréis saber más podéis revisar este enlace:

<https://kubernetes.io/docs/setup/production-environment/tools/kubeadm/ha-topology/>



CONFIGURACIÓN CLÚSTER KUBERNETES

Para implementar Kubernetes deberemos deshabilitar la memoria Swap en el master con el siguiente comando (para que sea persistente comentar la línea también de /etc/fstab):

```
root@KBMMASTER:~# swapoff -a
```

O:

```
swapoff -ased -i '/ swap / s/^/#/' /etc/fstab
```

Ahora inicializaremos el master, dando una subred para nuestra plataforma. Tardará unos minutos:

```
root@KBMMASTER:~# kubeadm init --pod-network-cidr=10.69.0.0/16
W0106 21:29:32.226994 8293 validation.go:28] Cannot validate kube-proxy config - no validator is available
W0106 21:29:32.227153 8293 validation.go:28] Cannot validate kubelet config - no validator is available
```

```
[init] Using Kubernetes version: v1.17.0

[preflight] Running pre-flight checks

[WARNING IsDockerSystemdCheck]: detected "cgroupfs" as the Docker
cgroup driver. The recommended driver is "systemd". Please follow the
guide at https://kubernetes.io/docs/setup/cri/

[preflight] Pulling images required for setting up a Kubernetes
cluster

[preflight] This might take a minute or two, depending on the speed
of your internet connection

[preflight] You can also perform this action in beforehand using
'kubeadm config images pull'

[kubelet-start] Writing kubelet environment file with flags to
file "/var/lib/kubelet/kubeadm-flags.env"

[kubelet-start] Writing kubelet configuration to file
"/var/lib/kubelet/config.yaml"

[kubelet-start] Starting the kubelet

[certs] Using certificateDir folder "/etc/kubernetes/pki"

[certs] Generating "ca" certificate and key

[certs] Generating "apiserver" certificate and key

[certs] apiserver serving cert is signed for DNS names [kbmaster
kubernetes      kubernetes.default      kubernetes.default.svc
kubernetes.default.svc.cluster.local]      and      IPs      [10.96.0.1
192.168.2.213]      [certs]      Generating      "apiserver-kubelet-client"
certificate and key

[certs] Generating "front-proxy-ca" certificate and key

[certs] Generating "front-proxy-client" certificate and key

[certs] Generating "etcd/ca" certificate and key

[certs] Generating "etcd/server" certificate and key

[certs] etcd/server serving cert is signed for DNS names [kbmaster
localhost] and IPs [192.168.2.213 127.0.0.1 ::1] [certs] Generating
"etcd/peer" certificate and key

[certs] etcd/peer serving cert is signed for DNS names [kbmaster
localhost] and IPs [192.168.2.213 127.0.0.1 ::1] [certs] Generating
"etcd/healthcheck-client" certificate and key
```

```
[certs] Generating "apiserver-etcd-client" certificate and key
[certs] Generating "sa" key and public key
[kubeconfig] Using kubeconfig folder "/etc/kubernetes"
[kubeconfig] Writing "admin.conf" kubeconfig file
[kubeconfig] Writing "kubelet.conf" kubeconfig file
[kubeconfig] Writing "controller-manager.conf" kubeconfig file
[kubeconfig] Writing "scheduler.conf" kubeconfig file
[control-plane] Using manifest folder "/etc/kubernetes/manifests"
[control-plane] Creating static Pod manifest for "kube-apiserver"
[control-plane] Creating static Pod manifest for "kube-controller-
manager"
W0106 21:30:07.388079 8293 manifests.go:214] the default kube-
apiserver authorization-mode is "Node,RBAC"; using "Node,RBAC"
[control-plane] Creating static Pod manifest for "kube-scheduler"
W0106 21:30:07.393356 8293 manifests.go:214] the default kube-
apiserver authorization-mode is "Node,RBAC"; using "Node,RBAC"
[etcd] Creating static Pod manifest for local etcd in
"/etc/kubernetes/manifests"
[wait-control-plane] Waiting for the kubelet to boot up the control
plane as static Pods from directory "/etc/kubernetes/manifests". This
can take up to 4m0s
[apiclient] All control plane components are healthy after
35.501609 seconds
[upload-config] Storing the configuration used in ConfigMap
"kubeadm-config" in the "kube-system" Namespace
[kubelet] Creating a ConfigMap "kubelet-config-1.17" in namespace
kube-system with the configuration for the kubelets in the cluster
[upload-certs] Skipping phase. Please see --upload-certs
[mark-control-plane] Marking the node kbmaster as control-plane by
adding the label "node-role.kubernetes.io/master=''"
```

```
[mark-control-plane] Marking the node kbmaster as control-plane by
adding the taints [node-role.kubernetes.io/master:NoSchedule]
[bootstrap-token] Using token: 24k2en.wbowfmuklkwlsmo
```

```
[bootstrap-token] Configuring bootstrap tokens, cluster-info
ConfigMap, RBAC Roles
```

```
[bootstrap-token] configured RBAC rules to allow Node Bootstrap
tokens to post CSRs in order for nodes to get long term certificate
credentials
```

```
[bootstrap-token] configured RBAC rules to allow the csrapprover
controller automatically approve CSRs from a Node Bootstrap Token
```

```
[bootstrap-token] configured RBAC rules to allow certificate
rotation for all node client certificates in the cluster
```

```
[bootstrap-token] Creating the "cluster-info" ConfigMap in the
" kube-public" namespace
```

```
[kubelet-finalize] Updating "/etc/kubernetes/kubelet.conf" to
point to a rotatable kubelet client certificate and key
```

```
[addons] Applied essential addon: CoreDNS
```

```
[addons] Applied essential addon: kube-proxy
```

```
Your Kubernetes control-plane has initialized successfully!
```

```
To start using your cluster, you need to run the following as a
regular user:
```

```
mkdir -p $HOME/.kube
```

```
sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
```

```
sudo chown $(id -u):$(id -g) $HOME/.kube/config
```

```
You should now deploy a pod network to the cluster.
```

```
Run "kubectl apply -f [podnetwork].yaml" with one of the options
listed at:
```

<https://kubernetes.io/docs/concepts/cluster-administration/addons/>

```
Then you can join any number of worker nodes by running the
following on each as root:
```

```
kubeadm join 192.168.2.190:6443 --token yssgk3.fuq017u179rjybht \
```

```
--discovery-token-ca-cert-hash  
sha256:a0fdd8e375b4593ca3cb7c38509c5ea75f0b132064c51f421daa67d66bcb2  
8f7
```

La última línea es importante apuntarla porque nos muestra el token y el resto de datos. Nos va a servir para unir nuevos nodos al clúster como veremos en los siguientes pasos. También nos indica que debemos hacer lo siguiente con un usuario que no sea root:

```
mkdir -p $HOME/.kube  
  
sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config  
  
sudo chown $(id -u):$(id -g) $HOME/.kube/config
```

Agregamos los otros nodos al clúster lanzando el anterior comando:

```
kubeadm join 192.168.2.190:6443 --token yssgk3.fuq017u179rjybht \  
--discovery-token-ca-cert-  
hash sha256:a0fdd8e375b4593ca3cb7c38509c5ea75f0b132064c51f421daa67d6  
6bcb28f7
```

Y ya podremos usar nuestro clúster. Si os surgen problemas podéis resetear la configuración con:

```
kubeadm reset
```

Primeros pasos clúster Kubernetes

Lo primero que haremos es comprobar que los nodos se han unido correctamente:

```
raulunzue@KMASTER:~# kubectl get nodes  
  
NAME STATUS ROLES AGE VERSION  
  
kbmaster NotReady master 13m v1.17.0  
  
kubernetes01 NotReady 99s v1.17.0  
  
kubernetes02 NotReady 71s v1.17.0
```

El estado es NotReady porque aún no se han generado las redes para los Pods:

```
raulunzue@KBMMASTER:~# kubectl apply -f
https://raw.githubusercontent.com/coreos/flannel/master/Documentation/kube-flannel.yml

podsecuritypolicy.policy/psp.flannel.unprivileged created
clusterrole.rbac.authorization.k8s.io/flannel created
clusterrolebinding.rbac.authorization.k8s.io/flannel created
serviceaccount/flannel created
configmap/kube-flannel-cfg created
daemonset.apps/kube-flannel-ds-amd64 created
daemonset.apps/kube-flannel-ds-arm64 created
daemonset.apps/kube-flannel-ds-arm created
daemonset.apps/kube-flannel-ds-ppc64le created
daemonset.apps/kube-flannel-ds-s390x created
```

Comprobamos los Pods:

```
raulunzue@KBMMASTER:~# kubectl get pods --all-namespaces
NAMESPACE NAME READY STATUS RESTARTS AGE
kube-system coredns-6955765f44-kjrnv 0/1 Running 0 18m
kube-system coredns-6955765f44-lb59t 0/1 Running 0 18m
kube-system etcd-kbmaster 1/1 Running 0 18m
kube-system kube-apiserver-kbmaster 1/1 Running 0 18m
kube-system kube-controller-manager-kbmaster 1/1 Running 0 18m
kube-system kube-flannel-ds-amd64-97pms 1/1 Running 0 53s
kube-system kube-flannel-ds-amd64-c2t7z 1/1 Running 0 53s
kube-system kube-flannel-ds-amd64-pn22m 1/1 Running 0 53s
```

```
kube-system kube-proxy-gvq7w 1/1 Running 0 6m35s  
kube-system kube-proxy-h9p6l 1/1 Running 0 18m  
kube-system kube-proxy-k57br 1/1 Running 0 7m3s  
kube-system kube-scheduler-kbmaster 1/1 Running 0 18m
```

Y volvemos a revisar que ahora están como Ready:

```
raulunzue@KBMMASTER:~# kubectl get nodes  
  
NAME STATUS ROLES AGE VERSION  
  
kbmaster Ready master 19m v1.17.0  
  
kubernetes01 Ready 7m51s v1.17.0  
  
kubernetes02 Ready 7m23s v1.17.0
```

Tenemos que agregar reglas al firewall:

Master node(s):

TCP	6443*	Kubernetes API Server
TCP	2379-2380	etcd server client API
TCP	10250	Kubelet API
TCP	10251	kube-scheduler
TCP	10252	kube-controller-manager
TCP	10255	Read-Only Kubelet API

Worker nodes (minions):

TCP	10250	Kubelet API
TCP	10255	Read-Only Kubelet API
TCP	30000-32767	NodePort Services

Así ya tenemos operativo nuestro clúster de Kubernetes.

ANEXO II. CASO PRÁCTICO: EJEMPLO DE APLICACIÓN CON CRONJOB

REFERENCIA: <https://www.maquinavirtuales.eu/kubernetes-aplicacion-con-cronjob/>

En este caso práctico, vamos a generar una aplicación web que consta de un Pod, un Service y un CronJob.

- Pod: el primer objeto es de tipo Pod, y consta de etiquetas "nginx". Se crea un contenedor llamado "front", que va a usar un volumen persistente en la carpeta "/mnt/web". Y como podéis intuir, se genera un servidor "nginx" al que se le monta un volumen persistente.
- Service: para darle visibilidad desde el interior, generaremos un servicio, en este caso LoadBalancer, y que va a abrir el puerto 80.
- CronJob: adicionalmente, vamos a generar un objeto que nos permite crear una programación. Lo lanzaremos cada minuto mediante un job, que tiene el mismo volumen que el servidor web y lanza una imagen "Ubuntu", y desde esa imagen usaremos el comando ECHO poniendo un "." en el fichero index.html de la aplicación web.

Contenido del fichero YML, hay que tener cuidado con los espacios al generararlo:

```
apiVersion: v1
kind: Pod
metadata:
  name: nginx
  labels:
    app: nginx
spec:
  volumes:
    - name: negu-volume
      hostPath:
        path: /mnt/web
        type: Directory
  containers:
```

```
- name: front

    image: nginx:latest

    volumeMounts:

        - mountPath: "/usr/share/nginx/html"

            name: negu-volume

---

apiVersion: v1

kind: Service

metadata:

    name: nginx

spec:

    type: LoadBalancer

    selector:

        app: nginx

    ports:

        - protocol: TCP

            port: 80

            targetPort: 80

---

apiVersion: batch/v1beta1

kind: CronJob

metadata:

    name: nginx

spec:

    schedule: "*/1 * * * *"
```

```

jobTemplate:
  spec:
    template:
      spec:
        volumes:
          - name: negu-volume
            hostPath:
              path: /mnt/web
              type: Directory
        containers:
          - name: nginx
            image: ubuntu:latest
            command: ["/bin/sh"]
            args: ["-c", "echo . >> /usr/share/nginx/html/index.html"]
        volumeMounts:
          - mountPath: "/usr/share/nginx/html"
            name: negu-volume
      restartPolicy: OnFailure

```

Creamos los directorios necesarios en los workers:

```

raulunzue@KUBERNETES02:~$ sudo mkdir /mnt/web
raulunzue@KUBERNETES02:~$ sudo mkdir /usr/share/nginx/html -p

```

Se ejecuta el fichero yml con el siguiente comando:

```
raulunzue@KMASTER:~$ kubectl create -f cronjob.yml
pod/nginx created
service/nginx created
cronjob.batch/hello created
```

Borrado de cronjobs, servicio y pods:

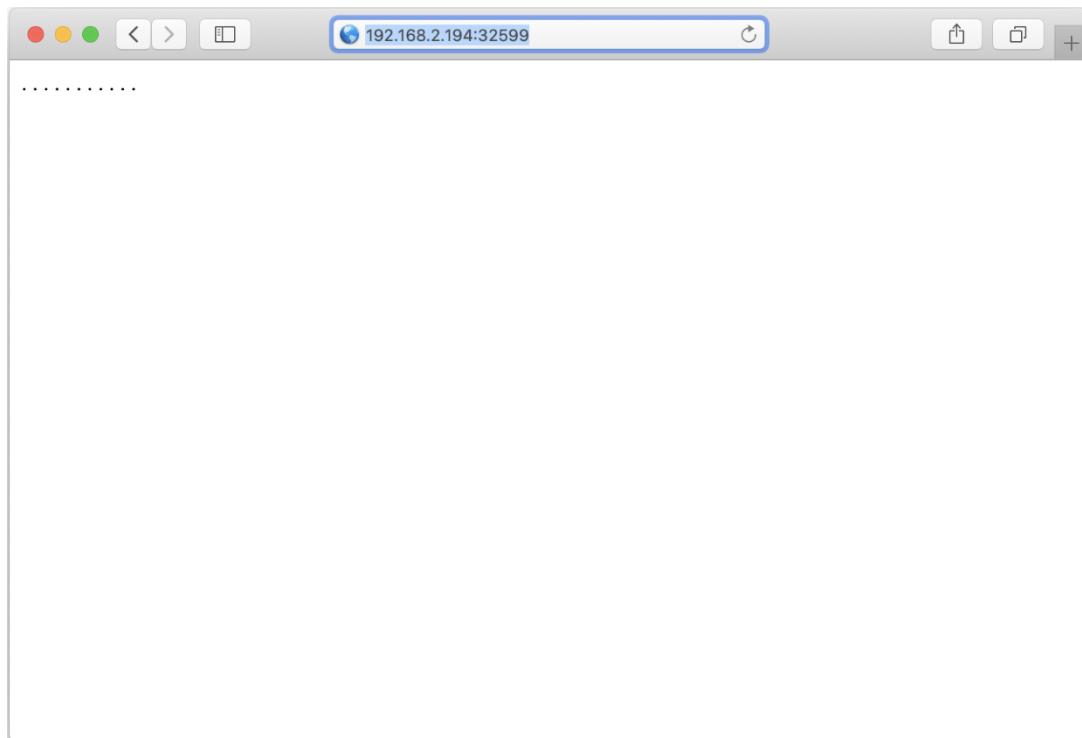
```
raulunzue@KMASTER:~$ kubectl delete cronjobs nginx
cronjob.batch "hello" deleted
raulunzue@KMASTER:~$ kubectl delete svc nginx
service "nginx" deleted
raulunzue@KMASTER:~$ kubectl delete pods nginx
pod "nginx" deleted
```

Comprobamos los pods, servicios y cronjobs generados y que no hay errores:

```
raulunzue@KMASTER:~$ kubectl get svc,cronjobs,pods
NAME          TYPE        CLUSTER-IP      EXTERNAL-
IP   PORT(S)    AGE
service/kubernetes   ClusterIP      10.96.0.1      <none>
443/TCP           73d
service/nginx       LoadBalancer   10.96.39.155   <pending>
80:32599/TCP      8m52s
NAME            SCHEDULE    SUSPEND   ACTIVE   LAST
SCHEDULE        AGE
cronjob.batch/nginx  */1 * * *
*   False         0          27s          8m52s
```

NAME	READY	STATUS	RESTARTS	AGE
pod/nginx	1/1	Running	0	8m52s
pod/nginx-1584740520-p4rc6	0/1	Completed	0	2m22s
pod/nginx-1584740580-nzvt5	0/1	Completed	0	81s
pod/nginx-1584740640-nptmd	0/1	Completed	0	21s

Si nos vamos a un navegador, hemos expuesto el puerto a través del puerto 80:32599, y podemos ver el resultado. Cada punto se añade cada minuto:



ANEXO III. CASO PRÁCTICO: INSTALACIÓN DASHBOARD

REFERENCIA: <https://www.máquinasvirtuales.eu/installacion-dashboard-en-kubernetes/>

Lo primero que haremos es revisar los nodos y su estado:

```
raulunzue@KMASTER:~$ kubectl get nodes

NAME STATUS ROLES AGE VERSION
kbmaster Ready master 66m v1.17.0
kubernetes01 Ready 60m v1.17.0
kubernetes02 Ready 55m v1.17.0
```

Nos conectamos al master vía SSH. Y haremos la instalación lanzando el siguiente comando en el master. Vamos a usar una versión estable y no la beta:

```
raulunzue@KMASTER:~$ kubectl create -f
https://raw.githubusercontent.com/kubernetes/dashboard/v1.10.1/src/deploy/alternative/kubernetes-dashboard.yaml

serviceaccount/kubernetes-dashboard created
role.rbac.authorization.k8s.io/kubernetes-dashboard-minimal created
rolebinding.rbac.authorization.k8s.io/kubernetes-dashboard-minimal created
deployment.apps/kubernetes-dashboard created
service/kubernetes-dashboard created

raulunzue@KMASTER:~$ kubectl create -f https://raw.githubusercontent.com/kubernetes/dashboard/v1.10.1/src/deploy/alternative/kubernetes-dashboard.yaml
serviceaccount/kubernetes-dashboard created
role.rbac.authorization.k8s.io/kubernetes-dashboard-minimal created
rolebinding.rbac.authorization.k8s.io/kubernetes-dashboard-minimal created
deployment.apps/kubernetes-dashboard created
service/kubernetes-dashboard created
```

Revisamos que se haya creado:

```
raulunzue@KMASTER:~$ kubectl get pods --namespace kube-system
```

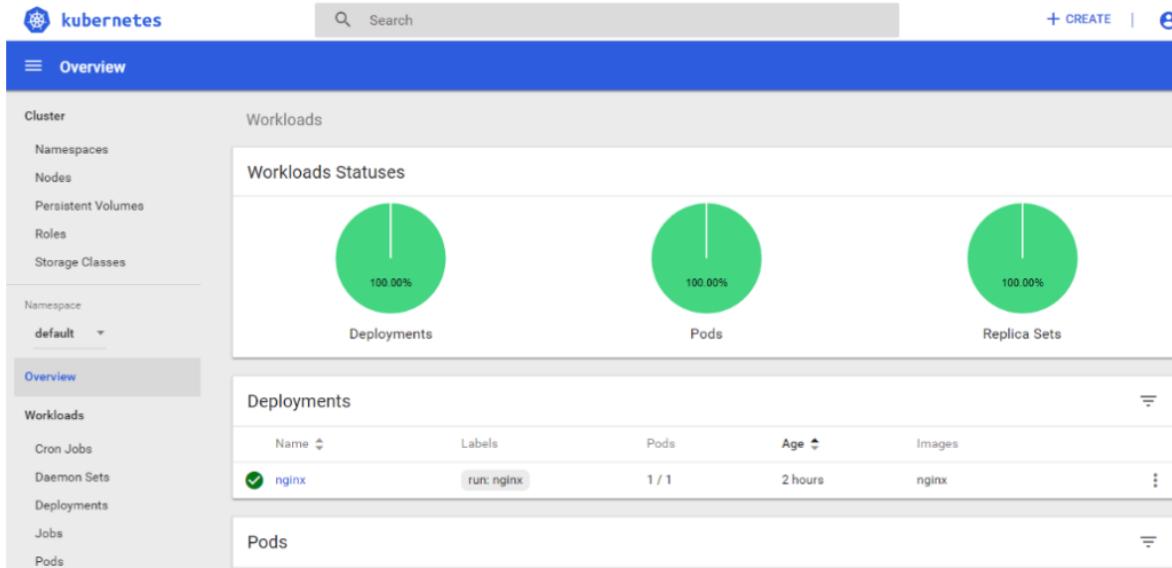
NAME	READY	STATUS	RESTARTS	AGE
coredns-6955765f44-kddw7	1/1	Running	1	16h
coredns-6955765f44-kkwrc	0/1	Running	0	16h
etcd-kbmaster	1/1	Running	1	16h
kube-apiserver-kbmaster	1/1	Running	1	16h
kube-controller-manager-kbmaster	1/1	Running	1	16h
kube-flannel-ds-amd64-ckhtc	1/1	Running	1	15h
kube-flannel-ds-amd64-xcppv	1/1	Running	0	15h
kube-flannel-ds-amd64-zjl64	1/1	Running	0	15h
kube-proxy-8ghrs	1/1	Running	0	16h
kube-proxy-f4k15	1/1	Running	1	16h
kube-proxy-r6nv9	1/1	Running	0	15h
kube-scheduler-kbmaster	1/1	Running	1	16h
kubernetes-dashboard-6bf999dbcc-hvtk2	1/1	Running	0	14s

Para mi LAB voy a dar acceso fuera de la máquina virtual, sino sólo podremos verlo sobre localhost:

```
raulunzue@KMASTER:~$ kubectl proxy --address 0.0.0.0 --accept-hosts '*'
Starting to serve on [::]:8001
```

Con esto ya podremos verlo desde nuestra red. Revisamos el estado del dashboard con el siguiente comando:

<http://192.168.2.193:8001/api/v1/namespaces/kubernetes-dashboard/services/https:kubernetes-dashboard:/proxy/>



DESINSTALAR DASHBOARD KUBERNETES

Si nos vemos en la necesidad de desinstalarlo lo podemos hacer de la siguiente manera:

```
raulunzue@KBMMASTER:~$ kubectl get service -A
NAMESPACE NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE
default kubernetes ClusterIP 10.96.0.1 443/TCP 16h
kube-system kube-dns ClusterIP 10.96.0.10 53/UDP,53/TCP,9153/TCP
16h
kubernetes-dashboard dashboard-metrics-scraper ClusterIP
10.96.125.89 8000/TCP 14h
kubernetes-dashboard kubernetes-dashboard ClusterIP 10.96.95.227
443/TCP 14h

raulunzue@KBMMASTER:~$ kubectl get deployments -A
NAMESPACE NAME READY UP-TO-DATE AVAILABLE AGE
kube-system coredns 1/2 2 1 16h
kubernetes-dashboard dashboard-metrics-scraper 1/1 1 1 14h
kubernetes-dashboard kubernetes-dashboard 0/1 1 0 14h
```

```
raulunzue@KMASTER:~$ kubectl delete deployment kubernetes-
dashboard --namespace=kubernetes-dashboard

deployment.apps "kubernetes-dashboard" deleted

raulunzue@KMASTER:~$ kubectl delete deployment dashboard-
metrics-scraper --namespace=kubernetes-dashboard

deployment.apps "dashboard-metrics-scraper" deleted

raulunzue@KMASTER:~$ kubectl get service -A

NAMESPACE NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE

default kubernetes ClusterIP 10.96.0.1 443/TCP 16h

kube-system kube-dns ClusterIP 10.96.0.10 53/UDP,53/TCP,9153/TCP
16h

kubernetes-dashboard dashboard-metrics-scraper ClusterIP
10.96.125.89 8000/TCP 14h

kubernetes-dashboard kubernetes-dashboard ClusterIP 10.96.95.227
443/TCP 14h
```

```
raulunzue@KMASTER:~$ kubectl delete service kubernetes-dashboard
--namespace=kubernetes-dashboard

service "kubernetes-dashboard" deleted

raulunzue@KMASTER:~$ kubectl delete service dashboard-metrics-
scraper --namespace=kubernetes-dashboard

service "dashboard-metrics-scraper" deleted

raulunzue@KMASTER:~$ kubectl delete sa kubernetes-dashboard --
namespace=kubernetes-dashboard

serviceaccount "kubernetes-dashboard" deleted

raulunzue@KMASTER:~$ kubectl delete secret kubernetes-dashboard-
certs --namespace=kubernetes-dashboard

secret "kubernetes-dashboard-certs" deleted

raulunzue@KMASTER:~$ kubectl delete secret kubernetes-dashboard-
key-holder --namespace=kubernetes-dashboard

secret "kubernetes-dashboard-key-holder" deleted
```

ANEXO IV. ACTUALIZAR CLÚSTER KUBERNETES

REFERENCIA: <https://www.máquinasvirtuales.eu/kubernetes-upgrade-cluster-debian/>

La idea es hacer un upgrade de un cluster Debian con Kubernetes. En el ejemplo que os voy a mostrar, existen 3 nodos:

- KBMASTER: Master
- KUBERNETES01: Worker
- KUBERNETES02: Worker

Todos en la versión 1.7.4 y vamos a pasar a la versión 1.8.2. Veréis que voy alternando entre usuario root y mi usuario de gestión del cluster. Así que vamos a ponernos manos a la obra:

Kubernetes: Upgrade Master

Nos conectamos al nodo Master y logueamos como root, para cancelar la retención de kubeadm que nos permitirá hacer un update:

```
raulunzue@KMASTER:~$ sudo su -  
root@KMASTER:~# apt-mark unhold kubeadm
```

Se ha cancelado la retención de kubeadm.

Lanzamos un escaneo:

```
root@KMASTER:~# apt-get update  
  
Des:1  http://security.debian.org/debian-security buster/updates  
InRelease [65,4 kB]  Obj:2  http://deb.debian.org/debian  buster  
InRelease  
  
Des:3  http://deb.debian.org/debian buster-updates InRelease [49,3  
kB]  Des:4  https://packages.cloud.google.com/apt  kubernetes-xenial  
InRelease [8.993 B] Des:5  http://security.debian.org/debian-security  
buster/updates/main Sources [116 kB]  Des:6  
http://security.debian.org/debian-security buster/updates/main amd64  
Packages [194 kB] Des:7  http://security.debian.org/debian-security  
buster/updates/main Translation-en [104 kB]  Des:8  
https://packages.cloud.google.com/apt  kubernetes-xenial/main amd64  
Packages [35,3 kB] Descargados 573 kB en 2s (247 kB/s)
```

Leyendo lista de paquetes... Hecho

Podemos probar a ver qué versión está disponible:

```
root@KBMMASTER:~# apt-get changelog kubeadm
E: Fallo al obtener changelog:/kubeadm.changelog No está
disponible el informe de cambios para kubeadm=1.18.2-00
```

Y forzamos el upgrade:

```
root@KBMMASTER:~# apt-get update && apt-get install -y
kubeadm=1.18.2-00 && apt-mark hold kubeadm

Obj:1 http://security.debian.org/debian-security buster/updates
InRelease

Obj:2 http://deb.debian.org/debian buster InRelease

Obj:3 http://deb.debian.org/debian buster-updates InRelease

Obj:4 https://packages.cloud.google.com/apt kubernetes-xenial
InRelease

Leyendo lista de paquetes... Hecho

Leyendo lista de paquetes... Hecho

Creando árbol de dependencias

Leyendo la información de estado... Hecho

Se actualizarán los siguientes paquetes:

kubeadm

1 actualizados, 0 nuevos se instalarán, 0 para eliminar y 24 no
actualizados.

Se necesita descargar 8.162 kB de archivos.

Se utilizarán 467 kB de espacio de disco adicional después de esta
operación.

Des:1 https://packages.cloud.google.com/apt kubernetes-
xenial/main amd64 kubeadm amd64 1.18.2-00 [8.162 kB] Descargados 8.162
kB en 1s (5.832 kB/s)
```

```
apt-listchanges: Leyendo lista de cambios...

(Leyendo la base de datos ... 135789 ficheros o directorios
instalados actualmente.)

Preparando para desempaquetar .../kubeadm_1.18.2-00_amd64.deb ...

Desempaquetando kubeadm (1.18.2-00) sobre (1.17.4-00) ...

Configurando kubeadm (1.18.2-00) ...

Scanning processes...

Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

kubeadm fijado como retenido.
```

Proseguimos con el proceso con estos comandos:

```
root@KBMMASTER:~# apt-mark unhold kubelet && apt-get update && apt-
get install -y kubelet=1.18.2-00 && apt-mark hold kubelet

kubelet ya no estaba retenido.

Obj:1 http://security.debian.org/debian-security buster/updates
InRelease

Obj:2 http://deb.debian.org/debian buster InRelease

Obj:3 http://deb.debian.org/debian buster-updates InRelease

Obj:4 https://packages.cloud.google.com/apt      kubernetes-xenial
InRelease

Leyendo lista de paquetes... Hecho

Leyendo lista de paquetes... Hecho

Creando árbol de dependencias
```

Leyendo la información de estado... Hecho

Se actualizarán los siguientes paquetes:

kubelet

1 actualizados, 0 nuevos se instalarán, 0 para eliminar y 23 no actualizados.

Se necesita descargar 19,5 MB de archivos.

Se utilizarán 1.646 kB de espacio de disco adicional después de esta operación.

```
Des:1      https://packages.cloud.google.com/apt      kubernetes-xenial/main amd64 kubelet amd64 1.18.2-00 [19,5 MB] Descargados 19,5 MB en 2s (12,0 MB/s)
```

apt-listchanges: Leyendo lista de cambios...

(Leyendo la base de datos ... 135789 ficheros o directorios instalados actualmente.)

Preparando para desempaquetar .../kubelet_1.18.2-00_amd64.deb ...

Desempaquetando kubelet (1.18.2-00) sobre (1.17.4-00) ...

Configurando kubelet (1.18.2-00) ...

Scanning processes...

Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

kubelet fijado como retenido.

Reiniciamos el servicio:

```
root@KBMMASTER:~# systemctl restart kubelet
```

Kubernetes: Upgrade el resto de nodos Master

Si tenéis más nodos master podéis usar estos comandos:

```
$ ssh raulunzue@kbmaster02  
$ kubeadm upgrade node experimental-control-plane  
  
$ ssh raulunzue@kbmaster03  
$ kubeadm upgrade node experimental-control-plane
```

KUBERNETES: UPGRADE WORKERS

Ya hemos actualizado el master, ahora nos conectamos a un nodo Worker. Os dejo el procedimiento:

```
raulunzue@KUBERNETES01:~$ sudo su -  
[sudo] password for raulunzue:  
  
root@KUBERNETES01:~# apt-mark unhold kubeadm  
kubeadm ya no estaba retenido.  
  
root@KUBERNETES01:~# apt-get update  
  
Obj:1 http://security.debian.org/debian-security buster/updates  
InRelease  
  
Obj:3 http://deb.debian.org/debian buster InRelease  
  
Obj:4 http://deb.debian.org/debian buster-updates InRelease  
  
Obj:2 https://packages.cloud.google.com/apt kubernetes-xenial  
InRelease  
  
Leyendo lista de paquetes... Hecho  
  
root@KUBERNETES01:~# apt-get install -y kubeadm=1.18.2-00 && apt-mark hold kubeadm  
  
Leyendo lista de paquetes... Hecho  
  
Creando árbol de dependencias  
  
Leyendo la información de estado... Hecho
```

Se actualizarán los siguientes paquetes:

kubeadm

1 actualizados, 0 nuevos se instalarán, 0 para eliminar y 24 no actualizados.

Se necesita descargar 8.162 kB de archivos.

Se utilizarán 467 kB de espacio de disco adicional después de esta operación.

```
Des:1      https://packages.cloud.google.com/apt      kubernetes-
xenial/main amd64 kubeadm amd64 1.18.2-00 [8.162 kB] Descargados 8.162
kB en 1s (6.110 kB/s)
```

apt-listchanges: Leyendo lista de cambios...

(Leyendo la base de datos ... 135633 ficheros o directorios instalados actualmente.)

Preparando para desempaquetar .../kubeadm_1.18.2-00_amd64.deb ...

Desempaquetando kubeadm (1.18.2-00) sobre (1.17.4-00) ...

Configurando kubeadm (1.18.2-00) ...

Scanning processes...

Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

kubeadm fijado como retenido.

```
raulunzue@KMASTER:~$ kubectl drain kubernetes01 --ignore-
daemonsets
```

node/kubernetes01 cordoned

WARNING: ignoring DaemonSet-managed Pods: kube-system/kube-
flannel-ds-amd64-mwtn4, kube-system/kube-proxy-w48xm

node/kubernetes01 drained

```
root@KUBERNETES01:~# kubeadm upgrade node
[upgrade] Reading configuration from the cluster...
[upgrade] FYI: You can look at this config file with 'kubectl -n kube-system get cm kubeadm-config -oyaml'
[upgrade] Skipping phase. Not a control plane node.
[kubelet-start] Downloading configuration for the kubelet from the "kubelet-config-1.17" ConfigMap in the kube-system namespace
[kubelet-start] Writing kubelet configuration to file "/var/lib/kubelet/config.yaml"
[upgrade] The configuration for this node was successfully updated!
[upgrade] Now you should go ahead and upgrade the kubelet package using your package manager.
```

```
root@KUBERNETES01:~# apt-mark unhold kubelet
kubelet ya no estaba retenido.
root@KUBERNETES01:~# apt-get update
Obj:1 http://security.debian.org/debian-security buster/updates
InRelease
Obj:2 http://deb.debian.org/debian buster InRelease
Obj:3 http://deb.debian.org/debian buster-updates InRelease
Obj:4 https://packages.cloud.google.com/apt kubernetes-xenial
InRelease
```

Leyendo lista de paquetes... Hecho

```
root@KUBERNETES01:~# apt-get install -y kubelet=1.18.2-00 && apt-mark hold kubelet
```

Leyendo lista de paquetes... Hecho

Creando árbol de dependencias

Leyendo la información de estado... Hecho

Se actualizarán los siguientes paquetes:

kubelet

1 actualizados, 0 nuevos se instalarán, 0 para eliminar y 23 no actualizados.

Se necesita descargar 19,5 MB de archivos.

Se utilizarán 1.646 kB de espacio de disco adicional después de esta operación.

```
Des:1      https://packages.cloud.google.com/apt      kubernetes-xenial/main amd64 kubelet amd64 1.18.2-00 [19,5 MB] Descargados 19,5 MB en 2s (12,6 MB/s)
```

apt-listchanges: Leyendo lista de cambios...

