Sistema de Gobierno de Federaciones de CPD

Fundación COMPUTAEX
David Cortés-Polo, Jesús Calle-Cancho, Alfonso López-Rourich y José-Luis González-Sánchez
CénitS - Centro Extremeño de iNvestigación, Innovación Tecnológica y Supercomputación
Cáceres, Extremadura, España
{david.cortes, jesus.calle, alfonso.lopez, joseluis.gonzalez}@cenits.es

***Resumen*—Debido a la creciente demanda de potencia computacional y, por consiguiente, de suministro eléctrico llevada a cabo por los Centros de Proceso de Datos (CPD), acompañado del intento de optimización de todo lo relacionado con el sistema de climatización requerido, surge la necesidad de maximizar la eficiencia de los equipos informáticos y de clima, minimizando el consumo energético de todos ellos.**

En este sentido, *Sistema de Gobierno de Federaciones de CPD* es uno de los dos proyectos enmarcados en CENITAL-2016, que propone obtener en los CPD el punto de equilibrio que maximice la eficacia de los equipos informáticos, eléctricos y climáticos; en línea con la eficiencia energética y la eficacia computacional de los *data center*. En este proyecto se ha conseguido dotar al CPD de CénitS con un Sistema de Gobierno que permite administrar simultáneamente el comportamiento de los equipos de los cuatro sistemas más importantes de un *data center* como son el sistema informático, el sistema telemático, el sistema de climatización y el sistema eléctrico, logrando un mejor desempeño desde el punto de vista computacional, energético y económico.

***Palabras clave*—** CPD, sistema de climatización, sistema eléctrico y sistema computacional

I. INTRODUCCIÓN

En las últimas décadas los procesos de deslocalización e internacionalización de las grandes empresas, unidos a la eclosión del uso de la Tecnologías de Información y las Comunicaciones (TIC) y al procesamiento de datos masivos, han hecho que las necesidades de cómputo hayan crecido a un ritmo superior al que lo hacía la capacidad de cálculo de los ordenadores personales. Por este motivo, y para satisfacer las necesidades de los sistemas de computación más exigentes, se ha producido una interesante evolución de las arquitecturas de computadores.

El tratamiento de datos masivo se lleva a cabo en instalaciones muy específicas, llamadas Centro de Proceso de Datos (CPD), donde se concentra una gran cantidad de recursos físicos, lógicos y humanos. Dichos recursos consisten en unas dependencias debidamente adaptadas para mantener en ellas una gran cantidad de equipamiento electrónico, equipos de cómputo y redes de comunicaciones, que tienen que cumplir rigurosas condiciones ambientales, energéticas y de seguridad; todo ello organizado y controlado por un equipo humano experto en las tecnologías utilizadas.

La gran densidad de equipos informáticos que existen en un CPD hace que el consumo eléctrico se incremente de forma considerable, al mismo tiempo que dichos equipos desprenden gran cantidad de calor de forma permanente, que debe disiparse al exterior para garantizar unas condiciones ambientales estables y adecuadas para el correcto funcionamiento de todos los equipos. Por lo tanto, es necesario contar en las instalaciones con sistemas de climatización para mantener la temperatura en los niveles adecuados de trabajo. Tal y como puede observarse en la Figura 1, resulta interesante mencionar que el coste energético de los sistemas de refrigeración es similar al coste de la energía consumida por el equipamiento informático del CPD, por lo que son los dos aspectos claves que tendrán que ser considerados.

Para optimizar la efectividad del uso de la energía se recurre a técnicas y mecanismos que monitorizan las instalaciones de forma muy precisa. Las prácticas más extendidas son las relacionadas con el confinamiento en pasillos fríos y calientes; la implantación de dispositivos de monitorización en los servidores y en el suministro eléctrico de los componentes de la infraestructura física; la reconfiguración de las salas de cómputo en áreas más pequeñas; el enfriamiento líquido o técnicas de *free-cooling*.

