

Fragmento del texto original traducido al español por profesores de la unidad de aprendizaje: "Sistemas Distribuidos" de la ESCOM-IPN.

El texto completo puede adquirirse en: https://www.amazon.com/-/es/Erl-Thomas-ebook/dp/B00CM9V7Q8/ref=sr_1_1?__mk_es_US=%C3%85M%C3%85%C5%BD%C3%95%C3%91&crid=2OSZQ5MPX6BTG&keywords=cloud+computing&qid=1688489268&s=digital-text&sprefix=cloud+computing%2Cdigital-text%2C174&sr=1-1

Los términos fundamentales se usan de manera indistinta en inglés y en español a lo largo del texto para familiarizar al alumno con su uso.

Contenido

Capítulo 1 Comprendiendo el cómputo en la nube	17
1.1. Orígenes e influencias	18
Una breve historia	
Definiciones	
Innovaciones tecnológicas	
Clustering	
Grid Computing	
Virtualización	
1.2. Conceptos básicos y terminología	22
Nube	
Recurso de TI	
On-Premise	
Consumidores de la nube y proveedores de la nube	
Escalamiento	
Escalamiento horizontal	
Escalamiento vertical	
Cloud Service	
Consumidor de servicios en la nube	
1.3. Metas y Beneficios	28
Reducción de Inversiones y Costos Proporcionales	
Mayor escalabilidad	
Mayor disponibilidad y confiabilidad	
1.4. Riesgos y desafíos	30
Mayores vulnerabilidades de seguridad	
Control de gobierno operativo reducido	
Portabilidad limitada entre proveedores de la nube	
Cumplimiento multirregional y cuestiones legales	
Capítulo 2 Conceptos Fundamentales y Modelos	35
2.1. Roles y límites	36
Cloud Provider (Proveedor de la nube)	
Cloud Consumer (Consumidor de la nube)	
Propietario del servicio en la nube	
Cloud Resource Administrator	
Roles adicionales	
Límite organizacional	
Límite de confianza	
2.2. Características de la nube	41
Uso bajo demanda	
Acceso ubicuo	
Multitenancy (Tenencia múltiple)	

Elasticidad	
Uso medido	
Resiliencia	
2.3. Modelos de entrega en la nube	45
Infraestructura como servicio (IaaS)	
Plataforma como servicio (PaaS)	
Software como servicio (SaaS)	
Comparación de modelos de entrega en la nube	
Combinación de modelos de entrega en la nube	
IaaS + PaaS + SaaS	
2.4. Modelos de implementación en la nube	52
Nubes Públicas	
Nubes comunitarias	
Nubes Privadas	
Nubes híbridas	
Otros modelos de implementación en la nube	
Capítulo 3 Tecnologías para la nube	57
3.1. Redes de Banda Ancha y Arquitectura de Internet	
Proveedores de servicios de Internet (ISP)	
Connectionless Packet Switching (Datagram Networks)	
Router-Based Interconnectivity	
Red física	
Transport Layer Protocol	
Application Layer Protocol	
Consideraciones técnicas y comerciales	
Problemas de conectividad	
Problemas de latencia y ancho de banda de la red	
Selección de operador de nube y proveedor de nube	
3.2. Tecnología del centro de datos	66
Virtualización	
Estandarización y modularidad	
Automatización	
Operación y administración remotas	
Alta disponibilidad	
Diseño Security-Aware, operación y administración	
Instalaciones	
Hardware de cómputo	
Hardware de almacenamiento	
Hardware de red	
Interconexión de Carrier y Redes Externas	
Balanceo de carga y aceleración Web-Tier	
Estructura LAN	
Estructura SAN	

Puertas de enlace NAS	
Otras Consideraciones	
3.3. Tecnología de virtualización	72
Independencia de hardware	
Consolidación de servidores	
Replicación de recursos	
Virtualización basada en el sistema operativo	
Virtualización basada en hardware	
Virtualization Management	
Otras Consideraciones	
3.4. Tecnología web	77
Tecnología web básica	
Aplicaciones web	
3.5. Tecnología multitenant	79
Multitenancy vs Virtualización	
3.6. Containerization	81
Containerization vs. virtualización	
Beneficios de los contenedores	
Alojamiento de contenedores y Pods	
Elementos fundamentales de la arquitectura de contenedores	
Motor de contenedores	
Container Build File	
Imagen de contenedor	
Container	
Networking Address	
Dispositivo de almacenamiento	
3.7. Ejemplo de Estudio de Caso	85
Capítulo 4 Seguridad en la nube	89
4.1. Términos y conceptos básicos	90
Confidencialidad	
Integridad	
Autenticidad	
Disponibilidad	
Amenaza	
Vulnerabilidad	
Riesgo	
Controles de seguridad	
Mecanismos de seguridad	
Políticas de seguridad	
4.2. Threat Agents	92
Atacante anónimo	
Agente de servicio malicioso	
Trusted Attacker	

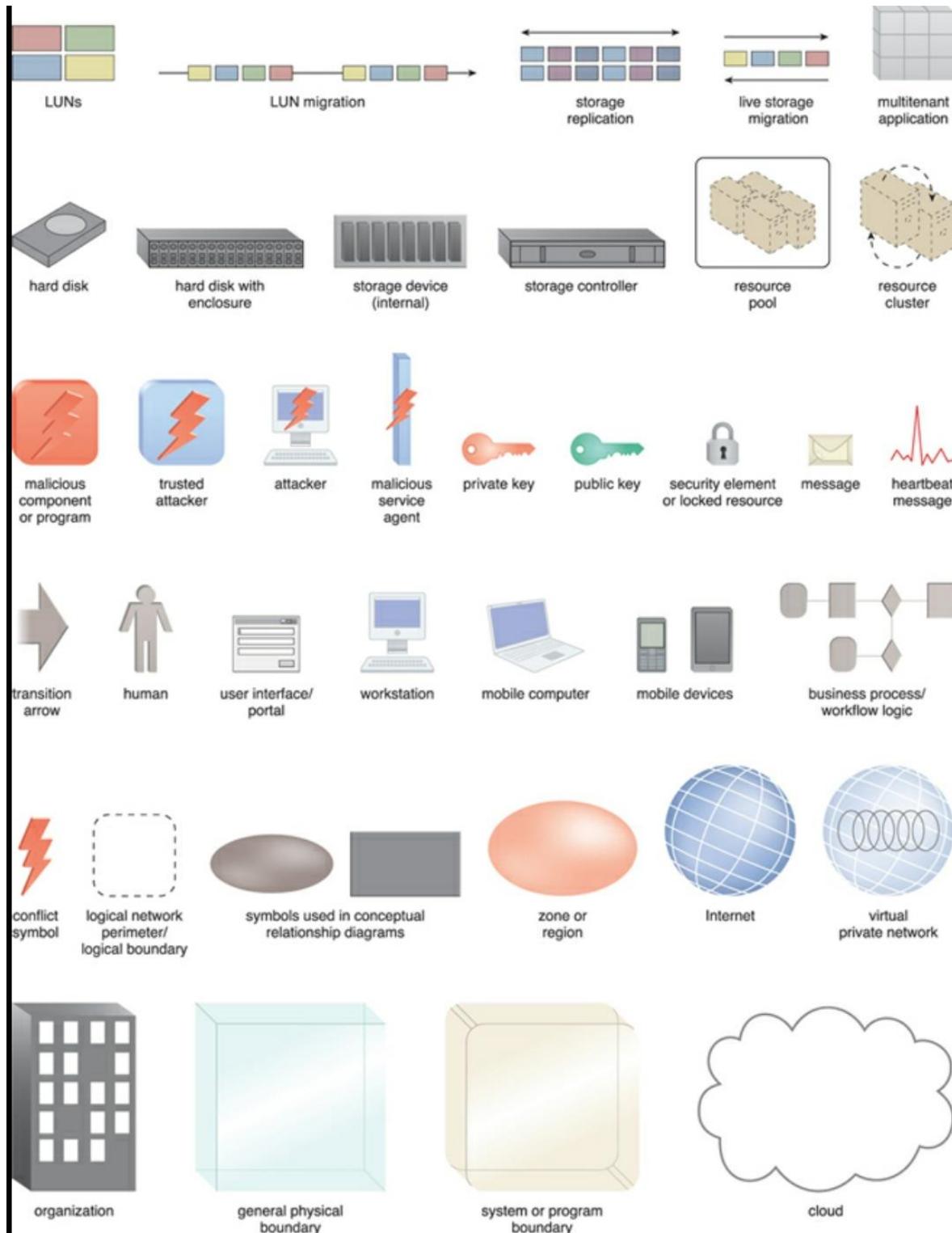
Insider Malicious	
4.3. Amenazas de seguridad en la nube	94
Espionaje de tráfico	
Intermediario malicioso	
Denegación de servicio	
Autorización insuficiente	
Ataque de virtualización	
Límites de confianza superpuestos	
Ataque de contenedores	
4.4. consideraciones adicionales	99
Implementaciones defectuosas	
Disparidad de políticas de seguridad	
Contratos	
4.5. Ejemplo de Estudio de Caso	101
Capítulo 5 Mecanismos en la infraestructura de la nube	102
5.1. Perímetro de red lógica	103
Ejemplo de Estudio de Caso	
5.2. Servidor virtual	106
Ejemplo de Estudio de Caso	
5.3. Cloud Storage Device	110
Cloud Storage Levels	
Network Storage Interfaces	
Object Storage Interfaces	
Database Storage Interfaces	
Almacenamiento de datos relacionales	
Almacenamiento de datos no relacionales	
Ejemplo de Estudio de Caso	
5.4. Monitoreo de uso de la nube	116
Agente de monitoreo	
Agente de recursos	
Polling agent	
Ejemplo de Estudio de Caso	
5.5. Replicación de recursos	121
Ejemplo de Estudio de Caso	
5.6. Entorno listo para usar	126
Ejemplo de Estudio de Caso	
5.7 Contenedor	127
Capítulo 6 Mecanismos especializados de la nube	129
6.1. Automated Scaling Listener	130
Ejemplo de Estudio de Caso	
6.2. Load Balancer	134
Ejemplo de Estudio de Caso	

6.3. Supervisión de SLA	136
Ejemplo de Estudio de Caso	
6.4. Monitor de pago por uso	142
Ejemplo de Estudio de Caso	
6.5. Audit Monitor	145
Ejemplo de Estudio de Caso	
6.6. Failover System	147
Activo-Activo	
Activo-pasivo	
Ejemplo de Estudio de Caso	
6.7. hipervisor	154
Ejemplo de Estudio de Caso	
6.8. Clúster de recursos	157
Ejemplo de Estudio de Caso	
6.9. Multi-device Bróker	160
Ejemplo de Estudio de Caso	
6.10. State Management Database	162
Ejemplo de Estudio de Caso	
Capítulo 7 Mecanismos de administración de la nube	165
7.1. Sistema de administración remota	166
Ejemplo de Estudio de Caso	
7.2. Sistema de gestión de recursos	170
Ejemplo de Estudio de Caso	
7.3. Sistema de gestión de SLA	172
Ejemplo de Estudio de Caso	
7.4. Sistema de Gestión de Facturación	174
Ejemplo de estudio de caso	
Capítulo 8 Cloud Security Mechanisms	177
8.1 Encriptado	178
Cifrado simétrico	
Cifrado asimétrico	
Ejemplo de Estudio de Caso	
8.2. Hashing	180
Ejemplo de Estudio de Caso	
8.3. Digital Signature	182
Ejemplo de Estudio de Caso	
8.4. Infraestructura de clave pública (PKI)	184
Ejemplo de Estudio de Caso	
8.5. Identity and Access Management (IAM)	187
Ejemplo de Estudio de Caso	
8.6. Single Sign-On (SSO)	188
Ejemplo de Estudio de Caso	

8.7. Grupos de seguridad basados en la nube	190
Ejemplo de Estudio de Caso	
8.8. Hardened Virtual Server Images	193
Ejemplo de Estudio de Caso	
Capítulo 9 Arquitecturas fundamentales de la nube	196
9.1. Arquitectura de distribución de carga de trabajo	197
9.2. Arquitectura de pooling de recursos	198
9.3. Arquitectura de escalabilidad dinámica	202
9.4. Arquitectura de capacidad de recursos elásticos	204
9.5. Arquitectura del servicio de balanceo de carga	207
9.6. Arquitectura de Cloud Bursting	209
9.7. Arquitectura de aprovisionamiento de discos elásticos	210
9.8. Arquitectura de almacenamiento redundante	213
9.9. Ejemplo de Estudio de Caso	215
Capítulo 10 Arquitecturas de nube avanzadas	218
10.1. Hypervisor Clustering Architecture	219
Migración de máquinas virtuales en vivo	
10.2. Arquitectura de instancias de servidor virtual con equilibrio de carga	225
10.3. Arquitectura de reubicación de servicios no disruptiva	227
10.4. Arquitectura Zero Downtime	232
10.5. Arquitectura de balanceo en la nube	234
10.6. Arquitectura de reserva de recursos	236
10.7. Arquitectura de recuperación y detección dinámica de fallas	240
10.8. Arquitectura de aprovisionamiento Bare-Metal	243
10.9. Arquitectura de aprovisionamiento rápido	246
10.10. Arquitectura de gestión de carga de trabajo de almacenamiento	248
Ejemplo de Estudio de Caso	

Referencia de símbolos usados en el texto:





Empresas para Estudios de Casos

Un estudio de caso es un método empírico que investiga un fenómeno contemporáneo (el caso) en profundidad y dentro de su contexto del mundo real, especialmente cuando las fronteras entre fenómeno y contexto pueden no ser claras o evidentes. (Yin, 2018)

Los ejemplos de estudios de casos que se verán a lo largo del texto proporcionan escenarios en los que las organizaciones evalúan, usan y administran modelos y tecnologías de computación en la nube. En este libro se presentan tres organizaciones de diferentes industrias para su análisis, cada una de las cuales tiene objetivos comerciales, tecnológicos y arquitectónicos distintivos.

Empresa #1: ATN

ATN es una empresa que proporciona equipos de red a las industrias de telecomunicaciones en todo el mundo. A lo largo de los años, ATN ha crecido considerablemente y su cartera de productos se ha ampliado para dar cabida a varias adquisiciones, incluidas empresas que se especializan en componentes de infraestructura para proveedores de Internet, GSM y celulares. ATN es ahora un proveedor líder de una diversa gama de infraestructura de telecomunicaciones.

En los últimos años, la presión del mercado ha ido en aumento. ATN ha comenzado a buscar formas de aumentar su competitividad y eficiencia aprovechando las nuevas tecnologías, especialmente aquellas que pueden ayudar en la reducción de costos.

Infraestructura Técnica y Medio Ambiente

Las diversas adquisiciones de ATN han resultado en un panorama de TI altamente complejo y heterogéneo. No se aplicó un programa de consolidación coherente al entorno de TI después de cada ronda de adquisición, lo que resultó en aplicaciones similares ejecutándose simultáneamente y un aumento en los costos de mantenimiento. En 2010, ATN se fusionó con un importante proveedor europeo de telecomunicaciones, agregando otra cartera de aplicaciones a su inventario. La complejidad de TI se convirtió en una obstrucción grave y se convirtió en una fuente de preocupación crítica para la junta directiva de ATN.

Objetivos comerciales y nueva estrategia

La gerencia de ATN decidió emprender una iniciativa de consolidación y subcontratar el mantenimiento y las operaciones de las aplicaciones en el extranjero. Esto redujo los costos, pero desafortunadamente no abordó su ineficiencia operativa general. Las aplicaciones aún tenían funciones superpuestas que no podían consolidarse fácilmente. Eventualmente se hizo evidente que la subcontratación era insuficiente ya que la consolidación se convirtió en una posibilidad solo si la arquitectura de todo el panorama de TI cambiaba.

Como resultado, ATN decidió explorar el potencial de adoptar la computación en la nube. Sin embargo, después de sus consultas iniciales, se vieron abrumados por la gran cantidad de proveedores de servicios en la nube y productos basados en la nube.

Hoja de ruta y estrategia de implementación

ATN no está seguro de cómo elegir el conjunto correcto de tecnologías y proveedores de computación en la nube; muchas soluciones parecen aún inmaduras y continúan surgiendo en el mercado nuevas ofertas basadas en la nube.

Se analiza una hoja de ruta preliminar de adopción de la computación en la nube para abordar una serie de puntos clave:

- *Estrategia de TI* - La adopción de la computación en la nube debe promover la optimización del marco de TI actual y producir inversiones más bajas a corto plazo y una reducción constante de costos a largo plazo.
- *Beneficios comerciales* - ATN debe evaluar cuáles de las aplicaciones e infraestructuras de TI actuales pueden aprovechar la tecnología de computación en la nube para lograr la optimización y las reducciones de costos deseadas. Es necesario aprovechar los beneficios adicionales de la computación en la nube, como una mayor agilidad comercial, escalabilidad y confiabilidad, para promover el valor comercial.
- *Consideraciones tecnológicas* - Se deben establecer criterios para ayudar a elegir los modelos de entrega e implementación de la nube y los proveedores y productos de la nube más apropiados.
- *Seguridad en la nube* - Se deben determinar los riesgos asociados con la migración de aplicaciones y datos a la nube.

ATN teme perder el control de sus aplicaciones y datos si se confían a los proveedores de la nube, lo que provocaría el incumplimiento de las políticas internas y las reglamentaciones del mercado de las telecomunicaciones. También se preguntan cómo se integrarían sus aplicaciones heredadas existentes en el nuevo dominio basado en la nube.

Para definir un plan de acción conciso, ATN contrata a una empresa consultora de TI independiente llamada CloudEnhance, reconocida por su experiencia en arquitectura tecnológica en la transición e integración de recursos de TI de computación en la nube. Los consultores de CloudEnhance comienzan sugiriendo un proceso de evaluación compuesto por cinco pasos:

- 1.** Una breve evaluación de las aplicaciones existentes para medir factores, como la complejidad, la criticidad comercial, la frecuencia de uso y la cantidad de usuarios activos. Los factores identificados luego se colocan en una jerarquía de prioridad para ayudar a determinar las aplicaciones candidatas más adecuadas para la migración a un entorno de nube.
- 2.** Una evaluación más detallada de cada aplicación seleccionada utilizando una herramienta de evaluación patentada.
- 3.** El desarrollo de una arquitectura de aplicación target que muestre la interacción entre las aplicaciones basadas en la nube, su integración con la infraestructura existente y los sistemas heredados de ATN, así como sus procesos de desarrollo e implementación.
- 4.** La creación de un caso comercial preliminar que documente los ahorros de costos proyectados en función de los indicadores de desempeño, como el costo de la preparación para la nube, el esfuerzo para la transformación e interacción de las aplicaciones, la facilidad de migración e implementación y varios beneficios potenciales a largo plazo.

5. El desarrollo de un detallado plan de proyecto para una aplicación piloto.

ATN continúa con el proceso y, como resultado, construye su primer prototipo centrándose en una aplicación que automatiza un área comercial de bajo riesgo. Durante este proyecto, ATN traslada varias de las aplicaciones más pequeñas del área comercial que se ejecutaban en diferentes tecnologías a una plataforma PaaS. Con base en los resultados positivos y la retroalimentación recibida para el proyecto prototipo, ATN decide embarcarse en una iniciativa estratégica para generar beneficios similares para otras áreas de la empresa.

[Empresa #2: DTGOV](#)

DTGOV es una empresa pública que fue creada a principios de la década de 1980 por el Ministerio de Seguridad Social. La descentralización de las operaciones de TI del ministerio a una empresa pública de derecho privado le dio a DTGOV una estructura de gestión autónoma con una flexibilidad significativa para gobernar y desarrollar su empresa de TI.

En el momento de su creación, DTGOV tenía aproximadamente 1000 empleados, sucursales operativas en 60 localidades en todo el país y operaba dos centros de datos basados en mainframe. Con el tiempo, DTGOV se ha expandido a más de 3000 empleados y sucursales en más de 300 localidades, con tres centros de datos que ejecutan entornos de plataforma de bajo nivel y mainframe. Sus principales servicios están relacionados con la tramitación de prestaciones de seguridad social en todo el país.

DTGOV ha ampliado su cartera de clientes en las últimas dos décadas. Ahora sirve a otras organizaciones del sector público y proporciona infraestructura y servicios básicos de TI, como alojamiento de servidores y colocación de servidores. Algunos de sus clientes también han subcontratado la operación, mantenimiento y desarrollo de aplicaciones a DTGOV.

DTGOV tiene importantes contratos con clientes que abarcan varios recursos y servicios de TI. Sin embargo, estos contratos, servicios y niveles de servicio asociados no están estandarizados: las condiciones de provisión de servicios negociadas generalmente se personalizan para cada cliente individualmente. Como resultado, las operaciones de DTGOV se están volviendo cada vez más complejas y difíciles de administrar, lo que ha llevado a ineficiencias y costos inflados.

El directorio de DTGOV se dio cuenta, hace algún tiempo, que la estructura general de la empresa podría mejorarse mediante la estandarización de su cartera de servicios, lo que implica la reingeniería de los modelos operativos y de gestión de TI. Este proceso ha comenzado con la estandarización de la plataforma de hardware a través de la creación de un ciclo de vida tecnológico claramente definido, una política de adquisiciones consolidada y el establecimiento de nuevas prácticas de adquisición.

Infraestructura técnica y entorno

DTGOV opera tres centros de datos: uno está dedicado exclusivamente a servidores de plataforma de bajo nivel, mientras que los otros dos tienen tanto mainframe como plataformas de bajo nivel. Los sistemas mainframe están reservados para el Ministerio de Seguridad Social y, por lo tanto, no están disponibles para la subcontratación.

La infraestructura del centro de datos ocupa aproximadamente 20 000 pies cuadrados de espacio de sala de computadoras y alberga más de 100 000 servidores con diferentes configuraciones de hardware. La capacidad total de almacenamiento es de aproximadamente 10.000 terabytes. La red de DTGOV tiene enlaces de datos redundantes de alta velocidad que conectan los centros de datos en una topología de malla completa. Se considera que su conectividad a Internet es independiente del proveedor, ya que su red interconecta a todos los principales operadores de telecomunicaciones nacionales.

Los proyectos de consolidación y virtualización de servidores han estado en marcha durante cinco años, disminuyendo considerablemente la diversidad de plataformas de hardware. Como resultado, el seguimiento sistemático de las inversiones y los costos operativos relacionados con la plataforma de hardware ha revelado una mejora significativa. Sin embargo, todavía existe una notable diversidad en sus plataformas y configuraciones de software debido a los requisitos de personalización del servicio al cliente.

Objetivos comerciales y nueva estrategia

Un objetivo estratégico principal de la estandarización de la cartera de servicios de DTGOV es lograr mayores niveles de rentabilidad y optimización operativa. Se estableció una comisión interna de nivel ejecutivo para definir las direcciones, los objetivos y la hoja de ruta estratégica para esta iniciativa. La comisión ha identificado la computación en la nube como una oportunidad para una mayor diversificación y mejora de los servicios y carteras de clientes.

La hoja de ruta aborda los siguientes puntos clave:

- *Beneficios comerciales* - Es necesario definir los beneficios comerciales concretos asociados con la estandarización de las carteras de servicios bajo el abanico de los modelos de entrega de computación en la nube. Por ejemplo, ¿cómo puede la optimización de la infraestructura de TI y los modelos operativos generar reducciones de costos directos y medibles?
- *Cartera de servicios* - ¿Qué servicios deberían basarse en la nube y a qué clientes deberían extenderse?
- *Desafíos técnicos* - Se deben comprender y documentar las limitaciones de la infraestructura tecnológica actual en relación con los requisitos de procesamiento en tiempo de ejecución de los modelos de computación en la nube. La infraestructura existente debe aprovecharse en la medida de lo posible para optimizar los costos iniciales asumidos por el desarrollo de las ofertas de servicios basados en la nube.
- *Precios y SLAs* - Es necesario definir una estrategia adecuada de precios, contratos y calidad del servicio. Se deben determinar los precios adecuados y los acuerdos de nivel de servicio (SLAs) para respaldar la iniciativa.

Una preocupación pendiente se relaciona con los cambios en el formato actual de los contratos y cómo pueden afectar el negocio. Es posible que muchos clientes no deseen, o no estén preparados para, adoptar modelos de contratación y prestación de servicios en la nube. Esto se vuelve aún más crítico si se considera el hecho de que el 90 % de la cartera actual de clientes de DTGOV está compuesta por organizaciones públicas que, por lo general, no tienen la autonomía o la agilidad

para cambiar los métodos operativos en tan poco tiempo. Por lo tanto, se espera que el proceso de migración sea a largo plazo, lo que puede volverse riesgoso si la hoja de ruta no se define de manera adecuada y clara. Otro problema pendiente se refiere a las regulaciones de contratos de TI en el sector público: las regulaciones existentes pueden volverse irrelevantes o poco claras cuando se aplican a las tecnologías de nube.

Hoja de ruta y estrategia de implementación

Se iniciaron varias actividades de evaluación para abordar los problemas antes mencionados. La primera fue una encuesta de clientes existentes para probar su nivel de comprensión, iniciativas en curso y planes con respecto a la computación en la nube. La mayoría de los encuestados conocían sobre las tendencias de la computación en la nube, lo que se consideró un hallazgo positivo.

Una investigación de la cartera de servicios reveló servicios de infraestructura claramente identificados relacionados con el hospedaje y la colocación. También se evaluaron la experiencia técnica y la infraestructura, determinando que la operación y administración del centro de datos son áreas clave de experiencia del personal de TI de DTGOV.

Con estos hallazgos, la comisión decidió:

- 1. Elegir IaaS como la plataforma de entrega de destino para iniciar la iniciativa de aprovisionamiento de computación en la nube**
- 2. Contratar una empresa de consultoría con suficiente experiencia y experiencia en proveedores de nube para identificar y corregir correctamente cualquier problema comercial y técnico que pueda afectar la iniciativa**
- 3. Implementar nuevos recursos de hardware con una plataforma uniforme en dos centros de datos diferentes, con el objetivo de establecer un entorno nuevo y confiable que usar para el aprovisionamiento de servicios iniciales alojados en IaaS**
- 4. Identificar tres clientes que planean adquirir servicios basados en la nube para establecer proyectos piloto y definir condiciones contractuales, precios y políticas y modelos de nivel de servicio**
- 5. Evaluar la prestación del servicio de los tres clientes elegidos durante el período inicial de seis meses antes de ofrecer públicamente el servicio a otros clientes**

Como proyecto piloto, se lanza un nuevo entorno de administración basado en la web para permitir el autoaproxionamiento de servidores virtuales, así como SLA y funcionalidad de seguimiento financiero en tiempo real. Los proyectos piloto se consideran muy exitosos y conducen al siguiente paso de abrir los servicios basados en la nube a otros clientes.

Empresa #3: Innovartus Technologies Inc.

La principal línea de negocios de Innovartus Technologies Inc. es el desarrollo de juguetes virtuales y productos de entretenimiento educativo para niños. Estos servicios se brindan a través de un portal web que emplea un modelo de juego de roles para crear juegos virtuales personalizados para PC y dispositivos móviles. Los juegos permiten a los usuarios crear y manipular juguetes virtuales (automóviles, muñecas, mascotas) que pueden equiparse con accesorios virtuales que se obtienen

al completar misiones educativas simples. El principal grupo demográfico son los niños menores de 12 años. Innovartus también tiene un entorno de red social que permite a los usuarios intercambiar artículos y colaborar con otros. Todas estas actividades pueden ser monitoreadas y rastreadas por los padres, quienes también pueden participar en un juego creando misiones específicas para sus hijos.

La característica más valiosa y revolucionaria de las aplicaciones de Innovartus es una interfaz de usuario final experimental que se basa en conceptos de interfaz natural. Los usuarios pueden interactuar a través de comandos de voz, gestos simples que se capturan con una cámara web y directamente al tocar las pantallas de las tabletas.

El portal de Innovartus siempre ha estado basado en la nube. Originalmente se desarrolló a través de una plataforma PaaS y desde entonces ha sido alojado por el mismo proveedor de nube. Sin embargo, recientemente este entorno ha revelado varias limitaciones técnicas que afectan las características de los marcos de programación de la interfaz de usuario de Innovartus.

Infraestructura técnica y entorno

Muchas de las otras soluciones de automatización de oficinas de Innovartus, como repositorios de archivos compartidos y varias herramientas de productividad, también están basadas en la nube. El entorno de TI corporativo en las instalaciones es relativamente pequeño y se compone principalmente de dispositivos de área de trabajo, computadoras portátiles y estaciones de trabajo de diseño gráfico.

Metas comerciales y estrategia

Innovartus ha estado diversificando la funcionalidad de los recursos de TI que se utilizan para sus aplicaciones móviles y basadas en la Web. La empresa también ha incrementado sus esfuerzos para internacionalizar sus aplicaciones; tanto el sitio web como las aplicaciones móviles se ofrecen actualmente en cinco idiomas diferentes.

Hoja de ruta y estrategia de implementación

Innovartus tiene la intención de continuar desarrollando sus soluciones basadas en la nube; sin embargo, el entorno de alojamiento en la nube actual tiene limitaciones que deben superarse:

- la escalabilidad necesita mejorarse para adaptarse a una mayor y menos predecible interacción del consumidor de la nube
- es necesario mejorar los niveles de servicio para evitar interrupciones que actualmente son más frecuentes de lo esperado
- es necesario mejorar la rentabilidad, ya que las tarifas de arrendamiento son más altas con el proveedor de la nube actual en comparación con otros

Estos y otros factores han llevado a Innovartus a decidir migrar a un proveedor de nube más grande y establecido a nivel mundial.

La hoja de ruta para este proyecto de migración incluye:

- un informe técnico y económico sobre los riesgos e impactos de la migración planificada

- un árbol de decisión y una rigurosa iniciativa de estudio centrada en los criterios para seleccionar el nuevo proveedor de nube
- evaluaciones de portabilidad de las aplicaciones para determinar qué parte de cada arquitectura de servicio de nube existente es propietario del entorno del proveedor de la nube actual Innovartus está más preocupado por cómo y en qué medida el proveedor de la nube actual apoyará y cooperará con el proceso de migración.

1 Comprendiendo el cómputo en la nube



Este es el primero de dos capítulos que brindan una descripción general de los temas introductorios de computación en la nube. Comienza con una breve historia de la computación en la nube junto con breves descripciones de sus impulsores comerciales y tecnológicos. A esto le siguen definiciones de conceptos básicos y terminología, además de explicaciones de los principales beneficios y desafíos de la adopción de la computación en la nube.

1.1. Orígenes e influencias

Una breve historia

La idea de computación en la "nube" se remonta a los orígenes de la computación utilitaria, un concepto que el científico informático John McCarthy propuso públicamente en 1961:

"Si las computadoras del tipo que he defendido se convierten en las computadoras del futuro, entonces la computación puede organizarse algún día como un servicio público, al igual que el sistema telefónico es un servicio público... El servicio de cómputo podría convertirse en la base de una industria nueva e importante."

En 1969, Leonard Kleinrock, científico jefe de la Red de la Agencia de Proyectos de Investigación Avanzada o proyecto ARPANET que sembró la semilla de Internet, declaró:

"A partir de ahora, las redes de computadoras todavía están en su infancia, pero a medida que crezcan y se vuelvan sofisticados, probablemente veremos la expansión de las 'utilidades de cómputo'..."

El público en general ha aprovechado variantes de utilidades de cómputo basadas en Internet desde mediados de la década de 1990 a través de los motores de búsqueda (Yahoo!, Google), servicios de correo electrónico (Hotmail, Gmail), plataformas de publicación abiertas (MySpace, Facebook, YouTube), y otros tipos de redes sociales (Twitter, LinkedIn). Aunque centrados en el consumidor, estos servicios popularizaron y validaron conceptos básicos que forman la base del cómputo en la nube moderno.

A fines de la década de 1990, Salesforce.com fue pionera en la idea de brindar servicios aprovisionados de forma remota a la empresa. En 2002, Amazon.com lanzó la plataforma Amazon Web Services (AWS), un conjunto de servicios orientados a la empresa que proporcionan de manera remota almacenamiento, recursos informáticos y funcionalidades de negocios.

Una evocación ligeramente diferente del término "Network Cloud" o "Cloud" se introdujo a principios de la década de 1990 en toda la industria de redes. Se refería a una capa de abstracción derivada de los métodos de entrega de datos a través de redes públicas y semipúblicas heterogéneas que eran principalmente conmutadas por paquetes¹, aunque las redes celulares también usaban el término "Nube". La interconexión de red en ese momento admitía la transmisión

¹ Una red conmutada por paquetes es un tipo de red de comunicación de datos en la que la información se divide en paquetes antes de ser transmitida a través de la red. Cada paquete contiene una parte de los datos, además de información de control que incluye las direcciones de origen y destino. Estos paquetes pueden seguir diferentes rutas hacia su destino y luego ser reensamblados en el orden correcto para formar la información completa. Fuente: Wikipedia

de datos desde un end-point² (red local) a la "nube" (red de área amplia) y luego se recomponía en otro end-point destino. Esto es relevante, ya que la industria de las redes todavía hace referencia al uso de este término y se considera una de las primeras en adoptar los conceptos que subyacen en el cómputo utilitario.

No fue sino hasta 2006 que el término "cloud computing" (computación en la nube) surgió en el ámbito comercial. Fue durante este tiempo que Amazon lanzó sus servicios Elastic Compute Cloud (EC2) que permitieron a las organizaciones rentar capacidad informática y potencia de procesamiento para ejecutar sus aplicaciones empresariales. Google Apps también comenzó a proporcionar aplicaciones empresariales basadas en navegador ese mismo año y, tres años después, Google App Engine se convirtió en otro hito histórico.

Definiciones

Un informe de Gartner³ que coloca la computación en la nube en la parte superior de sus áreas tecnológicas estratégicas, reafirmó aún más su prominencia como una tendencia de la industria al anunciar su definición formal como:

"... un estilo de computación en el que las capacidades escalables y elásticas habilitadas por TI son entregadas como un servicio a clientes externos utilizando tecnologías de Internet".

Esta es una ligera revisión de la definición original de Gartner de 2008, en la que se usó "masivamente escalable" en lugar de "escalables y elásticas". Esto reconoce la importancia de la escalabilidad en relación con la capacidad de escalar verticalmente y no solo en proporciones enormes.

Forrester Research⁴ proporcionó su propia definición de computación en la nube como:

"... una capacidad de TI estandarizada (servicios, software o infraestructura) entregada a través de tecnologías de Internet en una forma de pago por uso y autoservicio".

La definición que recibió la aceptación de toda la industria fue elaborada por el Instituto Nacional de Estándares y Tecnología (NIST). NIST publicó su definición original en 2009, seguida de una versión revisada después de una revisión adicional y aportes de la industria que se publicó en septiembre de 2011:

"La computación en la nube es un modelo para proporcionar el acceso por una red bajo demanda, de manera conveniente y ubicua a un pool⁵ compartido de recursos informáticos configurables (por ejemplo, redes, servidores, almacenamiento, aplicaciones y servicios) que se pueden aprovisionar y liberar rápidamente con un mínimo esfuerzo de administración o interacción con el proveedor de"

² Un end-point o punto final de comunicación es un tipo de nodo de red de comunicación. Es una interfaz expuesta por un comunicante o un canal de comunicación. Fuente: Wikipedia

³ Gartner Inc. es una empresa consultora y de investigación de las tecnologías de la información con sede en Stamford, Connecticut, Estados Unidos. Fuente: Wikipedia

⁴ Forrester es una empresa de investigación y asesoría que ofrece una variedad de servicios que incluyen investigación, consultoría y eventos. Forrester tiene su sede en Cambridge, MA y Londres. Fuente: Wikipedia

⁵ Un pool o fondo en informática es un conjunto de recursos inicializados que se mantienen listos para su uso, en lugar de ser asignados y destruidos bajo demanda. Un cliente del pool solicitará un objeto del pool y realizará operaciones en el objeto devuelto. Cuando el cliente ha terminado con un objeto (o recurso), lo devuelve al pool, en lugar de destruirlo. Fuente: Wikipedia

servicios. Este modelo de nube se compone de cinco características esenciales, tres modelos de servicio y cuatro modelos de implementación”.

Este libro proporciona una definición más concisa:

“La computación en la nube es una forma especializada de computación distribuida⁶ que presenta modelos de utilización para el aprovisionamiento remoto de recursos escalables y medidos”.

Esta definición simplificada está en línea con todas las variaciones de definición anteriores que fueron presentadas por otras organizaciones dentro de la industria de la computación en la nube. Las características, los modelos de servicio y los modelos de implementación a los que se hace referencia en la definición del NIST se tratan con más detalle en el siguiente capítulo.

Innovaciones tecnológicas

Las tecnologías establecidas se utilizan a menudo como inspiración y, a veces, son los cimientos reales sobre los que se derivan y construyen las nuevas innovaciones tecnológicas. Esta sección describe brevemente las tecnologías preexistentes que se consideran las influencias principales en la computación en la nube.

Clustering

Un clúster es un grupo de recursos de TI independientes que están interconectados y funcionan como un solo sistema. Las tasas de falla del sistema se reducen mientras que la disponibilidad⁷ y la confiabilidad⁸ aumentan, ya que la redundancia⁹ y failover¹⁰ (comutación por error) son inherentes al clúster.

Un prerequisito general del clustering de hardware es que los sistemas que lo componen tengan hardware y sistemas operativos razonablemente idénticos para proporcionar niveles de rendimiento similares cuando un componente falle y deba ser reemplazado por otro. Los

⁶ La computación distribuida es un paradigma de computación en el que múltiples dispositivos o computadoras trabajan juntos para resolver un problema o llevar a cabo una tarea. En lugar de tener una sola computadora que realice todas las operaciones, la computación distribuida divide el trabajo entre varias máquinas interconectadas para aumentar la eficiencia y la capacidad de procesamiento. Fuente: ChatGPT

⁷ La disponibilidad es la proporción de tiempo que un sistema está en condiciones de funcionamiento. Esto a menudo se describe como una tasa capaz de misión. Por ejemplo, una unidad que puede usarse 100 horas por semana (168 horas totales) tendría una disponibilidad de $100/168 = 0.5952$ (o 59.52%). Fuente: Wikipedia

⁸ La confiabilidad se define como la probabilidad de que un bien funcione adecuadamente durante un período determinado bajo condiciones operativas específicas. Fuente: Wikipedia

⁹ Los sistemas redundantes, en ingeniería de cómputo, son aquellos en los que se repiten aquellos datos o hardware de carácter crítico que se quiere asegurar ante los posibles fallos que puedan surgir por su uso continuado. Fuente: Wikipedia

¹⁰ Traducida al español como comutación por error, es la capacidad de un sistema de seguir funcionando, aún en caso de producirse algún fallo en el sistema. Fuente: Wikipedia. Se usa la palabra comutación porque en general se usa la comutación para evitar el error por ejemplo al reenviar paquetes por otra ruta usando un comutador de paquetes o al conmutar a otro soporte físico replicado cuando falla el principal. N del T.

dispositivos componentes que forman un clúster se mantienen sincronizados a través de redes de comunicación dedicados de alta velocidad.

El concepto básico de redundancia integrada y failover es fundamental para las plataformas en la nube.

Grid Computing

Una grid computing (malla de cómputo) proporciona una plataforma en la que los recursos informáticos se organizan en uno o más pools lógicos. Estos pools se coordinan colectivamente para proporcionar una malla distribuida de alto rendimiento, a veces denominada "supercomputadora virtual". La malla de cómputo se diferencia del clustering en que los sistemas de malla están débilmente acoplados. Como resultado, los sistemas de computación grid pueden involucrar recursos de cómputo que son heterogéneos (los nodos tienen diversas arquitecturas y sistemas operativos) y están dispersos geográficamente, lo que generalmente no es posible con los sistemas basados en clúster.

Grid computing ha sido un área de investigación en curso en las ciencias de la computación desde principios de la década de 1990. Los avances tecnológicos logrados por los proyectos de computación en malla han influido en varios aspectos de plataformas y mecanismos de computación en la nube, específicamente en relación con conjuntos de características comunes como el acceso a la red, el pooling de recursos, la escalabilidad¹¹ y la resiliencia. Estos tipos de características pueden establecerse tanto por la computación grid como por la computación en la nube, en sus propios enfoques distintivos.

Por ejemplo, Grid Computing se basa en una capa de middleware¹² que se implementa en los recursos informáticos. Estos recursos de TI participan en un grid pool que implementa una serie de funciones de distribución y coordinación de la workload¹³ (carga de trabajo). Esta capa intermedia puede contener la lógica del balanceo de carga, controles de failover y la administración de la configuración automática, cada una de los cuales inspiró previamente tecnologías de computación en la nube similares y otras más sofisticadas. Es por esta razón que algunos clasifican la computación en la nube como descendiente de iniciativas anteriores de grid computing.

Virtualización

La virtualización representa una plataforma tecnológica utilizada para la creación de instancias virtuales de recursos de TI. Una capa de virtualización por software permite que los recursos de TI físicos proporcionen múltiples imágenes virtuales de sí mismos para que varios usuarios puedan compartir las capacidades de procesamiento subyacentes.

¹¹ La escalabilidad, es la propiedad deseable de un sistema, una red o un proceso, que indica su habilidad para reaccionar y adaptarse sin perder calidad, o bien manejar el crecimiento continuo de trabajo de manera fluida, o bien para estar preparado para hacerse más grande sin perder calidad en los servicios ofrecidos. Fuente: Wikipedia

¹² Es un software que asiste a una aplicación para interactuar o comunicarse con otras aplicaciones, o paquetes de programas, redes, hardware o sistemas operativos subyacentes. Este simplifica el trabajo de los programadores en la compleja tarea de generar las conexiones y sincronizaciones que son necesarias en los sistemas distribuidos. Fuente: Wikipedia

¹³ El término "workload" se refiere a la cantidad de trabajo o la demanda de procesamiento que se impone a un sistema informático o a una red. Fuente: ChatGPT

Antes de la llegada de las tecnologías de virtualización, el software se limitaba a residir y combinarse con entornos de hardware estáticos. El proceso de virtualización elimina esta dependencia de software y hardware, ya que los requisitos de hardware pueden simularse mediante el software de emulación que se ejecuta en entornos virtualizados.

Las tecnologías de virtualización se pueden observar en diversos mecanismos de computación en la nube, habiendo inspirado muchas de sus características principales. A medida que evolucionó la computación en la nube, surgió una generación de tecnologías de virtualización modernas para superar las limitaciones de rendimiento, confiabilidad y escalabilidad de las plataformas de virtualización tradicionales.

1.2. Conceptos básicos y terminología

Esta sección establece un conjunto de términos básicos que representan los conceptos y aspectos fundamentales relacionados con la noción de nube.

Nube

Una nube se refiere a un entorno de TI que está diseñado con el propósito de aprovisionar de forma remota recursos de TI escalables y medidos. El término se originó como una metáfora de Internet, que es, en esencia, una red de redes que proporciona acceso remoto a un conjunto de recursos de TI descentralizados. Antes de que la computación en la nube se convirtiera en su propio segmento de la industria de TI, el símbolo de una nube se usaba comúnmente para representar Internet en una variedad de especificaciones y documentación principal de arquitecturas basadas en la Web. Este mismo símbolo ahora se usa para representar específicamente el límite de un entorno de nube, como se muestra en la Figura 1.1.

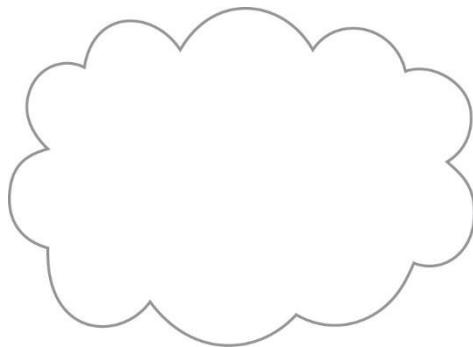


Figura 1.1 El símbolo utilizado para denotar el contorno de un ambiente cloud.

Es importante distinguir el término "nube" y el símbolo de la nube de Internet. Como un entorno específico que se utiliza para aprovisionar recursos de TI de forma remota, una nube tiene un límite finito. Hay muchas nubes individuales a las que se puede acceder a través de Internet. Mientras que Internet brinda acceso abierto a muchos recursos de TI basados en la Web, una nube generalmente es de propiedad privada y ofrece acceso a recursos de TI medidos.

Gran parte de Internet está dedicada al acceso a recursos de TI basados en contenido¹⁴ publicados a través de la World Wide Web. Los recursos de TI proporcionados por los entornos de nube, por

¹⁴ Los recursos de TI basados en contenido se refieren a las herramientas y tecnologías utilizadas para crear, gestionar, almacenar y distribuir contenido digital en diferentes formatos. Estos recursos permiten a las

otro lado, están dedicados a proporcionar capacidades de procesamiento de back-end y un acceso basado en el usuario a estas capacidades. Otra distinción clave es que no es necesario que las nubes estén basadas en la Web, aunque normalmente se basan en protocolos y tecnologías de Internet. Los protocolos se refieren a estándares y métodos que permiten que las computadoras se comuniquen entre sí de una manera predefinida y estructurada. Una nube puede basarse en el uso de cualquier protocolo que permita el acceso remoto a sus recursos de TI.

Recurso de TI

Un recurso de TI es un artefacto físico o virtual relacionado con TI que puede estar basado en software, como un servidor virtual o un programa de software específico, o basado en hardware, como un servidor físico o un dispositivo de red (Figura 1.2).

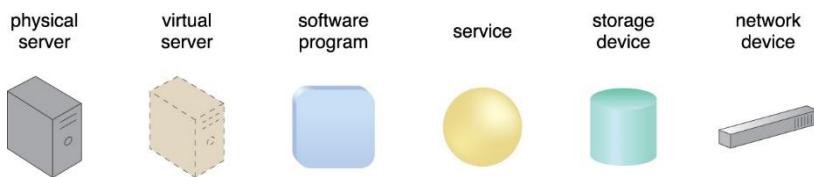


Figura 1.2 Ejemplos comunes de recursos TI y sus correspondientes símbolos.

La Figura 1.3 ilustra cómo se puede usar el símbolo de la nube para definir un límite para un entorno basado en la nube que aloja y aprovisiona un conjunto de recursos de TI. Por lo tanto, los recursos de TI mostrados se consideran recursos de TI basados en la nube.

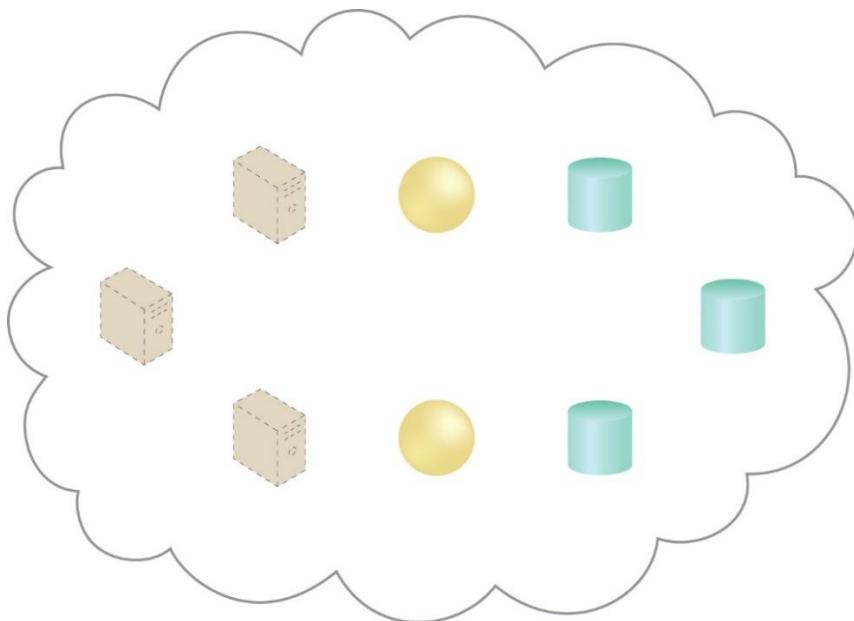


Figura 1.3 Una nube hospedando ocho recursos TI: tres servidores virtuales, dos servicios cloud, y tres dispositivos de almacenamiento.

organizaciones y personas administrar y compartir información, documentos, multimedia y otros tipos de contenido de manera eficiente y efectiva. Algunos ejemplos son repositorios de archivos, bibliotecas digitales, herramientas de edición y diseño, CMS, DMS y plataformas de distribución de contenido. Fuente: ChatGPT.

Arquitecturas tecnológicas así como varios escenarios de interacción que involucran recursos de TI se ilustran en diagramas como el que se muestra en la Figura 1.3. Es importante tener en cuenta los siguientes puntos al estudiar y trabajar con estos diagramas:

- Los recursos de TI que se muestran dentro de los límites de un símbolo de nube dado generalmente no representan todos los recursos de TI disponibles alojados por esa nube. Los subconjuntos de recursos de TI generalmente se resaltan para demostrar un tema en particular.
- Centrarse en los aspectos relevantes de un tema requiere que muchos de estos diagramas proporcionen intencionalmente vistas abstractas de las arquitecturas tecnológicas subyacentes. Esto significa que solo se muestra una parte de los detalles técnicos reales.

Además, algunos diagramas mostrarán recursos de TI fuera del símbolo de la nube. Esta convención se utiliza para indicar recursos de TI que no están basados en la nube.

On-Premise

Como un entorno aparte y accesible de forma remota, una nube representa una opción para la implementación de recursos de TI. Se considera que un recurso de TI alojado en una empresa de TI convencional dentro de un límite organizativo (que no representa específicamente una nube) está ubicado en las instalaciones de la empresa de TI, o **on-premise** para abreviar. En otras palabras, el término "on-premise" es otra forma de decir "en las instalaciones de un entorno de TI controlado que no está basado en la nube". Este término se utiliza para calificar un recurso de TI como una alternativa a "basado en la nube". Un recurso de TI on-premise no puede estar basado en la nube y viceversa.

Tenga en cuenta los siguientes puntos clave:

- Un recurso de TI on-premise puede acceder e interactuar con un recurso de TI basado en la nube.
- Un recurso de TI on-premise se puede mover a una nube, cambiándolo así a un recurso de TI basado en la nube.
- Pueden existir implementaciones redundantes de un recurso de TI tanto en entornos on-premise como basados en la nube.

Consumidores de la nube y proveedores de la nube

La parte que proporciona los recursos de TI basados en la nube es el **cloud provider** (proveedor de la nube). La parte que utiliza los recursos de TI basados en la nube es el **cloud consumer** (consumidor de la nube). Estos términos representan los roles que suelen asumir las organizaciones en relación con las nubes y los correspondientes contratos de aprovisionamiento de nubes.

Escalamiento

El escalamiento, desde la perspectiva de los recursos de TI, representa la capacidad del recurso de TI para manejar demandas de uso que aumentan o se reducen. Los siguientes son tipos de escalamiento:

- Escalamiento horizontal.
- Escalamiento vertical.

A continuación, se describe cada uno de ellos.

Escalamiento horizontal

La asignación o liberación de recursos de TI que son del mismo tipo se conoce como escalamiento horizontal (Figura 1.4). La asignación horizontal de recursos se denomina scaling out y la liberación horizontal de recursos se denomina scaling in. El escalamiento horizontal es una forma común de escalamiento dentro de los entornos de nube.

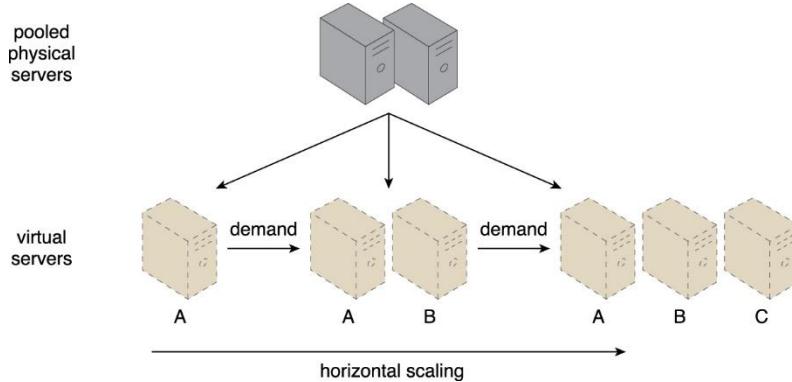


Figura 1.4 Un recurso de TI (servidor virtual A) es scaled out al añadir otro recurso idéntico (servidores virtuales B y C)

Escalamiento vertical

Cuando se reemplaza un recurso de TI existente por otro de mayor o menor capacidad, se considera que se ha producido un escalamiento vertical (Figura 1.5). En concreto, la sustitución de un recurso de TI por otro de mayor capacidad se denomina scaling up y la sustitución de un recurso de TI por otro de menor capacidad se considera scaling down. El escalamiento vertical es menos común en entornos de nube debido al tiempo de inactividad requerido mientras se realiza el reemplazo.

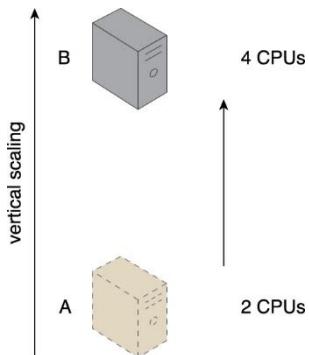


Figura 1.5 Un recurso de TI (un servidor virtual con dos CPUs) es scaled up al reemplazarlo con un recurso de TI más poderoso el cual incrementa la capacidad para el manejo de datos (un servidor físico con cuatro CPUs).

Escalamiento Horizontal	Escalamiento Vertical
Menos costoso (a través de componentes de hardware básicos)	Más costoso (servidores especializados)
Recursos de TI disponibles al instante	Recursos de TI normalmente disponibles al instante
replicación de recursos y escalado automatizado	normalmente se necesita una configuración adicional
se necesitan recursos de TI adicionales	no se necesitan recursos de TI adicionales
no limitado por la capacidad del hardware	limitado por la capacidad máxima del hardware

Tabla 1.1 Muestra un resumen de pros y contras asociados con el escalamiento horizontal y vertical.

Cloud Service

Aunque una nube es un entorno accesible de forma remota, no todos los recursos de TI que residen dentro de una nube pueden estar disponibles para el acceso remoto. Por ejemplo, es posible que una base de datos o un servidor físico desplegado dentro de una nube puedan ser accesibles solo por otros recursos de TI que se encuentren dentro de la misma nube. Un programa de software con una API¹⁵ publicada puede ser desplegada específicamente para permitir el acceso a los clientes remotos.

Un **cloud service** (servicio en la nube) es cualquier recurso de TI al que se puede acceder de forma remota a través de una nube. A diferencia de otros campos de TI que se encuentran bajo el ámbito de la tecnología de servicios, como la arquitectura orientada a servicios, el término "servicio" en el contexto de la computación en la nube es especialmente amplio. Un servicio en la nube puede existir como un simple programa de software basado en la Web con una interfaz invocada mediante el uso de un protocolo de mensajería o como un punto de acceso remoto para herramientas administrativas o entornos más grandes y otros recursos de TI.

En la Figura 1.6, el símbolo del círculo amarillo se usa para representar el servicio en la nube como un programa de software simple basado en Web. Se puede utilizar un símbolo de recurso de TI diferente en este último caso, dependiendo de la naturaleza del acceso proporcionado por el servicio en la nube.

¹⁵ Una API (del inglés, Application Programming Interface) es una pieza de código que permite a diferentes aplicaciones comunicarse entre sí y compartir información y funcionalidades. Por ejemplo, si se tiene una app para móviles acerca de recetas y se hace una búsqueda de una determinada receta, se puede utilizar una API para que esta aplicación se comunique con el sitio web de recetas y pida las recetas que cumplen con los criterios de búsqueda. La API entonces se encarga de recibir la solicitud, buscar las recetas apropiadas y regresar los resultados a la aplicación. Fuente: Wikipedia.

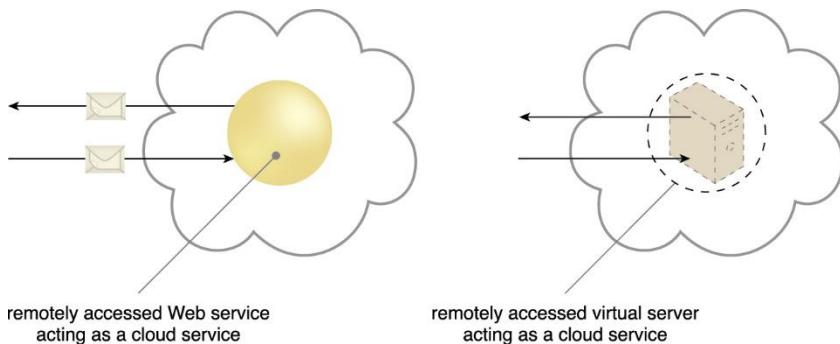


Figura 1.6 Un cloud service con una interfaz publicada siendo accedido por un consumidor fuera de la nube (izquierda). Un cloud service que existe como un servidor virtual está también siendo accedido desde afuera de la frontera cloud (derecha). El cloud service en la izquierda es invocado por un programa consumidor que fue diseñado para acceder gracias a la interfaz publicada. El cloud service en la derecha puede ser accedido por un usuario humano que se ha logueado de manera remota al servidor virtual.

El motor impulsor detrás de la computación en la nube es proporcionar recursos de TI como servicios que encapsulan otros recursos de TI, al tiempo que ofrecen funciones para que los clientes las usen y aprovechen de manera remota. Ha surgido una multitud de modelos para tipos genéricos de servicios en la nube, la mayoría de los cuales están etiquetados con el sufijo "as-a-service".

Nota

Las condiciones de uso del servicio en la nube generalmente se expresan en un service-level agreement (**SLA**) o acuerdo de nivel de servicio que es la parte legible por humanos de un contrato de servicio entre un proveedor de la nube y un consumidor de la nube que describe las características, los comportamientos y las limitaciones de QoS¹⁶ de un servicio basado en la nube u otros suministros.

Un SLA proporciona detalles de varias características medibles relacionadas con los resultados de TI, como el tiempo de actividad, las características de seguridad y otras características específicas de QoS, incluida la disponibilidad, la confiabilidad y el rendimiento. Dado que la implementación de un servicio está oculta para el consumidor de la nube, un SLA se convierte en una especificación crítica.

Consumidor de servicios en la nube

El consumidor de servicios en la nube es un rol temporal en tiempo de ejecución asumido por un programa de software cuando accede a un servicio en la nube.

¹⁶ La calidad de servicio (en inglés quality of service o QoS) es el rendimiento promedio de una red de telefonía o de computadoras, particularmente el rendimiento visto por los usuarios de la red. Cuantitativamente mide la calidad de los servicios que son considerados en varios aspectos del servicio de red, tales como tasas de errores, ancho de banda, rendimiento, retraso en la transmisión, disponibilidad, fluctuación del retardo o jitter, etc. Fuente: Wikipedia

Como se muestra en la Figura 1.7, los tipos comunes de consumidores de servicios en la nube pueden incluir programas de software y servicios capaces de acceder de manera remota a los servicios de la nube con contratos de servicio publicados, así como estaciones de trabajo, portátiles y dispositivos móviles que ejecutan software capaz de acceder de forma remota a otros recursos de TI desplegados como cloud services.



Figura 1.7 Ejemplos de consumidores de servicios en la nube. Dependiendo del diagrama especificado, un artefacto etiquetado como consumidor de servicio en la nube puede ser un programa o un dispositivo de hardware (en cuyo caso se asume que está ejecutando un programa capaz de actuar como un consumidor de servicio en la nube).

1.3. Metas y Beneficios

Los beneficios comunes asociados con la adopción de la computación en la nube se explican en esta sección.

Reducción de Inversiones y Costos Proporcionales

Al igual que un mayorista de productos que compran mercancías a granel a precios más bajos, los proveedores de nube pública basan su modelo comercial en la adquisición masiva de recursos de TI que luego se ponen a disposición de los consumidores de la nube a través de paquetes de arrendamiento a precios atractivos. Esto abre la puerta para que las organizaciones obtengan acceso a una infraestructura poderosa sin tener que comprarla ellos mismos.

La justificación económica más común para invertir en recursos de TI basados en la nube está en la reducción o eliminación total de las inversiones iniciales en TI, es decir, las compras de hardware y software y los costos de propiedad. La característica de uso medido de una nube representa un conjunto de funciones que permite que los gastos operativos medidos (directamente relacionados con el rendimiento comercial) reemplacen los gastos de capital anticipados. Esto también se conoce como costos proporcionales.

Esta eliminación o minimización de los compromisos financieros iniciales permite a las empresas comenzar poco a poco y, en consecuencia, aumentar la asignación de recursos de TI según sea necesario. Además, la reducción de los gastos de capital iniciales permite redirigir el capital a la inversión empresarial principal. En su forma más básica, las oportunidades para reducir costos se derivan de la implementación y operación de centros de datos a gran escala por parte de los principales proveedores de la nube. Estos centros de datos suelen estar ubicados en destinos donde los bienes raíces, los profesionales de TI y el ancho de banda de la red se pueden obtener a costos más bajos, lo que resulta en ahorros tanto de capital como operativos.

El mismo razonamiento se aplica a los sistemas operativos, el middleware o el software de plataforma y el software de aplicación. Los recursos de TI agrupados están disponibles y son compartidos por múltiples consumidores de la nube, lo que resulta en una mayor o incluso máxima utilización posible.

Los beneficios medibles comunes para los consumidores de la nube incluyen:

- Acceso sobre pedido a recursos informáticos de pago por uso a corto plazo (como procesadores por hora) y la capacidad de liberar estos recursos informáticos cuando no se necesiten por más tiempo.
- La percepción de tener recursos informáticos ilimitados que están disponibles bajo demanda, lo que reduce la necesidad de prepararse para el aprovisionamiento.
- La capacidad de agregar o eliminar recursos de TI en un nivel detallado, como modificar el espacio de almacenamiento disponible en el disco en incrementos de un solo gigabyte.
- Abstracción de la infraestructura para que las aplicaciones no estén bloqueadas en dispositivos o ubicaciones y se puedan mover fácilmente si es necesario.

Por ejemplo, una empresa con tareas considerables centradas en lotes puede completarlas tan rápido como su software de aplicación pueda escalar. Usar 100 servidores por una hora cuesta lo mismo que usar un servidor por 100 horas. Esta "elasticidad" de los recursos de TI, lograda sin necesidad de grandes inversiones iniciales para crear una infraestructura informática a gran escala, puede ser extremadamente atractiva.

A pesar de la facilidad con la que muchos identifican los beneficios financieros de la computación en la nube, los ahorros reales pueden ser complejos de calcular y evaluar. La decisión de proceder con una estrategia de adopción de la computación en la nube implicará mucho más que una simple comparación entre el costo de arrendamiento y el costo de compra. Por ejemplo, también se deben tener en cuenta los beneficios financieros del escalamiento dinámico y la transferencia de riesgos tanto del sobreaprovisionamiento (infrautilización) como del infraabastecimiento (sobreutilización).

Mayor escalabilidad

Al proporcionar grupos de recursos de TI, junto con herramientas y tecnologías diseñadas para aprovecharlos colectivamente, las nubes pueden asignar recursos de TI de manera instantánea y dinámica a los consumidores de la nube, bajo demanda o mediante la configuración directa del consumidor de la nube. Esto permite a los consumidores de la nube escalar sus recursos de TI basados en la nube para adaptarse a las fluctuaciones y los picos de procesamiento de forma automática o manual. De manera similar, los recursos de TI basados en la nube se pueden liberar (automática o manualmente) a medida que disminuyen las demandas de procesamiento.

En la Figura 1.8 se proporciona un ejemplo simple de las fluctuaciones de la demanda de uso a lo largo de un período de 24 horas.

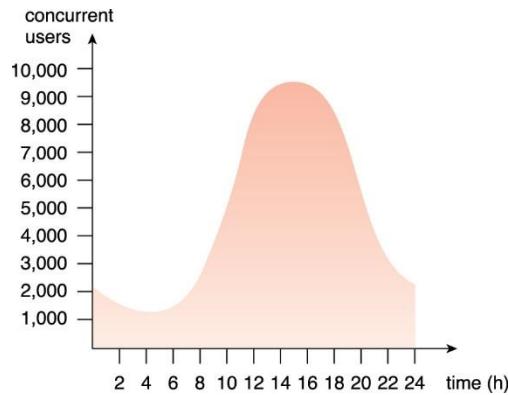


Figura 1.8 Un ejemplo de la demanda cambiante de una organización de un recurso de TI en el transcurso de un día.

Además de la evidente ganancia financiera de la reducción automatizada del escalamiento, la capacidad de los recursos de TI para cumplir siempre con las demandas de uso impredecibles evita la pérdida potencial de negocios que puede ocurrir cuando se alcanzan los umbrales de uso.

Mayor disponibilidad y confiabilidad

La disponibilidad y confiabilidad de los recursos de TI están directamente asociadas con beneficios comerciales tangibles. Las interrupciones limitan el tiempo que un recurso de TI puede estar "open for business" para sus clientes, lo que limita su uso y potencial de generación de ingresos. Las fallas de tiempo de ejecución que no se corrigen de inmediato pueden tener un impacto más significativo durante los períodos de uso de alto volumen. El recurso de TI no solo no puede responder a las solicitudes de los clientes, sino que su falla inesperada puede disminuir la confianza general del cliente.

Una característica distintiva del entorno de nube típico es su capacidad intrínseca para brindar un amplio soporte para aumentar la disponibilidad de un recurso de TI basado en la nube con objeto de minimizar o incluso eliminar las interrupciones, así como para aumentar su confiabilidad a fin de minimizar el impacto de las condiciones de falla en tiempo de ejecución.

Especificamente:

- Se puede acceder a un recurso de TI con mayor disponibilidad durante períodos de tiempo más prolongados (por ejemplo, 22 horas de un día de 24 horas). Los proveedores de la nube generalmente ofrecen recursos de TI "resilientes" para los que pueden garantizar altos niveles de disponibilidad.
- Un recurso de TI con mayor confiabilidad puede evitar y recuperarse mejor de las condiciones excepcionales. La arquitectura modular de los entornos de nube proporciona un amplio soporte de failover que aumenta la confiabilidad.

1.4. Riesgos y desafíos

Se presentan y examinan varios de los desafíos más críticos de la computación en la nube relacionados principalmente con los consumidores de la nube que usan recursos de TI ubicados en nubes públicas.

Mayores vulnerabilidades de seguridad

El traslado de datos comerciales a la nube significa que la responsabilidad sobre la seguridad de los datos se comparte con el proveedor de la nube. El uso remoto de los recursos de TI requiere una expansión de los límites de confianza por parte del consumidor de la nube para incluir la nube externa. Puede ser difícil establecer una arquitectura de seguridad que abarque ese límite de confianza sin introducir vulnerabilidades, a menos que los consumidores de la nube y los proveedores de la nube admitan los mismos marcos de seguridad o compatibles, lo que es poco probable con las nubes públicas.

Otra consecuencia de los límites de confianza superpuestos se relaciona con el acceso privilegiado del proveedor de la nube a los datos del consumidor de la nube. La medida en que los datos están seguros ahora se limita a los controles y políticas de seguridad aplicados tanto por el consumidor de la nube como por el proveedor de la nube. Además, puede haber límites de confianza superpuestos de diferentes consumidores de la nube debido al hecho de que los recursos de TI basados en la nube son comúnmente compartidos.

La superposición de los límites de confianza y la mayor exposición de los datos pueden brindar a los consumidores maliciosos de la nube (humanos y automatizados) mayores oportunidades para atacar los recursos de TI y robar o dañar los datos comerciales. La Figura 1.9 ilustra un escenario en el que dos organizaciones que acceden al mismo servicio en la nube deben extender sus respectivos límites de confianza a la nube, lo que da como resultado límites de confianza superpuestos. Puede ser un desafío para el proveedor de la nube ofrecer mecanismos de seguridad que se adapten a los requisitos de seguridad de ambos consumidores de servicios en la nube.

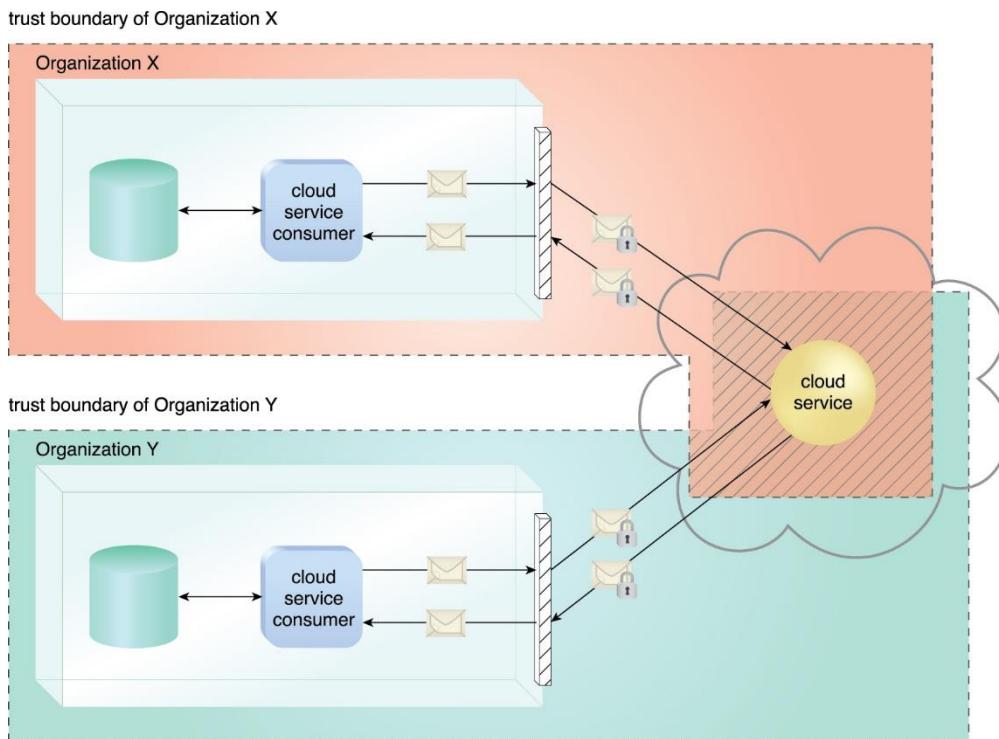


Figura 1.9 El área sombreada con líneas diagonales indica la superposición de los límites de confianza de dos organizaciones. La superposición de límites de confianza es una amenaza a la seguridad que se analiza con más detalle en un capítulo posterior.

Control de gobierno operativo reducido

A los consumidores de la nube generalmente se les asigna un nivel de control de gobierno que es más bajo que el de los recursos de TI on-premise. Esto puede presentar riesgos asociados con la forma en que el proveedor de la nube opera su nube, así como las conexiones externas que se requieren para la comunicación entre la nube y el consumidor de la nube.

Considere los siguientes ejemplos:

- Es posible que un proveedor de nube poco confiable no mantenga las garantías que ofrece en los SLA que se publicaron para sus servicios de nube. Esto puede poner en peligro la calidad de las soluciones de consumo en la nube que dependen de estos servicios en la nube.
- Las distancias geográficas más largas entre el consumidor de la nube y el proveedor de la nube pueden requerir saltos de red adicionales que introducen latencia fluctuante y posibles restricciones de ancho de banda. El último escenario se ilustra en la Figura 1.10.

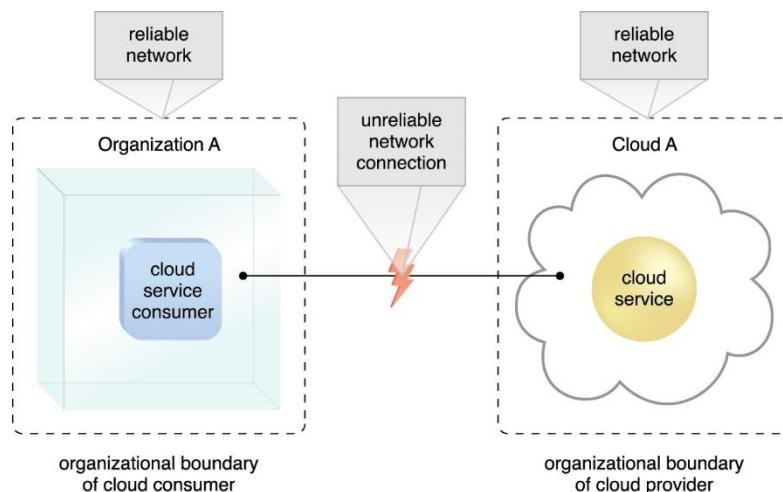


Figura 1.10 Una conexión de red poco confiable compromete la calidad de la comunicación entre el consumidor de la nube y los entornos del proveedor de la nube.