(Leyendo la base de datos ... 135633 ficheros o directorios instalados actualmente.)

Preparando para desempaquetar .../kubelet_1.18.2-00_amd64.deb ...

Desempaquetando kubelet (1.18.2-00) sobre (1.17.4-00) ...

Configurando kubelet (1.18.2-00) ...

Scanning processes...

Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

kubelet fijado como retenido.

```
raulunzue@KBMMASTER:~$ kubectl uncordon kubernetes01
```

node/kubernetes01 uncordoned

```
raulunzue@KBMMASTER:~$ kubectl get nodes
```

NAME	STATUS	ROLES	AGE	VERSION
kbmaster	Ready	master	119d	v1.18.2
kubernetes01	Ready		46d	v1.18.2
kubernetes02	Ready		119d	v1.17.4

KUBERNETES: ACTUALIZACIÓN SEGUNDO NODO(WORKER)

Ya tenemos un nodo y el master, ahora vamos a por el segundo nodo:

```
raulunzue@KMASTER:~$ ssh kubernetes02
```

```
raulunzue@kubernetes02's password:
```

```
Linux KUBERNETES02 4.19.0-6-amd64 #1 SMP Debian 4.19.67-2+deb10u2  
(2019-11-11) x86_64
```

The programs included with the Debian GNU/Linux system are free software;

the exact distribution terms for each program are described in the individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent permitted by applicable law.

```
Last login: Tue May 5 23:47:37 2020 from 192.168.2.193
```

```
raulunzue@KUBERNETES02:~$ apt-mark unhold kubeadm
```

kubeadm ya no estaba retenido.

```
raulunzue@KUBERNETES02:~$ sudo su -
```

```
[sudo] password for raulunzue:
```

```
root@KUBERNETES02:~# apt-mark unhold kubeadm
```

kubeadm ya no estaba retenido.

```
root@KUBERNETES02:~# apt-get update
```

```
Des:1 http://security.debian.org/debian-security buster/updates  
InRelease [65,4 kB] Obj:2 http://deb.debian.org/debian buster  
InRelease
```

```
Des:3 http://deb.debian.org/debian buster-updates InRelease [49,3  
kB] Des:4 https://packages.cloud.google.com/apt kubernetes-xenial  
InRelease [8.993 B] Des:5 http://security.debian.org/debian-security  
buster/updates/main Sources [116 kB] Des:6  
http://security.debian.org/debian-security buster/updates/main amd64  
Packages [194 kB] Des:7 http://security.debian.org/debian-security  
buster/updates/main Translation-en [104 kB] Des:8
```

```
https://packages.cloud.google.com/apt kubernetes-xenial/main amd64  
Packages [35,3 kB] Descargados 573 kB en 2s (252 kB/s)
```

Leyendo lista de paquetes... Hecho

```
root@KUBERNETES02:~# apt-get install -y kubeadm=1.18.2-00 && apt-mark hold kubeadm
```

Leyendo lista de paquetes... Hecho

Creando árbol de dependencias

Leyendo la información de estado... Hecho

Se actualizarán los siguientes paquetes:

kubeadm

```
1 actualizados, 0 nuevos se instalarán, 0 para eliminar y 24 no  
actualizados.
```

Se necesita descargar 8.162 kB de archivos.

```
Se utilizarán 467 kB de espacio de disco adicional después de esta  
operación.
```

```
Des:1      https://packages.cloud.google.com/apt      kubernetes-  
xenial/main amd64 kubeadm amd64 1.18.2-00 [8.162 kB] Descargados 8.162  
kB en 3s (2.446 kB/s)
```

apt-listchanges: Leyendo lista de cambios...

```
(Leyendo la base de datos ... 135633 ficheros o directorios  
instalados actualmente.)
```

Preparando para desempaquetar .../kubeadm_1.18.2-00_amd64.deb ...

Desempaquetando kubeadm (1.18.2-00) sobre (1.17.4-00) ...

Configurando kubeadm (1.18.2-00) ...

Scanning processes...

Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

```

No user sessions are running outdated binaries.

kubeadm fijado como retenido.

root@KUBERNETES02:~# exit

cerrar sesión

raulunzue@KUBERNETES02:~$ exit

cerrar sesión

Connection to kubernetes02 closed.

raulunzue@KBMMASTER:~$ kubectl drain kubernetes02 --ignore-daemonsets

node/kubernetes02 cordoned

WARNING: ignoring DaemonSet-managed Pods: kube-system/kube-flannel-ds-amd64-xcppv, kube-system/kube-proxy-r6nv9

evicting pod haproxy-controller/haproxy-ingress-596fb4b4f4-kn8qh

evicting pod haproxy-controller/ingress-default-backend-558fbe9b46-j5wfp

evicting pod kube-system/coredns-6955765f44-gwgk5

pod/haproxy-ingress-596fb4b4f4-kn8qh evicted

pod/coredns-6955765f44-gwgk5 evicted

pod/ingress-default-backend-558fbe9b46-j5wfp evicted

node/kubernetes02 evicted

raulunzue@KBMMASTER:~$ ssh kubernetes02

raulunzue@kubernetes02's password:

Linux KUBERNETES02 4.19.0-6-amd64 #1 SMP Debian 4.19.67-2+deb10u2
(2019-11-11) x86_64

The programs included with the Debian GNU/Linux system are free
software;

the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/*copyright.

```

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent permitted by applicable law.

```
Last login: Wed May  6 00:11:02 2020 from 192.168.2.193
```

```
raulunzue@KUBERNETES02:~$ sudo su -
```

```
[sudo] password for raulunzue:
```

```
root@KUBERNETES02:~# kubeadm upgrade node
```

```
[upgrade] Reading configuration from the cluster...
```

```
[upgrade] FYI: You can look at this config file with 'kubectl -n kube-system get cm kubeadm-config -oyaml'
```

```
[upgrade] Skipping phase. Not a control plane node.
```

```
[kubelet-start] Downloading configuration for the kubelet from the "kubelet-config-1.17" ConfigMap in the kube-system namespace
```

```
[kubelet-start] Writing kubelet configuration to file "/var/lib/kubelet/config.yaml"
```

```
[upgrade] The configuration for this node was successfully updated!
```

```
[upgrade] Now you should go ahead and upgrade the kubelet package using your package manager.
```

```
root@KUBERNETES02:~# apt-mark unhold kubelet
```

```
kubelet ya no estaba retenido.
```

```
root@KUBERNETES02:~# apt-get update
```

```
Obj:1 http://security.debian.org/debian-security buster/updates  
InRelease
```

```
Obj:2 http://deb.debian.org/debian buster InRelease
```

```
Obj:3 http://deb.debian.org/debian buster-updates InRelease
```

```
Obj:4 https://packages.cloud.google.com/apt kubernetes-xenial  
InRelease
```

```
Leyendo lista de paquetes... Hecho
```

```
root@KUBERNETES02:~# apt-get install -y kubelet=1.18.2-00 && apt-mark hold kubelet
```

```
Leyendo lista de paquetes... Hecho  
Creando árbol de dependencias  
Leyendo la información de estado... Hecho  
Se actualizarán los siguientes paquetes:  
  
kubelet  
  
1 actualizados, 0 nuevos se instalarán, 0 para eliminar y 23 no  
actualizados.  
  
Se necesita descargar 19,5 MB de archivos.  
  
Se utilizarán 1.646 kB de espacio de disco adicional después de  
esta operación.  
  
Des:1      https://packages.cloud.google.com/apt      kubernetes-  
xenial/main amd64 kubelet amd64 1.18.2-00 [19,5 MB] Descargados 19,5  
MB en 2s (12,3 MB/s)  
  
apt-listchanges: Leyendo lista de cambios...  
  
(Leyendo la base de datos ... 135633 ficheros o directorios  
instalados actualmente.)  
  
Preparando para desempaquetar .../kubelet_1.18.2-00_amd64.deb ...  
  
Desempaquetando kubelet (1.18.2-00) sobre (1.17.4-00) ...  
  
Configurando kubelet (1.18.2-00) ...  
  
Scanning processes...  
  
Scanning linux images...  
  
Running kernel seems to be up-to-date.  
  
No services need to be restarted.  
  
No containers need to be restarted.  
  
No user sessions are running outdated binaries.  
  
kubelet fijado como retenido.  
  
root@KUBERNETES02:~# systemctl restart kubelet  
root@KUBERNETES02:~# exit
```

cerrar sesión

```
raulunzue@KUBERNETES02:~$ exit
```

cerrar sesión

```
Connection to kubernetes02 closed.
```

```
raulunzue@KMASTER:~$ kubectl uncordon kubernetes02
```

```
node/kubernetes02 uncordoned
```

```
raulunzue@KMASTER:~$ kubectl get nodes
```

NAME	STATUS	ROLES	AGE	VERSION
kbmaster	Ready	master	119d	v1.18.2
kubernetes01	Ready		46d	v1.18.2
kubernetes02	Ready		119d	v1.18.2

Nuestro
objetivo:



Alimentar
mentes

NASCO
FEEDING
MINDS



#CAMBIA
LA
HISTORIA

La solución a
la inmigración
se encuentra
en el país de
origen



EasyVirt
MONITORIZACIÓN Y
GESTIÓN DE OPERACIONES
PARA VMWARE

Monitorización
Optimización

Planificación de recursos

Alertas

Reportes

Gestión de costos

Green IT

FREE TRIAL

www.easyvirt.com

DC SCOPE®



Capítulo 12

TANZU MISSION CONTROL



Jorge Torres

@J_Kolkes

DESCARGO DE RESPONSABILIDAD (DISCLAIMER)

Para mí es un placer hacer parte de este proyecto y su noble causa. Como miembro de la comunidad virtual de hispanohablantes entusiastas de tecnologías de la información, es un honor participar y contribuir en este libro conformado por miembros tan talentosos y admirados.

Precio dejar claro que mi participación y contribución en este material es enteramente personal y no representa en absoluto mi empleador y por lo tanto no constituye información oficial de ninguna compañía u organización. Lo escrito en este capítulo obliga ser tomado como mi opinión exclusivamente. De igual forma, advierto que no soy un experto y lo que aquí comparto es simplemente mi mejor entendimiento de los productos, tecnologías, conceptos e integraciones y posiblemente pueda estar equivocado. Por lo tanto, antes de efectuar o aplicar lo aquí leído en ambientes existentes o tomar alguna decisión al respecto, recomiendo consultar las organizaciones y representantes oficiales respectivos.

Mil gracias.

TANZU MISSION CONTROL

INTRODUCCIÓN

Hace algún tiempo que el término DevOps es común en el entorno de TI; esta definición especifica una manera de como equipos de desarrolladores y sus contrapartes en infraestructura, deben colaborar entre sí para ser más diligentes a la hora de desplegar aplicaciones y culminar proyectos de software más ágilmente, incurriendo menos costos.

Si has trabajado en TI y dependiendo tu rol, posiblemente estas o estuviste en uno de los dos lados, desarrollador(a) de aplicaciones o proveedor(a) de infraestructura, o a lo mejor tu equipo nunca tuviste nada que ver con la otra parte y el enfoque era simplemente “cumplir con lo necesario en tu campo” sin importar si afectara a otros. Lo que es cierto, es que DevOps es una metodología comprobada y cada día hay más organizaciones que adoptan dichas referencias para mejorar eficiencias y reducir costos y esto les ayuda a ser más competitivos, lo cual es una meta de cualquier negocio.

El motivo por el cual menciono DevOps iniciando este capítulo, es porque no es un secreto que los productos en el portafolio de Tanzu, están pensados y tienen el objetivo de ayudar en que esa integración y colaboración entre equipos de infraestructura y desarrolladores, sugerida en DevOps, sea más fluida y eficiente. Ese es el nuevo perfil de cliente al que VMware se quiere expandir con una bandeja de servicios que, para muchos negocios, será la combinación de herramientas perfectas, como una escalera de mano necesaria para alcanzar un nivel deseado de automatización, integración y agilidad.

Tanto VMware con Tanzu, como otras compañías compitiendo en este mismo espacio, saben que es ideal proveer una solución o combinación de productos integrados y flexibles que agilice la transformación de organizaciones y les permita desarrollar sus aplicaciones y servicios velozmente sin depender tanto y perder tiempo mientras la infraestructura es asignada, causando retrasos operativos no deseados.

Instituciones de investigación y análisis de renombre como lo es Gartner, predicen que para el 2025, la mitad de las empresas globales habrán de tener la capacidad de desplegar y mover funciones, aplicaciones y servicios en una variedad de infraestructuras (nubes), con la flexibilidad de hacerlo naturalmente, sin ataduras ni demoras; y entonces es ahí donde compañías como VMware ven un gran potencial de cubrir con sus productos, una larga lista de usos de caso y extender sus ofrecimientos a clientes más enfocados en desarrollo de software.

No es entonces sorpresa que en los últimos años adquisiciones como Pivotal, Bitnami y Heptio, hayan sido tan claves e importantes para VMware en su preparación para hoy en día presentar un portafolio tan completo como lo es Tanzu donde tienen herramientas para cubrir cada una de las áreas trascendentales para desplegar en, y administrar infraestructura sin importar en que plataforma estén desplegados las cargas de trabajo, o si son máquinas virtuales o contenedores, y al mismo tiempo brindar acceso transparente no solo a los administradores de recursos, sino también a los desarrolladores que consumen la infraestructura con más independencia.

DETALLES DE TANZU

Tanzu es una palabra de lenguaje *swahili* del este y sureste de África que tiene dos significados cuando es traducida al español: **Rama** o **Subsidiario**. Tanzu no es un producto, sino un portafolio de productos con enfoque en facilitar a sus usuarios la administración de ambientes Kubernetes, y cuya idea, y función, es simplificar como desarrollar, ejecutar y administrar aplicaciones de una manera consistente, y que puedan ser desplegadas en cualquier plataforma, o en casi todas. Tanzu es un grupo de servicios de suscripción (SaaS – Software as a Service) y no aplicaciones para instalar localmente.

Cabe mencionar también, que la vasta lista de productos en Tanzu, no son codependientes entre sí, es decir, que se puede usar un solo elemento o varios, y su integración es fluida.

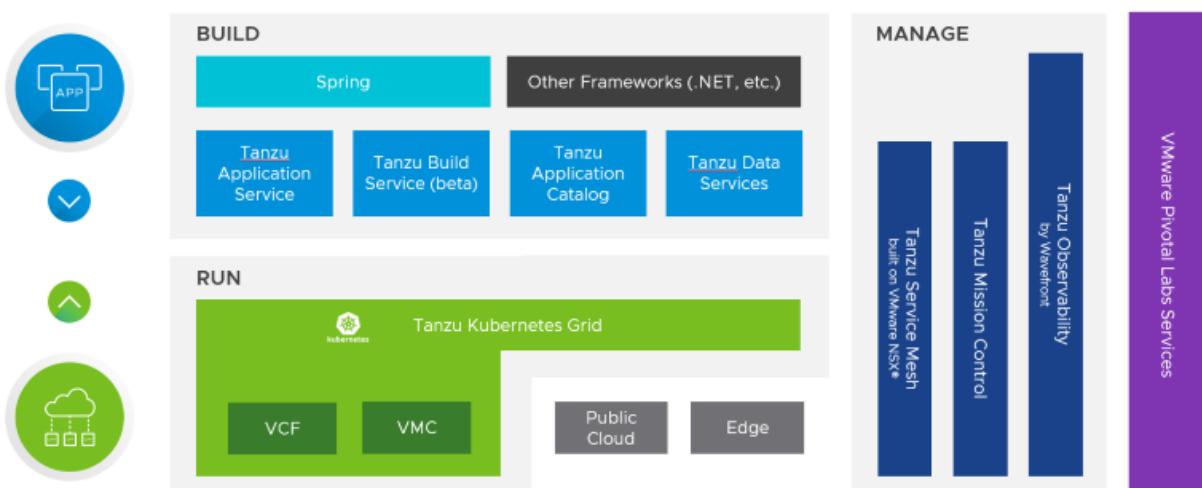
El propósito de VMware con Tanzu es poder brindar todo lo necesario para permitir a sus clientes la modernización de aplicaciones fundadas en contenedores y su correspondiente infraestructura, de la manera más efectiva y eficiente, sin necesidad de utilizar servicios o soluciones de terceros que no tengan opción de integración, o cuya integración sea complicada. Todo con enfoque en capacidad de fácil movilidad y mejores prácticas, cubriendo las áreas de construcción (**Build**), ejecución (**Run**) y, gestión y operaciones para día-2 (**Manage**).

Con la adquisición de Heptio en el 2019, VMware agrupó a expertos de dicha organización y de Pivotal, que hayan tenido experiencia en Kubernetes y manejo de aplicaciones respectivamente; VMware fue construyendo el grupo de ofrecimientos que es hoy Tanzu.

En este capítulo, mi intención es describir los conceptos desde la altura, cubrir un poco lo que son cada uno de los productos y servicios, que casos de uso tienen, y otras particularidades. Quizás no entraré en minúsculos detalles técnicos, sino una descripción de funciones y sus dependencias y/o requisitos.

Aquí un diagrama de como VMware divide los productos incluidos en su portafolio de Tanzu.

Existen 3 áreas que incluyen BUILD, RUN & MANAGE; y como puedes ver, cada uno de los productos sueltos representan uno de los espacios a servir.

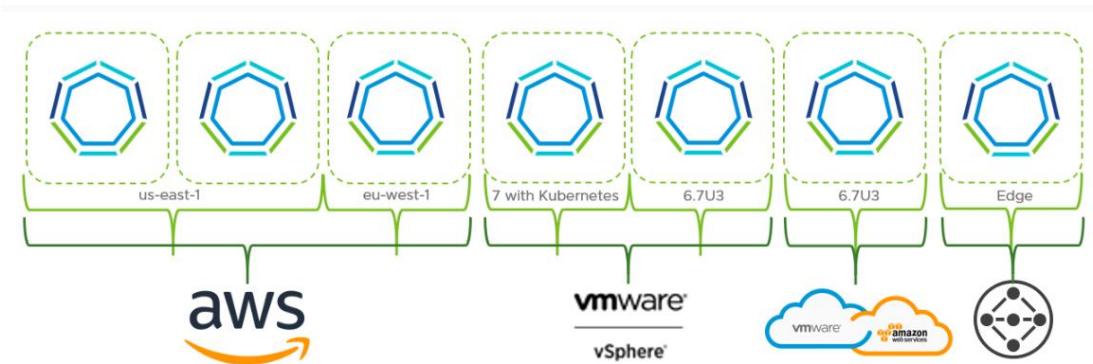


Al mes de Julio del 2020, el portafolio de Tanzu muestra las siguientes principales soluciones:

VMWARE TANZU KUBERNETES GRID (TKG)

Inicialmente llamada **Project Pacific**, es una solución que permitirá la distribución homogénea de ambientes Kubernetes en cualquier lugar donde se tenga infraestructura, bien sea en un centro de datos local o nube, instalado en vSphere, pero también incluso permitirá administrar despliegues de Kubernetes en otras plataformas, como nubes nativas (ej. AWS, Azure). Incluso desplegado en *Bare metal*.

Tanzu Kubernetes Grid habilita ejecutar Kubernetes en vSphere y permite que estos contenedores coexistan con máquinas virtuales comunes en vCenter Server de una manera eficiente y segura, y donde los administradores podrán manejar estos de una manera similar a como administran VMs comúnmente.



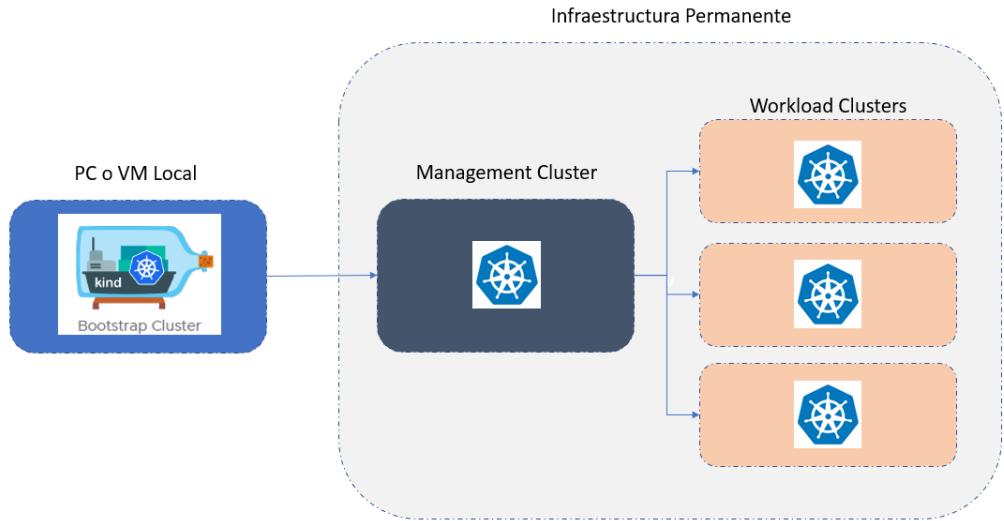
Como se detalla en la gráfica, *Tanzu Kubernetes Grid* puede existir o administrar despliegues en diversas plataformas. En este ejemplo se muestra AWS nativo, vSphere (tanto en vSphere 7 y 6.7U3), en VMware Cloud on AWS y Edge.

Un despliegue de TKG se construye de diferentes componentes; el principal es el *Management Cluster* que es el primero de los elementos a desplegar y pieza principal, ya que es aquí desde donde se ejecuta la creación de clústeres de Kubernetes posteriormente.

Otro de los elementos a incluir son los propios clústeres de Kubernetes, desplegados utilizando la herramienta CLI proveída en el *Management Cluster*. Diferentes versiones de clústeres de Kubernetes pueden ser utilizadas y administradas centralmente. Agregado a estas dos partes esenciales, se suman servicios como control de ingreso, logs, autenticación y ciclo de vida de los componentes.

Para la instalación de *Tanzu Kubernetes Grid* se debe descargar una herramienta CLI, la cual se ejecuta desde una máquina local, y es con esta que se empiezan a desplegar los elementos necesarios, iniciando con el *Management Cluster*. Dependiendo de en qué plataforma se vaya a instalar TKG, los requisitos y pasos de instalación pueden variar.

Comúnmente, se dice que el despliegue de TKG es usar Kubernetes para desplegar Kubernetes, ya que en la máquina local donde se inicia dicho despliegue, la herramienta CLI creará un *management cluster temporal (bootstrap cluster)*, y desde ahí se dirigirá la instalación final – infraestructura permanente.



Una vez que tu TKG Management Cluster esté tendido, lo siguiente será desplegar clústeres de Kubernetes; y acá es importante anotar que los clústeres de Kubernetes serán instalados en la misma plataforma donde el Management Cluster está corriendo. Ahora bien, si la intención es administrar clústeres de Kubernetes en vSphere 7.0 (conocido como vSphere with Kubernetes), se deben ejecutar pasos adicionales para conectar/autenticar la herramienta CLI dentro del *Management Cluster*, con el *Kubernetes Supervisor Cluster*, que es la función dentro de vSphere 7.0

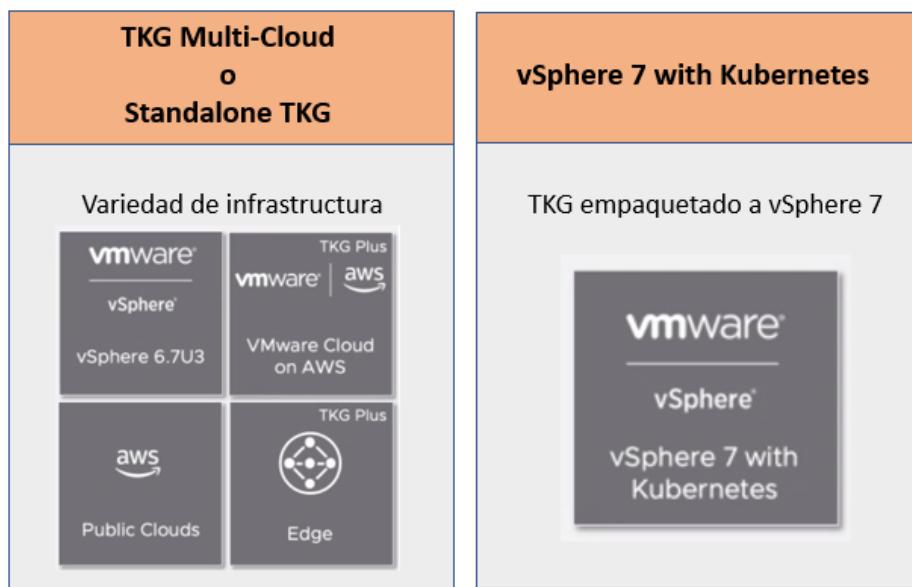
En este punto cabe hacer una aclaración... existen dos modalidades de cómo desplegar o utilizar **Tanzu Kubernetes Grid**, una es integrado a vSphere y la otra es sobre vSphere (u otra plataforma como instancias EC2 de AWS).

La primera, integrado a vSphere, traduce a correr TKG en vSphere 7.0, y en este caso, se requiere VCF (VMware Cloud Foundation versión 4.0), y su nombre oficial hoy día es **vSphere 7 with Kubernetes**; puedes pensar en esta modalidad como *embedded* o empaquetada en vSphere – Cuando se tiene vSphere 7.0 con VCF 4.0, la función de poder usar Kubernetes simplemente es habilitada, no requiere una instalación; el motor de VCF hace todo el despliegue interno necesario una vez sea seleccionado el clúster a habilitar con capacidad de correr Kubernetes.

El plano de control es creado, un agente llamado *Spherelet* es instalado en cada nodo del clúster donde se está haciendo la activación. Es ahí donde ya podrás correr contenedores nativamente en tus nodos ESXi y el clúster pasa a ser diferenciado como un **Supervisor Cluster**; de igual forma, los nodos toman el rol de *Worker Nodes*, pero sin dejar sus funciones naturales de ESXi para máquinas virtuales. Aunque no es una instalación como tal, sí hay varios parámetros y configuraciones que se deben escoger cuando se quiere activar TKG en un clúster de vSphere 7.0.

La segunda modalidad es sobre vSphere (u otra plataforma) y esta es también conocida como **TKG Multi-Cloud** o **Standalone TKG**. En esta variante, se puede instalar TKG en vSphere 6.7U3, AWS nativo, VMware Cloud on AWS o en un Edge

Cuando el Standalone TKG es desplegado (adquirido) en VMware Cloud on AWS, la versión es **TKG Plus**.



TANZU KUBERNETES GRID PLUS

TKG Plus consiste en Tanzu Kubernetes Grid previamente puntuado como base central, pero con ciertas funciones y aplicaciones adicionales. Algunas de estas cubren casos de uso como: Servidor de registro con *Harbor*, conformidad y auditoría con *Sonobuoy*, respaldo y migración con *Velero*, monitoreo con *Contour*, Autenticación con *Dex*, y visión (observability) con *Prometheus*, *Grafana*, *Alert Manager* y *Fluentbit*. Soporte para *NSX container Plug-in* (NCP) que facilita integración de NSX-T y Kubernetes, permitiendo monitoreo y manejo de segmentos, puertos y enruteadores.

TANZU KUBERNETES GRID SERVICE

Este componente es exclusivo de, y está sumamente integrado con, vSphere 7.0, y es proveído con *VMware Cloud Foundation* 4.0. Esta combinación es requerida para poder correr clústeres de Kubernetes nativamente en la misma infraestructura/nodos donde se corren VMs comunes. Igualmente, permite al administrador de vSphere usar interfaces, metodologías y formas comunes para interactuar y operar componentes de Kubernetes de manera muy similar a como se manipulan máquinas virtuales.

VMWARE TANZU MISSION CONTROL

Tanzu Mission Control es una herramienta substancial que permite la administración de los distintos despliegues de clústeres de Kubernetes desde un punto centralizado; permite gobernar la expansión, actualización y ciclo vital de los clústeres y aplicaciones ejecutándose en Kubernetes, sin importar en qué nube pública (AWS, Azure, Google Cloud), centro de datos o localidad existan. Es la consola central para estar al tanto de la salud de los despliegues y asignación de pólizas que determinarán qué, cómo y cuándo ciertos recursos deben estar disponibles, y quien puede acceder a ellos y utilizarlos; enfocado no en un solo clúster de Kubernetes, sino en todos los clústeres que se tengan bajo control con funciones de agrupación en *Workspaces* y *Namespaces*.

Sus funciones pueden ser usadas a través de API, línea de comando o interfaz gráfica. Hoy día, *Tanzu Mission Control* puede iniciar despliegues de clústeres de Kubernetes en AWS, Azure, vSphere y Google Cloud Platform, pero podría administrar clústeres en otros proveedores cuando estos son anexados después de su creación. Sin embargo, se debe anotar que actualmente, la funcionalidad de *Lifecycle* solo aplica para elementos que han sido instanciados desde TMS, no si fueron anexados. Así mismo es importante verificar que la versión a anexar sea la mínima requerida.

TMS, como es su acrónimo, brinda la facilidad de crear y organizar los distintos *namespaces* (subdivisiones lógicas de recursos en un ambiente) en grupos que ofrecen mejor manipulación de aplicaciones, recursos, usuarios y seguridad.

Como mencioné anteriormente, los productos bajo Tanzu son todos SaaS (Software as a Service) y cuando se adquiere Tanzu Mission Control, su activación es designada a una sola organización en específico bajo cloud.VMware.com. Su licenciamiento es por suscripción y su unidad de medida es por cada *core* en el ambiente.

Tener Tanzu Mission Control para administrar clústeres de Kubernetes dispersos, es poder orquestar todo bajo una sola consola con equilibrio: ciclos de vida, monitoreo y diagnósticos, conformidad de pólizas establecidas, backups, seguridad, acceso, comportamiento de redes, aumento de despliegues, etc. Y poder asignar dichos *namespaces* a desarrolladores que a su vez usarán esos recursos para instalar y desplegar sus propias aplicaciones cuando lo necesiten.

Imagina tener que administrar decenas de ambientes de Kubernetes para diferentes departamentos en una empresa, y en un despliegue común, estos estarán en una variedad de plataformas, nubes públicas, centros de datos; Tanzu Mission Control concede oportunidades para agrupar infraestructura lógicamente, por ejemplo: finanzas, marketing, recursos humanos, etc. Igualmente, facilita segregar o agrupar funciones en base a tipo de aplicación, por ejemplo: producción, preproducción, desarrollo, etc. Luego, imagina poder aplicar diferentes pólizas de algún tipo de control sobre dichos grupos previamente creados – suena bien, ¿verdad? El manejo de ambientes Kubernetes a gran escala, se hace más sencillo cuando se tiene Tanzu Mission Control y todo esto se traduce en más rapidez y consistencia.

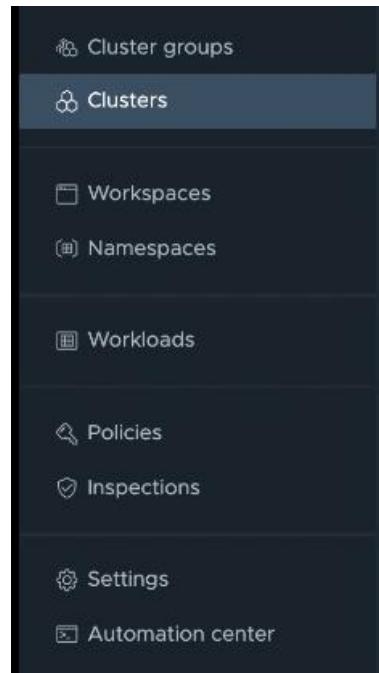
Aunque la siguiente imagen sale pequeña, quizás puedas notar los diferentes menús y opciones disponibles en la consola principal de TMS.

The screenshot shows the Tanzu Mission Control interface. On the left, a sidebar lists navigation options: Cluster groups, Clusters (selected), Workspaces, Namespaces, Workloads, Policies, Inspections, Settings, and Automation center. The main content area is titled "Overview" and displays the following information:

- Cluster group:** development
- Provider:** AWS
- Version:** 1.16.4-1-amazon2
- Region:** us-west-2
- Control plane nodes:** 1 (m5 large)
- Worker nodes:** 3
- Namespaces:** 6
- Pods:** 27
- Total memory:** 30.32 GB
- Total cores:** 8 CPUs
- Account name:** [redacted]
- Pod CIDR:** 192.168.0.0/16
- Service CIDR:** 10.96.0.0/12
- VPC CIDR:** 10.0.0.0/16
- SSH key name:** default
- Created:** 15 days ago

Below this, there are sections for Allocated CPU (33%), Allocated memory (5%), Component health (controller-manager, etcd-0, kube-apiserver, scheduler healthy), Agent and extensions health (all green), and an Inspection section (Success, last ran 7 days ago). Buttons for "VIEW INSPECTION" and "NEW INSPECTION" are at the bottom.

El panel izquierdo en esta vista muestra Cluster groups, Clusters, Workspaces, Namespaces, Workloads, Policies, Inspections, Settings y Automation center.



En el panel del medio en la parte superior, se puede ver las diferentes secciones con información muy relevante

Overview	Nodes	Node pools	Namespaces	Workloads	Inspections
Cluster group	development	Control plane nodes	1 (m5.large)	Total memory	30.32 GB
Provider	AWS	Worker nodes	3	Total cores	8 CPUs
Version	1.16.4-1-amazon2 ⓘ	Namespaces	6	Account name	[REDACTED]
Region	us-west-2	Pods	27	Pod CIDR	192.168.0.0/16
Labels				Service CIDR	10.96.0.0/12
				VPC CIDR	10.0.0.0/16
				SSH key name	default
				Created	15 days ago

Bajo la vista de *Clusters*, nótese como en este ambiente existen una variedad de despliegues de Kubernetes bajo administración, tres de ellos aprovisionados desde TMS (*provisioned*) en AWS, como también seis anexados (*attached*) en Azure, AWS vSphere y GKE.

The screenshot shows the Tanzu Mission Control interface for managing clusters. On the left, there's a sidebar with navigation links: Cluster groups, Clusters (which is selected and highlighted in blue), Workspaces, Namespaces, Workloads, Policies, Inspections, Settings, and Automation center. The main area is titled 'Clusters' and shows a table of all clusters. The table has columns for Cluster name, Provider, Type, Status, Health, Version, Allocated memory, Allocated CPU, and Nodes. Two specific rows are circled in red: one for an AWS cluster labeled 'aws-' with the status 'Provisioned', and another for an Azure cluster labeled 'azure-' with the status 'Attached'. Other clusters listed include AWS EKS, Google Cloud GKE, and various enterprise PKS clusters.