Los diseños más modernos de CPD se basan en el concepto de modularidad, adaptando la infraestructura a las necesidades de la sala de TI (Tecnología de la Información), es decir, la sala sólo se equipa con lo absolutamente indispensable: sistemas de suministros de energía de menor capacidad para un área definida, enfriamiento por el método de confinamiento, y enfoques de crecimiento cúbico, es decir, no sólo se amplía a lo largo y a lo ancho, sino también hacia arriba en racks de doble altura. Esta modularidad, además de crear una mayor eficiencia energética, ya que sólo se usa la capacidad requerida por los equipos de TI, ocasiona una liberación importante en las

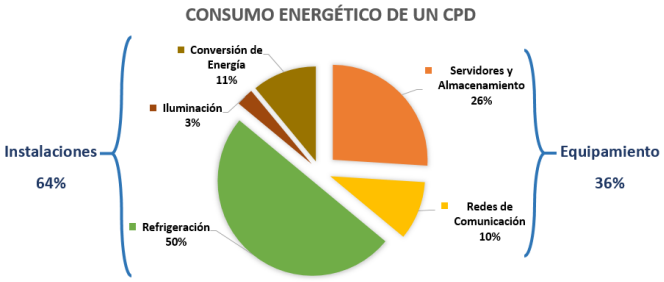

Fig. 1: Consumo energético en un Centro de Proceso de Datos.

inversiones de infraestructura, las cuales se pueden planificar en base a los crecimientos de TI y de los espacios ocupados en el CPD. Más allá de los beneficios expuestos anteriormente, se persigue tener la capacidad de crecer con las demandas futuras, por lo que si se lleva a cabo la instalación de nuevos equipos con mayor densidad, la infraestructura crecerá en base a esas demandas específicas.

Por otro lado, desde el punto de vista computacional, con la aparición de la computación en la nube o *cloud computing* se ha proporcionado a la industria TIC una infraestructura flexible capaz de ejecutar aplicaciones de computación de alto rendimiento (High Performance Computing, HPC), además de incorporar la capacidad de gestionar los recursos computacionales de la manera más eficiente posible. Por lo tanto, el concepto de *cloud computing* se ha consolidado como un nuevo paradigma tecnológico capaz de permitir el acceso ubicuo, adaptado y bajo demanda en red a un conjunto compartido de recursos de computación configurables compartidos, que pueden ser aprovisionados y liberados con un esfuerzo de gestión reducido o interacción mínima con el proveedor de servicio [1].

II. ANÁLISIS DE UN SISTEMA DE GOBIERNO PARA FEDERACIONES DE CPD

Hoy en día se manifiesta una tendencia ascendente del mercado de las TIC y las soluciones ofertadas de *cloud computing*. A este efecto, se le suma el fenómeno de Internet de las Cosas (IoT), por lo que una gran cantidad de dispositivos interconectados entre sí generan datos de manera exponencial. Por ello, para afrontar este desafío no se puede aplicar el concepto de fuerza bruta: a mayor cantidad de datos, mayor crecimiento de los equipos del CPD. En este sentido, hay que pensar que, a mayor cantidad, complejidad y frecuencia de los datos, más inteligente debe ser el enfoque para resolverlo.

Por lo tanto, debido a que en muchas ocasiones los CPD poseen máquinas infrautilizadas, en este estudio se ha desarrollado una solución de CPD federados que permita compartir la carga de trabajo de los distintos centros de datos, y así reducir considerablemente el consumo energético, afrontando el reto que se propone con el crecimiento de flujo de datos.

Una federación de CPD se basa en un conjunto de prácticas que interconectan dos o más centros de datos (o proveedores de servicios), donde los recursos informáticos se convierten en una extensión temporal o permanente de la unión de CPD, según el acuerdo establecido entre los participantes de la agrupación, buscando el propósito de equilibrar la carga de tráfico de datos y operaciones, acomodando de esta forma picos de demanda que pudieran surgir en un momento determinado [2].

Para la implementación de esta solución se ha optado por la utilización de una plataforma de gestión *cloud* ampliamente conocida como es OpenNebula [3], ya que resulta ser la que mejor se adapta a las necesidades del presente estudio, debido a su capacidad de virtualización de los activos del Centro de Datos y a su gran flexibilidad a la hora de implementar un gestor para una federación de CPD.