Portabilidad limitada entre proveedores de la nube

Debido a la falta de estándares industriales establecidos dentro de la industria de la computación en la nube, las nubes públicas suelen ser propietarias de varias características y entornos. Para los consumidores de la nube que tienen soluciones personalizadas con dependencias en estos entornos propietarios, puede ser un desafío pasar de un proveedor de la nube a otro.

La portabilidad es una medida que se utiliza para determinar el impacto de mover los recursos de TI y los datos de los consumidores de la nube entre las diversas nubes (Figura 1.11).

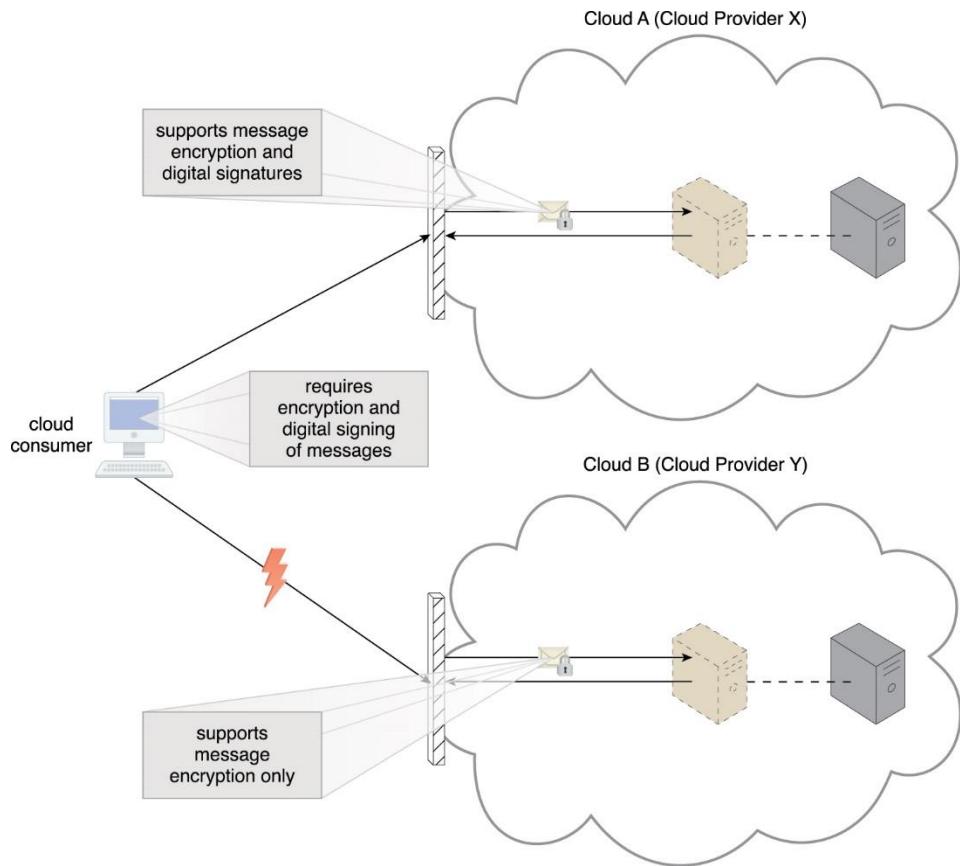


Figura 1.11 La aplicación de un consumidor de la nube tiene un nivel reducido de portabilidad al evaluar una posible migración de la Nube A a la Nube B, porque el proveedor de la Nube B no admite las mismas tecnologías de seguridad que la Nube A.

Cumplimiento multirregional y cuestiones legales

Los proveedores de servicios en la nube de terceros establecerán con frecuencia centros de datos en ubicaciones geográficas asequibles o convenientes. Los consumidores de la nube a menudo no conocen la ubicación física de sus recursos y datos de TI cuando están alojados en nubes públicas. Para algunas organizaciones, esto puede plantear serias preocupaciones legales relacionadas con las regulaciones gubernamentales o de la industria que especifican las políticas de privacidad y almacenamiento de datos. Por ejemplo, algunas leyes del Reino Unido exigen que los datos personales pertenecientes a ciudadanos del Reino Unido se mantengan dentro del Reino Unido.

Otro problema legal potencial se relaciona con la accesibilidad y divulgación de datos. Los países tienen leyes que requieren que algunos tipos de datos se divulguen a ciertas agencias gubernamentales. Por ejemplo, las agencias gubernamentales pueden acceder más fácilmente a los datos de un consumidor de la nube europeo que se encuentra en los EE. UU. (debido a la Ley USA PATRIOT¹⁷) en comparación con los datos ubicados en muchos países de la Unión Europea.

¹⁷ USA PATRIOT Act ; acrónimo de Uniting and Strengthening America by Providing Appropriate Tools Required to Intercept and Obstruct Terrorism Act, es decir «Ley para unir y fortalecer Estados Unidos proveyendo las herramientas apropiadas, requeridas para impedir y obstaculizar el terrorismo». Fuente: Wikipedia.

La mayoría de los marcos regulatorios reconocen que las organizaciones de consumidores de la nube son, en última instancia, responsables de la seguridad, la integridad y el almacenamiento de sus propios datos, incluso cuando están en manos de un proveedor externo de la nube.

2 Conceptos Fundamentales y Modelos



En esta unidad se cubren áreas temáticas introductorias relacionadas con los modelos fundamentales utilizados para categorizar y definir las nubes y sus ofertas de servicios más comunes, junto con definiciones de roles organizacionales y el conjunto específico de características que colectivamente distinguen una nube.

2.1. Roles y límites

Las organizaciones y los humanos pueden asumir diferentes tipos de roles predefinidos según cómo se relacionen o interactúen con una nube y sus recursos de TI alojados. Cada uno de los próximos roles participa y lleva a cabo responsabilidades en relación con la actividad basada en la nube. Las siguientes secciones definen estos roles e identifican sus principales interacciones.

Cloud Provider (Proveedor de la nube)

La organización que proporciona recursos de TI basados en la nube es el *proveedor de la nube*. Al asumir el rol de proveedor de la nube, una organización es responsable de hacer que los servicios de la nube estén disponibles para los consumidores de la nube, según las garantías de SLA acordadas. El proveedor de la nube también tiene la tarea de realizar las tareas administrativas y de gestión necesarias para garantizar el funcionamiento continuo de la infraestructura global de la nube.

Los proveedores de la nube normalmente son propietarios de los recursos de TI en la nube y los ponen a disposición de los consumidores para su alquiler; sin embargo, algunos proveedores de la nube también "revenden" recursos de TI alquilados a otros proveedores de la nube.

Cloud Consumer (Consumidor de la nube)

Un consumidor de la nube es una organización (o un ser humano) que tiene un contrato o acuerdo formal con un proveedor de la nube para utilizar los recursos de TI que el proveedor de la nube pone a su disposición. Específicamente, el consumidor de la nube utiliza un consumidor de servicios en la nube para acceder a un servicio en la nube (Figura 2.1).

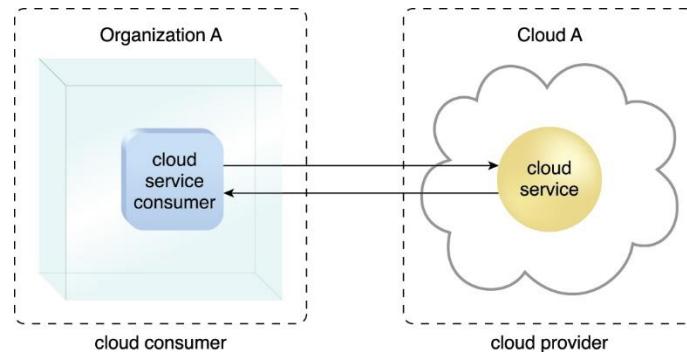


Figura 2.1 Un consumidor de la nube (Organización A) interactúa con un servicio de nube de un proveedor de nube (propietario de la Nube A). Dentro de la Organización A, el consumidor del servicio en la nube se utiliza para acceder al servicio en la nube.

Las figuras de este libro no siempre etiquetan explícitamente los símbolos como "consumidores de la nube". En cambio, generalmente se da a entender que las organizaciones o los seres humanos que acceden de forma remota a los recursos de TI basados en la nube se consideran consumidores de la nube.

Nota

Al representar escenarios de interacción entre los recursos de TI basados en la nube y las organizaciones de consumidores, no hay reglas estrictas sobre cómo se usan los términos "consumidor de servicios en la nube" y "consumidor de la nube" en este libro. El primero generalmente se usa para etiquetar programas de software o aplicaciones que interactúan mediante programación con la API de un servicio en la nube. El último término es más amplio porque se puede usar para etiquetar una organización, un individuo que accede a una interfaz de usuario o un programa de software que asume el rol de consumidor de la nube cuando interactúa con una nube, un recurso de TI basado en la nube o un proveedor de nube. La amplia aplicabilidad del término "consumidor de la nube" es intencional, ya que permite su uso en formas que exploran diferentes tipos de relaciones entre consumidores y proveedores dentro de diferentes contextos técnicos y comerciales.

Propietario del servicio en la nube

La persona u organización que posee legalmente un servicio en la nube se denomina propietario del servicio en la nube. El propietario del servicio en la nube puede ser el consumidor de la nube o el proveedor de la nube que posee la nube en la que reside el servicio en la nube.

Por ejemplo, el consumidor de la nube de Cloud X o el proveedor de la nube de Cloud X podrían ser propietarios del Servicio de la Nube A (Figuras 2.2 y 2.3).

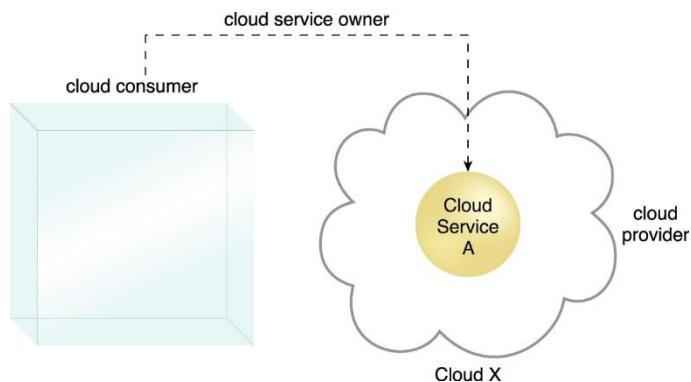


Figura 2.2 Un consumidor de la nube puede ser el propietario de un servicio cloud cuando este despliega su propio servicio en la nube.

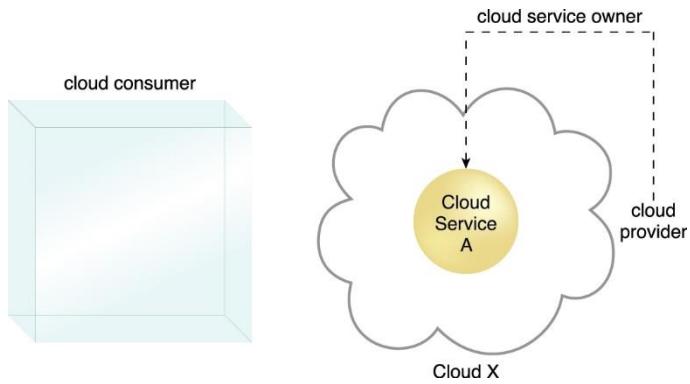


Figura 2.3 Un proveedor de la nube se convierte en propietario de un servicio en la nube si implementa su propio servicio en la nube, normalmente para que lo utilicen otros consumidores de la nube.

Tenga en cuenta que un consumidor de la nube que posee un servicio en la nube alojado por una nube de terceros no necesariamente tiene que ser el usuario (o consumidor) del servicio en la nube. Varias organizaciones de consumidores de nube desarrollan e implementan servicios de nube en nubes propiedad de terceros con el fin de poner los servicios de nube a disposición del público en general.

La razón por la que el propietario de un servicio en la nube no se denomina propietario de un recurso en la nube es que la función de propietario del servicio en la nube solo se aplica a los servicios en la nube (que, como se explica en el Capítulo 1, son recursos de TI accesibles desde el exterior que residen en una nube).

Cloud Resource Administrator

Un cloud resource administrator (administrador de recursos en la nube) es la persona u organización responsable de administrar un recurso de TI basado en la nube (incluidos los servicios en la nube). El administrador de recursos de la nube puede ser (o pertenecer a) el consumidor de la nube o el proveedor de la nube en el que reside el servicio de la nube. Alternativamente, puede ser (o pertenecer a) una organización de terceros contratada para administrar el recurso de TI basado en la nube.

Por ejemplo, el propietario de un servicio en la nube puede contratar a un administrador de recursos en la nube para administrar un servicio en la nube (Figuras 2.4 y 2.5).

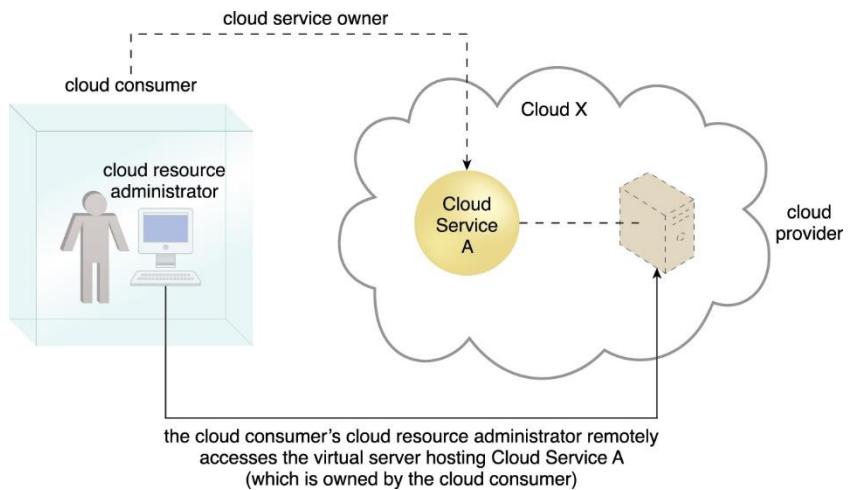


Figura 2.4 Un administrador de recursos de la nube puede estar con una organización consumidora de la nube y administrar recursos IT accesibles de manera remota que pertenecen al consumidor de la nube.

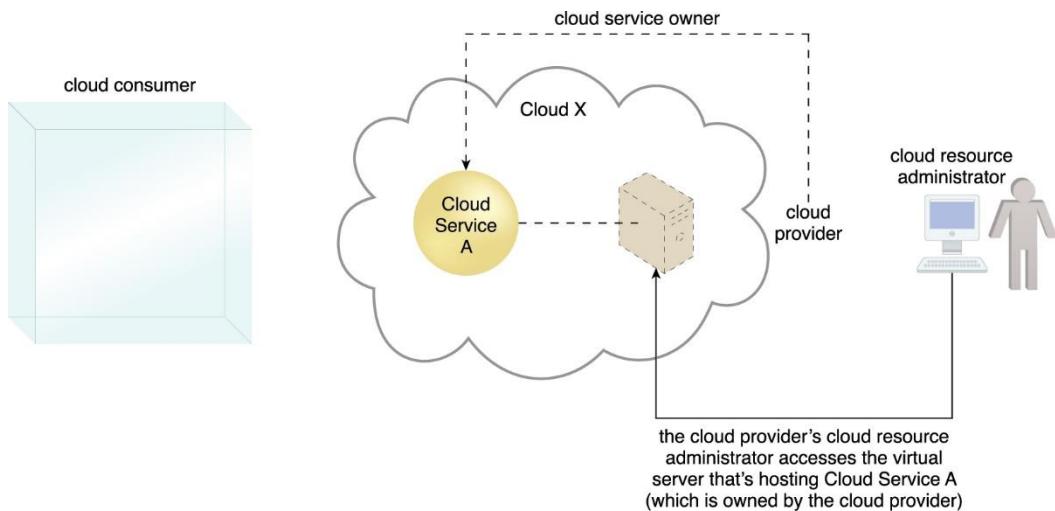


Figura 2.5 Un administrador de recursos de la nube puede estar con una organización proveedora de la nube para la cual puede administrar los recursos de TI disponibles interna y externamente del proveedor de la nube.

La razón por la que no se hace referencia a un administrador de recursos en la nube como "administrador de servicios en la nube" es porque esta función puede ser responsable de administrar los recursos de TI basados en la nube que no existen como servicios en la nube. Por ejemplo, si el administrador de recursos de la nube está contratado por (o pertenece a) el proveedor de la nube, entonces puede administrar los recursos de TI a los que no se puede acceder de forma remota (y estos tipos de recursos de TI no se clasifican como servicios en la nube).

Roles adicionales

La Cloud Computing Reference Architecture del NIST define los siguientes roles complementarios:

- **Cloud Auditor:** un tercero (a menudo acreditado) que realiza evaluaciones independientes de los entornos de la nube asume el rol del cloud auditor. Las responsabilidades típicas asociadas con este rol incluyen la evaluación de los controles de seguridad, los impactos en la privacidad y el rendimiento. El objetivo principal de la función de auditor de la nube es proporcionar una evaluación

imparcial (y una posible aprobación) de un entorno de nube para ayudar a fortalecer la relación de confianza entre los consumidores de la nube y los proveedores de la nube.

- Cloud Broker¹⁸: esta función la asume una parte que toma la responsabilidad de administrar y negociar el uso de servicios en la nube entre los consumidores de la nube y los proveedores de la nube. Los servicios de mediación proporcionados por el cloud broker incluyen servicios de intermediación, y arbitraje.
- Cloud Carrier: la parte responsable de proporcionar la conectividad a nivel de cable entre los consumidores de la nube y los proveedores de la nube asume el rol del cloud carrier. Esta función suele ser asumida por los proveedores de redes y telecomunicaciones.

Límite organizacional

Un límite organizacional representa el perímetro físico que rodea un conjunto de recursos de TI que son propiedad de una organización y que están gobernados por ella. El límite organizacional no representa el límite de una organización real, solo un conjunto organizacional de activos de TI y recursos de TI. De manera similar, las nubes tienen un límite organizacional (Figura 2.6).

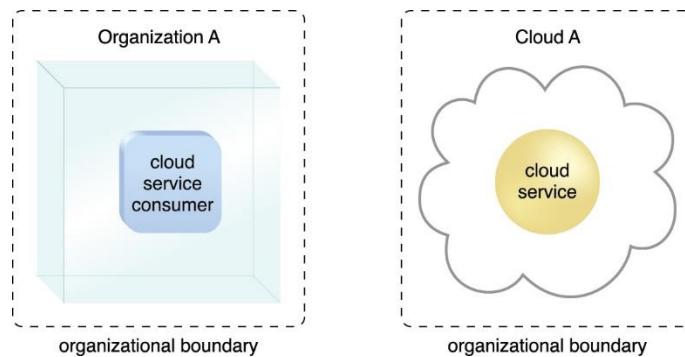


Figura 2.6 Límites organizacionales de un consumidor de nube (izquierda) y un proveedor de nube (derecha), representados por una notación de línea discontinua.

Límite de confianza

Cuando una organización asume el rol de consumidor de la nube para acceder a los recursos de TI basados en la nube, necesita extender su confianza más allá del límite físico de la organización para incluir partes del entorno de la nube. Un límite de confianza es un perímetro lógico que generalmente se extiende más allá de los límites físicos para representar la medida en que se confía en los recursos de TI (Figura 2.7). Al analizar entornos de nube, el límite de confianza se asocia con mayor frecuencia con la confianza emitida por la organización que actúa como consumidor de la nube.

¹⁸ Un broker o intermediario es un individuo o institución que organiza las transacciones entre un comprador y un vendedor en ciertos sectores a cambio de una comisión cuando se ejecute la operación. Fuente: Wikipedia.

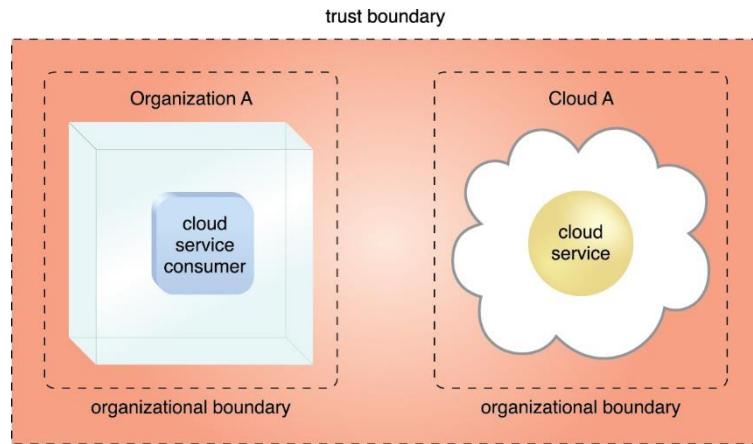


Figura 2.7 Un límite de confianza extendido abarca los límites organizacionales del proveedor de la nube y el consumidor de la nube.

Nota

Otro tipo de límite relevante para los entornos de nube es el perímetro de la red lógica. Este tipo de límite se clasifica como un mecanismo de computación en la nube y se trata en el Capítulo 5.

2.2. Características de la nube

Un entorno de TI requiere un conjunto específico de características para permitir el aprovisionamiento remoto de recursos de TI escalables y medidos de manera efectiva. Estas características deben existir en una medida significativa para que el entorno de TI se considere una nube eficaz.

Las siguientes seis características específicas son comunes a la mayoría de los entornos de nube:

- on-demand usage (uso bajo demanda)
- ubiquitous access (acceso ubicuo)
- multitenancy and resource pooling (tenencia múltiple y pooling de recursos)
- elasticity (elasticidad)
- measured usage (uso medido)
- resiliency (resiliencia)

Los proveedores de la nube y los consumidores de la nube pueden evaluar estas características de forma individual y colectiva para medir el valor que ofrece una determinada plataforma en la nube. Si bien los servicios basados en la nube y los recursos de TI heredarán y exhibirán características individuales en mayor o menor medida, por lo general, cuanto mayor sea el grado en que brinden soporte y se utilicen, mayor será el valor de la propuesta resultante.

Nota

La definición del NIST de computación en la nube define solo cinco características; se excluye la resiliencia. La resiliencia ha surgido como un aspecto de gran importancia y su nivel común de soporte constituye su inclusión necesaria como una característica común de la nube.

Uso bajo demanda

Un consumidor de la nube puede acceder unilateralmente a los recursos de TI basados en la nube, lo que le da la libertad de auto aprovisionarse de estos recursos de TI. Una vez configurado, el uso de los recursos de TI auto aprovisionados se puede automatizar, lo que no requiere más participación humana por parte del consumidor de la nube o del proveedor de la nube. Esto da como resultado un entorno de uso bajo demanda. También conocido como "on-demand self-service usage", esta característica habilita las funciones basadas en servicios y basadas en el uso que se encuentran en las nubes convencionales.

Acceso ubicuo

El acceso ubicuo representa la capacidad de un servicio en la nube para ser ampliamente accesible. Establecer acceso ubicuo para un servicio en la nube puede requerir soporte para una variedad de dispositivos, protocolos de transporte, interfaces y tecnologías de seguridad. Para habilitar este nivel de acceso, generalmente se requiere que la arquitectura del servicio en la nube se adapte a las necesidades particulares de los diferentes consumidores del servicio en la nube.

Multitenancy (Tenencia múltiple)

La característica de un programa de software que permite que una instancia del programa atienda a diferentes consumidores (inquilinos) donde cada uno está aislado de los demás, se denomina *multitenancy*. Un proveedor de la nube agrupa sus recursos de TI para servir a múltiples consumidores de servicios en la nube, mediante el uso de modelos de tenencia múltiple que con frecuencia se basan en el uso de tecnologías de virtualización. Mediante el uso de la tecnología de tenencia múltiple, los recursos de TI se pueden asignar y reasignar dinámicamente, de acuerdo con las demandas de los consumidores de servicios en la nube.

El pooling de recursos permite a los proveedores de la nube hacer un pool de recursos de TI a gran escala para atender a múltiples consumidores de la nube. Los diferentes recursos de TI físicos y virtuales se asignan y reasignan dinámicamente de acuerdo con la demanda de los consumidores de la nube, para continuar con su ejecución mediante un multiplexado basado en estadísticas. El pooling de recursos se logra comúnmente a través de la tecnología multitenancy.

Las figuras 2.8 y 2.9 ilustran la diferencia entre los entornos de single-tenant (un solo inquilino) y de multitenant (múltiples inquilinos).

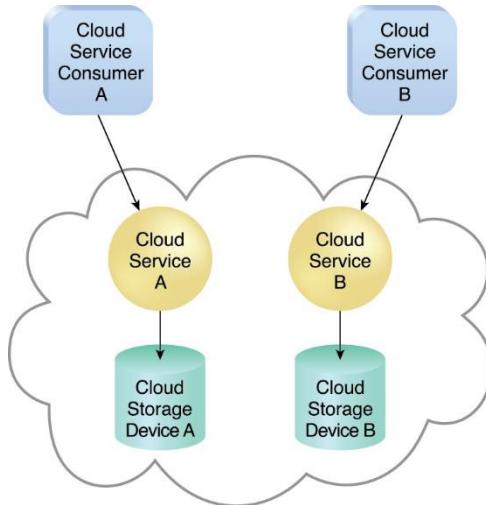


Figura 2.8 En un ambiente single-tenant, cada consumidor de cloud tiene una instancia de recursos IT separada.

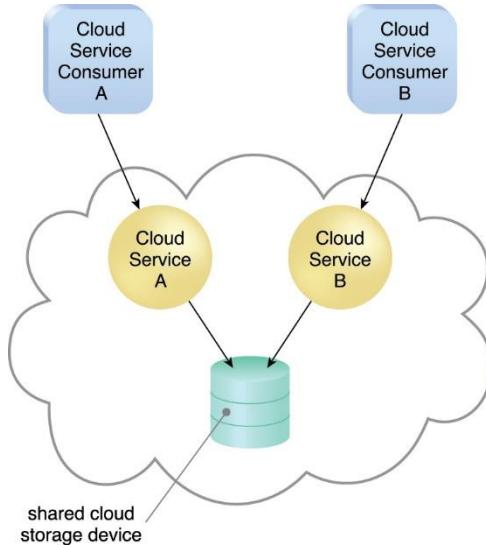


Figura 2.9 En un entorno de multitenant, una sola instancia de un recurso de TI, como podría ser un dispositivo de almacenamiento en la nube, sirve a múltiples consumidores.

Como se ilustra en la Figura 2.9, la multitenancy permite que varios consumidores de la nube usen el mismo recurso de TI o su instancia, mientras que cada uno ignora que otros pueden usarlo.

Elasticidad

La elasticidad es la capacidad automatizada de una nube para escalar de forma transparente¹⁹ los recursos de TI, según sea necesario en respuesta a las condiciones de tiempo de ejecución o según lo predeterminado por el consumidor de la nube o el proveedor de la nube. La elasticidad a menudo se considera una justificación central para la adopción de la computación en la nube, principalmente debido al hecho de que está estrechamente asociada con el beneficio de reducción de inversión y a

¹⁹ La transparencia se define como la ocultación al usuario de las acciones que realiza de manera interna un mecanismo de cómputo. Fuente: Wikipedia.

los costos proporcionales. Los proveedores de la nube con vastos recursos de TI pueden ofrecer la mayor variedad de elasticidad.

Uso medido

La característica de uso medido representa la capacidad de una plataforma en la nube para realizar un seguimiento del uso de sus recursos de TI, principalmente por parte de los consumidores de la nube. Basado en las mediciones, el proveedor de la nube puede cobrarle a un consumidor de la nube solo por los recursos de TI realmente utilizados y/o por el período de tiempo durante el cual se concedió el acceso a los recursos de TI. En este contexto, el uso medido está estrechamente relacionado con la característica on-demand.

El uso medido no se limita al seguimiento de estadísticas con fines de facturación. También abarca la supervisión general de los recursos de TI y los informes de uso relacionados (tanto para el proveedor de la nube como para los consumidores de la nube). Por lo tanto, el uso medido también es relevante para las nubes que no cobran por el uso (lo que puede ser aplicable al modelo de implementación de la nube privada que se describe en la sección *Modelos de implementación de la nube*).

Resiliencia

El cómputo resiliente es una forma de failover que distribuye implementaciones redundantes de recursos de TI en distintas ubicaciones físicas. Los recursos de TI se pueden preconfigurar para que, si uno se vuelve deficiente, el procesamiento se transfiera automáticamente a otra implementación redundante. Dentro de la computación en la nube, la característica de resiliencia puede referirse a recursos de TI redundantes dentro de la misma nube (pero en diferentes ubicaciones físicas) o en múltiples nubes. Los consumidores de la nube pueden aumentar tanto la confiabilidad como la disponibilidad de sus aplicaciones aprovechando la resiliencia de los recursos de TI basados en la nube (Figura 2.10).

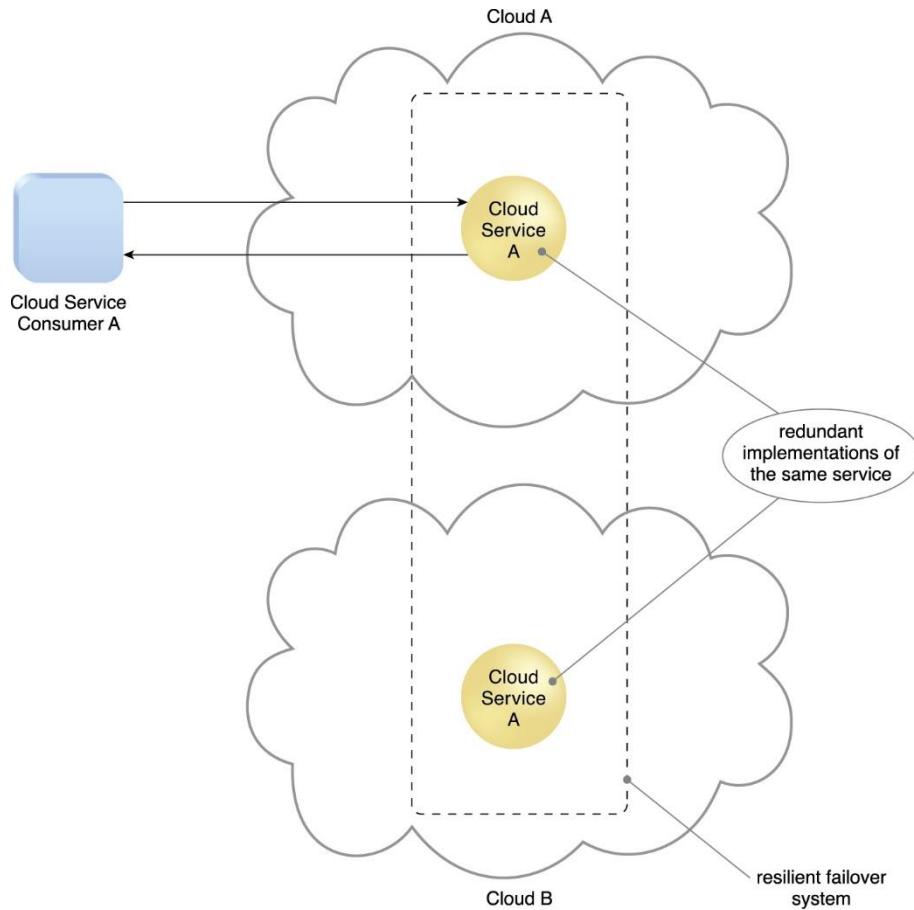


Figura 2.10 Un sistema resiliente en el que la Nube B aloja una implementación redundante del Servicio de la Nube A para proveer failover en caso de que el Cloud Service A en la Nube A deje de estar disponible.

2.3. Modelos de entrega en la nube

Un cloud delivery model(modelo de entrega en la nube) representa una combinación específica y preempaquetada de recursos de TI ofrecidos por un proveedor de la nube. Se han establecido y formalizado ampliamente tres modelos comunes de entrega en la nube:

- Infraestructura como servicio (IaaS)
- Plataforma como servicio (PaaS)
- Software como servicio (SaaS)

Nota

Han surgido muchas variaciones especializadas de los tres modelos básicos de entrega de la nube, cada una compuesta por una combinación distinta de recursos de TI. Algunos ejemplos incluyen:

- Almacenamiento como servicio
- Base de datos como servicio

- Seguridad como servicio
- Comunicación como servicio
- Integración como servicio
- Pruebas como servicio
- Proceso como servicio

Tenga en cuenta también que un modelo de entrega en la nube puede denominarse modelo de entrega de servicios en la nube porque cada modelo se clasifica como un tipo diferente de oferta de servicios en la nube.

Infraestructura como servicio (IaaS)

El modelo de entrega de IaaS representa un entorno de TI autónomo compuesto por recursos de TI centrados en la infraestructura a los que se puede acceder y administrar a través de interfaces y herramientas basados en servicios en la nube. Este entorno puede incluir hardware, red, conectividad, sistemas operativos y otros recursos de TI "raw". A diferencia de los entornos tradicionales de alojamiento o subcontratación, con IaaS, los recursos de TI generalmente se virtualizan en paquetes que simplifican la escalabilidad inicial y la personalización de la infraestructura.

El propósito general de un entorno IaaS es proporcionar a los consumidores de la nube un alto nivel de control y responsabilidad sobre su configuración y utilización. Los recursos de TI proporcionados por IaaS generalmente no están preconfigurados, lo que coloca la responsabilidad administrativa directamente sobre el consumidor de la nube. Por lo tanto, este modelo lo utilizan los consumidores de la nube que requieren un alto nivel de control sobre el entorno basado en la nube que pretenden crear.

A veces, los proveedores de la nube contratarán ofertas de IaaS de otros proveedores de la nube para escalar sus propios entornos de nube. Los tipos y marcas de los recursos de TI proporcionados por los productos IaaS ofrecidos por diferentes proveedores de nube pueden variar. Los recursos de TI disponibles a través de entornos IaaS generalmente se ofrecen como instancias virtuales recién inicializadas. Un recurso de TI central y principal dentro de un entorno típico de IaaS es el servidor virtual. Los servidores virtuales se alquilan especificando los requisitos de hardware del servidor, como la capacidad del procesador, la memoria y el espacio de almacenamiento local, como se muestra en la Figura 2.11.

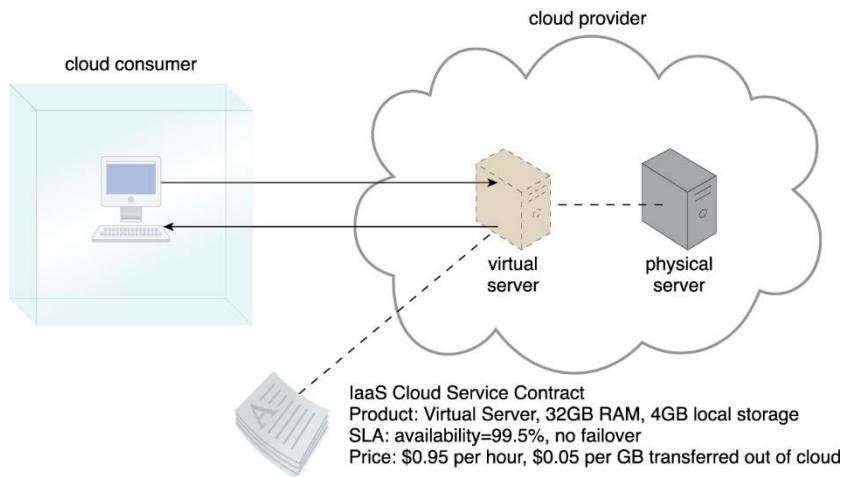


Figura 2.11 Un consumidor de la nube está utilizando un servidor virtual dentro de un entorno IaaS. El proveedor de la nube proporciona a los consumidores de la nube una serie de garantías contractuales, relacionadas con características como la capacidad, el rendimiento y la disponibilidad.

Plataforma como servicio (PaaS)

El modelo de entrega de PaaS representa un entorno predefinido "ready-to-use" que normalmente se compone de recursos de TI ya implementados y configurados. Específicamente, PaaS se basa en (y se define principalmente por) el uso de un entorno listo para usar que establece un conjunto de productos y herramientas preempaquetados que se utilizan para respaldar todo el ciclo de vida de entrega de aplicaciones personalizadas.

Las razones comunes por las que un consumidor de la nube usaría e invertiría en un entorno PaaS incluyen:

- El consumidor de la nube quiere extender los entornos locales a la nube por motivos económicos y de escalabilidad.
- El consumidor de la nube utiliza el entorno ya preparado para sustituir por completo un entorno local.
- El consumidor de la nube desea convertirse en proveedor de la nube e implementa sus propios servicios en la nube para que estén disponibles para otros consumidores de la nube externos.

Al trabajar dentro de una plataforma lista para usar, el consumidor de la nube se ahorra la carga administrativa de configurar y mantener los recursos de TI de la infraestructura básica proporcionados a través del modelo IaaS. Por el contrario, al consumidor de la nube se le otorga un nivel más bajo de control sobre los recursos de TI subyacentes que alojan y aprovisionan la plataforma (Figura 2.12).

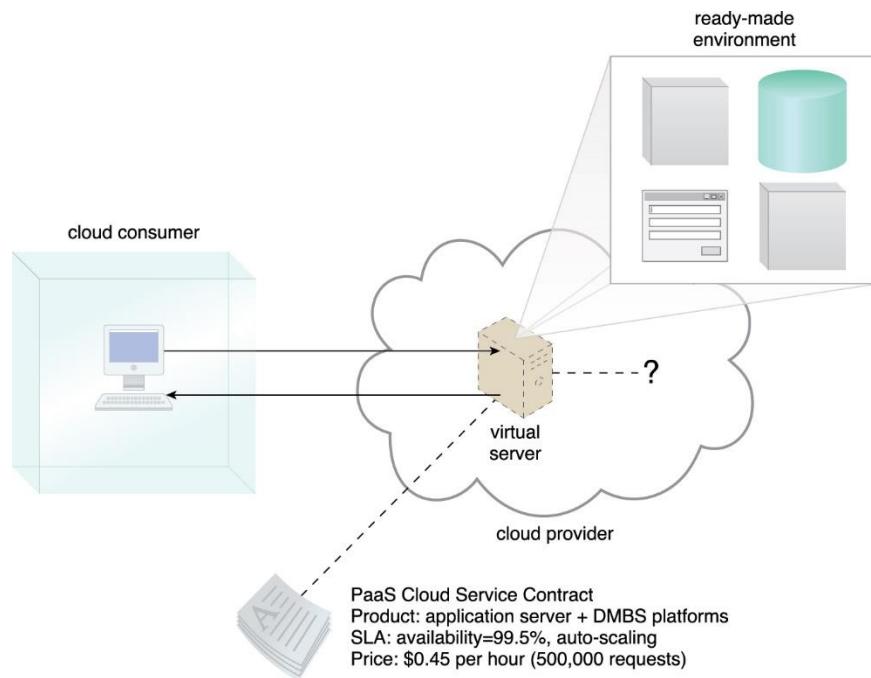


Figura 2.12 Un consumidor de la nube accede a un entorno PaaS listo para usar. El signo de interrogación indica que el consumidor de la nube está intencionalmente protegido de los detalles de implementación de la plataforma.

Los productos PaaS están disponibles con diferentes entornos de desarrollo. Por ejemplo, Google App Engine ofrece un entorno basado en Java y Python.

Software como servicio (SaaS)

Un programa de software expuesto como un servicio de nube compartido, y disponible como un "producto" o una utilidad genérica representa el perfil típico de una oferta de SaaS. El modelo de entrega SaaS generalmente se usa para hacer que un servicio reutilizable en la nube esté ampliamente disponible (a menudo comercialmente) para una variedad de consumidores de la nube. Existe todo un mercado en torno a los productos SaaS que se pueden rentar y utilizar para diferentes propósitos y bajo diferentes términos (Figura 2.13).

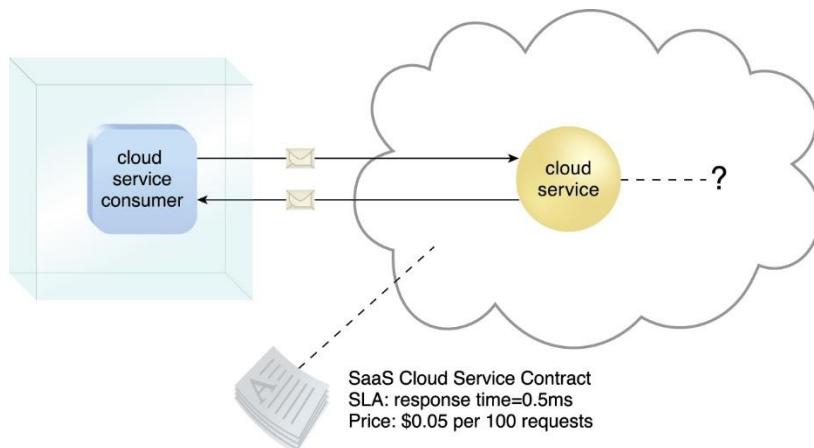


Figura 2.13 El consumidor del servicio en la nube tiene acceso al contrato del servicio en la nube, pero no a los recursos de TI subyacentes ni a los detalles de implementación.

Por lo general, a un consumidor de la nube se le otorga un control administrativo muy limitado sobre una implementación de SaaS. La mayoría de las veces lo proporciona el proveedor de la nube, pero puede ser propiedad legal de cualquier entidad que asuma el rol de propietario del servicio de la nube. Por ejemplo, una organización que actúa como consumidor de la nube mientras usa y trabaja con un entorno PaaS puede crear un nuevo servicio en la nube que decida implementar en ese mismo entorno y ofrecer como SaaS. Luego, la misma organización asume efectivamente el rol de proveedor de la nube, ya que el servicio en la nube basado en SaaS se pone a disposición de otras organizaciones que actúan como consumidores de la nube cuando usan ese nuevo servicio en la nube.

Comparación de modelos de entrega en la nube

En esta sección se proporcionan tres tablas que comparan diferentes aspectos del uso y la implementación del modelo de entrega en la nube. La Tabla 2.1 contrasta los niveles de control y la Tabla 2.2 compara las responsabilidades y el uso típicos.

Modelo de entrega en la nube	Nivel típico de control otorgado al consumidor de la nube	Funcionalidad típica disponible para el consumidor de la nube
SaaS	uso y configuración relacionada con el uso	acceso a la interfaz de usuario front-end
PaaS	administrativo limitado	nivel moderado de control administrativo sobre los recursos de TI relevantes para el uso de la plataforma por parte del consumidor de la nube
IaaS	administrativo completo	acceso total a los recursos de TI relacionados con la infraestructura virtualizada y, posiblemente, a los recursos de TI físicos subyacentes

Tabla 2.1 Una comparación de modelos de control típicos en los modelos de entrega en la nube

Modelo de entrega en la nube	Actividades comunes del consumidor de la nube	Actividades comunes del proveedor de la nube
SaaS	usa y configura el servicio en la nube	implementa, administra y mantiene el servicio en la nube supervisa el uso por parte de los consumidores de la nube
PaaS	desarrolla, prueba, implementa y administra servicios en la nube y soluciones basadas en la nube	preconfigura la plataforma y aprovisiona la infraestructura subyacente, el middleware y otros recursos de TI necesarios, según sea necesario.

		supervisa el uso por parte de los consumidores de la nube
IaaS	instala y configura la infraestructura básica e instala, administra y supervisa cualquier software necesario	aprovisiona y gestiona el procesamiento físico, el almacenamiento, las redes y el alojamiento necesarios supervisa el uso por parte de los consumidores de la nube

Tabla 2.2 Actividades típicas realizadas por los consumidores y proveedores de la nube en relación con los modelos de entrega de la nube.

Combinación de modelos de entrega en la nube

Los tres modelos básicos de entrega de la nube comprenden una jerarquía de aprovisionamiento natural, lo que permite explorar oportunidades para la aplicación combinada de los modelos. Las próximas secciones destacan brevemente las consideraciones relacionadas con dos combinaciones comunes.

IaaS + PaaS

Un entorno PaaS se construye sobre una infraestructura subyacente comparable a los servidores físicos y virtuales y otros recursos de TI proporcionados en un entorno IaaS. La Figura 2.14 muestra cómo estos dos modelos pueden combinarse conceptualmente en una arquitectura simple en capas.

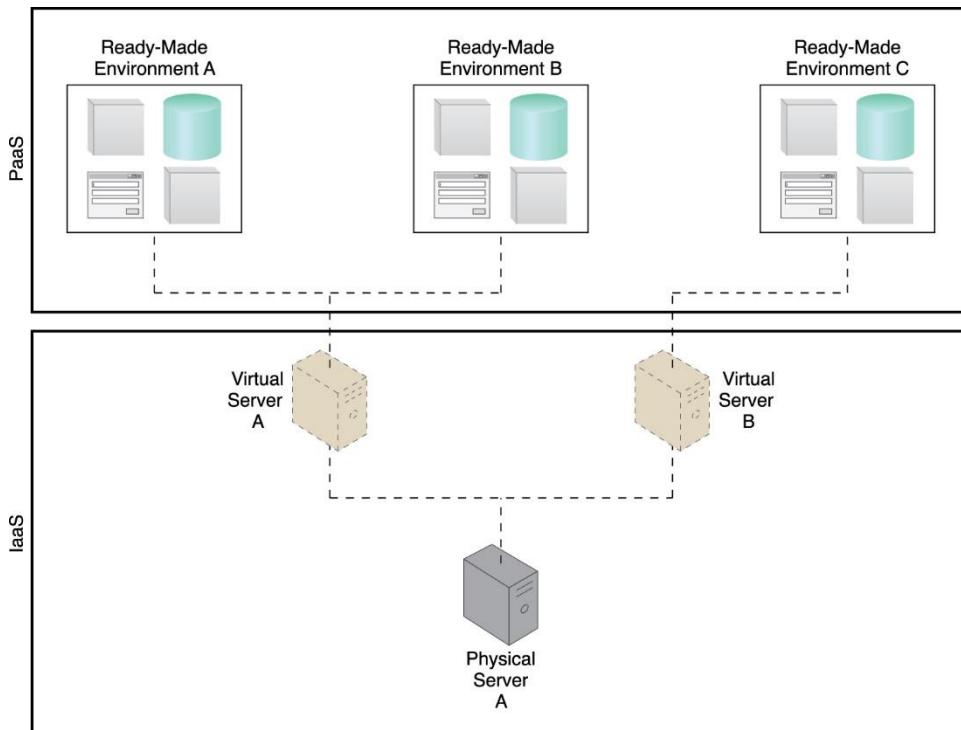


Figura 2.14 Un entorno PaaS basado en los recursos de TI proporcionados por un entorno IaaS subyacente.

Normalmente, un proveedor de la nube no necesitaría proveer un entorno IaaS desde su propia nube para que un entorno PaaS esté disponible a los consumidores de la nube. Entonces, ¿cómo sería útil o aplicable la arquitectura proporcionada por la Figura 2.14? Supongamos que el proveedor de la nube que ofrece el entorno PaaS eligió arrendar un entorno IaaS de un proveedor de la nube diferente.

La motivación para tal arreglo puede estar influenciada por la economía o tal vez porque el primer proveedor de la nube está cerca de exceder su capacidad existente al atender a otros consumidores de la nube. O quizás un consumidor de nube en particular impone un requisito legal para que los datos se almacenen físicamente en una región específica (diferente de donde reside la nube del primer proveedor de nube), como se ilustra en la Figura 2.15.

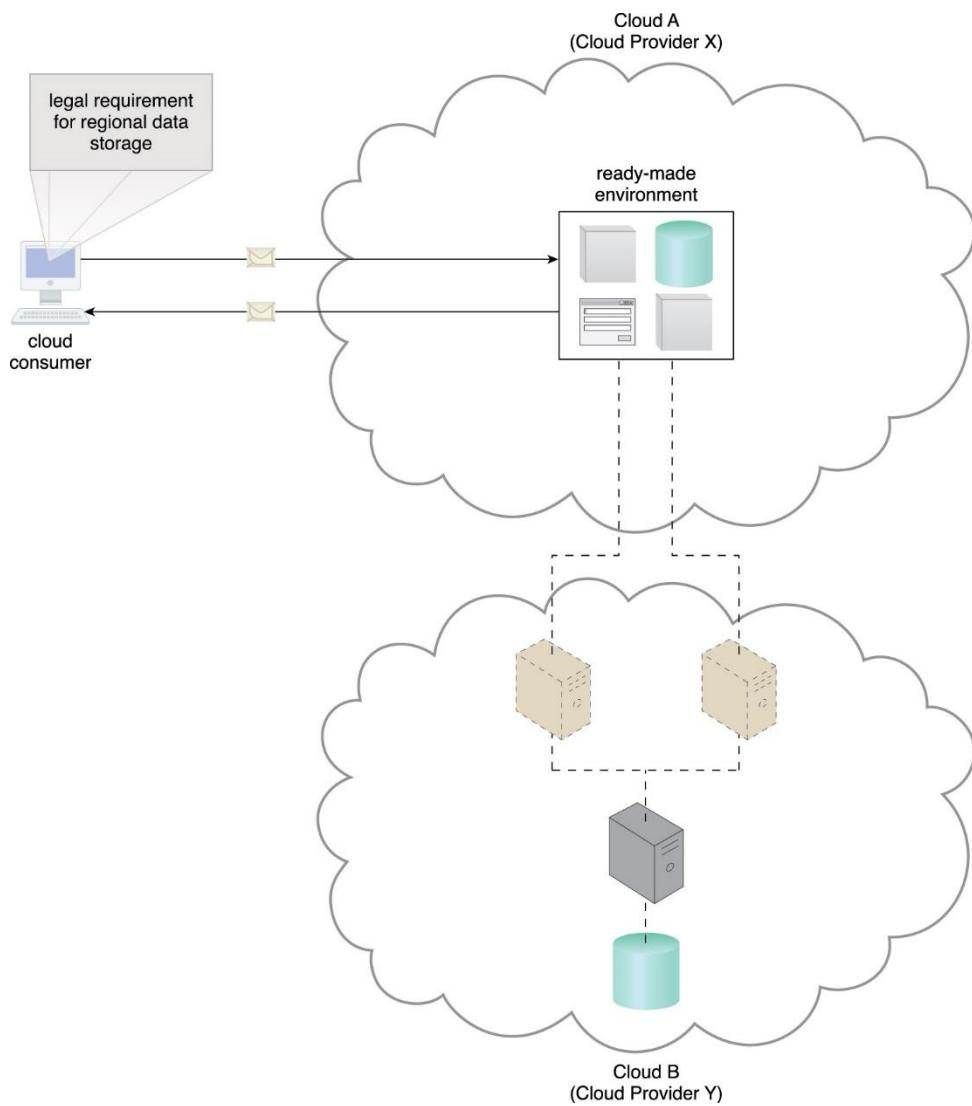


Figura 2.15 Un ejemplo de un contrato entre los proveedores de la nube X e Y, en el que los servicios ofrecidos por el proveedor de la nube X se alojan físicamente en servidores virtuales que pertenecen al

proveedor de la nube Y. Los datos confidenciales que están legalmente obligados a permanecer en una región específica se mantienen físicamente en Cloud B, que se encuentra físicamente en esa región.

IaaS + PaaS + SaaS

Los tres modelos de entrega en la nube se pueden combinar para establecer capas de recursos de TI que se construyen una sobre la otra. Por ejemplo, al agregar a la arquitectura en capas anterior que se muestra en la Figura 2.15, y usar el entorno proporcionado por PaaS para desarrollar e implementar otros servicios en la nube SaaS que luego se pueden hacer disponibles como productos comerciales. (Figura 2.16).

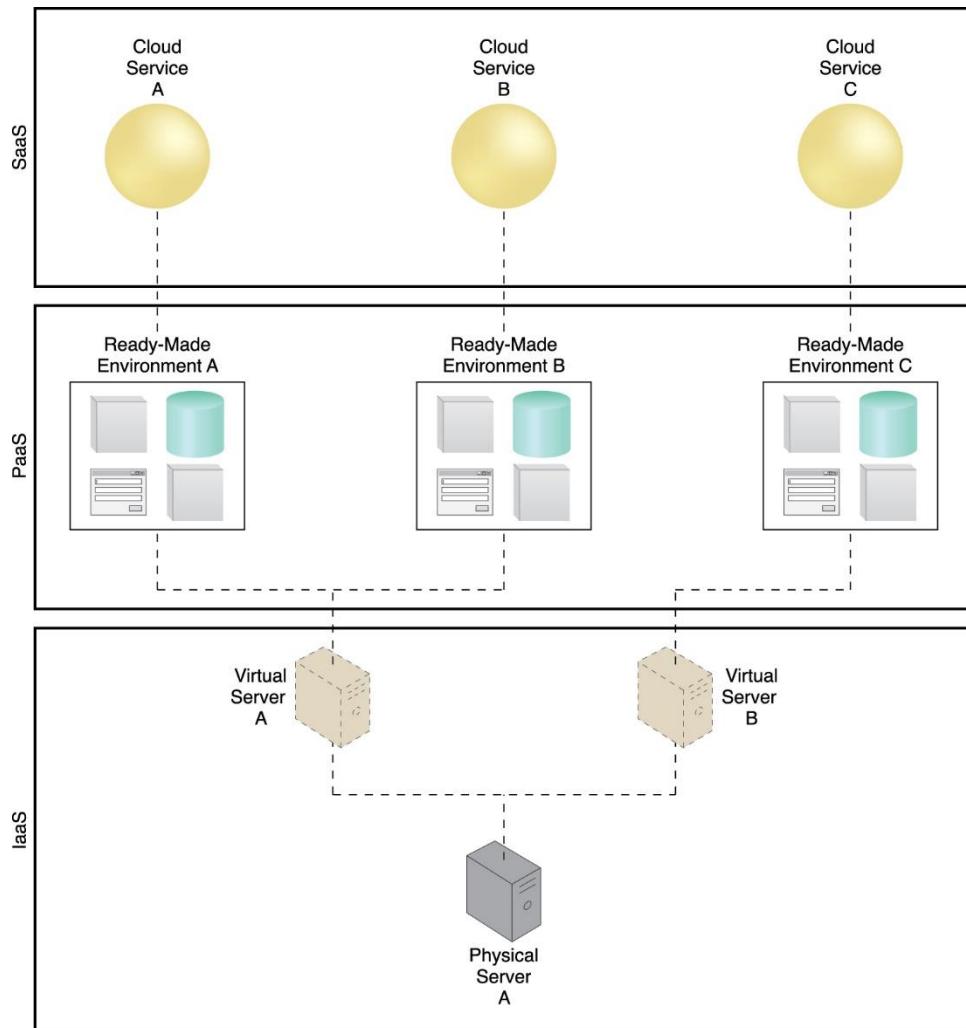


Figura 2.16 Una vista simple en capas de una arquitectura compuesta por entornos IaaS y PaaS que alojan tres implementaciones de servicios en la nube SaaS.

2.4. Modelos de implementación en la nube

Un modelo de implementación en la nube representa un tipo específico de entorno en la nube, que se distingue principalmente por el propietario, el tamaño y el acceso. Hay cuatro modelos comunes de implementación de la nube:

- Nube pública

- Nube comunitaria
- Nube privada
- Nube híbrida

Las siguientes secciones describen cada una.

Nubes Públicas

Una nube pública es un entorno de nube de acceso público propiedad de un proveedor de nube externo. Los recursos de TI en las nubes públicas generalmente se aprovisionan a través de los modelos de entrega en la nube descritos antes, y generalmente se ofrecen a los consumidores de la nube a un costo o se comercializan a través de otras vías (como la publicidad).

El proveedor de la nube es responsable de la creación y el mantenimiento continuo de la nube pública y sus recursos de TI.

La Figura 2.17 muestra una vista parcial del panorama de la nube pública y destaca algunos de los principales proveedores del mercado.

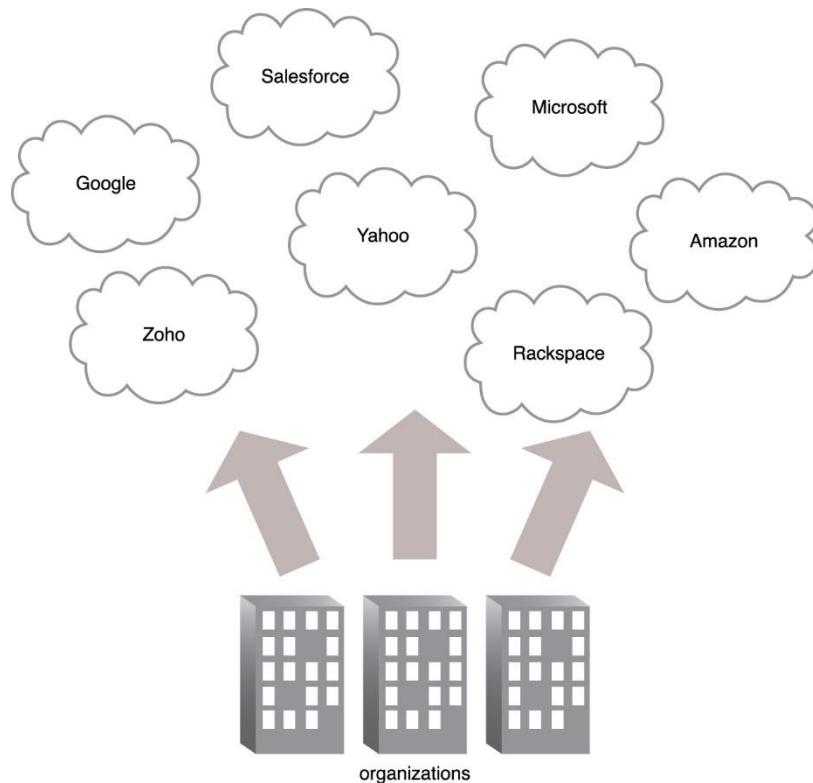


Figura 2.17 Las organizaciones actúan como consumidores de la nube cuando acceden a los servicios de la nube y los recursos de TI disponibles a través de diferentes proveedores de la nube.

Nubes comunitarias

Una nube comunitaria es similar a una nube pública excepto que su acceso está limitado a una comunidad específica de consumidores de la nube. La nube comunitaria puede ser propiedad conjunta de los miembros de la comunidad o de un proveedor de nube externo que aprovisione una

nube pública con acceso limitado. Los consumidores de la nube miembros de la comunidad suelen compartir la responsabilidad de definir y hacer evolucionar la nube de la comunidad (Figura 2.18).

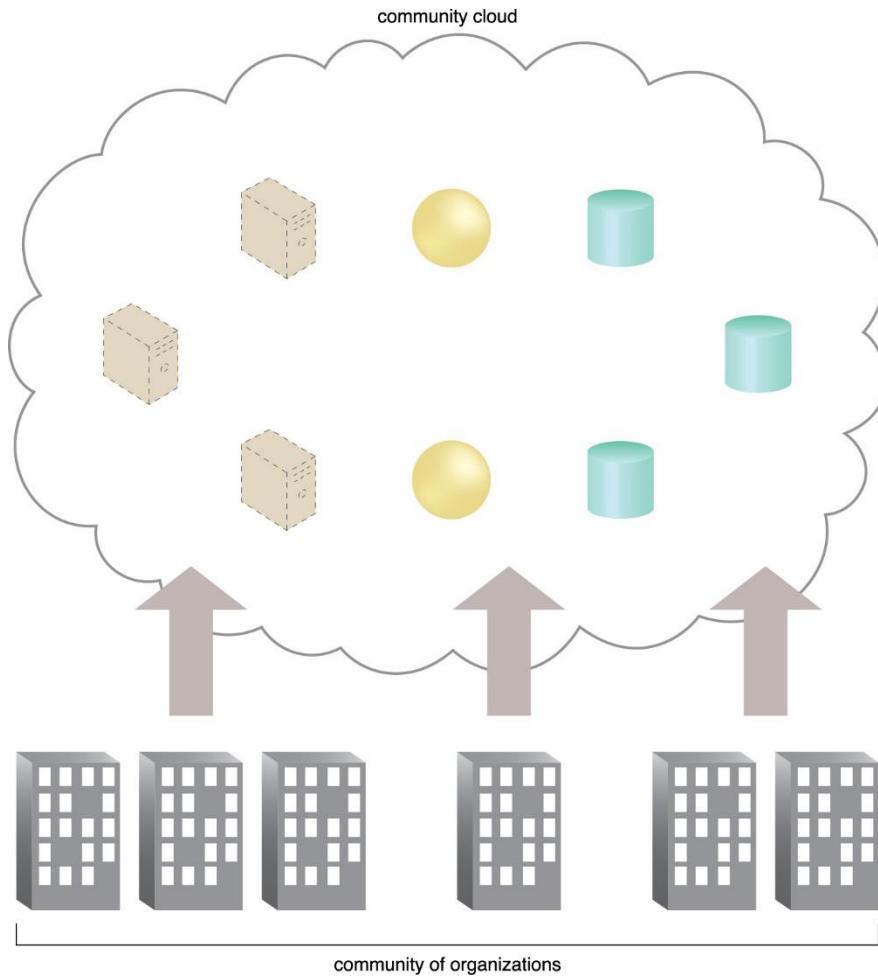


Figura 2.18 Ejemplo de una "comunidad" de organizaciones que acceden a los recursos de TI desde una nube comunitaria.

La pertenencia a la comunidad no garantiza necesariamente el acceso o el control de todos los recursos de TI de la nube. A las partes fuera de la comunidad generalmente no se les otorga acceso a menos que la comunidad lo permita.

Nubes Privadas

Una nube privada es propiedad de una sola organización. Las nubes privadas permiten que una organización use la tecnología de computación en la nube como un medio para centralizar el acceso a los recursos de TI por parte de diferentes partes, ubicaciones o departamentos de la organización.

El uso de una nube privada puede cambiar la forma en que se definen y aplican los límites organizacionales y de confianza. La administración propiamente dicha de un entorno de nube privada puede ser realizada por personal interno o externo.

Con una nube privada, la misma organización es técnicamente tanto el consumidor de la nube como el proveedor de la nube (Figura 2.19). Es posible diferenciar estos roles como sigue:

- un departamento organizativo separado generalmente asume la responsabilidad de aprovisionar la nube (y, por lo tanto, asume el rol de proveedor de la nube)
- los departamentos que requieren acceso a la nube privada asumen el rol de consumidor de la nube

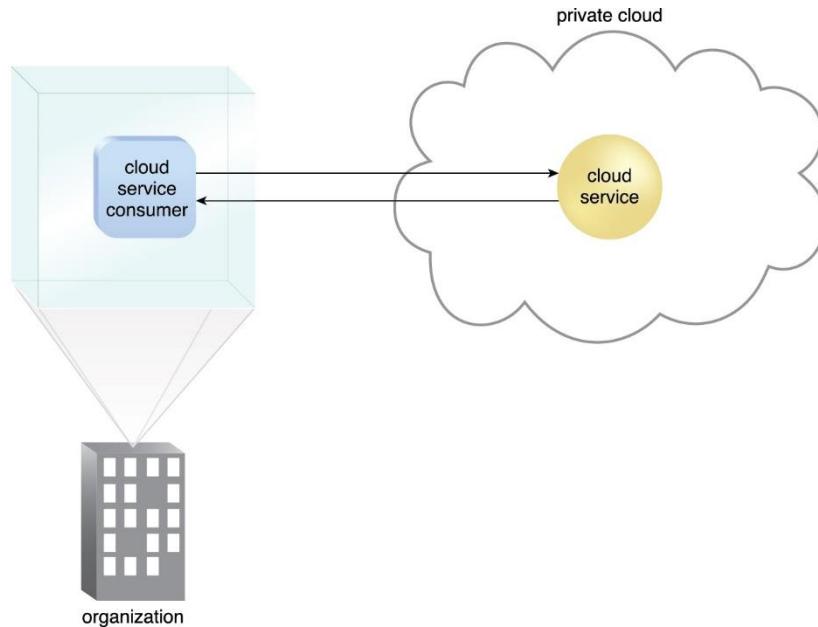


Figura 2.19 Un consumidor de servicios en la nube en el entorno local de la organización accede a un servicio en la nube alojado en la nube privada de la misma organización a través de una red privada virtual.

Es importante utilizar correctamente los términos "on-premise"(en las instalaciones) y "cloud-based" en el contexto de una nube privada. Aunque la nube privada puede residir físicamente en las instalaciones de la organización, los recursos de TI que aloja se siguen considerando "cloud-based" siempre que los consumidores de la nube puedan acceder a ellos de forma remota.

Los recursos de TI alojados fuera de la nube privada por los departamentos que actúan como consumidores de la nube se consideran, por lo tanto, "on-premise" en relación con los recursos de TI privados cloud-based.

Nubes híbridas

Una nube híbrida es un entorno de nube compuesto por dos o más modelos de implementación de nube diferentes. Por ejemplo, un consumidor de la nube puede optar por implementar servicios en la nube que procesan datos confidenciales en una nube privada y otros servicios en la nube menos confidenciales en una nube pública. El resultado de esta combinación es un modelo de despliegue híbrido (Figura 2.20).

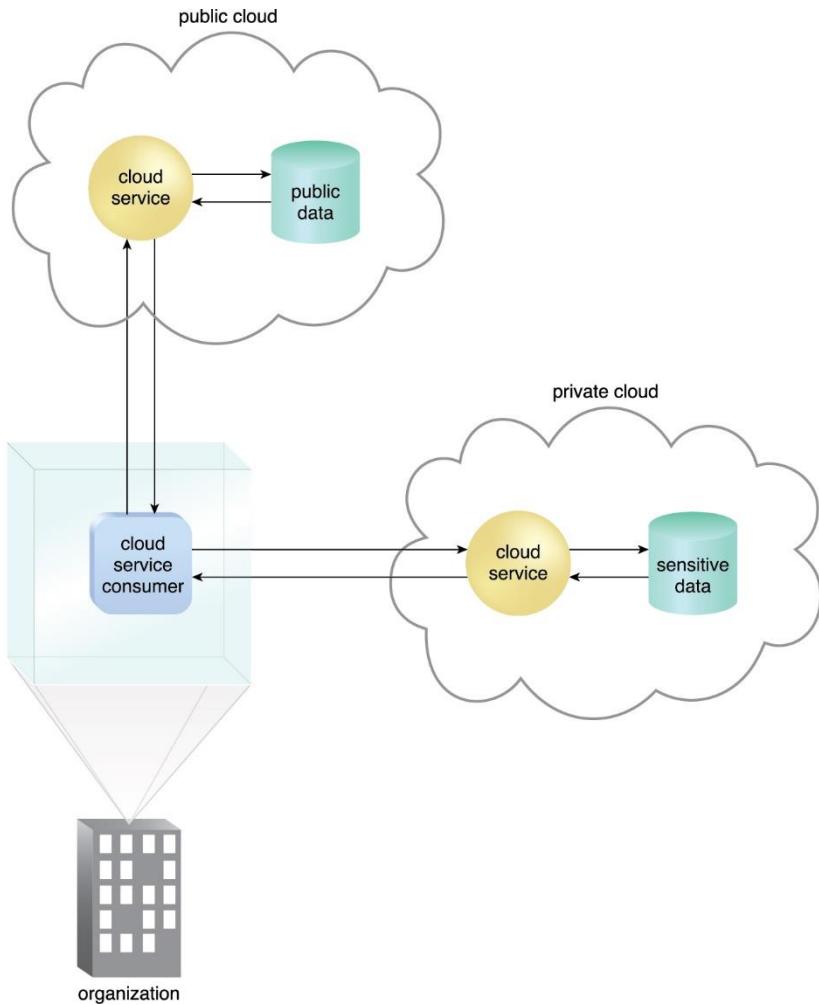


Figura 2.20 Una organización que usa una arquitectura de nube híbrida la cual utiliza tanto una nube privada como una pública.

Las arquitecturas de implementación híbrida pueden ser complejas y difíciles de crear y mantener debido a la disparidad potencial en los entornos de nube y al hecho de que las responsabilidades de administración generalmente se dividen entre la organización del proveedor de nube privada y el proveedor de nube pública.

Otros modelos de implementación en la nube

Pueden existir variaciones adicionales de los cuatro modelos básicos de implementación en la nube. Los ejemplos incluyen:

- *Nube privada virtual*: también conocida como "nube dedicada" o "nube alojada": este modelo da como resultado un entorno de nube autónomo alojado y administrado por un proveedor de nube pública, y puesto a disposición de un consumidor de nube.
- *Inter-Cloud*: este modelo se basa en una arquitectura compuesta por dos o más nubes interconectadas.

3 Tecnologías para la nube



Las nubes modernas están respaldadas por un conjunto de tecnologías base que, en conjunto, permiten funciones y características clave asociadas con la informática en la nube contemporánea. Las siguientes tecnologías de este tipo se tratan en este capítulo:

- Redes de banda ancha y arquitectura de Internet
- Tecnología de centro de datos
- Tecnología de virtualización
- Tecnología web
- Tecnología multitenant
- Containerization

Cada una existía y maduraba antes de la llegada de la computación en la nube, aunque los avances de la computación en la nube ayudaron a evolucionarlas aún más.

3.1. Redes de Banda Ancha y Arquitectura de Internet

Todas las nubes deben estar conectadas a una red. Este requisito inevitable forma una dependencia inherente a la interconexión de redes.

Internet, permite el aprovisionamiento remoto de recursos de TI y apoya directamente el acceso a la red ubicua. Los consumidores de la nube tienen la opción de acceder a la nube usando solo enlaces de red privados y dedicados en LAN, aunque la mayoría de las nubes están habilitadas para Internet. Por lo tanto, el potencial de las plataformas en la nube generalmente crece en paralelo con los avances en la conectividad a Internet y la calidad del servicio.

Proveedores de servicios de Internet (ISP)

Establecidas y desplegadas por los ISP, las redes troncales más grandes de Internet están estratégicamente interconectadas por enrutadores centrales que conectan las redes multinacionales del mundo. Como se muestra en la figura 3.1, una red ISP se interconecta con otras redes ISP y varias organizaciones.

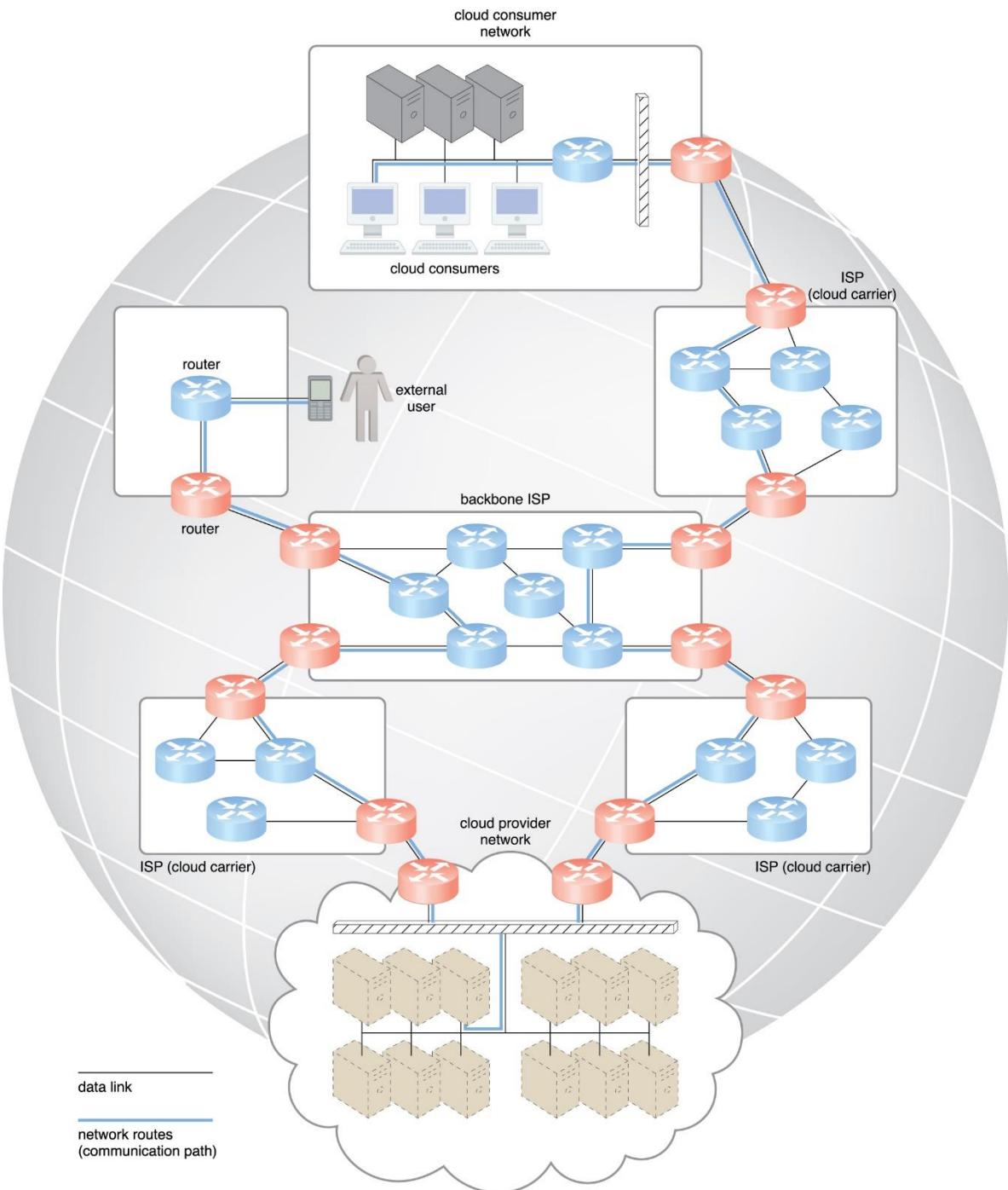


Figura 3.1 Los mensajes viajan a través de rutas de red dinámicas en esta configuración de interconexión de redes de ISPs.