Cluster	Provider	Type	Status	Health	Version	Allocated memory	Allocated CPU	Nodes
aws-[REDACTED]	aws	Provisioned	Ready	🟢	v1.16.4-1-amazon2 ⓘ	5% 161 GB / 30.32 GB	33% 2.67 CPUs / 8 CPUs	4
aws-[REDACTED]	aws	Provisioned	Ready	🟢	v1.17.2-1-amazon2	5% 161 GB / 30.32 GB	33% 2.67 CPUs / 8 CPUs	4
aws-[REDACTED]	aws	Provisioned	Ready	🟢	v1.17.2-1-amazon2	4% 161 GB / 45.57 GB	36% 4.27 CPUs / 12 CPUs	6
azure-[REDACTED]	az	Attached	Ready	🟢	v1.15.3	22% 1.76 GB / 9.95 GB	94% 1.87 CPUs / 2 CPUs	1
eks-[REDACTED]	aws	Attached	Ready	🟢	v1.14.9-eks-c0ecc	14% 1.61 GB / 11.36 GB	24% 1.45 CPUs / 6 CPUs	3
enterprise-pks-[REDACTED]	pk	Attached	Ready	🟢	v1.15.5	22% 4.20 GB / 19.23 GB	16% 1.63 CPUs / 10 CPUs	5
enterprise-pks-[REDACTED]	pk	Attached	Ready	🟢	v1.15.5	21% 4.10 GB / 19.23 GB	15% 1.53 CPUs / 10 CPUs	5
gke-prod-[REDACTED]	g	Attached	Ready	🟢	v1.13.12-gke.25	21% 2.19 GB / 10.56 GB	64% 2.42 CPUs / 3.76 CPUs	4
gke-staging-[REDACTED]	g	Attached	Ready	🟢	v1.13.12-gke.25	10% 2.17 GB / 21.11 GB	33% 2.45 CPUs / 7.52 CPUs	8

Ahora bien, ¿recuerdas que al inicio mencioné que los productos en el portafolio de Tanzu se podían usar independientemente unos de otros? Tanzu Mission Control puede ser utilizado para la administración de ambientes de Kubernetes que no fueron desplegados desde TMS; puedes anexar y administrar clústeres de Kubernetes nativos, por ejemplo. De igual manera, no por usar TKG o TKG Plus, estarás obligado a usar Mission Control.

VMWARE TANZU APPLICATION CATALOG

Esencialmente, es una tienda virtual para desarrolladores, donde podrán encontrar aplicaciones *Open-source*, al igual que otros elementos que en conjunto servirán para complementar las soluciones que dichos desarrolladores proporcionan a sus clientes en diferentes industrias.

Detrás de *Tanzu Application Catalog* está el motor de Bitnami, que VMware adquirió también en el 2019 y cuya experiencia ha sido de empaquetado de software incluyendo Kubernetes. Su licenciamiento es en base a cuántas imágenes concurrentes se usen en el ambiente.

Aparte de imágenes de aplicaciones, en esta solución también se pueden encontrar y descargar sistemas operativos predeterminados.

TANZU KUBERNETES GRID INTEGRATED EDITION

Tanzu Kubernetes Grid Integrated, anteriormente conocido como Enterprise PKS, es una variación similar para manejo a escala de contenedores y Kubernetes que incluye su propia consola de administración, configuraciones de redes avanzadas para NSX-T, un registro privado de contenedores, y manejo de ciclos de vida de sus componentes; todo empacado en un solo producto.

TANZU OBSERVABILITY

Tanzu Observability es en esencia Wavefront bajo el capó. De hecho, creo que el nombre Wavefront cambiará en el futuro. Esta herramienta es bastante potente, hecha con visión 100% para productos que corren en la nube (cualquier nube) y permite inteligentemente vincular y considerar muchos puntos métricos diferentes en un entorno para generación de reportes específicos, notificaciones de discrepancias en conformidad, utilización y estado de recursos, y en sí una visión amplia del estado de salud del ecosistema.

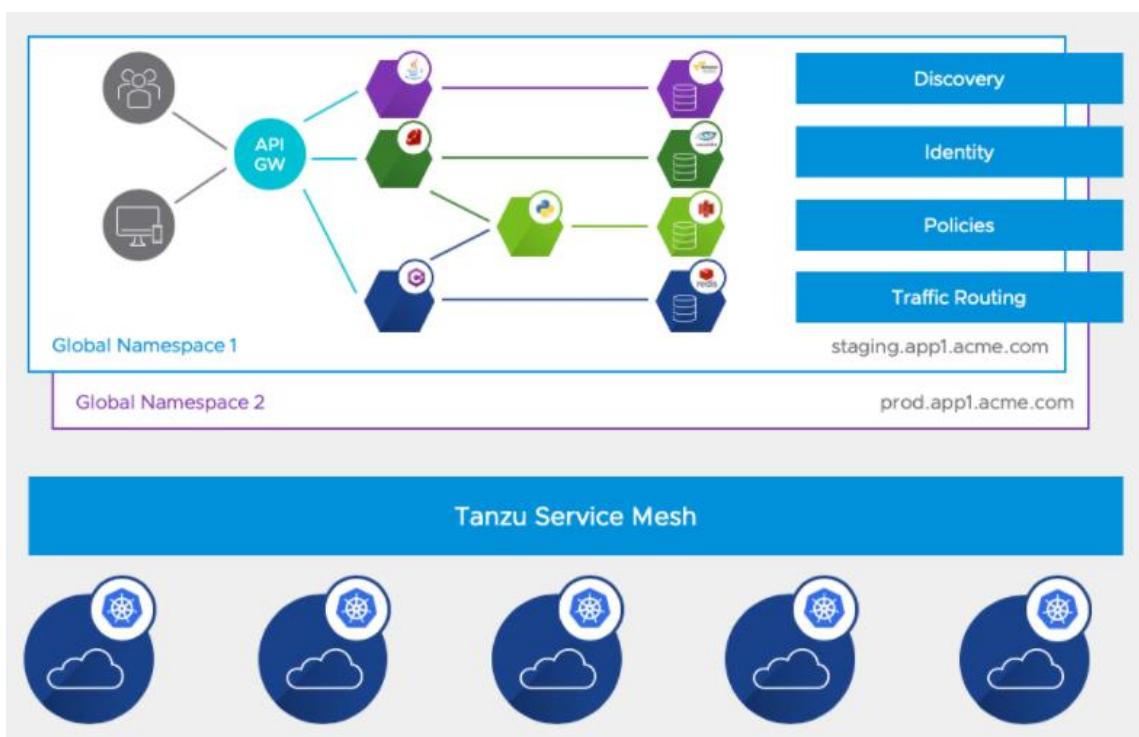
TANZU SERVICE MESH

En los últimos años, más y más cantidad de aplicaciones son distribuidas, es decir regadas por muchos clústeres, infraestructuras e incluso diferentes nubes; esto ha creado ecosistemas y redes tan dispersas, que su manejo unificado se ha complicado considerablemente; cuando a esta ecuación le agregas crecimiento con más microservicios, SaaS, serverless u otros sistemas tradicionales, poder escalar y asegurar esas plataformas y redes es muy complicado, y a menudo, esa complejidad es la causa de desplomes en los servicios. *Service Mesh* nace de la necesidad de poner control y consistencia a toda esa interconectividad y hacerla más segura, escalable y ajustada a pólizas.

¿Qué es Service Mesh? “En la arquitectura de software, Service Mesh es una capa de infraestructura dedicada para facilitar las comunicaciones de servicio a servicio entre microservicios, a menudo utilizando un ‘side proxy’.”

Fuente: https://en.wikipedia.org/wiki/Service_mesh

Hoy día el ‘service mesh’ más popular para contenedores se llama **Istio**, que es un proyecto open-source creado por IBM, Google & Lyft. Tanzu Service Mesh usa Istio como desacoplamiento del plano de datos para manejo de cargas de trabajo de Kubernetes distribuidas, y lo que hace es extender las capacidades de este, agregando más visibilidad, control y seguridad a nivel de aplicaciones y microservicios.



Agregando otro nivel de desacople, esta función presenta los namespaces sin las restricciones que normalmente podría tener una infraestructura; Service Mesh ‘estira’ las redes de namespaces lógicamente, por así decirlo. El resultado es presentado utilizando *Global Namespaces* (GNS), que permiten controlar el acceso más integralmente y tener mayor consistencia en enruteamiento de tráfico y pólizas de seguridad, sin importar donde estén corriendo dichas aplicaciones o servicios.

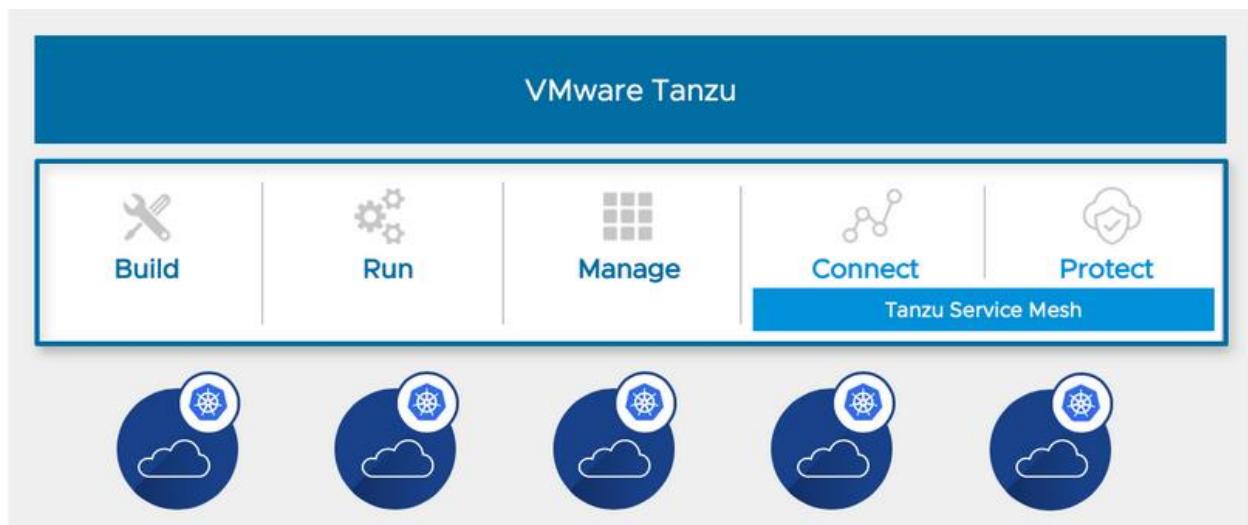
El principal objetivo de Tanzu Service Mesh es conectar y asegurar servicios que cruzan infraestructuras o ambientes desiguales de una manera más ágil cuando las cargas están muy dispersas.

Tanzu Service Mesh puede ser instalado en Tanzu Kubernetes Grid, clústeres de Kubernetes, y puede ser usado en clústeres administrados por Tanzu Mission Control.

CONCLUSIÓN

Con el portafolio tan extenso de Tanzu, se siente como si cada uno de los productos incluidos hubieran sido todos lanzados a la vez; para ser honesto nunca usé ni navegué previamente estas soluciones a profundidad, de hecho, he aprendido mucho contribuyendo con este capítulo del libro.

Lo que sí sé, es que este portafolio es una combinación de soluciones que han venido siendo afiliadas a VMware por medio de adquisiciones en los últimos meses y años. Supongo que anteriormente cada una tenía sus consumidores, pero ahora están siendo todas promovidas como parte del mismo empuje con metas a proveer soluciones a los diferentes retos en cada una de las áreas distinguidas en la siguiente imagen.



Para cada uno de los contornos en las anteriores celdas hay una solución, y eso es lo que hace a Tanzu diferente y llamativo, tienen todo cubierto.

ACTUALIZACIÓN

A mitad del mes de septiembre del 2020, VMware anunció cuatro ediciones diferentes de Tanzu; con esto permiten a sus clientes una manera más granular de adquirir y consumir los productos, los cuales son ‘empaquetados’ en estas diferentes ediciones para su comercialización. Dos de estas ediciones están ya listas este trimestre y las otras dos aún en desarrollo y de las cuales saldrá más información pronto – quizás durante VMworld (?)

Lo publicado hasta ahora en cuanto a estas nuevas ediciones se resume en lo siguiente:

Tanzu Basic: ideal para organizaciones que buscan una solución simple y liviana para correr contenedores estándares encima de vSphere. Esta viene siendo la versión de Tanzu más económica y accesible que permitirá combinar máquinas virtuales conocidas junto a contenedores. Esta edición puede ser licenciada en paquete con vSphere 7 Enterprise+ o como agregado para despliegue en vSphere 6.7U3.

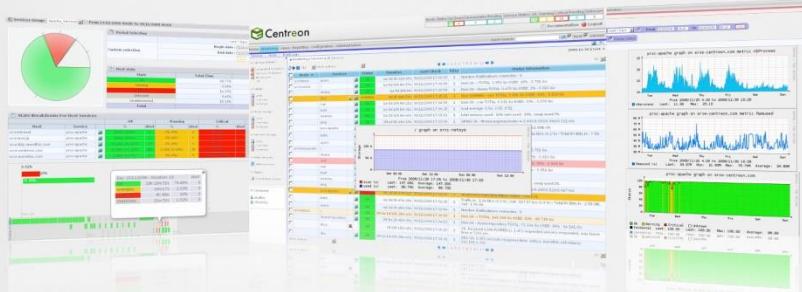
Tanzu Standard: esta edición es para organizaciones que tienen necesidad de administrar y expandir sus cargas de trabajo de contenedores en diferentes nubes, manteniendo consistencia y consola de control única centralizada. Incluye opciones incorporadas para backups, gestión de acceso y monitoreo, entre otras. Su licenciamiento se puede obtener a través de un paquete con VMware Cloud Foundation o como un agregado para despliegue en vSphere 7, 6.7U3, o en nubes públicas. Disponible este trimestre.

Tanzu Advanced: edición enfocada en organizaciones con la necesidad de crear despliegues más complejos que requieren aún más capacidades, y cuyas metas están encaminadas a seguir metodología de DevOps. Permitirá más agilidad en la administración de ambientes más dispersos, heterogéneos, con más énfasis en seguridad para contenedores y microservicios. Hoy en día no hay muchos detalles de licenciamiento o para cuándo estará disponible esta edición, pero quizás durante el VMworld será suministrada más información.

Tanzu Enterprise: en una frase... automatización a escala. Esta edición vendrá con todo lo necesario para administrar ambientes complejos y grandes de una manera más autónoma, eficiente y rápida. Al igual que Advanced, aún no se saben los detalles completos ni el modo de licenciamiento.

Por lo que he leído hasta ahora en redes sociales, estas opciones de diferentes niveles de consumo de Tanzu son esperadas por quienes son entusiastas del producto y van a usar la solución en sus ambientes. Creo que nunca se equivocarán si quieren brindar flexibilidad a clientes, ya que no todas las empresas son de la misma dimensión o requieren las mismas funciones.

openServices^{it}_{.eus}



[Video demo](#)

Monitoriza tu entorno

Descansa & tenlo todo controlado

Expertos en monitorización

Monitorización de negocio

Da movilidad a tu negocio

Trabaja desde cualquier sitio

Desde cualquier dispositivo

De forma segura



Somos distintos

Trato cercano

Honestos



Citrix para administradores de IT

Descarga gratis
eBook 400pág.



Capítulo 13

NSX-T Y MICROSERVICIOS



Elver Sena Sosa
@ELVERs_Opinion

NSX-T Y MICROSERVICIOS

INTRODUCCIÓN

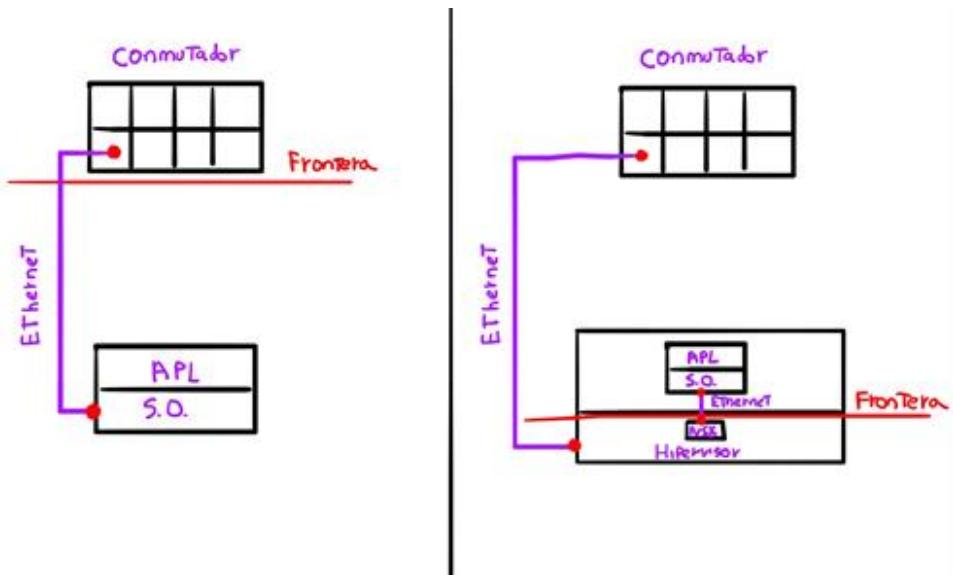
El tema de contenedores y microservicios es un poco complejo para explicar, aunque asumimos que el lector entiende los puntos básicos sobre el tema. Dada la naturaleza de microservicios, explicar cómo se ofrecen servicios de red y seguridad en NSX es aún un poco más complicado. También asumimos un conocimiento básico sobre NSX.

Este capítulo fue escrito con la intención de extender los detalles de las explicaciones sobre el capítulo completo. Usamos muchos diagramas para asistir en las explicaciones, pero aun así se espera que el lector tal vez tenga que leer el capítulo más de una vez para entender todos los detalles. Esperamos que el capítulo les sea de utilidad.

En el mundo de microservicios basados en contenedores, la ejecución de redes y seguridad toma un aspecto diferente al mundo de cargas de trabajos tradicionales, sean físicas o máquinas virtuales. Cuando las cargas de trabajo son físicas, no queda de otra que colocar la frontera donde se empieza a ofrecer el servicio de redes y seguridad en el conmutador de redes donde se conecta (hoy en día con un cable de Ethernet) la carga física.

Cuando las cargas de trabajo son virtuales, así como las máquinas virtuales, existe la opción de mover la frontera más cerca de la máquina de trabajo, como es el caso de NSX. Con NSX, la frontera donde se empieza a ofrecer el servicio de redes y seguridad existe en el hipervisor. Esto permite que se puedan expandir, mejorar y ofrecer nuevos servicios de redes y seguridad que se pueden brindar.

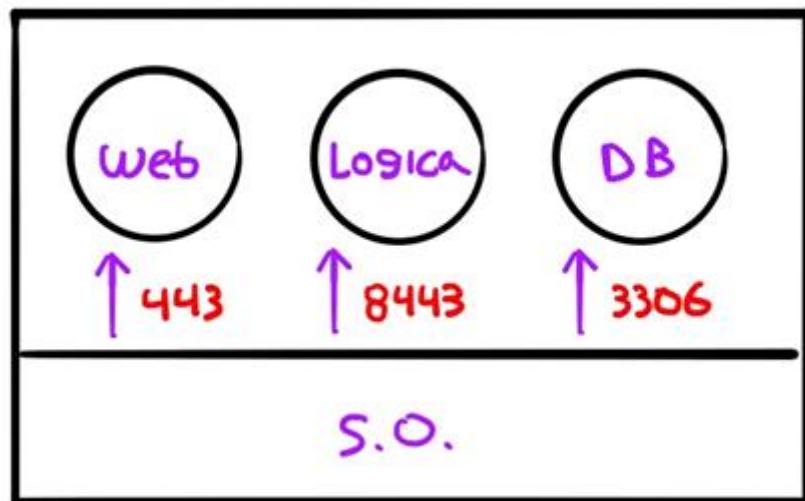
Por ejemplo, en NSX existe una entidad que se llama Cortafuegos Distribuido (DFW por sus siglas en inglés). El DFW permite ofrecer seguridad personal a cada máquina virtual, algo que no es posible de una forma práctica en un entorno donde existen máquinas de trabajo físicas.



Obra Maestra 1: Punto de entrada en la red y seguridad.

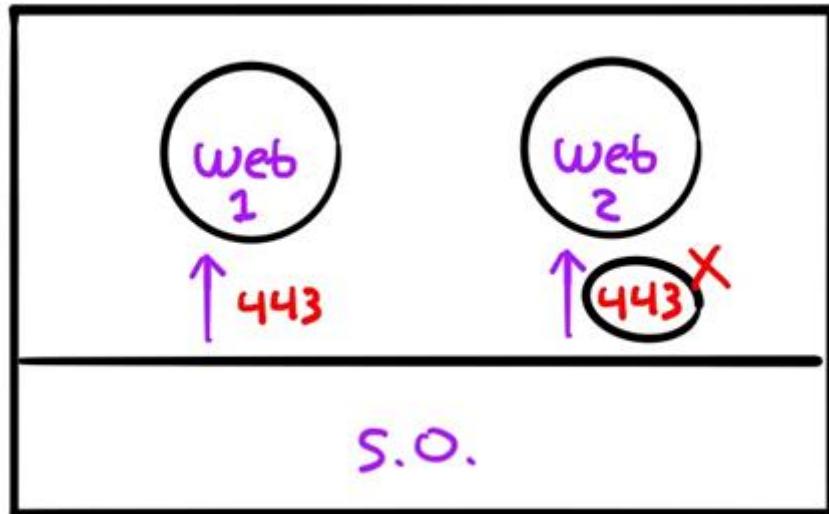
Antes de poder decidir cómo es mejor ofrecer los servicios de redes y seguridad a microservicios basados en contenedores, es necesario repasar como es la interacción entre una aplicación, el sistema de operación donde corre la aplicación y la entidad donde existe la frontera de servicios de redes y seguridad.

Cuando las aplicaciones corren en monolítico (tradicionalmente hablando), varios componentes de la aplicación corren en un mismo sistema de operación. Por ejemplo, si tienes una aplicación de tres niveles (web, lógica y db), cada nivel se registra con el sistema de operaciones y le dice al sistema de operaciones que puerto te TCP o UDP va a utilizar para poder comunicarse con otros programas. Siempre y cuando el puerto esté disponible y no exista otro proceso en el sistema de operaciones usando ese puerto ni el puerto este reservado para otro proceso. Por ejemplo, es universalmente aceptado que el nivel de web va a registrarse con el sistema de operación usando el puerto de TCP 443 y la db como MariaDB va a registrarse para usar el puerto TCP 3306.



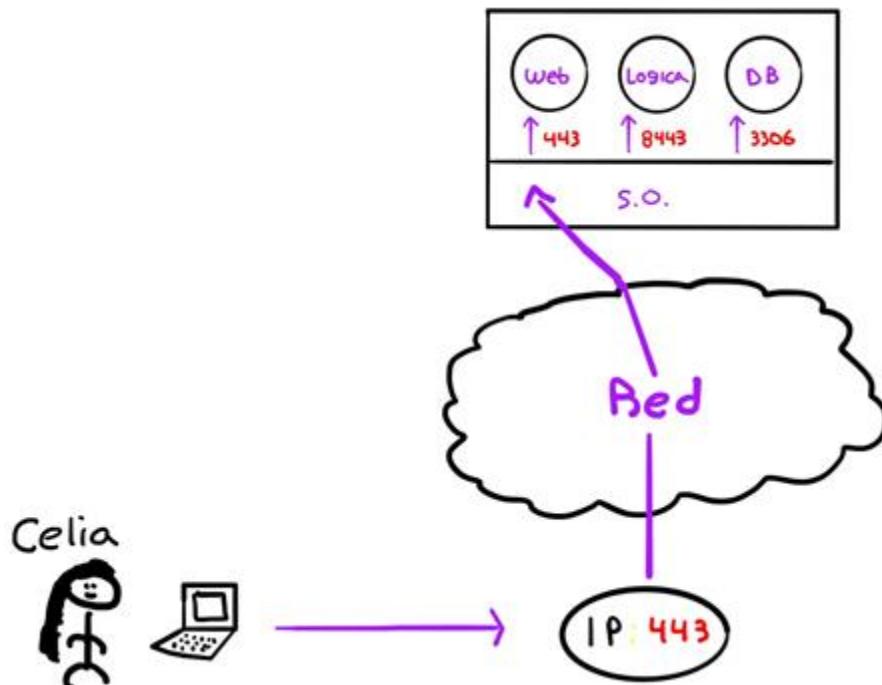
Obra Maestra 2: Niveles en la aplicación registrando puertos con el sistema de Operación.

¿Y qué tal si queremos añadir otra aplicación en el mismo sistema de operación con un nivel que también quiere utilizar el puerto TCP 443? No se puede. El sistema de operación va a generar un error diciendo que el puerto TCP 443 no está disponible. El único remedio es 1) pedir otro número de puerto o 2) mover la nueva a aplicación para otro sistema de operación (otra carga de trabajo).



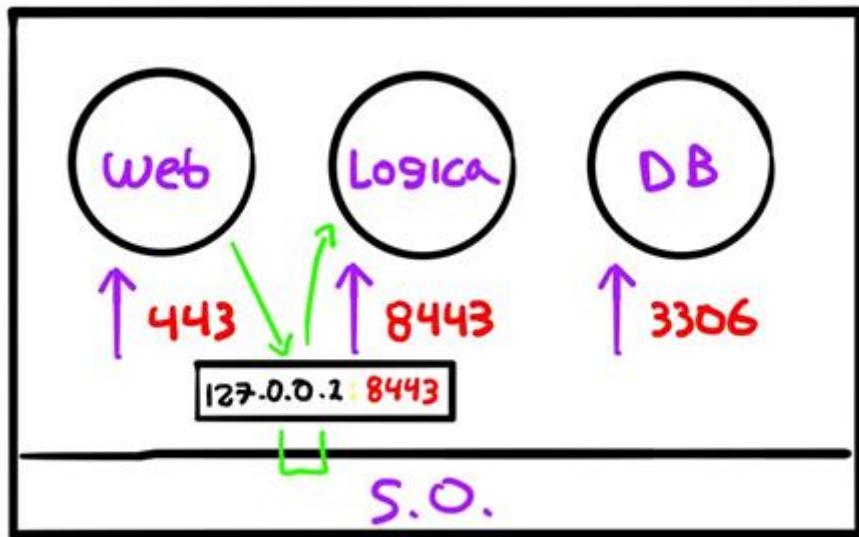
Obra Maestra 3: Un proceso pidiendo un puerto que ya está en uso.

¿Qué tiene que ver esto con redes y seguridad? ¡Todo! Según el protocolo de TCP/IP el cual es el protocolo de defecto para comunicación de redes, toda entidad que se quiera comunicar con otra entidad debe tener una IP (Capa 3 del modelo de OSI) de la otra entidad y el puerto (Capa 4 del modelo de OSI). Resulta que nuestra aplicación va por defecto a heredar la IP del sistema de operación. Si una entidad externa al sistema de operación quiere comunicarse con el nivel de web de nuestra aplicación, va a tener que hacerlo indicando la IP del sistema de operación (Capa 3) y el puerto reservado por nuestro nivel de web (443).



Obra Maestra 4: Usando la IP y el puerto para identificar el destino de una comunicación.

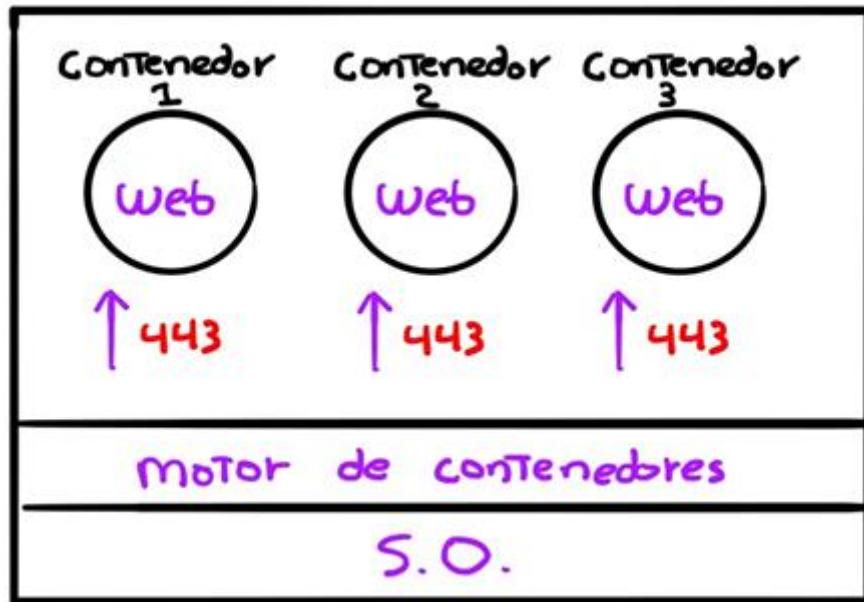
¡Chévere! ¿Pero qué tal si el web de nuestra aplicación quiere comunicarse con el nivel de lógica de nuestra aplicación, que IP utiliza? Puede utilizar la IP del sistema de operación o, que más eficiente, puede usar la IP 127.0.0.1, la cual es reservada para comunicaciones internas al sistema de operaciones. Cuando el sistema de operación recibe de uno de sus procesos una comunicación destinada al IP 127.0.0.1, sabe que el proceso de destino también es interno.



Obra Maestra 5: Procesos comunicándose internamente.

Ahora revisemos brevemente como contenedores funcionan. Una de las metas principales de contenedores es tener la habilidad de correr varios niveles de aplicaciones en el mismo sistema de operaciones mientras le brindan a cada nivel una separación lógica de los otros niveles corriendo en el mismo sistema de operaciones.

Esto incluye ofrecerles a los niveles de la aplicación la habilidad de utilizar el puerto de Capa 4 que deseen, permitiendo que otro nivel en el mismo sistema de operación pueda utilizar ese puerto. Para poder tener contenedores en un sistema de operación es necesario instalar en el sistema de operación una entidad llamada motor de contenedores que se sienta entre los contenedores y el sistema de operación. Es el motor de contenedores el que permite que el mismo puerto de Capa 4 pueda ser usado por distintos niveles de las aplicaciones.

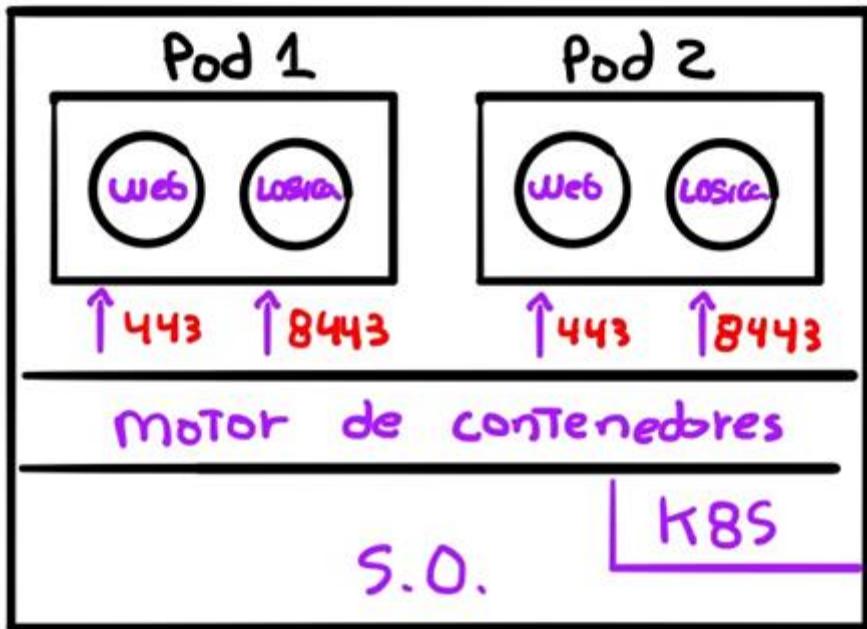


Obra Maestra 6: Varios contenedores usando el mismo puerto.

Vale mencionar que en práctica no es buena idea diseñar una aplicación en la cual varios de sus niveles utilicen el mismo puerto. Lo que es más común en el entorno de contenedores es que niveles que pertenecen a distintas aplicaciones utilicen el mismo puerto. Por ejemplo, una aplicación para leer la temperatura ambiental y otra aplicación para medir la presión atmosférica, ambas pueden tener un nivel de web que utiliza el puerto TCP 443 y ambas corran en el mismo sistema de operación utilizando contenedores.

Kubernetes (k8s de cariño) es una plataforma que ayuda en la administración de aplicaciones en contenedores. Una de las características de Kubernetes es que permite agrupar los contenedores que forman parte de una aplicación para que sean administrados en conjunto. Esta agrupación de contenedores es llamada POD. En Kubernetes los POD corren en cargas de trabajo llamados Nodos Trabajadores (Worker Node en inglés). Está a discreción del creador de la aplicación decidir qué contenedores van a formar parte de un POD y cuantos niveles de la aplicación va a incluir en un mismo contenedor o POD.

Nodo Trabajador



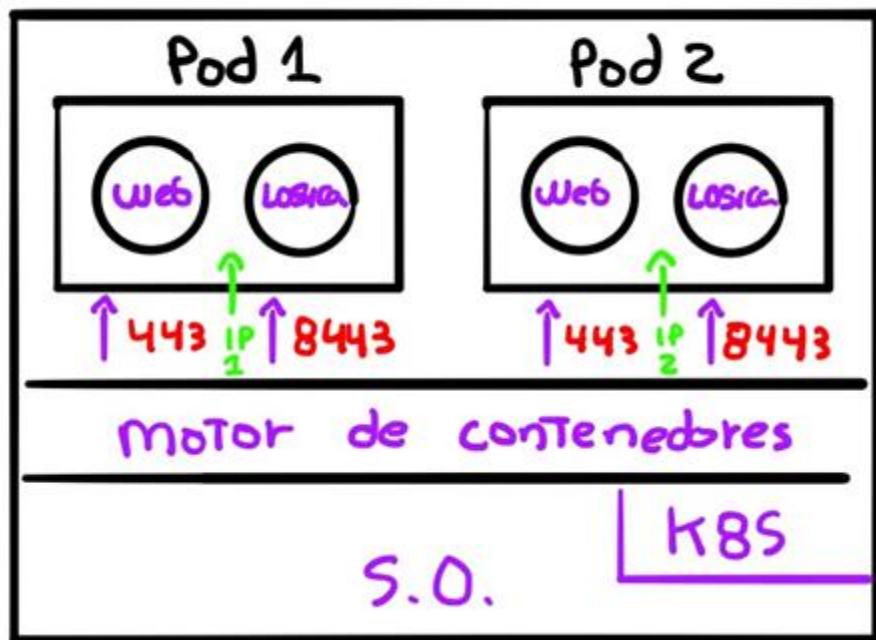
Obra Maestra 7: Un Nodo de Trabajo corriendo varios POD que usan los mismos puertos.

Los Nodos Trabajadores son creados en grupos. Aparte de los Nodos Trabajadores, Kubernetes tiene un componente llamado Nodo Maestro. Cuando un POD tiene que ser instanciado, es el Nodo Maestro que determina en ese momento en cual de todos los Nodos Trabajadores del grupo que estén disponibles instanciar el POD. Si es necesario destruir el POD (ya no se necesita), es el Nodo Maestro que da la orden al Nodo Trabajador que está corriendo el POD. Y si hay una caída de un Nodo Trabajador, el Nodo Maestro entra en acción y busca Nodos Trabajadores disponibles en el grupo para recuperar los POD que fueron afectados por la caída.