El escenario típico para una federación gestionada mediante OpenNebula es un organismo con varios CPD, distribuidos en diferentes ubicaciones geográficas. Esta integración no se basa únicamente en la API (Application Programming Interface) de OpenNebula, sino también mediante la sinergia de los administradores de sistemas de todos los Centros de Datos que colaboran en el mantenimiento de la infraestructura conjunta, formando una federación de administradores. Por lo tanto, una federación OpenNebula es una unificación de estas estructuras rígidas donde cada instancia de la federación, denominada *zona*, trabajará conjuntamente con las demás, para crear una topología de gobierno compuesta por un nodo maestro que orquestará todos los activos disponibles de varios centros de datos distribuidos, y actuando estos últimos como nodos esclavos del orquestador de la federación. El nodo maestro de OpenNebula es el único que tiene permiso para escribir en las tablas de bases de datos compartidas, mientras que los esclavos mantienen una copia local de sólo lectura, y un *proxy* para remitir al maestro cualquier acción importante. De este modo, se permite garantizar la consistencia de los datos de las tablas compartidas, sin ningún impacto en la velocidad de las acciones de lectura. La sincronización de los datos compartidos se logra configurando MySQL para replicar sólo ciertas tablas, pero hay que tener en cuenta que la replicación no suplanta los *backups*, sino que simplemente garantiza la operatividad del sistema, ya que la replicación MySQL soporta un esquema de repetición de los datos asíncrono de un servidor maestro a uno o varios servidores esclavos.

III. GESTIÓN INTELIGENTE DE UN CPD Y SU APLICABILIDAD A UNA FEDERACIÓN

La principal motivación de este trabajo es la gestión inteligente de un Centro de Proceso de Datos desde un punto de vista estructural, comenzando con el despliegue de servicios de gestión inteligente para un CPD y finalizando con el despliegue de una federación de CPD basado en un ecosistema abierto [4].

La necesidad de desplegar una federación de CPD viene dada por la creciente demanda en el ámbito de la computación empresarial donde se dispone de varios CPD realizando tareas de respaldo y que se deben gestionar de forma eficiente para maximizar la eficiencia de cómputo y reducir la huella de carbono generada por los equipos.

Para la primera fase de despliegue del proyecto se ha decidido implantar el sistema de gobierno en un subconjunto de la infraestructura de CénitS-COMPUTAEX, usando, por tanto, parte de la infraestructura:

- 3 Servidores: HP ProLiant BL465c Gen8 con dos procesadores AMD Opteron 6376 HE (2,3 GHz/16-core/16 MB), 128 GB de memoria RAM y dos discos duros SAS de 300 GB por servidor.
- 2 Servidores: HP ProLiant BL460c Gen6 con dos procesadores Intel Xeon E5540 (2,53 GHz/4-core/8 MB), 24GB de memoria RAM y dos discos duros SAS de 146 GB por servidor.

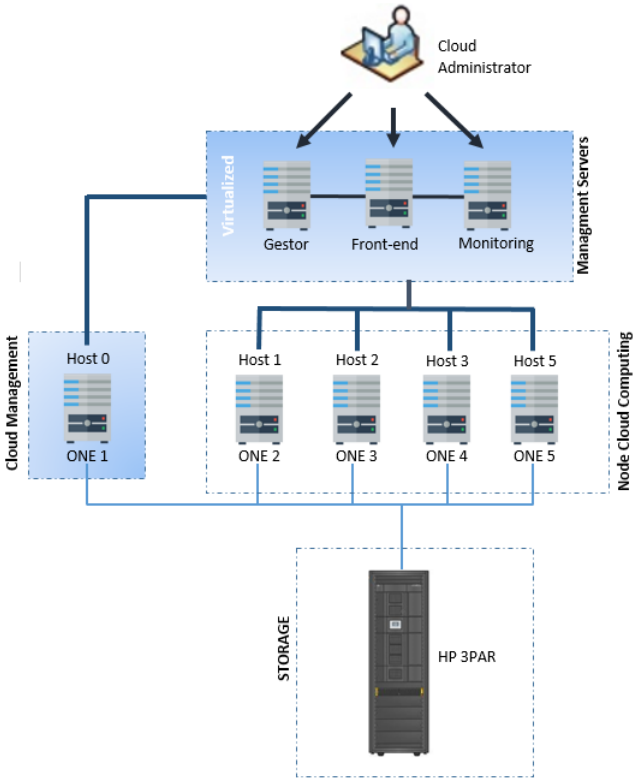

Fig. 2: Estructura de OpenNebula en CénitS

- 1 servidor de almacenamiento para el Datastore de OpenNebula: HP 3PAR StoreServ 7200 x (14 discos SAS 10k x 900 GB) = 11,5 TB de almacenamiento