El concepto de Internet se basaba en un modelo de provisión y gestión descentralizado. Los ISPs pueden implementar, operar y administrar libremente sus redes, además de seleccionar ISPs asociados para la interconexión. Ninguna entidad centralizada gobierna Internet de manera integral, aunque organismos como la Corporación de Internet para la Asignación de Nombres y Números (ICANN) supervisan y coordinan las comunicaciones de Internet.

Las leyes gubernamentales y reglamentos dictan las condiciones de prestación de servicios para las organizaciones y los ISPs, tanto dentro como fuera de las fronteras nacionales. Ciertos dominios de Internet aún requieren la demarcación de la jurisdicción nacional y los límites legales.

La topología de Internet se ha convertido en un agregado dinámico y complejo de ISPs que están altamente interconectados a través de sus propios protocolos. Ramas más pequeñas se extienden desde estos nodos principales de interconexión, ramificándose hacia el exterior a través de redes más pequeñas hasta llegar finalmente a todos los dispositivos electrónicos habilitados para Internet.

La conectividad mundial se habilita a través de una topología jerárquica compuesta por los Niveles 1, 2 y 3 (Figura 3.2). El nivel 1 principal está formado por proveedores de nube internacionales a gran escala que manejan las redes globales masivas interconectadas, las cuales están conectadas a los grandes proveedores regionales del nivel 2. Los ISPs interconectados del Nivel 2 se conectan con los proveedores del Nivel 1, así como los ISPs locales del Nivel 3. Los consumidores de la nube y los proveedores de la nube pueden conectarse directamente usando un proveedor del Nivel 1, ya que cualquier ISP operativo puede habilitar la conexión a Internet.

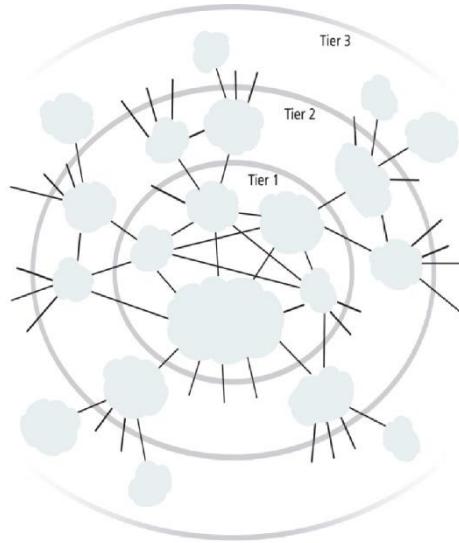


Figura 3.2 Una abstracción de la estructura de interconexión de Internet.

Los enlaces de comunicación y los enrutadores de Internet y las redes ISP son recursos de TI que se distribuyen entre innumerables rutas generadoras de tráfico. Dos componentes fundamentales son utilizados para construir la arquitectura de interconexión de redes: la connectionless packet switching (datagram networks) y la router-based interconnectivity.

Conectionless Packet Switching (Datagram Networks)

Los flujos de datos de extremo a extremo (par emisor-receptor) se dividen en paquetes de un tamaño limitado que se reciben y procesan a través de comutadores y enrutadores de red, luego se encolan y se reenvían de un nodo intermedio al siguiente. Cada paquete lleva la información de ubicación necesaria, como el Protocolo de Internet (IP) o la dirección de Control de acceso a medios (MAC), para ser procesada y enrutada en cada nodo de origen, intermedio y destino.

Router-Based Interconnectivity

Un router es un dispositivo que está conectado a múltiples redes a través de las cuales reenvía paquetes. Incluso cuando los paquetes sucesivos son parte del mismo flujo de datos, los enruteadores procesan y reenvían cada paquete individualmente mientras mantienen la información de topología de red que ubica el siguiente nodo en la ruta de comunicación entre los nodos de origen y destino. Los enruteadores administran el tráfico de la red y miden el salto más eficiente para la entrega de paquetes, ya que conocen tanto el origen como el destino del paquete.

La mecánica básica de la interconexión se ilustra en la Figura 3.3, en la que se fusiona un mensaje a partir de un grupo entrante de paquetes desordenados. El enruteador representado recibe y reenvía paquetes de múltiples flujos de datos.

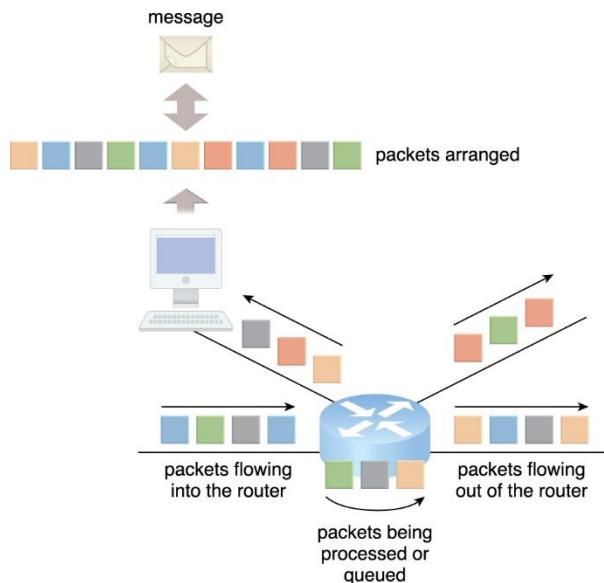


Figura 3.3 Los paquetes que viajan a través de Internet son dirigidos por un enruteador que los organiza en un mensaje.

La ruta de comunicación que conecta a un consumidor de nube con su proveedor de nube puede involucrar varias redes de ISP. La estructura de malla de Internet conecta los hosts de Internet (endpoint systems) mediante varias rutas de red alternativas que se determinan en tiempo de ejecución. Por lo tanto, la comunicación puede mantenerse incluso durante fallas simultáneas de la red, aunque el uso de múltiples rutas de red puede causar latencia y fluctuaciones de enruteamiento.

Esto aplica a los ISPs que implementan la capa de interconexión de redes de Internet e interactúan con otras tecnologías de red, de la siguiente manera:

Red física

Los paquetes IP se transmiten a través de redes físicas subyacentes que conectan nodos adyacentes, como Ethernet, red ATM y HSDPA móvil 3G. Las redes físicas comprenden una capa de enlace de datos que controla la transferencia de datos entre nodos vecinos y una capa física que transmite bits de datos a través de medios cableados e inalámbricos.

Transport Layer Protocol

Los protocolos de la capa de transporte, como el Protocolo de control de transmisión (TCP) y el Protocolo de datagramas de usuario (UDP), utilizan la IP para proporcionar soporte de comunicación estandarizado de extremo a extremo que facilita la navegación de paquetes de datos a través de Internet.

Application Layer Protocol

Protocolos como HTTP, SMTP para correo electrónico, BitTorrent para P2P y SIP para telefonía IP utilizan protocolos de la capa de transporte para estandarizar y habilitar métodos específicos de transferencia de paquetes de datos a través de Internet. Muchos otros protocolos también cumplen con los requisitos centrados en la aplicación y utilizan TCP/IP o UDP como método principal de transferencia de datos a través de Internet y LANs.

La figura 3.4 presenta el modelo de referencia de Internet y la pila de protocolos.

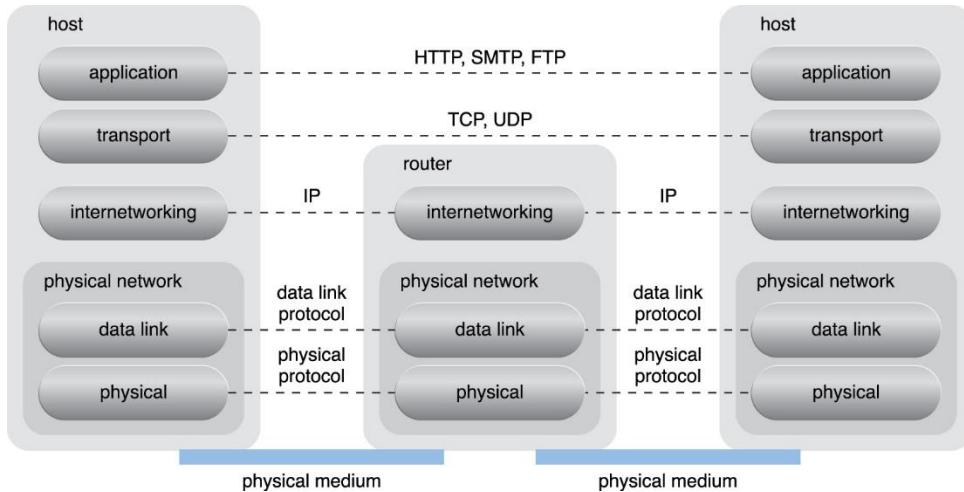


Figura 3.4 Una vista genérica del modelo de referencia de Internet y la pila de protocolos.

Consideraciones técnicas y comerciales

Problemas de conectividad

En los modelos tradicionales de implementación on-premise, las aplicaciones empresariales y diversas soluciones de TI suelen alojarse en servidores centralizados y dispositivos de almacenamiento que residen en el propio centro de datos de la organización. Los dispositivos de los usuarios finales, como teléfonos inteligentes y computadoras portátiles, acceden al centro de datos a través de la red corporativa, que proporciona conectividad a Internet ininterrumpida.

TCP/IP facilita tanto el acceso a Internet como el intercambio de datos on-premise a través de LANs (Figura 3.5). Aunque no se conoce comúnmente como un modelo de nube, esta configuración se ha implementado numerosas veces para redes on-premise medianas y grandes.

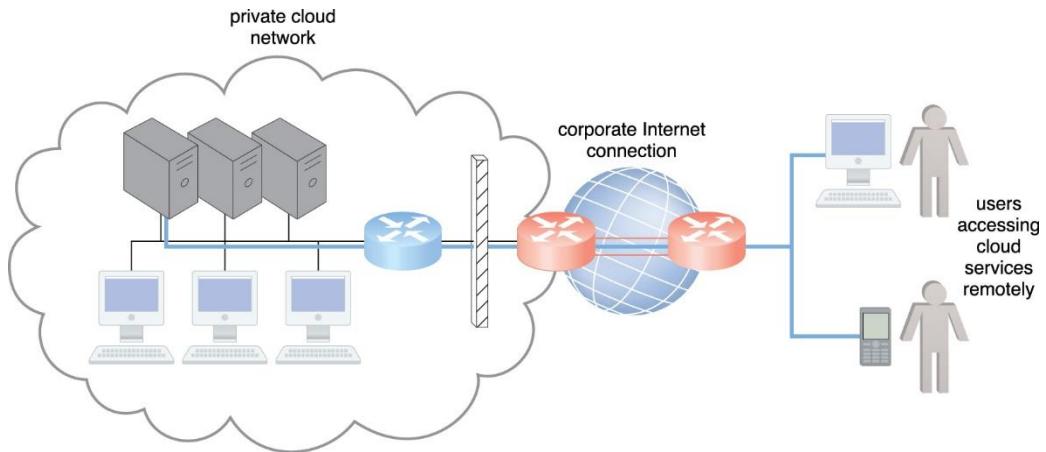


Figura 3.5 Arquitectura internetworking de una nube privada. Los recursos TI físicos que componen la nube están localizados y administrados dentro de la organización.

Las organizaciones que utilizan este modelo de implementación pueden acceder directamente al tráfico de la red hacia y desde Internet y, por lo general, tienen un control completo y pueden proteger sus redes corporativas mediante firewalls y software de monitoreo. Estas organizaciones también asumen la responsabilidad de implementar, operar y mantener sus recursos de TI y conectividad a Internet.

Los dispositivos de usuario final que están conectados a la red a través de Internet pueden tener acceso continuo a servidores y aplicaciones centralizados en la nube (Figura 3.6).

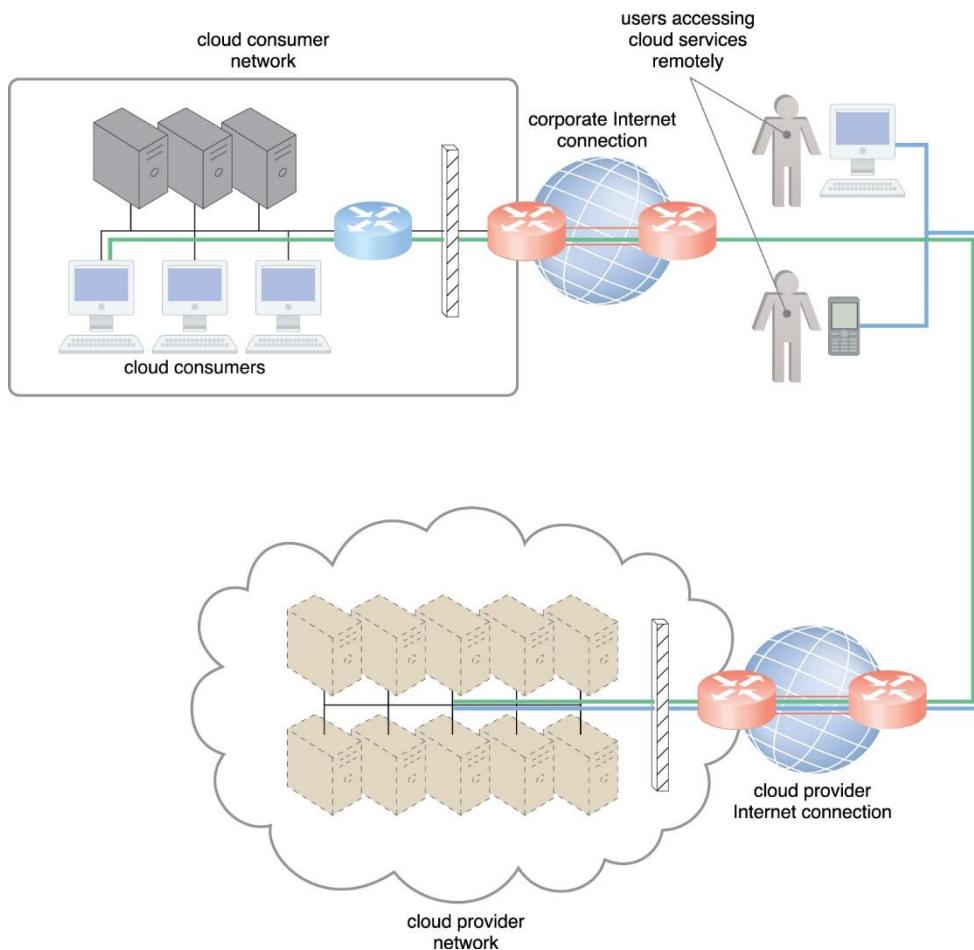


Figura 3.6 La arquitectura de interconexión de un modelo de implementación en la nube basado en Internet. Internet es el agente de conexión entre los consumidores de la nube no próximos, los usuarios finales que hacen roaming²⁰ y la propia red del proveedor de la nube.

Una característica destacada de la nube que se aplica a la funcionalidad del usuario final es cómo se puede acceder a los recursos de TI centralizados utilizando los mismos protocolos de red, independientemente de si residen dentro o fuera de una red corporativa. Si los recursos de TI son on-premise o internet-based, determina cómo los usuarios finales internos o externos acceden a los servicios (Tabla 3.1).

Recursos de TI On-Premise

Recursos de TI Cloud-Based

²⁰ El roaming es un concepto utilizado en telecomunicaciones para referirse a la posibilidad de un dispositivo inalámbrico de utilizar una cobertura de red distinta de la principal. Esto le permite conectarse a redes secundarias utilizando su identificador en la red principal.

los dispositivos internos de los usuarios finales acceden a los servicios de TI corporativos a través de la red corporativa	los dispositivos internos de los usuarios finales acceden a los servicios de TI corporativos a través de una conexión a Internet
los usuarios internos acceden a los servicios de TI corporativos a través de la conexión corporativa a Internet mientras hacen roaming en redes externas	los usuarios internos acceden a los servicios de TI corporativos mientras hacen roaming en redes externas a través de la conexión a Internet del proveedor de la nube
los usuarios externos acceden a los servicios TI corporativos a través de la conexión a Internet corporativa	los usuarios externos acceden a los servicios de TI corporativos a través de la conexión a Internet del proveedor de la nube

Tabla 3.1 Una comparación de la interconexión de redes on-premise y cloud-based.

Los proveedores de la nube pueden configurar fácilmente los recursos de TI cloud-based para que sean accesibles tanto para usuarios internos como externos a través de una conexión a Internet (como se mostró anteriormente en la Figura 3.6). Esta arquitectura de interconexión de redes beneficia a los usuarios internos que requieren un acceso ubicuo a las soluciones de TI corporativas, así como a los consumidores de la nube que necesitan brindar servicios basados en Internet a usuarios externos. Los principales proveedores de la nube ofrecen conectividad a Internet que es superior a la conectividad de las organizaciones individuales, lo que genera cargos adicionales por uso de la red como parte de su modelo de precios.

Problemas de latencia y ancho de banda de la red

Además de verse afectado por el ancho de banda del enlace de datos que conecta las redes a los ISPs, el ancho de banda de extremo a extremo está determinado por la capacidad de transmisión de los enlaces de datos compartidos que conectan los nodos intermediarios. Los ISPs necesitan usar tecnología de red de banda ancha para implementar la red central necesaria para garantizar la conectividad de extremo a extremo. Este tipo de ancho de banda aumenta constantemente, ya que las tecnologías de aceleración web, como el almacenamiento en caché dinámico, la compresión y la búsqueda previa, continúan mejorando la conectividad del usuario final.

También conocida como retardo de tiempo, la *latencia* es la cantidad de tiempo que tarda un paquete en viajar de un nodo de datos a otro. La latencia aumenta con cada nodo intermediario en la ruta del paquete de datos. Las colas de transmisión en la infraestructura de la red pueden generar condiciones de carga pesada que también aumentan la latencia de la red. Las redes dependen de las condiciones del tráfico en los nodos compartidos, lo que hace que la latencia de Internet sea muy variable y, a menudo, impredecible.

Las redes de paquetes con calidad de servicio (QoS) de "best effort" normalmente transmiten paquetes por orden de llegada. Los flujos de datos que utilizan rutas de red congestionadas sufren una degradación del nivel de servicio en forma de reducción del ancho de banda, aumento de la latencia o pérdida de paquetes cuando no se prioriza el tráfico.

La naturaleza de la conmutación de paquetes permite que los paquetes de datos elijan rutas dinámicamente a medida que viajan a través de la infraestructura de red de Internet. La QoS de extremo a extremo puede verse afectada como resultado de esta selección dinámica, ya que la

velocidad de viaje de los paquetes de datos es susceptible a condiciones como la congestión de la red y, por lo tanto, no es uniforme.

Las soluciones de TI deben evaluarse frente a los requisitos comerciales que se ven afectados por el ancho de banda y la latencia de la red, que son inherentes a la interconexión en la nube. El ancho de banda es fundamental para las aplicaciones que requieren la transferencia de cantidades sustanciales de datos hacia y desde la nube, mientras que la latencia es fundamental para las aplicaciones que requieren tiempos de respuesta rápidos.

Selección de operador de nube y proveedor de nube

Los niveles de servicio de las conexiones a Internet entre los consumidores de la nube y los proveedores de la nube están determinados por sus ISPs, que suelen ser diferentes y, por lo tanto, incluyen varias redes de ISP en sus rutas. La gestión de QoS en múltiples ISPs es difícil de lograr en la práctica, lo que requiere la colaboración de los operadores de la nube en ambos lados para garantizar que sus niveles de servicio de extremo a extremo sean suficientes para los requisitos comerciales.

Los consumidores de la nube y los proveedores de la nube pueden necesitar usar múltiples operadores de la nube para lograr el nivel necesario de conectividad y confiabilidad para sus aplicaciones en la nube, lo que genera costos adicionales. Por lo tanto, la adopción de la nube puede ser más fácil para las aplicaciones con requisitos de latencia y ancho de banda más relajados.

3.2. Tecnología del centro de datos

Agrupar los recursos de TI muy cerca unos de otros, en lugar de tenerlos dispersos geográficamente, permite compartir el poder, una mayor eficiencia en el uso de recursos de TI compartidos y una mejor accesibilidad para el personal de TI. Estas son las ventajas que naturalmente popularizaron el concepto de centro de datos. Los centros de datos modernos existen como infraestructura de TI especializada que se utiliza para albergar recursos de TI centralizados, como servidores, bases de datos, dispositivos de redes y telecomunicaciones y sistemas de software. A continuación, se muestra el interior de un centro de datos de Facebook.



Los centros de datos suelen estar compuestos por las siguientes tecnologías y componentes:

Virtualización

Los centros de datos constan de recursos de TI tanto físicos como virtualizados. La capa de recursos físicos de TI se refiere a la infraestructura de las instalaciones que alojan los sistemas y equipos de cómputo y de redes, junto con los sistemas de hardware y sus sistemas operativos (Figura 3.7). La capa de abstracción de recursos y el control de la virtualización se componen de herramientas operativas y de gestión que a menudo se basan en plataformas de virtualización que abstraen las TI de computación física y de redes, así como componentes virtualizados que son más fáciles de asignar, operar, liberar, monitorear y controlar.

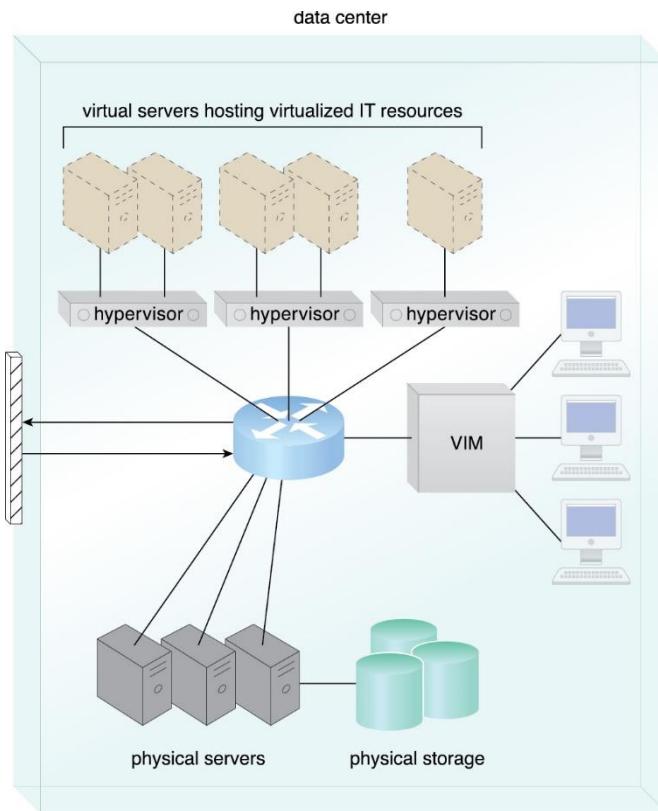


Figura 3.7 Los componentes comunes de un centro de datos que trabajan juntos para proporcionar recursos de TI virtualizados soportados por recursos de TI físicos. El cubo denominado VIM es el administrador de los recursos virtualizados (VI manager).

La virtualización de componentes se analiza por separado en la próxima sección Tecnologías de Virtualización.

Estandarización y modularidad

Los centros de datos se construyen sobre hardware básico estandarizado y se diseñan con arquitecturas modulares, agregando múltiples bloques de construcción idénticos de infraestructura y equipo para respaldar la escalabilidad, el crecimiento y los reemplazos rápidos de hardware. La modularidad y la estandarización son requisitos clave para reducir los costos operativos y de inversión, ya que permiten economías de escala para los procesos de adquisición, implementación, operación y mantenimiento.

Las estrategias comunes de virtualización y la mejora constante en la capacidad y el rendimiento de los dispositivos físicos favorecen la consolidación de los recursos de TI, ya que se necesitan menos componentes físicos para admitir configuraciones complejas. Los recursos de TI consolidados pueden servir a diferentes sistemas y compartirse entre diferentes consumidores de la nube.

Automatización

Los centros de datos tienen plataformas especializadas que automatizan tareas como el aprovisionamiento, la configuración, la aplicación de parches y el monitoreo sin supervisión. Los avances en las plataformas y herramientas de gestión de centros de datos aprovechan las tecnologías informáticas autónomas para permitir la autoconfiguración y la auto recuperación.

Operación y administración remotas

La mayoría de las tareas operativas y administrativas de los recursos de TI en los centros de datos se controlan a través de la red mediante consolas remotas y sistemas de gestión. El personal técnico no está obligado a visitar las salas dedicadas que albergan los servidores, excepto para realizar tareas muy específicas, como el manejo y cableado de equipos o la instalación y el mantenimiento a nivel de hardware.

Alta disponibilidad

Dado que cualquier forma de interrupción del centro de datos afecta significativamente la continuidad del negocio para las organizaciones que utilizan sus servicios, los centros de datos están diseñados para operar con niveles cada vez más altos de redundancia para mantener la disponibilidad. Los centros de datos suelen tener fuentes de alimentación redundantes e ininterrumpidas, cableado y subsistemas de control ambiental en previsión de fallas del sistema, junto con enlaces de comunicación y clusters de hardware para balancear la carga.

Diseño Security-Aware, Operación y Administración

Los requisitos de seguridad, como los controles de acceso físico y lógico y las estrategias de recuperación de datos, deben ser minuciosos y completos para los centros de datos, ya que son estructuras centralizadas que almacenan y procesan datos comerciales.

Debido a la naturaleza a veces prohibitiva de construir y operar centros de datos on-premise, la subcontratación de recursos de TI basados en centros de datos ha sido una práctica común en la industria durante décadas. Sin embargo, los modelos de subcontratación a menudo requerían un compromiso del consumidor a largo plazo y, por lo general, no podían proporcionar elasticidad, problemas que una nube típica puede abordar a través de características inherentes, como el acceso ubicuo, el aprovisionamiento bajo demanda, la elasticidad rápida y el pago por uso.

Instalaciones

Las instalaciones del centro de datos son ubicaciones diseñadas a la medida y que están equipadas con equipos especializados de computación, almacenamiento y red. Estas instalaciones cuentan con varias áreas de diseño funcional, así como varias fuentes de alimentación, cableado y estaciones de control ambiental que regulan la calefacción, ventilación, aire acondicionado, protección contra incendios y otros subsistemas relacionados.

El sitio y el diseño de las instalaciones de un centro de datos determinado suelen estar delimitados en espacios aislados.

Hardware de cómputo

Gran parte del procesamiento pesado en los centros de datos a menudo es ejecutado por servidores básicos estandarizados que tienen una potencia de cómputo y una capacidad de almacenamiento sustanciales. Varias tecnologías de hardware informático están integradas en estos servidores modulares, como:

- diseño de servidor con formato de montaje en rack, compuesto por racks estandarizados con interconexiones para alimentación, red y refrigeración interna
- soporte para diferentes arquitecturas de procesamiento de hardware, como x86-32bits, x86-64 y RISC
- una arquitectura de CPU multinúcleo de bajo consumo que alberga cientos de núcleos de procesamiento en un espacio tan pequeño como una sola unidad de rack estandarizado
- componentes redundantes e intercambiables aún en operación, como discos duros, fuentes de alimentación, interfaces de red y tarjetas controladoras de almacenamiento

Las arquitecturas informáticas, como las tecnologías de servidor blade, utilizan interconexiones físicas integradas en el bastidor (gabinetes blade), con switches y unidades de fuente de alimentación y ventiladores de refrigeración compartidos. Las interconexiones mejoran las redes entre componentes y su administración al tiempo que optimizan el espacio físico y la energía. Estos sistemas suelen admitir intercambio aún en operación, escalado, reemplazo y mantenimiento de servidores individuales, lo que beneficia la implementación de sistemas tolerantes a fallas que se basan en clústeres de computadoras.

En la siguiente imagen se muestra un servidor Blade que puede soportar ocho procesadores y 8 DIMMs por procesador.



Las plataformas de hardware de computación contemporáneas generalmente admiten software propietario y estándar operativo y de administración que configuran, monitorean y controlan los recursos de TI de hardware desde consolas de administración remotas. Con una consola de administración correctamente instalada, un solo operador puede supervisar cientos o miles de servidores físicos, servidores virtuales y otros recursos de TI.

Hardware de almacenamiento

Los centros de datos cuentan con sistemas de almacenamiento especializados que mantienen enormes cantidades de información digital para satisfacer necesidades considerables de capacidad

de almacenamiento. Estos sistemas de almacenamiento son contenedores que albergan numerosos discos duros que se organizan en matrices.

Los sistemas de almacenamiento generalmente involucran las siguientes tecnologías:

- Matrices de discos duros: estas matrices dividen y replican de manera inherente los datos entre varias unidades físicas y aumentan el rendimiento y la redundancia al incluir discos de repuesto. Esta tecnología a menudo se implementa mediante matrices redundantes de esquemas de discos independientes (RAID), que normalmente se manejan a través de hardware controlador de matriz de discos.
- Almacenamiento en caché I/O: esto generalmente se realiza a través de controladores de matriz de disco duro, que mejoran los tiempos de acceso al disco y el rendimiento mediante el almacenamiento en caché de datos.
- Discos duros hot-swappable: estos se pueden quitar de forma segura de los arreglos sin necesidad de apagarlos antes.
- Virtualización del almacenamiento: esto se logra mediante el uso de discos duros virtualizados y almacenamiento compartido.
- Mecanismos rápidos de replicación de datos: incluyen instantáneas, que guardan la memoria de una máquina virtual en un archivo legible por un hipervisor para recargas futuras, y clonación de volúmenes, que copia volúmenes y particiones de discos duros virtuales o físicos.

Los sistemas de almacenamiento abarcan redundancias terciarias, como bibliotecas de cintas robotizadas, que se utilizan como sistemas de copia de seguridad y recuperación que normalmente dependen de medios extraíbles. Este tipo de sistemas puede existir como recurso de TI en red o almacenamiento adjunto directo (DAS), en el que un sistema de almacenamiento está conectado directamente al recurso de TI informático mediante un adaptador de bus de host (HBA). En el primer caso, el sistema de almacenamiento está conectado a uno o más recursos de TI a través de una red.

Los dispositivos de almacenamiento en red suelen pertenecer a una de las siguientes categorías:

- Storage Area Network (SAN): Los medios de almacenamiento de datos físicos se conectan a través de una red dedicada y brindan acceso al almacenamiento de datos a nivel de bloque mediante protocolos estándar de la industria, como la Small Computer System Interface (SCSI).
- Network-Attached Storage (NAS): Las matrices de discos duros están contenidas y administradas por este dispositivo dedicado, que se conecta a través de una red y facilita el acceso a los datos mediante protocolos de acceso a datos centrados en archivos, como Network File System (NFS) o Server Message Block. (SMB).

NAS, SAN y otras opciones de sistemas de almacenamiento más avanzadas brindan failover en muchos componentes a través del controlador de redundancia, la redundancia de enfriamiento y las matrices de discos duros que usan tecnología de almacenamiento RAID.

[Hardware de red](#)

Los centros de datos requieren un extenso hardware de red para habilitar múltiples niveles de conectividad. Para una versión simplificada de la infraestructura de red, un centro de datos se divide

en cinco subsistemas de red. A continuación, un resumen de los elementos más comunes utilizados para su implementación.

Interconexión de Carrier y Redes Externas

Un subsistema relacionado con la infraestructura de interconexión de redes, esta interconexión generalmente se compone de enruteadores troncales que proporcionan enruteamiento entre las conexiones WAN externas y la LAN de los centros de datos, así como dispositivos de seguridad de la red perimetral²¹, como firewalls y puertas de enlace VPN.

Balanceo de carga y aceleración Web-Tier

Este subsistema comprende dispositivos de aceleración Web, como preprocesadores XML, dispositivos de encriptado/desencriptado y dispositivos de conmutación de nivel 7 que realizan enruteamiento dependiendo del contenido.

Estructura LAN

La estructura LAN constituye la LAN interna y proporciona conectividad redundante y de alto rendimiento para todos los recursos de TI habilitados para la red del centro de datos. A menudo se implementa con varios conmutadores de red que facilitan las comunicaciones de red y funcionan a velocidades de hasta diez gigabits por segundo. Estos conmutadores de red avanzados también pueden realizar varias funciones de virtualización, como la agrupación de LAN en VLAN, la agregación de enlaces, el enruteamiento controlado entre redes, el balanceo de carga y failover.

Estructura SAN

En relación con la implementación de redes de área de almacenamiento (SAN) que brindan conectividad entre servidores y sistemas de almacenamiento, la estructura SAN generalmente se implementa con conmutadores de red Fibre Channel (FC), Fibre Channel sobre Ethernet (FCoE) e InfiniBand.

Puertas de enlace NAS

Este subsistema proporciona puntos de conexión para dispositivos de almacenamiento basados en NAS e implementa hardware de conversión de protocolo que facilita la transmisión de datos entre dispositivos SAN y NAS.

Las tecnologías de red del centro de datos tienen requisitos operativos de escalabilidad y alta disponibilidad que se cumplen mediante el empleo de configuraciones redundantes y/o tolerantes a fallas. Estos cinco subsistemas de red mejoran la redundancia y la confiabilidad del centro de datos para garantizar que tengan suficientes recursos de TI para mantener un cierto nivel de servicio incluso ante múltiples fallas.

²¹ En seguridad informática, una red perimetral o DMZ (demilitarized zone) es una red local que se ubica entre la red interna de una organización y una red externa, generalmente en Internet. El objetivo de una red perimetral es que las conexiones desde la red externa a la red perimetral estén permitidas, mientras que en general las conexiones desde la red perimetral no se permitan a la red interna. Esto permite que los equipos de la DMZ puedan dar servicios a la red externa, a la vez que protegen la red interna en el caso de que unos intrusos comprometan la seguridad de los equipos situados en la zona desmilitarizada. Fuente: Wikipedia.

Los enlaces ópticos de red de ultra alta velocidad se pueden usar para agregar canales individuales de gigabit por segundo en fibras ópticas únicas utilizando tecnologías de multiplexación como la multiplexación por división de longitud de onda densa (DWDM). Repartidos en varias ubicaciones y utilizados para interconectar granjas de servidores, sistemas de almacenamiento y centros de datos replicados, los enlaces ópticos mejoran las velocidades de transferencia y la resiliencia.

Otras Consideraciones

El hardware de TI está sujeto a una rápida obsolescencia tecnológica, con ciclos de vida que suelen durar entre cinco y siete años. La necesidad continua de reemplazar el equipo con frecuencia da como resultado una combinación de hardware cuya heterogeneidad puede complicar las operaciones y la administración de todo el centro de datos (aunque esto puede mitigarse parcialmente a través de la virtualización).

La seguridad es otro problema importante cuando se considera el papel del centro de datos y las grandes cantidades de datos que se encuentran dentro de sus puertas. Incluso con amplias precauciones de seguridad implementadas, alojar datos exclusivamente en una instalación de centro de datos significa que una incursión de seguridad exitosa puede comprometer mucho más que si los datos se distribuyeran entre componentes individuales no vinculados.

3.3. Tecnología de virtualización

La virtualización es el proceso de convertir un recurso de TI físico en un recurso de TI virtual.

La mayoría de los tipos de recursos de TI se pueden virtualizar, incluidos:

- Servidores: Un servidor físico puede ser abstraído en un servidor virtual.
- Almacenamiento: Un dispositivo de almacenamiento físico se puede abstraer en un dispositivo de almacenamiento o un disco virtuales.
- Red: Los enrutadores y conmutadores físicos se pueden abstraer en estructuras de red lógica, como las VLAN.
- Energía: Un UPS físico y las unidades de distribución de energía pueden resumirse en lo que comúnmente se conoce como UPSs virtuales.

Esta sección se centra en la creación e implementación de servidores virtuales a través de la tecnología de virtualización de servidores.

Nota

Los términos *servidor virtual* y *máquina virtual* (VM) se usan como sinónimos en este libro.

El primer paso para crear un nuevo servidor virtual a través del software de virtualización es la asignación de recursos físicos de TI, seguido de la instalación de un sistema operativo. Los servidores virtuales utilizan sus propios sistemas operativos guest(invitado), que son independientes del sistema operativo en el que ellos fueron creados.

Tanto el sistema operativo invitado como el software de la aplicación que se ejecutan en el servidor virtual desconocen el proceso de virtualización, lo que significa que estos recursos de TI virtualizados se instalan y ejecutan como si estuvieran ejecutándose en un servidor físico separado. Esta uniformidad de ejecución que permite que los programas se ejecuten en sistemas físicos como lo harían en sistemas virtuales es una característica vital de la virtualización. Los sistemas operativos invitados generalmente requieren un uso continuo de productos y aplicaciones de software que no necesitan ser personalizados, configurados o parcheados para ejecutarse en un entorno virtualizado.

El software de virtualización se ejecuta en un servidor físico denominado *host* o *host físico*, cuyo hardware subyacente se hace accesible mediante el software de virtualización. La funcionalidad del software de virtualización abarca los servicios del sistema que están específicamente relacionados con la administración de las máquinas virtuales y que normalmente no se encuentran en los sistemas operativos estándar. Es por eso que a este software a veces se le denomina administrador de máquina virtual o monitor de máquina virtual (VMM), pero es más comúnmente conocido como hipervisor.

Independencia de hardware

La instalación del software de aplicación y configuración de un sistema operativo en una plataforma de hardware de TI único da como resultado muchas dependencias entre software y hardware. En un entorno no virtualizado, el sistema operativo está configurado para modelos de hardware específicos y requiere una reconfiguración si es necesario modificar estos recursos de TI.

La virtualización es un proceso de conversión que convierte el hardware de TI único en copias basadas en software emuladas y estandarizadas. A través de la independencia del hardware, los servidores virtuales se pueden mover fácilmente a otro host de virtualización, resolviendo automáticamente múltiples problemas de incompatibilidad de hardware y software. Como resultado, clonar y manipular recursos de TI virtuales es mucho más fácil que duplicar hardware físico.

Consolidación de servidores

La función de coordinación que proporciona el software de virtualización permite que se creen simultáneamente varios servidores virtuales en el mismo host de virtualización. La tecnología de virtualización permite que diferentes servidores virtuales comparten un servidor físico. Este proceso se denomina *consolidación de servidores* y se usa comúnmente para aumentar la utilización del hardware, el balanceo de carga y la optimización de los recursos de TI disponibles. La flexibilidad resultante es tal que diferentes servidores virtuales pueden ejecutar diferentes sistemas operativos invitados en el mismo host.

Esta capacidad fundamental es compatible directamente con las características comunes de la nube, como el uso bajo demanda, la agrupación de recursos, la elasticidad, la escalabilidad y la resiliencia.

Replicación de recursos

Los servidores virtuales se crean como imágenes de disco virtual que contienen copias de archivos binarios del contenido del disco duro. Estas imágenes de disco virtual son accesibles para el sistema operativo host, lo que significa que se pueden usar operaciones de archivos simples, como copiar, mover y pegar, para replicar, migrar y hacer una copia de seguridad del servidor virtual. Esta

facilidad de manipulación y replicación es una de las características más destacadas de la tecnología de virtualización, ya que permite:

- La creación de imágenes de máquinas virtuales estandarizadas comúnmente configuradas para incluir capacidades de hardware virtual, sistemas operativos invitados y software adicional de aplicaciones, para pre empaquetar en imágenes de disco virtual soportando así la implementación instantánea.
- Mayor agilidad en la migración y el despliegue de nuevas instancias de una máquina virtual al poder escalar horizontal y verticalmente rápidamente.
- La capacidad de hacer roll back, que es la creación instantánea de contextos de MV al guardar el estado de la memoria de los servidores virtuales y la imagen del disco duro en un archivo que entiende el host. (Los operadores pueden volver fácilmente a estas instantáneas y restaurar la máquina virtual a su estado anterior).
- El soporte de la continuidad del negocio con procedimientos eficientes de copia de seguridad y restauración, así como la creación de múltiples instancias para aplicaciones y recursos de TI que son críticos.

Virtualización basada en el sistema operativo

La virtualización basada en el sistema operativo es la instalación de un software de virtualización en un sistema operativo preexistente, que se denomina sistema operativo host (Figura 3.8). Por ejemplo, un usuario cuya estación de trabajo tiene instalada una versión específica de Windows desea generar servidores virtuales e instala software de virtualización en el sistema operativo host como cualquier otro programa. Este usuario necesita usar esta aplicación para generar y operar uno o más servidores virtuales. El usuario necesita usar software de virtualización para habilitar el acceso directo a cualquiera de los servidores virtuales generados. Dado que el sistema operativo host puede proporcionar a los dispositivos de hardware el soporte necesario, la virtualización del sistema operativo puede corregir los problemas de compatibilidad del hardware incluso si el controlador de hardware no está disponible para el software de virtualización.

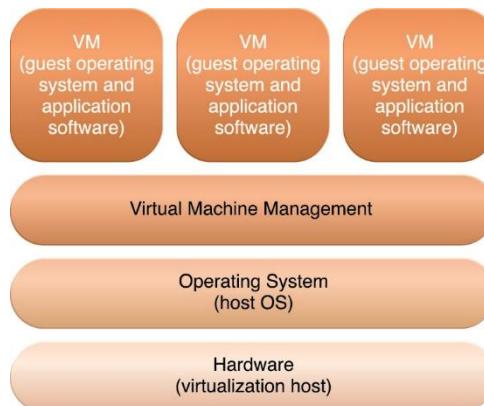


Figura 3.8 Las diferentes capas lógicas de la virtualización basada en el sistema operativo, en las que la máquina virtual se instala primero en un sistema operativo host completo y, posteriormente, se utiliza para generar máquinas virtuales.

La independencia del hardware que es habilitada por la virtualización permite que los recursos de TI del hardware se utilicen de manera más flexible. Por ejemplo, considere un escenario en el que el sistema operativo host tiene el software necesario para controlar cinco adaptadores de red que están disponibles para la computadora física. El software de virtualización puede hacer que los cinco adaptadores de red estén disponibles para el servidor virtual, incluso si el sistema operativo virtualizado no puede albergar físicamente cinco adaptadores de red.

El software de virtualización convierte los recursos de TI de hardware que requieren un software exclusivo para funcionar, en recursos de TI virtualizados que son compatibles con una variedad de sistemas operativos. Dado que el sistema operativo del host es un sistema operativo completo en sí mismo, muchos servicios basados en el sistema operativo que están disponibles como herramientas de administración se pueden usar para administrar el host físico.

Ejemplos de dichos servicios incluyen:

- Copia de seguridad y recuperación
- Integración con los servicios de directorio
- Gestión de la seguridad

La virtualización basada en el sistema operativo puede introducir demandas y problemas relacionados con el rendimiento, debido a:

- El sistema operativo host consume CPU, memoria y otros recursos de TI de hardware.
- Las llamadas al sistema de hardware de los sistemas operativos invitados deben atravesar varias capas hacia y desde el hardware, lo que reduce el rendimiento general.
- Por lo general, se requieren licencias para los sistemas operativos host, además de las licencias individuales para cada uno de sus sistemas operativos invitados.

Una preocupación con la virtualización basada en el sistema operativo es la sobrecarga de procesamiento necesaria para ejecutar el software de virtualización y los sistemas operativos host. La implementación de una capa de virtualización afectará negativamente el rendimiento general del sistema. Estimar, monitorear y administrar el impacto resultante puede ser un desafío porque requiere experiencia en cargas de trabajo del sistema, entornos de software y hardware y herramientas de monitoreo sofisticadas.

Virtualización basada en hardware

Esta opción representa la instalación del software de virtualización directamente en el hardware del host físico para pasar por alto el sistema operativo del host, que presumiblemente está comprometido en el caso de la virtualización basada en el sistema operativo (Figura 3.9). Permitir que los servidores virtuales interactúen con el hardware sin requerir una acción intermedia del sistema operativo host generalmente hace que la virtualización basada en hardware sea más eficiente.

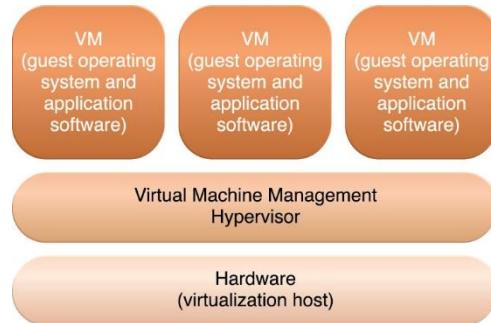


Figura 3.9 Las diferentes capas lógicas de la virtualización basada en hardware, la cual no requiere otro sistema operativo host.

El software de virtualización se suele denominar *hypervisor(hipervisor)* para este tipo de procesamiento. Un hipervisor tiene una interfaz de usuario simple que requiere una cantidad insignificante de espacio de almacenamiento. Existe como una capa delgada de software que maneja funciones de administración de hardware para establecer una capa de administración de virtualización. Los controladores de dispositivos y los servicios del sistema están optimizados para el aprovisionamiento de servidores virtuales, aunque muchas funciones estándar del sistema operativo no están implementadas. Este tipo de sistema de virtualización se utiliza esencialmente para optimizar la sobrecarga inherente a la coordinación que permite que varios servidores virtuales interactúen con la misma plataforma de hardware.

Uno de los principales problemas de la virtualización basada en hardware se refiere a la compatibilidad con los dispositivos de hardware. La capa de virtualización está diseñada para comunicarse directamente con el hardware del host, lo que significa que todos los controladores de dispositivos asociados y el software de soporte deben ser compatibles con el hipervisor. Es posible que los controladores de dispositivos de hardware no estén tan disponibles para las plataformas de hipervisor como lo están para los sistemas operativos. También es posible que las funciones de gestión y administración del host no incluyan la gama de funciones avanzadas que son comunes a los sistemas operativos.

Virtualization Management

Muchas tareas administrativas se pueden realizar más fácilmente utilizando servidores virtuales en lugar de utilizar sus contrapartes físicas. El software de virtualización moderno proporciona varias funciones de administración avanzadas que pueden automatizar las tareas de administración y reducir la carga operativa general de los recursos de TI virtualizados.

La administración de recursos de TI virtualizados a menudo es compatible con herramientas de administración de infraestructura de virtualización (VIM) que administran colectivamente los recursos de TI virtuales y se basan en un módulo de administración centralizado, también conocido como controlador, que se ejecuta en una computadora dedicada.

Otras Consideraciones

- **Sobrecarga de rendimiento:** La virtualización puede no ser ideal para sistemas complejos que tienen grandes cargas de trabajo que utilizan poco el uso compartido y la replicación de recursos. Un plan de virtualización mal formulado puede generar una sobrecarga de rendimiento excesiva. Una estrategia común utilizada para rectificar el problema de la sobrecarga es una técnica llamada

para-virtualización, que presenta una interfaz de software a las máquinas virtuales que no es idéntica a la del hardware subyacente. En cambio, la interfaz del software se ha modificado para reducir la sobrecarga de procesamiento del sistema operativo invitado, que es más difícil de administrar. Un inconveniente importante de este enfoque es la necesidad de adaptar el sistema operativo invitado a la API de para-virtualización, lo que puede afectar el uso de los sistemas operativos invitados estándar y disminuir la portabilidad de la solución.

- Compatibilidad de hardware especial: Es posible que muchos proveedores de hardware que distribuyen hardware especializado no tengan versiones de controladores de dispositivos que sean compatibles con el software de virtualización. Por el contrario, el software en sí puede ser incompatible con las versiones de hardware lanzadas recientemente. Estos tipos de problemas de incompatibilidad se pueden resolver utilizando plataformas de hardware establecidas estándar y productos de software de virtualización maduros.
- Portabilidad: las interfaces programáticas y de gestión que establecen los entornos de administración para que un programa de virtualización funcione con varias soluciones de virtualización pueden presentar brechas de portabilidad debido a incompatibilidades. Iniciativas como Open Virtualization Format (OVF) para la estandarización de formatos de imagen de disco virtual están dedicadas a aliviar esta preocupación.

3.4. Tecnología web

Debido a la dependencia fundamental de la computación en la nube en la interconexión, la universalidad del navegador web y la facilidad del desarrollo de los servicios web, la tecnología web generalmente se usa como medio de implementación e interfaz de administración para los servicios en la nube.

Esta sección presenta las principales tecnologías web y analiza su relación con los servicios en la nube.

Recursos vs recursos de TI

Los elementos accesibles a través de la World Wide Web se denominan recursos o recursos web. Este es un término más genérico que los recursos de TI, que hemos utilizado. Un recurso de TI, dentro del contexto de la computación en la nube, representa un artefacto relacionado con TI físico o virtual que puede estar basado en software o hardware. Sin embargo, un recurso en la Web puede representar una amplia gama de elementos accesibles a través de la World Wide Web. Por ejemplo, un archivo de imagen JPG al que se accede a través de un navegador web se considera un recurso.

Tecnología web básica

La World Wide Web es un sistema de recursos de TI interconectados a los que se accede a través de Internet. Los dos componentes básicos de la Web son el navegador Web cliente y el servidor Web. Otros componentes, como servidores proxy, servicios de almacenamiento en caché, puertas de enlace y balanceadores de carga, se utilizan para mejorar las características de las aplicaciones web,

como la escalabilidad y la seguridad. Estos componentes adicionales residen en una arquitectura en capas que se ubica entre el cliente y el servidor.

Tres elementos fundamentales componen la arquitectura tecnológica de la Web:

- Localizador uniforme de recursos (URL). Una sintaxis estándar utilizada para crear identificadores que apuntan a recursos basados en la Web, la URL a menudo se estructura utilizando una ubicación de red lógica.
- Protocolo de transferencia de hipertexto (HTTP): este es el principal protocolo de comunicaciones utilizado para intercambiar contenido y datos a través de la World Wide Web. Las URL normalmente se transmiten a través de HTTP.
- Lenguajes de marcado (HTML, XML): los lenguajes de marcado proporcionan un medio ligero para expresar datos y metadatos centrados en la Web. Los dos lenguajes de marcado principales son HTML (que se usa para expresar la presentación de páginas web) y XML (que permite la definición de vocabularios usados para asociar significado a datos basados en la web a través de metadatos).

Por ejemplo, un navegador web puede solicitar ejecutar una acción como leer, escribir, actualizar o eliminar sobre un recurso web en Internet y proceder a identificar y ubicar el recurso web a través de su URL. La solicitud se envía mediante HTTP al host de recursos, que también se identifica mediante una URL. El servidor web localiza el recurso web y realiza la operación solicitada, seguida de una respuesta que se envía al cliente. La respuesta puede incluir contenido que incluya declaraciones HTML y XML.

Los recursos web se representan como *hipermedia* en lugar de hipertexto, lo que significa que los medios como gráficos, audio, video, texto sin formato y direcciones URL pueden referenciarse colectivamente en un solo documento. Algunos tipos de recursos hipermedia no se pueden representar sin software adicional o plug-ins del navegador web.

Aplicaciones web

Una aplicación distribuida que usa tecnologías basadas en la web (y generalmente se basa en navegadores web para la presentación de las interfaces de usuario) generalmente se considera una aplicación web. Estas aplicaciones se pueden encontrar en todo tipo de entornos basados en la nube debido a su alta accesibilidad.

La figura 3.10 presenta una abstracción arquitectónica común para aplicaciones web que se basa en el modelo básico de tres niveles. El primer nivel se denomina capa de presentación, que representa la interfaz de usuario. El nivel medio es la capa de aplicación que implementa la lógica de la aplicación, mientras que el tercer nivel es la capa de datos que se compone de almacenes de datos persistentes.

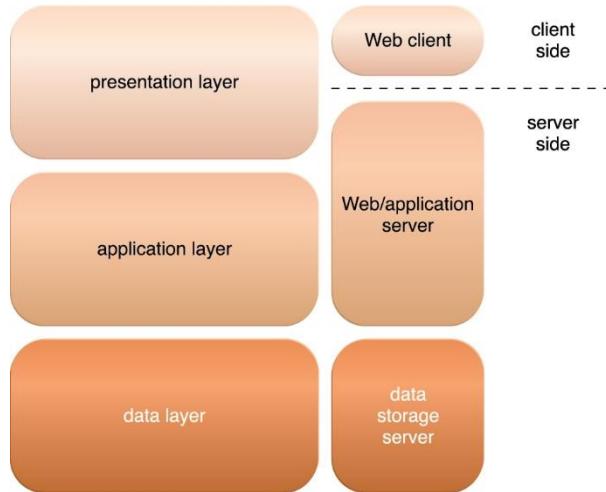


Figura 3.10 Los tres niveles arquitectónicos básicos de las aplicaciones web.

La capa de presentación tiene componentes tanto en el lado del cliente como en el del servidor. Los servidores web reciben solicitudes de clientes y recuperan los recursos solicitados directamente como contenido web estático e indirectamente como contenido web dinámico, que se genera de acuerdo con la lógica de la aplicación. Los servidores web interactúan con los servidores de aplicaciones para ejecutar la lógica de la aplicación solicitada, lo que normalmente implica la interacción con una o más bases de datos subyacentes.

Los entornos PaaS listos para usarse permiten a los consumidores de la nube desarrollar e implementar aplicaciones web. Las ofertas típicas de PaaS tienen instancias separadas de los entornos de servidor web, servidor de aplicaciones y servidor de almacenamiento de datos.

3.5. Tecnología multitenant

El diseño de la aplicación multitenant (multi inquilino) se creó para permitir que varios usuarios (tenants) accedan a la misma lógica de aplicación simultáneamente. Cada tenant tiene su propia vista de la aplicación que usa, administra y personaliza como una instancia dedicada del software sin tener conocimiento de otros tenants que usan la misma aplicación.

Las aplicaciones multitenant garantizan que los inquilinos no tengan acceso a datos e información de configuración que no les pertenezcan. Los inquilinos pueden personalizar individualmente las funciones de la aplicación, como:

- Interfaz de usuario: los inquilinos pueden definir un "aspecto y lectura" especializado para la interfaz de su aplicación.
- Proceso comercial: los inquilinos pueden personalizar las reglas, la lógica y los flujos de trabajo de los procesos comerciales que están implementado en la aplicación.
- Modelo de datos: los inquilinos pueden ampliar el esquema de datos de la aplicación para incluir, excluir o renombrar campos en las estructuras de datos de la aplicación.
- Control de acceso: los inquilinos pueden controlar de forma independiente los derechos de acceso para usuarios y grupos.

La arquitectura de una aplicación multitenant es a menudo significativamente más compleja que la de las aplicaciones de un solo tenant. Las aplicaciones multitenant deben admitir el uso compartido de varios artefactos por parte de múltiples usuarios (incluidos portales, esquemas de datos, middleware y bases de datos), al tiempo que mantienen niveles de seguridad que separan los ambientes operacionales de los tenants individuales.

Las características comunes de las aplicaciones multiusuario incluyen:

- Aislamiento de uso: el comportamiento de uso de un inquilino no afecta la disponibilidad ni el rendimiento de la aplicación de otros inquilinos.
- Seguridad de datos: los inquilinos no pueden acceder a los datos que pertenecen a otros inquilinos.
- Recuperación: los procedimientos de copia de seguridad y restauración se ejecutan por separado para los datos de cada inquilino.
- Actualizaciones de aplicaciones: los inquilinos no se ven afectados negativamente por la actualización síncrona de artefactos de software compartido.
- Escalabilidad: la aplicación se puede escalar para adaptarse a los incrementos en el uso por parte de los inquilinos existentes y/o aumentos en la cantidad de inquilinos.
- Uso medido: a los inquilinos se les cobra solo por el procesamiento de la aplicación y las funciones que realmente se consumen.
- Aislamiento de nivel de datos: los inquilinos pueden tener bases de datos, tablas o esquemas individuales aislados de otros inquilinos. Como alternativa, las bases de datos, las tablas o los esquemas se pueden diseñar para que los inquilinos los compartan de forma intencionada.

Una aplicación multitenant que se utiliza de manera concurrente por dos tenants distintos se ilustra en la Figura 3.11. Este tipo de aplicación es típica con implementaciones SaaS.

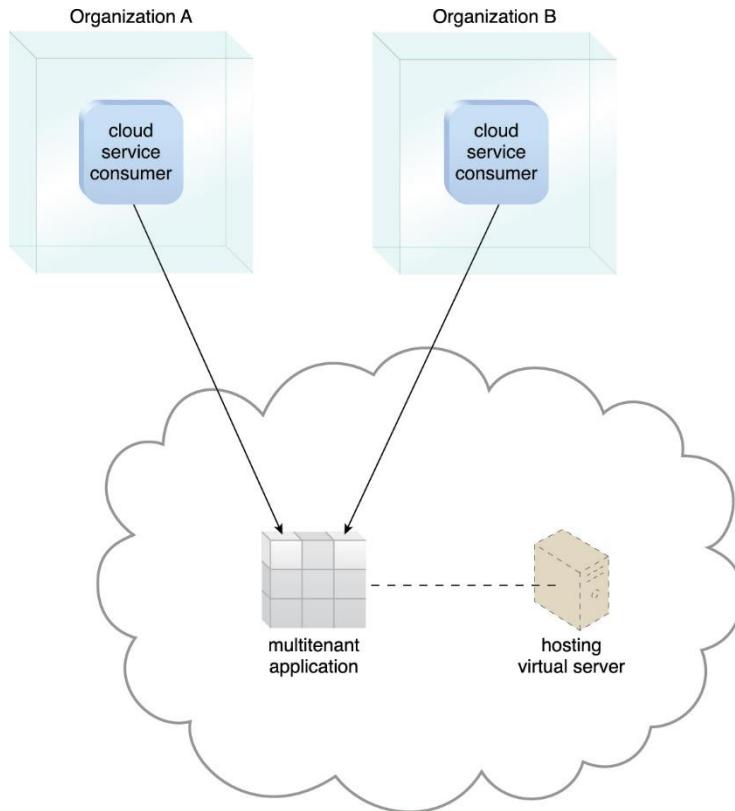


Figura 3.11 Una aplicación multitenant que atiende a múltiples consumidores de servicios en la nube simultáneamente.

Multitenancy vs Virtualización

En ocasiones, multitenancy se confunde con la virtualización porque el concepto de múltiples arrendatarios es similar al concepto de instancias virtualizadas.

Las diferencias radican en lo que se multiplica dentro de un servidor físico que actúa como host:

- Con virtualización: un solo servidor físico puede alojar varias copias virtuales del entorno del servidor. Cada copia se puede proporcionar a diferentes usuarios, se puede configurar de forma independiente y puede contener sus propios sistemas operativos y aplicaciones.
- Con multitenancy: un servidor físico o virtual que alberga una aplicación está diseñado para permitir el uso por parte de múltiples usuarios diferentes. Cada usuario siente que tiene un uso exclusivo de la aplicación.

3.6. Containerization

La containerization (contenedorización) es una tecnología de virtualización a nivel de sistema operativo que se utiliza para implementar y ejecutar aplicaciones y servicios en la nube sin necesidad de implementar un servidor virtual para cada solución. En lugar de eso, se implementan dentro de contenedores. El uso de contenedores permite que varios servicios en la nube aislados se ejecuten

en un solo servidor físico o servidor virtual mientras se accede al mismo kernel del sistema operativo.

El kernel del sistema operativo permite la existencia de múltiples instancias de espacio de usuario aisladas o múltiples entornos de ejecución aislados conocidos como contenedores, particiones, virtual engines, jails o jails chroot. Independientemente del entorno de ejecución que se utilice, cuando un servicio en la nube se ejecuta dentro de un contenedor, se ejecuta en una computadora real desde su punto de vista.

Un servicio en la nube que se ejecuta en un sistema operativo de servidor físico o virtual puede ver todos los recursos proporcionados, como dispositivos conectados, puertos, archivos, carpetas, recursos compartidos de red, CPU, así como la memoria física direccionable. Sin embargo, un servicio en la nube que se ejecuta dentro de un contenedor solo puede ver el contenido del contenedor y los dispositivos adjuntos al contenedor.

Contenedорización vs. virtualización

Como se explicó anteriormente, la virtualización se refiere al acto de crear una versión virtual, en lugar de una versión real de algo. Esto incluye plataformas de hardware de computadora virtual, dispositivos de almacenamiento y recursos de red informática. Los servidores virtuales son una abstracción del hardware físico a través de la visualización del servidor y el uso de hipervisores para abstraer un servidor físico determinado en varios servidores virtuales.

El hipervisor permite que varios servidores virtuales se ejecuten en un solo host físico. Los servidores virtuales ven el hardware emulado que les presenta el hipervisor como hardware real, y cada servidor virtual tiene su propio sistema operativo, también conocido como sistema operativo invitado, que debe desplegarse dentro del servidor virtual y administrarse y mantenerse como si fuera implementado en un servidor físico.

Por el contrario, los contenedores son una abstracción en la capa de aplicación o en la capa de servicio que empaquetan el código y las dependencias juntos. Se pueden implementar varios contenedores en la misma máquina y compartir un kernel de sistema operativo con otros contenedores. Cada contenedor se ejecuta como un proceso aislado en el espacio del usuario. Los contenedores no requieren el sistema operativo invitado que se necesita para los servidores virtuales y pueden ejecutarse directamente en el sistema operativo de un servidor físico. Los contenedores también consumen menos espacio de almacenamiento que los servidores virtuales. La Figura 3.12 muestra la diferencia entre servidores virtuales y contenedores.

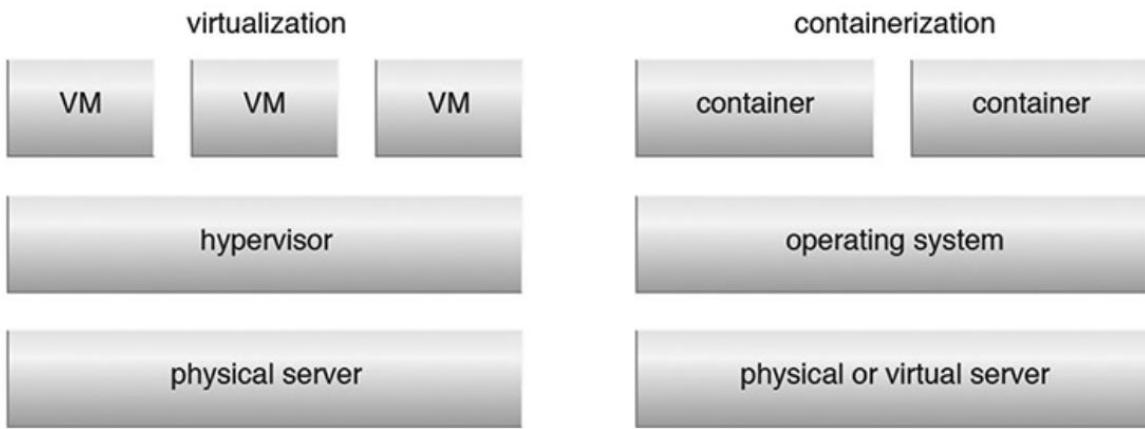


Figura 3.12 Una comparación de la virtualización de servidores y la creación de contenedores.

Los contenedores se pueden implementar en servidores virtuales, en cuyo caso se requiere virtualización anidada para permitir que el motor del contenedor se instale y opere. La virtualización anidada se refiere a la implementación en la que un sistema virtualizado se implementa sobre otro.

Beneficios de los contenedores

La portabilidad es uno de los beneficios clave de los contenedores, ya que permite a los administradores de recursos de la nube mover contenedores a cualquier entorno que comparta el mismo sistema operativo host y el mismo motor de contenedor en el que está alojado el contenedor, y sin necesidad de cambiar la aplicación o el software (lo cual normalmente requeriría cambios en el código fuente).

La utilización eficiente de los recursos se logra al reducir significativamente el uso de CPU, memoria y almacenamiento en comparación con los servidores virtuales. Es posible dar soporte a varios contenedores en la misma infraestructura requerida por un solo servidor virtual, lo que resulta en mejoras de rendimiento. Los contenedores se pueden crear e implementar mucho más rápido que los servidores virtuales, lo que admite un proceso más ágil y facilita la integración continua.

Además, los contenedores permiten rastrear las versiones de un código de software y sus dependencias. Algunas imágenes de contenedores proveen la capacidad de brindar un archivo de manifiesto que permite a los propietarios y desarrolladores de servicios en la nube mantener y realizar un seguimiento de las versiones de un contenedor y su software, inspeccionar las diferencias entre las diferentes versiones y retroceder a versiones anteriores, cuando sea necesario.

Alojamiento de contenedores y Pods

Normalmente se implementa un solo proceso de un servicio en la nube en cada contenedor, aunque se puede implementar más de un servicio o proceso en la nube en cada uno, si es necesario. En algunos casos, un proceso principal y sus procesos secundarios o funciones adicionales se implementan en el mismo contenedor.

La cantidad de recursos que consume cada contenedor se puede restringir. También es posible limitar los recursos de TI externos que son visibles y accesibles para un contenedor y los procesos que se ejecutan en el contenedor.

Los servicios en la nube implementados en un contenedor normalmente comparten el mismo ciclo de vida que el contenedor, lo que significa que el servicio en la nube implementado se iniciará, detendrá, pausará o reanudará cuando lo haga el contenedor.

Se pueden implementar múltiples contenedores en una construcción lógica llamada *pod*. Un pod es un grupo de uno o más contenedores que tienen almacenamiento y/o recursos de red compartidos, y también comparten la misma configuración que determina cómo se ejecutarán los contenedores. Por lo general, se emplea un pod cuando hay diferentes servicios en la nube que forman parte de la misma aplicación o espacio de nombres y que deben ejecutarse con una sola dirección IP. El pod establece este entorno, al tiempo que garantiza que los servicios en la nube se puedan aislar para que no afecten el tiempo de ejecución de otros contenedores.

Elementos fundamentales de la arquitectura de contenedores

Para comprender cómo funcionan los contenedores, es necesario identificar los elementos arquitectónicos más básicos que componen un entorno de contenedor y le permiten proporcionar sus funciones esenciales. Esta sección describe brevemente estos elementos arquitectónicos.

Motor de contenedores

El componente clave de la arquitectura de contenedores es el *container engine* (motor de contenedores), también conocido como *containerization engine*. El motor de contenedores es un software especializado que se implementa en un sistema operativo para abstraer los recursos necesarios y permitir la definición y el despliegue de contenedores. El software de motor de contenedor se puede implementar en máquinas físicas o máquinas virtuales. Cada motor de contenedor proporciona un conjunto de herramientas de administración y comandos/API para crear, modificar, programar, ejecutar, detener, iniciar o eliminar los contenedores.

Container Build File

Un container build file es un descriptor (creado por el usuario o el servicio) que representa los requerimientos de la aplicación y los servicios que se ejecutan dentro del contenedor, así como los parámetros de configuración requeridos por el motor del contenedor para crear e implementar el contenedor. La sintaxis y el formato del container build file así como los parámetros de configuración que define dependen de la elección del container engine.

Imagen de contenedor

El motor de contenedor utiliza una imagen de contenedor para implementar una imagen basada en requisitos predefinidos. Por ejemplo, si una aplicación requiere un componente de base de datos o un servicio de servidor web para funcionar, el usuario define estos requisitos en el archivo de creación del contenedor. Según las descripciones definidas, el container engine personaliza la imagen del sistema operativo y los comandos o servicios necesarios para la aplicación. Esta imagen personalizada normalmente es una imagen inmutable de solo lectura, que permite que la aplicación o los servicios desplegados en el contenedor funcionen y realicen tareas, pero previene que se vayan a hacer modificaciones.

Container

El contenedor es una instancia ejecutable de una imagen de contenedor predefinida o personalizada que contiene uno o más programas de software, más comúnmente una aplicación o servicio. Si bien los contenedores están aislados entre sí, es posible que necesiten acceder a un recurso compartido a través de la red, como un sistema de archivos o un recurso de TI remoto. Esto es posible sin impactar el aislamiento de los contenedores.

Cada contenedor puede tener una aplicación o proceso ejecutándose en él. Los contenedores también pueden albergar múltiples aplicaciones, servicios o procesos. Las aplicaciones implementadas en un contenedor normalmente se programan con el contenedor, lo que significa que comienzan y terminan con el contenedor.

Networking Address

Cada contenedor tiene su propia dirección de red (como una dirección IP) que se utiliza para comunicarse con otros contenedores y componentes externos. Un contenedor se puede conectar a más de una red asignando direcciones de red adicionales al contenedor.

Los contenedores utilizan la tarjeta de red física o virtual que despliega el container engine para comunicarse con otros contenedores y recursos de TI. Cuando es necesario implementar y aislar varias aplicaciones, los contenedores se utilizan para aislar las aplicaciones entre ellas mismas al mismo tiempo que comparten una dirección IP, así los contenedores pueden ser implementados en un pod. Aunque compartir dispositivos de almacenamiento entre contenedores dentro de un pod es opcional, todos los contenedores dentro del pod comparten la misma dirección IP.

Dispositivo de almacenamiento

De manera similar a la dirección de red, un contenedor puede conectarse a uno o más dispositivos de almacenamiento que están disponibles para los contenedores a través de la red. Cada contenedor tiene su propio nivel de acceso a los almacenamientos definidos por el sistema o los administradores.

3.7. Ejemplo de Estudio de Caso

DTGOV ha ensamblado infraestructuras compatibles con la nube en cada uno de sus centros de datos, que se componen de los siguientes elementos:

- Infraestructura de instalaciones de nivel 3, que proporciona configuraciones redundantes para todos los subsistemas centrales en la capa de instalaciones del centro de datos.
- Conexiones redundantes con proveedores de servicios públicos que tienen capacidad local instalada para la generación de energía y suministro de agua que se activa en caso de falla general.
- Una conexión inter redes que proporciona una interconexión de ultra alto ancho de banda entre los tres centros de datos a través de enlaces dedicados.
- Conexiones de Internet redundantes en cada centro de datos a múltiples ISP y la extranet .GOV, que interconecta a DTGOV con sus principales clientes gubernamentales.
- Hardware estandarizado de mayor capacidad agregada que se abstrae mediante una plataforma de virtualización compatible con la nube.

Los servidores físicos están organizados en racks de servidores, cada uno de los cuales tiene dos router switches redundantes en la parte superior del bastidor (capa 3) que están conectados a cada servidor físico. Estos router switches están interconectados con cores-switches LAN que se han configurado como un clúster. Los core-switches se conectan a routers que brindan capacidades de internetworking y firewalls que brindan capacidades de control de acceso a la red. La Figura 3.13 ilustra la capa física de las conexiones de red del servidor dentro del centro de datos.