Nota: No se han mencionado los motores de contenedores como Docker o Rocket. La razón es simple: el motor de contenedores no es relevante para este tema. Lo que es importante es que los contenedores son agrupados en POD (Kubernetes) y que todos los POD corren sobre el mismo sistema de operación.

Volviendo a los Nodos Trabajadores, sabemos que es posible (y necesario) que dos PODs en el mismo Nodo Trabajador puedan usar el mismo puerto, lo que causa un pequeño reto: ¿Cómo se comunican entidades externas con dos PODs que tengan el mismo puerto? Fácil, asignándoles a cada POD su propia IP. ¡Bingo!

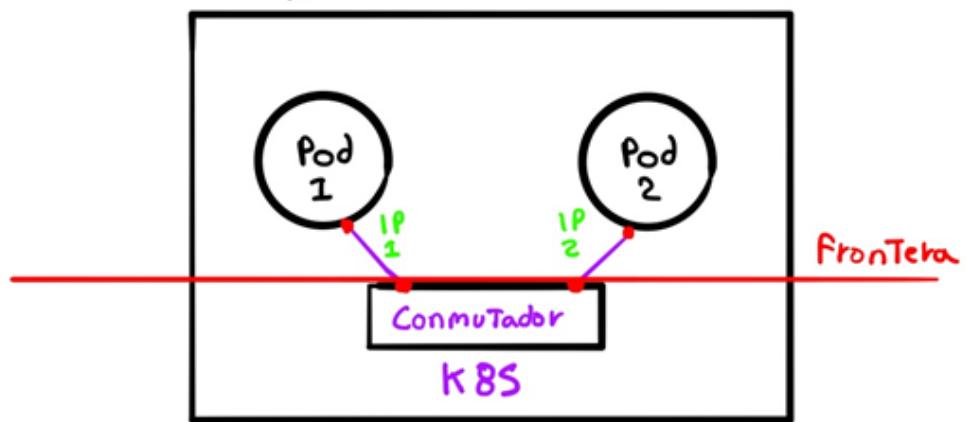
Nodo Trabajador



Obra Maestra 8: Varios PODs en el mismo Nodo de Trabajo con sus propias IP.

La implicación de esta decisión, la que une una entidad (Kubernetes) dentro del sistema de operación entre en el negocio de potencialmente mantener un montón de IP (lo que no es típico), es que la frontera de los servicios de redes y seguridad ahora reside dentro del sistema de operación.

Nodo Trabajador

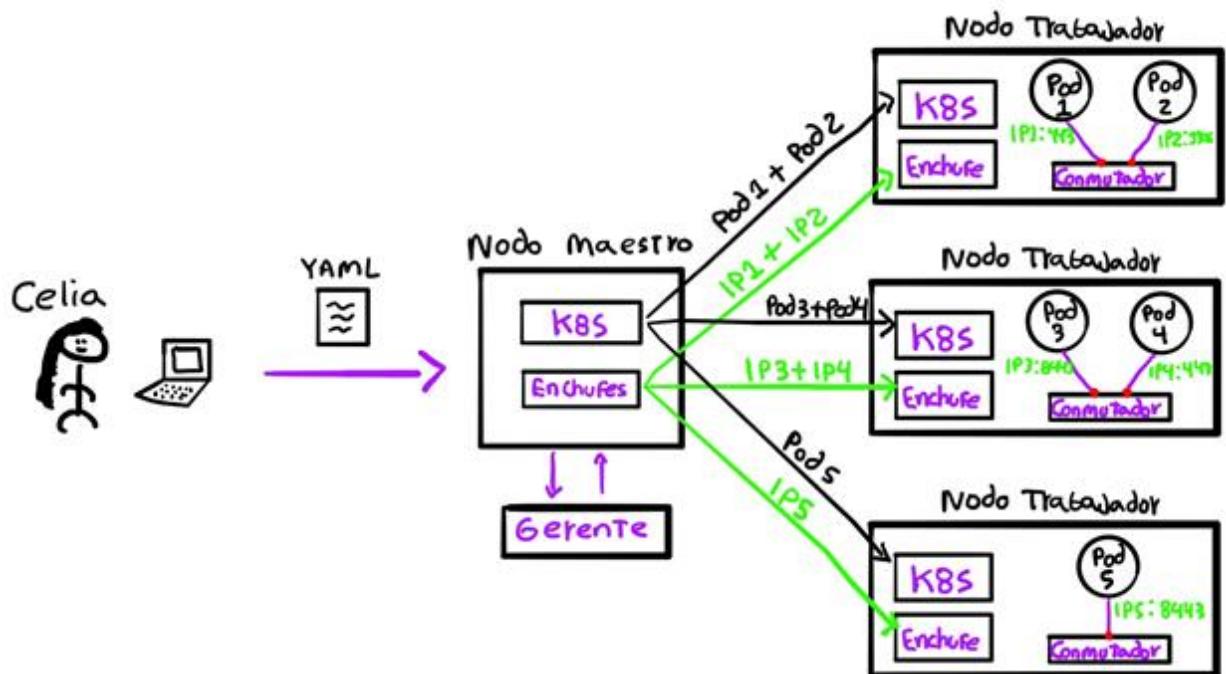


Obra Maestra 9: Punto de entrada de los POD para acceder la red y seguridad.

Cada vez que el Nodo Maestro instancia un POD en un Nodo Trabajador, obtiene una IP para uso exclusivo del POD (mientras esté funcionando) y se le pasa la IP a servicios y enchufes en el sistema de operación del Nodo Trabajador para que se la asigne al POD. Para

completar, el Nodo Maestro es también el que informa del puerto o puertos que el POD va a utilizar.

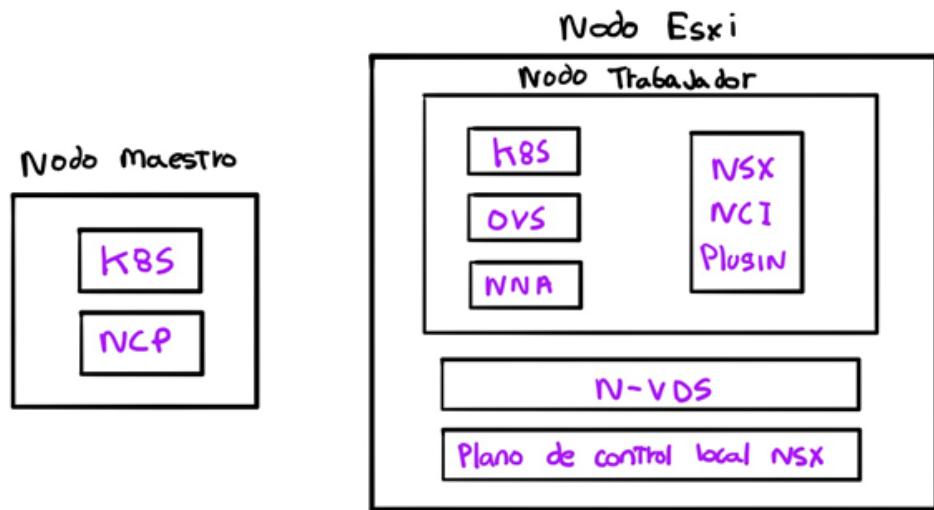
Los números de puertos que el POD va a usar viene del dueño de la aplicación, quien le pasa esa información al Nodo Maestro por medio de un archivo yaml. Por ahora vaya decir que el archivo yaml es el que contiene los detalles de los contenedores que van a ser parte del POD/aplicación y los servicios de infraestructura para ofrecer al POD/aplicación, como Equilibrador de Carga (LB por sus siglas en inglés). Un archivo yaml puede incluir un solo contenedor/POD o varios.



Kubernetes no sabe deletrear redes ni seguridad. Kubernetes depende en otros que ofrezcan servicios en esas áreas, y ahí es que entra NSX. NSX puede ofrecer los siguientes servicios de redes y seguridad a PODs en Kubernetes:

- Servicios de Capa 2, como conmutadores virtuales basados en túneles de GENEVE
- Servicios de Capa 3, como enrutadores distribuidos y lógicos
- Servicios de Capa 4 y 7, como NAT, LB, VPN
- Servicios de seguridad de Capa 2, 3, 4 y un poco de 7
- Reservaciones para uso de IP
- Separación lógica basado en inquilinos

Para ofrecer estos servicios, NSX necesita tener acceso directo al sistema de operación donde corre el Nodo Maestro y al sistema de operación donde corren los Nodos Trabajadores. NSX tiene enchufes en cada uno de ellos, y mediante estos enchufes se comunica con el Nodo Maestro para recibir pedidos de servicios de red y seguridad, y para brindar esos servicios durante la vida de los PODs.

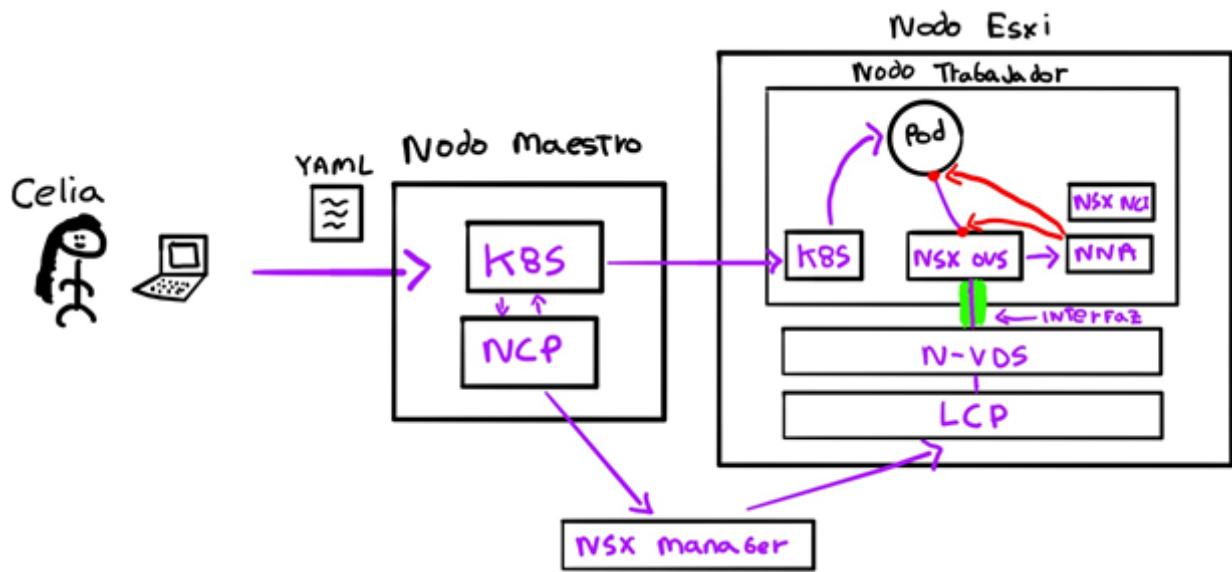


Obra Maestra 11: Los enchufes de NSX.

Nota: NSX para Kubernetes está disponible en Tanzu, la sombrilla de productos de VMware para contenedores. En el resto del capítulo, vamos a hablar de NSX en Kubernetes como está disponible en Tanzu Kubernetes Grid Integrated (TKGI), anteriormente conocido como VMware PKS.

Cuando un Nodo Maestro decide instanciar a un POD, él se comunica con el enchufe de NSX, llamado NSX Container Plugin en inglés (NCP) que está en el mismo sistema de operación que el Nodo Maestro. NCP entonces se comunica con uno de los NSX Managers para que haga las preparaciones necesarias para ofrecer el servicio de red o seguridad requerido para el POD (incluyendo el IP que se le va a configurar al POD).

Esto incluye comunicación con el plano de control local de NSX (LCP por sus siglas en inglés) en el nodo de ESXi donde corre el Nodo Trabajador que es elegido por el Nodo Maestro para instanciar el POD. El LCP entonces se comunica con el enchufe de NSX Node Agent que reside dentro del Nodo Trabajador. El NSX Node Agent configura las conexiones de red necesarias para ofrecer servicios de red y seguridad al POD.

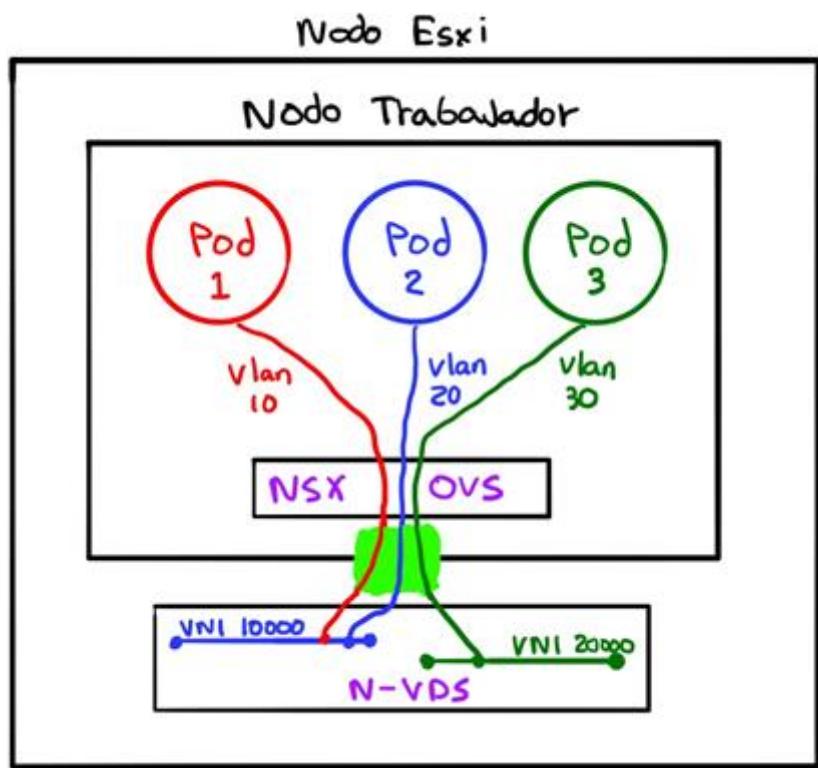


Obra Maestra 12: Creación de un POD y sus servicios de red y seguridad.

Todo Nodo Trabajador corre en una máquina virtual que está conectada con una interfaz virtual a un commutador distribuido virtual de NSX (N-VDS por sus siglas en inglés). La conexión está diseñada para aceptar varias VLAN.

Dentro del Nodo Trabajador existe un commutador virtual llamado Open vSwitch (OVS) que fue ajustado para trabajar con NSX. El OVS es donde termina el otro lado de la conexión al N-VDS. Cuando un POD es instanciado en el Nodo Trabajador, el POD es conectado al OVS y configurado bajo las direcciones del NSX Node Agent. Esas direcciones incluyen la IP del POD, una VLAN y una etiqueta basada en gran parte por los rótulos asignados por el Nodo Maestro.

La conexión de cada POD en el OVS es identificada por un numero de VLAN que es único dentro de cada Nodo Trabajador. Cuando el POD envía tráfico, el tráfico es mandado por el OVS al N-VDS, y el N-VDS utiliza el número de VLAN para determinar los servicios de redes y seguridad que el POD va a recibir.



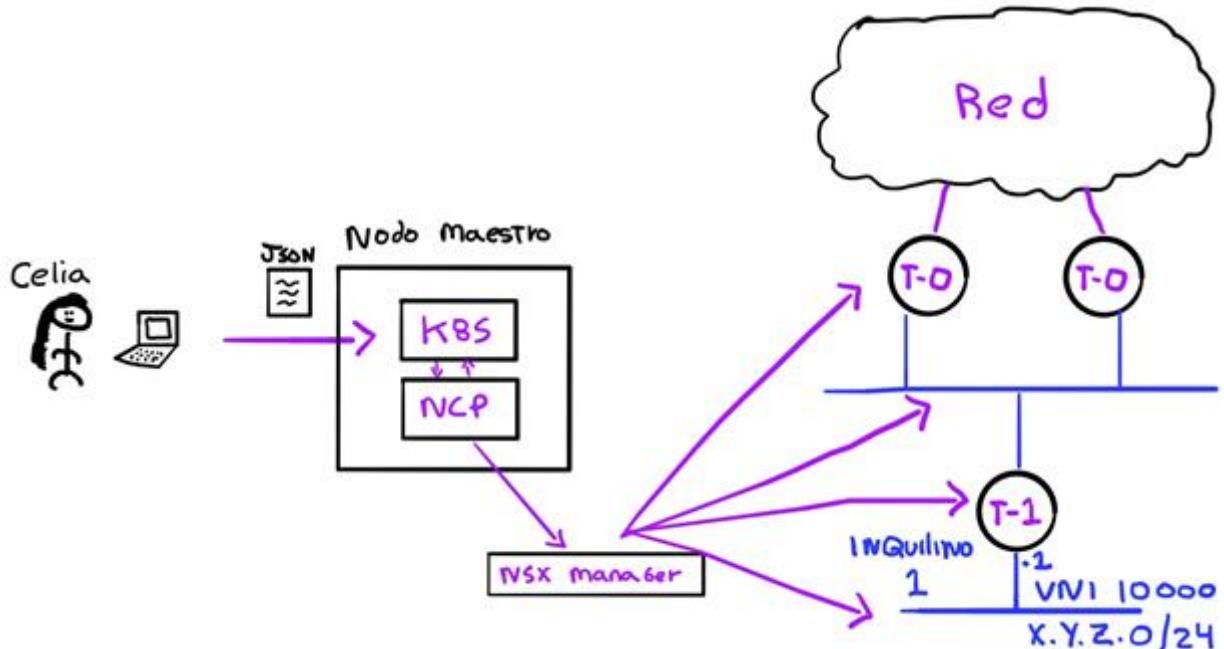
Obra Maestra 13: Conexión de un POD a un commutador lógico.

Ahora tomemos un momento para revisar, y añadir unos pequeños detalles, como es que un POD es creado y preparado para recibir servicios de redes y seguridad:

1. El Nodo Maestro recibe un pedido para crear un POD. Dentro de este pedido se indica los contenedores que van a ser instanciados, los puertos de capa 4 para utilizar y los servicios de red y seguridad deseados.
2. El Nodo Maestro le informa a NSX, por medio del NCP, que va a crear un POD y le pasa la información.
3. NSX toma los pasos necesarios para obtener una IP para el POD, asignarle una interfaz privada, conectar el POD a la capa dos adecuada y activar los otros servicios de redes y seguridad que el POD necesita.
4. El POD es instanciado (pero no está en servicio) en un Nodo Trabajador y luego que NSX termina de prepararlo todo, permite, por medio del NSX Node Agent, que el POD entre en funcionamiento.

NSX tiene la habilidad de crear muchos commutadores lógicos (más de dieciséis millones de ellos), así que tiene que haber un método para que NSX sepa en qué commutador lógico va a conectar a un POD. En TKGI antes de crear un POD, tiene que existir un inquilino (tenant en inglés). En Kubernetes los inquilinos son equivalente a un namespace. Cuando en Kubernetes se crea un namespace, NSX automáticamente crea un enrutador distribuido de tipo T-1 y por lo menos un commutador lógico el cual es conectado al enrutador distribuido T-1. NSX reserva una subred de clase C (/24) y le asigna la primera IP a la interfaz del enrutador T-1 y utiliza las otras IP de la clase C para asignarla a los POD. El enrutador T-1 va a ser la

puerta de enlace predeterminada (Default Gateway en inglés) para todos los POD del inquilino. Si se utilizan todas las IP de la clase C, entonces NSX obtiene otra clase C, crea otro commutador lógico, conecta el enrutador T-1 del inquilino al nuevo commutador lógico, asignándole la primera IP al enrutador T-1, y todo POD nuevo entonces es conectado al nuevo commutador lógico.



Obra Maestra 14: Creación de un inquilino (namespace) en TKGI y NSX.

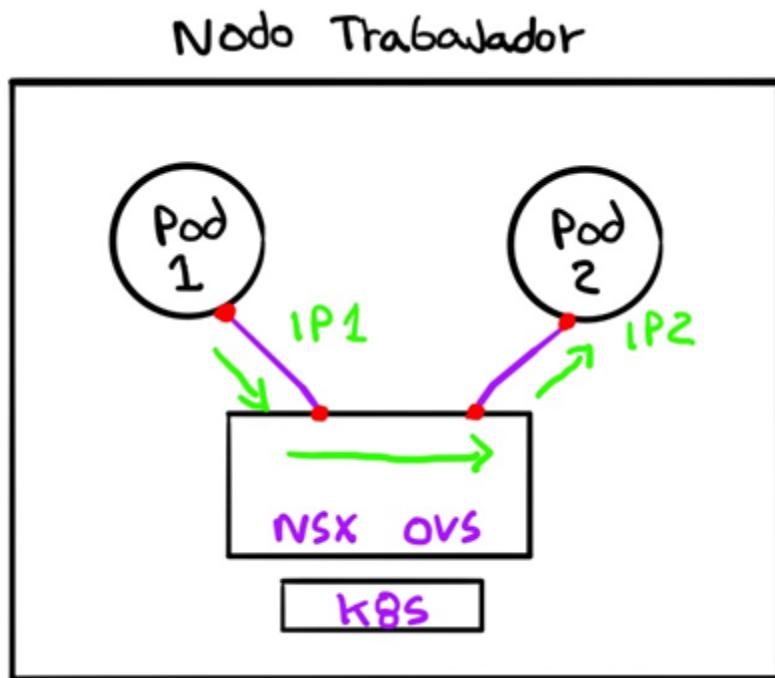
Cuando el Nodo Maestro quiere crear un POD, es requerido que el archivo yaml tenga información en cual namespace el POD va a existir. La información del namespace es pasada a NSX Manager para que determine cual commutador lógico va el POD a ser conectado. También, el rotulo del inquilino es añadido a la etiqueta que se le asigna al POD. Por ultima, NSX apoya la existencia de más de un inquilino en TKGI. Cada inquilino tendrá su propio enrutador T-1, los cuales se conectan a dos enrutadores T-0.

Tomemos unos momentos para ver como un POD se comunica con otro POD, ambos en el mismo inquilino y commutador lógico.

Si los POD están en el mismo Nodo Trabajador:

1. El POD1 crea una trama para POD2 y se la pasa al OVS.
2. El OVS revisa:
 - a. Si las reglas de seguridad permiten que el POD1 se comunique con el POD2.
 - i. ¡Ah caramba, no les había dicho antes! Mala mía. ¿Se recuerdan de las etiquetas que se le asignan a los POD? Bueno, usando esas etiquetas, NSX determina que reglas de seguridad le aplican a cada POD y aplica esas reglas (Capas 2, 3 y 4) en el punto de entrada de la red. Las reglas de seguridad son creadas por el administrador de seguridad en la infraestructura.
 - b. Si el POD2 es local o externo.

- II. En este caso es local.
- 3. El OVS le pasa la trama al POD2.

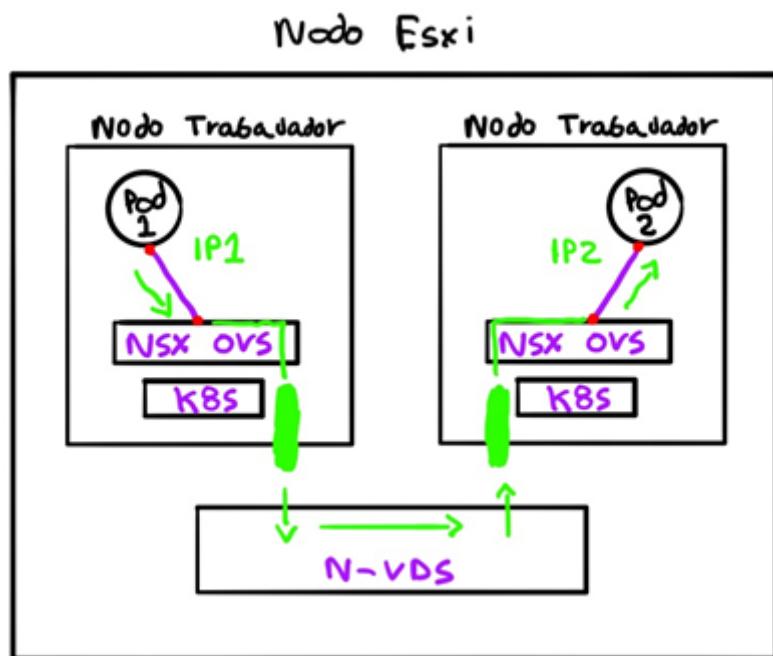


Obra Maestra 15: Comunicación de dos POD en el mismo Nodo de Trabajo.

Si los POD están en diferentes Nodos Trabajadores, pero en el mismo nodo de ESXi.

1. El POD1 crea una trama para POD2 y se la pasa al OVS.
2. El OVS revisa:
 - a. Si las reglas de seguridad permiten que el POD1 se comunique con el POD2.
 - b. Si el POD2 es local o externo
 - I. En este caso es externo.
3. El OVS le pasa la trama al N-VDS.
 - a. Mas detalles: cuando el OVS le pasa la trama al N-VDS, el OVS incluye el número de VLAN del POD1.
4. El N-VDS determina si POD2 es local o externo.
 - a. Varios puntos acá:
 - I. Primero, el punto de referencia (local o externo) es el nodo de ESXi.
 - II. Segundo, Y para ser más específico, es el proceso del conmutador lógico, dentro del N-VDS, al cual la VLAN del POD1 fue asignado que toma la decisión.

1. Este conmutador lógico es el que se conecta al enrutador T-1 del inquilino que pertenecen POD1 y POD2.
- III. Tercero, del punto de vista del N-VDS, local es otra máquina virtual corriendo un Nodo Trabajador.
 - b. En este caso POD2 está local
5. El N-VDS le pasa la trama al OVS en el Nodo Trabajador donde está POD2.
 - a. En la trama se incluye la VLAN del POD2.
 - I. La VLAN del POD2 es distinta y no tiene *nada que ver* con la VLAN del POD1.
6. El OVS revisa:
 - a. Si las reglas de seguridad permiten que el POD1 se comunique con el POD2.
 - I. Esta son las reglas de seguridad aplicadas usando las etiquetas del POD2.
 - b. El puerto de salida del POD2.
7. El OVS le pasa la trama al POD2.

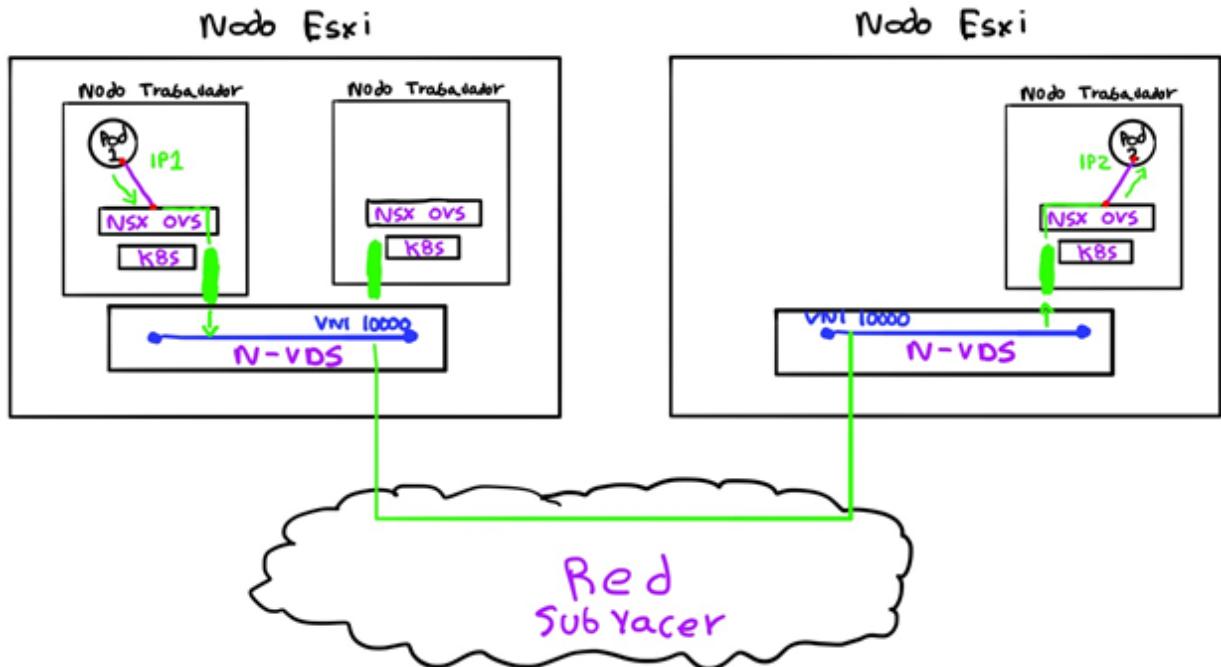


Obra Maestra 16: Comunicación de dos POD en diferentes Nodos de Trabajo.

Si los POD están en diferentes nodos de ESXi.

1. El POD1 crea una trama para POD2 y se la pasa al OVS.
2. El OVS revisa:
 - a. Si las reglas de seguridad permiten que el POD1 se comunique con el POD2.
 - b. Si el POD2 es local o externo.
 - I. En este caso es externo.
3. El OVS le pasa la trama al N-VDS.
 - a. El proceso del N-VDS es el dueño de las conexiones a las máquinas virtuales y los enlaces ascendentes físicos para comunicarse con el exterior.

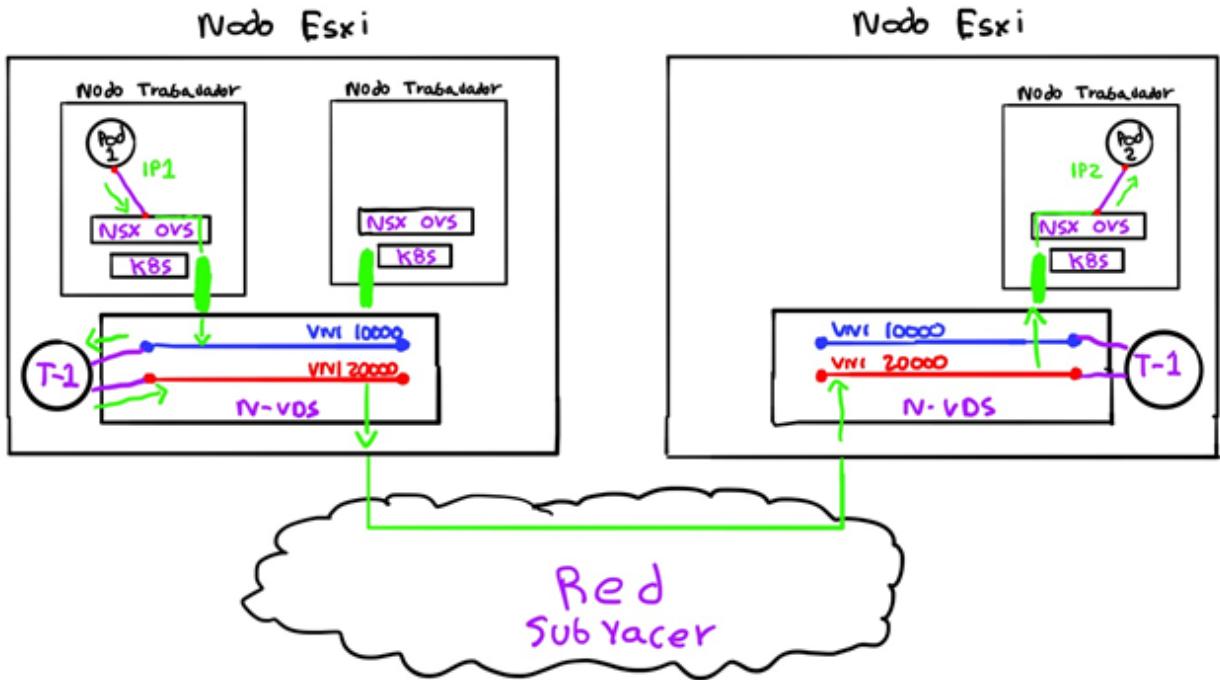
4. El conmutador lógico determina si es local o externo.
 - a. En este caso POD2 está externo.
 - I. Recuerda que es el proceso del conmutador lógico que toma esta determinación.
5. El conmutador lógico determina el nodo de ESXi del destino, entra la trama del POD1 en un túnel de GENEVE y manda el túnel por la red subyacente física (underlay en inglés).
 - a. El número de VXLAN (VNI) que se utiliza es determinado por NSX Manager cuando el conmutador lógico es creado.
6. El nodo de ESXi donde está el Nodo Trabajador donde está el POD2 recibe el túnel.
 - a. En particular, el túnel es procesado por el conmutador lógico, cual saca la trama de POD1 del túnel.
7. El N-VDS le pasa la trama al OVS en el Nodo Trabajador donde está POD2.
 - a. En la trama se incluye la VLAN del POD2.
 - I. Esta VLAN no tiene *nada que ver* con la VLAN del POD1.
8. El OVS revisa:
 - a. Si las reglas de seguridad permiten que el POD1 se comunique con el POD2.
 - b. El puerto de salida del POD2.
9. El OVS le pasa la trama al POD2.



Obra Maestra 17: Comunicación de dos POD en diferentes nodos de ESXi.

Ahora vamos a añadir un poco de complejidad. Vamos a mantener los POD en el mismo inquilino, pero vamos a ponerlos en diferentes conmutadores lógicos. También vamos a asumir que los POD están en diferentes nodos de ESXi.

1. El POD1 crea una trama para POD2 y se la pasa al OVS.
2. El OVS revisa:
 - a. Si las reglas de seguridad permiten que el POD1 se comunique con el POD2.
 - b. Si el POD2 es local o externo.
 - I. En este caso es externo.
3. El OVS le pasa la trama al N-VDS.
4. El commutador lógico determina si es local o externo.
 - a. Bueno, en verdad el commutador lógico va a ver que el destino (Capa 2) es el enrutador T-1 del inquilino ya que POD1 y POD2 están en diferentes redes.
 - b. Y muy, muy importante: este es el commutador lógico al que se conecta POD1. El commutador lógico que se conecta POD2 es diferente (tiene otro VNI).
5. El commutador lógico le pasa la trama al enrutador T-1.
6. El enrutador T-1 determina que POD2 está conectado localmente.
 - a. De nuevo, esto de local es relativo. Lo que le importa al enrutador T-1 es que él tiene una conexión en la subred donde reside el POD2. Es decir, es local.
7. El enrutador T-1 le pasa la trama al commutador lógico donde se conecta POD2.
8. El commutador lógico determina el nodo de ESXi donde reside POD2, entra la trama del POD1 en un túnel de GENEVE y manda el túnel por la red subyacente física.
 - a. Si POD2 hubiera estado local (recuerda, relativo), no se hubiera creado un túnel de GENEVE.
9. El nodo de ESXi donde está el Nodo Trabajador donde está el POD2 recibe el túnel.
 - a. Donde el commutador lógico donde se conecta POD2 saca la trama de POD1 del túnel.
10. El N-VDS le pasa la trama al OVS en el Nodo Trabajador donde está POD2.
 - a. En la trama se incluye la VLAN del POD2.
11. El OVS revisa:
 - a. Si las reglas de seguridad permiten que el POD1 se comunique con el POD2.
 - b. El puerto de salida del POD2.
12. El OVS le pasa la trama al POD2.