La Figura 2 muestra la estructura de OpenNebula desplegada en CénitS. En la zona de producción estarían ubicados los *host* físicos que albergan las Máquinas Virtuales (MV) en funcionamiento y sus redes virtuales, estando conectados todos ellos a través del servicio de red a los datastores del sistema de almacenamiento de la cabina de almacenamiento HP 3PAR. En estos datastores es donde se guarda la información relativa al sistema de OpenNebula y las imágenes de las MV apagadas entre otra información relevante.

Por otro lado, la infraestructura requiere de un *host* que realice la función de administración de la federación. Este nodo gestiona el servicio de *front-end*, portal de acceso y monitorización de la nube, mediante el cual se gestionará OpenNebula desplegando nuevas máquinas o modificando algunos de los aspectos de la configuración de cualquiera de las instancias o de toda la federación.

La herramienta desarrollada en este trabajo tiene un carácter cíclico, pensada para que cada cierto tiempo se repita y se reorganicen las máquinas virtuales entre los activos más eficientes energéticamente, disponibles dentro de la federación.

La Figura 3 describe las principales tareas de ejecución de la herramienta diseñada, así como las dependencias de las mismas, para la obtención de los datos y la comprobación de la integridad de los recursos.

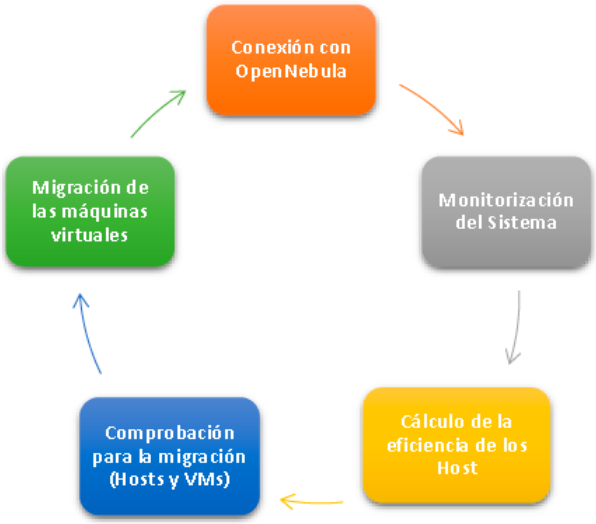

Fig. 3: Módulos de trabajo del gestor de la federación.

Entre las funciones que realiza cabe destacar el evento de migración de máquinas virtuales, que es el principal proceso que se ha desarrollado en este proyecto.

Tras la conexión con OpenNebula a través de la API de desarrollo que proporciona, se obtiene la información del estado de todos los nodos físicos que conforman la federación, así como las máquinas virtuales que ejecutan.

Una vez conocida la información referente al estado de los recursos y los requerimientos de cada MV, se deben ejecutar diferentes algoritmos para la adecuación del clúster a la mejor organización, teniendo en cuenta parámetros como el consumo energético. Para ello, se ha analizado el consumo energético de los equipos usados para realizar el piloto. La Figura 4 muestra un análisis de los consumos por core usados en este trabajo.

El sistema de gestión obtendrá un vector ordenado en el que aquellos servidores con menos recursos reservados por las MV y menor eficiencia energética. Tras esta ordenación, se realizarán los procesos de migración de los recursos que se encuentran en dichas máquinas y se destinarán hacia aquellas máquinas con mayor uso y eficiencia energética.

Una vez que ha concluido la migración de las máquinas virtuales, se vuelve a realizar la monitorización del sistema realizando un apagado ordenado de los *host* que han liberado todos sus recursos en el proceso de migración.