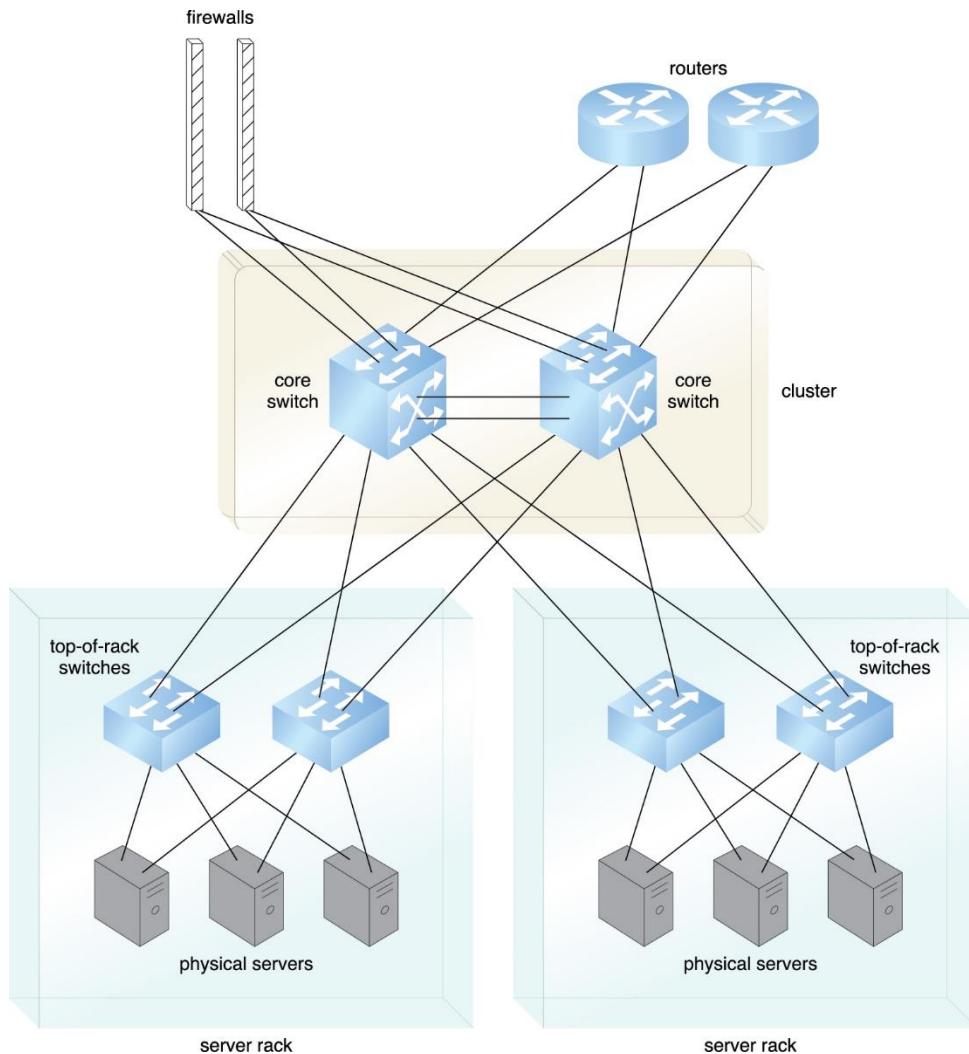


Figura 3.13 Una vista de las conexiones de red del servidor dentro del centro de datos DTGOV.

Una red separada que conecta los sistemas de almacenamiento con los servidores se instala con un clúster de switches SAN(Storage Area Network) y conexiones similares redundantes a varios dispositivos (Figura 3.14).

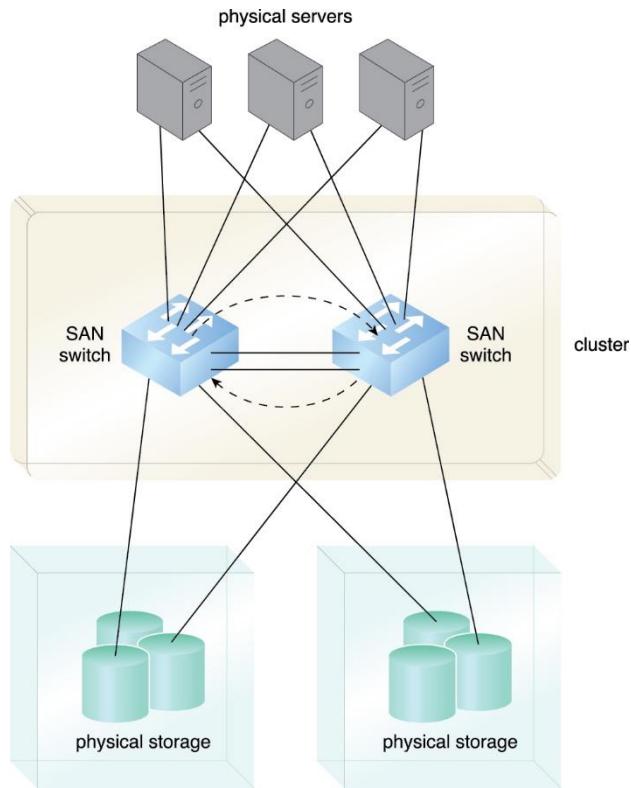


Figura 3.14 Vista de las conexiones de red del sistema de almacenamiento dentro del centro de datos DTGOV.

La Figura 3.15 ilustra una arquitectura de interconexión de redes que se establece entre cada par de centros de datos dentro de la infraestructura corporativa de DTGOV.

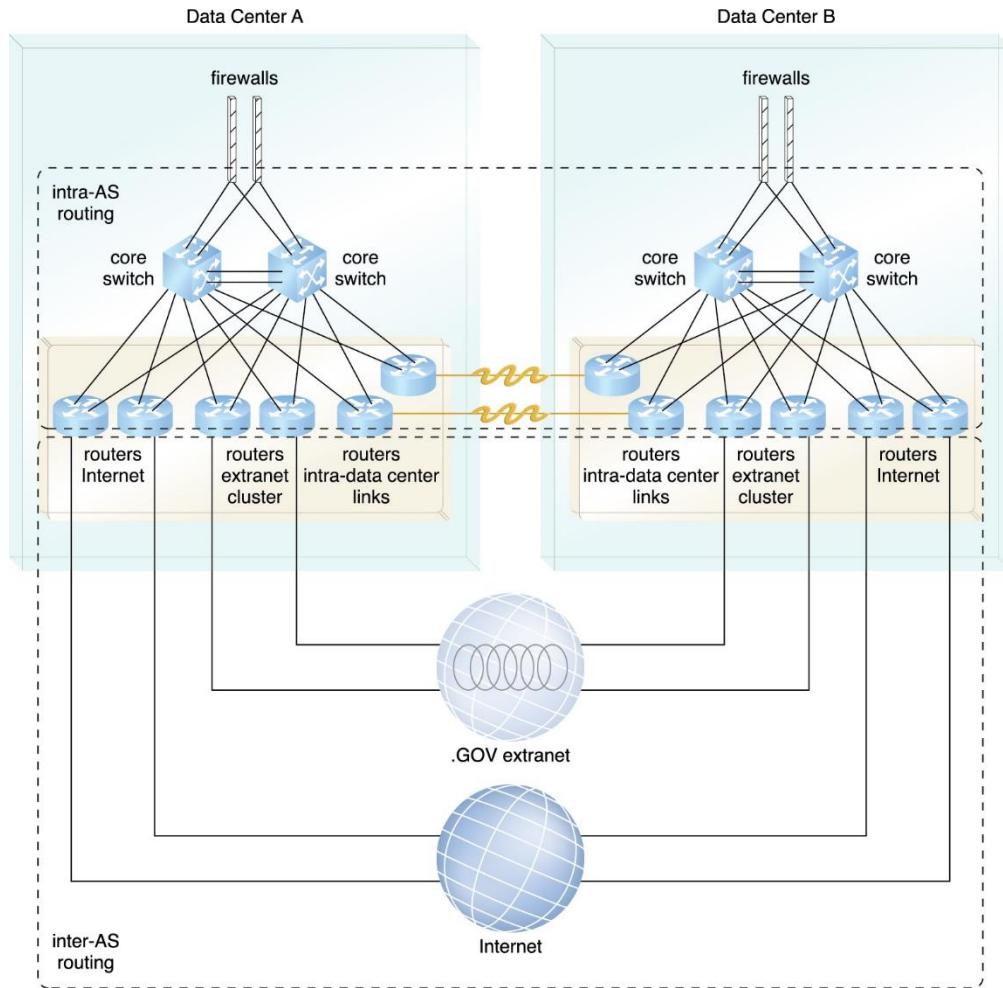


Figura 3.15 La configuración de interconexión de redes entre dos centros de datos que se implementa de manera similar entre cada par de centros de datos DTGOV. La interred DTGOV está diseñada para ser un sistema autónomo (AS) en Internet, lo que significa que los enlaces que interconectan los centros de datos con las LANs definen el dominio de enruteamiento intra-AS. Las interconexiones a los ISP externos se controlan a través de la tecnología de enruteamiento inter-AS, que dan forma al tráfico de Internet y permite configuraciones flexibles para el balanceo de carga y el failover.

Como se muestra en las Figuras 3.14 y 3.15, la combinación de recursos de TI físicos interconectados con recursos de TI virtualizados en la capa física permite la configuración y asignación dinámicas y bien administradas de recursos de TI virtuales.

4 Seguridad en la nube



Este capítulo presenta términos y conceptos que abordan la seguridad básica de la información dentro de las nubes y luego concluye definiendo un conjunto de amenazas y ataques comunes a los entornos de nube pública.

4.1. Términos y conceptos básicos

La seguridad de la información es un conjunto complejo de técnicas, tecnologías, regulaciones y comportamientos que colaborativamente protegen la integridad y el acceso a los sistemas informáticos y los datos. Las medidas de seguridad de TI tienen como objetivo defenderse contra amenazas e interferencias que surgen tanto de intenciones maliciosas como de errores no intencionales del usuario.

Las próximas secciones definen los términos de seguridad fundamentales relevantes para la computación en la nube y describen los conceptos asociados.

Confidencialidad

La confidencialidad es la característica de que algo se hace accesible solo a las partes autorizadas (Figura 4.1). Dentro de los entornos de nube, la confidencialidad se relaciona principalmente con la restricción del acceso a los datos en tránsito y almacenamiento.

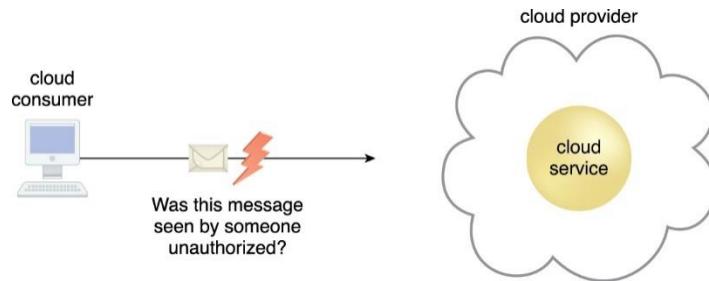


Figura 4.1 El mensaje emitido por el consumidor de la nube al servicio de la nube se considera confidencial solo si este no es accedido o leído por una parte no autorizada.

Integridad

La integridad es la característica de no haber sido alterada por un tercero no autorizado (Figura 4.2). Un tema importante relacionado con la integridad de los datos en la nube es si se puede garantizar a un consumidor de la nube que los datos que transmite a un servicio en la nube coinciden con los datos recibidos por ese servicio en la nube. La integridad puede extenderse a cómo los servicios en la nube y los recursos de TI basados en la nube almacenan, procesan y recuperan los datos.

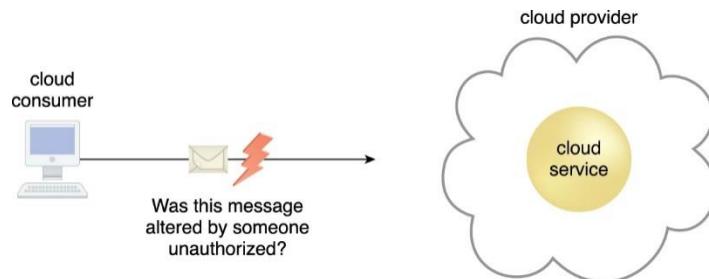


Figura 4.2 Se considera que el mensaje emitido por el consumidor de la nube al servicio de la nube tiene integridad si no ha sido alterado.

Autenticidad

La autenticidad es la característica de que algo ha sido proporcionado por una fuente autorizada. Este concepto abarca el no repudio, que es la incapacidad de una parte para negar o cuestionar la autenticación de una interacción. La autenticación en interacciones no repudiables proporciona prueba de que estas interacciones están vinculadas de manera única a una fuente autorizada.

Disponibilidad

Disponibilidad es la característica de ser accesible y utilizable durante un período de tiempo específico. En entornos de nube típicos, la disponibilidad de los servicios en la nube puede ser una responsabilidad compartida por el proveedor de la nube y el operador directo de la nube.

Amenaza

Una amenaza es una posible violación de la seguridad que puede desafiar las defensas en un intento de violar la privacidad y/o causar daño. Tanto las amenazas instigadas de forma manual como automática están diseñadas para explotar las debilidades conocidas, también conocidas como vulnerabilidades. Una amenaza que se lleva a cabo resulta en un *ataque*.

Vulnerabilidad

Una vulnerabilidad es una debilidad que puede explotarse porque está protegida por controles de seguridad insuficientes o porque los controles de seguridad existentes son superados por un ataque. Las vulnerabilidades de los recursos de TI pueden tener una variedad de causas, incluidas las deficiencias de configuración, las debilidades de la política de seguridad, los errores de usuario, las fallas de hardware o firmware, los errores de software y la arquitectura de seguridad deficiente.

Riesgo

El riesgo es la posibilidad de pérdida o daño derivado de la realización de una actividad. El riesgo generalmente se mide de acuerdo con su nivel de amenaza y la cantidad de vulnerabilidades posibles o conocidas. Dos métricas que se pueden usar para determinar el riesgo de un recurso de TI son:

- La probabilidad de que ocurra una amenaza para explotar vulnerabilidades en el recurso de TI.
- La expectativa de pérdida si el recurso de TI se ve comprometido.

Los detalles sobre la gestión de riesgos se tratan más adelante en este capítulo.

Controles de seguridad

Los controles de seguridad son contramedidas que se utilizan para prevenir o responder a las amenazas de seguridad y para reducir o evitar el riesgo. Los detalles sobre cómo usar las contramedidas de seguridad generalmente se describen en la política de seguridad, que contiene un conjunto de reglas y prácticas que especifican cómo implementar un sistema, servicio o plan de seguridad para la máxima protección de los recursos de TI críticos y confidenciales.

Mecanismos de seguridad

Las contramedidas generalmente se describen en términos de mecanismos de seguridad, que son componentes que comprenden un marco defensivo que protege los recursos, la información y los servicios de TI.

Políticas de seguridad

Una política de seguridad establece un conjunto de normas y reglamentos de seguridad. A menudo, las políticas de seguridad definirán aún más cómo se implementan y hacen cumplir estas reglas y regulaciones. Por ejemplo, la ubicación y el uso de controles y mecanismos de seguridad pueden determinarse mediante políticas de seguridad.

4.2. Threat Agents

Un threat agent (*agente de amenaza*) es una entidad que representa una amenaza porque es capaz de llevar a cabo un ataque. Las amenazas a la seguridad en la nube pueden originarse interna o externamente, a partir de humanos o programas de software. Los agentes de amenazas correspondientes se describen en las próximas secciones. La Figura 4.3 ilustra el papel que asume un agente de amenazas en relación con las vulnerabilidades, amenazas y riesgos, y las salvaguardas establecidas por las políticas de seguridad y los mecanismos de seguridad.

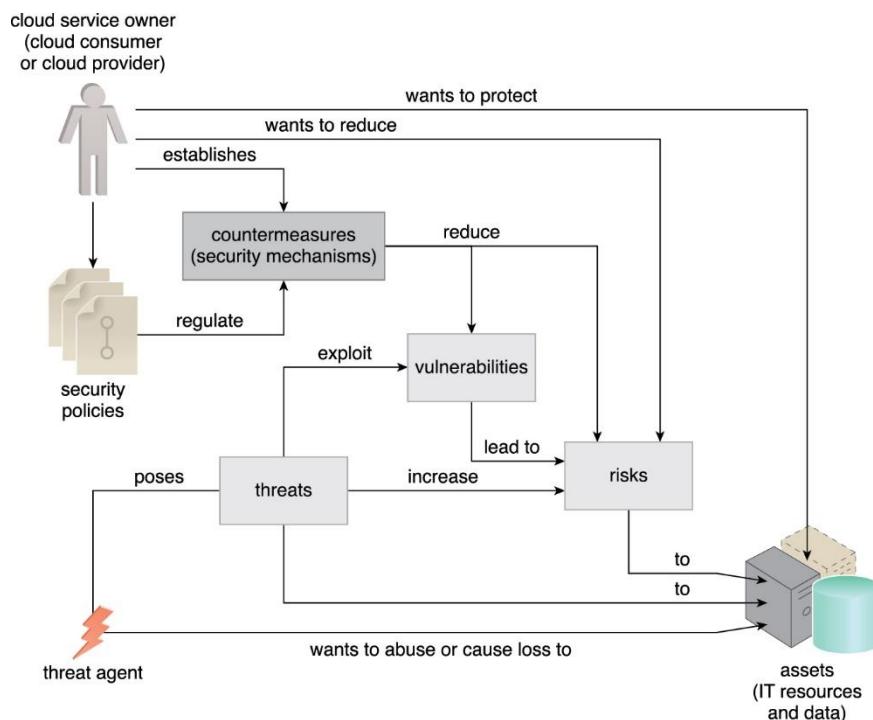


Figura 4.3 Como las políticas de seguridad y mecanismos de seguridad son utilizados contra amenazas, vulnerabilidades, y riesgos causados por threat agents.

Atacante anónimo

Un atacante anónimo es un consumidor de servicios en la nube que no ha sido descubierto y que no tiene permisos en la nube (Figura 4.4). Normalmente existe como un programa de software externo que lanza ataques a nivel de red a través de redes públicas. Cuando los atacantes anónimos tienen información limitada sobre políticas y defensas de seguridad, esto puede disminuir su capacidad para formular ataques efectivos. Por lo tanto, los atacantes anónimos a menudo recurren a cometer actos como saltarse las cuentas de los usuarios o robar las credenciales de los usuarios, mientras usan métodos que garantizan el anonimato o solicitan recursos sustanciales para conseguir su objetivo.



Figura 4.4 La notación utilizada para un atacante anónimo.

Agente de servicio malicioso

Un agente de servicio malicioso puede interceptar y reenviar el tráfico de red que fluye dentro de una nube (Figura 4.5). Por lo general, existe como un agente de servicio (o un programa que pretende ser un agente de servicio) con una lógica comprometida o maliciosa. También puede existir como un programa externo capaz de interceptar de forma remota y potencialmente corromper el contenido de los mensajes.



Figura 4.5 La notación utilizada para un agente de servicio malicioso.

Trusted Attacker

Un trusted attacker (atacante de confianza) comparte los recursos de TI en el mismo entorno de nube que el consumidor de la nube e intenta explotar las credenciales legítimas para atacar a los proveedores de la nube y los inquilinos de la nube con quienes comparten los recursos de TI (Figura 4.6). A diferencia de los atacantes anónimos (que no son de confianza), los atacantes de confianza suelen lanzar sus ataques desde dentro de los límites de confianza de una nube abusando de credenciales legítimas o mediante la apropiación de información sensible y confidencial.



Figura 4.6 La notación que se utiliza para un atacante de confianza.

Los atacantes de confianza (también conocidos como *malicious tenants*) pueden usar los recursos de TI basados en la nube para una amplia gama de abusos, incluida la piratería de procesos de autenticación débiles, la ruptura del cifrado, el envío de correo no deseado a cuentas de correo electrónico o para lanzar ataques comunes, como las campañas de denegación de servicio.

Insider Malicious

Los insider malicious (internos maliciosos) son agentes de amenazas humanos que actúan en nombre de o en relación con el proveedor de la nube. Por lo general, son empleados actuales o anteriores o terceros con acceso a las instalaciones del proveedor de la nube. Este tipo de agente de amenazas conlleva un tremendo potencial de daño, ya que el insider malicious puede tener privilegios administrativos para acceder a los recursos de TI del consumidor en la nube.

Nota

Una notación utilizada para representar una forma general de ataque impulsado por humanos es la estación de trabajo combinada con un rayo (Figura 4.7). Este símbolo genérico no implica un agente de amenaza específico, solo que un ataque se inició a través de una estación de trabajo.



Figura 4.7 La notación utilizada para un ataque que se origina en una estación de trabajo. El símbolo humano es opcional.

4.3. Amenazas de seguridad en la nube

Esta sección presenta varias amenazas y vulnerabilidades comunes en entornos basados en la nube y describe las funciones de los agentes de amenazas antes mencionados.

Espionaje de tráfico

El espionaje de tráfico ocurre cuando los datos que se transfieren hacia o dentro de una nube (generalmente del consumidor de la nube al proveedor de la nube) son interceptados pasivamente por un agente de servicio malicioso con fines ilegítimos de recopilación de información (Figura 4.8). El objetivo de este ataque es comprometer directamente la confidencialidad de los datos y, posiblemente, la confidencialidad de la relación entre el consumidor de la nube y el proveedor de la nube. Debido a la naturaleza pasiva del ataque, puede pasar desapercibido más fácilmente durante largos períodos de tiempo.

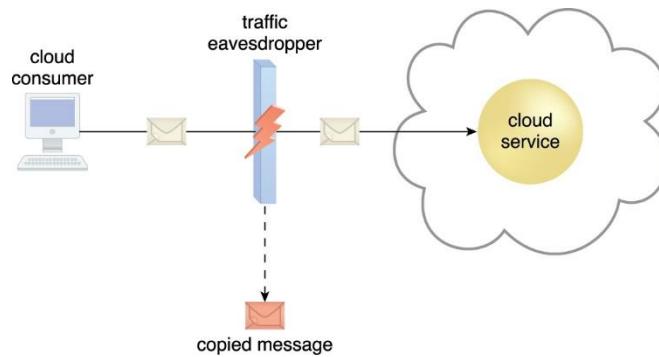


Figura 4.8 Un agente de servicio malicioso posicionado externamente lleva a cabo un ataque de espionaje de tráfico al interceptar un mensaje enviado por el consumidor del servicio en la nube al servicio en la nube. El agente de servicio hace una copia no autorizada del mensaje antes de enviarlo por su ruta original al servicio en la nube.

Intermediario malicioso

La amenaza del intermediario malicioso surge cuando los mensajes son interceptados y alterados por un agente de servicio malicioso, lo que puede comprometer la confidencialidad y/o la integridad del mensaje. también puede insertar datos dañinos en el mensaje antes de reenviarlo a su destino. La Figura 4.9 ilustra un ejemplo común del ataque de un intermediario malicioso.

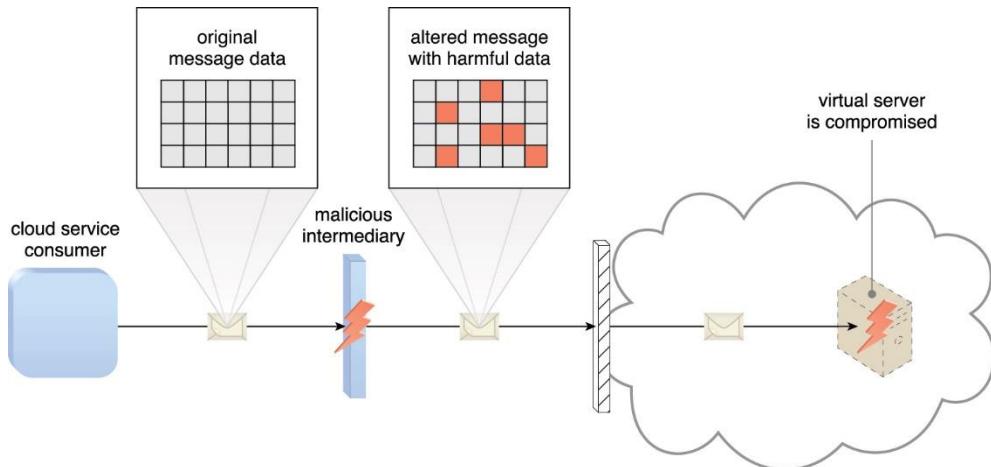


Figura 4.9 El agente de servicio malicioso intercepta y modifica un mensaje enviado por un consumidor de un servicio de la nube al servicio de la nube (no mostrado) hospedado en un servidor virtual. Debido a los datos dañinos almacenados en el mensaje, el servidor virtual estará comprometido.

Nota

Si bien no es tan común, el ataque de intermediario malicioso también puede ser llevado a cabo por un programa malicioso de consumidor de servicios en la nube.

Denegación de servicio

El objetivo del ataque de denegación de servicio (DoS) es sobrecargar los recursos de TI hasta el punto en que no puedan funcionar correctamente. Esta forma de ataque comúnmente se lanza de una de las siguientes maneras:

- La carga de trabajo en los servicios en la nube aumenta artificialmente con mensajes de imitación o solicitudes de comunicación repetidas.
- La red está sobrecargada de tráfico para reducir su capacidad de respuesta y paralizar su rendimiento.
- Se envían múltiples solicitudes de servicios en la nube, cada una de las cuales está diseñada para consumir memoria y recursos de procesamiento excesivos.

Los ataques DoS exitosos producen la degradación y/o la falla del servidor, como se ilustra en la figura 4.10.

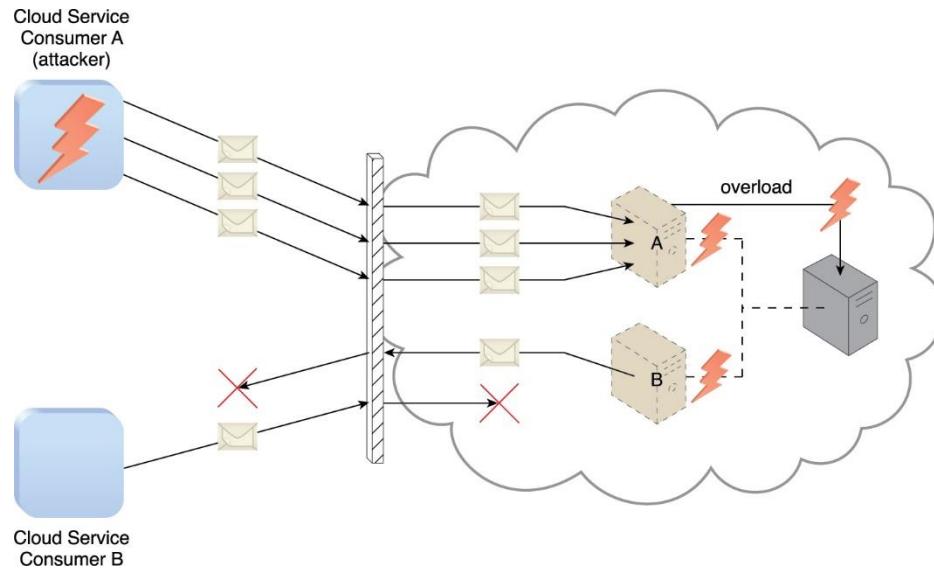


Figura 4.10 El consumidor de servicios en la nube A envía varios mensajes a un servicio en la nube (no mostrado) alojado en el servidor virtual A. Esto sobrecarga la capacidad del servidor físico subyacente, lo que provoca interrupciones en los servidores virtuales A y B. Como resultado, los consumidores legítimos del servicio en la nube, como el Consumidor de servicios en la nube B, no pueden comunicarse con ningún servicio en la nube alojado en los Servidores virtuales A y B.

Autorización insuficiente

El ataque de autorización insuficiente se produce cuando se concede acceso a un atacante de forma errónea o demasiado permisiva, lo que hace que el atacante obtenga acceso a los recursos de TI que normalmente están protegidos. Esto es a menudo el resultado de que el atacante obtenga acceso directo a los recursos de TI que se implementaron bajo el supuesto de que solo accederían a ellos los programas de consumo confiables (Figura 4.11).

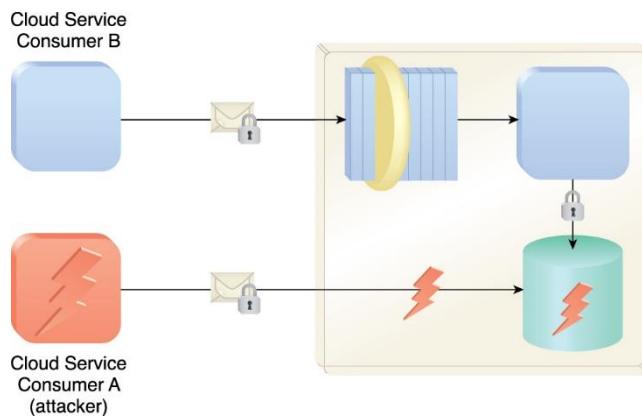


Figura 4.11 El Consumidor de servicios en la nube A obtiene acceso a una base de datos que se implementó bajo el supuesto de que solo se accedería a través de un servicio web con un contrato de servicio publicado (como ocurre con el Consumidor de Servicios en la Nube B).

Una variación de este ataque, conocida como autenticación débil, puede resultar cuando se usan contraseñas débiles o cuentas compartidas para proteger los recursos de TI. Dentro de los entornos

de nube, estos tipos de ataques pueden generar impactos significativos según el rango de recursos de TI y el rango de acceso a esos recursos que obtiene el atacante (Figura 4.12).

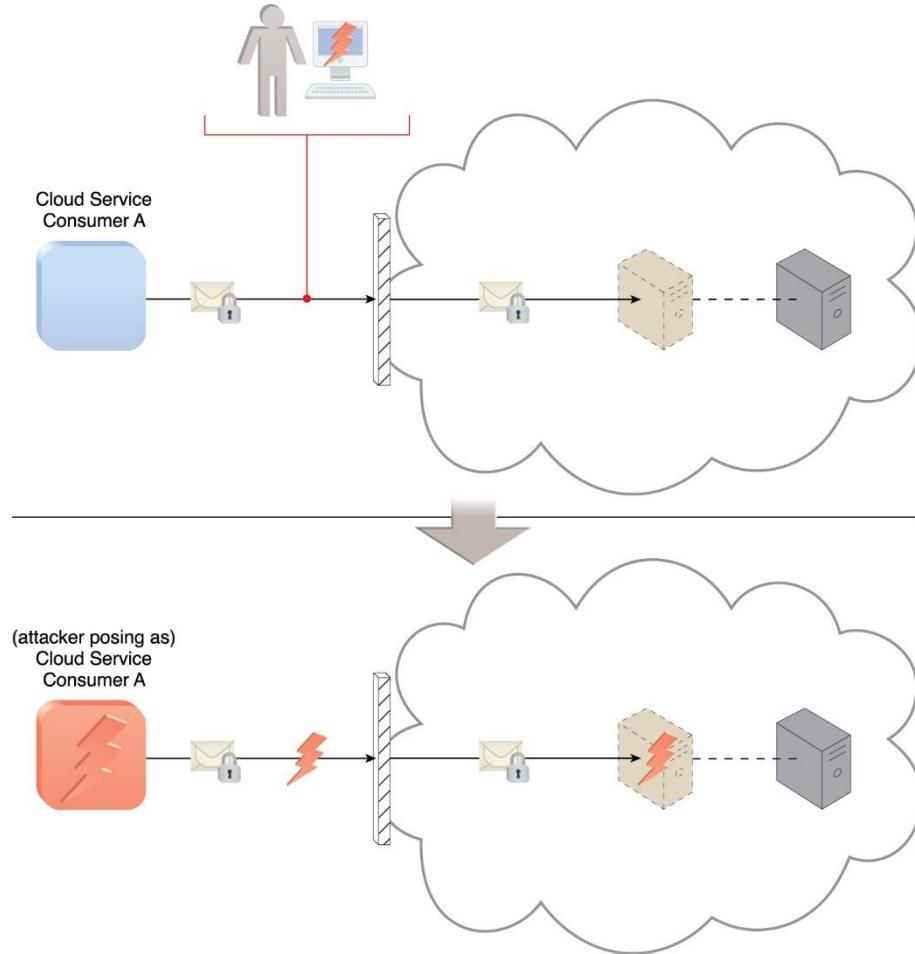


Figura 4.12 Un atacante ha descifrado una contraseña débil utilizada por el Consumidor de servicios en la nube A. Como resultado, un consumidor de servicios en la nube malicioso (propiedad del atacante) está diseñado para hacerse pasar por Consumidor de servicios en la nube A para obtener acceso al servidor virtual basado en la nube.

Ataque de virtualización

La virtualización brinda a múltiples consumidores de la nube acceso a recursos de TI que comparten hardware subyacente pero que están lógicamente aislados entre sí. Debido a que los proveedores de la nube otorgan a los consumidores de la nube acceso administrativo a los recursos de TI virtualizados (como servidores virtuales), existe un riesgo inherente de que los consumidores de la nube puedan abusar de este acceso para atacar los recursos de TI físicos subyacentes.

Un ataque de virtualización explota vulnerabilidades en la plataforma de virtualización para poner en peligro su confidencialidad, integridad y/o disponibilidad. Esta amenaza se ilustra en la Figura 4.13, donde un atacante de confianza accede con éxito a un servidor virtual para comprometer al servidor físico subyacente. Con las nubes públicas, donde un solo recurso de TI físico puede proporcionar recursos de TI virtualizados a múltiples consumidores de la nube, un ataque de este tipo puede tener repercusiones significativas.

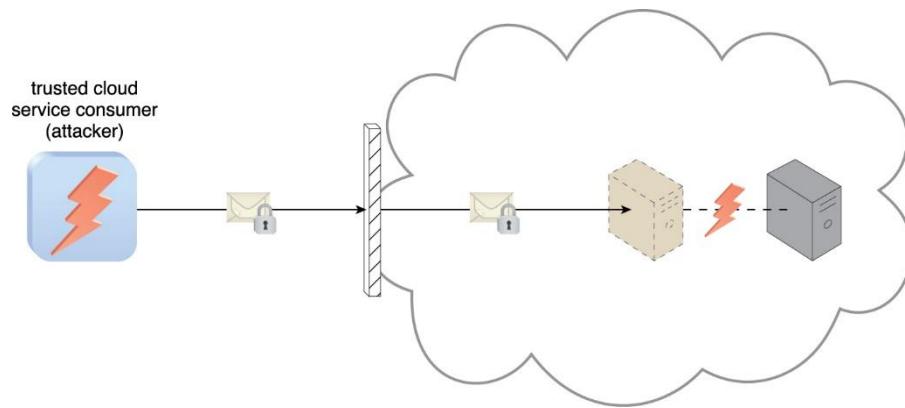


Figura 4.13 Un consumidor de servicios en la nube autorizado lleva a cabo un ataque de virtualización al abusar de su acceso administrativo a un servidor virtual para explotar el hardware subyacente.

Límites de confianza superpuestos

Si los recursos de TI físicos dentro de una nube son compartidos por diferentes consumidores de servicios en la nube, estos consumidores de servicios en la nube tienen límites de confianza superpuestos. Los consumidores de servicios en la nube maliciosos pueden tener como objetivo a los recursos de TI compartidos con la intención de comprometer a los consumidores de la nube u otros recursos de TI que comparten el mismo límite de confianza. La consecuencia es que algunos o todos los demás consumidores de servicios en la nube podrían verse afectados por el ataque y/o el atacante podría usar recursos de TI virtuales contra otros que también comparten el mismo límite de confianza.

La figura 4.14 ilustra un ejemplo en el cual dos consumidores de servicio en la nube comparten servidores virtuales hospedados por el mismo servidor físico, y como resultado sus límites de confianza se superponen.

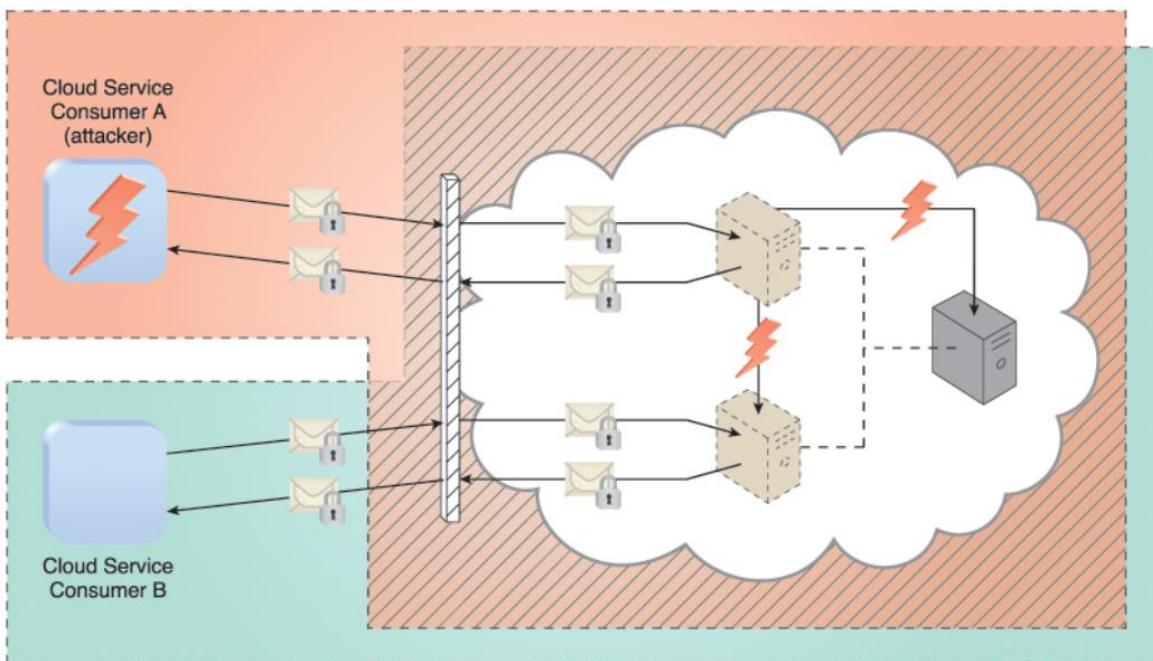


Figura 4.14 El consumidor de servicios en la nube A es de confianza para la nube, por lo tanto, obtiene acceso a un servidor virtual, al que luego ataca con la intención de atacar el servidor físico subyacente y al servidor virtual utilizado por el Consumidor de servicios en la nube B.

Ataque de contenedores

El uso de contenedores introduce una falta de aislamiento del nivel del sistema operativo del host. Dado que los contenedores implementados en la misma máquina comparten el mismo sistema operativo host, las amenazas a la seguridad pueden aumentar porque se puede obtener acceso a todo el sistema. Si el host subyacente se ve comprometido, todos los contenedores que se ejecutan en el host pueden verse afectados.

Los contenedores se pueden crear desde un sistema operativo que se ejecuta en un servidor virtual. Esto puede ayudar a garantizar que, si se produce una infracción de seguridad que afecta al sistema operativo en el que se ejecuta un contenedor, el atacante solo puede obtener acceso y modificar el sistema operativo del servidor virtual o los contenedores que se ejecutan en un único servidor virtual, mientras que otros servidores virtuales (o servidores físicos) permanecen intactos.

Otra opción es un modelo de implementación de un servicio por servidor físico donde todas las imágenes de contenedor implementadas en el mismo host son las mismas. Esto puede reducir el riesgo sin necesidad de virtualizar los recursos de TI. En este caso, una brecha de seguridad en una instancia de servicio en la nube solo permitiría el acceso a otras instancias, y el riesgo residual podría considerarse aceptable. Sin embargo, este enfoque puede no ser óptimo para implementar muchos servicios en la nube diferentes porque puede aumentar significativamente la cantidad total de recursos físicos de TI que deben implementarse y administrarse, al mismo tiempo que aumenta aún más el costo y la complejidad operativa.

4.4. consideraciones adicionales

Esta sección proporciona una lista de verificación diversa de problemas y pautas relacionadas con la seguridad en la nube. Las consideraciones enumeradas no están en ningún orden en particular.

Implementaciones defectuosas

El diseño, la implementación o la configuración deficientes de las implementaciones de servicios en la nube pueden tener consecuencias no deseadas, más allá de las excepciones y fallas en el tiempo de ejecución. Si el software y/o el hardware del proveedor de la nube tienen fallas de seguridad inherentes o debilidades operativas, los atacantes pueden explotar estas vulnerabilidades para afectar la integridad, la confidencialidad y/o la disponibilidad de los recursos de TI del proveedor de la nube y los recursos de TI del consumidor de la nube alojados por el proveedor de la nube.

La Figura 4.15 muestra un servicio en la nube mal implementado que provoca el apagado del servidor. Aunque en este escenario la falla es expuesta accidentalmente por un consumidor legítimo de servicios en la nube, un atacante podría haberla descubierto y explotado fácilmente.

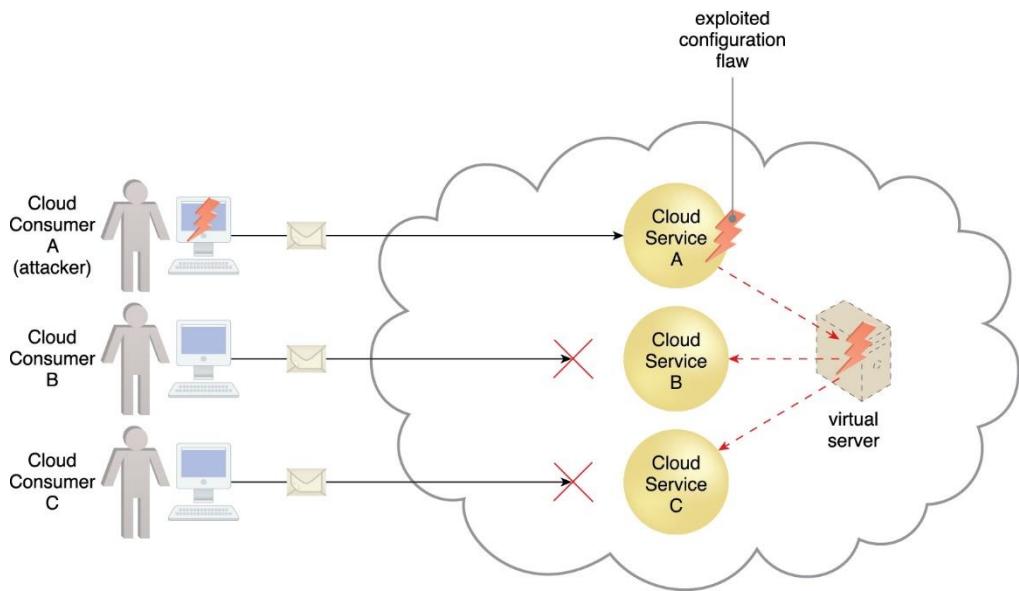


Figura 4.15 El mensaje del consumidor de servicios en la nube A desencadena una falla de configuración en el Servicio en la nube A, lo que a su vez hace que el servidor virtual que también aloja los Servicios en la nube B y C se bloquee.

Disparidad de políticas de seguridad

Cuando un consumidor de la nube coloca recursos de TI con un proveedor de nube pública, es posible que deba aceptar que su enfoque tradicional de seguridad de la información puede no ser idéntico o ni siquiera similar al del proveedor de la nube. Esta incompatibilidad debe evaluarse para garantizar que los datos u otros activos de TI que se reubiquen en una nube pública estén adecuadamente protegidos. Incluso cuando se rentan recursos crudos basados en infraestructura de TI, es posible que al consumidor de la nube no se le otorgue suficiente control administrativo o influencia sobre las políticas de seguridad que se aplican a los recursos de TI alquilados del proveedor de la nube. Esto se debe principalmente a que esos recursos de TI aún son propiedad legal del proveedor de la nube y continúan estando bajo su responsabilidad.

Además, con algunas nubes públicas, terceros adicionales, como brokers de seguridad y autoridades de certificación, pueden introducir su propio conjunto distinto de políticas y prácticas de seguridad, lo que complica aún más cualquier intento de estandarizar la protección de los activos de los consumidores en la nube.

Contratos

Los consumidores de la nube deben examinar detenidamente los contratos y acuerdos de nivel de servicio (SLAs) presentados por los proveedores de la nube para asegurarse de que las políticas de seguridad y otras garantías relevantes sean satisfactorias en lo que respecta a la seguridad de los activos. Debe haber un lenguaje claro que indique la cantidad de responsabilidad asumida por el proveedor de la nube y/o el nivel de indemnización que el proveedor de la nube puede solicitar. Cuanto mayor sea la responsabilidad asumida por el proveedor de la nube, menor será el riesgo para el consumidor de la nube.

Otro aspecto de las obligaciones contractuales es dónde se trazan las líneas entre el consumidor de la nube y los activos del proveedor de la nube. Un consumidor de la nube que implementa su propia

solución en la infraestructura proporcionada por el proveedor de la nube producirá una arquitectura tecnológica compuesta por artefactos que pertenecen tanto al consumidor de la nube como al proveedor de la nube. Si ocurre una brecha de seguridad (u otro tipo de falla en el tiempo de ejecución), ¿cómo se determina la culpa? Además, si el consumidor de la nube puede aplicar sus propias políticas de seguridad a su solución, pero el proveedor de la nube insiste en que su infraestructura de soporte se rija por políticas de seguridad diferentes (y tal vez incompatibles), ¿cómo se puede superar la disparidad resultante?

A veces, la mejor solución es buscar un proveedor de nube diferente con términos contractuales más compatibles.

4.5. Ejemplo de Estudio de Caso

Con base en una evaluación de sus aplicaciones internas, los analistas de ATN identifican un conjunto de riesgos. Uno de esos riesgos está asociado con la aplicación myTrendek que se adoptó de OTC, una empresa que ATN adquirió recientemente. Esta aplicación incluye una función que analiza el uso del teléfono y de Internet, y habilita un modo multiusuario que otorga distintos derechos de acceso. Por lo tanto, a los administradores, supervisores, auditores y usuarios regulares se les pueden asignar diferentes privilegios. La base de usuarios de la aplicación incluye usuarios internos y externos, como socios comerciales y contratistas.

La aplicación myTrendek plantea una serie de desafíos de seguridad relacionados con el uso por parte del personal interno:

- la autenticación no requiere ni impone contraseñas complejas
- la comunicación con la aplicación no está encriptada
- Las regulaciones europeas (ETelReg) requieren que ciertos tipos de datos recopilados por la aplicación sean eliminados después de seis meses

ATN planea migrar esta aplicación a una nube a través de un entorno PaaS, pero la amenaza de autenticación débil y la falta de confidencialidad que admite la aplicación los hacen reconsiderar. Una evaluación de riesgos posterior revela además que si la aplicación se migra a un entorno PaaS alojado en una nube que reside fuera de Europa, las regulaciones locales pueden entrar en conflicto con ETelReg. Dado que el proveedor de la nube no está preocupado por el cumplimiento de ETelReg, esto podría resultar fácilmente en sanciones monetarias para ATN. Con base en los resultados de la evaluación de riesgos, ATN decide no continuar con su plan de migración a la nube.

5 Mecanismos en la infraestructura de la nube



Los mecanismos en la infraestructura de la nube son bloques de construcción fundamentales de los entornos de la nube que establecen mecanismos para formar la base de la arquitectura fundamental de la tecnología de la nube.

Los siguientes mecanismos de infraestructura de nube se describen en este capítulo:

- Perímetro de red lógica
- Servidor virtual
- Dispositivo de almacenamiento en la nube
- Monitor de uso de la nube
- Replicación de recursos
- Entorno Ready-Meade

No todos estos mecanismos son necesariamente de amplio alcance, ni cada uno establece una capa arquitectónica individual. En su lugar, deben verse como componentes centrales que son comunes a las plataformas en la nube.

5.1. Perímetro de red lógica

Definido como el aislamiento de un entorno de red del resto de la red de comunicaciones, el perímetro de red lógica establece un límite de red virtual que puede abarcar y aislar un grupo de recursos de TI relacionados basados en la nube que pueden estar físicamente distribuidos (Figura 5.1).

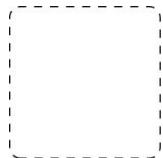


Figura 5.1 La notación de línea discontinua utilizada para indicar el límite de un perímetro de red lógica.

Este mecanismo se puede implementar para:

- aislar los recursos de TI en una nube de los usuarios no autorizados
- aislar los recursos de TI en una nube de los que no son usuarios
- aislar los recursos de TI en una nube de los consumidores de la nube
- controlar el ancho de banda que está disponible para los recursos aislados de TI

Los perímetros de red lógicos se establecen normalmente a través de dispositivos de red que suministran y controlan la conectividad de un centro de datos y se implementan comúnmente como entornos de TI virtualizados que incluyen:

- *Firewall virtual*: un recurso de TI que filtra activamente el tráfico de red hacia y desde la red aislada mientras controla sus interacciones con Internet.

- *Red virtual*: generalmente adquirido a través de VLANs, este recurso de TI aísla el entorno de red dentro de la infraestructura del centro de datos.

La Figura 5.2 introduce la notación utilizada para denotar estos dos recursos de TI. La Figura 5.3 muestra un escenario en el que un perímetro de red lógica contiene el entorno local de un consumidor de nube, mientras que otro contiene el entorno basado en la nube de un proveedor de nube. Estos perímetros están conectados a través de una VPN que protege las comunicaciones, ya que la VPN generalmente se implementa mediante el cifrado punto a punto de los paquetes de datos enviados entre los endpoints que se comunican.

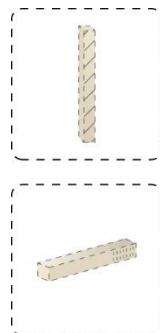


Figura 5.2 Los símbolos utilizados para representar un firewall virtual (arriba) y una red virtual (abajo).

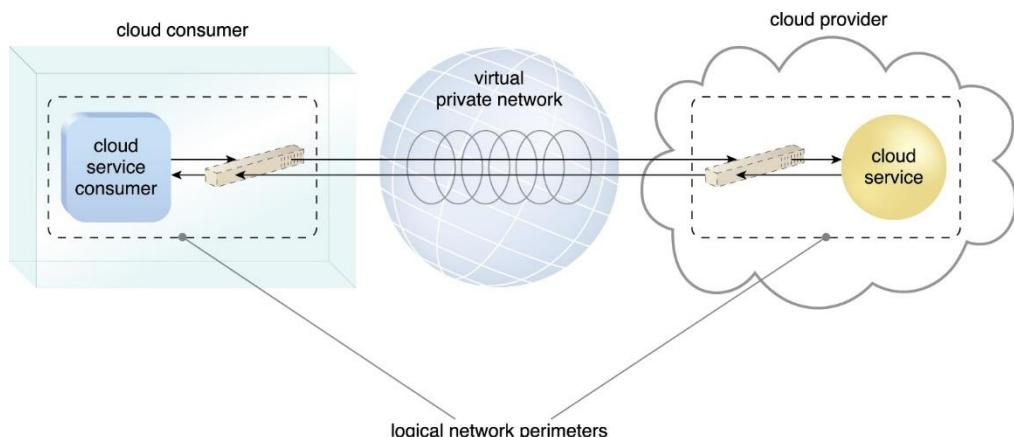


Figura 5.3 Dos perímetros de red lógica rodean los ambientes del consumidor de nube y del proveedor de nube.

Ejemplo de Estudio de Caso

DTGOV ha virtualizado su infraestructura de red para producir un diseño de red lógico que favorece la segmentación y el aislamiento de la red. La Figura 5.4 muestra el perímetro de red lógica implementado en cada centro de datos de DTGOV, de la siguiente manera:

- Los enrutadores que se conectan a Internet y la extranet²² están conectados en red a firewalls externos, que brindan control y protección de la red hasta los límites de la red externa más lejanos mediante redes virtuales que abstraen lógicamente los perímetros de la red externa y la extranet.

²² Una extranet es una red privada que utiliza protocolos de Internet y probablemente infraestructura pública de comunicación para compartir de forma segura información. Fuente: Wikipedia.

Los dispositivos conectados a estos perímetros de red están ligeramente aislados y protegidos de usuarios externos. No hay recursos de TI consumibles disponibles dentro de estos perímetros.

- Se establece un perímetro de red lógica clasificado como zona desmilitarizada (DMZ) entre los cortafuegos externos y los cortafuegos propios. La DMZ se abstrae como una red virtual que aloja los servidores proxy (que no se muestran en la Figura 5.3) que intermedian el acceso a los servicios de red de uso común (DNS, correo electrónico, portal web), así como a los servidores web con funciones de administración externa.
- El tráfico de red que sale de los servidores proxy pasa a través de un conjunto de firewalls de administración que aíslan la administración del perímetro de red, la cual alberga los servidores que brindan la mayor parte de los servicios de administración a los que los consumidores de la nube pueden acceder externamente. Estos servicios se brindan como soporte directo del autoservicio y la asignación bajo demanda de recursos de TI basados en la nube.
- Todo el tráfico a los recursos de TI basados en la nube fluye a través de la DMZ hacia los firewalls del servicio de la nube que aíslan la red perimetral de cada consumidor de la nube, que se extrae mediante una red virtual que también está aislada de otras redes.
- Tanto el perímetro de administración como las redes virtuales aisladas están conectados a los firewalls del centro de datos interno, que regulan el tráfico de red hacia y desde los otros centros de datos DTGOV que también están conectados a los enrutadores del centro de datos interno en el perímetro de la red del centro de datos interno.

Los cortafuegos virtuales están asignados y controlados por un único consumidor de nube para regular su tráfico de recursos de TI virtuales. Estos recursos de TI están conectados a través de una red virtual que está aislada de otros consumidores de la nube. El cortafuegos virtual y la red virtual aislada forman conjuntamente el perímetro de la red lógica del consumidor de la nube.

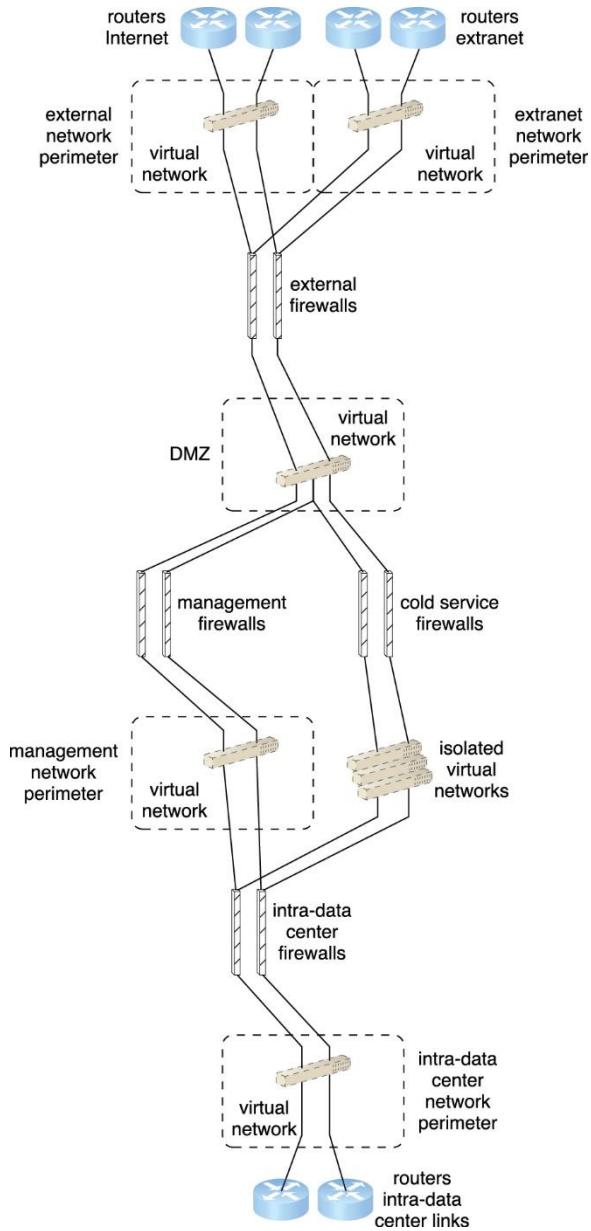


Figura 5.4 Se establece un diseño de red lógica a través de un conjunto de perímetros de red lógica utilizando varios firewalls y redes virtuales.

5.2. Servidor virtual

Un *servidor virtual* es una forma de software de virtualización que emula un servidor físico. Los servidores virtuales son utilizados por los proveedores de la nube para compartir el mismo servidor físico con múltiples consumidores de la nube al proporcionar a los consumidores de la nube instancias de servidores virtuales individuales. La Figura 5.5 muestra tres servidores virtuales alojados en dos servidores físicos. La cantidad de instancias que puede compartir un servidor físico determinado está limitada por su capacidad.

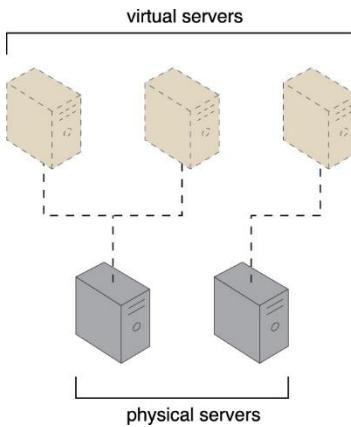


Figura 5.5 El primer servidor físico alberga dos servidores virtuales, mientras que el segundo servidor físico alberga un servidor virtual.

Nota

- Los términos servidor virtual y máquina virtual (VM) se usan como sinónimos en este libro.
- El mecanismo del hipervisor al que se hace referencia en este capítulo se describe con más detalle en el siguiente capítulo.

Como un mecanismo básico, el servidor virtual representa el bloque de construcción más fundamental de los entornos de nube. Cada servidor virtual puede alojar numerosos recursos de TI, soluciones basadas en la nube y algunos otros mecanismos de computación en la nube. La creación de instancias de servidores virtuales a partir de archivos imagen es un proceso de asignación de recursos que se puede completar rápidamente y bajo demanda.

Los consumidores de la nube que instalan o rentan servidores virtuales, pueden personalizar sus entornos independientemente de otros consumidores de la nube que puedan estar usando servidores virtuales alojados por el mismo servidor físico subyacente. La Figura 5.6 muestra un servidor virtual que aloja un servicio en la nube al que accede el consumidor de servicios en la nube B, mientras que el consumidor de servicios en la nube A accede directamente al servidor virtual para realizar una tarea de administración.

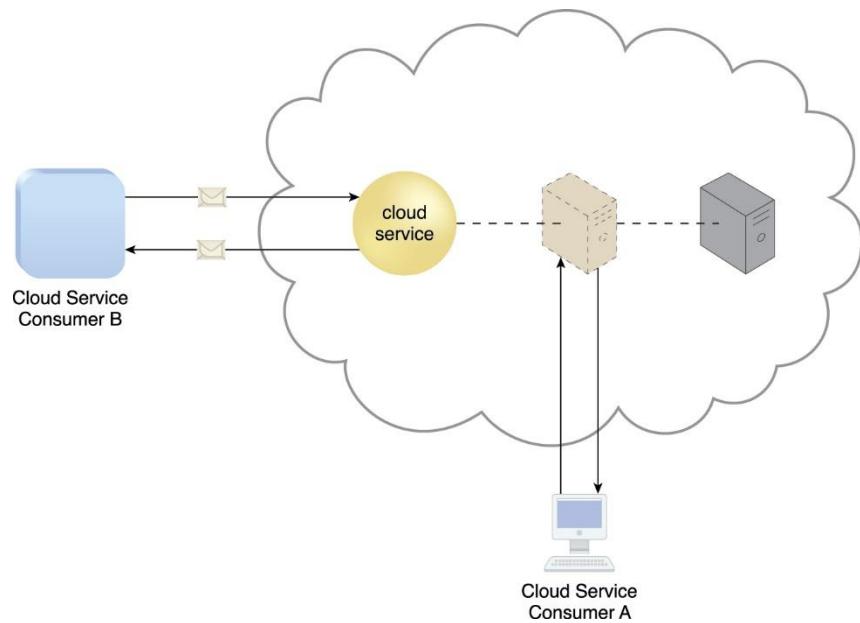
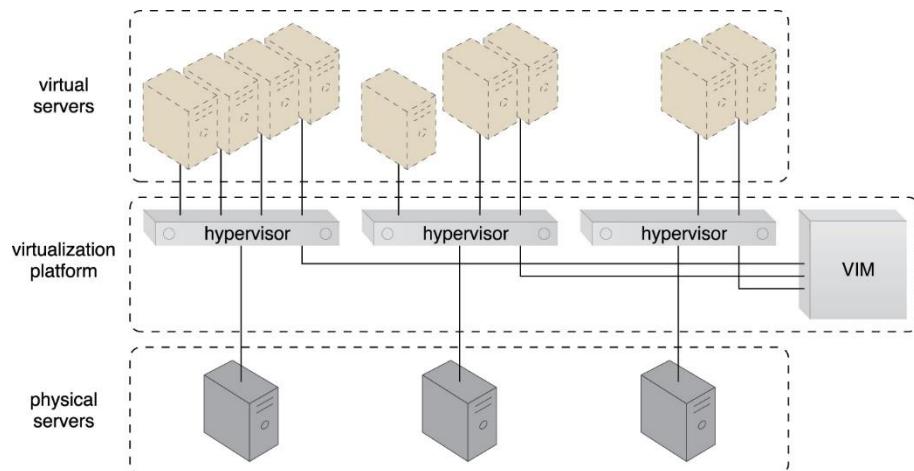


Figura 5.6 Un servidor virtual hospeda un servicio de nube activo y posteriormente es accedido por un consumidor de la nube A para propósitos administrativos.

Ejemplo de Estudio de Caso

El entorno de IaaS de DTGOV contiene servidores virtuales alojados que se instanciaron en servidores físicos que ejecutan el mismo software de hipervisor que controla los servidores virtuales. Su VIM²³ se utiliza para coordinar los servidores físicos en relación con la creación de instancias de servidores virtuales. Este enfoque se utiliza en cada centro de datos para aplicar una implementación uniforme de la capa de virtualización.

La figura 5.7 muestra varios servidores virtuales que se ejecutan sobre servidores físicos, todos los cuales están controlados conjuntamente por un VIM central.



²³ VIM (Virtual Infrastructure Manager) hace referencia al administrador de infraestructura virtual que se describe con mayor precisión en el capítulo siete.

Figura 5.7 Los servidores virtuales se crean a través de los hipervisores de los servidores físicos y un VIM central.

Para permitir la creación de servidores virtuales bajo demanda, DTGOV proporciona a los consumidores de la nube un conjunto de plantillas²⁴ de servidores virtuales que están disponibles a través de imágenes de VM prefabricadas.

Estas imágenes de VM son archivos que representan las imágenes de disco virtual utilizadas por el hipervisor para iniciar el servidor virtual. DTGOV permite que las plantillas de servidores virtuales tengan varias opciones de configuración inicial que difieren según el sistema operativo, los controladores y las herramientas de administración que se utilicen. Algunas plantillas de servidores virtuales también tienen una aplicación de software de servidor preinstalado adicional.

Los siguientes paquetes de servidores virtuales se ofrecen a los consumidores de la nube de DTGOV. Cada paquete tiene diferentes configuraciones y limitaciones de rendimiento predefinidas:

- Instancia pequeña de Servidor Virtual - 1 núcleo de procesador virtual, 4 GB de RAM virtual, 20 GB de espacio de almacenamiento en el sistema de archivos raíz
- Instancia mediana de servidor virtual - 2 núcleos de procesador virtual, 8 GB de RAM virtual, 20 GB de espacio de almacenamiento en el sistema de archivos raíz
- Instancia grande de servidor virtual - 8 núcleos de procesador virtual, 16 GB de RAM virtual, 20 GB de espacio de almacenamiento en el sistema de archivos raíz
- Instancia de servidor virtual de memoria amplia - 8 núcleos de procesador virtual, 64 GB de RAM virtual, 20 GB de espacio de almacenamiento en el sistema de archivos raíz
- Instancia de servidor virtual de procesador grande - 32 núcleos de procesador virtual, 16 GB de RAM virtual, 20 GB de espacio de almacenamiento en el sistema de archivos raíz
- Instancia de servidor virtual ultragrande - 128 núcleos de procesador virtual, 512 GB de RAM virtual, 40 GB de espacio de almacenamiento en el sistema de archivos raíz

Se puede agregar capacidad de almacenamiento adicional a un servidor virtual conectando un disco virtual desde un dispositivo de almacenamiento en la nube. Todas las imágenes de plantillas de máquinas virtuales se almacenan en un dispositivo de almacenamiento en la nube común al que solo se puede acceder a través de las herramientas de administración de los consumidores de la nube que se utilizan para controlar los recursos de TI implementados. Una vez que se necesita crear una instancia de un nuevo servidor virtual, el consumidor de la nube puede elegir la plantilla de servidor virtual más adecuada de la lista de configuraciones disponibles. Se crea una copia de la imagen de la máquina virtual y se asigna al consumidor de la nube, quien luego puede asumir las responsabilidades administrativas.

La imagen de la máquina virtual asignada se actualiza cada vez que el consumidor de la nube personaliza el servidor virtual. Una vez que el consumidor de la nube inicia el servidor virtual, la

²⁴ Una plantilla o template, es un medio o aparato o sistema, que permite guiar, portar, o construir, un diseño o esquema predefinido. Una plantilla agiliza el trabajo de reproducción o de muchas copias idénticas o casi idénticas. Fuente: Wikipedia.

imagen de la máquina virtual asignada y su perfil de rendimiento asociado se pasan al VIM, que crea la instancia del servidor virtual a partir del servidor físico adecuado.

DTGOV utiliza el proceso descrito en la Figura 5.8 para respaldar la creación y administración de servidores virtuales que tienen diferentes configuraciones iniciales de software y características de rendimiento.

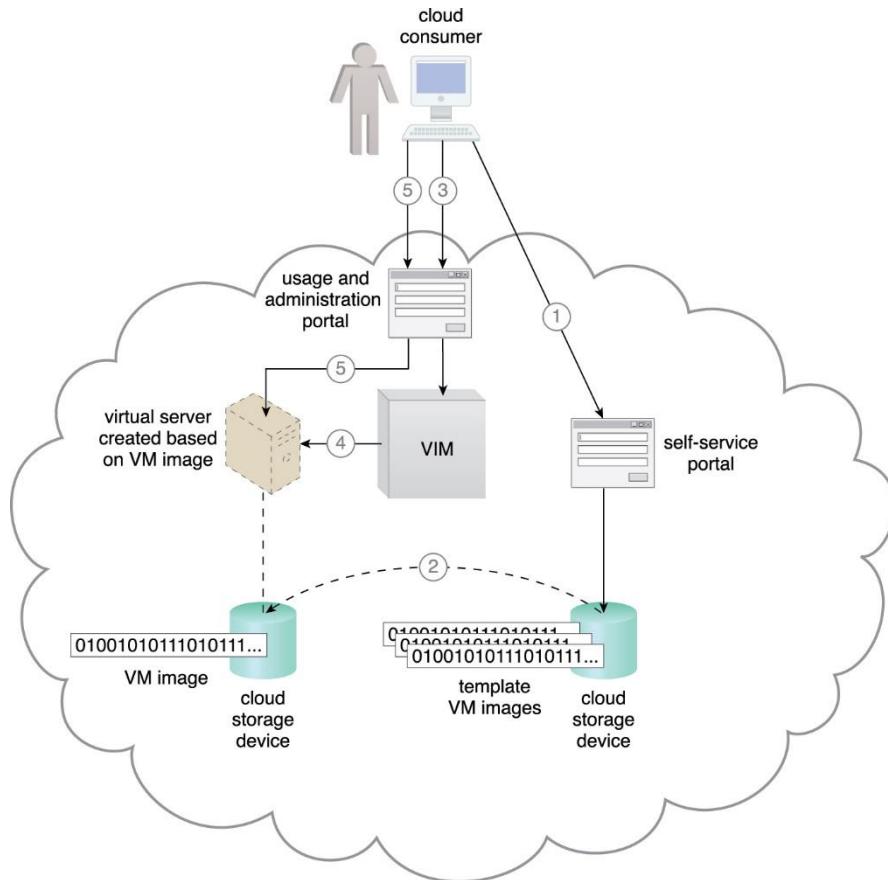


Figura 5.8 El consumidor de la nube utiliza el portal de autoservicio para seleccionar una plantilla de servidor virtual para el proceso de creación (1). Se crea una copia de la imagen de máquina virtual correspondiente en un dispositivo de almacenamiento en la nube controlado por el consumidor (2). El consumidor de la nube inicia el servidor virtual utilizando el portal de uso y administración (3), que interactúa con el VIM para crear la instancia del servidor virtual a través del hardware subyacente (4). El consumidor de la nube puede usar y personalizar el servidor virtual a través de otras funciones en el portal de uso y administración (5). Tenga en cuenta que el portal de autoservicio y el portal de uso y administración se explican en un capítulo posterior.

5.3. Cloud Storage Device

El mecanismo cloud storage device (dispositivo de almacenamiento en la nube) representa dispositivos de almacenamiento que están diseñados específicamente para el aprovisionamiento basado en la nube. Las instancias de estos dispositivos se pueden virtualizar, de forma similar a como los servidores físicos pueden generar imágenes de servidores virtuales. Los dispositivos de almacenamiento en la nube normalmente son capaces de proveer reserva de espacio en incrementos de tamaño fijo como soporte del mecanismo de pago por uso. Los dispositivos de

almacenamiento en la nube se pueden exponer para su acceso remoto a través de los servicios de almacenamiento en la nube.

Una preocupación primaria relacionada con el almacenamiento en la nube es la seguridad, la integridad y la confidencialidad de los datos, que se vuelven más propensos a verse comprometidos cuando se confían a proveedores de nube externos o a terceros. También puede haber implicaciones legales y regulatorias que resulten de la reubicación de datos a través de fronteras geográficas o nacionales. Otro problema se aplica específicamente al rendimiento de grandes bases de datos. Las LANs proporcionan datos almacenados localmente con redes confiables y niveles de latencia que son superiores a los de las WANs.

Cloud Storage Levels

Los mecanismos de cloud storage device proporcionan unidades lógicas de almacenamiento de datos, como lo son:

- Archivos - las colecciones de datos se agrupan en archivos que se ubican en carpetas.
- Bloques - el nivel más bajo de almacenamiento y el más cercano al hardware, un bloque es la unidad de datos más pequeña a la que aún se puede acceder individualmente.
- Datasets - los conjuntos de datos se organizan en un formato de registro, delimitado o basado en tablas.
- Objetos - los datos y sus metadatos asociados se organizan como recursos basados en Web.

Cada uno de estos niveles de almacenamiento de datos se asocia comúnmente con un cierto tipo de interfaz técnica, la cual corresponde a un tipo particular de dispositivo de almacenamiento en la nube y un servicio de almacenamiento en la nube utilizado para publicar su API (Figura 5.9).

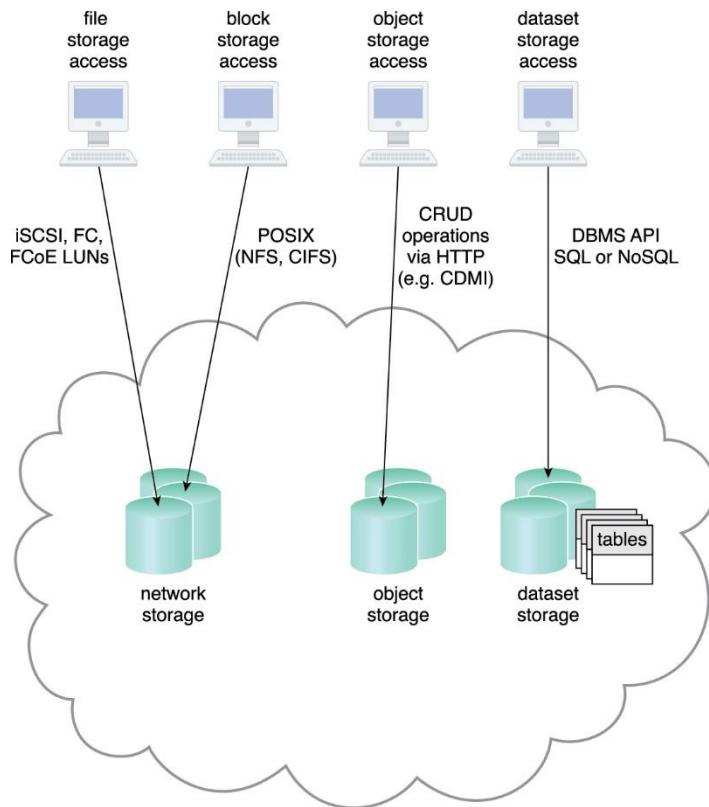


Figura 5.9 Diferentes consumidores de servicios en la nube utilizan diferentes tecnologías para interactuar con dispositivos de almacenamiento en la nube virtualizados. (Adaptado del Cloud Storage Reference Model CDMI).