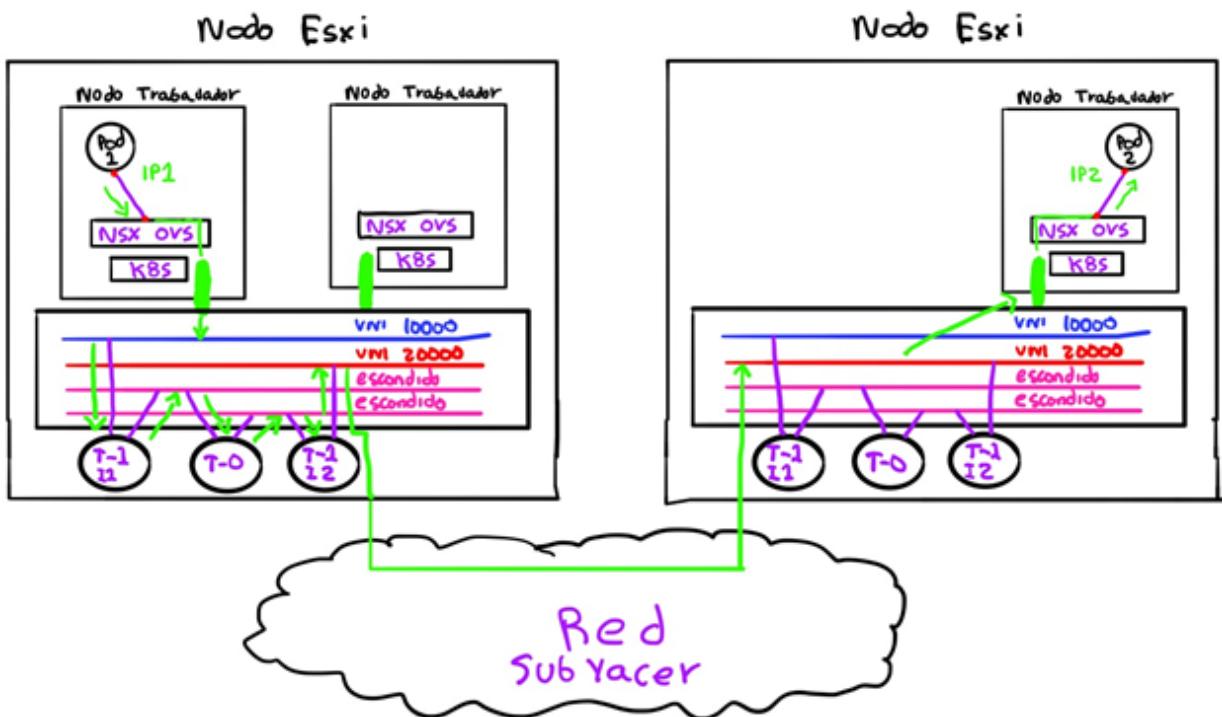


Obra Maestra 18: Comunicación de dos POD en diferentes conmutadores lógicos.

Ahora veamos lo que ocurre cuando POD1 y POD2 están en diferentes inquilinos.

- El POD1 crea una trama para POD2 y se la pasa al OVS.
- El OVS revisa:
 - Si las reglas de seguridad permiten que el POD1 se comunique con el POD2.
 - Si el POD2 es local o externo.
 - En este caso es externo.
- El OVS le pasa la trama al N-VDS.
- El conmutador lógico determina si es local o externo.
- El conmutador lógico le pasa la trama al enrutador T-1 del inquilino donde pertenece POD1.
- El enrutador T-1 determina que POD2 no está conectado localmente.
 - De nuevo, esto de local es relativo. Lo que le importa al enrutador T-1 es que él tiene una conexión en la subred donde reside el POD2. Es decir, es local.
- El enrutador T-1 sigue su ruta de defecto y le pasa la trama a uno de los dos enrutadores T-0.
 - En NSX, todos los enrutadores T-1 tienen ruta de defecto en dirección a los enrutadores T-0.
- El enrutador T-0 recibe la trama y determina que el destino está en dirección del enrutador T-1 del inquilino donde está POD2.
 - En TKGI, todos los POD solo utilizan a enrutadores T-1 como puerta de enlace predeterminada.
- El enrutador T-0 le pasa la trama al enrutador T-1.

- a. Hasta ahora, todo esto ha ocurrido en el kernel del nodo de ESXi donde reside el POD1.
10. El enrutador T-1 del inquilino donde reside el POD2 le pasa la trama al conmutador lógico donde se conecta POD2.
 11. El conmutador lógico determina el nodo de ESXi donde reside POD2, entra la trama del POD1 en un túnel de GENEVE y manda el túnel por la red subyacer física.
 12. El nodo de ESXi donde está el Nodo Trabajador donde está el POD2 recibe el túnel.
 13. El N-VDS le pasa la trama al OVS en el Nodo Trabajador donde está POD2.
 - a. En la trama se incluye la VLAN del POD2.
 14. El OVS revisa:
 - a. Si las reglas de seguridad permiten que el POD1 se comunique con el POD2.
 - b. El puerto de salida del POD2.
 15. El OVS le pasa la trama al POD2.



Obra Maestra 19: Comunicación de dos POD en diferentes inquilinos.

Por último, vamos a examinar como NSX ofrece servicios de LB a aplicaciones basadas en microservicios. Vamos a enfocarnos en tráfico que proviene fuera del entorno de NSX y TKGI.

Cuando el Nodo Maestro ejecuta los servicios pedidos en el archivo yaml, una de sus tareas es mantener el estado deseado (desired state en inglés) de la aplicación. Este estado puede incluir cuantas copias del POD que deben ser creadas (y mantenidas). Si mas de una copia del POD es creada, es casi siempre requerido un equilibrador de cargas. NSX ofrece servicios de LB para PODs y estos servicios son configurados dinámicamente. El proceso es parte de la creación del POD:

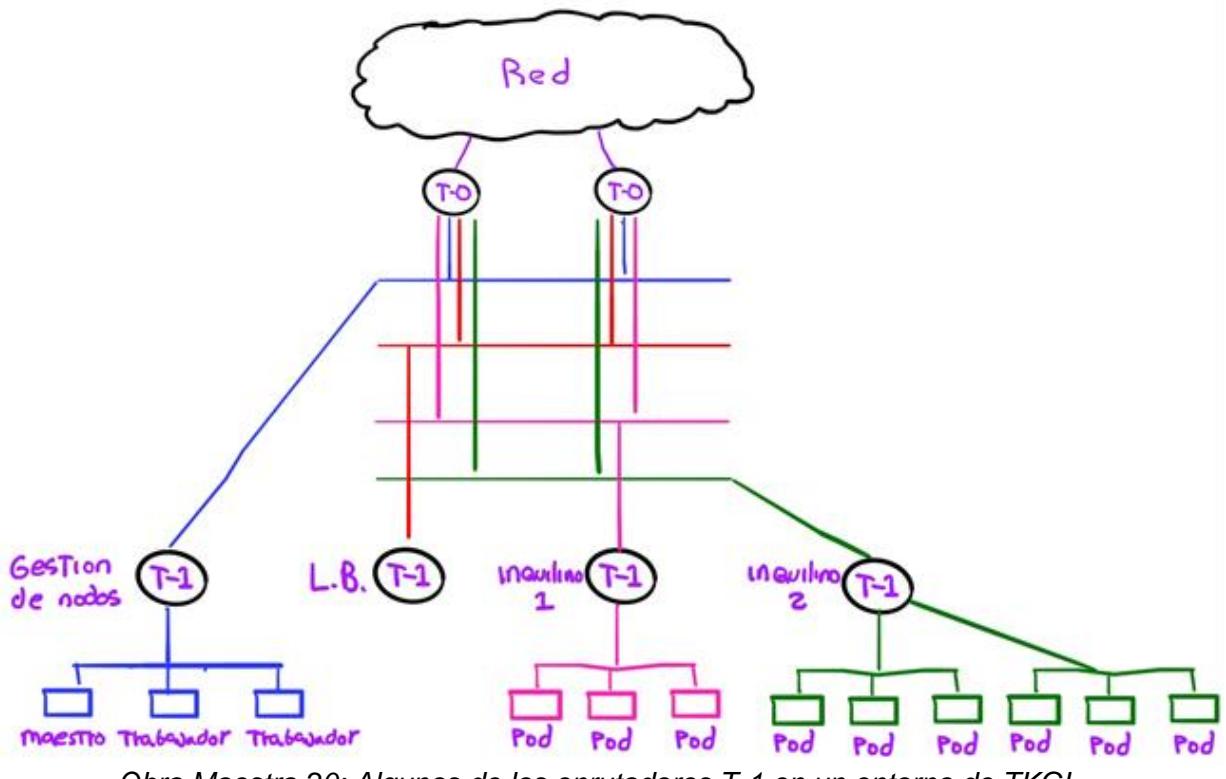
1. El Nodo Maestro recibe un pedido para crear un POD con más de una copia. Dentro de este pedido se indica los contenedores que van a ser instanciados, los puertos de capa 4 para utilizar y los servicios de red y seguridad deseados. Uno de esos pedidos sería el de LB. Sería ilógico pedir servicio de LB para un solo POD.
2. El Nodo Maestro le informa a NSX, por medio del NCP, que va a crear un POD y le pasa la información.
3. NSX toma los pasos necesarios para obtener una IP para el POD, asignarle una interfaz privada, conectar el POD a la Capa 2 adecuada.
 - a. Más relevante para nosotros ahora, NSX Manager también crea una política de LB que incluye añadir la IP del POD al grupo de servidores de la política y obtener una IP externa para usar como IP Virtual (VIP por sus siglas en inglés).
4. El POD es instanciado (pero no está en servicio) en un Nodo Trabajador y luego que NSX termina de prepararlo todo, permite, por medio del NSX Node Agent, que el POD entre en funcionamiento.

De importancia es entender dónde se ejecutan las políticas del LB. Para entenderlo, tomemos un momento para hablar un poco de la arquitectura de NSX en un entorno de TKG.

Cuando se implementa una solución de TKG, no hay ningún nodo de Kubernetes instalando. La instalación de nodos de Kubernetes tiene que hacerse aparte. El administrador de la plataforma decide cuando instalar un entorno de Kubernetes, lo que incluye Nodos Maestros (ellos trabajan en grupo para ofrecer redundancia) y la cantidad de Nodos Trabajadores que van a ser administrados por los Nodos Maestros.

Todos ellos son conectados a su propio enrutador T-1 en un conmutador lógico privado (ningún POD se conectará a este conmutador ni tendrá acceso al enrutador T-1). También es instalado un segundo enrutador T-1 que es utilizado solo para ofrecer servicios de LB. Es en este enrutador T-1 donde se configuran las políticas de LB de todos los PODs creados por el Nodo Maestro.

Nota: Una de las cualidades de TKG es la habilidad de crear más entornos de Kubernetes independientes. Cada entorno de Kubernetes recibe su propio enrutador T-1 para equilibrar cargas.

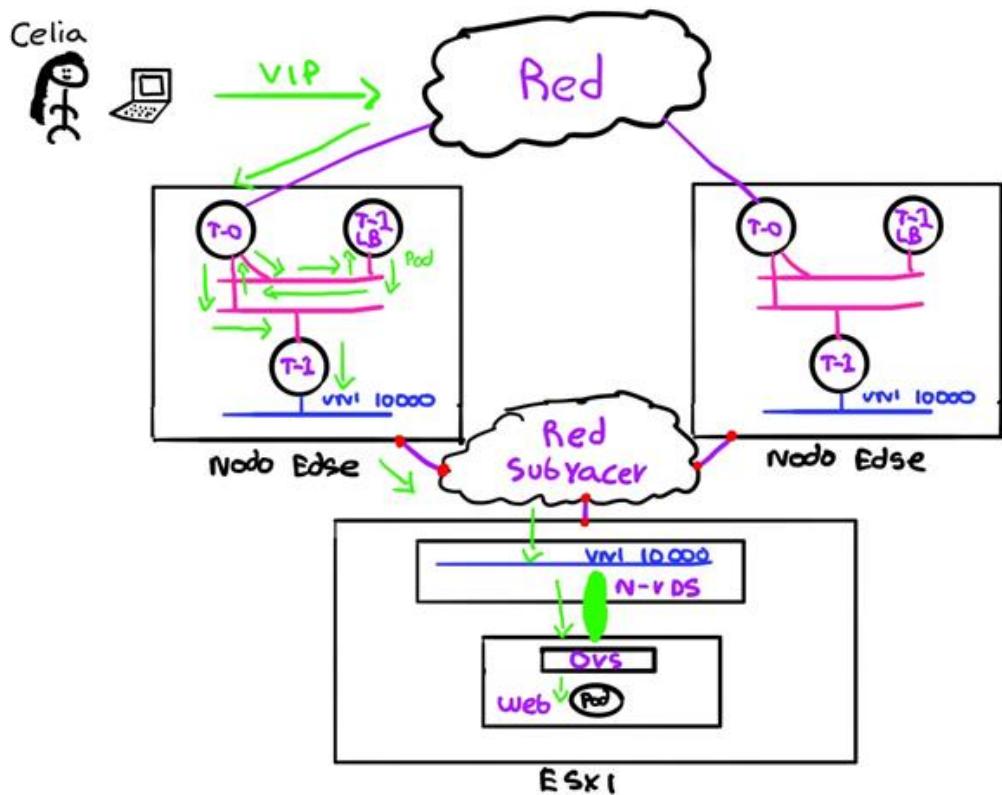


Obra Maestra 20: Algunos de los enruteadores T-1 en un entorno de TKGI.

Ahora veamos como una entidad fuera del entorno de TKGI se comunica con un POD usando el LB.

1. Tú quieres visitar una página de web para ver la temperatura de tu vecindario. Esta página está corriendo en una aplicación donde el nivel de web está en un POD.
2. Abres Safari y visitas la página. Esto genera un paquete que eventualmente llega a uno de los dos enruteadores T-0 de TKGI.
 - a. Recuerda que la dirección de la página de web de tu punto de vista es el VIP.
3. El enruteador T-0 determina que el VIP está en dirección al enruteador T-1 que ofrece servicios de LB.
 - a. *Muy importante:* el enruteador T-0 que recibe tu paquete reside en un nodo de NSX llamado Edge. El enruteador T-1 que ofrece servicios de LB puede que resida en el mismo nodo Edge o en otro nodo Edge. Si reside en otro nodo Edge, el paquete va a ser mandado entre nodos por medio de un túnel GENEVE.
4. El enruteador T-1 recibe tu paquete, substituye la VIP por la IP uno de los POD de la aplicación que corren el nivel de web.
 - a. Recuerda que hay varios POD corriendo el nivel de web de la aplicación.
5. El enruteador T-1 también substituye la IP tuya (la fuente) por la IP de él.
 - a. Esto lo hace para forzar el tráfico de retorno a ti (desde el POD) a que regrese donde el para reemplazar la VIP de nuevo antes de regresarte el paquete.
6. El enruteador T-1 determina como llegar al POD.

- a. Bueno, este enrutador T-1 solo tiene una ruta, que es por vía de uno de los dos enrutadores T-0 en TKGI, así que le manda el paquete a uno de los enrutadores T-0.
 - i. Del punto de vista de los enrutadores T-0, este es un paquete diferente ya que la fuente y destino de IP son diferentes al paquete que tu mandaste.
 - b. Otro punto interesante: no importa que enrutador T-0 el enrutador T-1 elija, el enrutador T-0 va a estar en el mismo nodo Edge.
7. El enrutador T-0 recibe el paquete y determina que el destino está en dirección del enrutador T-1 del inquilino donde reside el POD con el nivel web.
- a. Este enrutador T-1 es diferente al enrutador T-1 que ofrece el servicio de equilibrador de cargas.
8. El enrutador T-0 le pasa el paquete al enrutador T-1.
- a. El cual también está en el mismo nodo Edge.
 - i. ¿Y qué clase de brujería es esta?
9. El enrutador T-1 le pasa la trama al commutador lógico donde se conecta el POD con el nivel web.
10. El commutador lógico determina el nodo de ESXi donde reside el POD con el nivel web, entra tu paquete (con las IP alteradas) a un túnel de GENEVE y manda el túnel por la red subyacente física.
11. El nodo de ESXi donde está el Nodo Trabajador donde está el POD con el nivel web recibe el túnel.
12. El N-VDS le pasa la trama al OVS en el Nodo Trabajador donde está POD con el nivel web.
13. El OVS revisa:
- a. Si las reglas de seguridad permiten haya comunicaciones sobre el puerto.
 - b. El puerto de salida del POD2.
14. El OVS le pasa la trama al POD con el nivel web.



Obra Maestra 21: Accediendo un POD desde fuera de TKGI por medio del equilibrador de cargas.

Algunas aclaraciones: anteriormente usé la palabra *trama* y ahora usé la palabra *paquete*. La diferencia es que un paquete incluye información hasta la Capa 3 (IP), mientras que la trama incluye información hasta la Capa 2 (MAC).

Y hablando de MAC, ahora sería bueno mencionar un detallito. El MAC que se le asigna a los POD también es proveído por NSX Manager al momento de la creación del POD. Cuando un POD es destruido, el MAC y la IP, son devueltas a NSX Manager para que sean reusados en otros POD.

Y ahora si, por último. En verdad, no se mencionó mucho lo de seguridad. Solo se mencionó en el contexto del OVS y las etiquetas. Las políticas de seguridad también pueden ser aplicadas en el nodo de ESXi, a través del DFW, y en los enrutadores. Todo depende del servicio de seguridad que es requerido y las políticas de la empresa en torno a microservicios.

Lo más importante para entender sobre servicios de seguridad y NSX es que no importa donde un POD es instanciado ni cuantos del mismo POD haya. NSX Manager se va a encargar de que las políticas de seguridad sean aplicadas siempre consistente y correctamente.

QloudEA

ALMACENAMIENTO
Y SEGURIDAD DE DATOS

BACKUP
NAS - SAN - NUBE

QLOUDEA.COM

vmware®

NUTANIX

Microsoft
Hyper-V



Windows
Server



Office 365

Capítulo 14

DEL DATACENTER FÍSICO A LAS NUBES



Daniel Romero

@drsromero

DEL DATACENTER FÍSICO A LAS NUBES

INTRODUCCIÓN

La computación en la nube o *cloud computing* ha llegado para quedarse. Cada vez hay más compañías que apuestan por desplegar sus entornos de TI a través de un proveedor de *cloud* público. Tal es así, que las dos grandes compañías de virtualización - VMware y Nutanix - están haciendo una gran apuesta por la nube híbrida, realizando alianzas con los principales proveedores de servicios en la nube.

El *cloud computing* se basa en tres pilares fundamentales que son:

- **Agilidad** para desplegar recursos tecnológicos en cuestión de minutos.
- **Elasticidad** para escalar o disminuir los recursos en función de la demanda.
- **Ahorro de costes** pagando sólo por aquello que usas y sin la necesidad de una inversión inicial.

Este capítulo se centra en los aspectos básicos de computación que ofrecen las tres grandes compañías de nube pública como son Amazon Web Service, Microsoft Azure y Google Cloud. Al finalizarlo serás capaz de identificar los componentes más significativos de un centro de datos: máquina virtual, almacenamiento, red virtual, seguridad, etc. Y relacionarlos con los ofrecidos por los proveedores *cloud* públicos.

Por otro lado, no encontrarás capturas relacionadas con la consola de los proveedores al igual que tampoco se mencionarán los precios ni los tamaños específicos de las máquinas virtuales. Es muy común que estos elementos varíen cada cierto tiempo. Para ello te invito a que te aventures a completar y profundizar la información aquí recogida.

COMPUTO EN AMAZON WEB SERVICES

AWS ofrece diferentes servicios que permiten desplegar, en su infraestructura, todos aquellos recursos de computación que una compañía necesita para procesar, servir o almacenar datos como si estuviesen alojados en centro de datos físico.

Realmente todos los recursos que puedes aprovisionar en cualquier proveedor de servicios en la nube se encuentran alojados en lugares físicos, repartidos por todo el mundo. Una de las grandes ventajas del *cloud computing*, como usuario, es olvidarse del mantenimiento de la infraestructura física y el ahorro de costes que ello supone.

Es muy importante tener en cuenta una serie de conceptos básicos que te ayudarán a entender el funcionamiento de los servicios que ofrece AWS con relación al cómputo. En los siguientes apartados se irán explicando cada uno de ellos.

REGIÓN Y ZONA DE DISPONIBILIDAD

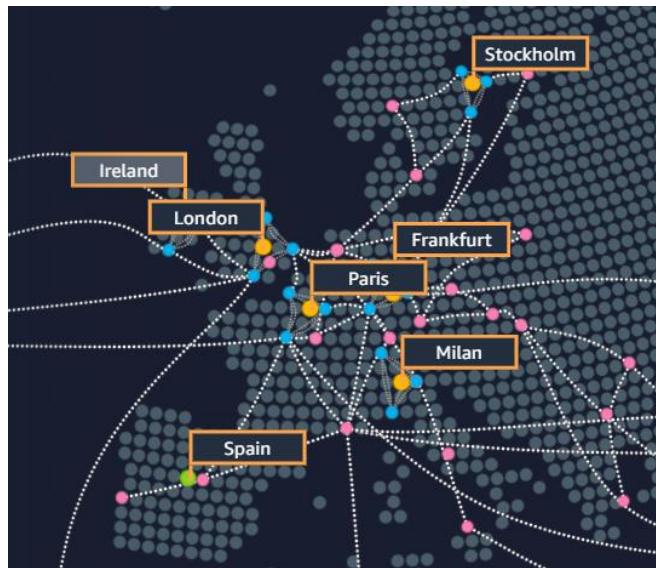
Antes de analizar los recursos informáticos, es necesario conocer dos aspectos muy importantes en AWS como son la región y la zona de disponibilidad.

Una **Región** consiste en áreas geográficas distintas y totalmente independientes en las cuales se agrupan los centros de datos. Antes de utilizar cualquier servicio de AWS es necesario elegir en qué región se aprovisionarán. ¿Cómo saber qué región es la más adecuada para mis recursos? Para contestar a esta pregunta hay que basarse en dos premisas:

- **Servicios disponibles.** No en todas las regiones se disponen de todos los servicios que AWS ofrece. Es posible encontrar regiones en las que muchos de ellos no puedan utilizarse. Por este motivo antes de decantarse por una región conviene conocer cuáles hay disponibles y si cubre las necesidades el proyecto en el que se trabaje. Por ejemplo, las regiones que más servicios disponen son: N. Virginia (us-east-1) e Irlanda (eu-west-1).
- **Proximidad.** La latencia, es uno de los mayores problemas al elegir una región, por eso hay que intentar elegir la que más cerca esté de nuestra situación geográfica.

Por ejemplo, si se desea alojar una aplicación en AWS que tendrá la mayor parte de sus accesos desde España, se elegirá una región lo más próxima al país y que permita desplegar los recursos necesarios para que funcione correctamente.

A través de la URL: <https://www.infrastructure.aws/> es posible conocer cada una de las regiones que dispone AWS. En la siguiente imagen se puede observar todas las que se encuentran disponibles en Europa.

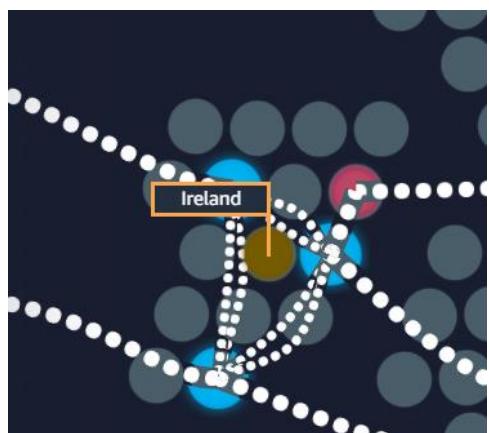


Cuando se trabaja en AWS es muy importante ser consciente de la región en la que se están desplegando los recursos. Posiblemente un día entres a la consola de administración y no encuentres tus aplicaciones. No te preocupes, no han desaparecido. Es muy probable que hayas entrado en una región distinta de dónde fueron desplegados y por eso no los veas.

Dentro de las regiones se encuentran las **Zonas de disponibilidad** que son cada uno de los grupos de centro de datos lógicos que pertenecen a una región y se suelen conocer por AZ (Availability Zone). La documentación oficial de AWS sobre AZ indica: “Cada AZ tiene alimentación, refrigeración y seguridad física independiente y está conectada a través de redes redundantes de latencia ultrabaja”. Las regiones pueden disponer de 1 o varias zonas de disponibilidad.

Una buena práctica consiste en crear aplicaciones que se ejecuten en múltiples AZ dentro de una misma región. De esta forma se consigue alta disponibilidad y mayor tolerancia a fallos, ya que los datos se encuentran replicados en distintos datacenters.

Por ejemplo, en la región de Irlanda (eu-west-1) hay disponibles tres AZ ubicadas en distintos puntos del país. En la siguiente imagen se pueden observar las AZ de Irlanda mediante los círculos azules.



RED VPC

Una vez se ha entendido los conceptos de región y AZ es el momento de conocer los recursos informáticos que están disponible en AWS. Se analizarán de forma ordenada teniendo en cuenta una dependencia jerárquica. Para entender esto, imagina un centro de datos. ¿Cuál es el primer componente necesario para comenzar a trabajar y desplegar aplicaciones? Si lo primero que se te viene a la mente es el hardware o componentes físicos es porque aún no has asimilado el concepto del *cloud computing* y sus ventajas, para ti como usuario. Cuando se trabaja con un proveedor en la nube hay que olvidarse de la infraestructura física. Así que, volviendo a la pregunta anterior, el primer componente sería la Red. En AWS el servicio que administra las redes se llama VPC o Red virtual privada en la nube.

VPC: es un servicio que permite la creación de una red privada virtual aislada de forma lógica. Ofrece la capacidad de utilizar rangos de direcciones IP propios, crear subredes y la configuración de tablas de ruteo y puertas de enlace de red.

AWS permite la interconexión entre redes conocido como VPC peering que consiste en direccionar el tráfico entre ellas de forma privada. De esta forma es posible disponer aplicaciones de diferentes sedes, repartidas por el mundo, conectadas de forma privadas.

Por defecto, AWS dispone de una VPC en cada una de las regiones. Ésta tiene una configuración sencilla con varias *subnets* creadas (una por cada AZ), servicio de DHCP activo y permite todo el tráfico de entrada y salida en cada una de las subredes. Si comienzas a trabajar con AWS y quieras hacer pruebas con esta VPC tienes suficiente.

Cuando se trabaja con una red de producción es conveniente diseñar la VPC correctamente. Por ejemplo, crear una subnet pública en cada una de las AZ para alojar servicios de tipo frontend y crear otras de forma privada, sin acceso al exterior, para el backend, al igual que si estuvieses trabajando en un datacenter.

Un componente muy importante que se encuentra dentro de una VPC son los **Security Groups** o **grupos de seguridad**. Éstos son cortafuegos virtuales que permiten controlar el tráfico entrante y saliente de ciertos servicios de AWS mediante reglas ACL. Su configuración es bien sencilla, consiste en especificar el puerto y la dirección IP (IPV4 o IPV6) para permitir o bloquear el tráfico. También es posible crear reglas para controlar el tráfico de red de otro grupo de seguridad distinto. De esta forma todos los recursos que se encuentren en dicho grupo heredarán la regla.

INSTANCIAS EC2 – MÁQUINAS VIRTUALES

En AWS, las máquinas virtuales se llaman **instancias EC2** (Elastic compute cloud). El concepto de *elastic* viene de las continuas necesidades de cambio para crecer, disminuir o aumentar recursos y/o servicios. AWS proporciona a las instancias EC2 flexibilidad para elegir la combinación de recursos y adaptarlas a las necesidades del usuario.

Las instancias EC2 están catalogadas por tipos. Si estás acostumbrado a trabajar con máquinas virtuales físicas es muy probable que alguna vez hayas desplegado alguna en un clúster diferente debido a que su hardware es más potente. AWS pone a disposición de sus usuarios diferentes tipos instancias basadas en el *hardware* dónde corren. Cada uno de ellos está agrupado en función de las cargas de trabajo que se vayan a ejecutar, disponiendo de distintos tamaños para elegir.

Los diferentes tipos de instancias EC2 que se pueden encontrar actualmente son:

- **Uso general:** pensadas para disponer de una combinación equilibrada de recursos informáticos, memoria y de red en función de las cargas.
- **Optimizadas para informática:** utilizan procesadores de alto rendimiento y están pensadas para cargas de trabajos de procesamiento por lotes, codificación de archivos multimedia, servidores web de alto rendimiento, análisis de datos o servidores de videojuegos. En definitiva, aplicaciones con uso intensivo de informática.
- **Optimizadas para memoria:** este tipo de instancias son utilizadas cuando se necesitan un alto rendimiento en cargas de trabajos que procesen gran cantidad de datos en memoria, sistemas de caché y análisis de *big data* en tiempo real.
- **Informática acelerada:** utilizan *hardware* de última generación y se combinan con GPU. Su utilización se centra en el campo científico y computacional. Como por ejemplo el *deep learning*, reconocimiento de voz, etc.
- **Optimizadas para almacenamiento:** están diseñadas para cargas de trabajos que necesitan gran cantidad de acceso de escritura y lectura. Además, optimizadas para ofrecer múltiples operaciones de E/S con bajas latencias por segundo (IOPS). Como, por ejemplo, base de datos NoSQL, Elasticsearch, etc.

Antes de decantarse por un tipo de instancia es recomendable acceder a la documentación de AWS y conocer todas las opciones disponibles, éstas pueden variar en el tiempo ofreciendo diferentes opciones más actualizadas.

La equivalencia al tipo de instancia EC2 y su tamaño en un entorno físico sería el clúster dónde esta sería desplegada, utilizará el hardware de los hipervisores, y el tamaño se ajustaría a la cantidad de vCPU y memoria RAM que se asignará.

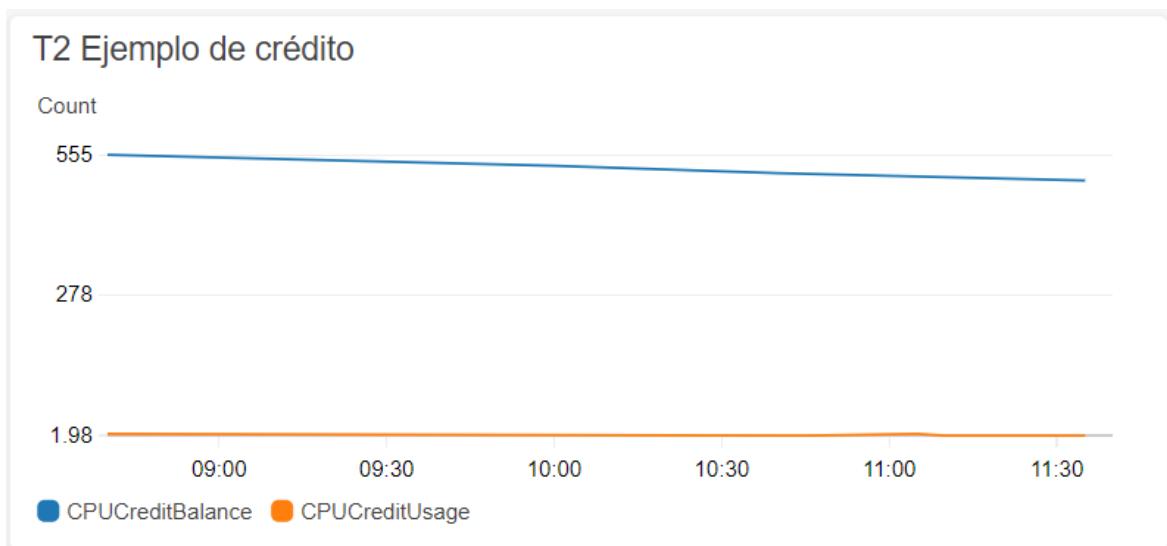
Instancias de tipo T2

Dentro de los tipos de instancias de uso general se encuentran las T2 de desempeño de ráfagas. Su uso está muy extendido ya que entran dentro de la capa gratuita que ofrece AWS, por ello es necesario explicar su funcionamiento. Este tipo de instancias funcionan a ráfagas en cuanto a uso de CPU, se basan en créditos que se van recibiendo por hora, dependiendo del nivel establecido a su tamaño. El crédito es consumido cuando existe un uso de CPU o se va acumulando cuando está inactiva. ¿Qué ocurre si se consume el crédito completo? Pues que se limita el uso de CPU al porcentaje que garantiza su tamaño.

Por ejemplo, una instancia t2.small recibe 12 créditos de CPU por hora. Esta capacidad proporciona un desempeño base equivalente al 20% del núcleo de una CPU. Si en algún momento la instancia no necesita los créditos que recibe, los almacena en el saldo de créditos por un tiempo de hasta 24 horas. Si la t2.small necesita alcanzar ráfagas de más del 20% de un núcleo, extrae esta capacidad del crédito acumulado para gestionar este aumento sin problemas. En caso de quedarse sin crédito únicamente se podrá utilizar el 20% de CPU, hasta que se reciban nuevos. Para entender el ejemplo mejor, pensad en las tarifas móviles e internet. Éstas te ofrecen XGb para navegar y cuando los consumes te bajan la velocidad. Puedes seguir navegando, pero a una velocidad reducida pues igualmente ocurre con las T2. Este tipo de instancias son muy baratas y se suelen utilizar para desarrollo o repositorio que no necesiten un alto uso de CPU. Existe la posibilidad de utilizarlas sin límites, pero ya se encuentra la nueva generación T3 y son más recomendable.

No es recomendable usar en producción una instancia de tipo T2, a menos que sepáis que el uso será mínimo, ya que os podéis encontrar con problemas de rendimientos.

Para detectar problemas de rendimiento en las instancias T2 es recomendable crear una gráfica en el servicio de **Cloudwatch** que combinen las métricas **CPUCreditBalance** y **CPUCreditUsage** tal y como se puede observar en la siguiente imagen. Si observas que el uso de CPU de la instancia no es muy alto, debes de revisar esta gráfica para comprobar si se ha consumido el crédito disponible.



