Monitorización de Entornos Virtuales desde cero

Nimbiformes, cúmuliformes, estratiformes y cirriformes... No, no hablaremos de nubes como tal sino de la "computación en la Nube". Hoy hablo sobre los Entornos Virtuales y su enfoque hacia la monitorización, ¡apuntaros en la clase 101!

Los Entornos Virtuales son un modelo para permitir el acceso a la red de forma ubicua, conveniente y bajo demanda a un grupo compartido de recursos informáticos que son de fácil configuración (por ejemplo: almacenamiento, redes, servidores, aplicaciones y servicios) y que pueden aprovisionarse y liberarse rápidamente con un mínimo esfuerzo de gestión o interacción del proveedor de servicios. Se le denomina Nube debido a que en los gráficos y esquematizaciones se le dibuja como tal, de allí el mote.

Internet esquematizado como Nube por eHorus

(imagen cortesía de https://www.ehorus.com/)

Dichos Entornos Virtuales no están exentos de una monitorización como cualquier otro sistema informático, lo que conlleva a que nos preguntemos **¿qué hay más allá de la Nube, cómo funcionan en realidad?** Para ello debemos comenzar desde cero...

¿Es un ave? ¿Es un avión? ¡No, es un hipervisor nativo!

Tipos de hipervisores (Wikipedia https://en.wikipedia.org/wiki/File:Hyperviseur.png)

Los hipervisores nativos, tipo I, interactúan directo con el hardware (ver gráfico anterior). Algunos de ellos son:

- AWS Nitro®.
- bhyve® ("BSD hypervisor").
- IBM z/VM®.
- Kernel-based Virtual Machine (KVM).
- Microsoft Hyper-V®.
- Nutanix AHV®.
- Oracle VM Server for SPARC®.
- Oracle VM Server for x86®.
- VMware EXSi®.
- Xen®.

Los hipervisores hospedados (tipo II) se diferencian porque necesitan de un sistema operativo para su funcionamiento:

- bhyve® ("BSD hypervisor").
- KVM.
- Parallels Desktop for Mac®.
- VMware Player®.
- VMware Workstation®.
- VirtualBox®.

No me he equivocado al repetir a KVM y bhyve® en ambas listas ya que en realidad son módulos del kernel Linux que se comportan como hospedado y como nativo, según el caso. Incluso voy más allá y pinto el panorama con un poco de complejidad: AWS Nitro® es un hipervisor ligero tipo uno basado en KVM... El escenario más común es correr un sistema operativo GNU/Linux con un hipervisor hospedado.

También con la potencia de los procesadores modernos y chips especializados (por ejemplo Intel VT® o AMD-V®) podemos ejecutar máquinas virtuales dentro de máquinas virtuales (anidadas).

Veamos con mayor detalle a KVM, para tener una noción de cómo funciona un hipervisor:

[SVG: Kernel-based Virtual Machine KVM (Wikipedia https://commons.wikimedia.org/wiki/File:Kernel-based_Virtual_Machine.svg)]

Kernel-based Virtual Machine KVM (Wikipedia https://commons.wikimedia.org/wiki/File:Kernel-

based_Virtual_Machine.svg)

En el caso de KVM, este tiene altos niveles de seguridad tales como Control de Acceso Obligatorio (en inglés *Mandatory Access Control* o MAC) y SELinux.

Una vez tengamos esto en mente, que lo he simplificado al extremo, demos el siguiente paso: ¿cómo manejamos dos o más servidores que albergarán nuestros Entornos Virtuales?

¿Cuándo veremos la monitorización? *Paciencia.*

Metal as a Service (MAAS)

MAAS, así, todo en mayúsculas (para diferenciarlo de las **MaaS**: *Mobility as a Service* y *Monitoring as a Service* -sí, la jerga en informática alcanza límites insospechados-) es un componente clave de la *convergencia*.

MAAS tiene bajo su mando gran cantidad de ordenadores apilados en un centro de datos, debidamente conectados en cuanto a potencia eléctrica y comunicación se refiere, y listos para ser "despertados" por medio de sus tarjetas de red vía concentradores y/o enrutadores. Echan mano de tecnologías tales como:

- PXE para arrancar el equipo.
- DHCP para obtener una dirección IP previamente planificada por IPAM.
- DNS's de la propia organización y que trabajan de manera muy dinámica y en armonía con los anteriores.

Todo esto permite una instalación super rápida desde cero de cualquier sistema operativo: Windows®, CentOS®, RHEL®, Ubuntu®, etcétera. MAAS trabaja con las imágenes personalizadas y las aplicaciones preinstaladas de su elección, y luego hace configuraciones de discos y de red adicionales a las que ya tiene . Para la gestión de usuarios MAAS tiene autentificación por medio de **Protocolo de Acceso Ligero a Directorios** (*Lightweight Directory Access protocol* o LDAP) y **Control de Acceso Basado en Roles** (*Role-Based Access Control* o RBAC).