Network Storage Interfaces

El almacenamiento en red tradicional suele pertenecer a la categoría de interfaces de almacenamiento en red. Incluye dispositivos de almacenamiento que cumplen con los protocolos estándar de la industria para archivos y almacenamiento en red, como SCSI para bloques de almacenamiento y el bloque de mensajes del servidor (SMB), el sistema de archivos común de Internet (CIFS) y el sistema de archivos de red (NFS). El almacenamiento de archivos implica el almacenamiento de datos individuales en archivos separados que pueden tener diferentes tamaños y formatos y estar organizados en carpetas y subcarpetas. Los archivos originales a menudo se reemplazan por los nuevos archivos que se crean cuando los datos se han modificado.

Cuando un mecanismo de cloud storage device se basa en este tipo de interfaz, su rendimiento de búsqueda y extracción de datos tenderá a ser subóptimo. Los niveles de procesamiento de almacenamiento y los umbrales para la asignación de archivos suelen estar determinados por el mismo sistema de archivos. El almacenamiento en bloque requiere que los datos estén en un formato fijo (conocido como *data block*), que es la unidad más pequeña que se puede almacenar y acceder con el formato de almacenamiento más cercano al hardware. El uso del LUN²⁵(Logical Unit Number) o el virtual volumen block-level storage generalmente tendrán un mejor rendimiento que el almacenamiento de nivel de archivo.

²⁵ Un número lógico de unidad (LUN) es un drive lógico que representa una partición en un drive físico.

Object Storage Interfaces

Varios tipos de datos pueden ser referenciados y almacenados como recursos web. Esto se conoce como almacenamiento de objetos, que se basa en tecnologías que pueden admitir una variedad de tipos de datos y medios. Por lo general, se puede acceder a los mecanismos del dispositivo de almacenamiento en la nube que implementan esta interfaz a través de REST o servicios en la nube basados en servicios web que utilizan HTTP como protocolo principal. La Storage Networking Industry Association's Cloud Data Management Interface (SNIA's CDMI) soporta el uso de interfaces de almacenamiento de objetos.

Database Storage Interfaces

Los mecanismos de dispositivos de almacenamiento en la nube basados en database storage interfaces suelen admitir un lenguaje de consulta además de las operaciones básicas de almacenamiento. La gestión del almacenamiento se lleva a cabo utilizando una API estándar o una interfaz de usuario administrativa. Esta clasificación de la interfaz de almacenamiento se divide en dos categorías principales según la estructura de almacenamiento, de la siguiente manera.

Almacenamiento de datos relacionales

Tradicionalmente, muchos entornos de TI on-premise almacenan datos mediante bases de datos relacionales o sistemas de administración de bases de datos relacionales (RDBMSs). Las bases de datos relacionales (o dispositivos de almacenamiento relacional) se basan en tablas para organizar datos similares en filas y columnas. Las tablas pueden tener relaciones entre sí para dar a los datos una mayor estructura, proteger la integridad de los datos y evitar la redundancia de datos (lo que se conoce como normalización de datos). Trabajar con almacenamiento relacional suele implicar el uso del lenguaje de consulta estructurado (SQL) estándar.

Un mecanismo de dispositivo de almacenamiento en la nube implementado mediante el almacenamiento de datos relacionales podría basarse en cualquier cantidad de productos de bases de datos disponibles comercialmente, como IBM DB2, Oracle Database, Microsoft SQL Server y MySQL.

Los desafíos con las bases de datos relacionales basadas en la nube comúnmente se relacionan con la escalabilidad y el rendimiento. Escalar verticalmente un dispositivo de almacenamiento en la nube relacional puede ser más complejo y caro que escalar horizontalmente. Las bases de datos con relaciones complejas y/o que contienen grandes volúmenes de datos pueden verse afectadas por una mayor latencia y sobrecarga de procesamiento, especialmente cuando se accede de forma remota a través de servicios en la nube.

Almacenamiento de datos no relacionales

El almacenamiento no relacional (también conocido comúnmente como almacenamiento NoSQL) se aleja del modelo de base de datos relacional tradicional en el sentido de que establece una "estructura más flexible para los datos almacenados con menos énfasis en la definición de relaciones y la normalización de datos". La motivación para usar el almacenamiento no relacional es evitar la complejidad potencial y la sobrecarga de procesamiento que pueden imponer las bases de datos relacionales. Además, el almacenamiento no relacional puede ser más escalable horizontalmente que el almacenamiento relacional.

La desventaja del almacenamiento no relacional es que los datos pierden gran parte de la forma nativa y la validación debido a esquemas o modelos de datos limitados o primitivos. Además, los repositorios no relacionales no suelen admitir funciones de bases de datos relacionales, como transacciones o uniones.

Los datos normalizados exportados a un repositorio de almacenamiento no relacional normalmente se desnormalizarán, lo que significa que el tamaño de los datos normalmente crecerá. Se puede conservar un grado de normalización, pero normalmente no para relaciones complejas. Los proveedores de la nube a menudo ofrecen almacenamiento no relacional que brinda escalabilidad y disponibilidad de los datos almacenados en múltiples entornos de servidor. Sin embargo, muchos mecanismos de almacenamiento no relacional son propietarios y, por lo tanto, pueden limitar gravemente la portabilidad de los datos.

Ejemplo de Estudio de Caso

DTGOV brinda a los consumidores de la nube acceso a un dispositivo de almacenamiento en la nube basado en una interfaz de almacenamiento de objetos. El servicio en la nube que expone esta API ofrece funciones básicas en los objetos almacenados, como buscar, crear, eliminar y actualizar. La función de búsqueda utiliza una disposición jerárquica de objetos que se asemeja a un sistema de archivos. DTGOV ofrece además un servicio en la nube que se usa exclusivamente con servidores virtuales y permite la creación de dispositivos de almacenamiento en la nube a través de una interfaz de red de almacenamiento en bloque. Ambos servicios en la nube usan APIs que cumplen con CDMI v1.0 de SNIA.

El dispositivo de almacenamiento en la nube basado en objetos tiene un sistema de almacenamiento subyacente con capacidad de almacenamiento variable, que está controlado directamente por un componente de software que también expone la interfaz. Este software permite la creación de dispositivos de almacenamiento en la nube aislados que se asignan a los consumidores de la nube. El sistema de almacenamiento utiliza un sistema de administración de credenciales de seguridad para administrar el control de acceso basado en el usuario a los objetos de datos del dispositivo (Figura 5.10).

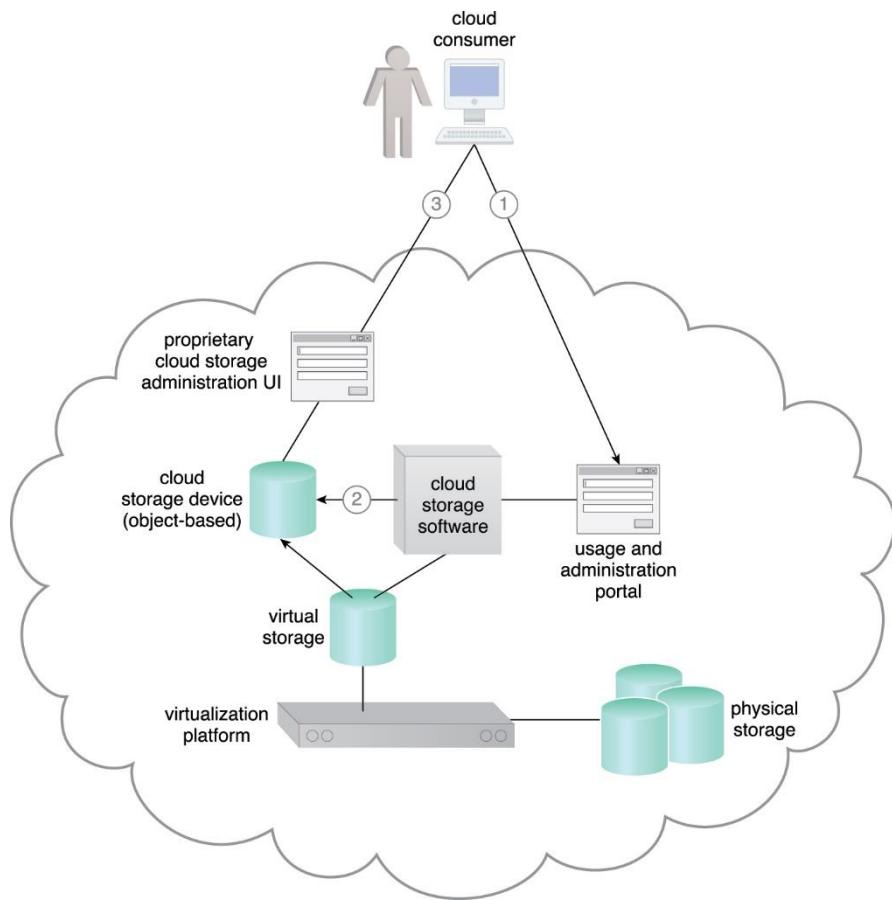


Figura 5.10 El consumidor de la nube interactúa con el portal de uso y administración para crear un dispositivo de almacenamiento en la nube y definir políticas de control de acceso (1). El portal de uso y administración interactúa con el software de almacenamiento en la nube para crear la instancia del dispositivo de almacenamiento en la nube y aplicar la política de acceso requerida a sus datos objeto (2). Cada dato objeto se asigna a un dispositivo de almacenamiento en la nube y todos los objetos se almacenan en el mismo volumen de almacenamiento virtual. El consumidor de la nube utiliza la interfaz de usuario del dispositivo de almacenamiento en la nube patentado para interactuar directamente con los objetos (3). (Tenga en cuenta que el portal de uso y administración se explica en otro capítulo posterior).

El control de acceso se otorga por objeto y utiliza políticas de acceso separadas para crear, leer y escribir en cada objeto de datos. Se permiten permisos de acceso público, aunque son de solo lectura. Los grupos de acceso están formados por usuarios nominados que deben registrarse previamente a través del sistema de gestión de credenciales. Se puede acceder a los objetos de datos desde las aplicaciones web y las interfaces de servicios web, que son implementadas por el software de almacenamiento en la nube.

La creación de los dispositivos de almacenamiento en la nube basados en bloques para los consumidores de la nube es administrada por la plataforma de virtualización, que instancia la implementación del LUN del almacenamiento virtual (Figura 5.11). El VIM debe asignar el dispositivo de almacenamiento en la nube (o el LUN) a un servidor virtual existente antes de poder utilizarlo.

La capacidad de los dispositivos de almacenamiento en la nube basados en bloques se maneja por incrementos de un GB. Estos pueden ser creados como de almacenamiento fijo que los

consumidores de la nube pueden modificar administrativamente o como almacenamiento de tamaño variable que tiene una capacidad inicial de 5 GB que aumenta y disminuye automáticamente en incrementos de 5 GB según las demandas de uso.

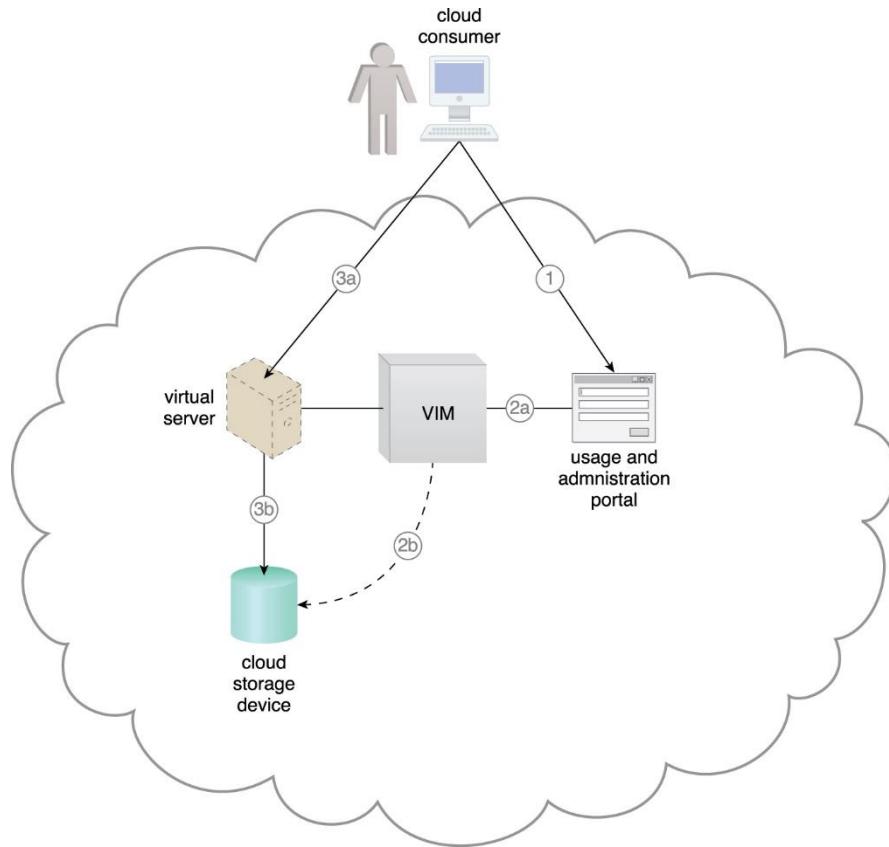


Figura 5.11 El consumidor de la nube usa el portal de uso y administración para crear y asignar un dispositivo de almacenamiento en la nube a un servidor virtual existente (1). El portal de uso y administración interactúa con el software VIM (2a), que crea y configura el LUN apropiado (2b). Cada dispositivo de almacenamiento en la nube utiliza un LUN independiente controlado por la plataforma de virtualización. El consumidor de la nube inicia sesión de forma remota en el servidor virtual directamente (3a) para acceder al dispositivo de almacenamiento en la nube (3b).

5.4. Monitor de uso de la nube

El mecanismo del monitor de uso de la nube es un programa de software liviano y autónomo responsable de recopilar y procesar datos sobre el uso de recursos de TI.

Según el tipo de métricas de uso que se van a recopilar y la manera en que se deben recopilar los datos de uso, los monitores de uso de la nube pueden existir en diferentes formatos. Las próximas secciones describen tres formatos comunes de implementación basados en agentes. Cada uno puede diseñarse para reenviar los datos de uso recopilados a una base de datos de registro con fines de procesamiento posterior y generación de informes.

Agente de monitoreo

Un *monitoring agent* (agente de monitoreo) es un programa intermediario controlado por eventos²⁶ que existe como un agente de servicio y reside a lo largo de las rutas de comunicación existentes para monitorear y analizar de manera transparente los flujos de datos (Figura 5.12). Este tipo de monitor de uso de la nube se usa comúnmente para medir el tráfico de la red y las métricas de mensajes.

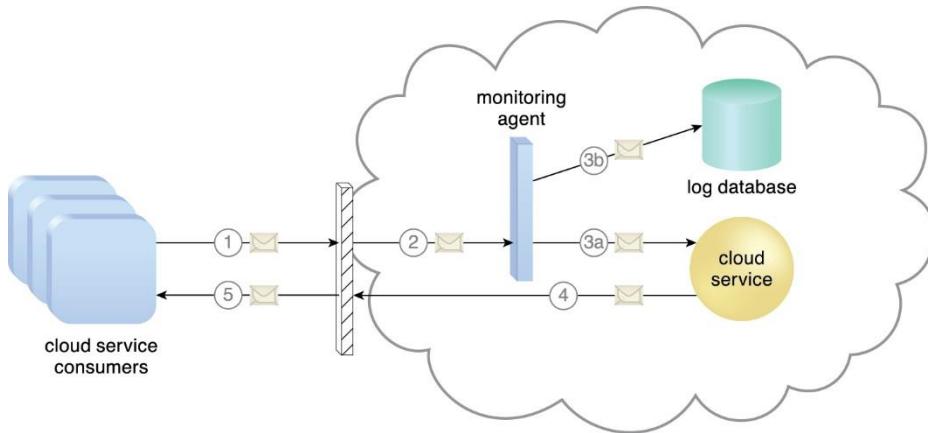


Figura 5.12 Un consumidor de servicios en la nube envía un mensaje de solicitud a un servicio en la nube (1). El agente de monitoreo intercepta el mensaje para recopilar datos de uso relevantes (2) antes de permitir que continúe hacia el servicio en la nube (3a). El agente de monitoreo almacena los datos de uso recopilados en una base de datos de registro (3b). El servicio en la nube responde con un mensaje de respuesta (4) que se envía de vuelta al consumidor del servicio en la nube sin ser interceptado por el agente de monitoreo (5).

Agente de recursos

Un *resource agent* (agente de recursos) es un módulo de procesamiento que recopila datos de uso mediante interacciones basadas en eventos con recursos de software especializados (Figura 5.13). Este módulo se usa para monitorear las métricas de uso basadas en eventos observables predefinidos en el nivel de recursos de software, como iniciar, suspender, reanudar y escalar verticalmente.

²⁶ Para la informática, un evento es una acción que es detectada por un programa; éste, a su vez, puede hacer uso del mismo o ignorarlo. Por lo general, una aplicación cuenta con uno o más hilos de ejecución dedicados a atender los distintos eventos que se le presenten. Entre las fuentes más comunes de eventos se encuentran las acciones del usuario con el teclado o el ratón. Fuente: Wikipedia

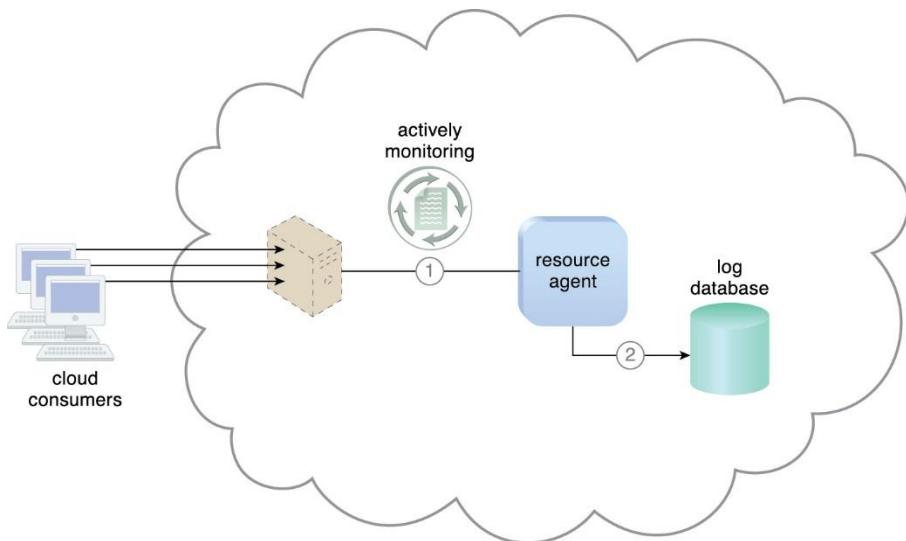


Figura 5.13 El agente de recursos está monitoreando activamente un servidor virtual y detecta un aumento en el uso (1). El agente de recursos recibe una notificación del programa de administración de recursos subyacente de que el servidor virtual se está escalando y almacena los datos de uso recopilados en una base de datos de registro, según sus métricas de monitoreo (2).

Polling agent

Un *polling agent* (agente de sondeo) es un módulo de procesamiento que recopila datos de uso del servicio en la nube mediante el sondeo de los recursos de TI. Este tipo de monitor de servicios en la nube se usa comúnmente para monitorear periódicamente el estado de los recursos de TI, como el tiempo de actividad y el tiempo de inactividad (Figura 5.14).

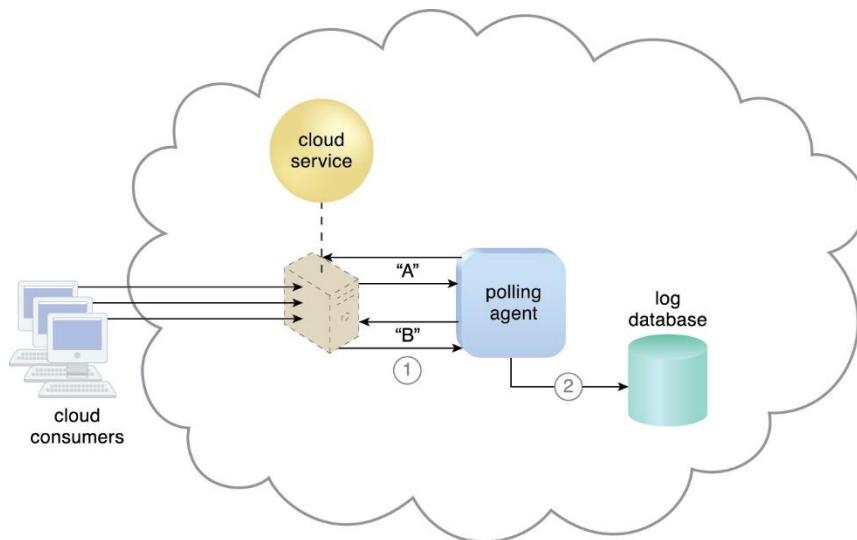


Figura 5.14 Un agente de sondeo monitorea el status de un servicio en la nube hospedado por un servidor virtual mediante el envío de mensajes de solicitud de sondeo periódicos y la recepción de mensajes de respuesta de sondeo que informan el status de uso "A" después de una serie de ciclos de sondeo, hasta que

²⁷ Polling o sondeo en computación hace referencia a una operación de consulta constante, generalmente hacia un dispositivo de hardware, para crear una actividad sincrónica sin el uso de interrupciones, aunque también puede suceder lo mismo para recursos de software. Fuente:Wikipedia.

recibe un status de uso "B" (1), cuando esto sucede el agente de sondeo registra el nuevo status de uso en el log²⁸ database (2).

Ejemplo de Estudio de Caso

Uno de los desafíos encontrados durante la iniciativa de adopción de la nube de DTGOV ha sido garantizar que los datos de uso recopilados sean precisos. Los métodos de asignación de recursos de los modelos de subcontratación de TI anteriores habían dado lugar a que a sus clientes se les facturaran tarifas en función de la cantidad de servidores físicos que figuraban en los contratos de arrendamiento anuales, independientemente del uso real.

DTGOV ahora necesita definir un modelo que permita rentar y facturar por hora servidores virtuales de diferentes niveles de rendimiento. Los datos de uso deben estar en un nivel extremadamente granular para lograr el grado necesario de precisión. DTGOV implementa un agente de recursos que se basa en los eventos de uso de recursos generados por la plataforma VIM para calcular los datos de uso del servidor virtual.

El agente de recursos está diseñado con lógica y métricas que se basan en las siguientes reglas:

1. Cada evento de uso de recursos generado por el software VIM puede contener los siguientes datos:

- Tipo de evento (EV_TYPE) - Generado por la plataforma VIM, hay cinco tipos de eventos:

VM Starting (creación en el hipervisor)

VM Started (finalización del procedimiento de inicio)

VM Stopping (apagado)

VM Stopped (terminación en el hipervisor)

VM Scaled (cambio de parámetros de rendimiento)

- Tipo de VM (VM_TYPE) - Esto representa un tipo de servidor virtual, según lo dictan sus parámetros de rendimiento. Una lista predefinida de posibles configuraciones de servidores virtuales proporciona los parámetros que describen los metadatos cada vez que se inicia o escala una máquina virtual.

- Identificador único de máquina virtual (VM_ID) - Este identificador lo proporciona la plataforma VIM.

- Identificador único del consumidor de la nube (CS_ID) - Otro identificador proporcionado por la plataforma VIM para representar al consumidor de la nube.

²⁸ En informática, se usa el término registro, log o historial de log para referirse a la grabación secuencial en un archivo o en una base de datos de todos los acontecimientos (eventos o acciones) que afectan a un proceso particular (aplicación, actividad de una red informática, etc.). De esta forma constituye una evidencia del comportamiento del sistema. Fuente:Wikipedia.

- Timestamp²⁹ del evento (EV_T): Una identificación de la ocurrencia de un evento que se expresa en formato de fecha y hora, con la zona horaria del centro de datos y con referencia a UTC como se define en RFC 3339 (según el perfil ISO 8601).

2. Las medidas de uso se registran para cada servidor virtual que crea un consumidor de nube.
3. Las mediciones de uso se registran durante un período de medición cuya duración está definida por dos marcas de tiempo denominadas t_{start} y t_{end} . El inicio del período de medición se establece de manera predeterminada al comienzo del mes calendario ($t_{start} = 2012-12-01T00:00:00-08:00$) y finaliza al final del mes calendario ($t_{end} = 2012-12-31T23:59:59-08:00$). También se admiten períodos de medición personalizados.
4. Las mediciones de uso se registran en cada minuto de uso. El período de medición del uso del servidor virtual comienza cuando se crea el servidor virtual en el hipervisor y se detiene cuando finaliza.
5. Los servidores virtuales se pueden iniciar, escalar y detener varias veces durante el período de medición. El intervalo de tiempo entre cada ocurrencia i ($i = 1, 2, 3, \dots$) de estos pares de eventos sucesivos que se declaran para un servidor virtual se denomina ciclo de uso y es conocido como T_{cycle_i}
 - VM_Starting, VM_Stopping – El tamaño de la VM no ha cambiado al final del ciclo.
 - VM_Starting, VM_Scaled – El tamaño de la VM ha cambiado al final del ciclo.
 - VM_Scaled, VM_Scaled – El tamaño de la VM ha cambiado mientras se escalaba al final del ciclo.
 - VM_Scaled, VM_Stopping – El tamaño de la VM ha cambiado al final del ciclo.

6. El uso total, U_{total} , para cada servidor virtual durante el período de medición se calcula utilizando las siguientes ecuaciones de la base de datos log de recursos:

- Para cada VM_TYPE y VM_ID en la base de datos log:

$$U_{total_VM_type_j} = \sum_{t_{start}}^{t_{end}} T_{cycle}$$

- Según el tiempo de uso total que se mide para cada VM_TYPE, el vector de uso para cada VM_ID es

$$U_{total}: U_{total} = \{tipo\ 1, U_{total_VM_type_1}, tipo\ 2, U_{total_VM_type_2}, \dots\}$$

La Figura 7.15 muestra el agente de recursos interactuando con la API controlada por eventos de la VIM.

²⁹ Un timestamp (marca temporal), conocida también como registro de tiempo, o sello de tiempo, es una secuencia de caracteres que denotan la hora y fecha (o alguna de ellas) en la/s que ocurrió determinado evento. Fuente: Wikipedia.

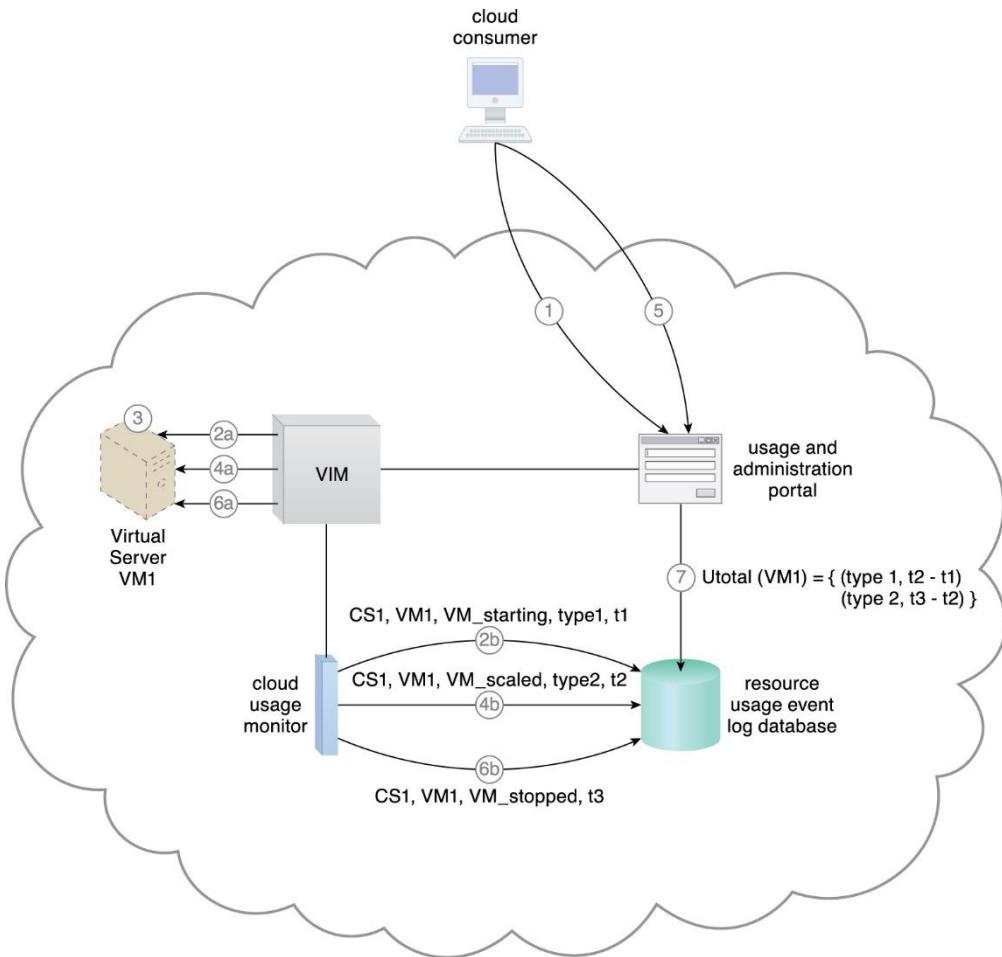


Figura 7.15 El consumidor de la nube ($CS_ID = CS1$) solicita la creación de un servidor virtual ($VM_ID = VM1$) de tamaño de configuración tipo 1 ($VM_TYPE = type1$) (1). El VIM crea el servidor virtual (2a). La API basada en eventos de VIM genera un evento de uso de recursos con marca de tiempo = $t1$, que el agente de software del monitor de uso de la nube captura y registra en la base de datos log de eventos de uso de recursos (2b). El uso del servidor virtual aumenta y alcanza el umbral de escalado automático (3). El VIM amplía el servidor virtual VM1 (4a) del tipo de configuración 1 al tipo 2 ($VM_TYPE = type2$). La API basada en eventos de VIM genera un evento de uso de recursos con marca de tiempo = $t2$, que el agente de software del monitor de uso de la nube (4b) captura y registra en la base de datos de registro de eventos de uso de recursos. El consumidor de la nube apaga el servidor virtual (5). El VIM detiene el servidor virtual VM1 (6a) y su API basada en eventos genera un evento de uso de recursos con marca de tiempo = $t3$, que el agente de software del monitor de uso de la nube captura y registra en la base de datos log (6b). El portal de uso y administración accede a la base de datos log y calcula el uso total (U_{total}) para el servidor virtual VM1 (7).

5.5. Replicación de recursos

Definida como la creación de varias instancias del mismo recurso de TI, la replicación generalmente se realiza cuando es necesario mejorar la disponibilidad y el rendimiento de un recurso de TI. La tecnología de virtualización se utiliza para implementar el mecanismo de replicación de recursos que replica los recursos de TI basados en la nube (Figura 5.16).

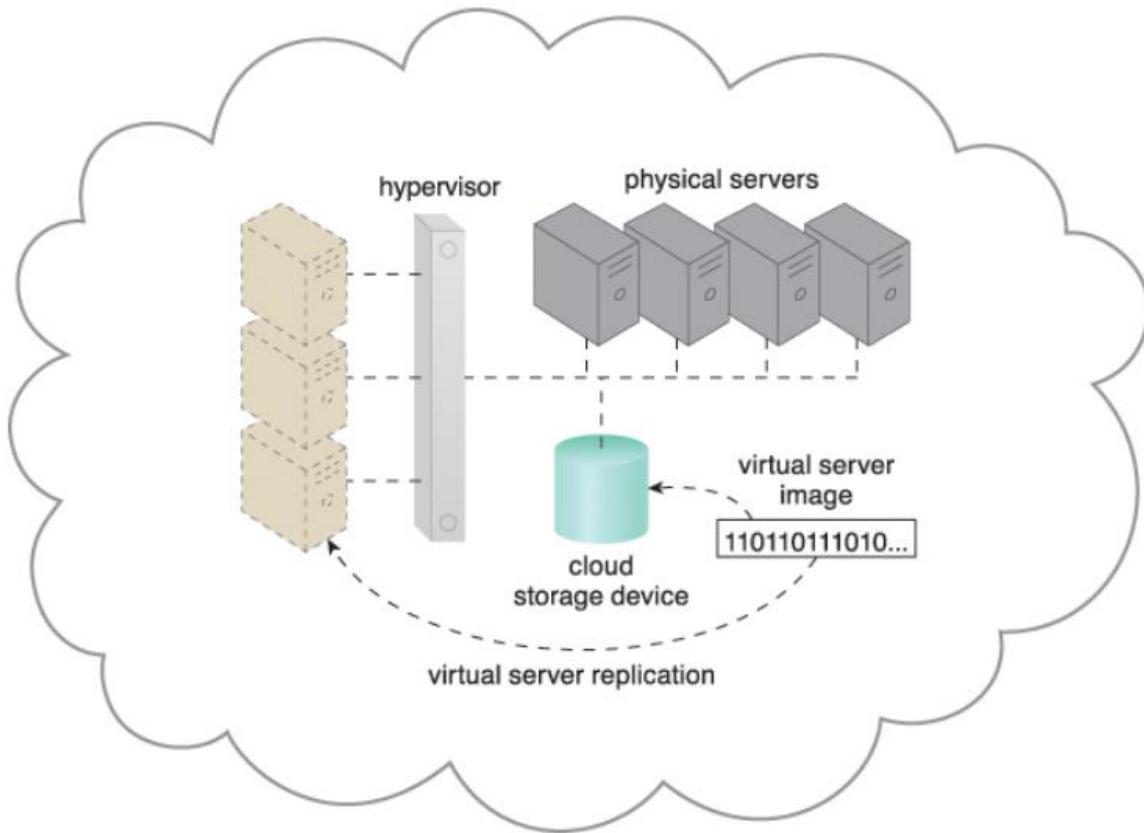


Figura 5.16 El hipervisor replica varias instancias de un servidor virtual, utilizando una imagen de servidor virtual almacenada.

Ejemplo de Estudio de Caso

DTGOV establece un conjunto de servidores virtuales de alta disponibilidad que se pueden reubicar automáticamente en servidores físicos que se ejecutan en diferentes centros de datos en respuesta a condiciones de falla severa. Esto se ilustra en el escenario representado en las Figuras 5.17 a 5.19, donde un servidor virtual que reside en un servidor físico que se ejecuta en un centro de datos experimenta una condición de falla. Los VIMs de diferentes centros de datos se coordinan para superar la falta de disponibilidad mediante la reasignación del servidor virtual a un servidor físico diferente que se ejecuta en otro centro de datos.

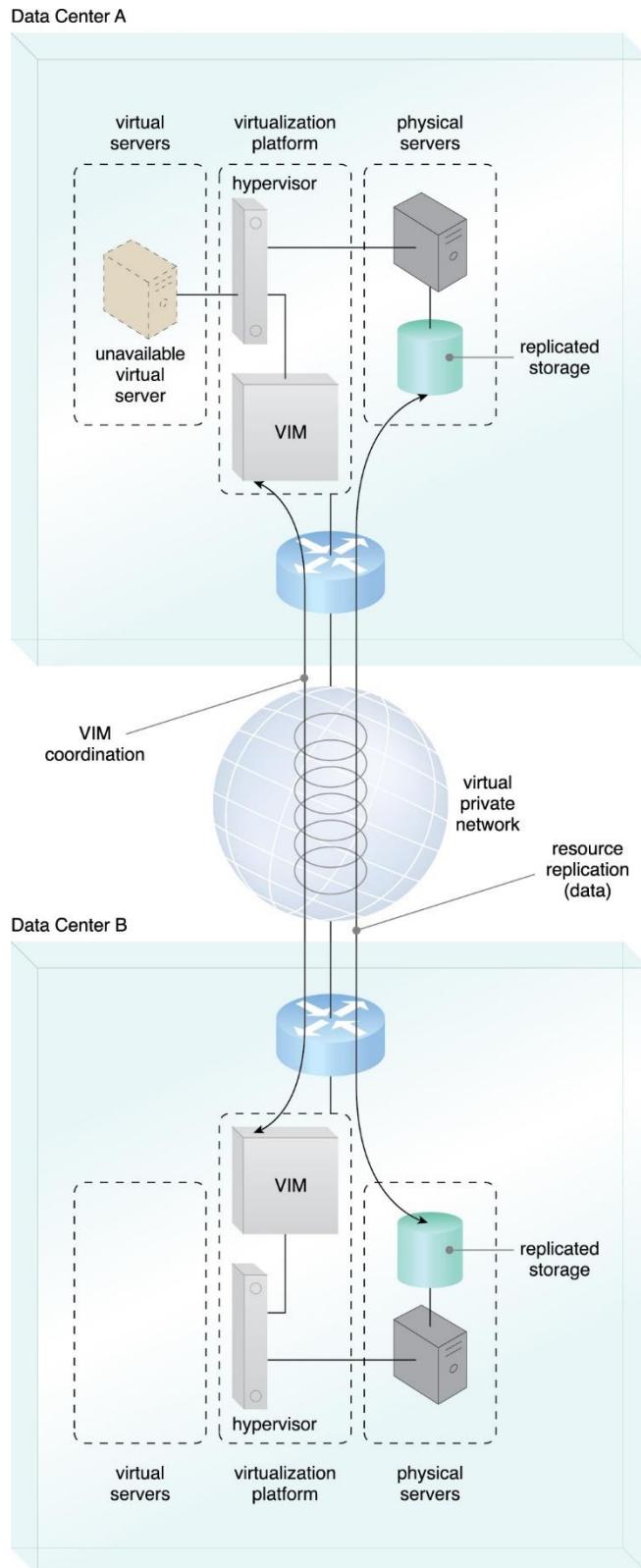


Figura 5.17 Un servidor virtual de alta disponibilidad se ejecuta en el Centro de datos A. Las instancias de VIM en los Centros de datos A y B ejecutan una función de coordinación que permite la detección de condiciones

de falla. Las imágenes de VM almacenadas se replican entre centros de datos como resultado de la arquitectura de alta disponibilidad.

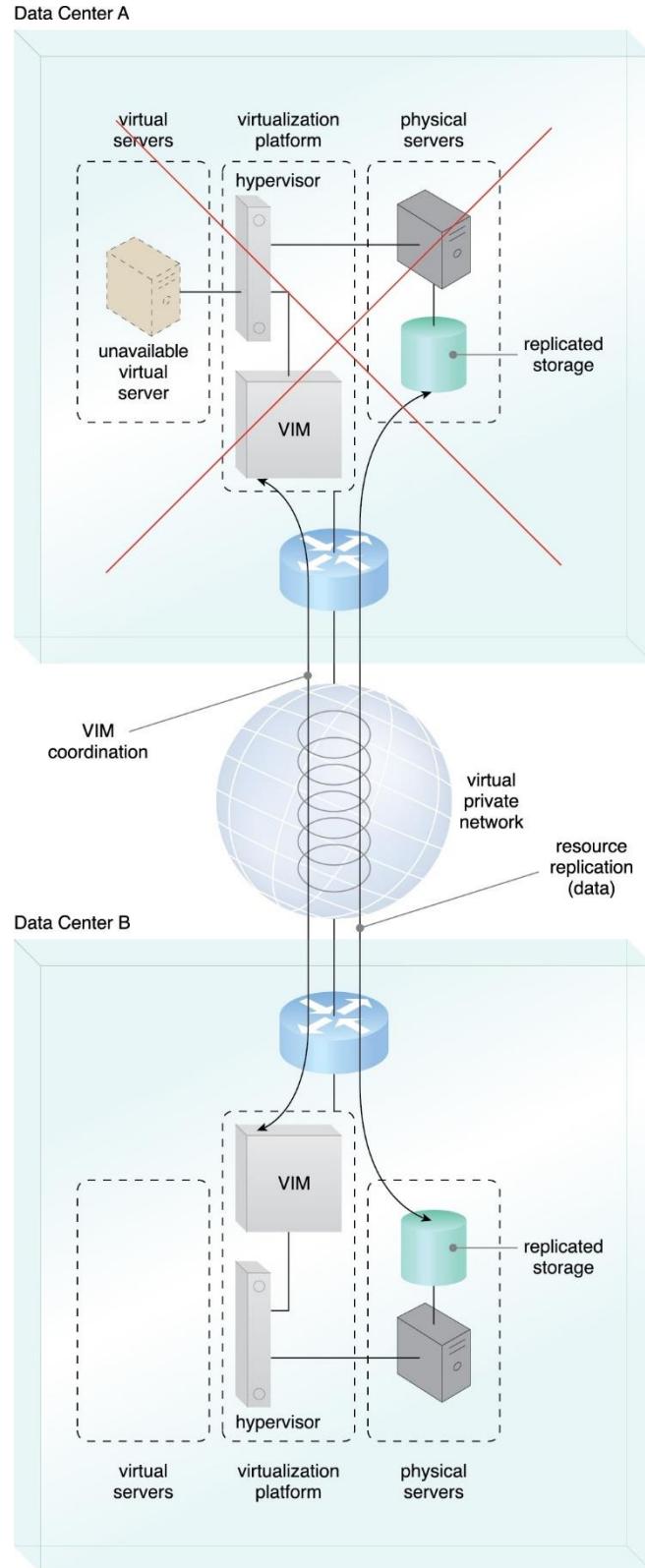


Figura 5.18 El servidor virtual deja de estar disponible en el Centro de datos A. El VIM en el Centro de datos B detecta la condición de falla y comienza a reasignar el servidor de alta disponibilidad del Centro de datos A al Centro de datos B.

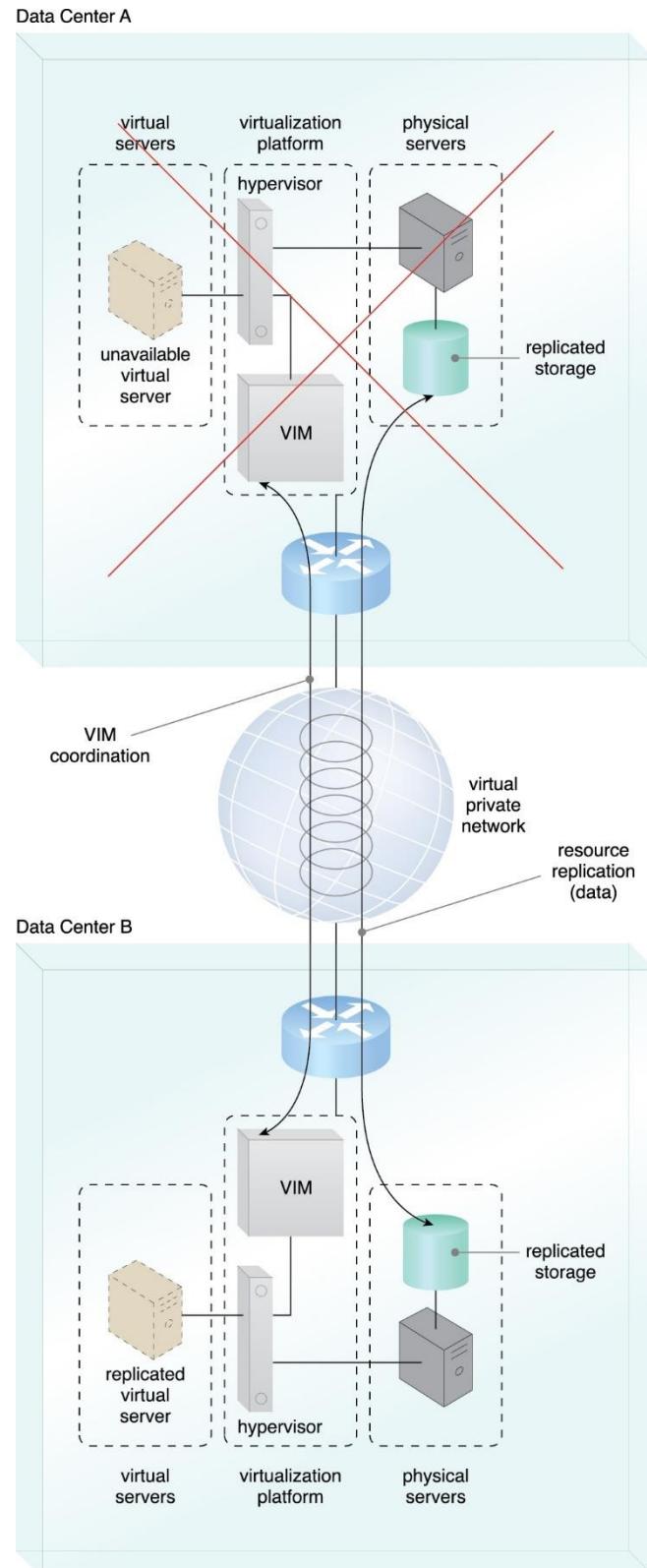


Figura 5.19 Se crea una nueva instancia del servidor virtual y está disponible en el Centro de datos B.

5.6. Entorno listo para usar

El mecanismo de entorno listo para usar (Figura 5.20) es un componente definitorio del modelo de entrega de la nube PaaS que representa una plataforma predefinida basada en la nube compuesta por un conjunto de recursos de TI ya instalados, listos para ser utilizados y personalizados por un consumidor de la nube. Los consumidores de la nube utilizan estos entornos para desarrollar e implementar de forma remota sus propios servicios y aplicaciones dentro de una nube. Los entornos típicos listos para usar incluyen recursos de TI preinstalados, como bases de datos, middleware, herramientas de desarrollo y herramientas de control.

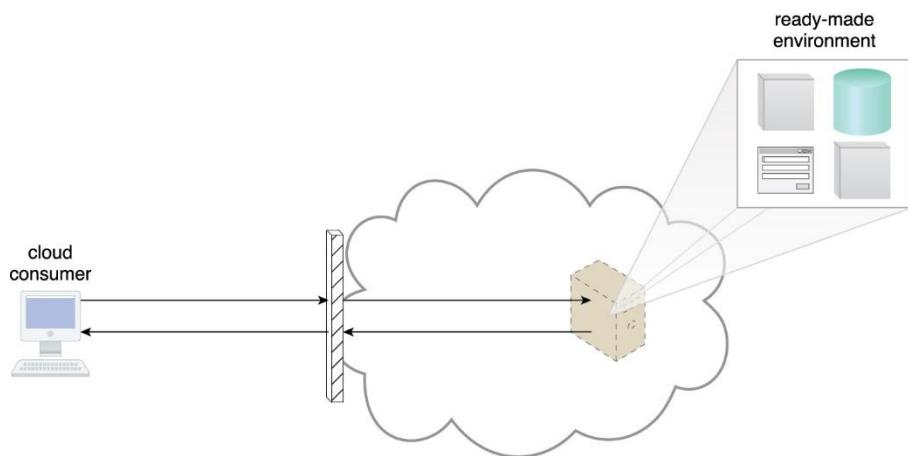


Figura 5.20 Un consumidor de la nube accede a un entorno listo para usar alojado en un servidor virtual.

Un entorno listo para usar generalmente está equipado con un kit de desarrollo de software (SDK) completo que brinda a los consumidores de la nube acceso programático a las tecnologías de desarrollo que comprenden sus pilas de programación preferidas.

El middleware está disponible para plataformas multitenant que permiten el desarrollo y la implementación de aplicaciones web. Algunos proveedores de nube ofrecen entornos para servicios en la nube que se basan en diferentes parámetros de rendimiento y facturación en tiempo de ejecución. Por ejemplo, una instancia de front-end de un servicio en la nube se puede configurar para responder a solicitudes sensibles al tiempo de manera más efectiva que una instancia de back-end. La primera variación se facturará a una tarifa diferente a la segunda.

Como se demuestra con más detalle en el próximo ejemplo de estudio de caso, una solución se puede dividir en grupos lógicos que se pueden designar para la invocación de instancias de frontend y backend a fin de optimizar la ejecución y la facturación en tiempo de ejecución.

Ejemplo de Estudio de Caso

ATN desarrolló e implementó varias aplicaciones comerciales no críticas utilizando un entorno PaaS alquilado. Una era una aplicación web de catálogo de números de pieza basada en Java que se usaba para los switches y routers que fabricaban. Esta aplicación es utilizada por diferentes fabricantes, pero no manipula los datos de las transacciones, los cuales son procesados por un sistema de control de stock separado.

La lógica de la aplicación se dividió en lógica de procesamiento de front-end y back-end. La lógica de front-end se utilizó para procesar consultas simples y actualizaciones del catálogo. La parte de back-end contiene la lógica requerida para representar el catálogo completo y correlacionar componentes similares y números de parte.

La Figura 5.21 ilustra el entorno de desarrollo e implementación de la aplicación del catálogo de números de piezas de ATN. Observe cómo el consumidor de la nube asume los roles de desarrollador y usuario final.

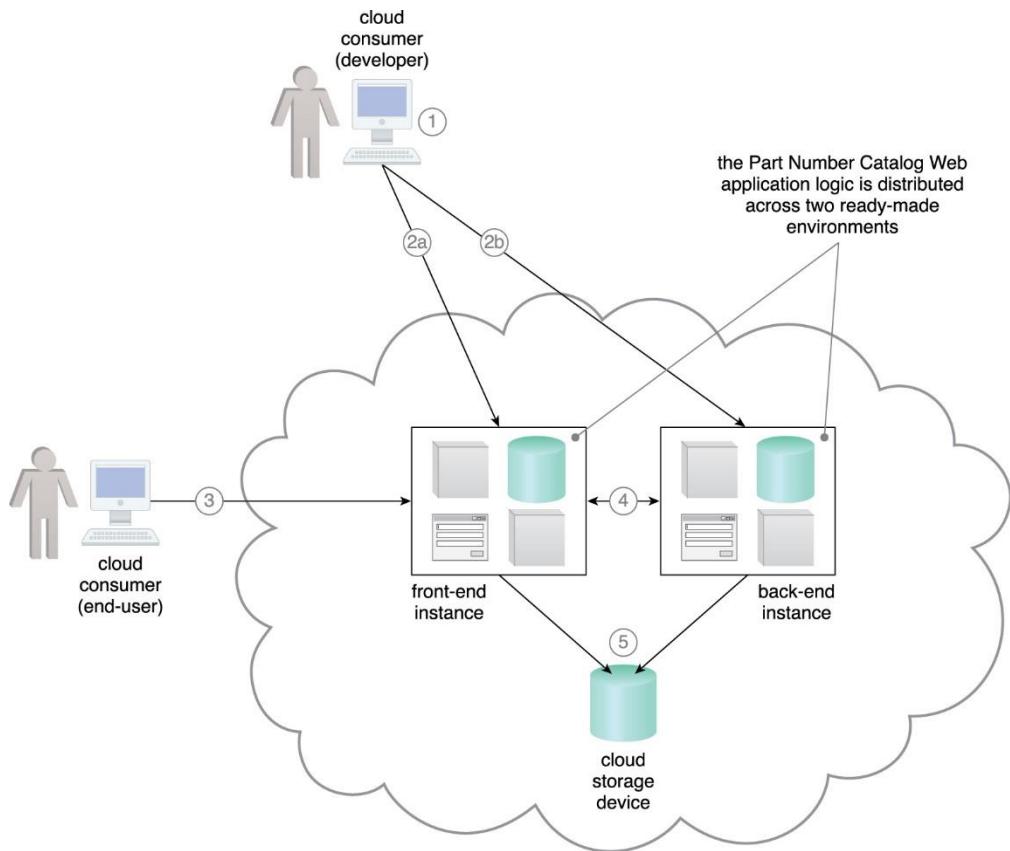


Figura 5.21 El desarrollador utiliza el SDK proporcionado para desarrollar la aplicación web del catálogo de números de pieza (1). El software de la aplicación se implementa en una plataforma web establecida por dos entornos listos para usar denominados instancia de front-end (2a) e instancia de back-end (2b). La aplicación está disponible para su uso y un usuario final accede a su instancia de front-end (3). El software que se ejecuta en la instancia de front-end invoca una tarea de larga duración en la instancia de back-end que corresponde al procesamiento requerido por el usuario final (4). El software de la aplicación implementado en las instancias de front-end y back-end está respaldado por un dispositivo de almacenamiento en la nube que proporciona almacenamiento persistente de los datos de la aplicación (5).

5.7 Contenedores

Los contenedores se explican en la sección *Containerization* en un capítulo anterior. Los contenedores pueden proporcionar un medio efectivo para implementar y entregar servicios en la nube. Un contenedor está representado por un símbolo similar al símbolo de límite organizacional presentado en el Capítulo 4, excepto que tiene esquinas redondeadas en lugar de esquinas afiladas (Figura 5.22).



Figura 5.22 El símbolo utilizado para representar un contenedor.

6 Mecanismos especializados de la nube



Una arquitectura de tecnología de nube típica contiene numerosos componentes para abordar distintos requerimientos de recursos y soluciones de TI. Cada mecanismo cubierto en este capítulo cumple una función específica en apoyo de una o más características de la nube.

En este capítulo se describen los siguientes mecanismos especializados en la nube:

- Automated Scaling Listener
- Load Balancer
- Monitor de SLA
- Monitor de pago por uso
- Audit monitor
- Failover System
- Hypervisor
- Clúster de recursos
- Multi-device Broker
- State Management Database

Todos estos mecanismos se pueden considerar extensiones de la infraestructura de la nube y se pueden combinar de muchas maneras como parte de arquitecturas tecnológicas distintas y personalizables.

[6.1. Automated Scaling Listener](#)

El mecanismo de automated scaling listener (escucha de escalamiento automatizado) es un agente de servicio que supervisa y rastrea las comunicaciones entre los consumidores de servicios en la nube y los servicios en la nube con fines de escalamiento dinámico. Los oyentes de escalamiento automatizado se implementan dentro de la nube, generalmente cerca del firewall, desde donde rastrean automáticamente la información del estado de la carga de trabajo. Las cargas de trabajo se pueden determinar por el volumen de las solicitudes generadas por los consumidores en la nube o mediante las demandas de procesamiento de back-end desencadenadas por ciertos tipos de solicitudes. Por ejemplo, una pequeña cantidad de datos entrantes puede resultar en una gran cantidad de procesamiento.

Los escuchas de escalamiento automatizado pueden proporcionar diferentes tipos de respuestas a las condiciones de fluctuación de la carga de trabajo, como:

- Escalar automáticamente los recursos de TI según los parámetros definidos previamente por el consumidor de la nube (comúnmente conocido como *auto-scaling*).
- Notificación automática al consumidor de la nube cuando las cargas de trabajo superan los umbrales actuales o caen por debajo de los recursos asignados (Figura 6.1). De esta forma, el consumidor de la nube puede optar por ajustar su asignación actual de recursos de TI.

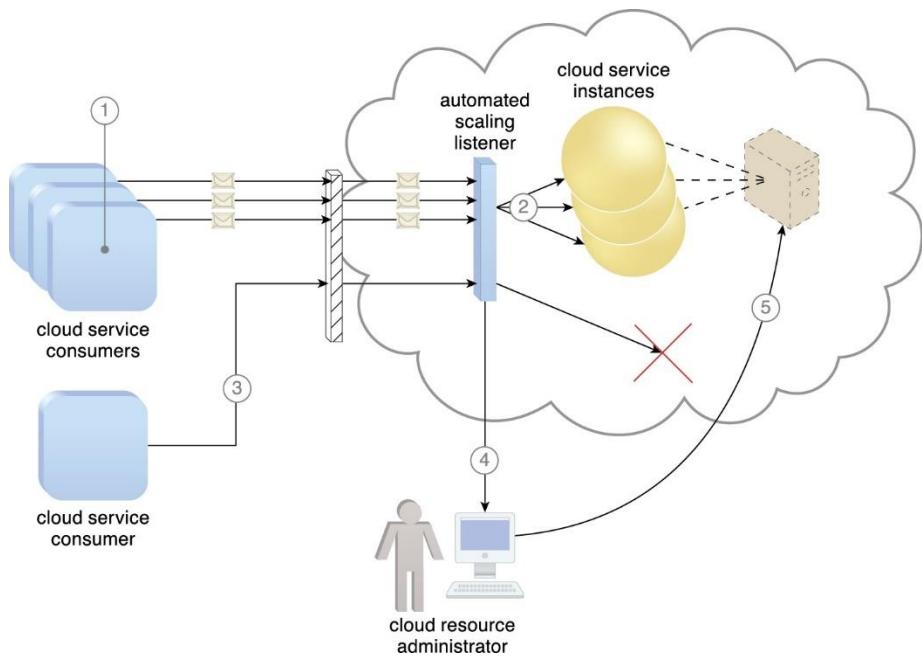


Figura 6.1 Tres consumidores de servicios en la nube intentan acceder a un servicio en la nube simultáneamente (1). El agente de escucha de escalamiento automatizado, escala horizontalmente e inicia la creación de tres instancias redundantes del servicio (2). Un cuarto consumidor de servicios en la nube intenta utilizar el servicio en la nube (3). Programado para permitir hasta solo tres Instancias del servicio en la nube, el escucha de escalado automatizado rechaza el cuarto intento y notifica al consumidor de la nube que se excedió el límite de carga de trabajo solicitado (4). El administrador de recursos de la nube del consumidor de la nube accede al entorno de administración remota para ajustar la configuración de aprovisionamiento y aumentar el límite de instancias redundantes (5).

Los diferentes proveedores de servicios en la nube tienen diferentes nombres para los agentes de servicio que actúan como escuchas de escalamiento automatizado.

Ejemplo de Estudio de Caso

Los servidores físicos de DTGOV escalan verticalmente las instancias de servidores virtuales, desde la configuración de máquina virtual más pequeña (1 núcleo de procesador virtual, 4 GB de RAM virtual) hasta la más grande (128 núcleos de procesador virtual, 512 GB de RAM virtual). La plataforma de virtualización está configurada para escalar automáticamente un servidor virtual en tiempo de ejecución, de la siguiente manera:

- Scaling-Down - El servidor virtual continúa residiendo en el mismo servidor de host físico mientras se reduce a una configuración de menor rendimiento.
- Scaling-Up - La capacidad del servidor virtual se duplica en su servidor host físico original. El VIM también puede migrar el servidor virtual a otro servidor físico si el servidor host original está sobrecargado. La migración se realiza automáticamente en tiempo de ejecución y no requiere que el servidor virtual se apague.

La configuración de escalado automático controlada por los consumidores de la nube determina el comportamiento en tiempo de ejecución de los agentes de escucha de escalamiento automático, que se ejecutan en el hipervisor que supervisa el uso de recursos de los servidores virtuales. Por

ejemplo, un consumidor de la nube lo ha configurado para que cada vez que el uso de recursos supere el 80% de la capacidad de un servidor virtual durante 60 segundos consecutivos, el oyente de escalamiento automatizado activa el proceso de escalado enviando a la plataforma VIM un comando de escalado. Por el contrario, el oyente de escalamiento automatizado también ordena al VIM que reduzca la escala cada vez que el uso de recursos cae un 15 % por debajo de la capacidad durante 60 segundos consecutivos (Figura 6.2).

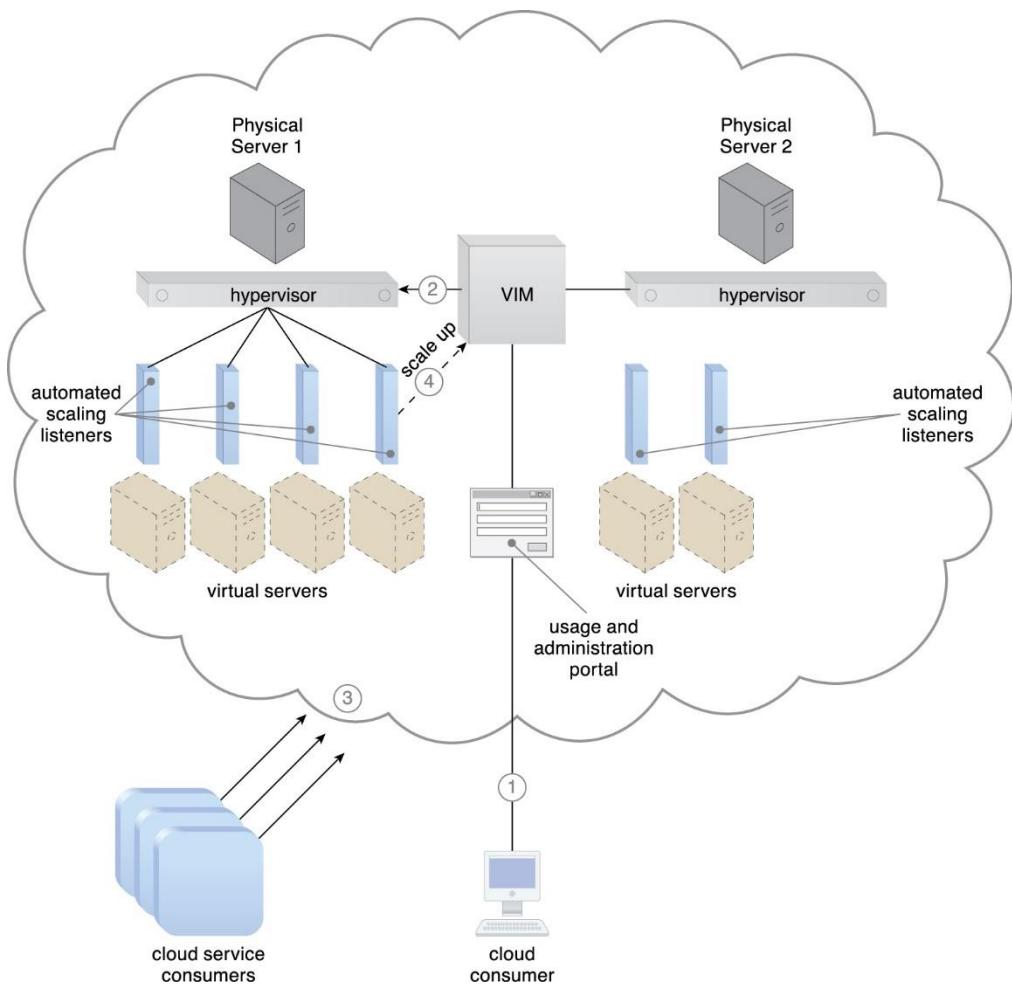


Figura 6.2 Un consumidor de la nube crea e inicia un servidor virtual con 8 núcleos de procesadores virtuales y 16 GB de RAM virtual (1). El VIM crea el servidor virtual a pedido del consumidor del servicio en la nube y lo asigna al Servidor Físico 1 para unirse a otros 3 servidores virtuales activos (2). La demanda de los consumidores de la nube hace que el uso del servidor virtual aumente en más del 80 % de la capacidad de la CPU durante 60 segundos consecutivos (3). El oyente de escalado automatizado que se ejecuta en el hipervisor detecta la necesidad de escalar y ordena al VIM como consecuencia (4).

La Figura 6.3 ilustra la migración en vivo de una máquina virtual, tal como la realiza el VIM.

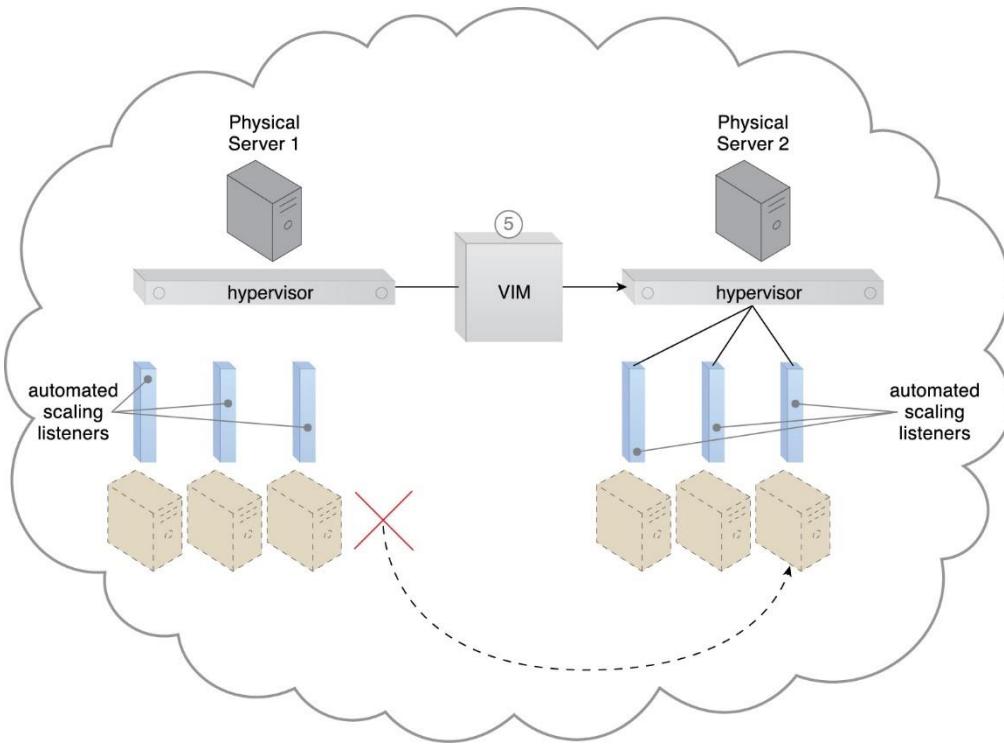


Figura 6.3 El VIM determina que no es posible escalar el servidor virtual en el Servidor físico 1 y procede a migrarlo en vivo al Servidor físico 2.

La reducción de escala del servidor virtual por parte del VIM se muestra en la Figura 6.4.

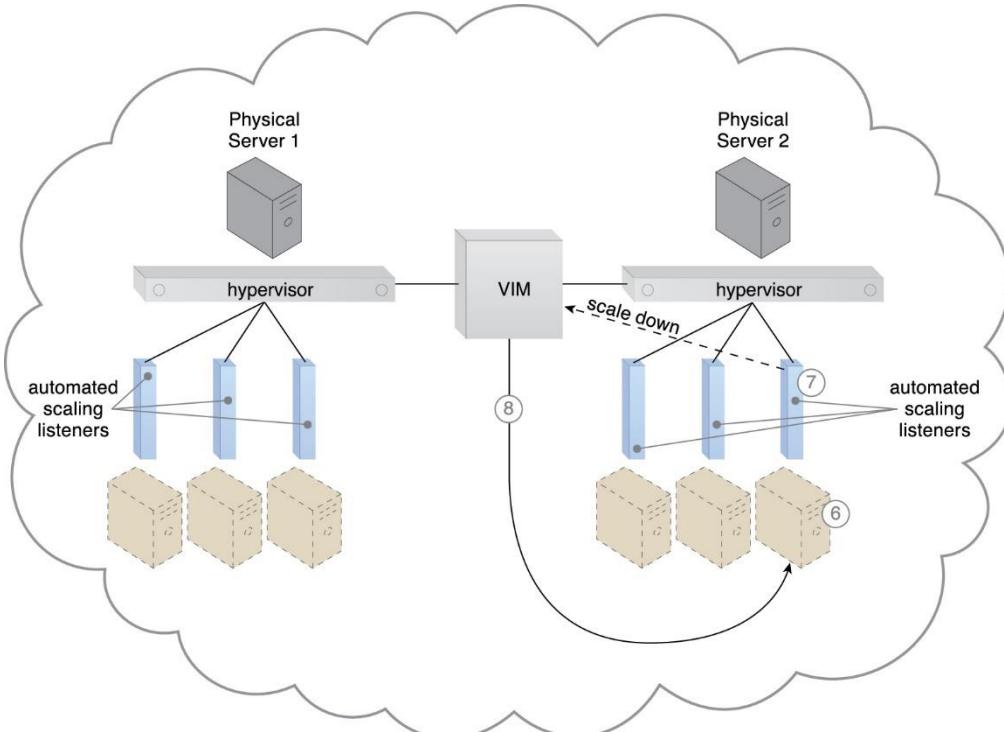


Figura 6.4 El uso de CPU/RAM del servidor virtual permanece por debajo del 15 % de su capacidad durante 60 segundos consecutivos (6). El automated scaling listener detecta la necesidad de reducir y ordena al VIM (7), que reduce el servidor virtual (8) mientras permanece activo en el servidor físico 2.

6.2. Load Balancer

Un enfoque común para el escalado horizontal es equilibrar una carga de trabajo entre dos o más recursos de TI para aumentar el rendimiento y la capacidad más allá de lo que puede proporcionar un solo recurso de TI. El mecanismo del load balancer (balanceador de carga) es un agente de tiempo de ejecución cuya lógica se basa fundamentalmente en esta premisa.

Más allá de los simples algoritmos de división del trabajo (Figura 6.5), los balanceadores de carga pueden realizar una variedad de funciones de distribución de carga de trabajo en tiempo de ejecución especializadas que incluyen:

- Distribución asimétrica - las cargas de trabajo más grandes se envían a los recursos de TI con capacidades de procesamiento más altas.
- Priorización de la carga de trabajo - las cargas de trabajo se calendarizan, se encolan, se descargan y distribuyen de acuerdo con sus niveles de prioridad.
- Distribución Content-Aware - las solicitudes se distribuyen a diferentes recursos de TI según lo dicte el contenido de la solicitud.

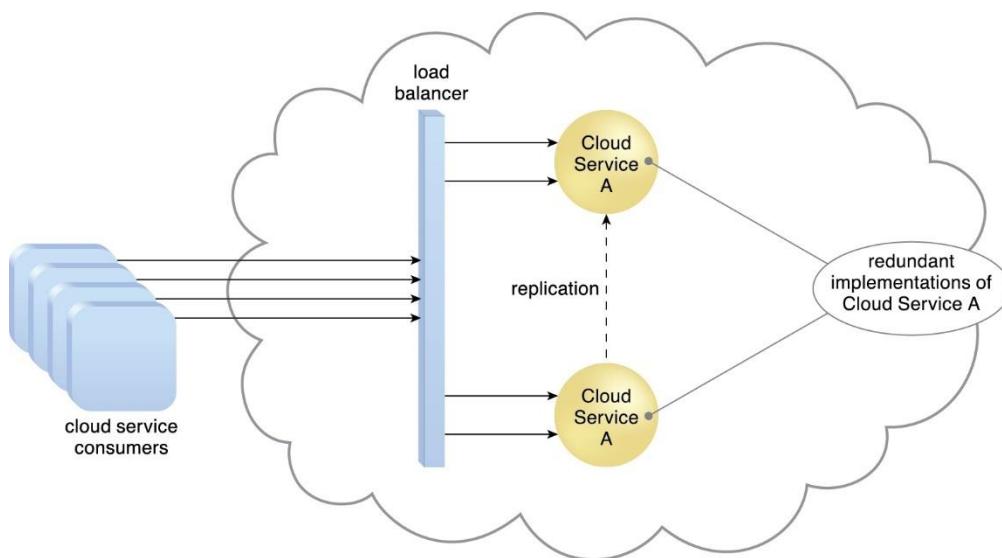


Figura 6.5 Un load balancer implementado como agente de servicio distribuye de manera transparente los mensajes de solicitud de carga de trabajo entrantes entre dos implementaciones redundantes de servicios en la nube, lo que a su vez maximiza el rendimiento para los consumidores del servicio en la nube.

Un load balancer se programa o configura con un conjunto de reglas y parámetros de rendimiento y QoS con los objetivos generales de optimizar el uso de recursos de TI, evitar sobrecargas y maximizar el rendimiento.

Los mecanismos del balanceador de carga pueden existir como:

- comutador de red multicapa

- dispositivo de hardware dedicado
- sistema basado en software dedicado (común en sistemas operativos de servidor)
- agente de servicio (generalmente controlado por software de administración de la nube)

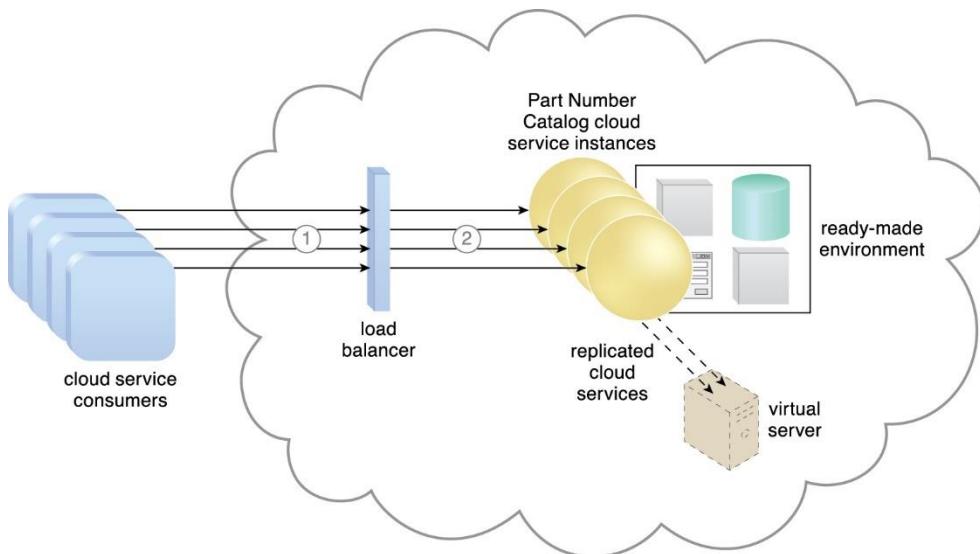
El balanceador de carga generalmente se encuentra en la ruta de comunicación entre los recursos de TI que generan la carga de trabajo y los recursos de TI que realizan el procesamiento de la carga de trabajo. Este mecanismo puede diseñarse como un agente transparente que permanece oculto para los consumidores del servicio en la nube o como un componente proxy que abstrae los recursos de TI que realizan su carga de trabajo.