Instancias con procesadores AWS Graviton basados en Arm

AWS ha creado una serie de procesadores con núcleos Arm Neoverse de 64 bits que están disponibles para las instancias EC2. Ofrecen una gran relación de precio y rendimiento. En algunos casos superan a las instancias con procesadores Intel o AMD. Son compatibles con las distribuciones de Linux más populares que puedes usar en una EC2 (Amazon Linux, Red Hat, SUSE y Ubuntu). Existen dos generaciones:

- Los de **primera generación** se encuentran dentro de las instancias de uso general y se identifican por el tipo A1. Son recomendable para servidores web, microservicios en contenedores, que utilicen cargas de trabajo que pueden ejecutarse en pequeños núcleos.
- Los de **segunda generación** ofrecen una relación entre precio y rendimiento pudiendo llegar hasta un 40 % superior al de instancias comparables basadas en x86 y pueden utilizar almacenamiento SSD basadas en NVMe local. Cuentan con cifrado DRAM de 256 bits siempre activo y un rendimiento de cifrado por núcleo un 50% más rápido en comparación con los procesadores AWS Graviton de primera generación. Se encuentran dentro de los tipos:
 - Uso general, identificados como **M6gd** y los nuevos **T4g**.
 - Optimizados para informática, identificados como **C6gd**.
 - Optimizados para memoria identificados como **R6gd**.

AWS ha anunciado recientemente las nuevas instancias **EC2 T4g** para uso general que utilizan la segunda generación de procesadores Graviton, permitiendo aumentar el uso de la CPU en cualquier momento durante el tiempo que sea necesario. Además, puedes probarlas de forma gratuita ya que las t4g.micro entran dentro del *Free Tier* de AWS.

Revisa tu región para comprobar la disponibilidad de instancias EC2 basadas en los procesadores AWS Graviton.

Las instancias con este tipo de procesadores pueden ser utilizadas en los distintos servicios de AWS como, por ejemplo: Amazon ECS, Amazon EKS, Amazon ECR, etc.

Si estás interesado en usar este tipo de instancia es recomendable oír la Guía de introducción sobre el uso de procesadores AWS Graviton basados en Arm que puedes encontrar en el repositorio oficial: <https://github.com/aws/aws-graviton-getting-started>

COMPONENTES DE UNA INSTANCIA EC2

Para crear una instancia EC2 es necesario configurar una serie de componentes que harán que ésta funcione correctamente. Éstos se pueden clasificar de forma general en los siguientes grupos:

- Imágenes – Amazon Machine Image
- Red y seguridad
- Almacenamiento
- Etiquetas

Imágenes – Amazon Machine Image

Toda máquina virtual utiliza un sistema operativo y con las instancias EC2 no iba a ser menos. Pero a diferencia de un entorno físico en el que tienes que subir la ISO e instalarla, AWS dispone de imágenes que permiten ser desplegadas en un volumen de la instancia en cuestión de segundos. Este proceso ofrece la ventaja de ahorrar la instalación inicial del sistema operativo.

El servicio se llama Amazon Machine Image (AMI) y existen cuatro tipos de opciones:

- **Quick Start o generales:** son un listado de AMIs que utilizan los sistemas operativos más usados como, por ejemplo: Amazon Linux, Ubuntu, Suse, Red Hat, Windows Server, etc. Hay diferentes versiones del sistema para elegir.
- **My AMIs:** son imágenes personalizadas por el usuario. Con una EC2 en funcionamiento es posible crear una AMI para ser desplegada posteriormente. Es el equivalente a los *templates* en el sistema físico.
- **AWS Marketplace:** es un *marketplace* dónde se pueden encontrar AMIs de diferentes fabricantes. Hay desde sistemas operativos generales, pero con versiones diferentes a las que se pueden elegir en la opción *Quick Start* o imágenes de fabricantes que ofrecen SaaS. A la hora de elegir una de este tipo hay que revisar si éstas llevan cargos adicionales por el uso del software.
- **Imágenes de la comunidad:** Las AMIs personalizadas pueden ser compartidas entre diferentes usuarios. Existe una comunidad activa en AWS que crea imágenes con software ya preinstalado y configurado, listo para usar. Este tipo de AMIs pueden ser monetizables. Para compartir una de ella es muy importante seguir las recomendaciones que AWS ofrece en su documentación oficial.

Una AMI no es una copia de seguridad de la máquina virtual, al igual que un snapshot tampoco lo es.

Amazon Linux es uno de los sistemas operativos que pueden ser encontrados a la hora de seleccionar una AMI. Es una distribución de Linux, mantenida por AWS. Utiliza paquetería RPM y YUM. Si estáis familiarizado en trabajar con Red Hat, Centos o Fedora, no debéis tener problemas por usarlo. Utiliza un *kernel* optimizado para correr bajo la infraestructura de AWS además de disponer de los agentes y herramientas necesarias para conectarse a diferentes servicios.

La AMI de Amazon Linux 1 finalizará el soporte estándar el 31 de diciembre de 2020 y entrará en una fase de soporte de mantenimiento. AWS recomienda a los clientes actualizar sus aplicaciones para el uso de Amazon Linux 2, que incluye soporte a largo plazo hasta el 2023.

Red y seguridad

Toda instancia EC2 se despliega en una VPC, la red virtual privada. Si piensas en un *datacenter* tradicional sería el equivalente al entorno donde la máquina virtual sería creada. Por ejemplo, uno de producción con sus diferentes direccionamientos para *backend*, *frontend*, *backups*, etc. El direccionamiento se le asigna dependiendo de la *subnet* elegida. Éstas se encuentran asociadas a las zonas de disponibilidad tal y como se comentó anteriormente.

Con la VPC y *subnet* elegida, a la instancia se le asigna por defecto una interfaz de red virtual llamada Elastic Network Interfaces. Se le pueden asignar múltiples IP privadas tanto en IPV4 como IPV6. Además, si es necesario, puedes añadirle más interfaces asignadas a *subnets* diferentes y que existan dentro de la VPC o incluso desactivar las secundarias. Existen límites del número de interfaces y direcciones IP que se pueden añadir a una EC2 y dependen del tipo de instancia. Si crees que necesitas múltiples tarjetas de red virtuales te recomiendo que accedas a la documentación de AWS sobre Elastic network interfaces y revises la tabla que informa de dichos límites.

Cuando se asigna una interfaz de red a una instancia que está en funcionamiento (*hot attach*) o que está parada (*warm attach*), ésta debe ser levantada y configurada. Esto no ocurre si se usa como sistema operativo Amazon Linux y Winodws Server.

No es recomendable asignar varias interfaces de red de una misma subnet a una instancia EC2. Además, no puedes hacer bond para incrementar el ancho de banda uniendo varias interfaces.

También es posible asignarle una IP pública para que sea posible acceder a los servicios publicados que pueda tener la instancia. Existen dos tipos:

- **Auto asignación IP pública:** Cuando se crea una instancia se puede elegir que AWS le auto asigne IP pública. A través de ella se permite el acceso desde internet a dicha instancia. Además, va acompañada de un DNS. El inconveniente de usar este tipo de direccionamiento es que cuando la instancia es reiniciada o parada, ésta se libera y al arrancar de nuevo puede cambiar. Piensa en una red doméstica y lo que ocurre cuando reinicias el *router*.

- **Elastic IP:** Son las que comúnmente se conocen como IP fijas. Se asignan a las *Elastic network interfaces*. Es posible liberar la IP de una interfaz y asignarla a otra de forma sencilla.

AWS factura por las Elastic IP reservadas y no usadas. Mientras está asignada a una interfaz de red virtual y se encuentre en funcionamiento.

AWS permite utilizar rangos de IP públicos que ya tengas asignados, se conoce como **BYOIP (Bring your own IP addresses)** y forma parte de **AWS Global Accelerator**. Es importante leer los requisitos detenidamente ya que no es un servicio apto para todos los públicos.

Por último, relacionado con la red, se encuentran los **grupos de seguridad**. Todas las instancias EC2 tienen uno asignado. O bien es creado por defecto con todo el tráfico de acceso cerrado o es posible asignar uno que previamente se haya configurado. Éstos hacen de cortafuegos y son los encargados de permitir o bloquear el tráfico de red.

Almacenamiento

Las máquinas virtuales necesitan de discos duros virtuales para poder funcionar. Al menos es necesario uno en el que alojar el sistema operativo y los sectores de arranque. En una instancia EC2 el almacenamiento es por bloque y son volúmenes lo que se les añaden. Los EBS (Amazon Elastic Block Store) proporcionan diferentes opciones permitiendo optimizar el rendimiento del almacenamiento y los costes de las cargas de trabajo. Se dividen en dos grupos: Los basados en SSD usados para cargas de trabajo transaccionales como bases de datos y volúmenes de arranque (el rendimiento depende principalmente de las IOPS) y los HDD para cargas de trabajo intensivas como MapReduce y el procesamiento de registros (el rendimiento depende principalmente de los MB/s). Existen cuatro tipos:

- **SSD de IOPS provisionadas de EBS (io1):** son volúmenes de grandes rendimientos y están pensados para cargas de trabajo transaccionales sensibles a la latencia como por ejemplo base de datos que tengan un uso intensivo de operaciones de E/S.
- **SSD de uso general (gp2) de EBS:** son los volúmenes generales y que están seleccionados por defecto a la hora de desplegar una instancia. Tienen un equilibrio entre el coste y el rendimiento. Suelen utilizarse en servidores de aplicaciones, desarrollo, etc.
- **HDD optimizados para procesamiento (st1):** son volúmenes de bajo costes diseñados para utilizarse con cargas de trabajos de procesamiento intensivo y con accesos frecuentes. Su uso más común es para el Big data, aunque también se utilizan para almacenes de datos o procesamiento de registros.
- **HDD fríos (sc1):** Son los volúmenes más baratos pensados para cargas de trabajos con accesos menos frecuentes. La información almacenada en estos volúmenes es comúnmente conocida como datos en frío.

Existe un tipo de volumen llamado **Magnético** que son de generación anterior y están en desuso. Se solían utilizar para cargas de trabajo en las que el acceso a los datos es infrecuente. Aunque es posible usarlos, no son recomendables.

La partición raíz de una instancia EC2 solo puede ser de tipo SSD y el nivel de IOPS dependerá de si es gp2 o st1. Al igual que con todos los recursos existen límites de tamaño, en el momento de diseñar la instancia EC2 consulta la guía de volúmenes EBS en la documentación oficial de AWS.

Todos pueden ser **criptados** o bien utilizando una clave por defecto o bien creando claves a través del servicio **AWS Key Management Service**. Además, todos los volúmenes secundarios pueden ser desacoplados de una EC2 y añadidos a otra. Se pueden ampliar mediante un clic y dependiendo del sistema operativo y la generación de la instancia EC2 se podrá visualizar los cambios en caliente. También es posible ampliar la cantidad de IOPS o cambiar el tipo de volumen.

Si se utilizan instancias EC2 de generaciones antiguas o el volumen a ampliar es la partición raíz siempre se debe de reiniciar la instancia EC2 para que los cambios se apliquen.

Otra opción que ofrece AWS sobre los EBS es la capacidad de realizar **snapshots** de los mismos. Éstos son almacenados en S3, aunque no están visibles en la consola de S3. Permiten ser compartidos entre cuentas de AWS, aunque existen ciertas limitaciones. Pueden lanzarse de forma manual o bien de forma automatizada a través del **Administrador del ciclo de vida (Data Lifecycle Manager)** que permite configurar ventanas para que se hagan de forma automática.

Es recomendable disponer de copias de seguridad en diferentes regiones. A través del Administrador del ciclo de vida se puede configurar que se guarde una copia del snapshot en otra región diferente.

No existen cargos adicionales por utilizar esta herramienta ni por la realización de **snapshots**. Únicamente pagas por la cantidad de almacenamiento utilizado. Es decir, el volumen y la ocupación de éstos.

Etiquetas

Las etiquetas, aunque no es un componente informático, es una utilidad muy recomendable de usar y que está disponible en casi todos los servicios de AWS. Funcionan como clave valor y permite inventariar los servicios de una forma muy sencilla. Al crear una EC2 te solicita añadir etiquetas, no son obligatorias, pero si recomendables.

AWS dispone de un apartado para administrarlas, desde las que permiten realizar búsquedas para encontrar servicios de forma sencilla. Se podría considerar como una CMDB básica que permite inventariar los recursos desplegados en AWS.

Una buena práctica a la hora de desplegar instancias EC2 es utilizar etiquetas para identificarlas. Por ejemplo, incluir el entorno si es producción o desarrollo, el cliente al que pertenece, la persona que la despliega, si perteneces a un grupo de aplicaciones, etc.

Existen múltiples usos para las etiquetas. Por ejemplo, CodeDeploy permite realizar despliegues basándose en éstas. Es posible programar *snapshots* de volúmenes EBS basados en ella. También es posible explorar costes de forma personalizada desde el panel de exploración.

Ahora que ya conoces los recursos de cómputo más importante que puedes utilizar en AWS, es conveniente que existen límites establecidos por la propia compañía. Los recursos, aunque se encuentren en diferentes *datacenters* repartidos por el planeta, son finitos. De ahí que AWS establezca diferentes límites por servicios. Dentro de la consola de EC2 tienes un apartado llamado Límites en el que puedes consultar cada uno de ellos. Existe la posibilidad de ampliar alguno de ellos contactando con AWS y exponiéndoles tu caso en particular.

COMPUTO EN MICROSOFT AZURE

Azure es el proveedor de *cloud* público de Microsoft. Ofrece una serie de servicios y recursos que permiten a usuarios y/o empresas desplegar aplicaciones de forma muy sencilla. Además, ofrece una gran integración con los productos de Microsoft. Empresas que ya disponen de controladores de dominio, directorio activo, etc. Pueden expandir su infraestructura rápidamente a la nube creando un *cloud* híbrido entre Azure y su propio centro de datos de forma segura. El concepto de elasticidad también se encuentra presente en Azure y permite añadir, quitar, o modificar casi cualquier componente de una máquina virtual.

Azure, a diferencia de AWS, ofrece una interfaz más sencilla para trabajar con máquinas virtuales. Utiliza un asistente, paso a paso, como los que Microsoft está acostumbrado a incluir en sus productos. En este apartado se seguirá el asistente como guía, pero utilizará la misma estructura que el anterior. Se explicarán los principales componentes que son necesarios para que una máquina virtual pueda funcionar como si estuviese en un centro de datos físico.

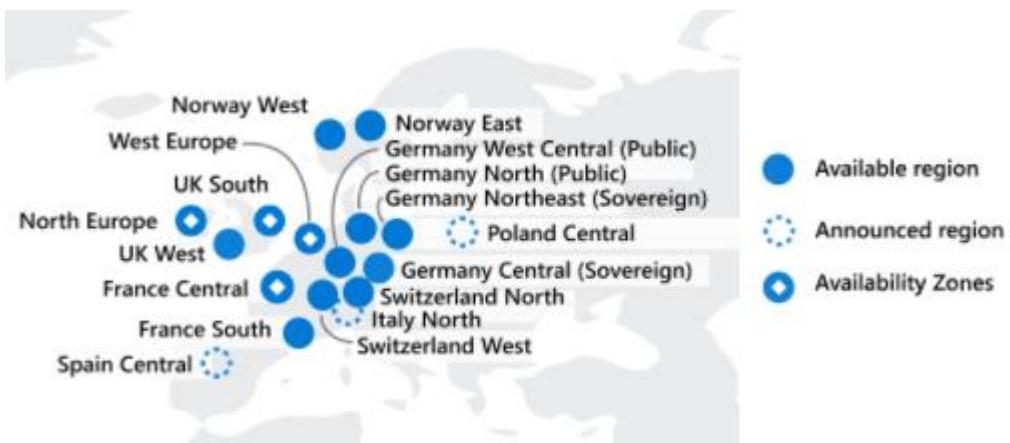
REGIÓN, ZONA DE DISPONIBILIDAD Y CONJUNTO DE DISPONIBILIDAD

Azure también utiliza regiones y zonas de disponibilidad. Es algo muy común en cualquier proveedor público que ofrezca sus servicios a nivel mundial. Es cuestión de disponibilidad y de latencias.

Las regiones están divididas por áreas geográficas distintas y totalmente independientes. A diferencia con AWS, ésta es elegida a la hora de desplegar la máquina virtual, aunque también se puede configurar una por defecto. Antes de decantarse por una región es muy importante tener en cuenta lo siguiente:

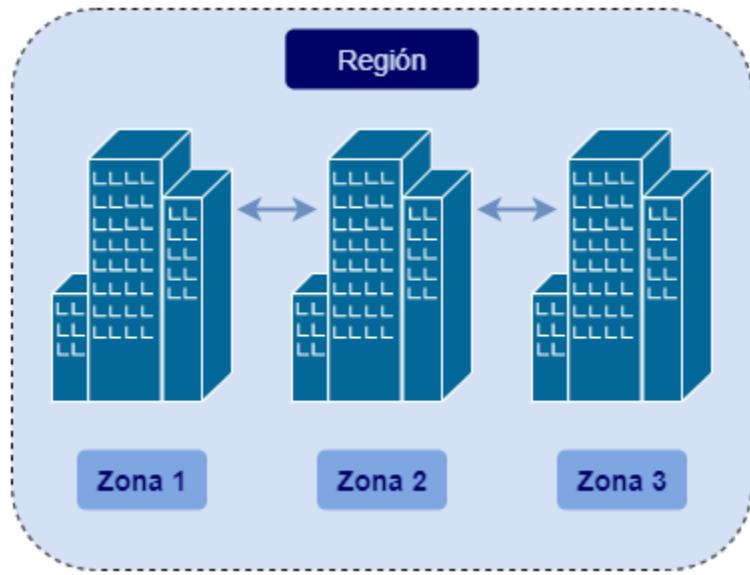
- **Latencia.** Si quieras publicar una aplicación o conectar tu empresa a los servicios de Azure es conveniente trabajar en una región donde se encuentren tus usuarios.
- **Disponibilidad de los servicios.** No en todas las regiones se encuentran todos los servicios de que Azure oferta. Por este motivo es necesario revisarlos antes.
- **Cumplimiento normativo y residencia de los datos.** Si el proyecto en el que estás trabajando necesita cumplir una serie de normativas y que los datos residan en país concreto, asegúrate de elegir bien la región.
- **Precios.** Existen precios distintos para los servicios que oferta Azure dependiendo de la región. Si ninguna de las dos

En la siguiente imagen se pueden observar las regiones, zonas de disponibilidad y próximas regiones que dispone Azure en Europa.



Para consultar el listado de regiones disponibles, sus características o las próximas en abrir puedes hacerlo desde la URL: <https://azure.microsoft.com/es-es/global-infrastructure/geographies/>

Dentro de éstas se encuentran las **zonas de disponibilidad** que son ubicaciones aisladas que proporcionan alimentación, refrigeración y funcionalidad de red redundantes. De esta forma es posible ejecutar aplicaciones que necesiten alta disponibilidad y tolerancia a fallos. Si te fijas, estos conceptos sin iguales que los vistos en el apartado anterior en AWS.



También es posible desplegar máquinas virtuales sin la importancia de elegir la zona de disponibilidad. Esta opción no es recomendable para producción. Utilícela para desarrollo o repositorios.

Azure va un paso por delante en cuanto a tolerancia a fallo y añade el concepto de **conjuntos de disponibilidad** que permite realizar configuraciones para proporcionar redundancia y disponibilidad de máquina virtual. De esta forma se garantiza que durante mantenimientos programados o posibles fallos hay al menos una máquina virtual disponible. Se componen de dos tipos de dominios:

- **Dominios de error:** contiene las máquinas virtuales que comparten fuente de alimentación y conmutador de red físico.
- **Dominios de actualización:** máquinas que se reinician juntas durante mantenimientos programados. Azure nunca reinicia más de un dominio de actualización a la vez.

Es recomendable agrupar dos o más máquinas virtuales a un conjunto de disponibilidad para asegurar que al menos una se encuentra disponible en caso de mantenimiento programado o algún fallo en el conmutador de red o alimentación.

GRUPO DE RECURSOS

Todo recurso en la nube necesita permisos para poder acceder a distintos servicios. Un grupo de recursos no es más que es una colección de éstos que comparten los mismos permisos, ciclo de vida y directivas. Cuando se desea crear una máquina virtual en Azure es necesario definir uno y que todos los recursos que se utilicen estén dentro de este.

Disponer de todos los componentes bajo un grupo de recursos permite administrarlos de forma conjunta. Funciona en cascada, por ejemplo, si se desea eliminar el grupo de recursos,

se eliminaría también todo lo que estuviese asociado a él. También es posible realizar otras acciones o incluso migrar a un nuevo grupo.

Utiliza los grupos de recursos para crear alarmas de supervisión y de costes para tener siempre una visión global de tu infraestructura.

REDES VIRTUALES

Para trabajar con máquinas virtuales, y sobre todo en la nube, es preciso seleccionar una **red virtual**. Esta, al igual que el resto de los recursos, depende de la región en la que se vaya a trabajar. En Azure se pueden crear redes virtuales privadas utilizando topologías de red sofisticadas, permitiendo el uso de optimizadores de WAN, equilibradores de carga, firewall de aplicaciones (WAF), etc. También es posible definir flujos de tráfico para conseguir una red con un mayor grado de control.

Dentro de la configuración de la red virtual puedes definir los intervalos de direcciones IP, subredes, tablas de ruta, puertas de enlace y configuración de seguridad. Si estás acostumbrado a diseñarlas en tu centro de datos local te será muy sencillo hacerlo en la nube. Una característica muy importante de la Red virtual de Azure es la facilidad para conectarse con un centro de datos o bien utilizando VPN Ipsec o el servicio **Azure ExpressRoute** que utiliza una red privada.

Con Azure es muy sencillo crear aplicaciones híbridas en las que el frontend de la aplicación se encuentre en la nube y el backend o base de datos estén en un centro de datos local.

Dentro de las redes virtuales se encuentran los grupos de seguridad que contienen reglas de listas de control de acceso (ACL) que permiten o bloquen el tráfico de red entrante o saliente de máquinas virtuales. Son de dos tipos:

- **Grupos de seguridad a nivel de interfaz de red.** Son asociados a una NIC y controlan el tráfico entrante y saliente de dicha interfaz.
- **Grupos de seguridad a nivel de subredes.** Se crean bajo una subred y toda NIC de máquina virtual, que esté asociado a ella, heredará las reglas de seguridad. Son los más recomendados de utilizar ya que facilita la administración de las reglas.

Durante la configuración de seguridad de la red es posible activar una serie de características que la hacen más robusta como son:

- **BastionHost.** Esta opción permite usar el servicio **Azure BastionHost** para conectarse a una máquina virtual mediante RDP o SSH de forma segura e ininterrumpida mediante SSL. Durante este tipo de conexiones la máquina virtual no necesita IP pública.

- **Protección contra DDoS.** Con esta opción se habilita la protección contra ataques de denegación de servicios de manera estándar.
- **Firewall.** Habilita un cortafuegos que protege los recursos de la red virtual.

Por último, a nivel de red virtual existe un servicio, aunque aún no está disponible en todas las regiones, que es **Azure Virtual Network TAP**. Consiste en un punto de acceso terminal que transmite continuamente el tráfico de red de una máquina virtual a una herramienta de terceros que permita el análisis de red. Existen en el mercado diferentes soluciones para el análisis de seguridad, administración del rendimiento de red o de una aplicación como, por ejemplo: Awake Security, Cisco Stealthwatch Cloud, Flowmon, vSTREAM de Netscout, etc.

MÁQUINAS VIRTUALES

Tal y como se comentó al principio, Azure apuesta por la sencillez y las máquinas virtuales se llaman como tal. Son creadas a través de un sencillo asistente ofreciendo agilidad. Éstas pueden escalar de forma sencilla, además de poder añadir componentes extras una vez desplegadas. Ofrecen muy buena compatibilidad con el sistema operativo de Microsoft Windows Server, aunque también es posible utilizar sistemas Linux.

Éstas se encuentran clasificadas en tamaños dependiendo de las necesidades para ejecutar aplicaciones y cargas de trabajo. Estos tamaños corren en *hardware* distinto y son los siguientes:

- **Uso general:** estas máquinas disponen de un uso equilibrado de CPU en proporción de memoria. Están pensadas para cargas de trabajos medias y bajas como, por ejemplo: para desarrollo y pruebas, bases de datos pequeñas o medianas, y servidores web de tráfico bajo o medio.
- **Proceso optimizado:** este tamaño es ideal para cargas de trabajo que tienen un uso elevado de CPU en proporción de memoria. Se utilizan para servidores web de tráfico medio, aplicaciones de red, procesos por lotes y servidores de aplicaciones.
- **Memoria optimizada:** se centra en un uso alto de memoria en proporción de CPU. Este tipo de máquinas virtuales son excelentes para bases de datos relacionales, memorias caché de capacidad media o grande y análisis en memoria.
- **Almacenamiento optimizado:** están optimizadas para un alto rendimiento de disco y operaciones E/S. Son utilizadas en servidores que usan macrodatos, bases de datos SQL y NoSQL, almacenamiento de datos y bases de datos transaccionales grandes.
- **GPU:** se basan en el uso de GPU y diseñadas para la representación de gráficos pesados y la edición de vídeo, así como para el entrenamiento e inferencia de modelos (ND) con aprendizaje profundo. Están disponibles con una o varias GPU.
- **Procesos de alto rendimiento:** estas máquinas virtuales utilizan las CPU más rápidas de Azure. Además, tienen la posibilidad de utilizar interfaces de red de alto rendimiento (RDMA).

Si has leído el capítulo anterior podrás ver que tanto Azure como AWS, agrupan el tamaño o tipo de las máquinas virtuales de forma similar.

Hay que tener en cuenta que no todos los tamaños están habilitados en todas las regiones. Antes de diseñar la arquitectura de una aplicación en Azure, comprueba que tamaños puedes elegir.

Azure dispone de un servicio muy útil llamado **VM Scale Sets** que permite crear o escalar miles de máquinas virtuales idénticas en cuestión de minutos. Puede utilizar las plantillas de **Azure Resource Manager** para su creación. Se integra con **Azure Insights Autoscale** para conseguir un escalado automático y real. Este servicio se basa a reglas que desencadenan acciones permitiendo agregar o disminuir recursos.

COMPONENTES DE UNA MÁQUINA VIRTUAL

Durante el asistente de creación de una máquina virtual en Azure te encontrarás paso a paso con cada uno de los componentes que harán que ésta entre en funcionamiento, además de otras funciones avanzadas. En este apartado se van a clasificar en los siguientes grupos:

- Imágenes
- Discos
- Red y seguridad
- Etiquetas

Imágenes

Las imágenes contienen los sistemas operativos base que hay disponibles para las máquinas virtuales. Los que Azure ofrece de forma predeterminada van desde las últimas versiones que Microsoft dispone hasta diferentes distribuciones de Linux más populares (Red Hat, Suse, Centos, Ubuntu, Debian o Oracle Linux). Éstas están disponibles en todas las regiones independientemente del tamaño de la máquina virtual.

También es posible instalar imágenes con sistemas operativos o aplicaciones preparadas por terceros desde el Azure Marketplace. Aquí puedes encontrar un sinfín de software listo para funcionar. También puedes subir imágenes personalizadas, con una versión específica del sistema operativo, algún parche aplicado o algún software preinstalado. Es muy importante seguir la guía paso a paso que ofrece Azure para publicar imágenes.

Antes de desplegar una máquina virtual utilizando una imagen del Marketplace es importante leer todas las condiciones de servicio ya que muchas de ellas necesitan licencias de uso.

El servicio Shared Image Galleries es el encargado de organizar las imágenes creadas. Éstas se pueden compartir con usuarios o grupos de *Active Directory* dentro de la organización. Ofrece la posibilidad de replicación global y control de versiones de las imágenes, entre otras características. Se podría decir que es el equivalente a los templetes de máquinas virtuales que estás acostumbrado a utilizar en un entorno físico, pero utilizando una interfaz de administración para gestionarlas.

Discos

El siguiente paso para la creación de la máquina virtual es añadir discos para el almacenamiento. Éstos se encuentran dentro del servicio Azure Storage. En un entorno físico, dependes de los sistemas de almacenamiento que dispones ya que estos son finitos. En el proveedor *cloud* esto no es así. Por ejemplo, el número de discos que se pueden añadir a una máquina virtual en Azure depende del tamaño de ésta y la máxima capacidad de un disco está entre los 64 TB para los discos ultra o los 32TB para el resto de los tipos de discos.

Éstos se clasifican en:

- **Ultra disk:** son discos de tipo SSD preparados para un uso intensivo de operaciones de entradas y salidas. Están pensados para ser utilizados para bases de datos con gran cantidad de transacciones como Oracle, SQL Server o SAP Hana. Con este tipo de discos se pueden alcanzar un máximo de 160.000 IOPS.
- **Premium SSD:** Estos discos están pensados para servidores de producción que tengan una carga media- alta ya que ofrecen un gran rendimiento.
- **Standard SSD:** Son los discos SSD de menor rendimiento y su objetivo son los servidores con poca carga, como por ejemplo para desarrollo o *testing*.
- **Standard HDD:** Estos discos solo están recomendados para copias de seguridad, servidores no críticos o almacenamiento con accesos poco frecuentes.

Como mínimo se debe elegir un disco para la máquina virtual que contiene el sistema operativo, pero también es posible añadir más, dependiendo del tamaño. El tipo de disco puede ser cambiado posteriormente, dependiendo de las necesidades. Además, ofrecen un cifrado en reposo, pudiendo utilizar una clave administrada por la plataforma o bien una proporcionada por el usuario. Dentro de las opciones de configuración.

Otra característica del almacenamiento es la posibilidad utilizar discos efímeros. Estos discos son independientes de Azure Storage y se crean en la máquina virtual. Se usan para cargas de trabajo sin estado, donde las aplicaciones toleran errores de máquinas virtuales individuales, pero necesitan un despliegue de sistema operativo o restablecimiento de ésta de forma rápida.

Red y seguridad

Anteriormente se ha explicado el funcionamiento de las redes virtuales en Azure. Éstas son redes aisladas del resto. Cuando se habla de redes, no pienses en una vLAN o *subnet*, sino en una red completa, con sus distintas subredes, direccionamiento, reglas de seguridad, etc. Por defecto, todas las máquinas virtuales dentro de una red virtual se pueden comunicar entre sí.

Recuerda que éstas se crean dentro de una región. Durante el asistente de despliegue de la máquina virtual debes elegir la misma que la red a utilizar, para que se muestre. En caso de que no dispongas de una, puedes crearla en el mismo paso. En función de la red seleccionada, se mostrarán las subredes que están disponibles. Durante este proceso se añadirá una interfaz de red y se le asignará una dirección IP privada.

Para que la máquina virtual tenga conectividad con el exterior es muy importante asignarle una IP pública, ésta puede ser estática o dinámica. Azure cobra por el uso de IP estática. Úsalas solo si es necesario.

A las interfaces de red se le pueden añadir un grupo de seguridad que permita controlar el tráfico entrante y saliente. Éstos se pueden asociar justo en el momento de configuración del despliegue o posteriormente, editando los recursos de red de la máquina. Durante el asistente solo se permite añadir la interfaz de red principal. Si deseas añadir nuevas NICs debes hacerlo desde la consola de administración de la máquina virtual.

Etiquetas

Aunque las etiquetas no son un componente de la máquina virtual para su funcionamiento, es muy importantes usarlas. Gracias a éstas es posible identificar rápidamente a qué aplicación pertenece, quien es el propietario o para qué se ha levantado. Es importante en tu organización establecer reglas a la hora de crear las etiquetas para que exista homogeneidad.

Al crearlas, puedes observar que éstas son asignadas a los recursos que ya se han ido configurando durante el asistente. Esto ayuda a identificar rápidamente todo aquello que se encuentre asociado a la máquina virtual como, por ejemplo: Discos, dirección IP pública, interfaz de red, etc.

COMPUTO EN GOOGLE CLOUD PLATFORM

Google también dispone de su propio *cloud* público: **Google Cloud Platform (GCP)**. Ofrece distintos recursos para poder publicar aplicaciones o cargas de trabajo en la nube de forma sencilla y segura. Una de las características diferenciales de GCP con respecto a AWS y Azure es su facilidad para migrar aplicaciones desde servidores físicos, VMware vSphere, Instancias EC2 de AWS o máquinas virtuales de Azure. Si estás acostumbrado a trabajar con VMware o Nutanix debes saber que puedes conectar de forma sencilla tu centro de dato con la plataforma de Google, extendiendo el entorno físico a la nube.

Al igual que en los apartados anteriores, se describirán cada uno de los componentes básicos que GCP ofrece como **Compute Engine** para trabajar con recursos informáticos.

REGIÓN Y ZONA DE DISPONIBILIDAD

Es repetitivo, si has leído los apartados anteriores, volver a hablar de región y zona de disponibilidad. En GCP se trabaja de la misma forma que en AWS y Azure. Se disponen de diferentes regiones repartidas por todo el mundo en la que es posible alojar recursos. Por cada región es posible encontrar tres o más zonas de disponibilidad.

La principal diferencia que existe con el resto de los proveedores es que hay disponibles recursos zonales y regionales:

- **Recursos regionales:** son recursos que dependen de la región como por ejemplo las direcciones IP externas estáticas o subredes.
- **Recursos zonales:** estos recursos dependen de la zona de disponibilidad. Las máquinas virtuales o los discos persistentes zonales son un ejemplo de este tipo de recursos.

Disponer de recursos en diferentes zonas permite aislarlos de fallos de la infraestructura física. Tenerlos en distintas regiones aporta un mayor grado de independencia a estos. Existe algunos recursos que son globales, independientemente de la región y la zona, como, por ejemplo: imágenes, firewall, red VPC, etc.

En la siguiente imagen se pueden observar las distintas regiones disponibles actualmente en GCP.



RED VPC

La red privada virtual (VPC) de GCP ofrece herramientas de redes globales, escalables y flexibles para las máquinas virtuales de Compute Engine, los contenedores de Google Kubernetes Engine (GKE) y App Engine.

Si has estado atento en el punto anterior, te habrás dado cuenta de que una VPC en GCP no está vinculada a una región como si ocurre con el resto de los proveedores. Éstas son globales, pudiendo abarcar varias regiones sin la necesidad de comunicarse vía internet pública. Sin embargo, las subredes, dentro de éstas, si son regionales. Es posible establecer una conexión segura entre tu red existente y la red de VPC mediante una **VPN** utilizando Ipsec. Se pueden conectar dos VPC de forma privada usando el **Emparejamiento de redes VPC**.

En cuanto a seguridad permite segmentar redes con un **Firewall** de distribución global para restringir el acceso a las máquinas virtuales. Además, dispone de un registro de reglas (ACL) te permite inspeccionar, verificar y analizar los efectos de tus reglas de *firewall*. Éstas son aplicadas a nivel de proyecto o de red determinada, y las conexiones se permiten o deniegan por máquina virtual.

Otra característica importante que dispone la VPC es la **Duplicación de paquetes** que permite clonar el tráfico (TCP, UDP y ICMP) de las máquinas virtuales. Se captura todo el tráfico de entrada y salida, además de los datos de paquetes, como las cargas útiles y los encabezados para su posterior análisis. El objetivo de usar este servicio es para detectar amenazas, anomalías o incluso problemas de rendimiento de las aplicaciones.

COMPUTO ENGINE – MÁQUINAS VIRTUALES

Compute Engine es el nombre que Google le ha brindado a su servicio de máquinas virtuales. Éstas se configuran de forma sencilla y soportan los siguientes sistemas operativos: CentOS, CoreOS, Debian, OpenSUSE, Red Hat, SLES, Ubuntu y Windows. También puedes usar tu propia variante de Linux si lo deseas.