Para el resto de aprovisionamiento de software que no esté incluido en las imágenes de instalación de cada sistema operativo, emplea alguno de los que hemos presentado en este vuestro blog con anterioridad: Ansible®, Chef®, Puppet®, SALT® o incluso Juju® (de Canonical®, empresa que desarrolla Ubuntu®). No menos importante son las actualizaciones de software, que

se descargan una sola vez a un nodo central y se reparten a las decenas o centenas de máquinas reales -y virtuales- que tengamos a nuestro cargo.

Hiperconvergencia

Concepto de reciente adopción que consiste en la convergencia y los hipervisores (de allí su nombre): considero que no puede estar completa sin la presencia de las herramientas de monitorización. Por acá hemos publicado, por ejemplo, el enfoque de monitorización orientado a Nutanix y VMware.

Esquema de arquitectura de Pandora FMS para monitorizar VMware

También Pandora FMS tiene un capítulo completo de su manual de uso enfocado a la monitorización de Entornos Virtuales.

Monitorización de Entornos virtuales

No me canso de repetirlo: la efe en el nombre de Pandora FMS viene por flexibilidad *¡pero hasta ahora explico por qué y de dónde proviene esa flexibilidad!*

Pandora FMS se ha labrado su propio camino, fue desarrollado desde cero teniendo muy en cuenta -y agregando- los buenos conceptos y hábitos de la monitorización. **No depende de terceros para obtener sus datos**, como todo buen periodista de sucesos se llega hasta la fuente, hasta el origen de los eventos. Para saber lo que aconteció debemos preguntarlo a sus protagonistas, de primera mano, es por ello que Pandora FMS se apoya:

- En consultas SNMP el cual tiene su uso más que todo en artefactos que controla el tráfico de red: enrutadores, concentradores, módem, etc. Pero no os quedéis allí nada más, solo preguntando, también que nos retribuyan datos cuando suceda un evento en particular (alertas).
- Gracias a la monitorización de red avanzada de Pandora FMS podemos ir un paso más allá en el comportamiento de los dispositivos del punto anterior.
- Por medio de **Interfaz de Programación de Aplicaciones** (Application Programming Interface o API) de los hipervisores nativos u hospedados, por ejemplo para Red Hat Enterprise Virtualization (RHEV) y el **RHEV Monitoring Plugin** de Pandora FMS.
- Aunque abarca mucho más allá de la monitorización, en OpenNebula con formato JSON y con la Transferencia de Estado de Representación (**Representational State Transfer** o REST) podemos invocar la OpenFlow API y el lenguaje Perl para consultar y obtener métricas para el hipervisor KVM. OpenNebula utiliza también otros hipervisores como VMware o Xen, así que el abanico es amplio y flexible.
- Con el Juego de Herramientas de Desarrollo de Software (*Software Development Kit* o SDK), como es el caso de VMware, que permite desplegar en la consola de Pandora FMS a *VMware View* y *VMware Manager*. Con **VMware View** podrá examinar toda la arquitectura VMware de un solo vistazo y gracias a **VMware Manager** podrá gestionar máquinas virtuales en un solo lugar.
- En el caso de **Docker Swarm** pues Pandora FMS corre en modo virtual para entrar y formar parte del entorno y obtener las métricas necesarias; a su vez podremos unificar en una metaconsola (versión Enterprise) con otros servidores de monitorización.
- Todos los métodos anteriores que menciono **no representan carga alguna para los ambientes que vamos a monitorizar** pues fueron desarrolladas por los propios fabricantes y/o programadores en su funcionamiento normal común y corriente: nuestras consultas tienen un impacto minúsculo en su marcha diaria.
- Además de utilizar la API propia y original de los Entornos Virtuales también podemos combinar con el uso de Agentes Software de Pandora FMS, como es el caso de **Amazon EC2 Cloud**. Fijaros muy bien que aquí sí, agregando nuestros pequeños Agentes Software, cargaremos un trabajo adicional pero bien vale la inversión porque obtendremos muchos datos avanzados de los sistemas operativos o cualesquiera de sus aplicaciones, por complejas que sean. Recordad lo de la flexibilidad: el equipo de desarrollo de Pandora FMS Enterprise siempre está en espera de ayudaros y agregar nueva funcionalidades a

vuestra red.

Además existen más de una docena de razones para utilizar Pandora FMS Enterprise, aparte de los que he mencionado en el campo de los Entornos Virtuales *(Virtual server monitoring)*

Ya para finalizar hago un reconocimiento público al equipo de Pandora FMS y a los colegas de la comunidad, ya que han publicado acá en el blog los artículos cuyos enlaces web han permitido que este tema tan denso quede escrito de manera tan concisa. Muchas gracias.