Ejemplo de Estudio de Caso

El servicio en la nube del catálogo de números de piezas de ATN no manipula los datos de las transacciones, aunque lo utilicen varias empresas en diferentes regiones. Tiene períodos de uso máximo durante los primeros días de cada mes que coinciden con el procesamiento preparatorio de las pesadas rutinas de control de existencias en las fábricas. ATN siguió las recomendaciones de su proveedor de la nube y actualizó el servicio de la nube para que sea altamente escalable a fin de admitir las fluctuaciones de carga de trabajo anticipadas.

Después de desarrollar las actualizaciones necesarias, ATN decide probar la escalabilidad mediante el uso de una herramienta de prueba de automatización de bots³⁰ que simula grandes cargas de trabajo. Las pruebas deben determinar si la aplicación puede escalar sin problemas para atender cargas de trabajo máximas que son 1000 veces mayores que sus cargas de trabajo promedio. Los bots proceden a simular cargas de trabajo que duran 10 minutos.

La funcionalidad de escalado automático resultante de la aplicación se muestra en la Figura 6.6.



³⁰ Un bot es un programa informático que efectúa automáticamente tareas reiterativas mediante Internet a través de una cadena de comandos o funciones autónomas previas para asignar un rol establecido; y que posee capacidad de interacción, cambiando de estado para responder a un estímulo. Fuente: Wikipedia.

Figura 6.6 Las nuevas instancias de los servicios en la nube se crean automáticamente para satisfacer las crecientes solicitudes de uso. El load balancer utiliza la asignación round-robin para garantizar que el tráfico se distribuya uniformemente entre los servicios de nube activos.

6.3. Monitor de SLA

El mecanismo de monitoreo de SLA se usa para observar específicamente el rendimiento en tiempo de ejecución de los servicios en la nube para garantizar que cumplan con los requisitos de QoS contractuales que se publican en SLAs (Figura 6.7). Los datos recopilados por el monitor de SLA son procesados por un sistema de gestión de SLA para agregarse a las métricas de informes de SLA. El sistema puede reparar o comutar por error de forma proactiva los servicios en la nube cuando se producen condiciones excepcionales, como cuando el monitor SLA informa que un servicio en la nube se "cayo".

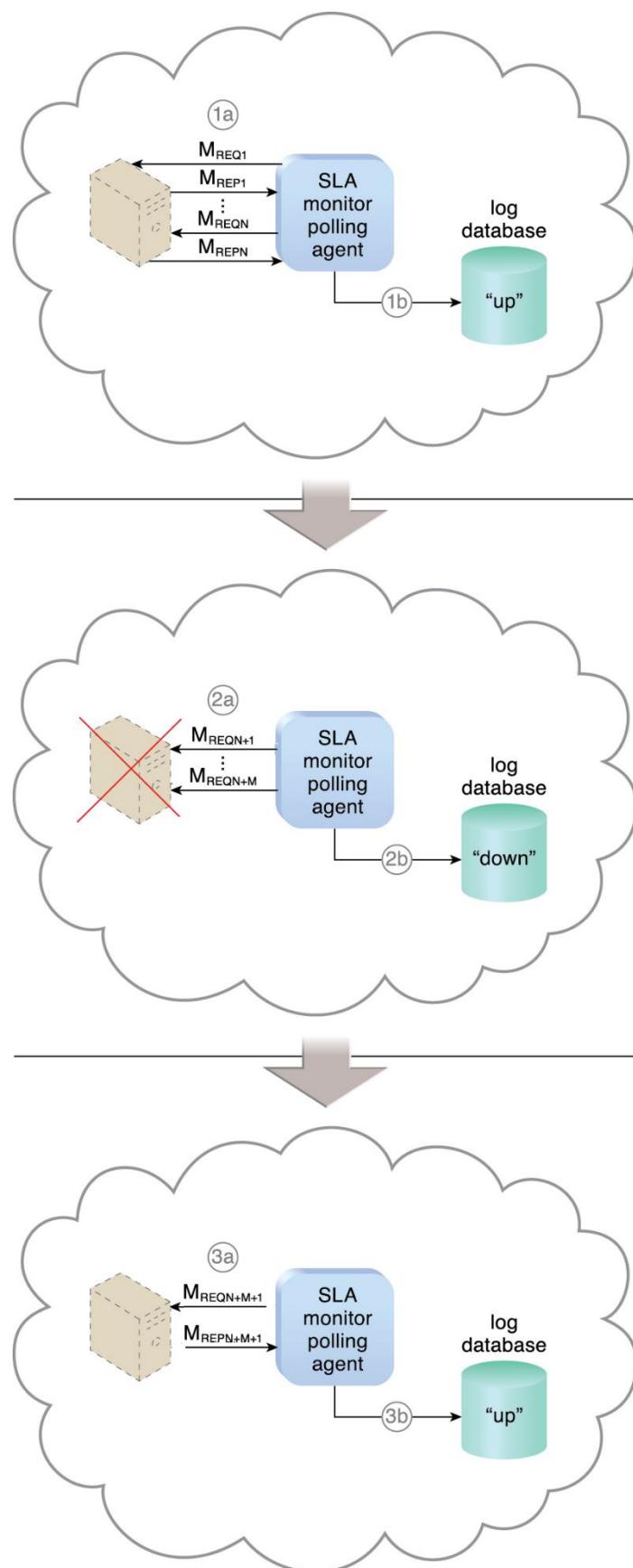


Figura 6.7 El monitor SLA sondea el servicio en la nube mediante el envío de mensajes de solicitud de sondeo (M_{REQ1} a M_{REQN}). El monitor recibe mensajes sondeo de respuesta (M_{REP1} a M_{REPN}) que informan que el servicio estaba activo en cada ciclo de sondeo (1a). El monitor SLA almacena el tiempo de actividad (período de tiempo de todos los ciclos de sondeo del 1 al N) en la base de datos log (1b). El monitor SLA sondea el servicio en la nube que envía mensajes de solicitud de sondeo (M_{REQN+1} a M_{REQN+M}). No se reciben mensajes de sondeo de respuesta (2a). Los mensajes de respuesta siguen agotando el tiempo de espera, por lo que el monitor SLA almacena el tiempo de "inactividad" (período de tiempo de todos los ciclos de sondeo N+1 a N+M) en la base de datos de log (2b). El monitor SLA envía un mensaje de solicitud de sondeo ($M_{REPN+M+1}$) y recibe el mensaje de respuesta de sondeo ($M_{REPN+M+1}$) (3a). El monitor SLA almacena el tiempo de actividad en la base de datos log (3b).

Ejemplo de Estudio de Caso

El SLA estándar para servidores virtuales en los contratos de arrendamiento de DTGOV define una disponibilidad mínima de recursos de TI del 99,95 %, que se rastrea utilizando dos monitores de SLA: uno basado en un agente de polling y el otro basado en la implementación de un agente de monitoreo regular.

SLA Monitor Polling Agent

El monitor de polling de SLA en DTGOV se ejecuta en la red perimetral externa para detectar los tiempos de espera del servidor físico. Es capaz de identificar fallas de red, hardware y software del centro de datos (con granularidad de minutos) que resultan en la falta de respuesta del servidor físico. Se requieren tres tiempos de espera consecutivos de períodos de sondeo de 20 segundos para declarar que el recurso de TI no está disponible.

Se generan tres tipos de eventos:

- PS_timeout - el sondeo del servidor físico ha agotado el tiempo de espera
- PS_Unreachable - el sondeo del servidor físico ha expirado consecutivamente tres veces
- PS_Reachable - el servidor físico que antes no estaba disponible vuelve a responder al sondeo

SLA Monitoring Agent

La API basada en eventos de VIM implementa el monitor SLA como un agente de monitoreo para generar los siguientes tres eventos:

- VM_Unreachable - el VIM no puede comunicarse con la VM
- VM_Failure - la VM falló y no está disponible
- VM_Reachable - la VM es accesible

Los eventos generados por el agente de polling tienen marcas de tiempo que se registran en una base de datos log de eventos de SLA y que el sistema de gestión de SLA utiliza para calcular la disponibilidad de recursos de TI. Se utilizan reglas complejas para correlacionar eventos de diferentes monitores de sondeo SLA y los servidores virtuales afectados, y para descartar cualquier falso positivo durante períodos de indisponibilidad.

Las Figuras 6.8 y 6.9 muestran los pasos que toman los monitores SLA durante una falla y recuperación de la red del centro de datos.

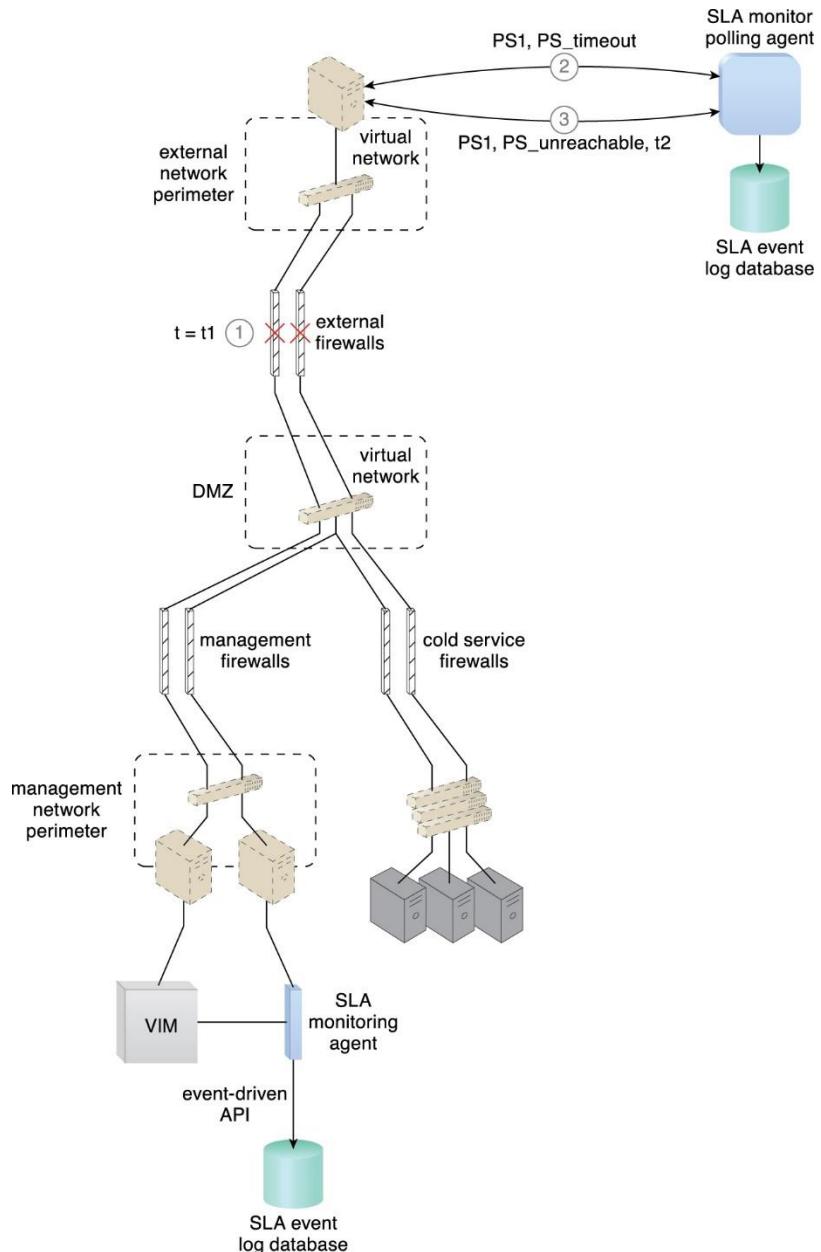


Figura 6.8 En timestamp = t_1 , un clúster de firewalls ha fallado y todos los recursos de TI en el centro de datos dejan de estar disponibles (1). El agente de SLA monitor polling deja de recibir respuestas de los servidores físicos y comienza a emitir eventos PS_timeout (2). El agente de SLA monitor polling comienza a emitir eventos PS_unreachable después de tres eventos PS_timeout sucesivos. La marca de tiempo ahora es t_2 (3).

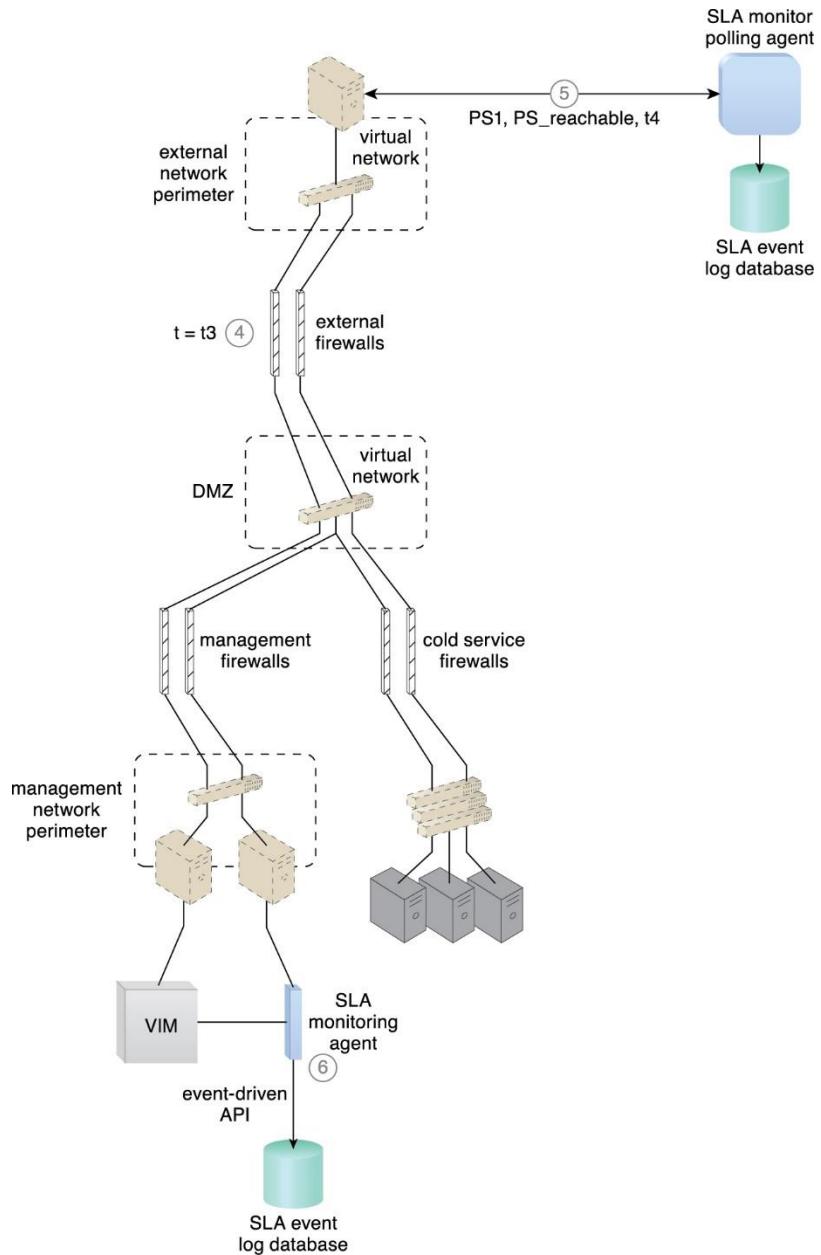


Figura 6.9 El recurso de TI se vuelve operativo en la marca de tiempo = t3 (4). El agente de sondeo del monitor SLA recibe respuestas de los servidores físicos y emite eventos PS_reachable. La marca de tiempo ahora es t4 (5). El agente de monitoreo SLA no detectó ninguna indisponibilidad ya que la comunicación entre la plataforma VIM y los servidores físicos no se vio afectada por la falla (6).

El sistema de gestión de SLA utiliza la información almacenada en la base de datos log para calcular el período de indisponibilidad como t4 - t3, que afectó a todos los servidores virtuales del centro de datos.

Las Figuras 6.10 y 6.11 ilustran los pasos que toman los monitores SLA durante la falla y la subsiguiente recuperación de un servidor físico que aloja tres servidores virtuales (VM1, VM2, VM3).

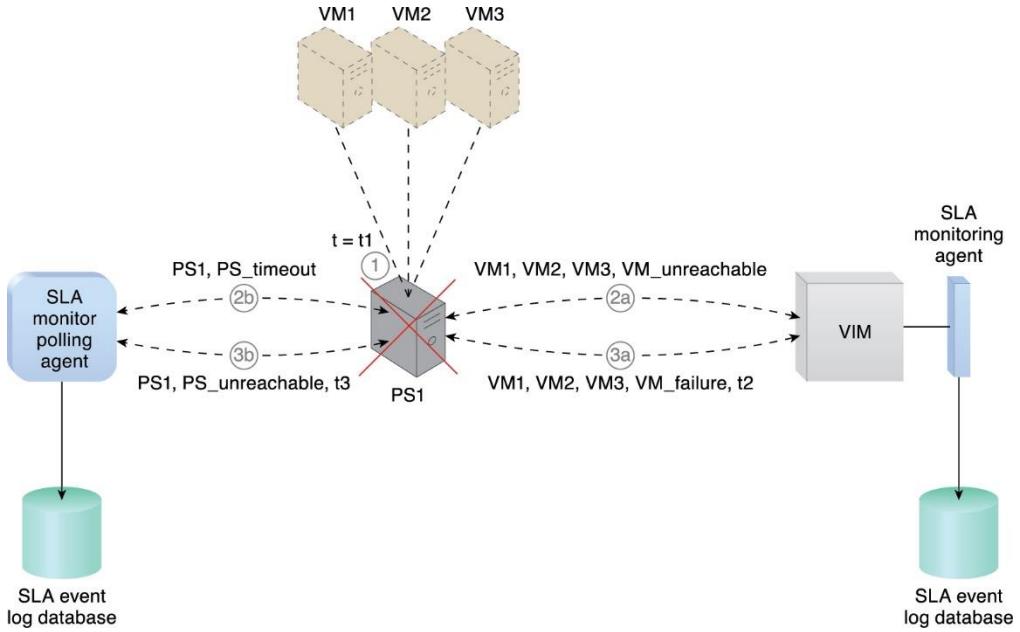


Figura 6.10 En la marca de tiempo = $t1$, el servidor host físico ha fallado y deja de estar disponible (1). El agente de supervisión de SLA captura un evento **VM_unreachable** que se genera para cada servidor virtual en el servidor host fallido (2a). El agente de sondeo del monitor SLA deja de recibir respuestas del servidor host y emite eventos **PS_timeout** (2b). En timestamp = $t2$, el agente de supervisión de SLA captura un evento **VM_failure** que se genera para cada uno de los tres servidores virtuales de los servidores host fallidos (3a). El agente de sondeo del monitor SLA comienza a emitir eventos **PS_unreachable** después de tres eventos **PS_timeout** sucesivos en la marca de tiempo = $t3$.

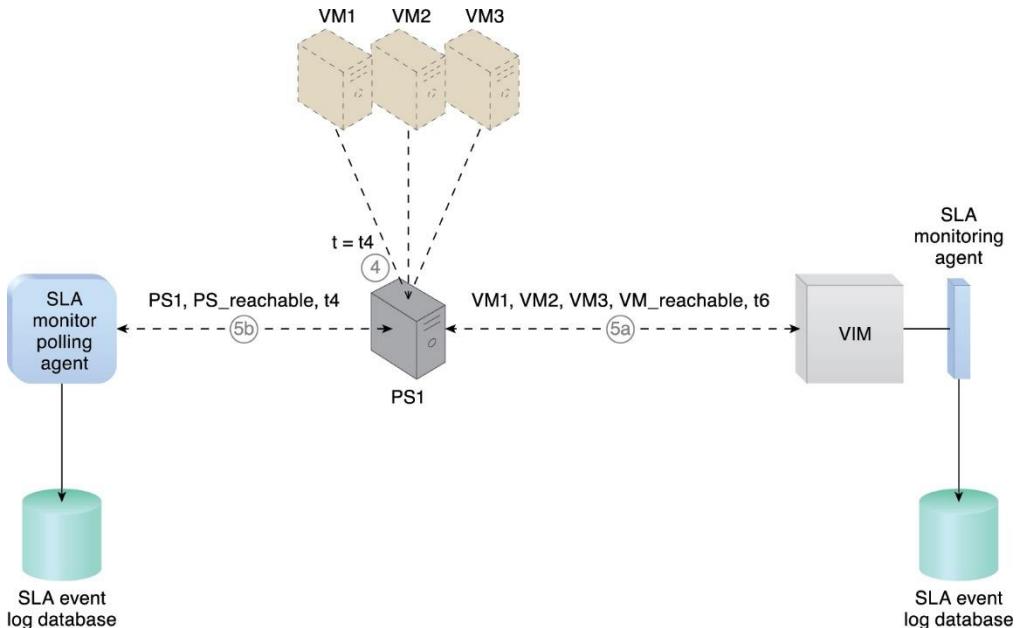


Figura 6.11 El servidor host se vuelve operativo en timestamp = $t4$ (4). El agente de sondeo del monitor SLA recibe respuestas del servidor físico y emite eventos del tipo **PS_reachable** en la marca de tiempo = $t5$ (5a). En timestamp = $t6$, el agente de supervisión de SLA captura un evento **VM_reachable** que se genera para cada servidor virtual (5b). El sistema de gestión de SLA calcula el período de indisponibilidad que afectó a todos los servidores virtuales como $t6 - t2$.

6.4. Monitor de pago por uso

El mecanismo de monitoreo *pay-per-use* mide el uso de recursos de TI basados en la nube de acuerdo con parámetros de precios predefinidos y genera registros de uso para calcular tarifas con propósitos de facturación.

Algunas variables típicas de monitoreo son:

- cantidad de mensajes de solicitud/respuesta
- volumen de datos transmitidos
- consumo de ancho de banda

Los datos recopilados por el monitor de pago por uso son procesados por un sistema de administración de facturación que calcula las tarifas de pago. La figura 6.12 muestra un monitor de pago por uso implementado como un agente de recursos que se utiliza para determinar el período de uso de un servidor virtual.

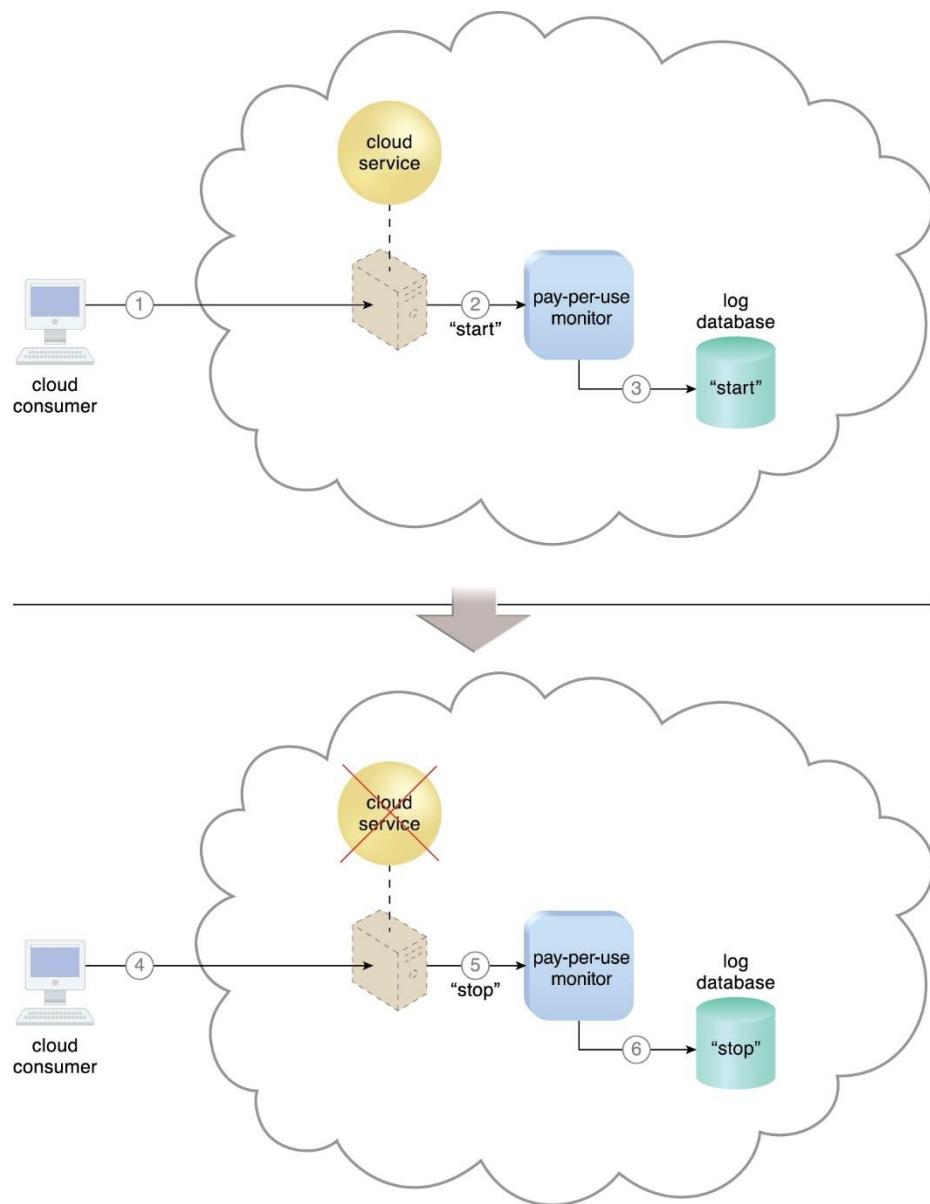


Figura 6.12 Un consumidor de la nube solicita la creación de una nueva instancia de un servicio en la nube (1). Se crea una instancia del recurso de TI y el monitor de pago por uso recibe una notificación de evento de "start" del software de recursos (2). El monitor de pago por uso almacena la marca de tiempo del valor en la base de datos log (3). El consumidor de la nube luego solicita que se detenga la instancia del servicio en la nube (4). El monitor de pago por uso recibe una notificación de evento de "detención" del software de recursos (5) y almacena la marca de tiempo del valor en la base de datos log (6).

La Figura 6.13 ilustra un monitor de pago por uso diseñado como un agente de monitoreo que intercepta y analiza de manera transparente la comunicación en tiempo de ejecución con un servicio en la nube.

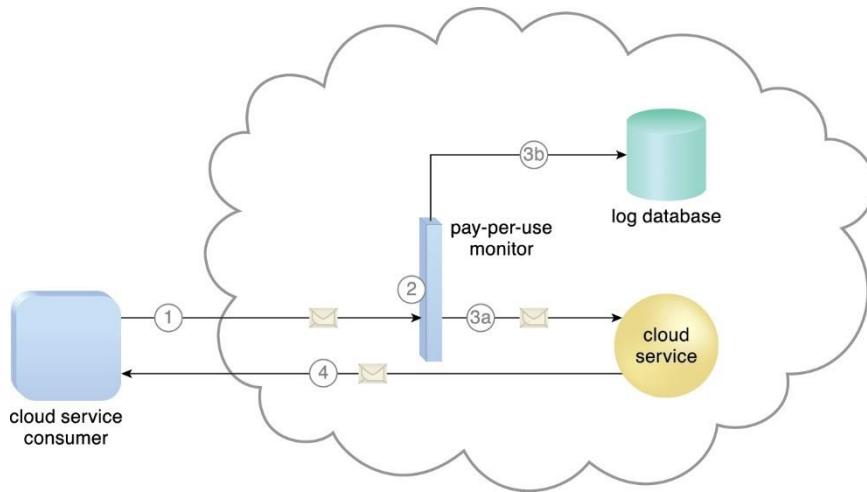


Figura 6.13 Un consumidor de servicios en la nube envía un mensaje de solicitud al servicio en la nube (1). El monitor de pago por uso intercepta el mensaje (2), lo reenvía al servicio en la nube (3a) y almacena la información de uso de acuerdo con sus métricas de monitoreo (3b). El servicio en la nube reenvía los mensajes de respuesta al consumidor del servicio en la nube para proporcionar el servicio solicitado (4).

Ejemplo de Estudio de Caso

DTGOV decide invertir en un sistema comercial capaz de generar facturas basadas en eventos predefinidos como "facturables" y modelos de precios personalizables. La instalación del sistema da como resultado dos bases de datos propietarias: la base de datos de eventos de facturación y la base de datos del esquema de precios personalizables.

Los eventos en tiempo de ejecución se recopilan a través de monitores de uso de la nube que se implementan como extensiones de la plataforma VIM utilizando la API de VIM. El agente de monitor por sondeo de pago por uso proporciona periódicamente al sistema de facturación información de eventos facturables. Un agente de monitoreo independiente proporciona más datos complementarios de facturación como:

- *Tipo de suscripción del consumidor de la nube* - Esta información se usa para identificar el tipo de modelo de precios para los cálculos de tarifas de uso, incluida la suscripción prepago con cuota de uso, la suscripción pospago con cuota máxima de uso y la suscripción pospago con uso ilimitado.
- *Categoría de uso de recursos* - El sistema de administración de facturación utiliza esta información para identificar el rango de tarifas de uso que se aplican a cada evento de uso. Los ejemplos incluyen el uso normal, el uso de recursos de TI reservados y el uso de servicios premium (administrados).
- *Cuotas de Consumo por uso de recursos* - Cuando los contratos de uso definen cuotas de uso de recursos de TI, las condiciones de los eventos de uso generalmente se complementan con las cuotas de consumo y los límites de cuota actualizados.

La Figura 6.14 ilustra los pasos que toma el monitor de pago por uso de DTGOV durante un evento de uso típico.

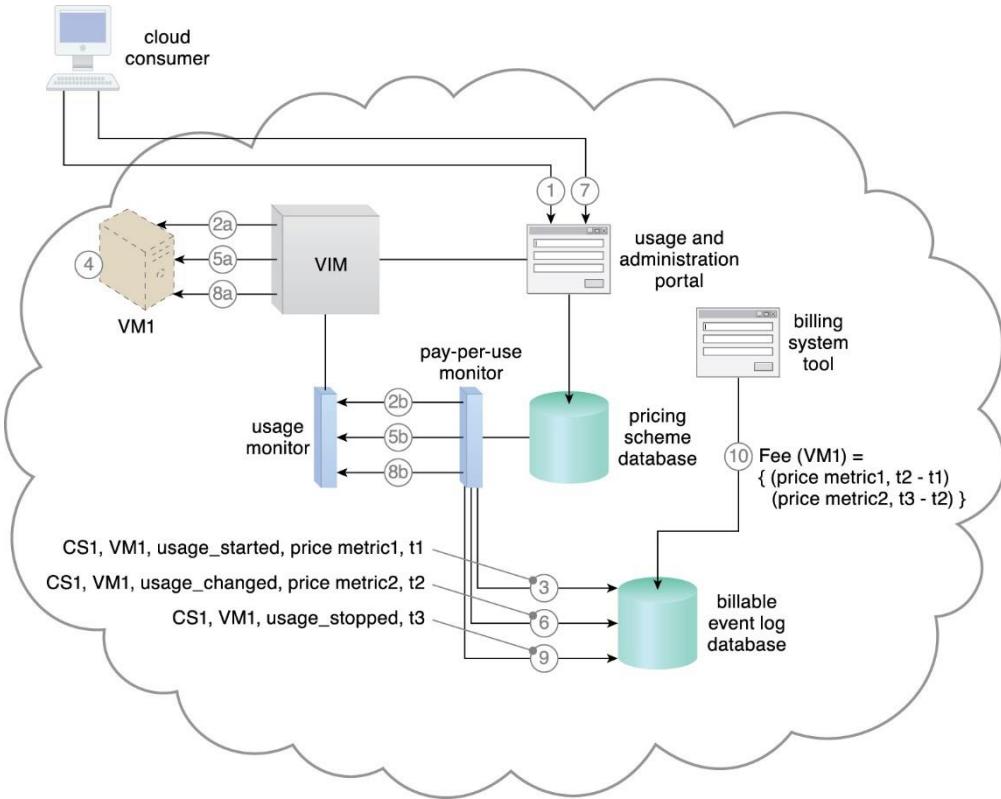


Figura 6.14 El consumidor de la nube ($CS_ID = CS1$) crea e inicia un servidor virtual ($VM_ID = VM1$) de tamaño de configuración tipo 1 ($VM_TYPE = type1$ (1)). El VIM crea la instancia del servidor virtual según lo solicitado (2a). La API basada en eventos de VIM genera un evento de uso de recursos con marca de tiempo = $t1$, que el monitor de uso de la nube captura y reenvía al monitor de pago por uso (2b). El monitor de pago por uso interactúa con la base de datos del plan de precios para identificar las métricas de contracargo y uso que se aplican al uso de recursos. Se genera un evento facturable de "uso iniciado" y se almacena en la base de datos log de eventos facturables (3). El uso del servidor virtual aumenta y alcanza el umbral de escalado automático (4). El VIM amplía el servidor virtual VM1 (5a) del tipo de configuración 1 al tipo 2 ($VM_TYPE = type2$). La API basada en eventos de VIM genera un evento de uso de recursos con marca de tiempo = $t2$, que el monitor de uso de la nube captura y reenvía al monitor de pago por uso (5b). El monitor de pago por uso interactúa con la base de datos del plan de precios para identificar las métricas de contracargo y uso que se aplican al uso de recursos de TI actualizado. Se genera un evento facturable de "uso modificado" y se almacena en la base de datos log de eventos facturables (6). El consumidor de la nube apaga el servidor virtual (7) y el VIM detiene el servidor virtual VM1 (8a). La API basada en eventos de VIM genera un evento de uso de recursos con marca de tiempo = $t3$, que el monitor de uso de la nube captura y reenvía al monitor de pago por uso (8b). El monitor de pago por uso interactúa con la base de datos del plan de precios para identificar las métricas de contracargo y uso que se aplican al uso de recursos de TI actualizado. Se genera un evento facturable de "uso terminado" y se almacena en la base de datos de registro de eventos facturables (9). El proveedor de la nube ahora puede utilizar la herramienta del sistema de facturación para acceder a la base de datos log y calcular la tarifa de uso total del servidor virtual como $(Fee(VM1))$ (10).

6.5. Audit Monitor

El mecanismo de Audit monitor (monitor de auditoría) se utiliza para recopilar datos de seguimiento de auditoría para redes y recursos de TI en apoyo de (o dictado por) obligaciones regulatorias y contractuales. La Figura 6.15 representa un monitor de auditoría implementado como un agente de

monitoreo que intercepta las solicitudes de "inicio de sesión" y almacena las credenciales de seguridad del solicitante, así como los intentos de inicio de sesión fallidos y exitosos, en una base de datos log para futuros informes de auditoría.

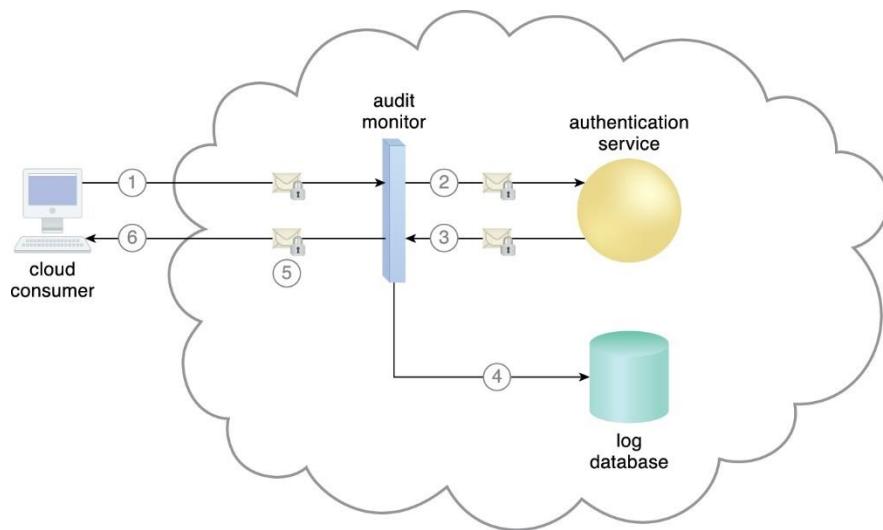


Figura 6.15 Un consumidor de servicios en la nube solicita acceso a un servicio en la nube enviando un mensaje de solicitud de inicio de sesión con credenciales de seguridad (1). El monitor de auditoría intercepta el mensaje (2) y lo reenvía al servicio de autenticación (3). El servicio de autenticación procesa las credenciales de seguridad. Se genera un mensaje de respuesta para el consumidor del servicio en la nube, además de los resultados del intento de inicio de sesión (4). El monitor de auditoría intercepta el mensaje de respuesta y almacena todos los detalles del evento de inicio de sesión recopilados en la base de datos log, según los requisitos de la política de auditoría de la organización (5). Se ha concedido el acceso y se envía una respuesta al consumidor del servicio en la nube (6).

Ejemplo de Estudio de Caso

Una característica clave de la solución de juego de roles de Innovartus es su interfaz de usuario única. Sin embargo, las tecnologías avanzadas utilizadas para su diseño han impuesto restricciones de licencia que legalmente impiden que Innovartus cobre a los usuarios en ciertas regiones geográficas por el uso de la solución. El departamento legal de Innovartus está trabajando para resolver estos problemas. Pero mientras tanto, ha proporcionado al departamento de TI una lista de países en los que los usuarios no pueden acceder a la aplicación o en los que el acceso de los usuarios debe ser gratuito.

Para recopilar información sobre el origen de los clientes que acceden a la aplicación, Innovartus solicita a su proveedor de nube que establezca un sistema de monitoreo de auditoría. El proveedor de la nube implementa un agente de monitoreo de auditoría para interceptar cada mensaje entrante, analizar su encabezado HTTP correspondiente y recopilar detalles sobre el origen del usuario final. Según la solicitud de Innovartus, el proveedor de la nube agrega además una base de datos log para recopilar los datos regionales de cada solicitud de usuario final para futuros informes. Innovartus actualiza aún más su aplicación para que los usuarios finales de países seleccionados puedan acceder a la aplicación sin cargo (Figura 6.16).

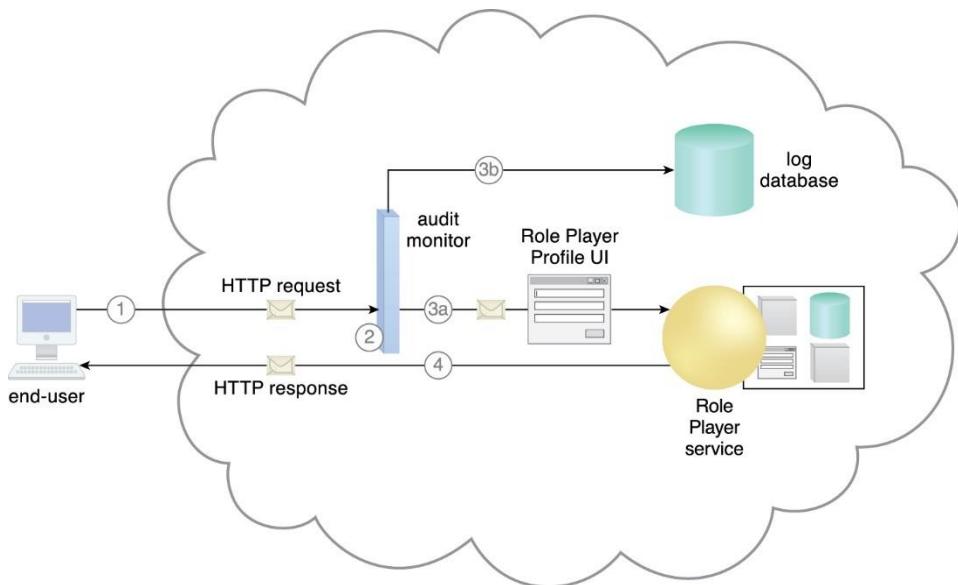


Figura 6.16 Un usuario final intenta acceder al servicio en la nube de Role Player (1). Un monitor de auditoría intercepta de forma transparente el mensaje de solicitud HTTP y analiza el encabezado del mensaje para determinar el origen geográfico del usuario final (2). El agente de monitoreo de auditoría determina que el usuario final es de una región en la que Innovartus no está autorizado a cobrar una tarifa por el acceso a la aplicación. El agente reenvía el mensaje al servicio en la nube (3a) y genera la información de la pista de auditoría para almacenarla en la base de datos de registro (3b). El servicio en la nube recibe el mensaje HTTP y otorga acceso al usuario final sin cargo (4).

6.6. Failover System

El failover system (sistema de conmutación por error) se utiliza para aumentar la confiabilidad y disponibilidad de los recursos de TI mediante el uso de tecnología de agrupación en clústeres establecida para proporcionar implementaciones redundantes. Un sistema de conmutación por error está configurado para cambiar automáticamente a una instancia de recurso de TI redundante o a standby cada vez que el recurso de TI actualmente activo se vuelva no disponible.

Los failover systems se utilizan comúnmente para programas de misión crítica y servicios reutilizables que pueden introducir un único punto de falla para múltiples aplicaciones. Un sistema de conmutación por error puede abarcar más de una región geográfica para que cada ubicación albergue una o más implementaciones redundantes del mismo recurso de TI.

El sistema de conmutación por error utiliza a veces el mecanismo de replicación de recursos para proporcionar instancias de recursos de TI redundantes, que se supervisan activamente para detectar errores y condiciones de indisponibilidad.

Los sistemas de conmutación por error vienen en dos configuraciones básicas:

Activo-Activo

En una configuración activo-activo, las implementaciones redundantes del recurso de TI atienden activamente la carga de trabajo de forma síncrona (Figura 6.17). Se requiere balanceo de carga entre las instancias activas. Cuando se detecta una falla, la instancia fallida se elimina del programador de

balanceo de carga (Figura 6.18). Cualquiera que sea el recurso de TI que permanece operativo cuando se detecta una falla, se hace cargo del procesamiento (Figura 6.19).

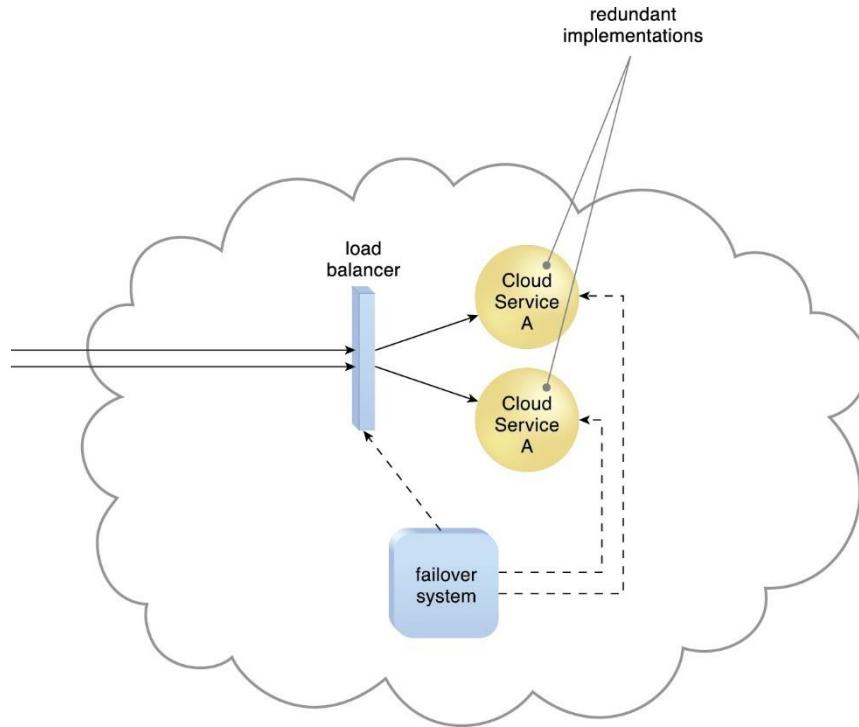


Figura 6.17 El failover system monitorea el estado operacional del servicio de nube A.

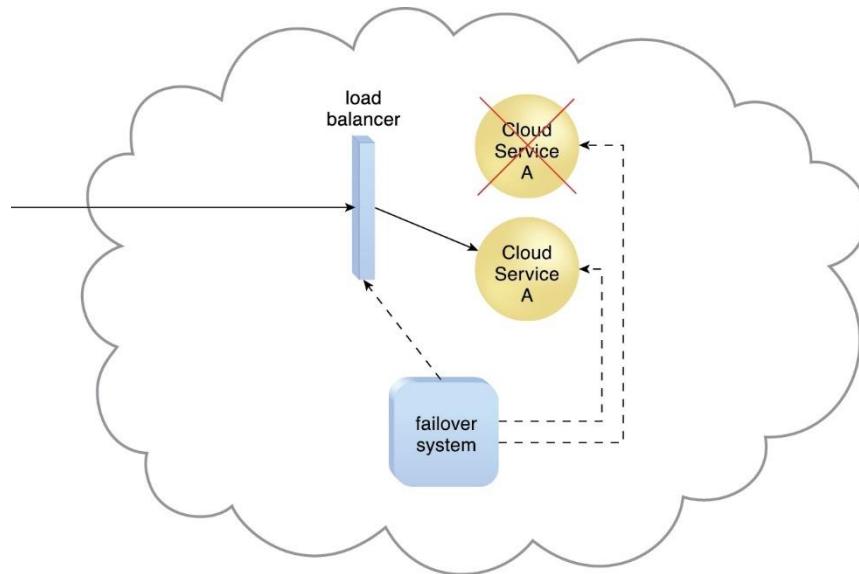


Figura 6.18 Cuando una falla es detectada en una implementación de servicio de nube A, el failover system ordena al balanceador de carga que commute la carga de trabajo a una implementación de servicio de nube A redundante.

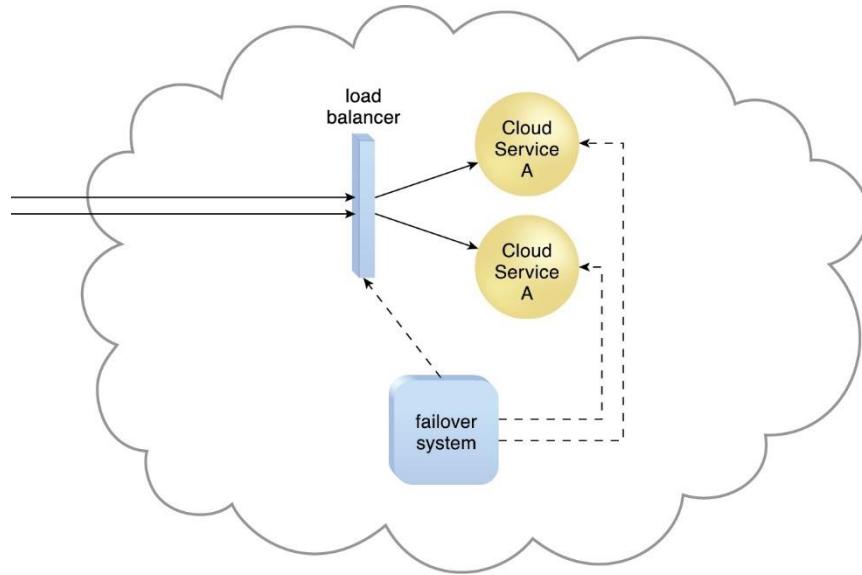


Figura 6.19 La implementación fallida del servicio en la nube A se recupera o replica en un servicio en la nube operativo. El sistema de conmutación por error ahora ordena al balanceador de carga que distribuya la carga de trabajo nuevamente.

Activo-pasivo

En una configuración activo-pasivo, se activa una implementación en espera o inactiva para hacerse cargo del procesamiento del recurso de TI que deja de estar disponible, y la carga de trabajo correspondiente se redirige a la instancia que se hace cargo de la operación (Figuras 6.20 a 6.22).

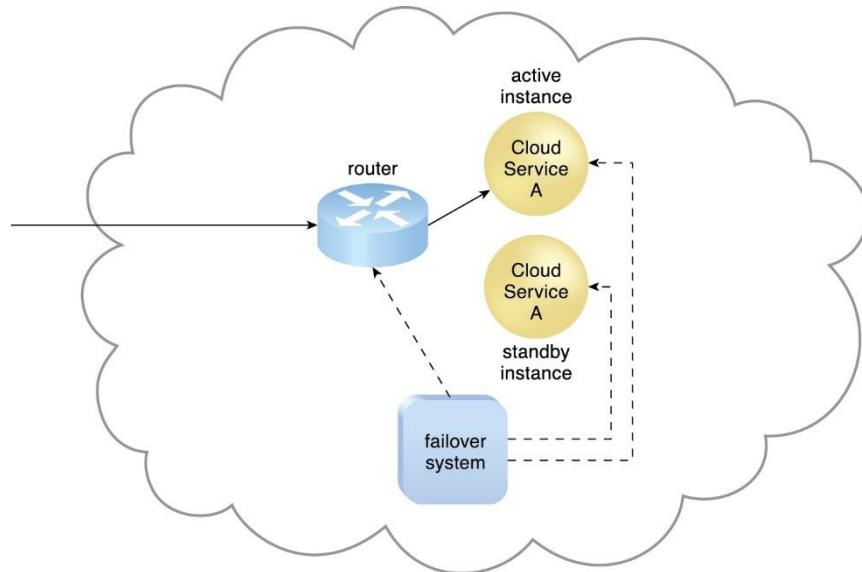


Figura 6.20 El failover system monitorea el estado operacional del servicio de nube A. La implementación del servicio de nube A que está actuando como la instancia activa está recibiendo las solicitudes de los consumidores del servicio de nube.

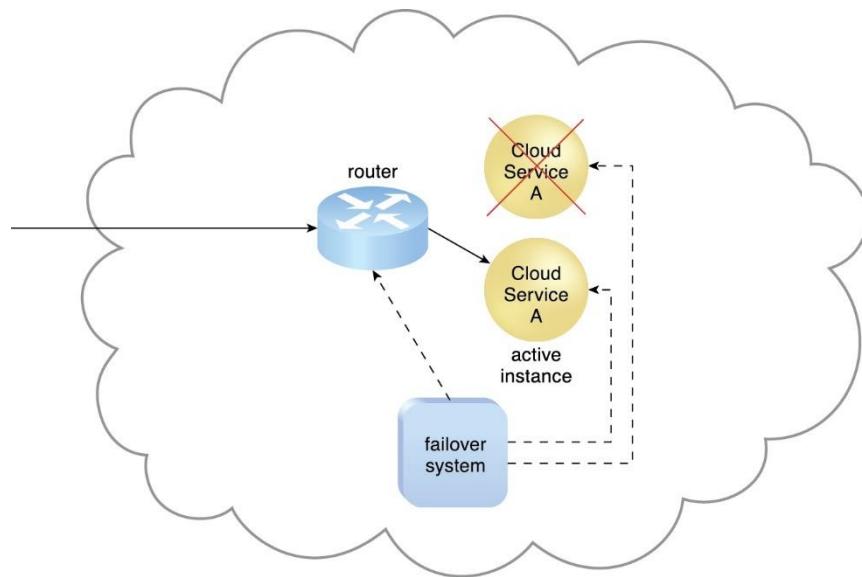


Figura 6.21 La implementación del Servicio en la nube A que actúa como la instancia activa encuentra una falla que es detectada por el failover system, que posteriormente activa la implementación del Servicio en la nube A inactiva y redirige la carga de trabajo hacia ella. La implementación del Servicio en la Nube A recién invocada ahora asume el rol de instancia activa.

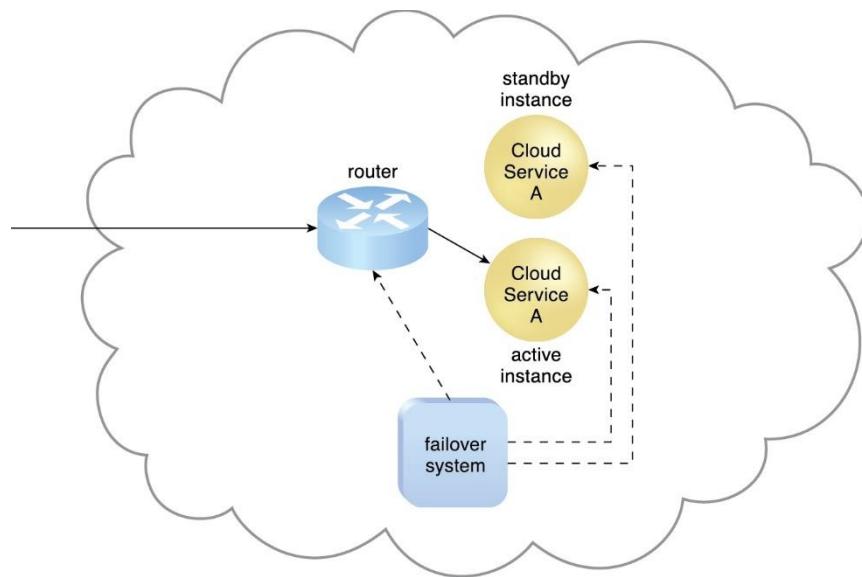


Figura 6.22 La implementación fallida del Servicio en la nube A se recupera o replica en un servicio en la nube operativo y ahora se posiciona como la instancia en espera, mientras que el Servicio en la nube A invocado anteriormente continúa sirviendo como la instancia activa.

Algunos failover system están diseñados para redirigir las cargas de trabajo a recursos de TI activos que se basan en平衡adores de carga especializados que detectan condiciones de falla y excluyen las instancias de recursos de TI fallidas de la distribución de la carga de trabajo. Este tipo de failover system es adecuado para recursos de TI que no requieren administración del estado de ejecución y brindan capacidades de procesamiento sin estado. En las arquitecturas tecnológicas que normalmente se basan en tecnologías de agrupación en clústeres y virtualización, también se

requiere que las implementaciones de recursos de TI redundantes o en espera comparten su estado y contexto de ejecución. Una tarea compleja que se ejecutó en un recurso de TI fallido puede permanecer operativa en una de sus implementaciones redundantes.

Ejemplo de Estudio de Caso

DTGOV crea un servidor virtual resiliente para soportar la asignación de instancias de servidor virtual que alojan aplicaciones críticas, las cuales se replican en múltiples centros de datos. El servidor virtual resiliente replicado tiene un failover system activo-pasivo asociado. Su flujo de tráfico de red se puede cambiar entre las instancias de recursos de TI que residen en diferentes centros de datos, si la instancia activa fallara (Figura 6.23).

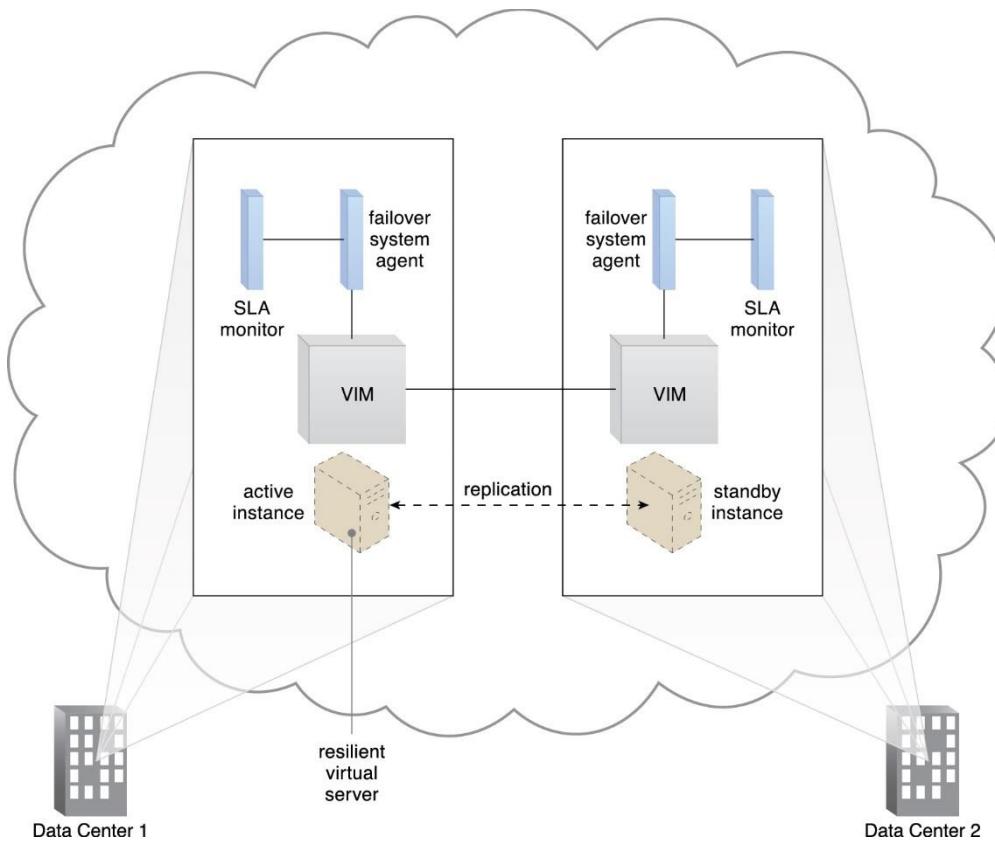


Figura 6.23 Se establece un servidor virtual resiliente mediante la replicación de la instancia del servidor virtual en dos centros de datos diferentes, tal como lo realiza el VIM que se ejecuta en ambos centros de datos. La instancia activa recibe el tráfico de la red y se escala verticalmente como respuesta, mientras que la instancia en espera no tiene carga de trabajo y se ejecuta con la configuración mínima.

La Figura 6.24 ilustra los monitores SLA que detectan fallas en una instancia activa de un servidor virtual.

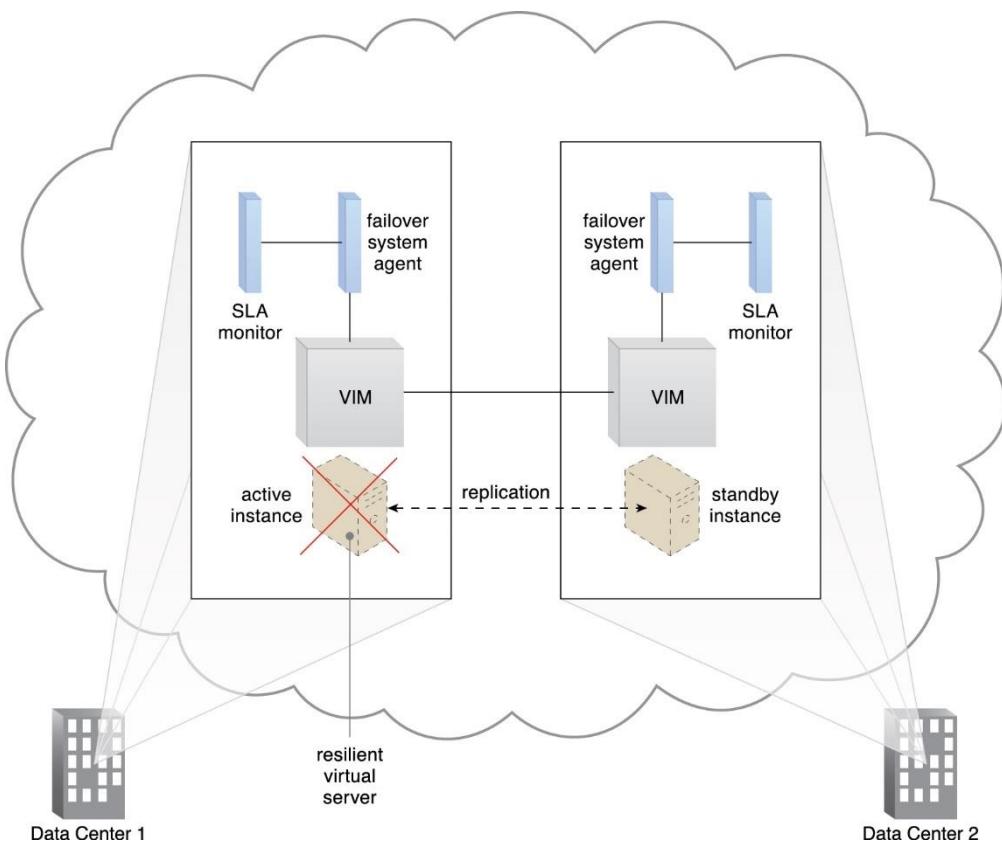


Figura 6.24 Los monitores de SLA detectan cuándo la instancia activa del servidor virtual deja de estar disponible.

La Figura 6.25 muestra el tráfico que se cambia a la instancia en espera, que ahora se ha vuelto activa.

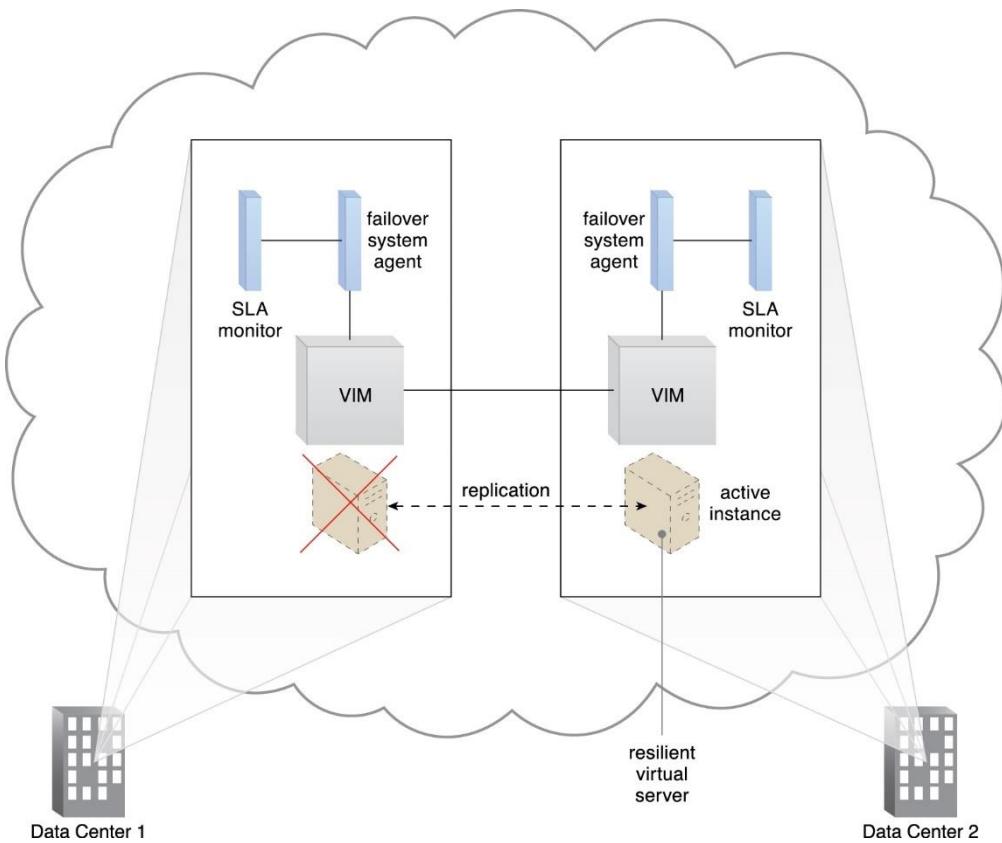


Figura 6.25 El failover system se implementa como un agente de software controlado por eventos que intercepta las通知aciones de mensajes que los monitores SLA envían con respecto a la falta de disponibilidad del servidor. En respuesta, el failover system interactúa con el VIM y las herramientas de administración de red para redirigir todo el tráfico de red a la instancia de reserva ahora activa.

En la Figura 6.26, el servidor virtual fallido se vuelve operativo y se convierte en la instancia de reserva.

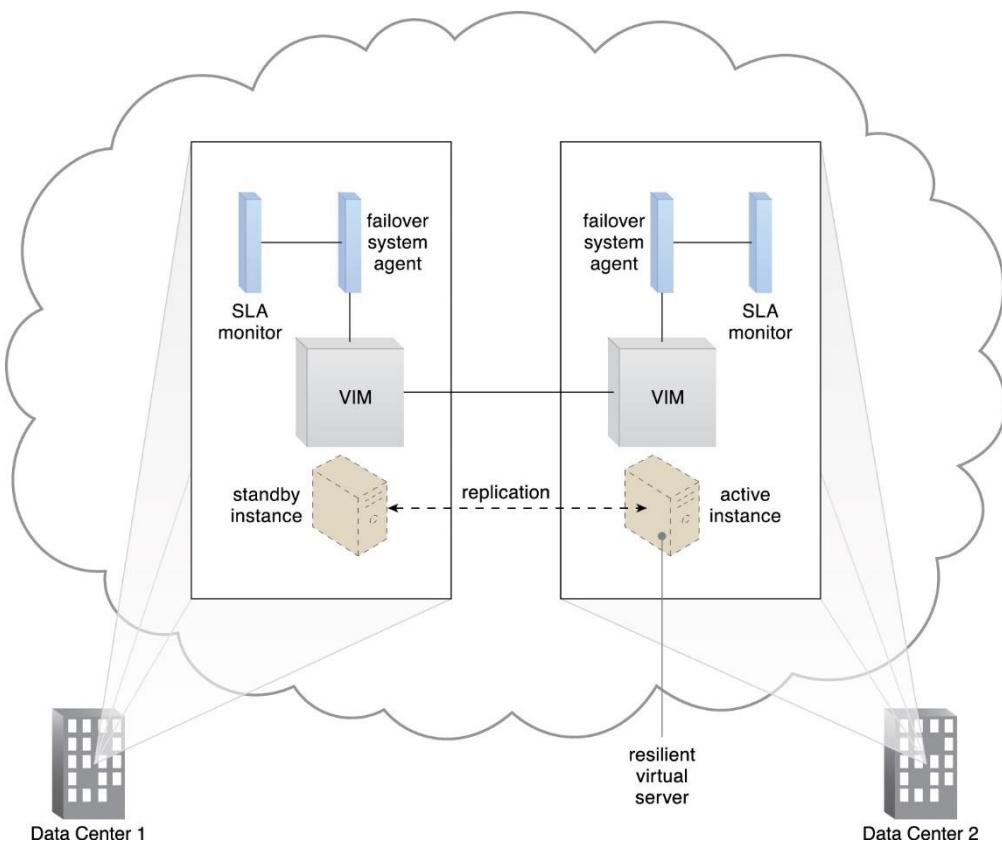


Figura 6.26 La instancia de servidor virtual fallida se reactiva y se reduce a la configuración mínima de instancia en standby después de que reanuda su funcionamiento normal.

6.7. Hipervisor

El mecanismo de *hipervisor* es una parte fundamental de la infraestructura de virtualización que se utiliza principalmente para generar instancias de servidor virtual a partir de un servidor físico. Un hipervisor generalmente se limita a un servidor físico y, por lo tanto, solo puede crear imágenes virtuales de ese servidor (Figura 6.27). De manera similar, un hipervisor solo puede asignar servidores virtuales que generan a grupos de recursos que residen en el mismo servidor físico subyacente. Un hipervisor tiene funciones de administración de servidores virtuales limitadas, como aumentar la capacidad del servidor virtual o apagarlo. El VIM proporciona una variedad de características para administrar múltiples hipervisores actuando en servidores físicos.

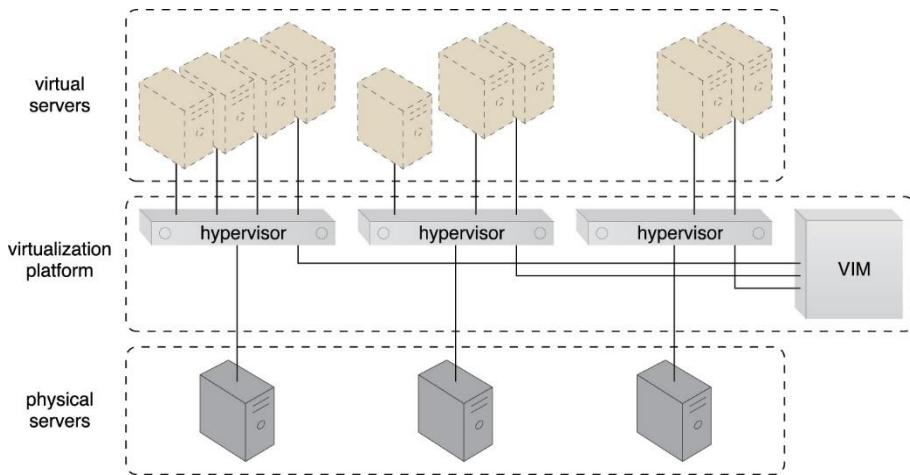


Figura 6.27 Los servidores virtuales se crean a través de un hipervisor individual en servidores físicos individuales. Los tres hipervisores están controlados conjuntamente por el mismo VIM.

El software de hipervisor se puede instalar directamente en servidores bare-metal³¹ y proporciona funciones para controlar, compartir y calendarizar el uso de los recursos de hardware, como la potencia del procesador, la memoria y la E/S. Estos pueden aparecer en el sistema operativo de cada servidor virtual como recursos dedicados.

Ejemplo de Estudio de Caso

DTGOV ha establecido una plataforma de virtualización en la que el mismo producto de software de hipervisor se ejecuta en todos los servidores físicos. El VIM coordina los recursos de hardware en cada centro de datos para que se puedan crear instancias de servidor virtual desde el servidor físico subyacente más conveniente.

Como resultado, los consumidores de la nube pueden arrendar servidores virtuales con funciones de escalado automático. Para ofrecer configuraciones flexibles, la plataforma de virtualización DTGOV proporciona migración de VM en vivo de servidores virtuales entre servidores físicos dentro del mismo centro de datos. Esto se ilustra en las Figuras 6.28 y 6.29, donde un servidor virtual se migra en vivo de un servidor físico ocupado a otro que está inactivo, lo que le permite escalar en respuesta a un aumento en su carga de trabajo.

³¹ Una máquina desnuda, o 'bare metal', en informática, significa cuando no hay un núcleo (S.O.) instalado en el hardware. También se utiliza en los sistemas cloud cuando se parte de una máquina sin Sistema Operativo en cloud para incluir máquinas virtuales. Fuente: Wikipedia.

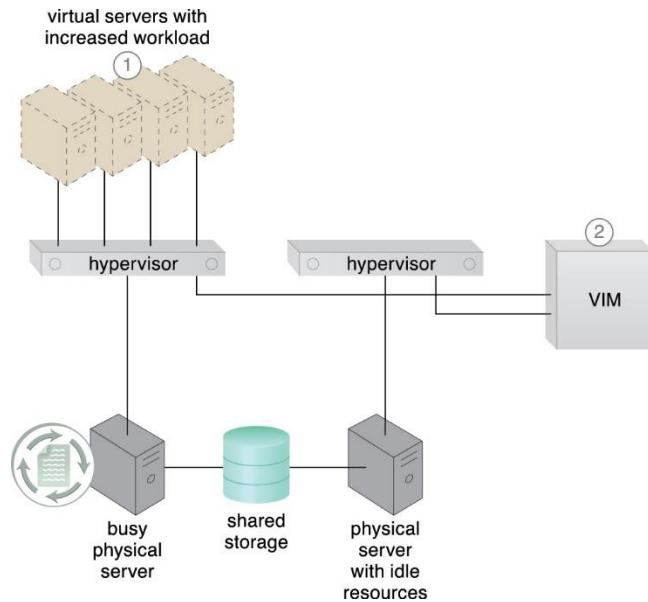


Figura 6.28 Un servidor virtual capaz de escalar automáticamente experimenta un aumento en su carga de trabajo (1). El VIM decide que el servidor virtual no puede escalar porque su servidor físico subyacente está siendo utilizado por otros servidores virtuales (2).