Al igual que los proveedores analizados anteriormente GCP ofrece tres familias de máquinas virtuales:

- **Uso general:** son las que ofrecen la mejor relación entre precio y rendimiento para diferentes cargas de trabajo. Suelen ser utilizadas para servidores web, bases de datos, aplicaciones administrativas, servidores de caché, microservicios, entornos de desarrollo, etc.
- **Con memoria optimizada:** son utilizadas en máquinas virtuales que lanzan cargas de trabajo y tienen un uso intensivo de memoria. Están pensadas para bases de datos grande en memoria como SAP HANA, o cargas de trabajo que necesiten mucho análisis de memoria.
- **Optimizada para la computación:** son máquinas que necesitan un rendimiento ultraalto para cargas de trabajo de procesamiento intensivo como por ejemplo videojuegos, HPC, etc.

Una característica diferenciadora con el resto de los proveedores se puede encontrar en las máquinas virtuales de uso general. GCP permite al usuario **seleccionar el tipo de CPU a utilizar: ADM o Intel**. Además, todas las máquinas virtuales de esta familia pueden ser personalizables, es decir, puedes seleccionar la cantidad de vCPU y memoria RAM que más se ajuste a tus necesidades. Ésto es muy similar a un entorno físico.

También existen las **Máquinas virtuales interrumpibles**. Son instancias que tienen un precio bastante asequible, pero con una duración de hasta 24 horas. Se utilizan para ejecutar tareas diarias, que sean tolerante a fallos.

Utiliza máquinas virtuales interrumpibles para ejecutar tareas por lotes puntuales, que no necesiten estar encendidas más de 24 horas. Ahorrarás bastante en tu factura. Como, por ejemplo, generación de informes y/o facturas que se realicen a una hora específica del día.

Otra opción disponible consiste en poder **aislar las máquinas virtuales en nodos físicos** sin compartirlos con el resto de los clientes. Ésto es muy útil para empresas que necesiten certificar su infraestructura.

Una de las opciones avanzadas de las máquinas virtuales es poder protegerla para estar seguro de que las instancias no se expusieron a software malicioso o *rootkits* a nivel de inicio y *kernel*. **Las máquinas virtuales protegidas** se encuentran dentro de la iniciativa de **Shield Cloud** para proteger los recursos de GCP.

Antes de desplegar una máquina virtual, es recomendable comprobar las familias que hay disponibles y el precio de cada tipo, ya que éstos suelen varia con el tiempo. Aun así, cuando estás en el asistente de despliegue, se te mostrará un precio estimado en función del tipo de máquina virtual seleccionado.

COMPONENTES DE UNA MÁQUINA VIRTUAL

En GCP no cambian los componentes de la máquina virtual. La clasificación es siempre la misma y solo variará en los nombres que el proveedor quiera darle a alguno de ellos.

- Imágenes
- Discos
- Redes y seguridad
- Etiquetas

Imágenes

Consiste en los discos que contienen el sistema operativo. Por defecto dispones de distintas versiones públicas de: CentOS, CoreOS, Debian, OpenSUSE, Red Hat, SLES, Ubuntu y Windows. Algunas de las versiones que encontrarás pertenecen al programa Shield cloud, mencionado anteriormente. Seleccionar una de éstas permitirá activar las opciones de máquina protegida.

Puedes crear imágenes personalizadas a partir de discos de origen, de otras imágenes, *snapshots* o imágenes almacenadas en Cloud Storage. A través de la herramienta de importación de disco local puedes importar discos de máquinas virtuales que tienes en tu entorno físico. Soporta los formatos VMDK de VMware, VHDM de Microsoft o QCOW2. Como requisito principal solo es posible utilizar los sistemas operativos mencionados anteriormente.

Si vas a importar los discos de tu máquina virtual de tu entorno local a GCP puedes ejecutar la herramienta de verificación previa dentro de ella, para buscar problemas de compatibilidad.

Por último, está disponible el Marketplace que permite desplegar máquinas virtuales con software ya preinstalado. Podrás encontrar todas aquellas soluciones más populares que están a disposición de los fabricantes o la comunidad para ser desplegadas de manera sencilla y rápida. Hay que tener en cuenta que existen soluciones con costes de licencia, por lo que puedes encontrar imágenes en las que puedes usar tu propia licencia o bien pagar por suscripción.

Utilizar imágenes del Marketplace ahorra tiempo ya que evita tener que instalar y configurar software ya que dispones de soluciones listas para usar en producción.

Discos

Las máquinas virtuales utilizan almacenamiento por bloques y de alto rendimiento. Son discos persistentes y se encriptan automáticamente para proteger los datos. Éstos pueden ser zonal o regional. Tal y como se mencionó al comienzo de este apartado puedes utilizar discos zonales o regionales replicados en dos zonas. Están disponibles los siguientes tipos:

- **Estándar:** son discos de precio asequible, pero con menor rendimiento. Son adecuados para grandes cargas de trabajo de procesamiento de datos que usan más que nada E/S secuenciales. Ofrecen 0.75 IOPS por GB de Lectura.
- **SSD:** se utilizan para aplicaciones empresariales y bases de datos de alto rendimiento que requieren una latencia baja y más IOPS que los discos estándar. Están diseñados para latencias de milisegundos. Ofrecen 30 IOPS por GB de Lectura.
- **Balanceados:** son una alternativa para los discos SSD que disponen de un equilibrio entre el rendimiento y el coste. Tienen la misma cantidad máxima de IOPS que los discos SSD y una cantidad más baja de IOPS por GB en lectura, ofreciendo niveles

de rendimiento adecuados para la mayoría de las aplicaciones de uso general. Ofrecen 6 IOPS por GB de Lectura.

- **SSD temporalmente local:** Éstos pueden ser de tipo SCSI o NVMe y están diseñados para ofrecer IOPS muy altos y baja latencia. Solo están disponibles hasta que se detiene o elimina la instancia. Dependen de la familia de la máquina virtual para ser usados, no estando disponibles en las de uso general.

Además de los discos mencionados, es posible crear un servidor de archivos o un sistema de archivos distribuido en Compute Engine para usarlo como un sistema de archivos de red con capacidades **NFSv3** y **SMB3**.

Otra característica disponible es la capacidad de poder activar un **disco RAM** dentro de la memoria de la máquina virtual para crear un volumen de almacenamiento en bloque con alta capacidad de procesamiento y baja latencia.

Redes y seguridad

Una máquina virtual en GCP puede tener entre una y ocho interfaces de red. Debe haber disponibles subredes para que se puedan utilizar en varias interfaces de red dentro de una instancia. Tal y como se mencionó anteriormente una VPC es global y las subredes son regionales. Por lo que podrás usar como subred aquellas disponibles en la región donde vaya a ser desplegada la máquina virtual.

En GCP no se utilizan grupos de seguridad. Para aplicar reglas de firewall, se asignan etiquetas de red a las máquinas virtuales. Éstas serán usadas por el cortafuegos para aplicar las rutas o reglas que se deseen.

Es recomendable utilizar direcciones IP externas para que la máquina virtual pueda publicar aplicaciones o se pueda acceder desde el exterior. Éstas pueden ser efímeras, es decir, pueden cambiar si se para la instancia o estáticas.

Dentro de la configuración de red, existe una opción de reenvío de IP que permite que la instancia ayude a asignar la ruta de los paquetes. De esta forma se puede evitar que una máquina virtual reenvíe paquetes que originó otra. Para poderla usar se debe de activar justo en el momento de la creación de la máquina virtual y se debe de configurar a nivel de sistema operativo.

Etiquetas

Por último y no menos importantes son las etiquetas. Éstas no son obligatorias para el funcionamiento de una máquina virtual, pero aportan bastante valor. Utilizarlas ayuda a agrupar recursos que están relacionados entre sí. Además de poder utilizar herramientas de terceros para disponer de un buen inventariado o CMDB. Recuerda utilizarlas, establece una regla en tu empresa o equipo para mantener un orden y que están ofrezcan información útil sobre el proyecto, propietario, entorno, etc.



PRUEBA GRATUITA

Backup y recuperación n.º 1

VMware y Hyper-V

- Solución independiente de hipervisor para todas sus máquinas virtuales
- Soporte para VMware, Hyper-V y Nutanix AHV
- Recuperación rápida y confiable de VM completas a elementos individuales

vmware



LEER INFORME



Capítulo 15

COMUNIDAD EN TI - VMUG, vExpert, vBrownBag, Social Networks



Ariel Sánchez

@arielsanchezmor

COMUNIDAD EN TI

Bienvenidos a la segunda edición del libro escrito por vExperts en la lengua hispana. Este es mi segundo aporte y voy a entrar más a fondo en un tema que mencione en la primera edición: involucrarse en la comunidad de VMware (y otros vendedores) es ahora una herramienta esencial para los arquitectos y administradores latinos.

BARRERAS Y LÍMITES

Parecerá extraño empezar esta discusión hablando de barreras y límites. Pero en mi experiencia, es esencial hablar de esto primero para poder tener éxito a largo plazo, y particularmente en términos de aprovechamiento de las comunidades de Tecnologías de Información (TI).

Una barrera importante para muchos hispanohablantes es el **idioma**. Al fin y al cabo, el mundo de TI se maneja principalmente en inglés. Debemos estar dispuestos a comunicarnos en nuestra lengua nativa y también en inglés, para sacar el mayor provecho a los recursos y a la comunidad mundial. Verdaderamente, *mejorar el manejo del idioma inglés* es importante para todo administrador y arquitecto hispanohablante, y puede tener gran impacto en nuestra carrera profesional. Debemos poder leerlo, comprenderlo, redactarlo y conversarlo.

Otra barrera que tenemos es **mental**. A veces pensamos que no podemos lograr algo, o que no tenemos el tiempo. *Propongo que busquemos vencer esta barrera mental cada día*. Por más que lo logremos, queda mucho por perseguir, más gente por ayudar, nuevos retos a la vuelta de cada esquina. Perdamos el miedo, y decidamos usar todo recurso a nuestra disposición para mejorar. He visto a muchos en la comunidad que, con un poco de esfuerzo y tiempo, logran tener influencia y avanzar en sus carreras, con solo proponérselo.

En cuanto a límites, aconsejo dedicar **tiempo** a pensar en cómo queremos manejar a largo plazo estas herramientas, y cuánto de nuestra vida personal queremos invertir en lo que se convierte en parte de nuestro perfil profesional. Siempre debemos cuidar de que nuestras interacciones sean algo de lo cual estemos orgullosos, ya sea mañana o dentro de 5 años; esto aplica a nivel personal y en línea. En mi caso, mis interacciones familiares a través de redes sociales son mínimas, pero soy muy abierto con las amistades que he hecho en la comunidad.

Un concepto importante en todas nuestras interacciones es el **respeto a los otros**. Cuando ponemos algo en línea, debemos respetar los derechos y sensibilidades de los demás. Si tenemos la duda, es mejor aclararla antes de actuar, pues podemos hacer mucho daño digitalmente. También debemos ser cuidadosos con la información de nuestros empleadores y clientes.

Finalmente, gracias al COVID hemos empezado a compartir más en nuestro hogar y vida diaria, sobre todo para los que conseguimos seguir trabajando remotamente. Tener un buen micrófono, presentarnos favorablemente a través de la cámara y poder trabajar eficazmente es una habilidad crítica en el año 2020. Nuestra **presencia digital** es más importante que nunca, y debemos encontrar el balance adecuado.

CONSEJOS PARA EL USO DE REDES SOCIALES Y CHATS

Debemos empezar aceptando que las redes sociales y grupos de chat son métodos de comunicación modernos que podemos aprovechar para obtener información, formar relaciones y para crear influencia. A medida que pasan los años encontramos nuevas tecnologías que nos permiten acortar distancias y compartir más información. Debemos pensar en el uso de las redes sociales como herramientas para crear nuestra marca personal.

CREAR NUESTRA MARCA PERSONAL

Una buena definición de nuestra marca personal es como otros nos describen cuando no estamos presentes. ¿Cómo nos presentan nuestros colegas? ¿Qué cualidades nos asignan? Tal vez seamos presentados como los expertos de una tecnología en particular, como el líder multi disciplinario de proyectos complejos, o reconocidos por nuestras experiencias.

Si no sabemos cuál es nuestra marca personal, debemos buscar conocerla y asegurarnos de estar a gusto con ella. Les quiero compartir consejos que siento que aplican para muchas situaciones en lo que respecta a construir una marca personal.

Empecemos con el pie derecho. Primero, debemos escoger un *correo electrónico* que sea apropiado para interacciones profesionales – un correo electrónico que tenga nuestro nombre. Pensemos que este correo electrónico sería uno que pondríamos en un currículum vitae.

Ahora, usaremos ese correo electrónico para hacer nuestras *cuentas en redes sociales*. Muchos de los proveedores solicitarán también algún número de teléfono móvil para tener autenticación de dos factores, y esto es buena idea para darle cierto nivel de protección a nuestras interacciones.

Cuando creamos nuestros perfiles, debemos usar nuestros nombres reales, incluir una foto que muestre nuestra cara, e incluir información que nos haga fácilmente reconocibles. Esto, con el objetivo de que seamos fáciles de encontrar y que, los que nos encuentren, inmediatamente vean el valor que aportamos.

Es cierto que las medidas que he explicado pueden ser ignoradas, y aun así se puede conseguir mucha información con un usuario “de mentiras”. Pero además de conseguir información, queremos aprovechar y crear un legado de influencia, y la habilidad de poder ser reconocido como un experto. Esto vendrá al tema más tarde en este capítulo, cuando hablamos de algunos programas como vExpert.

Una de las lecciones más poderosas para tener el respeto y la consideración de otros es ser **valioso**. En mi opinión, *somos más valiosos cuando ayudamos a otros* y nuestras interacciones aportan en vez de ser problemáticas.

Piénsalo dos veces antes de criticar públicamente; ¿si estuvieras en la posición inversa, como te gustaría que otros te dieran su opinión, o te corrigieran? Es importante lograr amistades a largo plazo, aunque sea en forma digital.

LINKEDIN Y TWITTER

En particular, me parece que debemos manejar y ser activos en al menos dos plataformas sociales: **LinkedIn** y **Twitter**. Sin embargo, cada una son útiles por razones diferentes.

Twitter es la plataforma de más uso en las comunidades de TI. Funciona como un directorio mundial. Muchos expositores solo incluyen su usuario de Twitter en su diapositiva de introducción. No tiene tanta acogida en Latinoamérica y España como otras redes sociales, por lo que puede resultar más sencillo usar esta plataforma como una herramienta exclusiva de acceso a las comunidades de TI.

Una diferencia importante de Twitter es que, en general, no se necesita el permiso de una persona para seguirla. Twitter trata a cada persona como una fuente de contenido. El contenido puede ser original o simplemente compartido. Esto permite que cualquier persona pueda crear una cuenta y personalizarla para las comunidades de TI.

Aconsejo empezar siguiendo a los autores de este libro y otros que has encontrado útil. Es común que en Twitter los autores respondan y se comuniquen con su público. También recomiendo seguir a los que ya lees por su blog o sus presentaciones.

En Twitter se encuentran opiniones y discusiones directas entre profesionales. Muchas empresas como VMware la usan como su plataforma principal de comunicación y difusión, y es común que haya más discusión en Twitter que en los comentarios de las páginas web.

Cada producto tiene su propia cuenta, y también se usan hashtags (etiquetas con el símbolo numeral) como #vSphere, #RunNSX y en particular para nuestro caso, #vCommunity y #CloudPorvExperts. Los hashtags se usan para encontrar mensajes relacionados a un tema – se colocan en la caja de búsqueda, y permiten encontrar todos los mensajes que contienen ese hashtag.

LinkedIn es la plataforma profesional más universal. Es particularmente importante para aquellos que están en posiciones de liderazgo, ventas y mercadeo. Las interacciones tienden a ser formales, y es un buen lugar para colocar nuestras certificaciones, historial de empleo, pensamientos y eventos corporativos en los que participamos. Una de las mejores cualidades de LinkedIn es poder pedir referencias personales, por lo que debemos asegurarnos de tener algunas referencias que nos den una buena carta de presentación.

En Latinoamérica, se acostumbra a mandar una invitación de conexión para alguien que uno desea conocer, pero en otros países mucha gente no acepta invitaciones si no conocen a la persona o no están interesados en un trabajo. Se ha vuelto más útil con la adopción de los hashtags y la habilidad de “seguir” a alguien sin estar conectado directamente.

Existen comunidades virtuales en Facebook e Instagram, pero no tienen la misma acogida que Twitter y LinkedIn, por lo que las considero plataformas adicionales y no estrictamente requeridas. Este es un caso donde debemos ir a la fuente, y para muchas comunidades de TI y de proveedores de la nube, la fuente está en Twitter.

FOROS Y GRUPOS DE CHAT

La plataforma con la que empezó la comunidad para VMware fue el foro virtual llamado VMware Technology Network (<https://communities.VMware.com>). A pesar de ser un medio más lento, aún tiene gran acogida, y para ciertas tecnologías son aún el mejor recurso. Es un recurso especialmente valioso para los que están interesados en los detalles más técnicos.

Se debe hacer una importante mención para las tecnologías de chat. Particularmente, en Latinoamérica y España hay muchos que usan grupos de Facebook, Whatsapp y Telegram con mucho éxito. Nuevas plataformas como Slack y Discord también están gozando de buena acogida.

Un factor importante de estas plataformas es que no son enteramente públicas, por lo que el grado de aprovechamiento y utilidad dependen de estar en el grupo adecuado. También es de mal gusto compartir lo que uno ha logrado o discutido en un grupo privado, por lo que muchas veces no podemos usar estas interacciones como referencias de nuestra actividad digital.

vCOMMUNITY

Existe en inglés el concepto de vCommunity: la comunidad mundial de ingenieros, administradores y empleados de VMware que están dispuestos a ayudarse entre sí. Típicamente lo verás con un hashtag (#vCommunity) porque se usa principalmente en Twitter, cuando alguien hace un post donde se pide o se aprecia ayuda. El término vCommunity es un término amplio que sirve para incluir a todas las diferentes expresiones de la comunidad virtual.

En mi opinión, las principales vías donde se manifiesta la vCommunity son:

- Los foros de VMware, principalmente VMTN (communities.VMware.com) y Twitter.
- Las reuniones locales y virtuales del VMware User Group.
- Las contribuciones individuales en blogs, podcasts y grabaciones de YouTube.
- El programa vExpert

Gracias a programas como vExpert, que se dedica a fomentar la discusión y ayuda entre usuarios, y al VMware User Group (VMUG), la vCommunity se ha caracterizado por ofrecer un ambiente amigable a principiantes y nuevos integrantes. Yo le debo muchísimo de mi crecimiento profesional a estos programas, y quiero asegurarme de que los lectores de este libro sepan que existen y que también le saquen provecho.

Mi consejo para la mayoría de las personas que empiezan a explorar estos programas es que primero vean y aprendan, pero rápidamente se animen a participar y compartir sus propias

experiencias. Compartir con otros nos hace mejores y nos ayuda a crear nuestra marca personal. El siguiente paso es ayudar a otros a descubrir estos recursos y ser un mentor.

VMUG

El grupo de usuarios de VMware (<https://www.vmug.com>) es una entidad dedicada a ayudar a los administradores y arquitectos a encontrar información y apoyo en la adopción de tecnologías de VMware. Se agrupa en capítulos locales, que tienen 2 o más líderes, típicamente voluntarios, y que no son empleados de VMware.

Hasta septiembre de 2020, estos son los capítulos existentes en Latinoamérica y España:

- Argentina
- Chile
- Costa Rica
- Ecuador
- Honduras
- El Salvador
- Guatemala
- México
- Nicaragua
- Panamá

Colombia

- Bogotá
- Medellín

Brasil

- Brasilia
- Brasil (Minas Gerais)
- Brasil (Paraná)
- Brasil (Rio de Janeiro)
- Brasil (Rio Grande do Sul)
- Brasil (Sao Paulo)

España

- Asturias
- Barcelona
- Galicia
- Iberia Norte
- Madrid

Cada capítulo se compromete a efectuar un número de reuniones cada año. Para no perderse las comunicaciones, hay que crear una cuenta en el sitio web, buscar el capítulo y “unirse” virtualmente. El sitio web ofrece un foro para cada capítulo y ahí se anuncian los detalles.

Una reunión de VMUG típicamente es patrocinada por una marca a la que le interesa que la comunidad local conozca su producto, y a veces se acompaña con una ligera comida. En particular, mi parte favorita de atender estos eventos es la posibilidad de conocer a otros profesionales y compartir conocimientos e historias. Es buena idea conocer a los líderes y proponerles dar una charla sobre algún tema que conocemos bien (esto nos va a ayudar con nuestra marca personal).

También acostumbro a tomar al menos una foto de cada evento al que voy, y compartirla en Twitter. Tomo fotos de los líderes y de las personas que hablan, para que quede evidencia del evento; les pregunto su usuario para poder etiquetarlos y que ellos las tengan también. Es un gesto pequeño, pero ayuda a hacer comunidad, y puede ayudar con sus aplicaciones para vExpert u otros programas del trabajo.

Recomiendo también involucrarse como líder de VMUG si se ven oportunidades para mejorar o ayudar. Fui líder de un VMUG antes de ser empleado de VMware y fue una experiencia muy bonita. Los líderes de VMUG son muy respetados en la comunidad y tienen acceso a ciertos beneficios importantes y útiles.

BLOGS Y PODCASTS

Una forma común de compartir información sin tener que esperar una reunión de VMUG es a través de un blog. Este es uno de los métodos más efectivos para controlar nuestra marca personal pues podemos ajustar los mensajes, imágenes y contenido.

Es relativamente barato conseguir un URL con registro privado; por ejemplo, arielsanchezmora.com me cuesta \$12 al año usando domains.google.com. Escoger una plataforma es un tema más personal – Blogger y Wordpress.com son gratuitos, pero no permiten completa flexibilidad. Otras opciones populares son Github Pages y manejar nuestro propio servidor web. Tal vez es más importante el contenido que los detalles técnicos, pero cada uno tiene derecho a escoger como hace las cosas, y hablar de estas elecciones es un tema muy divertido en nuestra comunidad.

Muchos empezamos nuestros blogs con el propósito de tener repositorios de información; si no encontramos algo fácilmente en Google o en la documentación, hay una alta posibilidad de que otros también tengan el mismo problema. Es muy gratificante recibir un comentario de que nuestro esfuerzo en escribir un post ayudó a alguien más.

Cuando escribimos algo, no importa cuán pequeño sea, recordemos compartirlo en nuestras redes sociales. Busquemos la opinión de otros y preguntémos como mejorar. Esto nos va a ayudar a ser mejores comunicadores y a mejorar nuestra marca personal.

De nuevo, el idioma puede ser una barrera. Algunos bloggers hispanohablantes han logrado instalar servicios de traducción en sus blogs. Mi recomendación es hacer los posts en español e inglés para tener el máximo impacto.

Los podcasts son también muy importantes en la vCommunity. Permiten mucha más información que un blog post, y muchas veces son solo audio, por lo que pueden ser

consumidos mientras hacemos otra cosa. Tienden a ser más informales, y nos ayudan a relajarnos y conversar con nuestras comunidades.

Empezar un podcast requiere más compromiso que un blog, pero tampoco es tan difícil. Lo más difícil es seguir creando contenido; si no se está seguro, es mejor empezar siendo un invitado.

En particular, para los lectores de este libro, les recomiendo altamente buscar “Un Podcast para TI” liderado por Federico Cinalli (@FCinalliP) y Héctor Herrero (nheobug). En inglés, busquen el “VMTN Community Podcast”, donde se discute mucha información del vCommunity.

vBROWNBAG Y VMUNDERGROUND

Quiero resaltar a la organización llamada vBrownbag, de la cual soy parte. No lo veo como un podcast tradicional, sino más bien como una herramienta de aprendizaje creada por y para la comunidad. Cada semana se invita a un especialista y se le pide que exponga sobre un tema. Esto se graba en vídeo y se publica tanto en YouTube, como en otras plataformas como iTunes. Tiene capítulos para América, Europa y Asia, y se graban en español e inglés. La información se puede encontrar en <https://vbrownbag.com/>

Es común encontrar en vBrownBag guías de estudio para certificaciones, exposiciones de nuevas tecnologías, y grabaciones en vivo de conferencias de TI. Yo siempre ando buscando personas para exponer en los canales de Europa o Latinoamérica. Si algún día quieres presentar un tema, solo déjamelo saber a través de Twitter.

vBrownbag también colabora con el equipo de VMworld para aceptar charlas de la comunidad, grabarlas y subirlas en Youtube. Las mesas cerca de este escenario siempre constituyen un excelente punto de encuentro para ver a otros miembros de nuestra vCommunity.

VMunderground es otro grupo de voluntarios, famoso en nuestra comunidad por organizar una fiesta informal en VMworld y también por organizar los Opening Acts. Opening Acts son paneles abiertos que se efectúan al inicio de VMworld con miembros de la comunidad.

vEXPERT

El programa vExpert es un programa oficial de VMware que premia a los miembros de la comunidad que demuestran ayudar a otros. Cada autor de este libro es un vExpert, algunos de muchos años y en varias categorías. Se puede encontrar todo el directorio de vExperts en <https://vexpert.VMware.com/directory>

El programa acepta aplicaciones dos veces al año. Para aplicar al programa se debe hacer una cuenta en <https://vexpert.VMware.com/> y proveer datos de como se ha ayudado a otros. Toda nuestra actividad en la comunidad aplica. Aquí es cuando podemos referirnos a nuestros blogs, los eventos que asistimos en Twitter, las charlas que dimos en VMUG o las

participaciones en los foros y podcasts, o nuestra labor como líderes de proyectos de comunidad. Se considera un premio y conlleva beneficios que se detallan en la página web.

La ayuda puede ser de varias formas, pero debe ser claramente ayuda para otros. Por ejemplo, asistir a un evento no califica – pero presentar en el evento para ayudar a la comunidad sí. También hay que ser claro q la ayuda debe ser adicional a nuestro trabajo diario – ser ingeniero de ventas y presentar el producto que vendemos no se considera ayuda a la comunidad.

Ofrezco dos consejos para aquellos interesados en aplicar al programa de vExpert:

- Conseguir un mentor que te ayude a valorar tu progreso.
- Documentar toda actividad que haces a lo largo del año.

Cualquier vExpert puede ser mentor, pero hay una categoría de vExpert que se especializa en ayudar a los principiantes – vExpert PRO. Se trata de que haya para cada país alrededor del mundo, para que se pueda ayudar en el idioma local.

Están listados en <https://vexpert.VMware.com/directory/pro>.

No es necesario acudir específicamente al de un país; cualquiera puede ayudar, y yo personalmente me ofrezco para los lectores de este libro.

NOTAS FINALES

Cada uno de los autores de este libro incluyó su usuario de Twitter. Es buena idea seguirlos, y a medida que lees el libro, mandarles un mensaje con las impresiones que te dejó su capítulo. Es evidente que estamos muy dispuestos a oír tu opinión y discutir contigo cómo mejorar el contenido, puesto que todos tenemos esa actitud de ayudar a otros y mejorar continuamente. Esa actitud es para mí es el legado más valioso de ser parte de esta vCommunity.

Cuando ayudas al resto a mejorar, estarás participando automáticamente en el vCommunity. Todos tenemos un trabajo que nos da de comer, pero cuando somos parte de una comunidad mundial donde se incentiva a ayudar a otros, el trabajo se hace más liviano y placentero.

Descubrirás que hay muchas más personas que saben de VMware y hablan español de lo que pensabas. Además, las relaciones que se hacen en línea, a nivel mundial, se vuelven reales cuando asistes a eventos como VMworld y VMUGs en varias ciudades.

Me gustaría aclarar que he hablado bastante de la comunidad relacionada a VMware, pero hay otros grupos de usuarios y premios/reconocimientos que ayudan a otras empresas y otros ecosistemas. El programa Vanguard de Veeam es muy reconocido en nuestra comunidad. Microsoft tiene VMPs, Amazon tiene sus AWS Heroes; cada marca tiene sus programas.

También es cierto que no siempre es necesario ocupar una marca para hacer comunidad. Una de mis comunidades favoritas es el grupo de usuarios de BSD, cuyas reuniones son puramente entre usuarios. Es bueno recordar que el valor de estos grupos siempre está en los usuarios – en las historias que compartimos y los amigos que hacemos.

Me despido con una foto que tomamos en Barcelona, en VMworld 2019. Fue tomada después de la publicación de la primera edición de este libro. Yo no conocía a ninguno de estos caballeros personalmente, pero ya los había conocido a través de Twitter, sus blogs, sus capítulos e interacciones por WhatsApp. Ahora los considero hermanos, y faltan varios autores en la foto, varios nuevos para esta edición.



El estar involucrado con las comunidades nos abre puertas que no se lograrían de otra manera. Esta escena se ha repetido muchísimas veces, en varias ciudades, por simplemente tener la disposición de ser parte de la comunidad y querer ayudar a otros. ¡Espero en el futuro poder tomarme una foto contigo, y que me cuentes como te fue con estos consejos!

#VMwarePorvExperts

www.vmwareporvexperts.org

¡Descarga gratuita
sin registro!

El Libro de VMware

VMware por vExperts

por Bloggers

Versión 1.0

Bloggers unidos por un Proyecto solidario.

Todo el dinero recaudado gracias a nuestros Sponsors va a ser donado a dos causas solidarias.



Miguel Angel Alonso • Xavier Caballé • Patricio Cerdá • Federico Cinalli • Jorge de la Cruz • Xavier Genestos • Hector Herrero • Ricard Ibáñez • Gorka Izquierdo • Leandro Ariel Leonhardt • Miquel Mariano • Daniel Romero • Ariel Sánchez • Raúl Unzué

Prólogo por Duncan Epping

Capítulo 16

EL ESTRÉS: una constante en TI



Celia Cristaldo Canterbury

@celiacri

EL ESTRÉS: UNA CONSTANTE EN TI

Últimamente hemos escuchado bastante la palabra “burnout” o Desgaste Profesional. En TI, nuestro trabajo nos demanda gran cantidad de horas extras, mucho esfuerzo mental para focalizarnos en resolver problemas de clientes (muchas veces de capa 8), eternas horas sentados, fines de semana y noches sin dormir.

Estamos casi acostumbrados a este ritmo de vida, que en cierto momento pasa a ser una rutina más; incluso, si tenemos tiempo de descansar, nos cuesta conciliar el sueño o dejar de pensar.

Personalmente, he trabajado en un centro de datos como operadora, administradora de virtualización y nivel 3 de troubleshooting, y sé que empatizan conmigo con solo nombrar estos puestos. Ni siquiera hace falta describir los estragos que causaron en mi vida, en mi cuerpo, en mis relaciones.

El tema es, ¿sabemos realmente a lo que nos exponemos con este tipo de rutina? Me atrevo a decir que no... quizás pensamos en que afectaría nuestro sueño, quizás un poco nuestra interacción familiar, pero déjenme explicarles en las próximas páginas que es mucho más que eso, y que silenciosamente nos corroen desde las entrañas. #Drama

Lo que quiero resaltar en este capítulo es qué amenazas a nuestra salud hay debajo de la superficie y qué estamos dejando pasar desapercibidamente, que nos ronda como enemigo silencioso esperando el momento para atacar. Mi intención es generar conciencia para darnos cuenta de los riesgos con los que convivimos, de esta forma poder actuar en contrapartida.

Más de una persona se sentirá identificada con los síntomas que iré citando, que pasan a ser parte de una, ya ni lo notamos tras varios años en esta seudo rutina.

UN POCO DE TEORÍA: ¿QUÉ ES EL “BURNOUT”?

Vayamos al principio de todo...

La palabra atribuida *burnout* se debe a [Herbert Freudenberger](#) (psicólogo clínico neoyorkino) que la empleó para describir a aquellos voluntarios que en un periodo de entre uno y tres años, se encontraban desmotivados y sin interés por su trabajo. Luego, Christina Maslach dio a conocer este concepto en 1977 en una convención de la Asociación Americana de Psicólogos.

También conocido como Síndrome del Desgaste Profesional. La Organización Mundial de la Salud (OMS) la define como '*la reacción que puede tener el individuo ante exigencias y presiones laborales que no se ajustan – (o cree que no) – a sus capacidades y conocimientos, y ponen a prueba su capacidad para afrontar la situación*'. La OMS ha incluido el *burnout* oficialmente en la Clasificación Internacional de Enfermedades.

La Comisión Europea de Seguridad y Salud en el Trabajo define el desgaste laboral como 'las nocivas reacciones físicas y emocionales que ocurren cuando las exigencias del trabajo no igualan las capacidades, los recursos o las necesidades del trabajador'

El burnout es la respuesta negativa del individuo al Estrés Laboral, pero ¿qué es el estrés?

HABLEMOS DEL ESTRÉS

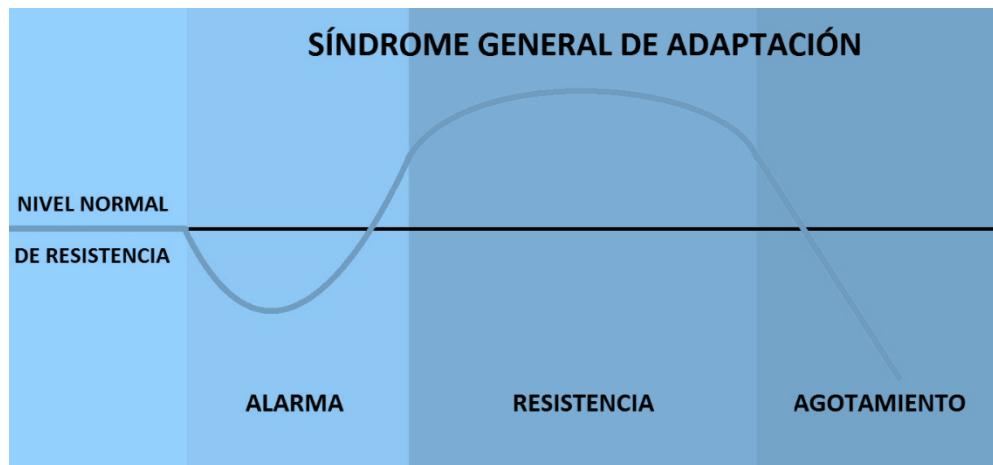
En los años 30, Hans Selye, fisiólogo y médico austrohúngaro-canadiense, director del Instituto de Medicina y Cirugía Experimental de la Universidad de Montreal, descubrió un trastorno generalizado de adaptación, el cual conocemos como estrés, y Selye la definió como una '*respuesta del organismo ante la percepción de una amenaza, caracterizada por una fase de alarma, una fase de resistencia, y una fase de agotamiento*'.

Richard Lazarus (psicólogo, profesor e investigador estadounidense) y Susan Folkman (psicóloga estadounidense y profesora emérita de medicina en la Universidad de California) definieron el estrés crónico como '*una relación particular que se establece entre el sujeto y el entorno que es evaluada por éste como amenazante y desbordante*'.