ID_HOST	Nombre	Procesador	Eficiencia Energética	POSICIÓN RELATIVA
2	ONE 3	AMD Opteron 6376	7.18 w/core	Primeras posiciones
3	ONE 4	AMD Opteron 6376	7.18 w/core	
5	ONE 5	AMD Opteron 6376	7.18 w/core	
0	ONE 1	INTEL Xeon 5540	10 w/core	Últimas Posiciones
1	ONE 2	INTEL Xeon 5540	10 w/core	

Fig. 4: Análisis del consumo energético de los nodos y ordenación para el algoritmo.

IV. MECANISMOS DE SEGURIDAD Y ALTA DISPONIBILIDAD EN FEDERACIONES DE CPD

En esta sección se describe la propuesta de mecanismos de seguridad y de alta disponibilidad para proporcionar a las federaciones de CPD de herramientas las protejan de ataques y fallos en los sistemas de información. La herramienta de seguridad que se ha desarrollado dota de mecanismos de detección y mitigación de ataques DoS (Denegación de Servicio) a la federación. De esta manera, se protegen los activos más importantes de la misma que son los recursos que se están ejecutando.

Además, se describen mecanismos de alta disponibilidad en forma de sistemas de respaldo de la información y de las máquinas virtuales que se están ejecutando en la federación, de forma que se salvaguarde la información ante posibles incidencias o catástrofes.

A. Mecanismos de seguridad para la detección y mitigación de un ataque DoS a la federación de CPD

La provisión de soluciones encaminadas a la virtualización de los elementos que tienen que ver, en general, con el tratamiento de la información usada en infraestructuras federadas, conlleva un buen número de elementos que deben ser controlados y gestionados de forma eficaz. Es por esto que la virtualización, no sólo de los nodos de cómputo, sino también de los elementos que las interconectan, dotan a la infraestructura de una mayor flexibilidad a la hora de gestionar los recursos.

Por este motivo, el uso de funciones de red virtualizadas (NFV, Network Function Virtualization) en redes definidas por software (SDN, Software Defined Networks) está desarrollándose muy rápidamente en entornos de virtualización.

SDN y NFV son tecnologías usualmente complementarias entre sí; en general, las redes de comunicaciones basadas en SDNFV combinan la gestión de la red introducida por SDN con la simplificación de la organización y uso de recursos y servicios que proporciona la virtualización de funciones de red a través de NFV.

Como consecuencia de la interacción de distintos tipos de tecnologías y protocolos durante el proceso de orquestación de la red, la seguridad de la información juega un papel crucial para preservar la integridad de los sistemas. Este proceso de orquestación permite programar comportamientos automáticos en una red, con el objetivo de coordinar elementos hardware y software que se destinan al despliegue de servicios y aplicaciones.

En la propuesta que se realiza en este trabajo, el controlador-orquestador implementa dos módulos para la detección y mitigación de ataques DoS, los cuales se encuentran interconectados, ya que la mitigación necesita información específica sobre el flujo de tráfico implicado, las IP origen y destino, los puertos origen y destino y el protocolo utilizado para realizar el ataque.

En ese sentido, el módulo de detección es una hebra de ejecución que periódicamente envía una solicitud a los switches de OpenFlow para que obtengan información estadística sobre los flujos de tráfico.

Como puede observarse en la Figura 5, el bucle principal solicita estadísticas de los flujos de tráfico hacia el switch, para posteriormente analizar los flujos de tráfico, mediante el uso de la primitiva *OFPExpStateStatsMultipartRequestAndDelete*.

Así mismo, para recibir un respuesta del switch se ha creado un gestor de eventos que escucha la llegada de eventos de tipo *EventOFPExperimenterStatsReply*, codificados en mensajes que incluyen las estadísticas solicitadas.

Una vez recibido el mensaje, las estadísticas son analizadas para obtener la información previamente mencionada en secciones anteriores, a saber: IP destino y origen de paquete enviado a través del flujo correspondiente y puertos origen y destino (protocolos TCP o UDP), para hacer el recuento de la información que se está transmitiendo a través de los distintos flujos de tráfico.

Una vez que los flujos son analizados se calcula la entropía, el cual es un parámetro que se utiliza para detectar un ataque DoS, a partir de la medición de parámetros estadísticos con información de la cabecera de los paquetes distribuidos a través de los flujos de tráfico.