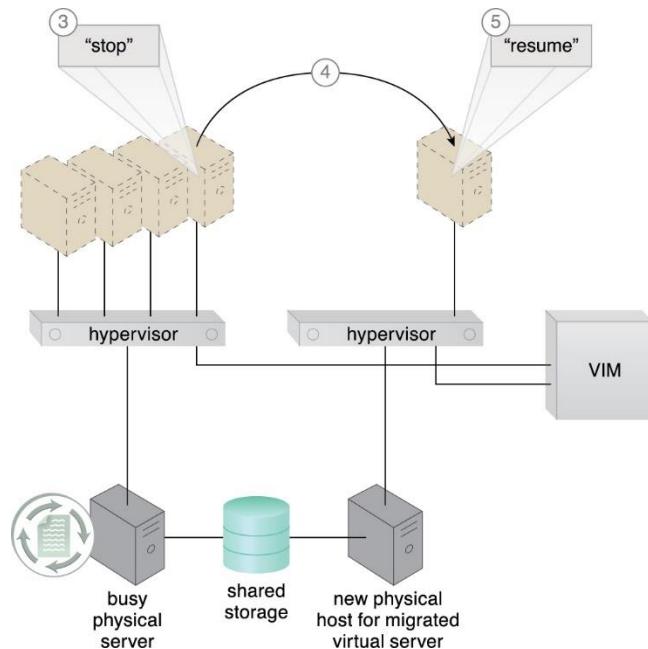


Figura 6.29 El VIM ordena al hipervisor en el servidor físico ocupado que suspenda la ejecución del servidor virtual (3). Luego, el VIM ordena la instalación del servidor virtual en el servidor físico inactivo. La información de estado (como las páginas de memoria y los registros del procesador) se sincroniza a través de un dispositivo de almacenamiento en la nube compartido (4). El VIM ordena al hipervisor en el nuevo servidor físico que reanude el procesamiento del servidor virtual (5).

6.8. Clúster de recursos

Los recursos de TI basados en la nube que están geográficamente dispersos se pueden combinar lógicamente en grupos para mejorar su asignación y uso. El mecanismo de clúster de recursos (Figura 6.30) se utiliza para agrupar varias instancias de recursos de TI para que puedan funcionar como un solo recurso de TI. Esto aumenta la capacidad informática combinada, el balanceo de carga y la disponibilidad de los recursos de TI como cluster.

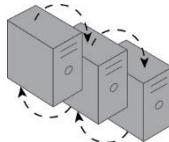


Figura 6.30 Las líneas discontinuas curvas se utilizan para indicar que los recursos de TI están agrupados.

Las arquitecturas de clúster de recursos se basan en conexiones de red dedicadas de alta velocidad, o nodos de clúster entre instancias de recursos de TI que se comunican para la distribución de la carga de trabajo, la programación de tareas, el uso compartido de datos y la sincronización del sistema. Una plataforma de administración de cluster que se ejecuta como middleware distribuido en todos los nodos del clúster suele ser responsable de estas actividades. Esta plataforma implementa una función de coordinación que permite que los recursos de TI distribuidos aparezcan como un recurso de TI y también ejecuta recursos de TI dentro del clúster.

Los tipos de clústeres de recursos comunes incluyen:

- **Clúster de servidores:** Los servidores físicos o virtuales se agrupan para aumentar el rendimiento y la disponibilidad. Los hipervisores que se ejecutan en diferentes servidores físicos se pueden configurar para compartir el estado de ejecución del servidor virtual (como las páginas de memoria y el estado de registro del procesador) para establecer servidores virtuales en clúster. En tales configuraciones, que generalmente requieren servidores físicos para tener acceso al almacenamiento compartido, los servidores virtuales pueden migrar en vivo de uno a otro. En este proceso, la plataforma de virtualización suspende la ejecución de un servidor virtual determinado en un servidor físico y la reanuda en otro servidor físico. El proceso es transparente para el sistema operativo del servidor virtual y se puede utilizar para aumentar la escalabilidad mediante la migración en vivo de un servidor virtual que se ejecuta en un servidor físico sobrecargado a otro servidor físico que tenga la capacidad adecuada.
- **Clúster de base de datos:** Diseñado para mejorar la disponibilidad de datos, este clúster de recursos de alta disponibilidad tiene una función de sincronización que mantiene la coherencia de los datos almacenados en diferentes dispositivos de almacenamiento utilizados en el clúster. La capacidad redundante generalmente se basa en un sistema de failover activo-activo o activo-pasivo comprometido con el mantenimiento de las condiciones de sincronización.
- **Clúster de grandes conjuntos de datos:** Se implementa la partición y distribución de datos para que los conjuntos de datos de destino se puedan dividir de manera eficiente sin comprometer la integridad de los datos o la precisión informática. Cada nodo de clúster procesa cargas de trabajo sin comunicarse con otros nodos, así como con otros tipos de clúster.

Muchos clústeres de recursos requieren que los nodos del clúster tengan una capacidad informática y características casi idénticas para simplificar el diseño y mantener la coherencia dentro de la arquitectura del clúster de recursos. Los nodos de clúster en arquitecturas de clúster de alta disponibilidad necesitan acceder y compartir recursos de TI de almacenamiento comunes. Esto puede requerir dos capas de comunicación entre los nodos: una para acceder al dispositivo de almacenamiento y otra para ejecutar la orquestación de recursos de TI (Figura 6.31). Algunos clústeres de recursos están diseñados con recursos de TI menos acoplados que solo requieren la capa de red (Figura 6.32).

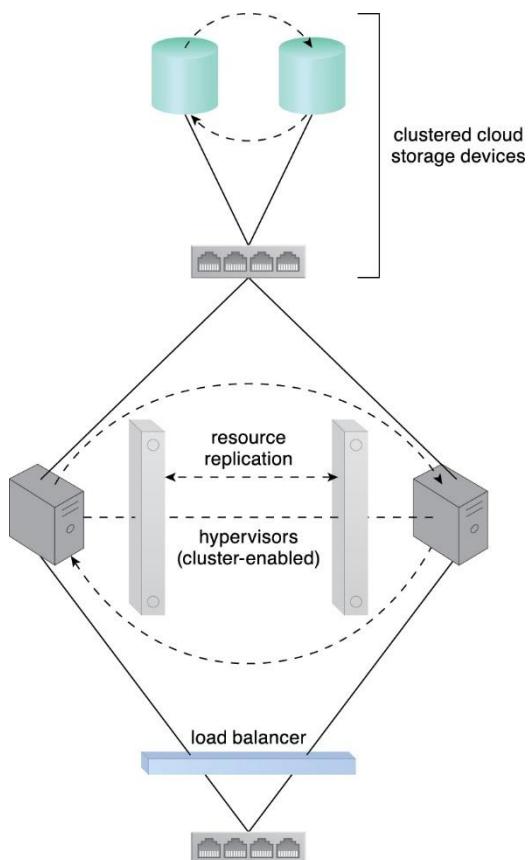


Figura 6.31 El balanceo de carga y la replicación de recursos se implementan a través de un hipervisor habilitado para clúster. Se utiliza una red en área de almacenamiento dedicada para conectar el almacenamiento en clúster y los servidores en clúster, que pueden compartir dispositivos comunes de almacenamiento en la nube. Esto simplifica el proceso de replicación de almacenamiento, que se lleva a cabo de forma independiente en el clúster de almacenamiento.

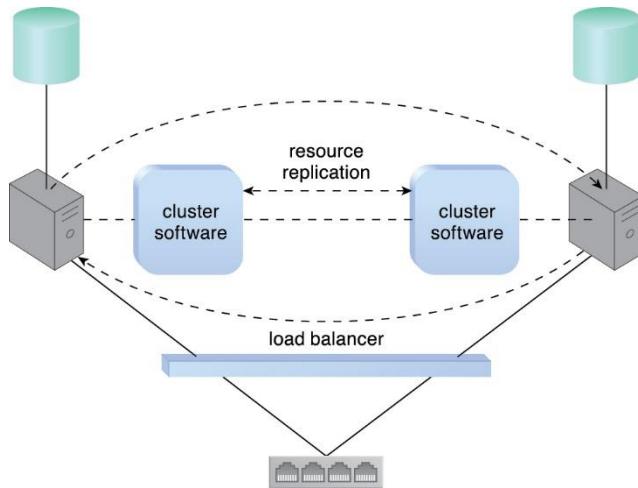


Figura 6.32 Un clúster de servidores débilmente acoplado que incorpora un balanceador de carga. No hay almacenamiento compartido. La replicación de recursos se utiliza para replicar dispositivos de almacenamiento en la nube a través de la red mediante el software del clúster.

Hay dos tipos básicos de clústeres de recursos:

- Clúster de carga balanceada: este clúster de recursos se especializa en distribuir cargas de trabajo entre los nodos del clúster para aumentar la capacidad de los recursos de TI y preservar la centralización de la administración de recursos de TI. Por lo general, implementa un mecanismo balanceador de carga que está integrado en la plataforma de administración de clústeres o configurado como un recurso de TI separado.
- Clúster HA: un clúster de alta disponibilidad (High-Availability) mantiene la disponibilidad del sistema en caso de fallas de varios nodos y tiene implementaciones redundantes de la mayoría o la totalidad de los recursos de TI agrupados. Implementa un mecanismo de sistema de failover que monitorea las condiciones de falla y automáticamente redirige la carga de trabajo lejos de cualquier nodo fallido.

El aprovisionamiento de recursos de TI agrupados puede ser considerablemente más costoso que el aprovisionamiento de recursos de TI individuales que tienen una capacidad informática equivalente.

Ejemplo de Estudio de Caso

DTGOV está considerando introducir un servidor virtual en clúster para ejecutarse en un clúster de alta disponibilidad como parte de la plataforma de virtualización (Figura 6.33). Los servidores virtuales pueden migrar en vivo entre los servidores físicos, que se agrupan en un clúster de hardware de alta disponibilidad controlado por hipervisores coordinados habilitados para clústeres. La función de coordinación mantiene instantáneas replicadas de los servidores virtuales en ejecución para facilitar la migración a otros servidores físicos en caso de falla.

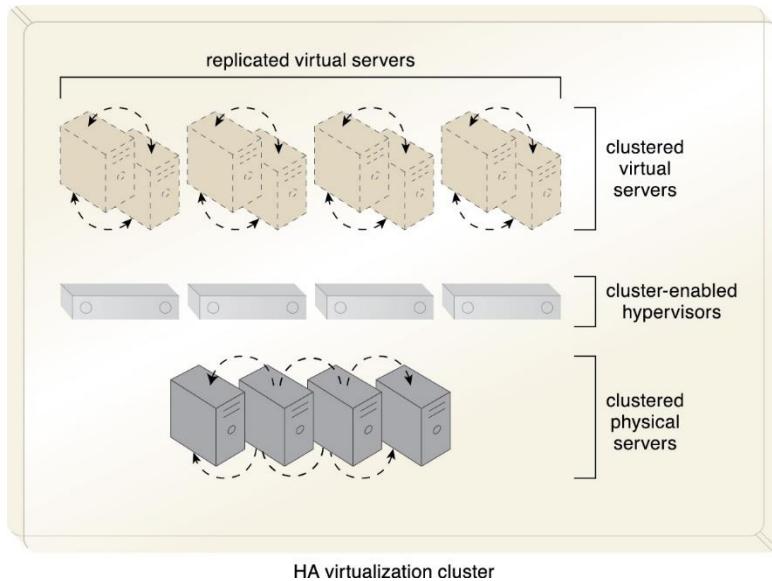


Figura 6.33 Se implementa un clúster de virtualización HA de servidores físicos mediante un hipervisor habilitado para clúster, lo que garantiza que los servidores físicos estén constantemente sincronizados. Cada servidor virtual que se instancia en el clúster se replica automáticamente en al menos dos servidores físicos.

En la Figura 6.34 se identifica a los servidores virtuales que se migran desde su servidor host físico fallido a otros servidores físicos disponibles.

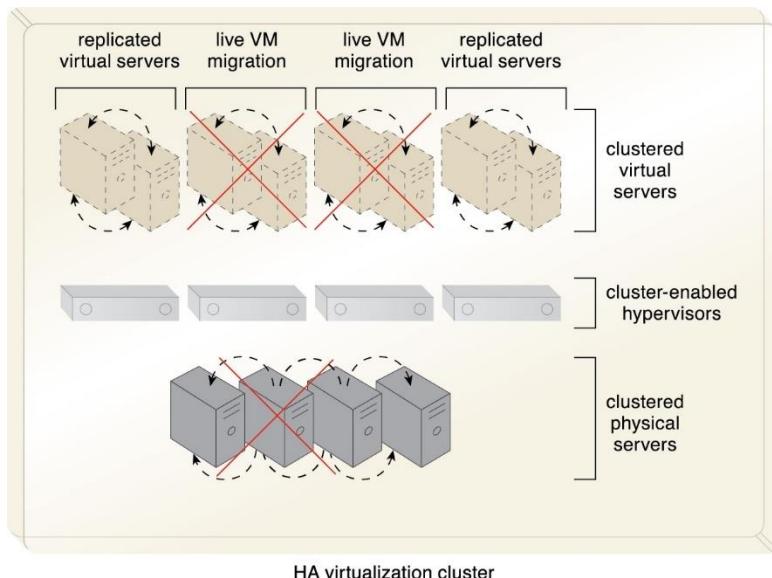


Figura 6.34 Todos los servidores virtuales alojados en un servidor físico que presenta fallas migran automáticamente a otros servidores físicos.

6.9. Multi-device Broker

Es posible que una variedad de consumidores de servicios en la nube, deban acceder a un servicio en la nube individual diferenciados por sus dispositivos de hardware y/o requisitos de comunicación. Para superar las incompatibilidades entre un servicio en la nube y un consumidor de servicios en la

nube diferente, se debe crear un mapeo lógico para transformar (o convertir) la información que se intercambia en tiempo de ejecución.

El multi-device broker(mecanismo intermediario de dispositivos múltiples) se utiliza para facilitar la transformación de datos en tiempo de ejecución para que un servicio en la nube sea accesible para una gama más amplia de programas y dispositivos consumidores de servicios en la nube (Figura 8.35).

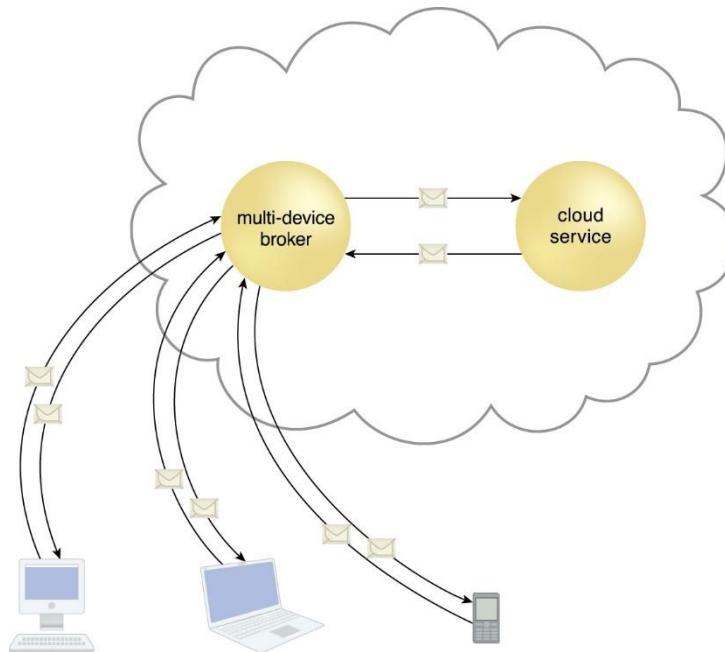


Figura 6.35 Un multi-device broker contiene la lógica de mapeo necesaria para transformar los intercambios de datos entre un servicio en la nube y diferentes tipos de dispositivos de consumidores de servicios en la nube. Este escenario representa al multi-device broker como un servicio en la nube con su propia API. Este mecanismo también se puede implementar como un agente de servicio que intercepta mensajes en tiempo de ejecución para realizar las transformaciones necesarias.

Los intermediarios de dispositivos múltiples suelen existir como gateway³² o incorporan componentes de un gateway, como:

- XML Gateway: transmite y valida datos XML
- Cloud Storage Gateway: transforma los protocolos de almacenamiento en la nube y codifica dispositivos de almacenamiento para facilitar la transferencia y el almacenamiento de datos
- Mobile Device Gateway: transforma los protocolos de comunicación utilizados por los dispositivos móviles en protocolos que son compatibles con un servicio en la nube.

Los niveles en los que se puede crear la lógica de transformación incluyen:

- protocolos de transporte

³² La puerta de enlace (en inglés gateway) es el dispositivo que actúa de interfaz de conexión entre aparatos o dispositivos, y también posibilita compartir recursos entre dos o más ordenadores. Fuente: Wikipedia.

- protocolos de mensajería
- protocolos de dispositivos de almacenamiento
- esquemas de datos/modelos de datos

Por ejemplo, multi-device broker puede contener una lógica de mapeo que cubra los protocolos de transporte y mensajería para un consumidor de servicios en la nube que accede a un servicio en la nube con un dispositivo móvil.

Ejemplo de Estudio de Caso

Innovartus ha decidido hacer que su aplicación de juegos de rol esté disponible para varios dispositivos móviles y teléfonos inteligentes. Una complicación que obstaculizó al equipo de desarrollo de Innovartus durante la etapa de diseño de mejoras a móviles fue la dificultad de reproducir experiencias de usuario idénticas en diferentes plataformas móviles. Para resolver este problema, Innovartus implementa un multi-device broker para interceptar los mensajes entrantes de los dispositivos, identificar la plataforma de software y convertir el formato del mensaje al formato nativo de la aplicación del lado del servidor (Figura 6.36).

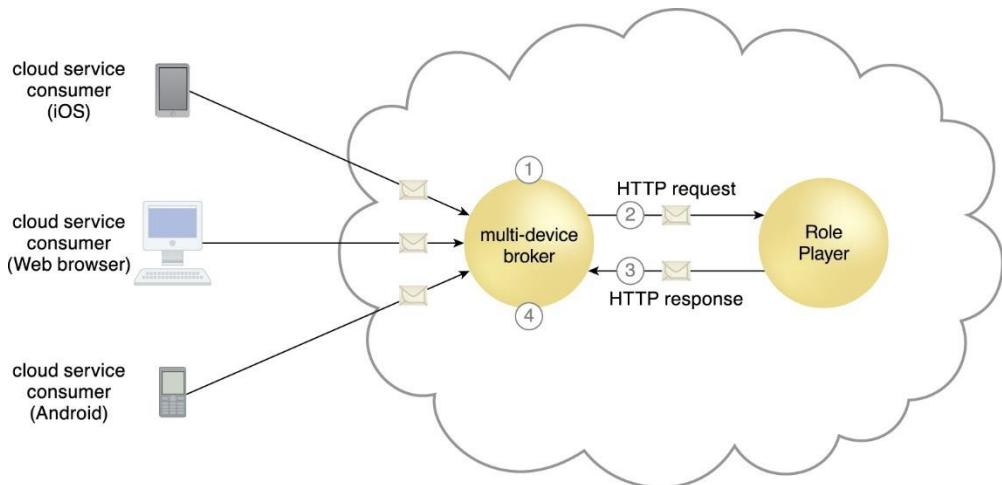


Figura 6.36 El multi-device broker intercepta los mensajes entrantes y detecta la plataforma (navegador web, iOS, Android) del dispositivo de origen (1). El broker multidispositivo transforma el mensaje al formato estándar requerido por el servicio en la nube de innovartus (2). El servicio en la nube procesa la solicitud y responde utilizando el mismo formato estándar (3). El intermediario de dispositivos múltiples transforma el mensaje de respuesta al formato requerido por el dispositivo de origen y entrega el mensaje (4).

6.10. State Management Database

Una state management database (base de datos de administración de estado) es un dispositivo de almacenamiento que se utiliza para almacenar temporalmente datos persistentes de estado³³ en programas de software. Como una alternativa al almacenamiento en caché de datos de estado en la memoria, los programas de software pueden descargar datos de estado a la base de datos para

³³ Un estado es una configuración única de información en un programa o máquina. Un Sistema de información o protocolo que se basa en estados se dice que es con estado(stateful). Uno que no lo es por el contrario se le denomina sin estado(stateless). Por ejemplo, hay firewalls y servidores sin estado, y HTTP se considera un protocolo sin estado. Fuente: Wikipedia.

reducir la cantidad de memoria en tiempo de ejecución que consumen (Figuras 6.37 y 6.38). Al hacerlo, los programas de software y la infraestructura circundante son más escalables. Las state management database son comúnmente utilizadas por los servicios en la nube, especialmente aquellos involucrados en actividades de tiempo de ejecución de larga duración.

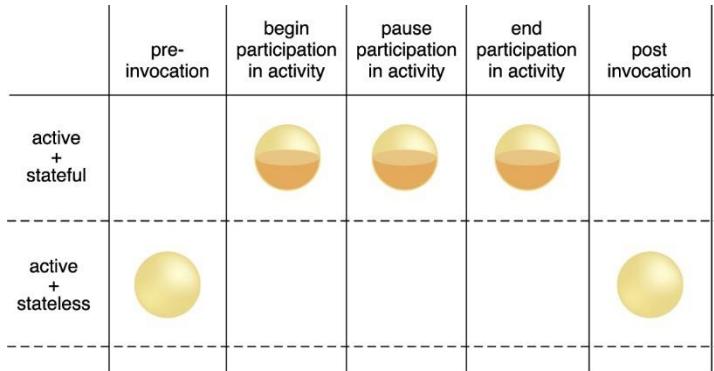


Figura 6.37 Durante el tiempo de vida de una instancia de servicio de nube esta puede requerir mantenerse con estado(stateful) y mantener el estado de los datos que se encuentran en memoria caché, aun cuando está inactivo.

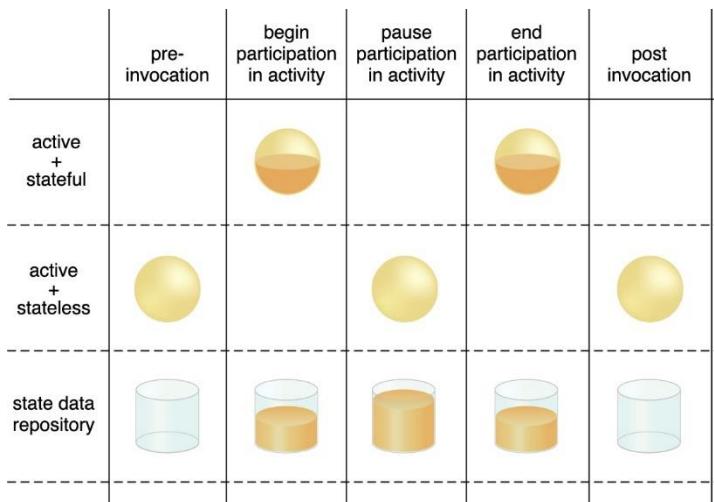


Figura 6.38 Al diferir los datos de estado a un repositorio de estado, el servicio en la nube puede pasar a una condición stateless (sin estado) (o parcialmente stateless), liberando temporalmente los recursos del sistema.

Ejemplo de Estudio de Caso

ATN está ampliando su arquitectura de entorno ready-made para permitir postergar la información de estado durante períodos prolongados mediante la utilización del mecanismo de state management database. La Figura 6.39 demuestra cómo un consumidor de servicios en la nube que trabaja con un entorno ya preparado pausa la actividad, lo que hace que el entorno descargue los datos de estado almacenados en caché.

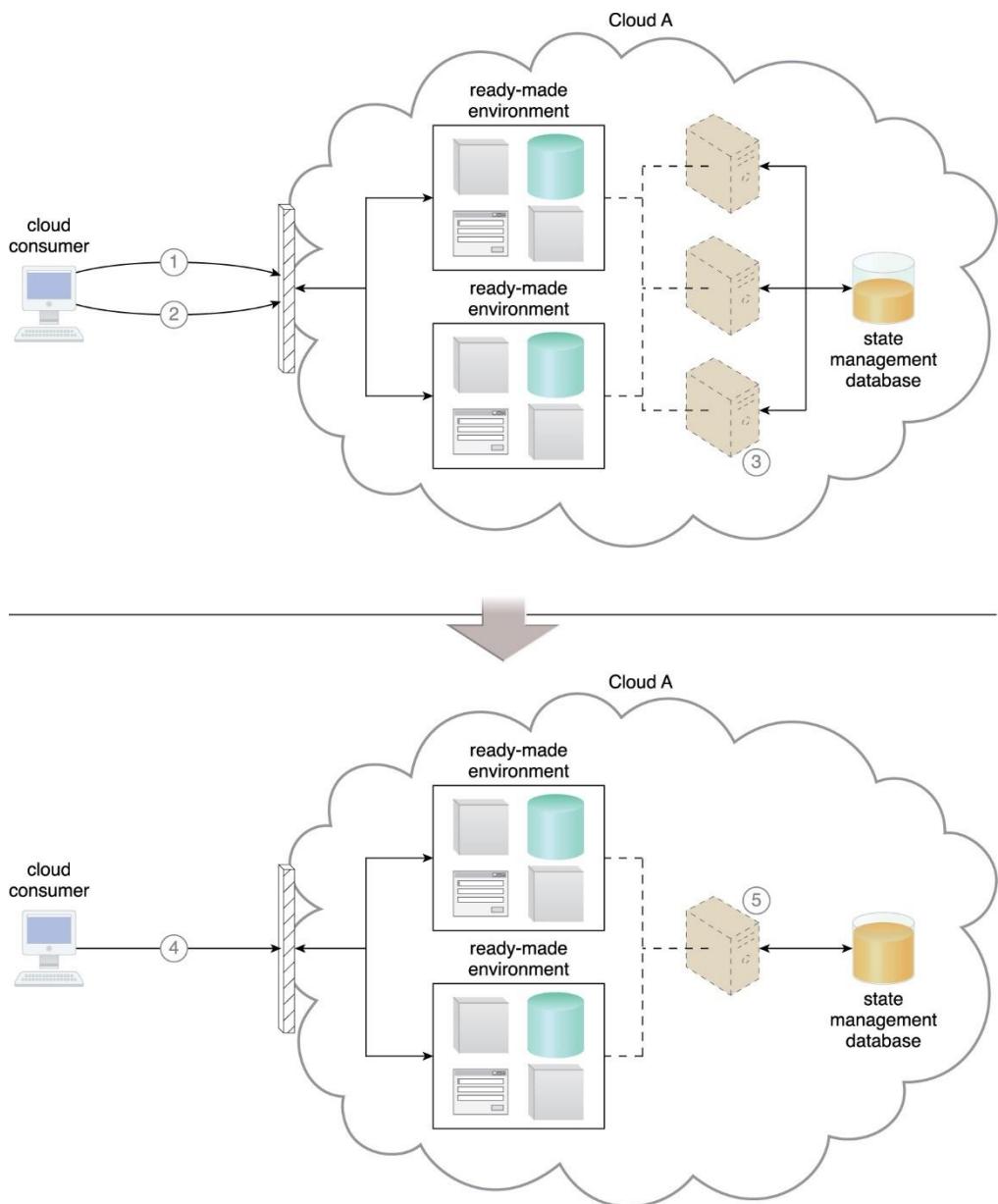


Figura 6.39 El consumidor de la nube accede al entorno ready-made y requiere tres servidores virtuales para realizar todas las actividades (1). El consumidor de la nube pausa la actividad. Todos los datos de estado deben conservarse para el acceso futuro al entorno ready-made (2). La infraestructura subyacente se escala automáticamente al reducir la cantidad de servidores virtuales. Los datos de estado se guardan en la base de datos de administración de estado y un servidor virtual permanece activo para permitir futuros inicios de sesión por parte del consumidor de la nube (3). En un momento posterior, el consumidor de la nube inicia sesión y accede al entorno preparado para continuar con la actividad (4). La infraestructura subyacente se escala automáticamente aumentando la cantidad de servidores virtuales y recuperando los datos de estado de la state management database (5).

7 Mecanismos de administración de la nube



Los recursos de TI basados en la nube deben instalarse, configurarse, mantenerse y monitorearse. Los sistemas cubiertos en este capítulo son mecanismos que abarcan y permiten este tipo de tareas de gestión. Forman partes clave de las arquitecturas de tecnología en la nube al facilitar el control y la evolución de los recursos de TI que forman las plataformas y soluciones en la nube.

Los siguientes mecanismos relacionados con la gestión se describen en este capítulo:

- Sistema de administración remota
- Sistema de gestión de recursos
- Sistema de gestión de SLA
- Sistema de gestión de facturación

Estos sistemas suelen proporcionar APIs integradas y se pueden ofrecer como productos individuales, aplicaciones personalizadas o combinados en varios productos, suites o aplicaciones multifunción.

7.1. Sistema de administración remota

El mecanismo del sistema de administración remota (Figura 7.1) proporciona herramientas e interfaces de usuario para que los administradores de recursos de nube externos configuren y administren recursos de TI basados en la nube.

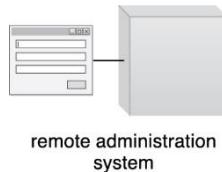


Figura 7.1 El símbolo utilizado en este libro para el sistema de administración remota. La interfaz de usuario mostrada normalmente estará etiquetada para indicar un tipo específico de portal.

Un sistema de administración remota puede establecer un portal para acceder a las funciones de administración y gestión de varios sistemas subyacentes, incluidos los sistemas de gestión de recursos, gestión de SLA y gestión de facturación descritos en este capítulo (Figura 7.2).

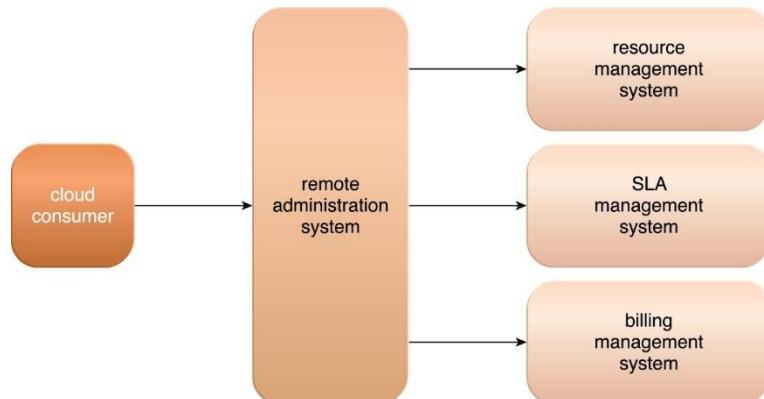


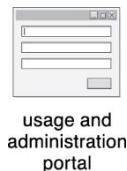
Figura 7.2 El sistema de administración remota abstracta los sistemas de administración subyacentes para exponer y centralizar los controles de administración a los administradores de recursos de nube externos. El

sistema proporciona una consola de usuario personalizable, mientras interactúa con los sistemas de administración subyacentes a través de sus APIs.

Las herramientas y las API proporcionadas por un sistema de administración remota generalmente son utilizadas por el proveedor de la nube para desarrollar y personalizar portales en línea que brindan a los consumidores de la nube una variedad de controles administrativos.

Los siguientes son los dos tipos principales de portales que se crean con el sistema de administración remota:

- Portal de uso y administración: Un portal de propósito general que centraliza los controles de administración para diferentes recursos de TI basados en la nube y puede proporcionar informes de uso de recursos de TI.



- Portal de autoservicio: se trata esencialmente de un portal de compras que permite a los consumidores de la nube buscar una lista actualizada de servicios en la nube y recursos de TI que están disponibles a través de un proveedor de la nube (generalmente para alquilarse). El consumidor de la nube envía sus artículos elegidos al proveedor de la nube para el aprovisionamiento.

La Figura 7.3 ilustra un escenario que involucra a un sistema de administración remota y también los portales de uso y administración como de auto servicio.

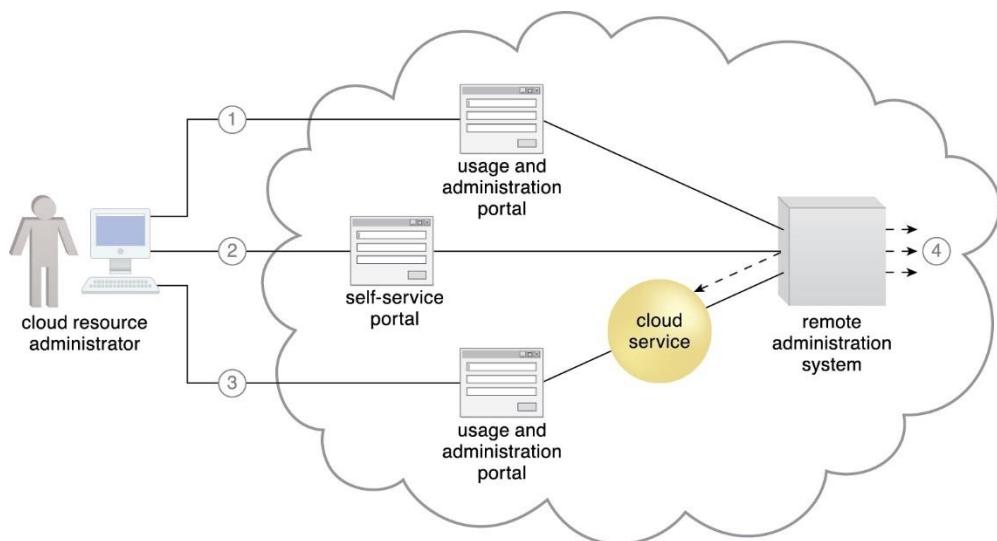


Figura 7.3 Un administrador de recursos de la nube usa el portal de uso y administración para configurar un servidor virtual ya alquilado (no se muestra) para prepararlo para el alojamiento (1). El administrador de recursos de la nube luego usa el portal de autoservicio para seleccionar y solicitar el aprovisionamiento de un nuevo servicio en la nube (2). Luego, el administrador de recursos de la nube accede nuevamente al portal de uso y administración para configurar el servicio en la nube recién aprovisionado que está alojado en el servidor

virtual (3). A lo largo de estos pasos, el sistema de administración remota interactúa con los sistemas de gestión necesarios para realizar las acciones solicitadas (4).

Dependiendo de:

- el tipo de producto en la nube o modelo de entrega en la nube que el consumidor de la nube está alquilando o utilizando del proveedor de la nube,
- el nivel de control de acceso otorgado por el proveedor de la nube al consumidor de la nube, y
- además, dependiendo de con qué sistema de gestión subyacente interactúa el sistema de administración remota,

... las tareas que normalmente pueden realizar los consumidores de la nube a través de una consola de administración remota incluyen:

- configurar y establecer servicios en la nube
- aprovisionar y liberar recursos de TI para servicios en la nube bajo demanda
- monitorear el estado, uso y rendimiento
- monitorear el cumplimiento de QoS y SLA
- administrar los costos de arrendamiento y las tarifas de uso
- administrar las cuentas de usuario, las credenciales de seguridad, la autorización y el control de acceso
- rastrear el acceso interno y externo a los servicios arrendados
- planificar y evaluar el aprovisionamiento de recursos de TI
- planificar la capacidad

Mientras la interfaz de usuario provista por el sistema de administración remota tenderá a ser propiedad del proveedor de la nube, existe una preferencia entre los consumidores de la nube por trabajar con sistemas de administración remota que ofrecen APIs estandarizadas. Esto permite que un consumidor de la nube invierta en la creación de su propio front-end con el conocimiento previo de que puede reutilizar esta consola si decide cambiarse a otro proveedor de la nube que admite la misma API estandarizada. Además, el consumidor de la nube podría aprovechar aún más las APIs estandarizadas si está interesado en alquilar y administrar de forma centralizada los recursos de TI de múltiples proveedores de la nube y/o los recursos de TI que residen en la nube y en entornos locales.

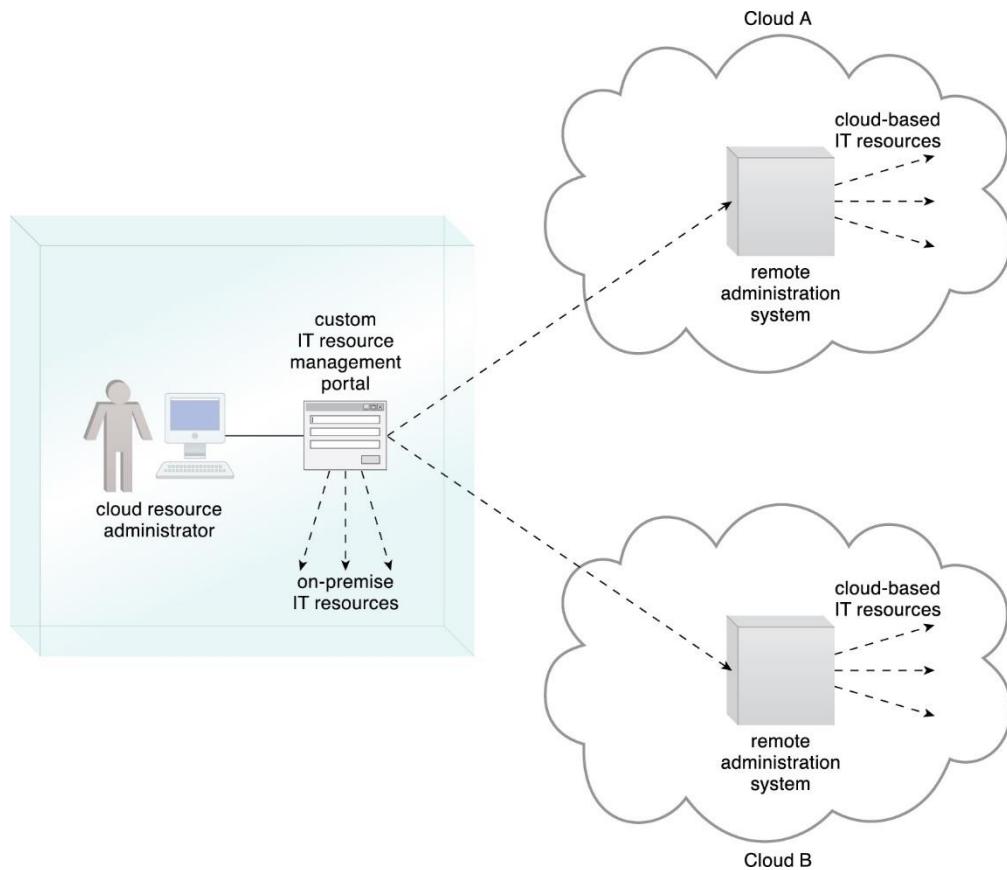


Figura 7.4 Las APIs estandarizadas publicadas por los sistemas de administración remota de diferentes nubes permiten a un consumidor de la nube desarrollar un portal personalizado que centraliza un único portal de administración de recursos de TI para los recursos de TI locales y basados en la nube.

Ejemplo de Estudio de Caso

DTGOV ha estado ofreciendo a sus consumidores de la nube un sistema de administración remota fácil de usar durante algún tiempo, y recientemente determinó que se requieren actualizaciones para adaptarse al creciente número de consumidores de la nube y la creciente diversidad de solicitudes. DTGOV está planificando un proyecto de desarrollo para extender el sistema de administración remota para cumplir con los siguientes requisitos:

- Los consumidores de la nube deben poder autoaprovisionarse de servidores virtuales y dispositivos de almacenamiento virtual. El sistema necesita específicamente interoperar con la API patentada de la plataforma VIM habilitada para la nube para habilitar las capacidades de autoaprovionamiento.
- Es necesario incorporar un mecanismo de inicio de sesión único (descrito en el siguiente Capítulo) para autorizar y controlar de forma centralizada el acceso de los consumidores a la nube.
- Es necesario exponer una API que admita el aprovisionamiento, el inicio, la detención, la liberación, la ampliación y la replicación de comandos para servidores virtuales y dispositivos de almacenamiento en la nube.

En apoyo de estas funciones, se desarrolla un portal de autoservicio y se amplía el conjunto de funciones del portal de uso y administración existente de DTGOV.

7.2. Sistema de gestión de recursos

El mecanismo del sistema de gestión de recursos ayuda a coordinar los recursos de TI en respuesta a las acciones de gestión realizadas tanto por los consumidores de la nube como por los proveedores de la nube (Figura 7.5). El núcleo de este sistema es el administrador de infraestructura virtual (VIM) que coordina el hardware del servidor para que se puedan crear instancias de servidor virtual en el servidor físico subyacente más conveniente. Un VIM es un producto comercial que se puede usar para administrar una variedad de recursos de TI virtuales en varios servidores físicos. Por ejemplo, un VIM puede crear y administrar varias instancias de un hipervisor en diferentes servidores físicos o asignar un servidor virtual en un servidor físico a otro (o a un pool de recursos).



Figura 7.5 Un sistema de administración de recursos que abarca una plataforma VIM y un repositorio de imágenes de máquinas virtuales. El VIM puede tener depósitos adicionales, incluido uno dedicado a almacenar datos operativos.

Las tareas que normalmente se automatizan e implementan a través del sistema de administración de recursos incluyen:

- administrar plantillas de recursos de TI virtuales que se utilizan para crear instancias prediseñadas, como imágenes de servidores virtuales
- asignar y liberar recursos de TI virtuales en la infraestructura física disponible en respuesta a el inicio, la pausa, la reanudación y la finalización de instancias de recursos de TI virtuales
- la coordinación de los recursos de TI en relación con la participación de otros mecanismos, como la replicación de recursos, el balanceador de carga y el sistema de failover
- la aplicación de políticas de uso y seguridad durante todo el ciclo de vida de las instancias de servicio en la nube
- monitorear las condiciones operativas de los recursos de TI

Los administradores de recursos de la nube empleados por el proveedor de la nube o el consumidor de la nube pueden acceder a las funciones del sistema de administración de recursos. Aquellos que trabajan en nombre de un proveedor de la nube a menudo podrán acceder directamente a la consola nativa del sistema de gestión de recursos.

Los sistemas de gestión de recursos suelen exponer APIs que permiten a los proveedores de la nube crear portales de sistemas de administración remota que se pueden personalizar para ofrecer controles de gestión de recursos de forma selectiva a los administradores de recursos de la nube externos que actúan en nombre de las organizaciones de consumidores de la nube a través de portales de uso y administración.

Ambas formas de acceso se describen en la Figura 7.6.

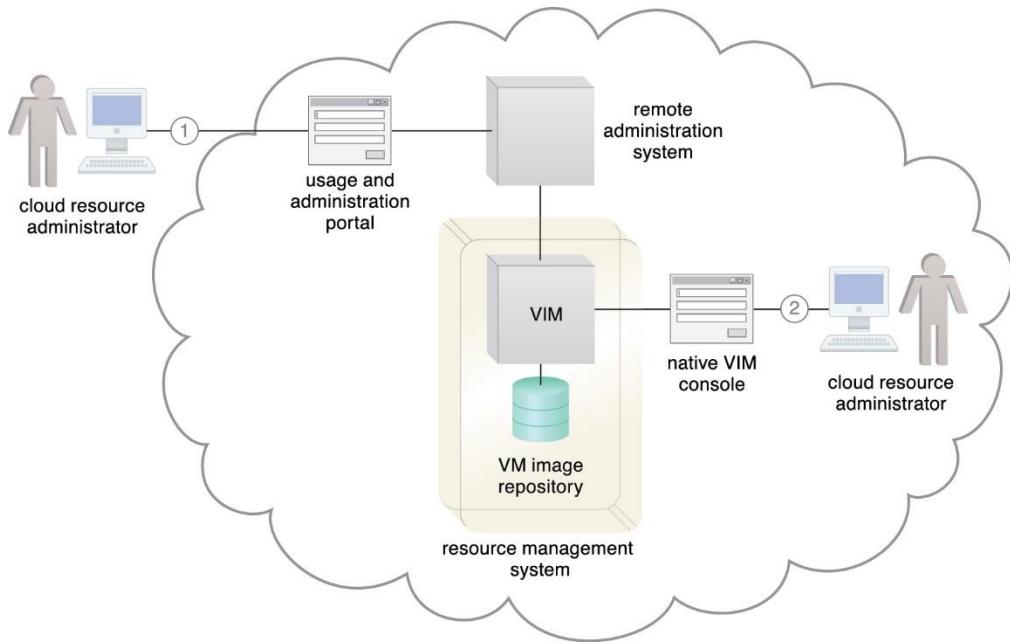


Figura 7.6 El administrador de recursos de nube de los consumidores de la nube accede a un portal de uso y administración de forma externa para administrar un recurso de TI alquilado (1). El administrador de recursos de la nube del proveedor de la nube utiliza la interfaz de usuario nativa proporcionada por el VIM para realizar tareas internas de administración de recursos (2).

Ejemplo de Estudio de Caso

El sistema de administración de recursos DTGOV es una extensión de un nuevo producto VIM que compró y proporciona las siguientes funciones principales:

- administración de recursos de TI virtuales con una asignación flexible de recursos de TI agrupados en diferentes centros de datos
- administración de bases de datos de consumidores en la nube
- aislamiento de recursos de TI virtuales en redes perimetrales lógicas
- administración de un inventario de imágenes de servidores virtuales de plantilla disponible para instantiación inmediata
- replicación automática (“snapshotting”) de imágenes de servidores virtuales para la creación de servidores virtuales
- escalamiento automático hacia arriba o hacia abajo de servidores virtuales de acuerdo a los umbrales de uso para permitir la migración de máquinas virtuales en vivo entre servidores físicos

- una API para la creación y administración de servidores virtuales y dispositivos de almacenamiento virtual
- una API para la creación de reglas de control de acceso a la red
- una API para la ampliación vertical de recursos de TI virtuales
- una API para la migración y la replicación de recursos de TI virtuales en varios centros de datos
- interoperación con un mecanismo de inicio de sesión único a través de una interfaz LDAP

Se implementan scripts de comando SNMP diseñados a medida para interoperar con las herramientas de administración de red para establecer redes virtuales aisladas en múltiples centros de datos.

7.3. Sistema de gestión de SLA

El mecanismo del sistema de gestión de SLA representa una gama de productos de gestión de la nube disponibles comercialmente que proporcionan funciones relacionadas con la administración, la recopilación, el almacenamiento, la generación de informes y la notificación en tiempo de ejecución de los datos de SLA (Figura 7.7).



Figura 7.7 Un sistema de gestión de SLA que incluye un administrador de SLA y un repositorio de mediciones de QoS.

La implementación de un sistema de gestión de SLA generalmente incluirá un repositorio utilizado para almacenar y recuperar datos de SLA recopilados en función de métricas predefinidas y parámetros de informes. Además, dependerá de uno o más mecanismos de monitoreo de SLA para recopilar los datos de SLA que luego pueden estar disponibles casi en tiempo real para los portales de uso y administración, para proporcionar comentarios continuos sobre los servicios de nube activos (Figura 7.8). Las métricas monitoreadas para servicios de nube individuales están alineadas con las garantías de SLA en los contratos correspondientes de aprovisionamiento de nube.

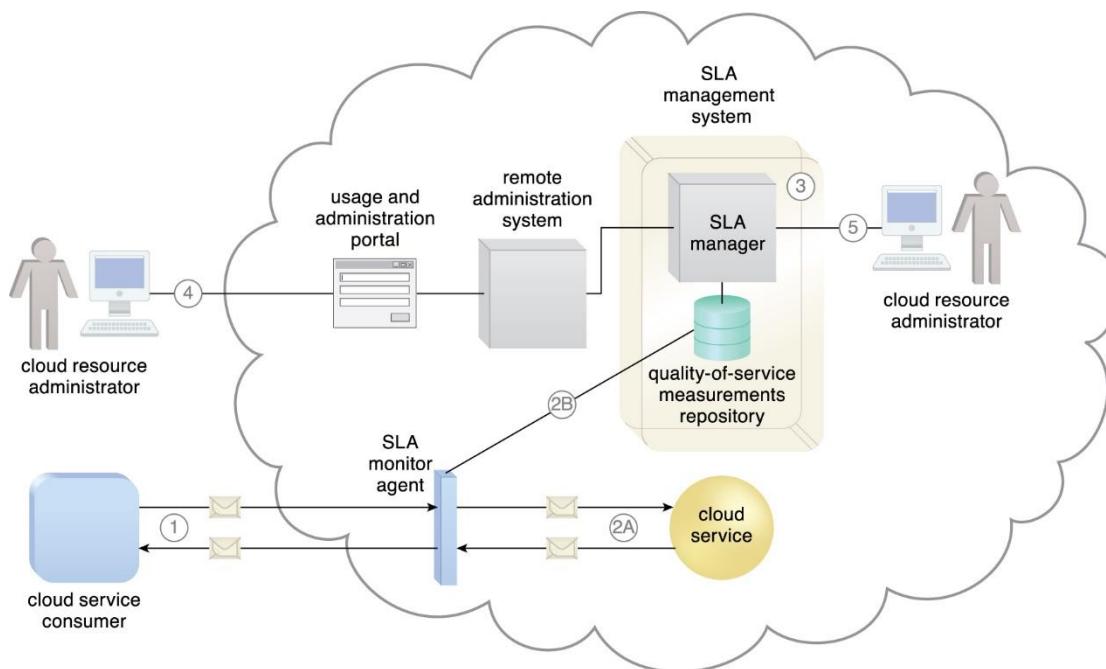


Figura 7.8 Un consumidor de servicios en la nube interactúa con un servicio en la nube (1). Un monitor de SLA intercepta los mensajes intercambiados, evalúa la interacción y recopila datos en tiempo de ejecución relevantes en relación con las garantías de calidad de servicio definidas en el SLA del servicio en la nube (2A). Los datos recopilados se almacenan en un repositorio (2B) que forma parte del sistema de gestión SLA (3). Se pueden emitir consultas y generar informes para un administrador de recursos de la nube externo a través de un portal de uso y administración (4) o para un administrador de recursos de la nube interno a través de la interfaz de usuario nativa del sistema de gestión de SLA (5).

Ejemplo de Estudio de Caso

DTGOV implementa un sistema de gestión de SLA que interactúa con su VIM existente. Esta integración permite a los administradores de recursos en la nube de DTGOV monitorear la disponibilidad de una variedad de recursos de TI alojados a través de monitores SLA.

DTGOV trabaja con los reportes del sistema de gestión de SLA para crear los siguientes informes predefinidos que están disponibles a través de paneles personalizados:

- *Panel de disponibilidad por centro de datos* - Accesible públicamente a través del portal corporativo en la nube de DTGOV, este panel muestra las condiciones operativas generales de cada grupo de recursos de TI en cada centro de datos, en tiempo real.
- *Panel de disponibilidad del consumidor por nube* - Este panel muestra las condiciones operativas en tiempo real de los recursos de TI individuales. Solo el proveedor de la nube y el consumidor de la nube que alquilan o son propietarios del recurso de TI pueden acceder a la información sobre cada recurso de TI.
- *Informe de SLA del consumidor por nube* - Este informe consolida y resume las estadísticas de SLA para los recursos de TI del consumidor de la nube, incluidos los tiempos de inactividad y otros eventos de SLA con marca de tiempo.

Los eventos de SLA generados por los monitores de SLA representan el estado y el rendimiento de los recursos de TI físicos y virtuales que están controlados por la plataforma de virtualización. El sistema de gestión de SLA interactúa con las herramientas de gestión de red a través de un agente de software SNMP diseñado a medida que recibe las notificaciones de eventos de SLA.

El sistema de gestión de SLA también interactúa con el VIM a través de su API patentada para asociar cada evento de SLA de red al recurso de TI virtual afectado. El sistema incluye una base de datos propietaria que se utiliza para almacenar eventos de SLA (como servidores virtuales y tiempos de inactividad de la red).

El sistema de gestión de SLA expone una API REST que DTGOV utiliza para interactuar con su sistema central de administración remota. La API propietaria tiene una implementación de servicio de componentes que se puede utilizar para el procesamiento por lotes con el sistema de gestión de facturación. DTGOV utiliza esto para proporcionar periódicamente datos de tiempo de inactividad que se traducen en crédito aplicado a las tarifas de uso de la nube del consumidor.

7.4. Sistema de Gestión de Facturación

El mecanismo del sistema de gestión de facturación está dedicado a la recopilación y el procesamiento de datos de uso relacionados con la contabilidad del proveedor de la nube y la facturación del consumidor de la nube. Específicamente, el sistema de gestión de facturación se basa en monitores de pago por uso para recopilar datos de uso de tiempo de ejecución que se almacenan en un repositorio del que luego extraen los componentes del sistema con fines de facturación, generación de informes y cobros (Figuras 7.9 y 7.10).

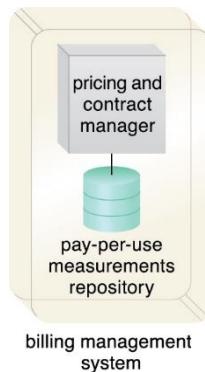


Figura 7.9 Un sistema de administración de facturación conformado por un administrador de precios y contratos y un repositorio de mediciones de pago por uso.

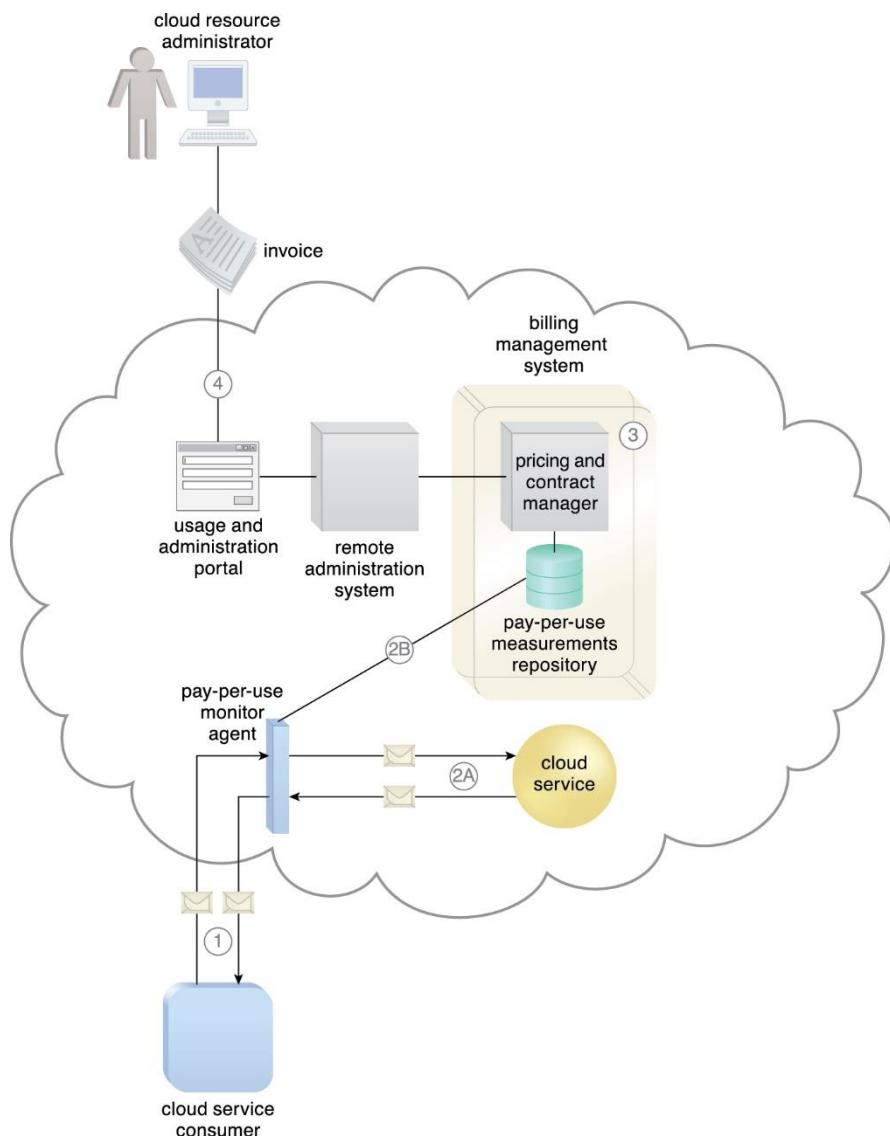


Figura 7.10 Un consumidor de servicios en la nube intercambia mensajes con un servicio en la nube (1). Un monitor de pago por uso realiza un seguimiento del uso y recopila datos relevantes para la facturación (2A), que se envían a un depósito que forma parte del sistema de gestión de facturación (2B). El sistema calcula periódicamente las tarifas consolidadas de uso del servicio en la nube y genera una factura para el consumidor de la nube (3). La factura se puede proporcionar al consumidor de la nube a través del portal de uso y administración (4).

El sistema de gestión de facturación permite la definición de diferentes políticas de precios, así como modelos de precios personalizados por consumidor de nube y/o por recurso de TI. Los modelos de precios pueden variar desde los modelos tradicionales de pago por uso hasta los modos de tarifa plana o pago por asignación, o combinaciones de estos.

Los arreglos de facturación se basarán en pagos previos al uso y posteriores al uso. Este último tipo puede incluir límites predefinidos o puede configurarse (con el acuerdo mutuo del consumidor de la nube) para permitir un uso ilimitado (y, en consecuencia, sin límite en la facturación posterior). Cuando se establecen límites, suelen ser en forma de cuotas de uso. Cuando se superan las cuotas,

el sistema de gestión de facturación puede bloquear más solicitudes de uso por parte de los consumidores de la nube.

Ejemplo de estudio de caso

DTGOV decide establecer un sistema de gestión de facturación que les permita crear facturas para eventos facturables definidos de forma personalizada, como suscripciones y uso de volumen de recursos de TI. El sistema de gestión de facturación se personaliza con los eventos necesarios y los metadatos del esquema de precios.

Incluye las siguientes dos bases de datos propietarias correspondientes:

- repositorio de eventos facturables
- repositorio de planes de precios

Los eventos de uso se recopilan por los monitores de pago por uso que se implementan como extensiones de la plataforma VIM. Los eventos de uso de granularidad delgada, como el inicio, la detención, el escalado vertical y el desmantelamiento del servidor virtual, se almacenan en un repositorio administrado por la plataforma VIM.

Además, los monitores de pago por uso suministran regularmente al sistema de gestión de facturación los eventos facturables apropiados. Se aplica un modelo de precios estándar a la mayoría de los contratos de consumidores en la nube, aunque se puede personalizar cuando se negocian términos especiales.

8 Cloud Security Mechanisms



Este capítulo establece un conjunto de mecanismos fundamentales de seguridad en la nube, varios de los cuales se pueden utilizar para contrarrestar las amenazas de seguridad descritas en el capítulo cuatro.

8.1 Encriptado

Los datos de forma predeterminada están codificados en un formato legible conocido como *texto plano*. Cuando se transmiten a través de una red, el texto plano es vulnerable al acceso no autorizado y potencialmente malicioso. El mecanismo de cifrado es un sistema de codificación digital dedicado a preservar la confidencialidad e integridad de los datos. Se utiliza para codificar datos de texto plano en un formato protegido e ilegible.

La tecnología de encriptación generalmente se basa en un algoritmo estandarizado llamado *cifrado* para transformar los datos de texto plano originales en datos encriptados, denominado texto cifrado (*ciphertext*). El acceso al texto cifrado no divulga los datos del texto plano original, ni tampoco algunas formas de metadatos, como la longitud del mensaje y la fecha de creación. Cuando se aplica el cifrado a los datos de texto plano, los datos se emparejan con una cadena de caracteres llamada *clave de cifrado*, un mensaje secreto que establecen y comparten las partes autorizadas. La clave de cifrado se utiliza para descifrar el texto cifrado y devolverlo a su formato de texto plano original.

El mecanismo de encriptación puede ayudar a contrarrestar las amenazas de seguridad de escuchas ilegales de tráfico, intermediarios maliciosos, autorización insuficiente y límites de confianza superpuestos. Por ejemplo, los agentes de servicios malintencionados que intentan espiar el tráfico no pueden descifrar los mensajes en tránsito si no tienen la clave de cifrado (Figura 8.1).

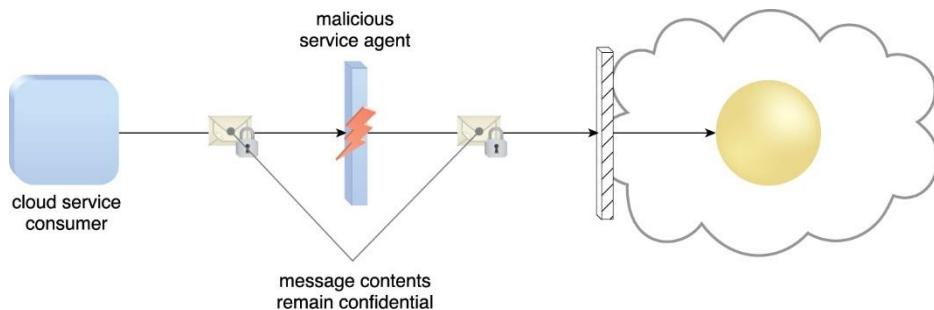


Figura 8.1 Un agente de servicio malintencionado no puede recuperar datos de un mensaje cifrado. Además, el intento de recuperación puede revelarse al consumidor del servicio en la nube. (Observe el uso del símbolo de candado para indicar que se ha aplicado un mecanismo de seguridad al contenido del mensaje).

Hay dos formas comunes de encriptación conocidas como encriptación (o cifrado) simétrica y encriptación asimétrica.

Cifrado simétrico

El cifrado simétrico usa la misma clave tanto para el cifrado como para el descifrado, ambos realizados por partes autorizadas que usan una clave compartida. También conocida como criptografía de clave secreta, los mensajes que están encriptados con una clave específica solo pueden descifrarse con esa misma clave. A las partes que legítimamente descifran los datos se les proporciona evidencia de que el cifrado original fue realizado por partes que legítimamente poseen la clave. Siempre se realiza una verificación de autenticación básica, porque solo las partes

autorizadas que poseen la clave pueden crear mensajes. Esto mantiene y verifica la confidencialidad de los datos.

Tenga en cuenta que el cifrado simétrico no tiene la característica de no repudio, ya que no es posible determinar exactamente qué parte realizó el cifrado o descifrado del mensaje si más de una parte está en posesión de la clave.

Cifrado asimétrico

El cifrado asimétrico se basa en el uso de dos claves diferentes, a saber, una clave privada y una clave pública. Con el cifrado asimétrico (que también se conoce como criptografía de clave pública), la clave privada solo la conoce su propietario, mientras que la clave pública está comúnmente disponible. Un documento que fue cifrado con una clave privada solo puede descifrarse correctamente con la clave pública correspondiente. En cambio, un documento que se haya cifrado con una clave pública solo se puede descifrar utilizando su contraparte de clave privada. Como resultado de que se utilizan dos claves diferentes en lugar de una sola, el cifrado asimétrico casi siempre es computacionalmente más lento que el cifrado simétrico.

El nivel de seguridad que se logra depende de si se utilizó una clave privada o una clave pública para cifrar los datos de texto plano. Como cada mensaje cifrado asimétricamente tiene su propio par de claves pública y privada, los mensajes cifrados con una clave privada pueden ser descifrados correctamente por cualquier parte con la clave pública correspondiente. Este método de cifrado no ofrece ninguna protección de confidencialidad, aunque el descifrado exitoso prueba que el texto fue cifrado por el propietario legítimo de la clave privada. Por lo tanto, el cifrado de clave privada ofrece protección de integridad además de autenticidad y no repudio. Un mensaje cifrado con una clave pública solo puede ser descifrado por el propietario legítimo de la clave privada, lo que brinda protección de confidencialidad. Sin embargo, cualquier parte que tenga la clave pública puede generar el texto cifrado, lo que significa que este método no brinda integridad del mensaje ni protección de la autenticidad debido a la naturaleza comunitaria de la clave pública.

Nota

El mecanismo de encriptación, cuando se usa para proteger las transmisiones de datos basadas en la Web, se aplica más comúnmente a través de HTTPS, que se refiere al uso de SSL/TLS como protocolo de encriptación subyacente para HTTP. TLS (Transport Layer Security) es la sucesora de la tecnología SSL (Secure Sockets Layer). Debido a que el cifrado asimétrico suele consumir más tiempo que el cifrado simétrico, TLS usa el primero solo para su método de intercambio de claves. Los sistemas TLS luego cambian a cifrado simétrico una vez que se han intercambiado las claves.

La mayoría de las implementaciones de TLS admiten principalmente RSA como el cifrado asimétrico principal, mientras que los cifrados como RC4, Triple-DES y AES son compatibles con el cifrado simétrico.

Ejemplo de Estudio de Caso

Innovartus se enteró recientemente de que los usuarios que acceden a su Portal de Registro de Usuarios a través de zonas activas Wi-Fi públicas y LAN no seguras pueden estar transmitiendo detalles del perfil de usuario personal a través de texto plano. Innovartus soluciona inmediatamente esta vulnerabilidad aplicando el mecanismo de cifrado a su portal web mediante el uso de HTTPS (Figura 8.2).

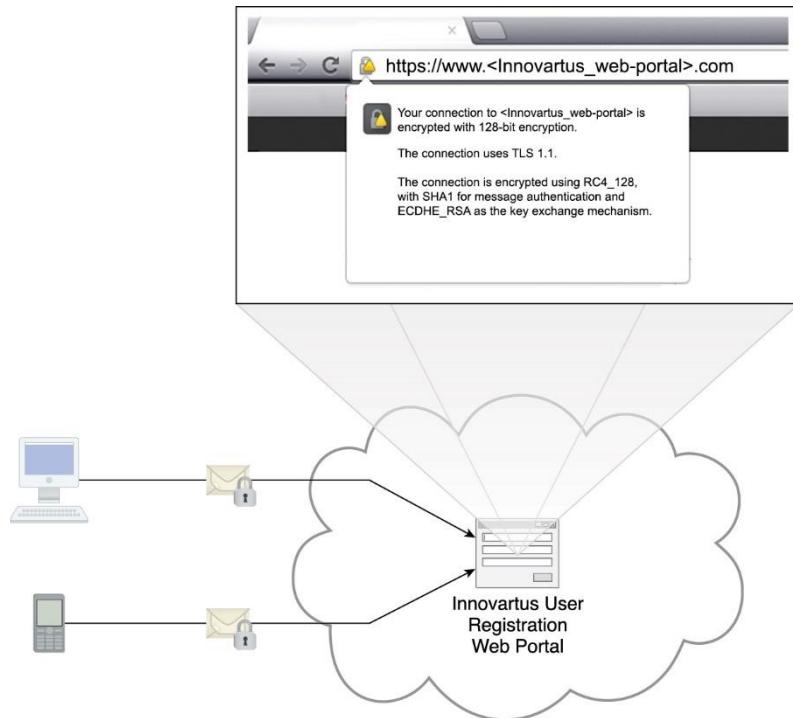


Figura 8.2 El mecanismo de encriptación es agregado al canal de comunicación entre los usuarios externos y el portal de registro de usuarios de Innovartus. Este resguarda la confidencialidad del mensaje mediante el uso de HTTPS.

8.2. Hashing

El mecanismo de hashing se utiliza cuando se requiere una forma de protección de datos unidireccional y no reversible. Una vez que se ha aplicado hash a un mensaje, queda encriptado y no se proporciona ninguna clave para desencriptar el mensaje. Una aplicación común de este mecanismo es el almacenamiento de contraseñas.

La tecnología hash se puede utilizar para衍生 un código hash o *message digest* de un mensaje, que a menudo tiene una longitud fija y es más pequeño que el mensaje original. Quien envía el mensaje puede entonces utilizar el mecanismo hash para adjuntar el message digest al mensaje. El destinatario aplica la misma función hash al mensaje para verificar que el message digest producido es idéntico al que acompaña al mensaje. Cualquier alteración de los datos originales da como resultado un message digest completamente diferente e indica claramente que se ha producido una manipulación.

Además de su uso para proteger los datos almacenados, las amenazas en la nube que pueden mitigarse mediante el mecanismo hash incluyen intermediarios maliciosos y autorización insuficiente. Un ejemplo de lo primero se ilustra en la figura 8.3.

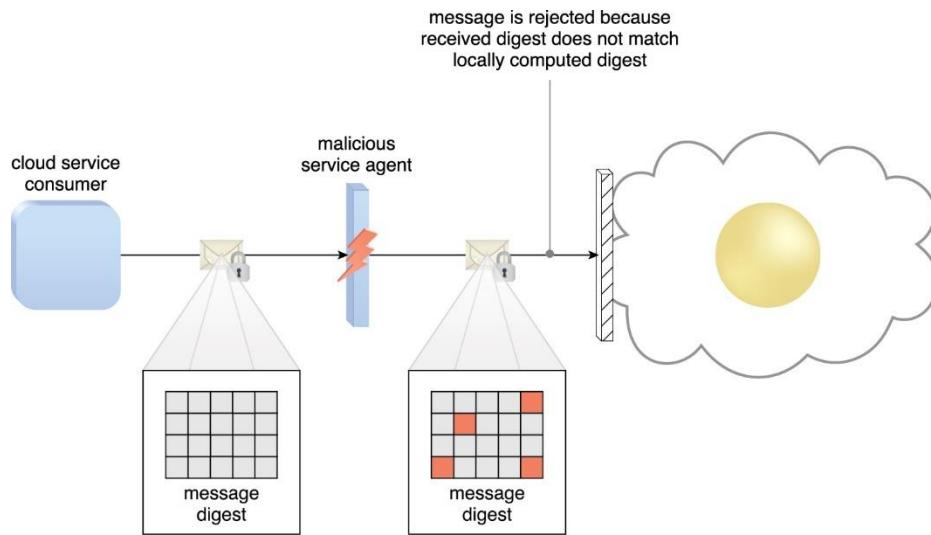


Figura 8.3 Se aplica una función hash para proteger la integridad de un mensaje que es interceptado y alterado por un agente de servicio malintencionado, antes de que se reenvíe. El cortafuegos se puede configurar para determinar que el mensaje ha sido alterado, lo que le permite rechazar el mensaje antes de que pueda continuar con el servicio en la nube.

Ejemplo de Estudio de Caso

Un subconjunto de las aplicaciones que han sido seleccionadas para ser portadas a la plataforma PaaS de ATN permite a los usuarios acceder y modificar datos corporativos altamente confidenciales. Esta información está alojada en una nube para permitir el acceso de socios confiables que pueden usarla para fines críticos de cálculo y evaluación. Preocupada de que los datos puedan ser manipulados, ATN decide aplicar el mecanismo de hashing como un medio para proteger y preservar la integridad de los datos.

Los administradores de recursos de la nube de ATN trabajan con el proveedor de la nube para incorporar un procedimiento de generación de digest con cada versión de la aplicación que se implementa en la nube. Los valores actuales se registran en una base de datos local segura y el procedimiento se repite regularmente con los resultados analizados. La Figura 8.4 ilustra cómo ATN implementa hashing para determinar si se han realizado acciones no autorizadas contra las aplicaciones portadas.

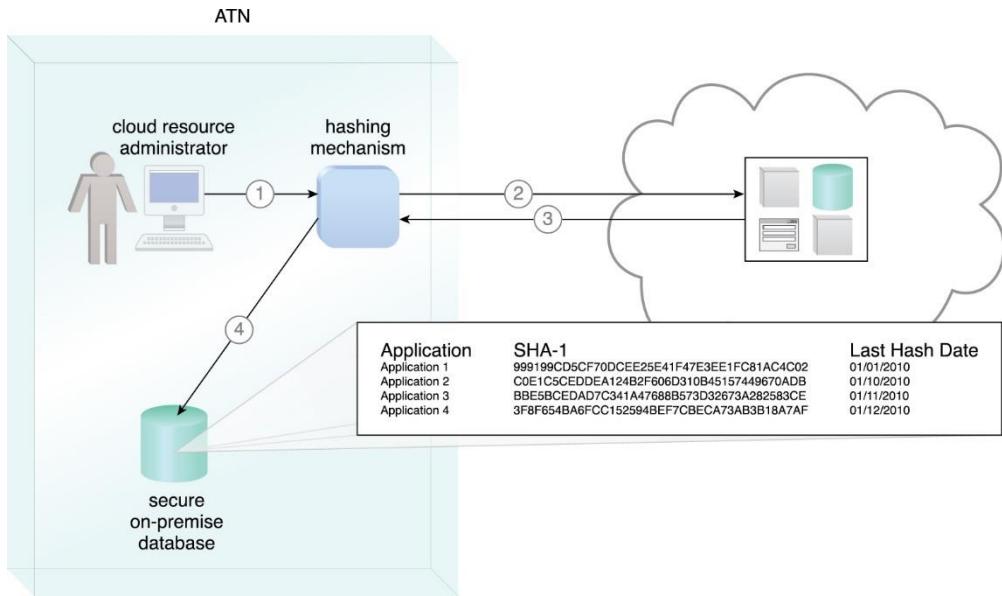


Figura 8.4 Se invoca un procedimiento hash cuando se accede al entorno PaaS (1). Las aplicaciones que se trasladaron a este entorno se comprueban (2) y se calculan sus message digest (3). Los message digest se almacenan en una base de datos local segura (4) y se emite una notificación si alguno de sus valores no es idéntico a los almacenados.