El estrés no es negativo, de hecho, es *necesario*, ya que es lo que nos protege de las amenazas en el entorno y hace que reaccionemos para "salvarnos", se manifiesta durante un examen, previo a una cirugía, antes de hablar en público, si nos quieren asaltar, incluso nos ayuda a mejorar en algunas tareas (Estrés Positivo), etc. El cerebro tiene el objetivo de que sobrevivamos.

El problema empieza cuando el ciclo de *percepción de amenaza – alarma – resistencia – agotamiento* no se rompe y no hay una recuperación de nuestro sistema ante esta reacción. No nos reparamos, nos quedamos en un *loop*.

Entonces, completando el concepto anterior, el burnout es la respuesta negativa del individuo al Estrés Laboral cuando la *amenaza* está relacionada con la gestión del trabajo y la organización donde se desempeña, haciendo que su organismo quede *atrapado* en las fases de resistencia y agotamiento de forma permanente.



Fases del Estrés natural según Selye

Ignacio Morón, profesor de la Universidad de Granada (España), investigador del Centro de Investigación Mente, Cerebro y Comportamiento (CIMCYC) dice que, en reposo, "el cerebro puede consumir unas 350 calorías en 24 horas, o sea, un 20% de la energía corporal que gastamos al día".

Cuando estamos muy concentrados en una tarea, el cerebro desactiva todo lo demás para enfocarse, dando paso a las fases del estrés, que como les mencioné, es un estrés positivo.

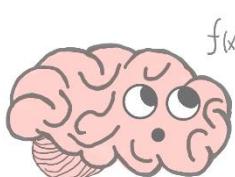
La última fase de *agotamiento* es la que lleva al desgaste emocional, físico y sobre todo si se extiende, al desgaste profesional. Este desgaste profesional tiende a causar una pérdida lenta de motivación, compromiso personal y frustración en la labor que se realiza. La persona se siente exhausta, agotada, “quemada” y su rutina le lleva a repetirlo constantemente.

En general, en mayor o menor medida, podemos verlo reflejado en nuestros años laborales, sobre todo en los primeros años en los cuales se requiere un alto esfuerzo y demostraciones de valía para ganarnos el puesto, un mejor salario, reputación, experiencia, etc.

Los síntomas más comunes que vemos a primera vista sin indagar mucho son las ojeras y el cansancio implícito, por largas horas de insomnio, producto de instalaciones o desarrollos que haya que entregar con premura o atraso, problemas que solucionar en clientes, caídas de servicios que, como la bendita Ley de Murphy manda, tienden a ocurrir durante la noche o fines de semana. Ni qué decir cuando Murphy aparece durante trabajos de actualización de sistemas o hardware.

Cansancio físico constante, sueño arrastrado, muchas veces una tendencia a tener más apetito de lo normal a cualquier hora. Dolores de cabeza, musculares, sobre todo en el área de la espalda, cuello y hombros. El cuerpo tiende a descoordinarse, o simplemente no logra tener una rutina biológica circadiana, o sea, nuestro sistema pierde la noción del tiempo y se desordena lentamente.

Y si sumamos los efectos emocionales como la desmotivación, nos cuesta levantarnos para ir a trabajar, ansiedad y lentitud en nuestras reacciones. Muchas veces nos ponemos hiperactivos y terminamos abrumados, con el cerebro haciendo eco y la sensación de que tenemos cosas pendientes todo el tiempo.



$$f(x) = \sin(2x^3)$$

Un tema no menor: los neurocientíficos advierten que el cerebro no está preparado para el multitasking o multitarea. Lo “ideal” es enfocarse en la menor cantidad de cosas. El llenarnos de tareas al mismo tiempo deja *sin caché* a nuestro cerebro. Esto aumenta la tasa de errores.

Earl Miller, neurocientífico del MIT (Massachusetts Institute of Technology), uno de los expertos en Atención Dividida, expresa que **el cerebro no está preparado para realizar múltiples tareas correctamente**. “*Cuando se intenta ser multitarea, lo que hace realmente el cerebro es cambiar de una tarea a otra muy rápidamente*”. Y esto implica un costo cognitivo alto.

Uno de los principales actores químicos del estrés en nuestro cuerpo es el **cortisol**, que es una hormona esteroide producida por las glándulas suprarrenales. Es la encargada de mantenernos focalizados.

Para resumir sus funciones en palabras simples y sencillas, el cortisol eleva el azúcar en la sangre y ácidos grasos, entre otras sustancias, para tener la energía de reaccionar ante el peligro que nos alarma, es como comerse una barra de proteína que inyecta de energía al cerebro y al cuerpo. Desconecta cualquier función “accesorio” de nuestro sistema para poder centralizar todos sus recursos en “salvarnos” de la situación que causa el estrés.

Y aquí viene algo preocupante. Esta desactivación de funciones “innecesarias” en el ciclo de estrés es donde empiezan los problemas si nos mantenemos en este estado.

Algunas de esas funciones que se detienen son:

- El apetito sexual
- Las funciones reproductivas
- El crecimiento
- La digestión
- El sueño

La sanación rápida de heridas y la inmunidad logran su peak durante el estado de alerta, pero se degeneran si este estado se prolonga

Sin entrar mucho en detalles científicos, lo que cabe resaltar basado en lo expuesto anteriormente, es que, *si el cuerpo se encuentra constantemente amenazado en ambiente "hostil"*, sin descanso adecuado, en constante alerta, empiezan los problemas físicos, emocionales, y que a su vez afectan el entorno relacional del individuo.

A continuación, entro en detalle en algunos de los problemas más comunes del estrés negativo prolongado convertido en burn out.



RESPUESTA SEXUAL Y EMOCIONAL

Manteniendo el cortisol aumentado en el organismo, pueden producirse problemas en el funcionamiento sexual de la persona, lo que puede ocasionar aparición de Disfunción Eréctil, Deseo Sexual Hipoactivo (falta de deseo), Eyaculación precoz (normalmente causado por el constante estado de ansiedad), Vaginismo o dificultad para el coito, por citar algunos. A su vez, esto genera más estrés de la que ya se tiene, una reacción en cadena.

Si la persona se encuentra en pareja, esto resulta en peleas, discusiones y posible separación, con un sin número de consecuencias más.

Lo mismo si se encuentran en busca de un embarazo, el cuerpo desactiva esta función “poco importante” durante el estrés y esto deriva en un problema para concebir. Tanto en hombres como en mujeres.

Es importante que, si detectan estas anomalías y estén en tratamiento, mencionen al profesional médico su profesión y cuál es la rutina de trabajo. Seguro le ayudará saber que puede ser una arista del problema.

PROBLEMAS METABÓLICOS

También mencionaba anteriormente el aumento de glucosa en la sangre para mantener al cuerpo alerta. Aquí aparece la insulina, que es una hormona segregada normalmente por el páncreas, el cual permite que ingrese la glucosa necesaria a las células a través de la sangre.

Cuando los receptores de insulina no funcionan correctamente, empieza lo que se conoce como Resistencia a la insulina. El resultado es que la glucosa producida en exceso no puede entrar a las células y se queda en la sangre. Con el tiempo, el nivel de glucosa en la sangre se acumula y puede resultar en Diabetes. Adicionalmente, el páncreas trabaja más duro para producir más insulina. Este proceso crea niveles de insulina altos en la sangre y se llama hiperinsulinemia.

Uff... en palabras simples y sencillas, tenemos más hambre y ganas de comer a cada rato porque las células no reciben alimento correctamente, causando los famosos "antojos". Por lo general, nos llenamos con comida chatarra y dulces cuando estamos bajo mucha presión o muy cansados. Nunca una manzana, nunca... el cuerpo nos pide carbohidratos y azúcar... o es lo que nos decimos para culpar a algo de nuestro mal comer.



A menudo, la resistencia a la insulina es asociada con niveles de triglicéridos elevados, alta presión en la sangre y *obesidad*. Separadamente, estos factores aumentan su riesgo para desarrollar enfermedades cardíacas. Juntos, son un riesgo mucho más serio. Toda esta explicación es lo que se conoce como Síndrome Metabólico.

Alarmante, pero realista. Y muchas veces no somos conscientes de lo que puede acarrear esas noches de trabajo comiéndonos la pizza que nos "suplica" la ansiedad, sumado a la falta de sueño necesario.

Ni siquiera entraré en detalles sobre la suma de otros factores como el sedentarismo y el tabaquismo (uno de los mecanismos más comunes para paliar la ansiedad, muy utilizado en nuestra profesión).

DEFICIENCIA INMUNOLÓGICA

Repasando un poco, durante la respuesta al estrés equilibrado, el cuerpo se prepara para “luchar” contra una amenaza, por lo que está en el peak de inmunidad reforzada y sanación de heridas. Pero, al prolongarse esto, ocurre lo contrario, el cuerpo se desequilibra y el sistema inmunológico se deteriora. ¿Cuántas veces no hemos caído en una gripe que parece ser eterna? Piensen y vean si coincide con una época de trabajo duro. Yo tengo como para llenar un libro.

Investigadores de la Universidad Politécnica de Dortmund, mediante el estudio Pragdis «Seguridad laboral y sanitaria preventiva en carreras laborales discontinuas» (financiado por el Fondo Social Europeo, 2007 – 2011), revelaron que especialistas del sector de Informática no sobresalen en las estadísticas de bajas por enfermedad. Lo curioso de esto es la razón: ¡los profesionales de TI seguimos trabajando... aunque estemos enfermos! Este fenómeno se llama *Presentismo Laboral*.

No en vano los médicos mandan a reposar a los enfermos, porque el cuerpo necesita descansar y salir del estado de estrés para sanar. Pero si no nos damos estos espacios, el cuerpo estresado constantemente no logra salir de la enfermedad, prolongándose. Todo esto suena muy básico puesto por escrito, pero si lo piensan, aun así, seguimos haciéndolo.

Dermatológicas	Pruritos / ronchas Alopecia Dermatitis atípica Sudoración excesiva
Gastrointestinales	Úlcera Colon irritable Digestión lenta Colitis ulcerosa Dispepsia / Aerofagia (gases)
Respiratorios	Apnea (ronquidos) Sensación de opresión en la caja torácica Asma Hiperventilación
Otros	Presión sanguínea con cambios extremos

Otras reacciones físicas notorias de la deficiencia de inmunidad a causa del estrés prolongado

RELACIONES INTERPERSONALES Y AFECTIVIDAD

A nivel social, las principales consecuencias vienen asociadas a las actitudes y conductas de carácter negativo desarrolladas por la persona, como consecuencias del estrés y el desgaste profesional, tales como la suspicacia, la agresividad, el aislamiento, la irritabilidad.

El trabajo se ha vuelto el centro de nuestra vida, ocupa 1/3 de nuestro tiempo, considerando que una de esa parte corresponde a estar dormidos.

A mayor tiempo de trabajo o contingencia, menos tiempo con la familia o pareja, sumado al cansancio que conlleva, y a su vez podemos perder un poco la paciencia ante situaciones que requieren un cerebro fresco para enfrentarlas. Todos estamos expuestos a problemas familiares.

Como si faltara más, se nos vino una pandemia... Esas horas de trabajo se alargaron, ahora hay una delgada línea entre las horas laborales y la vida familiar. El escenario se nos complicó bastante. Además del estrés que nos genera nuestro trabajo, sumamos la incertidumbre de no saber qué pasará en esta nueva realidad que se nos presenta. Tememos por nuestro puesto de trabajo, por nuestra salud, por la de nuestra familia. Por lo tanto, no es menor esta carga de preocupación adicional. Suma y sigue.

No dudo que más de uno pasa ahora tiempo extra sin remuneración frente al pc, en videollamadas, o atendiendo mensajes del jefe o clientes, porque todos asumen que, si estás trabajando desde la casa, estás disponible 7 x 24. La empatía brilla por su ausencia.

Es más fácil para el ser humano reaccionar que pensar, ya que pensar gasta más energía. Entonces, cuando la persona está cansada y estresada de más, reacciona.

Estas actitudes pueden deteriorar progresivamente las relaciones interpersonales generando una serie de consecuencias como son los conflictos interpersonales, la evitación de contactos sociales, o la pérdida de redes de apoyo. Es una bola de nieve.

Sabemos que esto no va a detenerse, ya que la presión del mercado es cada vez más fuerte, entre la competencia con nuestros propios colegas, y las presiones para cumplir con nuestros objetivos, no hay tregua.

En cuanto al tema afectivo, en mi continua búsqueda personal de integrar en mí lo femenino, debido a la alta demanda de trabajar en un ambiente donde la mayoría de los roles son ocupados por hombres (y a veces me mimetizo con este género), he podido observar y ser consciente de ciertos aspectos que quiero compartir con Ustedes en este espacio.

Para que se entienda el punto, quiero aclarar lo siguiente: Cada persona, sea hombre o mujer, tiene una parte femenina y una masculina. Lo femenino y lo masculino poco tienen que ver netamente con el género, sino que se refiere a una construcción social como parte de los símbolos y conocimientos que son implícitamente impuestos desde que nacemos en la sociedad, esto se llama Imaginario Colectivo o Realidad Creada, que separa las características de una persona en estas dos partes:

Características femeninas. Ejemplos:

- Delicadeza
- Servicio
- Dulzura
- Empatía

- Tolerancia
- Cuidados

Características masculinas. Ejemplos:

- Lógica
- Fuerza física
- Brutalidad
- Toma de decisiones
- Racionalidad
- Competitividad

En general, el mundo de la informática tiene mucho de “masculino”, nuestro entorno profesional exige mucha competitividad, tomas drásticas de decisiones, y todas esas características lógicas y mentales citadas bajo la etiqueta de lo masculino. Justamente por eso la parte más emocional queda muy relegada.

Esto da pie a que, por ejemplo, en mi caso y de otras colegas mujeres, nos “masculinizamos” para competir en esta profesión, por eso mencionaba mi búsqueda personal de amigarme con lo femenino.

Mercé Brey (Licenciada en Ciencias Empresariales y Máster en Finanzas Internacionales y Comercio Exterior, Escritora, Conferencante, Experta en Diversidad, trabajó casi 30 años en la industria Bancaria y es ex presidente de la Cámara de Comercio Italiana en Barcelona), en su libro “Alfas & Omegas” menciona algo muy llamativo: que estamos ‘desequilibrados’ porque hay mucho más de lo masculino que de lo femenino en la industria. Falta desbloquear la esencia femenina en el liderazgo. O sea, que los roles de liderazgo, así lo ocupen hombres o mujeres, incluyan no solo la parte masculina, sino lo complementen con su lado emocional, humano, centrado en las personas.

Los hombres fueron obligados desde niños a cumplir con las características masculinas puras, haciendo de menos a las femeninas, y créanme que eso ha causado muchos estragos inconscientes en la sociedad. Les recomiendo realizar una introspección personal, y dejo abierta la pregunta de cómo podemos equilibrar lo femenino con lo masculino en nuestras organizaciones desde nuestros roles, desde nuestro ser.

Las personas hacen evolucionar a las empresas. Si los individuos cambian, las organizaciones se ven forzadas a hacerlo. Si las personas que trabajan en una organización no están en equilibrio, la organización tampoco lo estará.

El ser humano tiene la capacidad de sanar física y emocionalmente después de un trauma o suceso que nos devasta. Esto se llama *Resiliencia*. Esta capacidad nos permite auto repararnos.

Para que esto suceda, primero, debemos estar conscientes de cuál fue el golpe, y cuáles son los síntomas y consecuencias de este.

MEJORES PRÁCTICAS

Ahora que creo que estamos un poco más conscientes, o al menos mejor informados, sobre aspectos negativos resaltantes del excesivo estado de estrés laboral al cual nos sometemos, quiero dejarles unas “Best Practices” para mejorar nuestra calidad de vida desde lo corporal, lo emocional y lo cognitivo.

Desde lo corporal, lo básico... tomar mucha agua. La comunicación neuronal funciona mejor con un cerebro hidratado. Si no, se pone lenta.

Dormir bien dentro de lo posible. Nuestro cerebro y corazón no paran, pero en la noche el cerebro se limpia. Es importante para anclar el aprendizaje. Y regulamos mejor las emociones, limpia los recuerdos.

Moverse. Al cerebro le encanta el ejercicio. Desde una caminata, o hacer deportes que a uno le guste. Lo importante es que haya un disfrute de por medio. No por obligación, de lo contrario se convierte en otro factor de estrés.

Estirarse y respirar entre trabajos, cada 30 minutos. El cerebro necesita oxigenarse para mejorar la transmisión neuronal.

Comida de verdad para nutrir el cerebro, por esto del consumo de energía. Grasas saludables, vegetales de estación, vitamina B12, chocolate > 70% cacao. No digo que dejen de comer comida chatarra, pero al menos en momentos donde sientan que están bajo mucha presión laboral o de estudio, tomar estos recaudos. El café secreta adrenalina, ojo durante el estrés.

Copa de vino ocasional. Me los imagino riéndose de este punto, sabemos que no será ni una sola copa ni ocasional... pero vamos, es una “mejor práctica”, y no siempre se cumple al pie de la letra ¿no?

Esto de que necesitamos algo dulce artificialmente (golosinas) para pensar mejor o para tener energía es un mito. Los productos azucarados artificialmente son una de las causas del Brain Fog, que es la limitación de la agudeza cerebral, dificultad para pensar, para concentrarse. No consumas azúcar antes de un momento cerebral importante. Lo mismo pasa con el aspartamo.

Evita en lo posible fumar tabaco, sobre todo en momentos de trabajo intensivo. Después de consumir un químico adictivo que te gusta, te relajas, cierto. Pero con el tabaco, pierdes la oxigenación del cerebro.

Desde lo emocional, lo primero es la Relajación. Algo tan simple como la respiración consciente y la meditación (mindfulness).

Esto traducido a lo cotidiano sería poder darse un rato para simplemente parar y enfocarte en la respiración unos minutos, inspirar y expirar profundamente, tratando de no hacer caso a los pensamientos que aparezcan, solo dejarlos pasar, sin tener que tomar decisiones. Ser conscientes por un momento de tu cuerpo, de tu humanidad.

Si te animas, practica yoga, taichi, o simplemente escucha la música que te gusta. También es recomendable hacer manualidades, ayudan a enfocar la mente hacia el cuerpo, hacia los sentidos, a desconectar el cable un rato del trabajo, las preocupaciones, y estar conectado con algo que te gusta. En algunas empresas de tecnología empezaron a crear el cargo de

“Chief Mindfulness Officer” o CMO para enseñar y compartir acerca de mindfulness. Por ejemplo, Google, IBM, SAP.

Mantener contacto social, presencial o virtual. Sobre todo, si se vive solo. Somos gregarios, trabajamos en tribus. Aunque algunos seamos ermitaños, necesitamos algunos lazos. Además, el contacto social es un protector contra el Alzheimer.

Darse tiempo para platicar con amistades. La amistad nos brinda un apoyo emocional, fortalece la autoestima y la seguridad en uno mismo. El contar con distintos círculos de amigos nos da la oportunidad de tener diversos puntos de vista sobre temas laborales, familiares, de pareja, personales, entre otros.

Busca crear emociones positivas. Todas las emociones tienen un objetivo de supervivencia. Las bromas, los memes, los comics, libros, distracciones que fomenten tu interés y a la vez creen emociones positivas en ti.

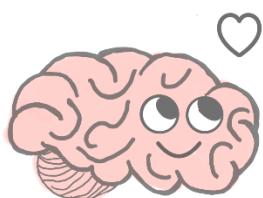
Fomenta la curiosidad buscando cosas que te interesen aparte del trabajo.

Capacidad de cuidado, preocupados del resto, capacidad emocional para cuidar al otro y alejarnos de nuestro propio estrés. Aquí tiene mucho peso el lado femenino (recalco, no importa si eres hombre o mujer, tienes un lado femenino y masculino dentro).

Ponte un horario de trabajo y trata de respetarlo. Sé que habrá excepciones, pero que sean solo eso, excepciones.

Cuida los espacios familiares. Reserva tiempos de calidad con la familia para poder estar con energía para participar en ellos. También implica tiempo con la pareja donde se tengan conversaciones sobre metas personales, familiares, sueños y expectativas.

Brindarse tiempo personal es muy importante.



Desde lo cognitivo, aprender a coordinar roles y ¡PEDIR AYUDA! Delegar funciones.

Los superhéroes dejémoslos para los comics. Las personas que saben delegar funciones se dan la oportunidad de confiar en otros, trabajan en equipo y reciben soluciones inesperadas que los liberan de presiones.

Realiza pausas para recuperar la energía. 2 minutos al menos ya refresca al cerebro.

Mantener el cerebro activo aprendiendo cosas que les interese. ¡Buenas noticias para los Gamers! Jugar ayuda a desconectar del trabajo y del estrés. Y si son juegos de equipo, mejor. También la gamificación en el trabajo ayuda mucho con la motivación y el compromiso, sobre todo este mensaje va para los que son líderes de equipo.

Si dejas de hacer cosas cognitivas, el cerebro no va a desestresarse, porque no fomenta nuevas conexiones neuronales. Busquemos tener un cerebro Fit, o sea, entrenarlo constantemente. Si el trabajo es muy mental e intenso, mejor leer algo liviano o ver una serie, pero mantener al cerebro ocupado, pero sin estrés.

Mentalidad de crecimiento. Enfocarse en metas, aprendizajes y soluciones. El cerebro aprende rápido y acepta los desafíos.

Lo que a mí me estresa no necesariamente estresa a otro. Cada uno debe revisar qué específicamente le estresa. Esto tiene que ver mucho con la historia de cada uno, incluso desde la infancia. Si el estrés es permanente y no te permite pensar, no te deja seguir adelante, te recomiendo ver un Coach o incluso un psicólogo. Nunca está demás la ayuda de un profesional si está dentro de tus posibilidades económicas.

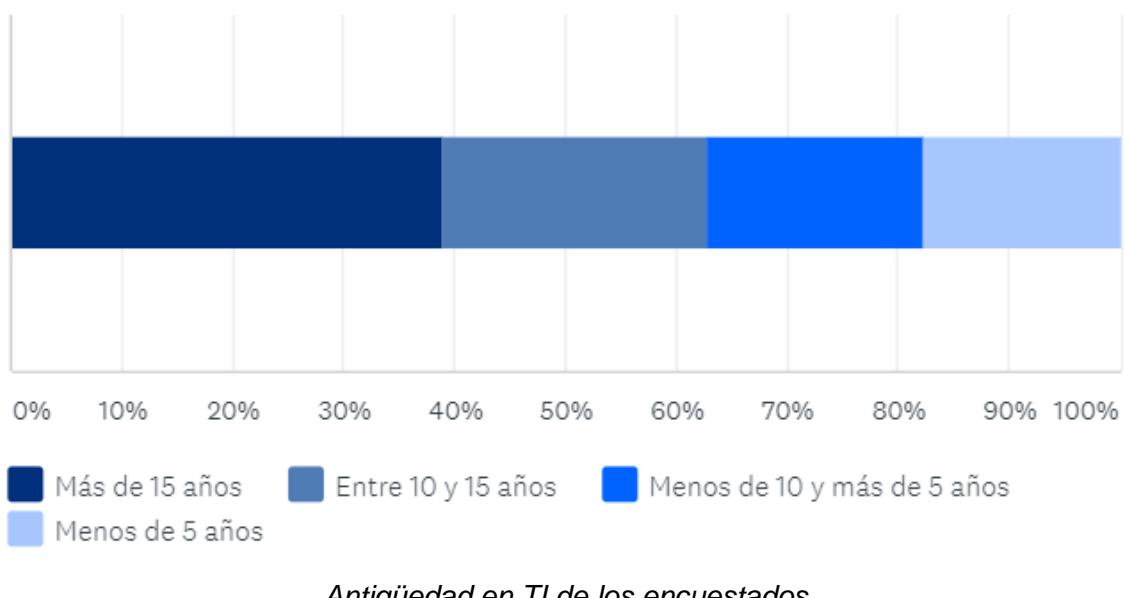
Capacitarse. Un personal capacitado facilita la retroalimentación y el enriquecimiento de las perspectivas del negocio. Además, te permite estar un poco más preparado para manejar los problemas técnicos que surjan.

Mira y escucha con atención a tus compañeros de trabajo cuando comentan algún punto, esto es esencial para que las labores cotidianas sean resueltas de la mejor forma posible. Pedir consejo o apoyo abre otras perspectivas que quizás no habíamos considerado.

MIREMOS ALGUNOS NÚMEROS

Realicé una encuesta a profesionales de TI de varios países de Latinoamérica. En total, 113 encuestado/as.

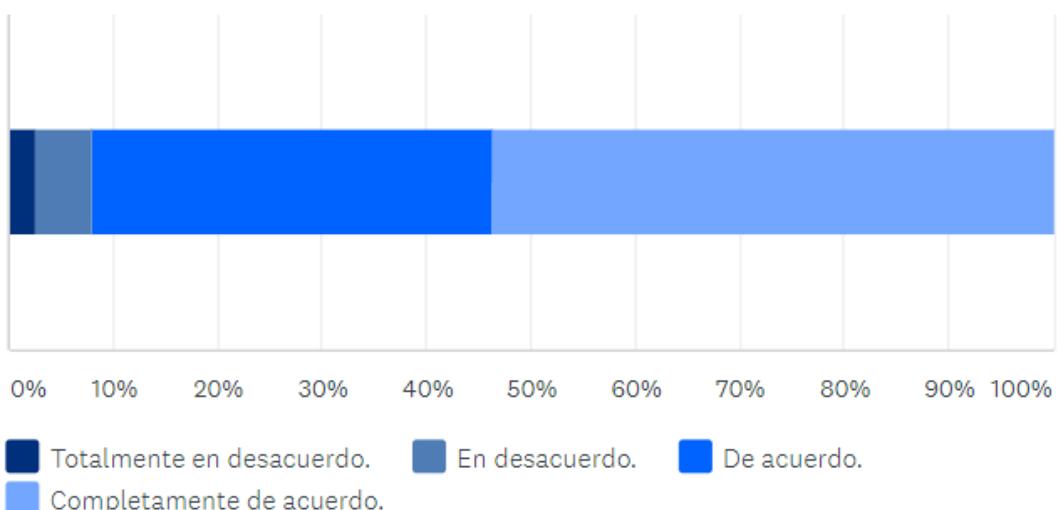
Participaron 87 hombres, 25 mujeres, 1 otro, de Chile, Paraguay, Argentina, Uruguay, Perú, Colombia, España, México, Latinos en USA



soluciones servicios Administrador de sistemas Responsable Proyectos de negocios y Arquitecto Técnico para Ingeniero Engineer Analista manager de los TI de sistemas Consultor Soporte Jefe de equipo Desarrollador Jefe de software Especialista

Esta es la nube de palabras de los puestos que ocupan los encuestados
Recopilé los resultados más relevantes, los cuáles les comarto a continuación.

- El 70.8% del puesto de los profesionales requiere que desarrollen nuevas habilidades.
- El 93.81% de los puestos necesita un nivel elevado de especialización o calificación.
- El 92.04% necesita de la creatividad en la resolución de problemas y realización de tareas.
- Sólo el 23% contestó que tiene una tarea muy repetitiva día tras día.
- 78.76% opinan que tienen la posibilidad de desarrollar sus habilidades personales en sus trabajos.
- El 70% tiene influencia sobre las decisiones en sus labores y cómo hacerlas.
- 88.4% concuerda que sus labores exigen ir muy deprisa. Presión.



91.96% dicen que deben realizar mucho esfuerzo mental en sus labores.

- En el punto donde pregunto si tienen el tiempo suficiente para realizar sus tareas, hubo solo una diferencia del 2,5%, siendo el valor mayor el que NO tienen el tiempo suficiente para terminar sus tareas, o sea, horas extras a la orden.
- Un punto curioso de la encuesta es que el 56.64% recibe peticiones contradictorias de distintas personas de la misma organización acerca de las tareas que debe realizar.
- 88.49% de las personas deben concentrarse durante largos periodos de tiempo para realizar su trabajo. Esto diariamente. Y el 82.14% es interrumpida a menudo, así tiene que nuevamente punto para las horas...
- El 83.28% dice que sus jefes facilitan la realización de las tareas. ¡Maravilloso!
- El 96.46% trabaja con gente amigable. ¡Esto ayuda a fomentar la buena salud emocional y mental!
- 54.86% de los encuestados realiza desde 3 horas extras en adelante.
- Desde que trabajan en tecnología, el 53.10% ha notado algunas veces falta de motivación o angustia, mientras que el 37.17% trabaja con angustia y desmotivación constante. Benditos 9.73% que dicen nunca haber sentido estas emociones negativas.
- En cuanto a calidad y cantidad de alimentación, el 74.14% dice que empeoró, y un afortunado 4% dice que mejoró.
- En cuanto a actividad física, sólo el 30.09% realiza ejercicios o deportes a menudo; el 58.41% poco ejercicio; y los que prefieren estar acostados cada vez que pueden son 11.50%... debo confesar que este último grupo son de mi total simpatía.
- En cuanto a relaciones de pareja, el 50.89% ha recibido reclamos de sus parejas por falta de tiempo y de energía; 39.29% no ha tenido problemas; 8.93% no tiene parejas; y un afortunado o afortunada dijo que ha mejorado. ¡Felicitaciones! Sin embargo, es bastante preocupante la cantidad de personas con problemas de pareja desde que decidió meterse en el campo de TI.
- Al 52.21% le cuesta un poco empezar el día laboral en cuanto al ánimo, pero una vez que se enchufa, lo logra. Sólo el 28.32 se levanta totalmente motivado.
- El 59.29% ha tenido alguna que otra enfermedad durante el periodo de trabajo en TI.
- El 61.95% trabaja en TI porque realmente le gusta y quiere seguir creciendo en su profesión. Mientras un 25.66% quiere cambiar de rumbo.
- El 68.14% se ha sentido frustrado algunas veces en lo laboral.

MIS CONCLUSIONES SOBRE LA ENCUESTA

A raíz de los resultados, tomando en cuenta el conjunto de personas encuestadas, quiero destacar cuanto sigue.

La exigencia en cuanto a capacidades y habilidades técnicas es elevada. Nuestro entorno nos exige una continua capacitación, así como tener habilidades para resolver problemas o reaccionar rápidamente a eventos no contemplados. Sumado a esto, la mayoría de las tareas que realizamos son diversas, exigen un prolongado trabajo mental enfocado y ágil, lo que nos desafía aún más a aprender nuevas habilidades diariamente.

Las horas extras son una realidad que seguirán existiendo, debido a la naturaleza de nuestro trabajo y a las continuas interrupciones que aparecen durante la jornada laboral. Las famosas "contingencias". Es muy probable que todos tengamos cuadros de estrés elevados y más de uno haya caído en el burn out.

El trabajo ha traído consecuencias inevitables en la vida personal de cada individuo encuestado. Desde la salud hasta en lo relacional.

Los cargos de jefatura han tenido muy buena disposición en la facilitación de las tareas. Esto da una luz de esperanza de que pueden ir surgiendo más líderes y menos jefes.

QUIERO DEJARLES UN PENSAMIENTO FINAL MUY DESDE LO PERSONAL

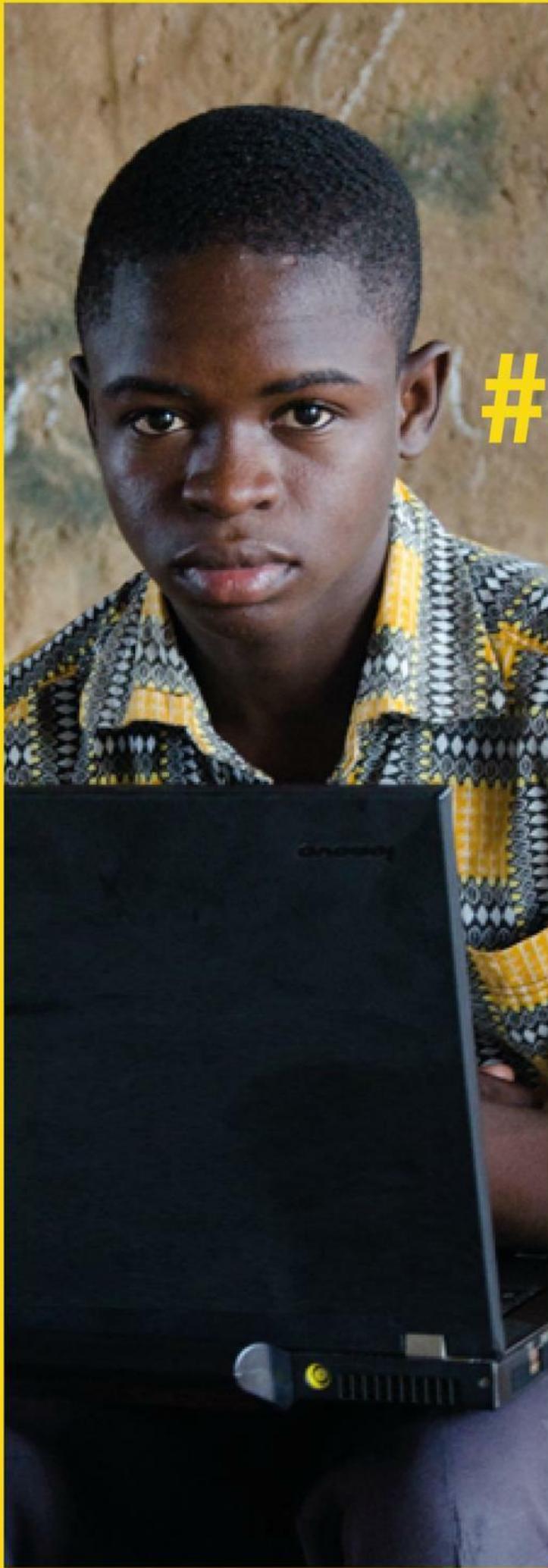
La frustración y la desmotivación están a la orden del día. Como seres humanos parte de una sociedad que mide a las personas según su éxito, su estatus, sus títulos y su capacidad de realizar tareas sin errores, hemos crecido con la presión inserta en el inconsciente, lo que nos genera un estrés inevitable desde el principio de nuestra vida escolar y laboral.

Algunos hemos aprendido poco o nada a poner límites, aprendemos por las malas a separar la vida personal de lo laboral. Si bien es cierto que nos consume el día a día, también hay una cuota de responsabilidad emocional y personal a la que no estamos acostumbrados a mirar.

Necesitamos aprender a deconstruir estas creencias y aprender a valorar las virtudes que hemos cultivado, de manera a no ser sólo números, sino personas completas con inteligencia emocional, capaces de modificar nuestro entorno laboral, centrándolo en las personas, y no sólo en los objetivos. Empatía, queridas personas, el saber ponerse en el lugar del otro.

Dejemos de celebrar el trabajo sin descanso, el no cometer errores, el perfeccionismo y el estar siempre ocupados.

Personas felices generan organizaciones felices.



#CAMBIA LA HISTORIA

**"Si me alimentas
la barriga, sacias
mi hambre un
día, pero si
alimentas mi
mente, lo haces
para más de
cien años"**

NASCO
FEEDING
MINDS