En este caso, la muestra analizada se basa en la comparación de la entropía generada por muestras consecutivas de paquetes, lo que servirá para determinar si se sufre un ataque de este tipo.

La entropía puede definirse mediante la siguiente ecuación:

$$E = -\sum_{i=1}^{n} \rho_i \log_2 \rho_i \tag{1}$$

Donde E es la entropía, n el número de elementos detectados en el análisis de los flujos de tráfico y ρ_i es la probabilidad de encontrar el elemento i en la combinación de elementos detectados en el análisis.

Así mismo, es destacable mencionar que para el cálculo de la entropía se han utilizado las IP origen y destino, así como los puertos origen y destino.

El algoritmo de detección de ataques DoS comienza su ejecución calculando la entropía, para posteriormente continuar con la detección del ataque como tal, mediante la siguiente ecuación:

$$l - inf = \bar{x}_p - precision * \sigma_p \tag{2}$$

$$l - sup = \bar{x}_p - precision * \sigma_p \tag{3}$$

donde,

$$ataque = \begin{cases} falso & \text{Si} \quad l - \inf < x < l - \sup \\ verdadero & resto \end{cases} \tag{4}$$

```
while True:
  # Send the states requests to the state tables
  for datapath in datapaths:
      request_stats(datapath)
      sleep (X)  # Wait X seconds
```

Fig. 5: Bucle principal del análisis.

Donde $\bar{x}_p$ es el valor medio de los elementos analizados y $precision$ define la precisión del algoritmo de detección de ataques. Algunos valores de precisión utilizados por el algoritmo son el 68% (precisión baja) 95% (valor medio de precisión) o 99,7% para especificar una alta precisión. Por último, σ_p es la desviación estándar.

Así mismo, si el ataque se detecta, el controlador-orquestador comprueba en una base de datos que contiene información sobre el clúster de Dockers en qué servidor desplegar la función de red virtualizada. Una vez que el servidor se ha elegido, el controlador abre un canal seguro para el despliegue del Docker correspondiente.

El mecanismo para el despliegue del Docker sigue la descripción del fichero *Dockerfile*, el cual incluye, entre otros, el comportamiento del Docker. El código en la Figura 6 muestra un ejemplo de *Dockerfile*.

Una vez que el Docker ha sido creado, las interfaces de red ya están disponibles y el *bridge* entre ellas se ha establecido, por lo que el Docker pasa a ser desplegado al servidor correspondiente del clúster. A partir de ese momento, el controlador debe modificar las tablas de enrutamiento de los switches que se han visto envueltos en el ataque DoS para mitigarlo.

Con este mecanismo, los switches que han sido atacados solo tienen que distribuir paquetes al puerto de salida, y el ataque es mitigado mediante el uso de un servidor designado específicamente para ello. En este caso, el firewall usado como NFV es una implementación de reglas *iptables*.

B. Copias de seguridad

Como parte de los mecanismos de seguridad desarrollados dentro de la federación de CPD, se ha decidido implementar un sistema de copias de respaldo de máquinas virtuales, cuyo objetivo sería el de permitir que los usuarios puedan restaurar sus máquinas virtuales en caso de que hubiera un problema con alguna de ellas (disco dañado, borrado accidental de la máquina, configuración errónea de parámetros etc.) en alguna de las otras zonas de la federación.

Esta funcionalidad es complementaria a los mecanismos de seguridad aportados por OpenNebula para la propia zona, los cuales no permiten que el usuario haga copias en zonas distintas de la federación, sino solamente dentro del mismo conjunto de recursos de *cloud* de la propia zona local (por ejemplo *snapshots* del disco o de la propia máquina virtual). Ello resulta problemático, ya que se obliga a almacenar las copias de respaldo en la misma zona que los elementos virtuales a proteger, por lo que un error de acceso a la zona impedirá la obtención, tanto de los recursos originales, como de sus copias. Por ello, se plantea que el sistema de respaldo de las máquinas virtuales almacene las copias en recursos de *cloud* situados en una localización geográfica distinta.

```
# Allow HTTP traffic only

FROM base

ENTRYPOINT ifinit && \
 brinit && \
 iptables -A FORWARD -p tcp --dport 80 -j ACCEPT && \
 iptables -A FORWARD -j DROP &&    \
 /bin/bash
```

Fig. 6: Fichero para el despliegue de una función de firewall virtualizado.

1) Esquema general de la solución: El esquema general está representado en la Figura 7:

El sistema dispone de dos procedimientos principales:

- **Proceso creación copia MV**: Permite crear una máquina virtual de respaldo a partir de una copia del disco duro de la MV a respaldar, enviada a la zona remota, y la creación de los elementos virtuales de OpenNebula necesarios (como la plantilla).
- **Proceso restauración MV**: La restauración simplemente consiste en lanzar la MV, en la zona remota, a partir de la plantilla que incluirá el disco donde se ha copiado la información de la MV original.

De la lectura de los pasos del procedimiento se deduce que realmente no se copia la MV como tal (disco, memoria principal, etc.) sino que se copia el disco y se generan los recursos virtuales que permitirán desplegar una MV con los mismos datos en la zona remota.

2) Implementación: La implementación se ha realizado tanto con Ruby (a través de la API de OpenNebula) como mediante *shell scripts* de *bash*. Esta interfaz de operaciones permite implementar aplicaciones que acceden a todos los recursos asociados al usuario identificado, y que son capaces de ejecutar toda una gama de operaciones para operar con esos recursos. Por ello, accediendo mediante el usuario administrador (por ejemplo *oneadmin*) es posible implementar mecanismos automáticos de generación de copias de seguridad, independientemente de que las máquinas pertenezcan a otros usuarios.

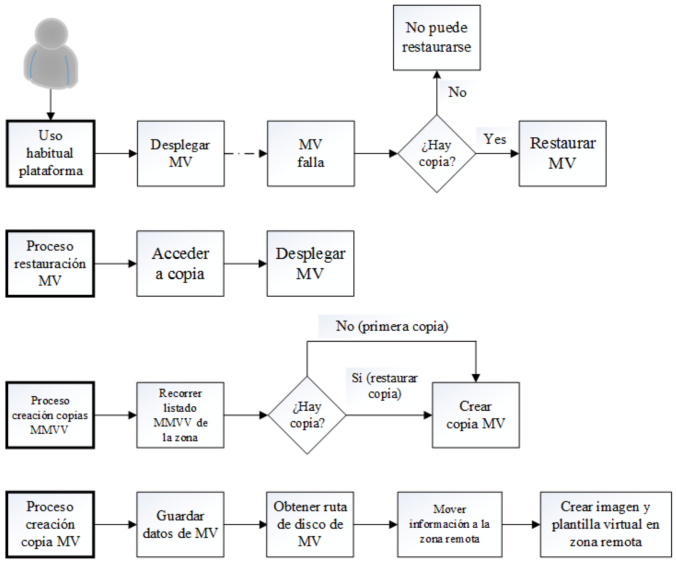

Fig. 7: Esquema general del sistema de copias de respaldo de la Máquina Virtual.

Por otra parte, es destacable mencionar que el sistema requiere que en las zonas implicadas el *hypervisor* sea KVM (Kernel-based Virtual Machine). Aunque OpenNebula permite gestionar recursos de *cloud* con otros *hypervisores* (concretamente Xen y VMware), el despliegue de copias de respaldo de máquinas virtuales en servidores con distintos *hypervisores* requeriría la conversión del formato de la información donde se definen las características de la MV, entre otras cuestiones, lo cual se aleja de los objetivos propuestos inicialmente para el proyecto.

El código se estructura en dos carpetas distintas, localizadas en las máquinas de las zonas donde se encuentran instalados el *frontend* y los servicios de OpenNebula. Los detalles más significativos de la implementación son los siguientes:

- **Creación de la copia de respaldo**:
  - *Lanzamiento del proceso de copia*: La primera parte del proceso sería lanzar el procedimiento para guardar los datos de los procesos abiertos en la MV que se desee copiar.
  - *Hacer un snapshot del disco de la MV*: Lo que se hace es crear una imagen virtual del disco seleccionado, a partir de la operación *disk_snapshot* de la API de OpenNebula, lo que conlleva que los datos que están usando los procesos en ejecución de la MV se guarden al disco duro.
  - *Obtener el fichero asociado al disco*: A continuación, el sistema debe obtener el fichero real (ISO) donde se encuentra almacenada la información asociada a la imagen virtual obtenida en el paso anterior.
  - *Copiar el disco a la zona de respaldo*: Este paso consiste en copiar y mover la ISO con los datos del disco a la zona remota de OpenNebula en la que se quiere almacenar la copia (específicamente hacia la máquina donde se encuentra instalado el correspondiente *front-end*).
  - *Crear la imagen virtual*: Gracias a que el fichero ISO se encuentra en una carpeta temporal de la máquina *front-end* de la zona remota, será posible crear una imagen virtual que podrá ser utilizada como el disco de las nuevas MV desplegadas.
  - *Crear una plantilla asociada a la copia*: Este paso es análogo al anterior, aunque lo que se está creando en la zona remota es una plantilla virtual.
- **Restauración de la copia de respaldo**:
  - *Lanzamiento de la MV*: El procedimiento consiste simplemente en acceder a la interfaz de OpenNebula, entrar en la zona remota y lanzar la máquina virtual, tal y como se hace normalmente. Ya que la plantilla referencia a una imagen virtual con el disco de respaldo, aunque la MV que se está desplegando sea nueva, ésta contiene toda la información que el usuario tenía en su MV en el momento de hacer la copia.

V. CONCLUSIONES Y LÍNEAS FUTURAS

Actualmente, la gestión de una federación de CPD con capacidad para reubicar la carga computacional entre los diferentes centros de datos para incrementar la eficiencia energética de los mismos, es una de las líneas de trabajo, tanto de la industria, como de los investigadores. En ese sentido, La reducción del consumo energético de los centros de datos supone a una disminución del coste de mantenimiento de los mismos, así como una reducción indirecta de la huella de carbono asociada.

Por esto, la gestión inteligente de la energía también permitirá planificar y programar el funcionamiento de los equipos que mayor volumen de energía consumen. Las líneas de trabajo en este ámbito son muy amplias y abarcan muchos campos de acción. En este proyecto se ha desarrollado un conjunto de herramientas bajo el gestor de la infraestructura que permiten, no sólo la recolección de los datos del estado del clúster, sino la clasificación de los recursos activos y su gestión eficiente, teniendo en cuenta diferentes modelos.

Este trabajo es, por tanto, el inicio de un sistema inteligente que recopila datos del clúster activo, procesa la información, aprende de ella y toma decisiones teniendo en cuenta los requerimientos definidos.

En los trabajos futuros que se deben desarrollar para continuar el proyecto, se debe tener en cuenta los servicios que permiten retrasar su respuesta en comparación con los servicios interactivos, los cuales son muy dependientes del retardo. Además, la aplicación del algoritmo debe ser comprobado en otros tipos de trabajo y peticiones de usuarios en entornos *cloud*. La selección de la máquina virtual a migrar dependerá de la aplicación que esté siendo ejecutada, así como el nivel de servicio negociado con el usuario.

Otra de las principales líneas de trabajo es la inclusión de sistemas inteligentes que planifiquen las tareas de mayor consumo energético, así como la introducción de la toma de decisión a la hora de introducir fuentes de energía alternativas, como la geotermia o la energía fotovoltaica, que serán, en un futuro no muy lejano, mecanismos para el ahorro de los costes de mantenimiento en el CPD. Así, la introducción de sistemas inteligentes que basen su toma de decisiones en predicciones de los costes energéticos establecerá uno de los principales pilares de la gestión eficiente y económica de los CPD.

REFERENCÍAS

[1] Mell, Peter and Grance, Tim and others, "The NIST definition of cloud computing," Computer Security Division, Information Technology Laboratory, National Institute of Standards and Technology Gaithersburg, 2011.

[2] Kundra, Vivek, "Federal cloud computing strategy," White House [Chief Information Officers Council], 2011.

[3] OpenNebula, https://opennebula.org/.

[4] Goudarzi, Hadi y Pedram, Massoud. Energy-Efficient Virtual Machine Replication. 2012. https://pdfs.semanticscholar.org/3fd9/f945c3cc00436acfb7aef84ce48eeda25e50.pdf.