8.3. Digital Signature

El mecanismo de digital signature (firma digital) es un medio para proporcionar autenticidad e integridad de los datos a través de la autenticación y el no repudio. A un mensaje se le asigna una firma digital antes de la transmisión, que luego se invalida si el mensaje sufre modificaciones posteriores no autorizadas. Una firma digital proporciona evidencia de que el mensaje recibido es el mismo que el creado por su legítimo remitente.

Tanto el hash como el cifrado asimétrico están involucrados en la creación de una firma digital, que esencialmente existe como un message digest que fue cifrado por una clave privada y agregado al mensaje original. El destinatario verifica la validez de la firma y utiliza la clave pública correspondiente para descifrar la firma digital, lo que produce el message digest. El mecanismo hash también se puede aplicar al mensaje original para producir este message digest. Los resultados idénticos de los dos procesos diferentes indican que el mensaje mantuvo su integridad.

El mecanismo de firma digital ayuda a mitigar las amenazas de seguridad del intermediario malicioso, la autorización insuficiente y los límites de confianza superpuestos (Figura 8.5).

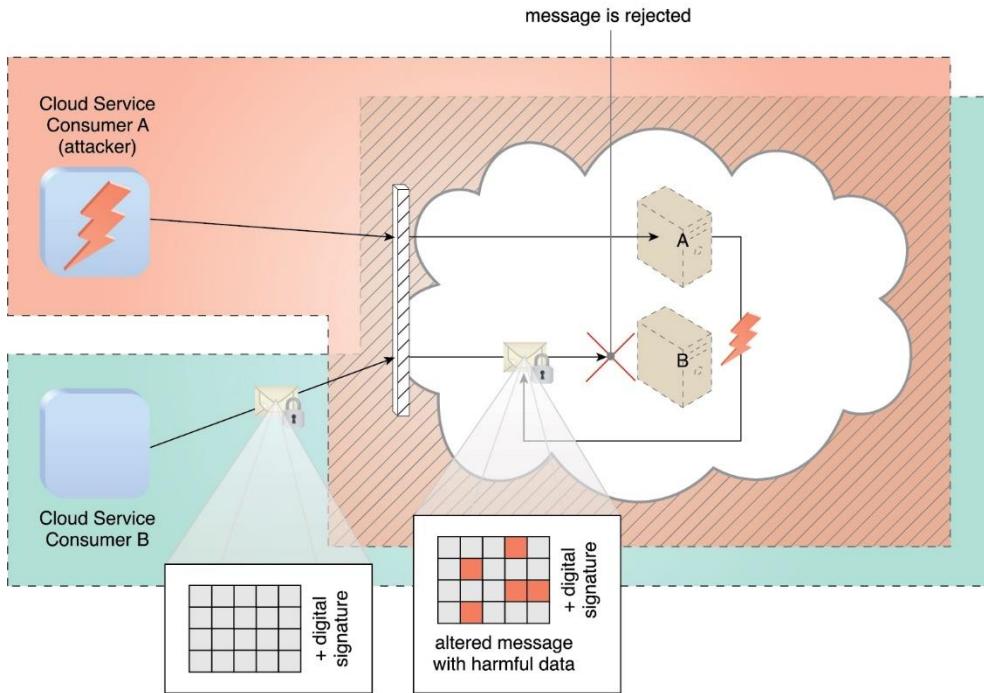


Figura 8.5 El Consumidor de Servicios en la Nube B envía un mensaje que fue firmado digitalmente pero que fue alterado por un atacante de confianza y Consumidor de Servicios en la Nube A. El Servidor Virtual B está configurado para verificar las firmas digitales antes de procesar los mensajes entrantes, incluso si están dentro de su límite de confianza. El mensaje se revela como ilegítimo debido a su firma digital no válida y, por lo tanto, es rechazado por el servidor virtual B.

Ejemplo de Estudio de Caso

Con la expansión de la cartera de clientes de DTGOV para incluir organizaciones del sector público, muchas de sus políticas de computación en la nube se han vuelto inadecuadas y requieren modificaciones. Teniendo en cuenta que las organizaciones del sector público manejan con frecuencia información estratégica, es necesario establecer salvaguardas de seguridad para proteger la manipulación de datos y establecer un medio para auditar las actividades que pueden afectar las operaciones del gobierno.

DTGOV procede a implementar el mecanismo de firma digital específicamente para proteger su entorno de gestión basado en la Web (Figura 8.6). El autoaprovigionamiento del servidor virtual dentro del entorno IaaS y la funcionalidad de seguimiento del SLA y la facturación en tiempo real se realizan a través de portales web. Como resultado, el error del usuario o las acciones maliciosas podrían tener consecuencias legales y financieras.

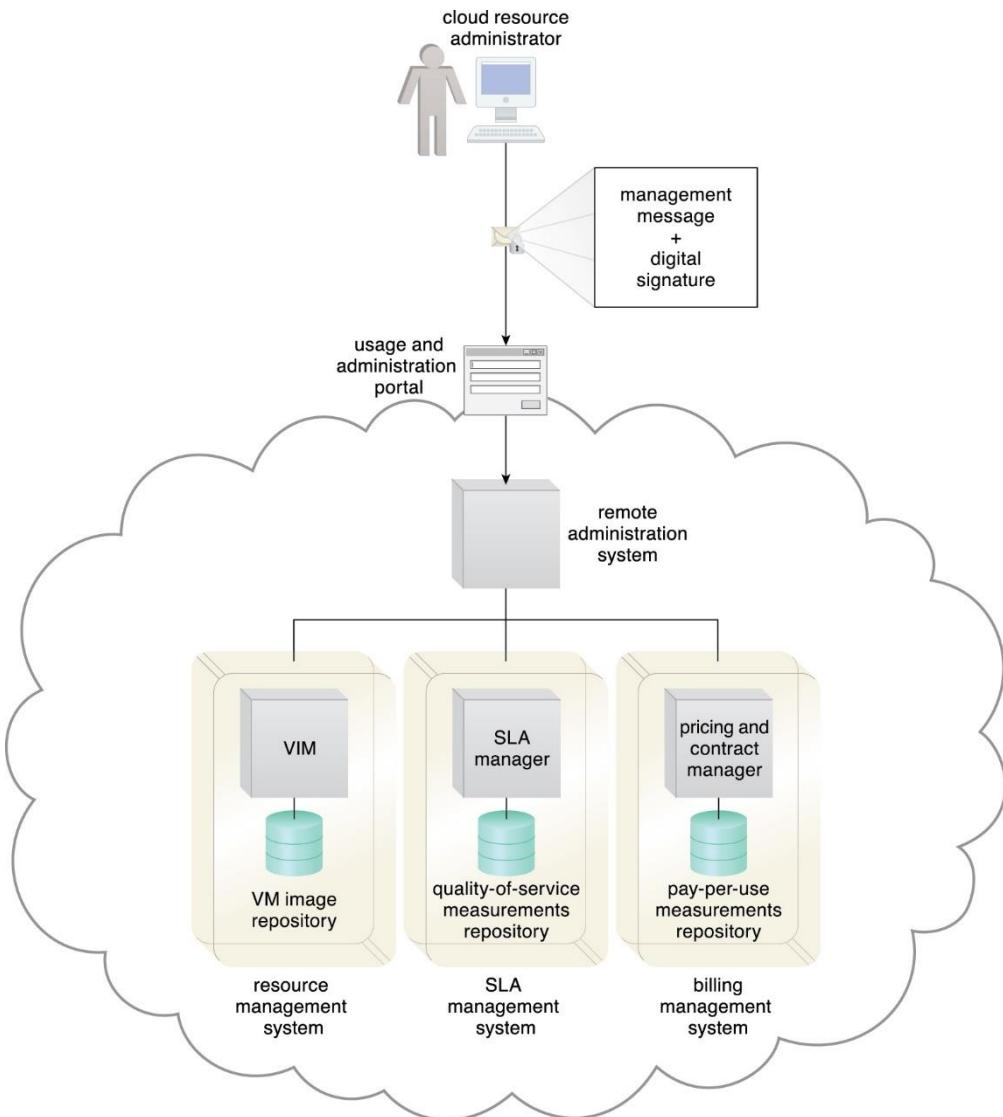


Figura 8.6 Siempre que un consumidor de la nube realice una acción de gestión relacionada con los recursos de TI provistos por DTGOV, el programa del consumidor del servicio de la nube debe incluir una firma digital en la solicitud del mensaje para demostrar la legitimidad de su usuario.

Las firmas digitales brindan a DTGOV la garantía de que cada acción realizada está vinculada a su legítimo autor. Se espera que el acceso no autorizado sea altamente improbable, ya que las firmas digitales solo se aceptan si la clave de cifrado es idéntica a la clave secreta que posee el propietario legítimo. Los usuarios no tendrán motivos para negar intentos de adulteración de mensajes porque las firmas digitales confirmarán la integridad del mensaje.

8.4. Infraestructura de clave pública (PKI)

Un enfoque común para gestionar la emisión de claves asimétricas se basa en la infraestructura de clave pública (*public key infrastructure PKI*), que existe como un sistema de protocolos, formatos de datos, reglas y prácticas que permiten que los sistemas a gran escala utilicen criptografía de clave pública de forma segura. El sistema se utiliza para asociar claves públicas con sus correspondientes

propietarios de claves (lo que se conoce como identificación de clave pública) al mismo tiempo que permite la verificación de la validez de la clave. PKI confía en el uso de certificados digitales, las cuales son estructuras de datos firmadas digitalmente que ligan las claves públicas a la identidad del propietario del certificado, así como a la información relacionada tales como los períodos de validez. Los certificados digitales generalmente están firmados digitalmente por una autoridad de certificación (CA) de terceros, como se ilustra en la Figura 8.7.

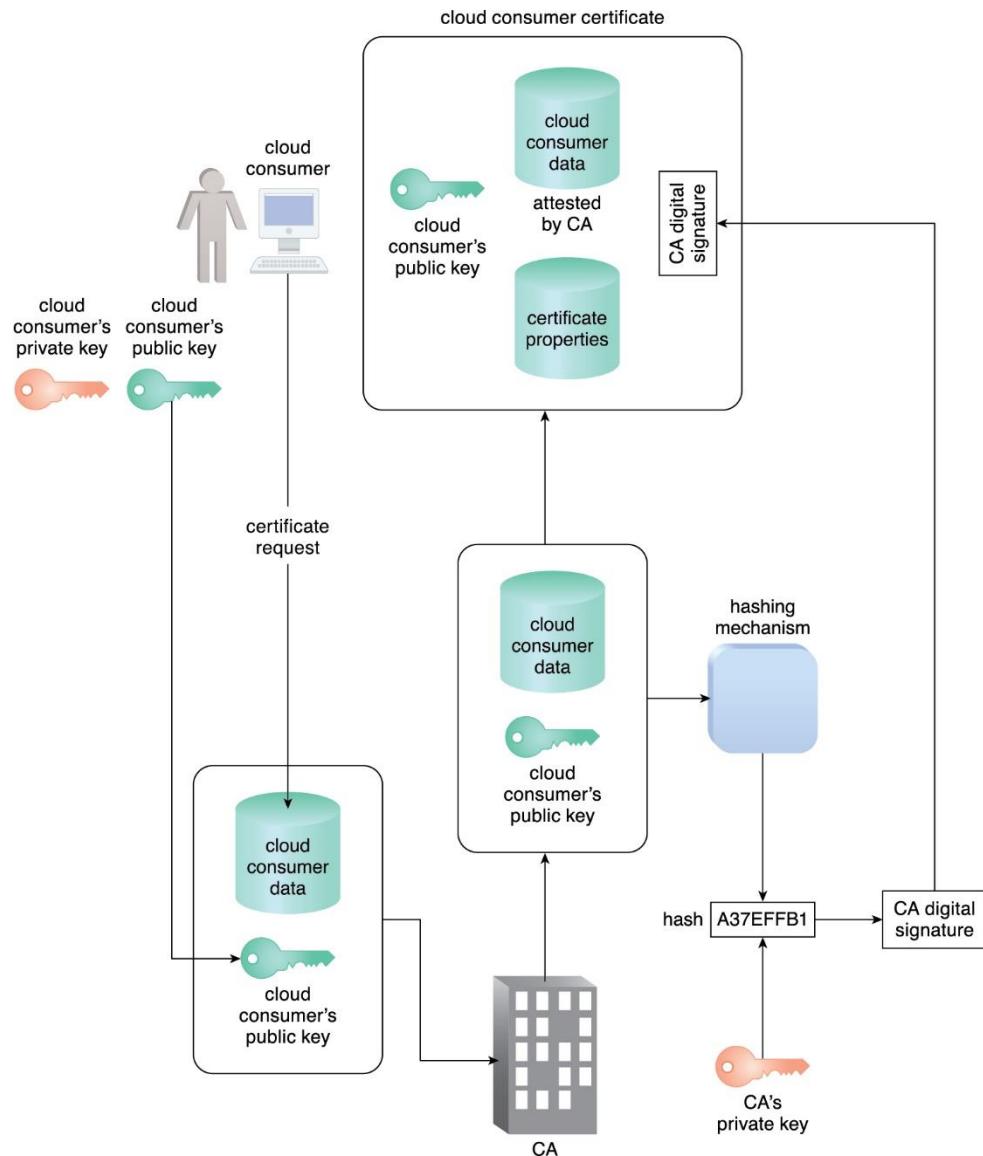


Figura 8.7 Los pasos comunes involucrados durante la generación de certificados por parte de una autoridad certificadora.

Se pueden emplear otros métodos para generar firmas digitales, aunque la mayoría de los certificados digitales son emitidos por solo un puñado de CAs confiables como VeriSign y Comodo. Las organizaciones más grandes, como Microsoft, pueden actuar como su propia CA y emitir certificados para sus clientes y el público, ya que incluso los usuarios individuales pueden generar certificados siempre que cuenten con las herramientas de software adecuadas.

Desarrollar un nivel aceptable de confianza para una CA requiere mucho tiempo, pero es necesario. Las medidas de seguridad rigurosas, las inversiones sustanciales en infraestructura y los procesos operativos estrictos contribuyen a establecer la credibilidad de una CA. Cuanto mayor sea su nivel de confianza y confiabilidad, más estimados y reputados serán sus certificados. La PKI es un método confiable para implementar el cifrado asimétrico, administrar la información de identidad del consumidor y del proveedor de la nube y ayudar a defenderse contra intermediarios maliciosos y amenazas por insuficiencias en la autorización.

El mecanismo PKI se utiliza principalmente para contrarrestar la amenaza de autorización insuficiente.

Ejemplo de Estudio de Caso

DTGOV requiere que sus clientes utilicen firmas digitales para acceder a su entorno de administración basado en la Web. Estos deben generarse a partir de claves públicas que han sido certificadas por una autoridad de certificación reconocida (Figura 8.8).

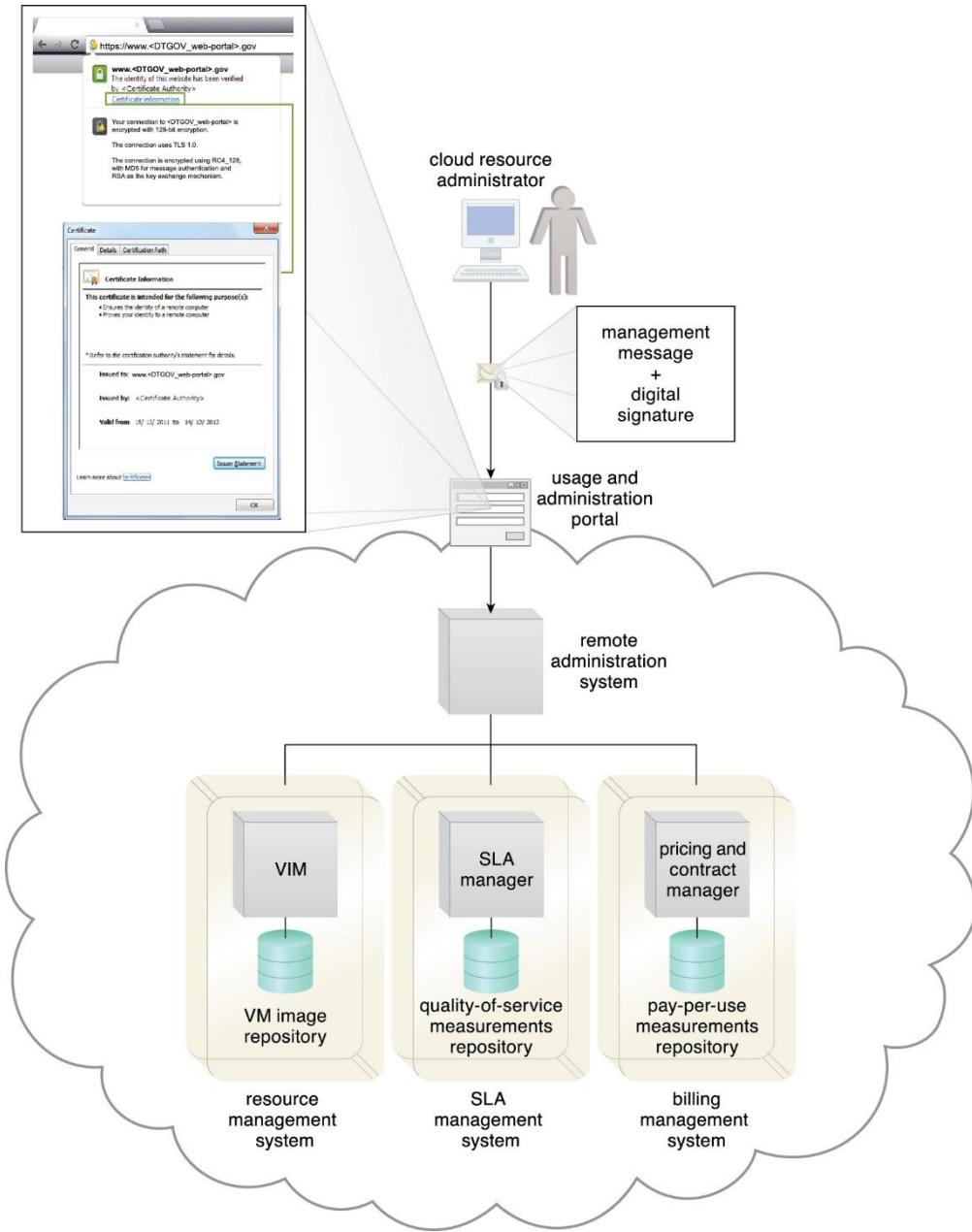


Figura 8.8 Un administrador de recursos de la nube externo utiliza un certificado digital para acceder al entorno de administración basado en la Web. El certificado digital de DTGOV se utiliza en la conexión HTTPS y luego lo firma una CA de confianza.

8.5. Identity and Access Management (IAM)

El mecanismo de administración de identidades y accesos (IAM) abarca los componentes y políticas necesarios para controlar y rastrear las identidades de los usuarios y los privilegios de acceso a los recursos, entornos y sistemas de TI.

Especificamente, los mecanismos de IAM existen como sistemas compuestos por cuatro componentes principales:

- *Autenticación*: Las combinaciones de nombre de usuario y contraseña siguen siendo las formas más comunes de credenciales de autenticación de usuario administradas por el sistema IAM, que también puede admitir firmas digitales, certificados digitales, hardware biométrico (lectores de huellas dactilares), software especializado (como programas de análisis de voz) y bloqueo de cuentas de usuario a direcciones IP o MAC registradas.
- *Autorización*: El componente de autorización define la granularidad correcta para los controles de acceso y supervisa las relaciones entre identidades, derechos de control de acceso y disponibilidad de recursos de TI.
- *Administración de usuarios*: Relacionado con las capacidades administrativas del sistema, el programa de administración de usuarios es responsable de crear nuevas identidades de usuarios y grupos de acceso, restablecer contraseñas, definir políticas de contraseñas y administrar privilegios.
- *Gestión de credenciales*: El sistema de gestión de credenciales establece identidades y reglas de control de acceso para cuentas de usuario definidas, lo que mitiga la amenaza de autorización insuficiente.

Aunque sus objetivos son similares a los del mecanismo PKI, el alcance de implementación del mecanismo IAM es distinto porque su estructura abarca controles y políticas de acceso además de asignar niveles específicos de privilegios de usuario.

El mecanismo IAM se utiliza principalmente para contrarrestar las amenazas de autorización insuficiente, denegación de servicio, superposición de límites de confianza, ataques de virtualización y ataques de contenedорización.

Ejemplo de Estudio de Caso

Como resultado de varias adquisiciones corporativas anteriores, el panorama heredado de ATN se ha vuelto complejo y altamente heterogéneo. Los costos de mantenimiento han aumentado debido a aplicaciones y bases de datos redundantes y similares que se ejecutan simultáneamente. Los repositorios heredados de credenciales de usuario son igualmente variados.

Ahora que ATN ha portado varias aplicaciones a un entorno PaaS, se crean y configuran nuevas identidades para otorgar acceso a los usuarios. Los consultores de CloudEnhance sugieren que ATN aproveche esta oportunidad iniciando una iniciativa piloto de sistema IAM, especialmente porque se necesita un nuevo grupo de identidades basadas en la nube.

ATN está de acuerdo y se diseña un sistema IAM especializado específicamente para regular los límites de seguridad dentro de su nuevo entorno PaaS. Con este sistema, las identidades asignadas a los recursos de TI basados en la nube difieren de las identidades locales correspondientes, que se definieron originalmente de acuerdo con las políticas de seguridad interna de ATN.

8.6. Single Sign-On (SSO)

Propagar la información de autenticación y autorización para un consumidor de servicios en la nube a través de múltiples servicios en la nube puede ser un desafío, especialmente si es necesario invocar numerosos servicios en la nube o recursos de TI basados en la nube como parte de la misma actividad general en tiempo de ejecución. El mecanismo de inicio de sesión único (SSO) permite que un consumidor de servicios en la nube sea autenticado por un agente de seguridad, lo que establece

un contexto de seguridad que persiste mientras el consumidor del servicio en la nube accede a otros servicios en la nube o recursos de TI basados en la nube. De lo contrario, el consumidor del servicio en la nube tendría que volver a autenticarse con cada solicitud posterior.

El mecanismo SSO esencialmente permite que los servicios en la nube y los recursos de TI sean independientes entre sí para generar y hacer circular credenciales de autorización y autenticación en tiempo de ejecución. Las credenciales proporcionadas inicialmente por el consumidor del servicio en la nube siguen siendo válidas durante una sesión, mientras se comparte su información de contexto de seguridad (Figura 8.9). El broker del mecanismo de seguridad SSO es especialmente útil cuando un consumidor de servicios en la nube necesita acceder a servicios en la nube que residen en diferentes nubes (Figura 8.10).

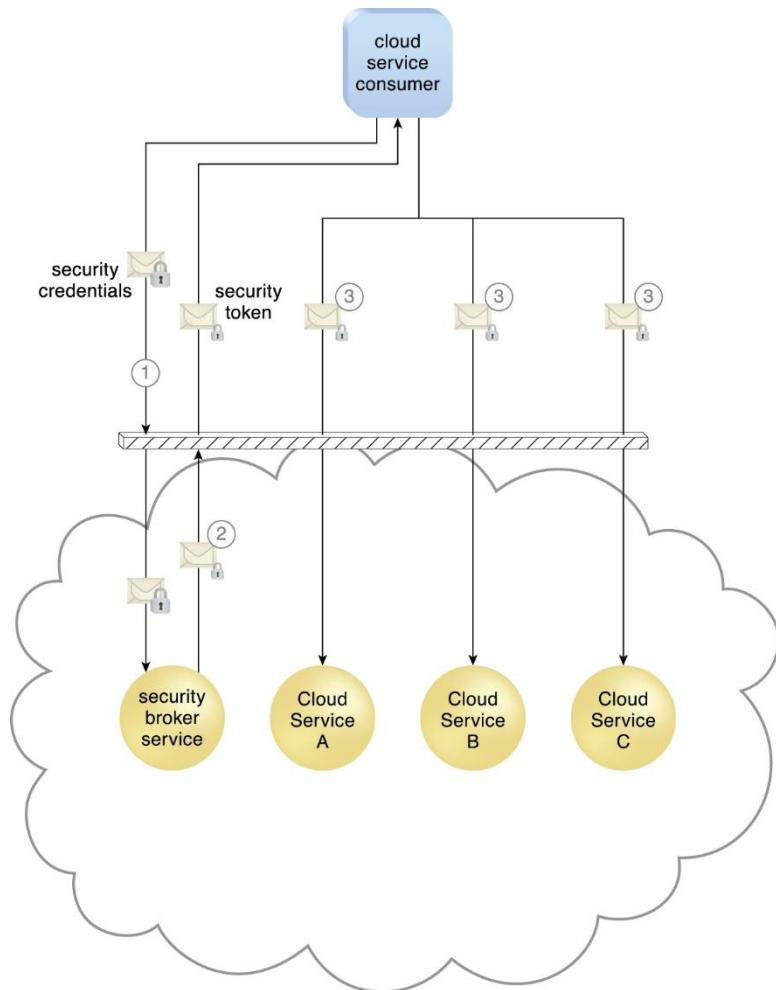


Figura 8.9 Un consumidor de servicios en la nube proporciona al broker de seguridad las credenciales de inicio de sesión (1). El broker de seguridad responde con un token de autenticación (mensaje con un pequeño símbolo de candado) después de la autenticación exitosa, que contiene información de identidad del consumidor del servicio en la nube (2) que se usa para autenticar automáticamente al consumidor del servicio en la nube en los Servicios A, B y C (3).

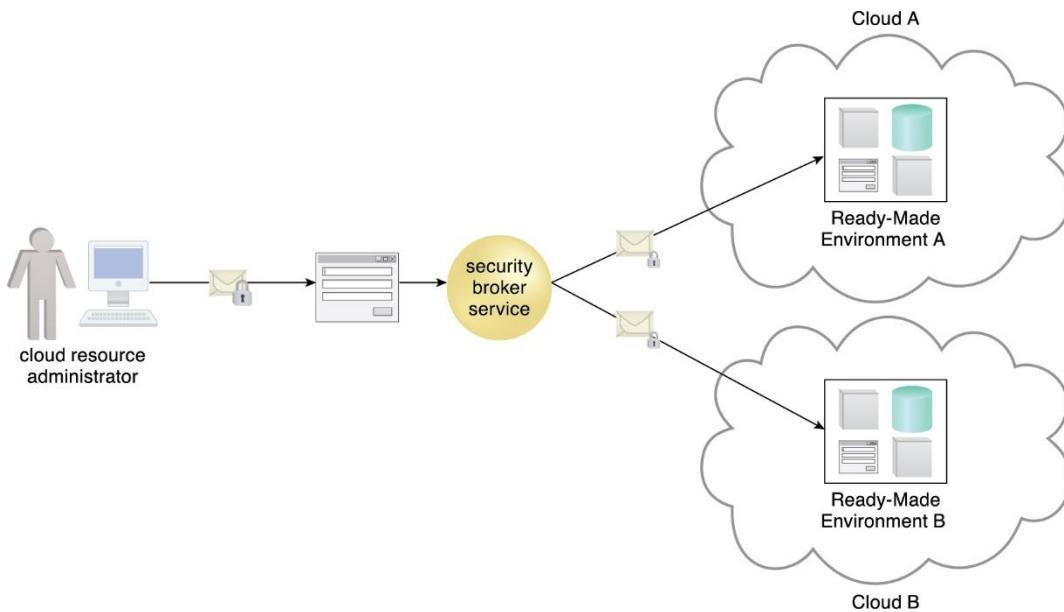


Figura 8.10 Las credenciales recibidas por el broker de seguridad se propagan a entornos ready-made a través de dos nubes diferentes. El broker de seguridad es el responsable de seleccionar el procedimiento de seguridad adecuado con el que contactar con cada nube.

Este mecanismo no contrarresta directamente ninguna de las amenazas de seguridad en la nube enumeradas en el Capítulo 4. Principalmente, mejora la facilidad de uso de los entornos basados en la nube para el acceso y la administración de soluciones y recursos de TI distribuidos.

Ejemplo de Estudio de Caso

La migración de aplicaciones a la nueva plataforma PaaS de ATN fue exitosa, pero también planteó una serie de nuevas inquietudes relacionadas con la capacidad de respuesta y la disponibilidad de los recursos de TI alojados en PaaS. ATN tiene la intención de trasladar más aplicaciones a una plataforma PaaS, pero decide hacerlo estableciendo un segundo entorno PaaS con un proveedor de nube diferente. Esto les permitirá comparar proveedores de nube durante un período de evaluación de tres meses.

Para adaptarse a esta arquitectura de nube distribuida, el mecanismo SSO se utiliza para establecer un agente de seguridad capaz de propagar las credenciales de inicio de sesión en ambas nubes (Figura 8.10). Esto permite que un solo administrador de recursos de la nube acceda a los recursos de TI en ambos entornos PaaS sin tener que iniciar sesión por separado en cada uno.

8.7. Grupos de seguridad basados en la nube

De manera similar a la construcción de diques que separan la tierra del agua, la protección de datos aumenta al colocar barreras entre los recursos de TI. La segmentación de recursos en la nube es un proceso mediante el cual se crean entornos de TI físicos y virtuales separados para diferentes usuarios y grupos. Por ejemplo, la WAN de una organización se puede dividir según los requisitos de seguridad de la red individual. Se puede establecer una red con un cortafuegos resistente para el acceso externo a Internet, mientras que una segunda se implementa sin un cortafuegos porque sus usuarios son internos y no pueden acceder a Internet.

La segmentación de recursos se utiliza para habilitar la virtualización al asignar una variedad de recursos de TI físicos a máquinas virtuales. Debe optimizarse para entornos de nube pública, ya que los límites de confianza organizacional de diferentes consumidores de nube se superponen cuando se comparten los mismos recursos de TI físicos subyacentes.

El proceso de segmentación de recursos basado en la nube crea mecanismos de *grupos de seguridad basados en la nube* que se determinan a través de políticas de seguridad. Las redes se segmentan en grupos de seguridad lógicos basados en la nube que forman perímetros de red lógicos. Cada recurso de TI basado en la nube se asigna al menos a un grupo de seguridad lógico basado en la nube. A cada grupo de seguridad lógico basado en la nube se le asignan reglas específicas que rigen la comunicación entre los grupos de seguridad.

Varios servidores virtuales que se ejecutan en el mismo servidor físico pueden convertirse en miembros de diferentes grupos de seguridad lógicos basados en la nube (Figura 8.11). Los servidores virtuales pueden dividirse además en grupos público-privado, grupos de desarrollo-producción o cualquier otra designación configurada por el administrador de recursos de la nube.

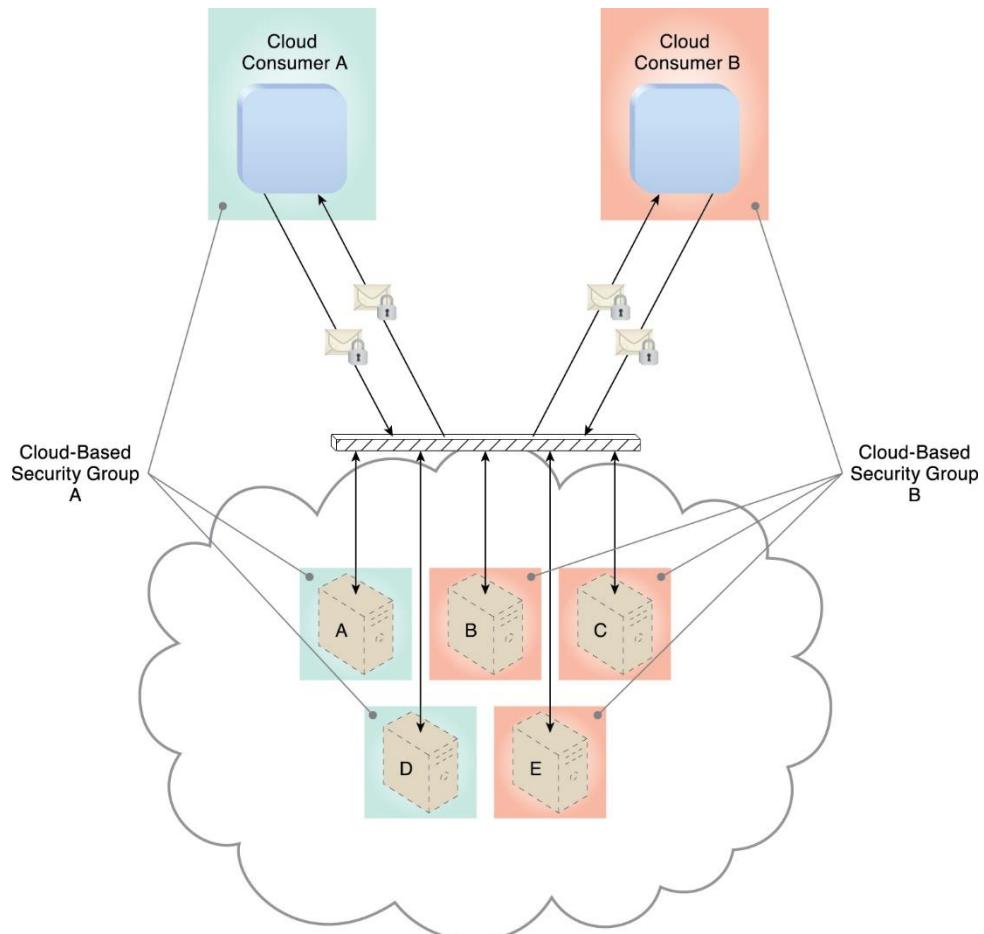


Figura 8.11 El Grupo de seguridad basado en la nube A abarca los Servidores virtuales A y D y está asignado al Consumidor en la nube A. El Grupo de seguridad basado en la nube B está compuesto por los Servidores virtuales B, C y E y está asignado al Consumidor en la nube B. Si las credenciales del consumidor del Servicio

en la nube A están comprometidas, el atacante solo podría acceder y dañar los servidores virtuales en el grupo de seguridad basado en la nube A, protegiendo así los servidores virtuales B, C y E.

Los grupos de seguridad basados en la nube delimitan áreas donde se pueden aplicar diferentes medidas de seguridad. Los grupos de seguridad basados en la nube correctamente implementados ayudan a limitar el acceso no autorizado a los recursos de TI en caso de una brecha de seguridad. Este mecanismo se puede utilizar para ayudar a contrarrestar la denegación de servicio, la autorización insuficiente, los límites de confianza superpuestos, los ataques de virtualización y las amenazas de ataques de contenedores, y está estrechamente relacionado con el mecanismo del perímetro de la red lógica.

Ejemplo de Estudio de Caso

Ahora que DTGOV se ha convertido en un proveedor de nube, surgen preocupaciones de seguridad relacionadas con su alojamiento de datos para clientes del sector público. Se contrata a un equipo de especialistas en seguridad en la nube para definir grupos de seguridad basados en la nube junto con los mecanismos de firma digital y PKI.

Las políticas de seguridad se clasifican en niveles de segmentación de recursos antes de integrarse en el entorno de gestión del portal web de DTGOV. De acuerdo con los requisitos de seguridad garantizados por sus SLAs, DTGOV mapea la asignación de recursos de TI al grupo de seguridad lógico basado en la nube apropiado (Figura 8.12), que tiene su propia política de seguridad que estipula claramente sus niveles de control y aislamiento de recursos de TI.

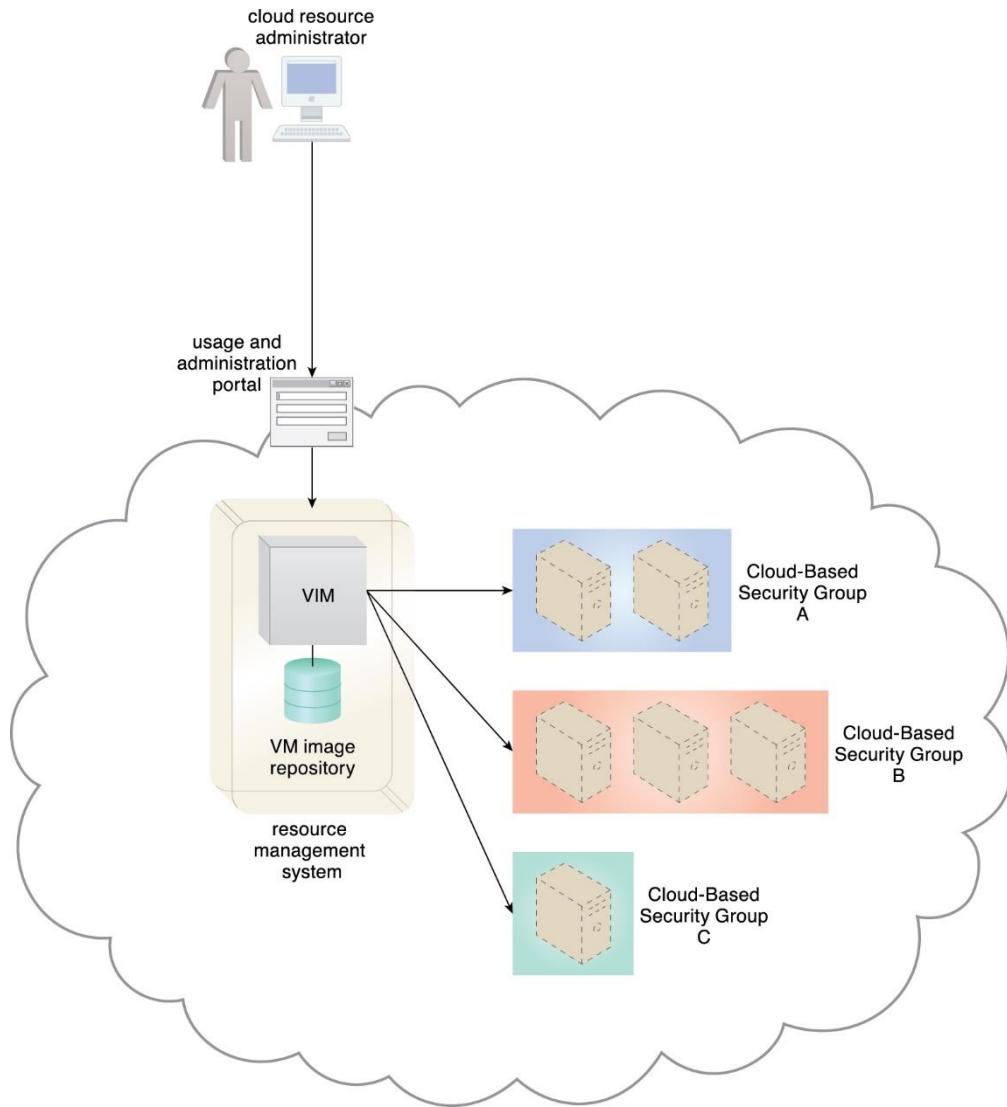


Figura 8.12 Cuando un administrador de recursos de la nube externo accede al portal web para asignar un servidor virtual, las credenciales de seguridad solicitadas se evalúan y asignan a una política de seguridad interna que asigna un grupo de seguridad basado en la nube correspondiente al nuevo servidor virtual.

DTGOV informa a sus clientes sobre la disponibilidad de estas nuevas políticas de seguridad. Los consumidores de la nube pueden optar opcionalmente por utilizarlos y, al hacerlo, se incrementan las tarifas.

8.8. Hardened Virtual Server Images

Como se mencionó anteriormente, un servidor virtual se crea a partir de una configuración de plantilla denominada imagen de servidor virtual (o imagen de máquina virtual). El hardening es el proceso de eliminar el software innecesario de un sistema para limitar las posibles vulnerabilidades que pueden explotar los atacantes. La eliminación de programas redundantes, el cierre de puertos de servidor innecesarios y la desactivación de servicios no utilizados, cuentas raíz internas y acceso de invitado son ejemplos de hardening.

Un *hardened virtual server image* es una plantilla para la creación de instancias de servicios virtuales que se ha sometido a un proceso de hardening (Figura 8.13). Esto generalmente da como resultado una plantilla de servidor virtual que es significativamente más segura que la imagen estándar original.

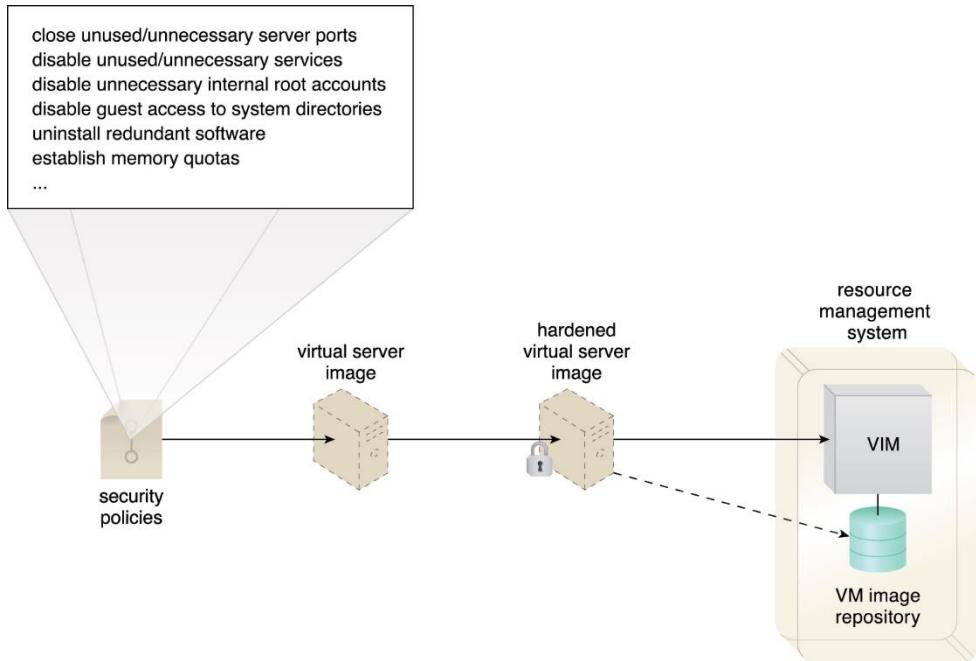


Figura 8.13 Un proveedor de nube aplica sus políticas de seguridad para fortalecer sus imágenes de servidor virtual estándar. La plantilla de imagen hardened se guarda en el repositorio de imágenes de VM como parte de un sistema de gestión de recursos.

Las imágenes de servidor virtual hardened ayudan a contrarrestar las amenazas de denegación de servicio, autorización insuficiente y superposición de límites de confianza.

Ejemplo de Estudio de Caso

Una de las funciones de seguridad que se hicieron disponibles para los consumidores de la nube como parte de la adopción de DTGOV de grupos de seguridad basados en la nube, es una opción para fortalecer algunos o todos los servidores virtuales dentro de un grupo determinado (Figura 8.14). Cada imagen de servidor virtual hardened genera una tarifa adicional, pero evita que los consumidores de la nube tengan que llevar a cabo el proceso de hardening ellos mismos.

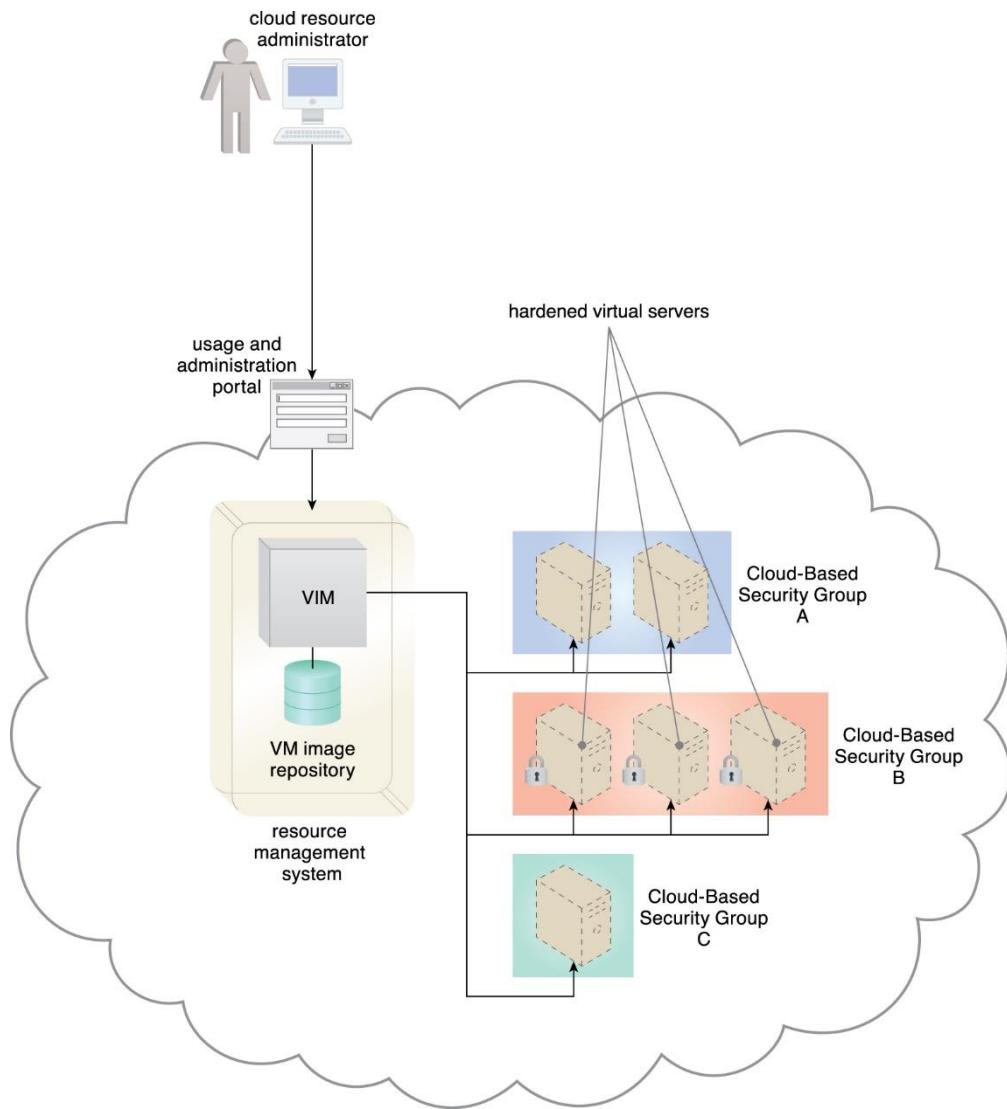


Figura 8.14 El administrador de recursos de la nube elige la opción de imagen de servidor virtual hardened para los servidores virtuales aprovisionados para el grupo de seguridad B basado en la nube.

9 Arquitecturas fundamentales de la nube



Este capítulo presenta y describe varios de los modelos arquitectónicos básicos de la nube más comunes, cada uno ejemplifica el uso que comúnmente se le da y que es una característica de los entornos basados en la nube contemporáneos. Se explora la implicación y la importancia de diferentes combinaciones de mecanismos de computación en la nube en relación con estas arquitecturas.

9.1. Arquitectura de distribución de carga de trabajo

Los recursos de TI se pueden escalar horizontalmente mediante la adición de uno o más recursos de TI idénticos y un balanceador de carga que proporciona una lógica en tiempo de ejecución capaz de distribuir uniformemente la carga de trabajo entre los recursos de TI disponibles (Figura 9.1). La arquitectura de distribución de la carga de trabajo resultante reduce tanto la sobreutilización como la infroutilización de los recursos de TI en una medida que depende de la sofisticación de los algoritmos de balanceo de carga y la lógica del tiempo de ejecución.

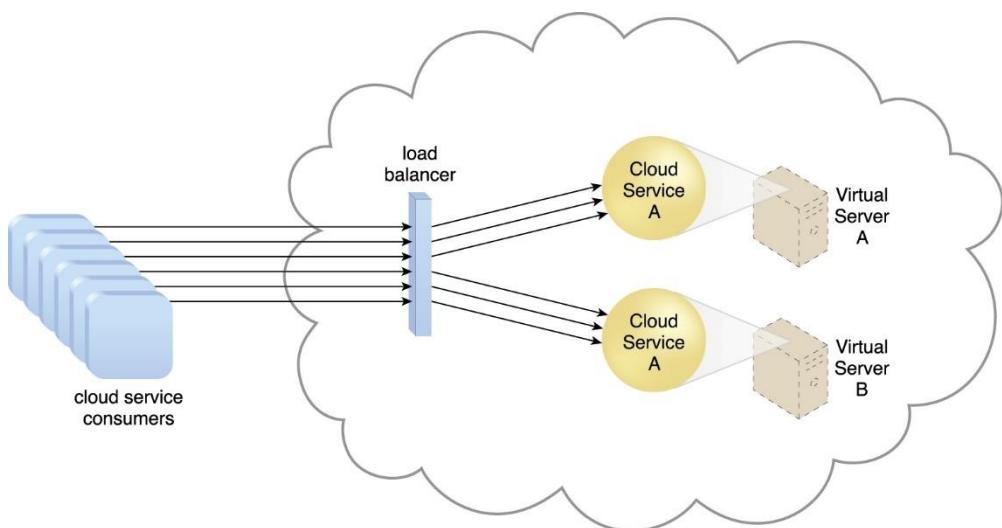


Figura 9.1 Se implementa una copia redundante del servicio en la nube A mediante el servidor virtual B. El balanceador de carga intercepta las solicitudes de los consumidores del servicio en la nube y las dirige a los servidores virtuales A y B para garantizar una distribución uniforme de la carga de trabajo.

Este modelo arquitectónico fundamental se puede aplicar a cualquier recurso de TI, y la distribución de la carga de trabajo suele llevarse a cabo en apoyo de servidores virtuales distribuidos, dispositivos de almacenamiento en la nube y servicios en la nube. Los sistemas de balanceo de carga aplicados a recursos de TI específicos suelen producir variaciones especializadas de esta arquitectura.

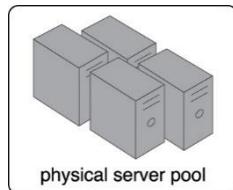
Además del mecanismo balanceador de carga, el servidor virtual y el mecanismo del dispositivo de almacenamiento en la nube a los que se puede aplicar el balanceo de carga, los siguientes mecanismos también pueden formar parte de esta arquitectura de nube:

- **Monitor de auditoría:** al distribuir cargas de trabajo en tiempo de ejecución, el tipo y la ubicación geográfica de los recursos de TI que procesan los datos pueden determinar si es necesario hacer un monitoreo para cumplir con los requisitos legales y reglamentarios.
- **Monitor de uso de la nube:** varios monitores pueden estar involucrados para llevar a cabo el seguimiento de la carga de trabajo en tiempo de ejecución y el procesamiento de datos.

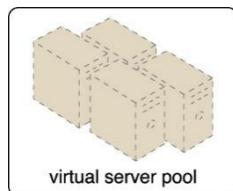
- *Hipervisor*: las cargas de trabajo entre los hipervisores y los servidores virtuales que alojan pueden requerir distribución.
- *Perímetro de la red lógica*: el perímetro de la red lógica aísla los límites de la red del consumidor de la nube en relación con cómo y dónde se distribuyen las cargas de trabajo.
- *Clúster de recursos*: los recursos de TI agrupados en modo activo/activo se usan comúnmente para respaldar el balanceo de carga de trabajo entre diferentes nodos de clúster.
- *Replicación de recursos*: este mecanismo puede generar nuevas instancias de recursos de TI virtualizados en respuesta a las demandas de distribución de la carga de trabajo en tiempo de ejecución.

9.2. Arquitectura de pooling de recursos

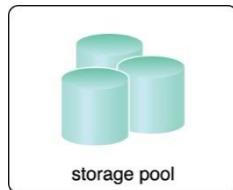
Una arquitectura de pooling de recursos se basa en el uso de uno o más pools de recursos, en los que los recursos de TI idénticos se agrupan y mantienen mediante un sistema que garantiza automáticamente que permanezcan sincronizados. A continuación, se proporcionan ejemplos comunes de grupos de recursos:



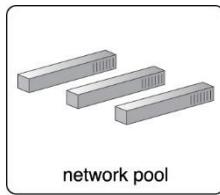
Un pool de servidores físicos está compuesto de servidores conectados en red que han sido instalados con sistemas operativos y otros programas necesarios y/o aplicaciones y están listos para su uso inmediato.



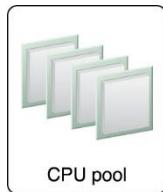
Un pool de servidores virtuales está usualmente configurado para utilizar una de varias plantillas disponibles el cual se selecciona por el consumidor de la nube durante el aprovisionamiento. Por ejemplo, un consumidor de la nube puede seleccionar un pool de servidores nivel medio con 4GB de RAM o un pool de servidores Ubuntu de nivel bajo con 2GB de RAM.



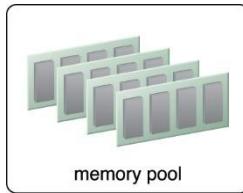
Un pool de almacenamiento, o pool de dispositivos de almacenamiento en la nube, consiste de estructuras de almacenamiento basados en archivos o en bloques que contienen dispositivos de almacenamiento en la nube vacíos o llenos.



Un pool de red o pool de interconexión está compuesto de distintas redes preconfiguradas para la conectividad entre dispositivos. Por ejemplo, un pool de dispositivos de firewall virtual o switches de red físicos puede ser creado para la conectividad redundante, balanceo de carga o adición de enlaces.



Un pool de CPU está listo para ser asignado a servidores virtuales, y son típicamente divididos en núcleos de procesamiento individual (core).



Un pool de RAM física se puede usar en servidores físicos recién aprovisionados o para escalar servidores físicos verticalmente.

Se pueden crear pools dedicados para cada tipo de recurso de TI y los pools individuales se pueden agrupar en un pool más grande, en cuyo caso cada pool individual se convierte en un sub-pool (Figura 9.2).

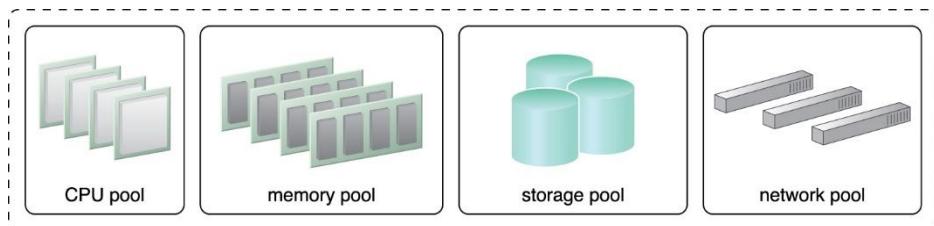


Figura 9.2 Un pool de recursos de muestra que se compone de cuatro sub-pools de CPU, memoria, dispositivos de almacenamiento en la nube y dispositivos de red virtual.

Los pools de recursos pueden volverse muy complejos, con múltiples pools creados para consumidores de nube específicos o aplicaciones. Se puede establecer una estructura jerárquica para pools principales, hermanos y anidados para facilitar la organización de diversos requisitos de pooling (Figura 9.3).

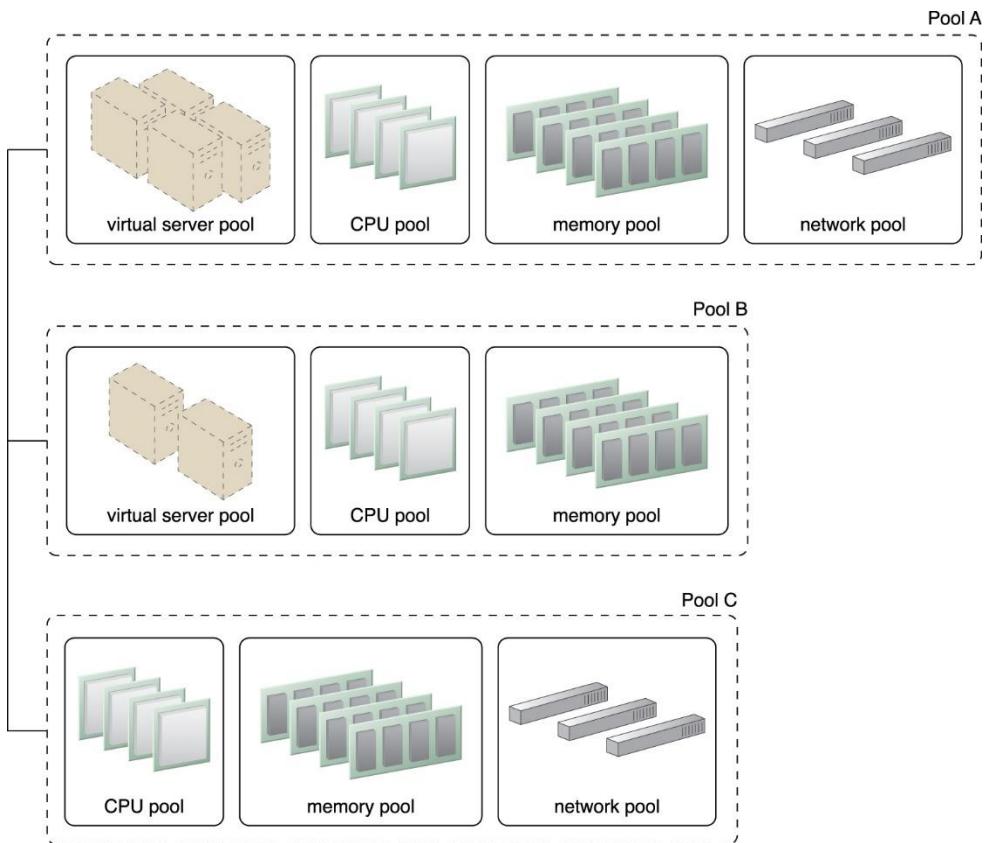


Figura 9.3 Los pools B y C son grupos hermanos que son tomados del pool A más grande, el cual se ha asignado a un consumidor de nube. Esta es una alternativa para tomar los recursos de TI para el Grupo B y el Grupo C de una reserva general de recursos de TI que se comparte en toda la nube.

Los pools de recursos hermanos generalmente se extraen de recursos de TI agrupados físicamente, a diferencia de los recursos de TI que se distribuyen en diferentes centros de datos. Los pools hermanos están aislados entre sí para que cada consumidor de la nube solo tenga acceso a su respectivo pool.

En el modelo de pools anidados, los pools más grandes se dividen en pools más pequeños que agrupan individualmente el mismo tipo de recursos de TI (Figura 9.4). Los grupos anidados se pueden usar para asignar pools de recursos a diferentes departamentos o grupos en la misma organización de consumidores de la nube.

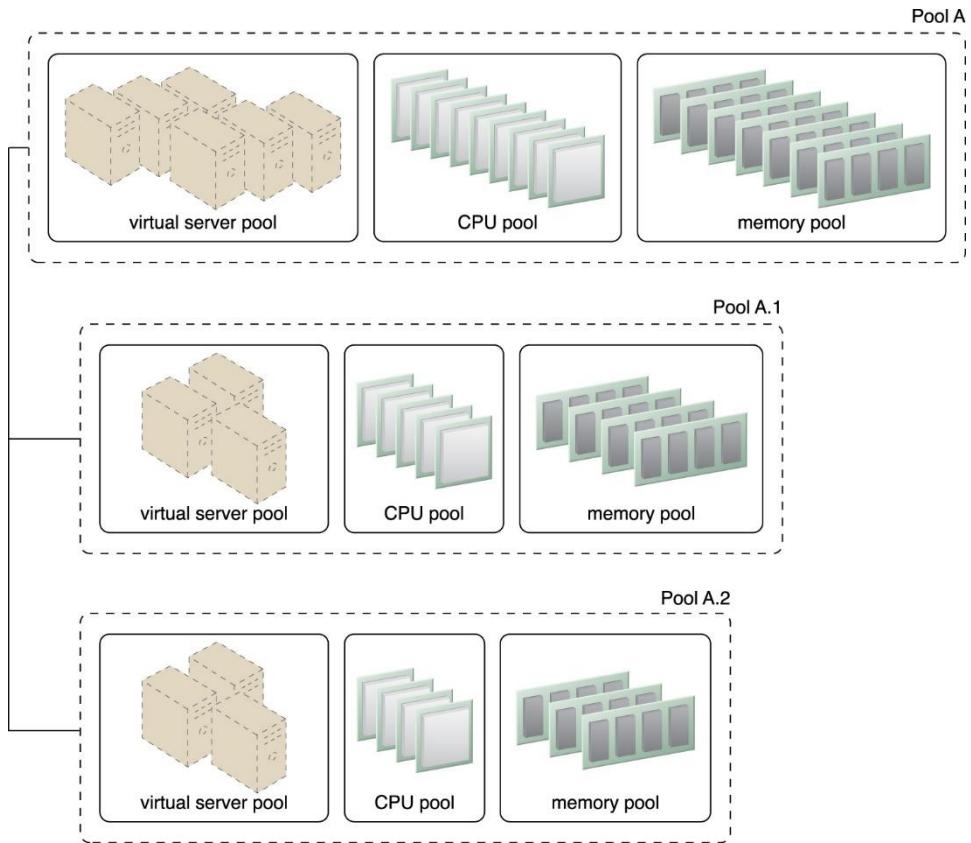


Figura 9.4 Los pools anidados A.1 y A.2 están compuestos por los mismos recursos de TI que el pool A, pero en cantidades diferentes. Los pools anidados generalmente se usan para aprovisionar servicios en la nube que necesitan instanciarse rápidamente usando el mismo tipo de recursos de TI con los mismos ajustes de configuración.

Una vez que se han definido los grupos de recursos, se pueden crear varias instancias de recursos de TI de cada pool para proporcionar un grupo en memoria de pools de TI “live”.

Además de los dispositivos de almacenamiento en la nube y los servidores virtuales, que suelen ser mecanismos pooled, los siguientes mecanismos también pueden formar parte de esta arquitectura en la nube:

- *Monitor de auditoría*: este mecanismo supervisa el uso del pool de recursos para garantizar el cumplimiento de los requisitos de privacidad y regulación, especialmente cuando los pools contienen dispositivos de almacenamiento en la nube o datos cargados en la memoria.
- *Monitor de uso de la nube*: varios monitores de uso de la nube están involucrados en el seguimiento y la sincronización del tiempo de ejecución que requieren los recursos de TI pooled y cualquier sistema de administración subyacente.
- *Hipervisor*: el mecanismo del hipervisor es responsable de proporcionar a los servidores virtuales acceso a los pools de recursos, además de hospedar los servidores virtuales y, a veces, los propios pools de recursos.

- *Perímetro de red lógica*: el perímetro de red lógica se utiliza para organizar y aislar pools de recursos de forma lógica.
- *Monitor de pago por uso*: el monitor de pago por uso recopila información de uso y facturación sobre cómo se asignan los consumidores de nube individuales y cómo usan los recursos de TI de varios pools.
- *Sistema de administración remota*: este mecanismo se usa comúnmente para interconectarse con sistemas y programas de back-end a fin de proporcionar funciones de administración de pools de recursos a través de un portal de front-end.
- *Sistema de administración de recursos*: el mecanismo del sistema de administración de recursos proporciona a los consumidores de la nube las herramientas y la administración de permisos y opciones para administrar pools de recursos.
- *Replicación de recursos*: este mecanismo se utiliza para generar nuevas instancias de recursos de TI para pools de recursos.

9.3. Arquitectura de escalabilidad dinámica

La arquitectura de escalabilidad dinámica es un modelo arquitectónico basado en un sistema de condiciones de escalamiento predefinidas que activan la asignación dinámica de recursos de TI de los pools de recursos. La asignación dinámica permite la utilización variable según lo dicten las fluctuaciones de la demanda de uso, ya que los recursos de TI innecesarios se recuperan de manera eficiente sin necesidad de interacción manual.

El escucha de escalado automatizado está configurado con umbrales de carga de trabajo que dictan cuándo es necesario agregar nuevos recursos de TI al procesamiento de la carga de trabajo. Este mecanismo se puede proporcionar con una lógica que determina cuántos recursos de TI adicionales se pueden proporcionar dinámicamente, según los términos de un contrato de aprovisionamiento de consumidores de nube determinado.

Los siguientes tipos de escalamiento dinámico se usan comúnmente:

- *Escalamiento horizontal dinámico*: Las instancias de recursos de TI hacen escalamiento in y out para manejar cargas de trabajo fluctuantes. El escucha de escalado automático supervisa las solicitudes y señala la replicación de recursos para iniciar la duplicación de recursos de TI, según los requisitos y permisos.
- *Escalamiento vertical dinámico*: Las instancias de recursos de TI hacen escalamiento up y down cuando es necesario ajustar la capacidad de procesamiento de un solo recurso de TI. Por ejemplo, un servidor virtual que se está sobrecargando puede tener su memoria aumentada dinámicamente o puede tener un núcleo de procesamiento agregado.
- *Reubicación dinámica*: El recurso de TI se reubica en un host con más capacidad. Por ejemplo, es posible que sea necesario mover una base de datos de un dispositivo de almacenamiento SAN basado en cinta con una capacidad de E/S de 4 GB por segundo a otro dispositivo de almacenamiento SAN basado en disco con una capacidad de E/S de 8 GB por segundo.

Las Figuras 9.5 a 9.7 ilustran el proceso de escalamiento horizontal dinámico.

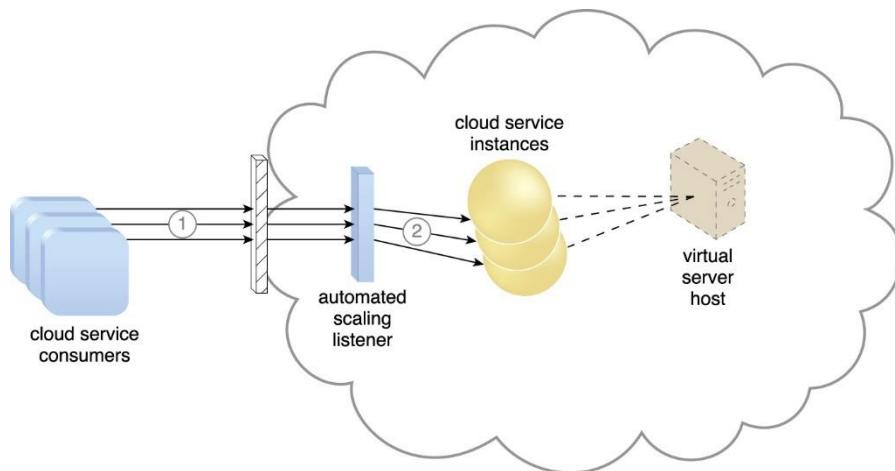


Figura 9.5 Consumidores del servicio cloud están enviando solicitudes a un servicio cloud (1). El escucha de escalamiento automatizado monitorea el servicio cloud para determinar si los límites de capacidad predefinidos están siendo excedidos (2).

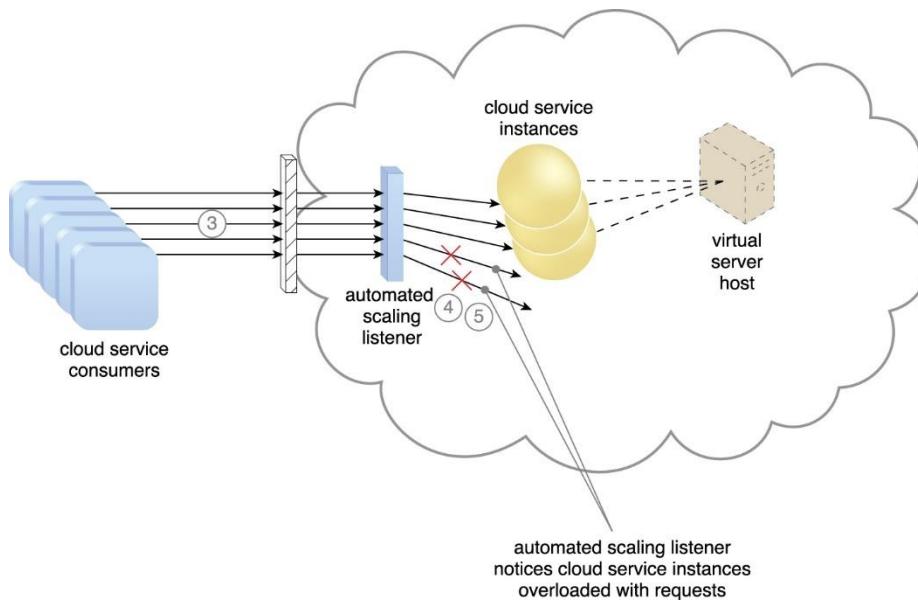


Figura 9.6 El número de solicitudes provenientes de consumidores de servicios en la nube aumenta (3). La carga de trabajo supera los umbrales de rendimiento. El escucha de escalado automatizado determina el siguiente curso de acción en función de una política de escalado predefinida (4). Si la implementación del servicio en la nube se considera elegible para un escalado adicional, el escucha de escalado automatizado inicia el proceso de escalado (5).

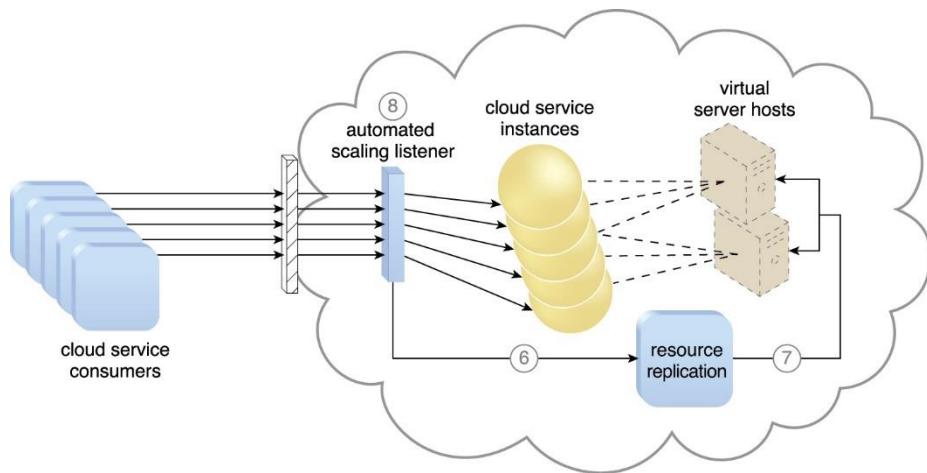


Figura 9.7 El escucha de escalado automatizado envía una señal al mecanismo de replicación de recursos (6), que crea más instancias del servicio en la nube (7). Ahora que se ha acomodado el aumento de la carga de trabajo, el escucha de escalado automatizado reanuda el monitoreo y quita o agrega recursos de TI, según sea necesario (8).

La arquitectura de escalabilidad dinámica se puede aplicar a una variedad de recursos de TI, incluidos servidores virtuales y dispositivos de almacenamiento en la nube. Además de los mecanismos básicos de replicación de recursos y escucha de escalado automatizado, los siguientes mecanismos también se pueden usar en esta forma de arquitectura de nube:

- **Monitor de uso de la nube:** Los monitores de uso de la nube especializados pueden rastrear el uso del tiempo de ejecución en respuesta a las fluctuaciones dinámicas causadas por esta arquitectura.
- **Hipervisor:** El hipervisor es invocado por un sistema de escalabilidad dinámica para crear o eliminar instancias de servidores virtuales, o para escalarse a sí mismo.
- **Monitor de pago por uso:** El monitor de pago por uso se utiliza para recopilar información sobre costos de uso en respuesta a la escalabilidad de los recursos de TI.

9.4. Arquitectura de capacidad de recursos elásticos

La arquitectura de capacidad de recursos elásticos se relaciona principalmente con el aprovisionamiento dinámico de servidores virtuales, utilizando un sistema que asigna y reclama CPU y RAM en respuesta inmediata a los requisitos de procesamiento fluctuantes de los recursos de TI alojados (Figuras 9.8 y 9.9).

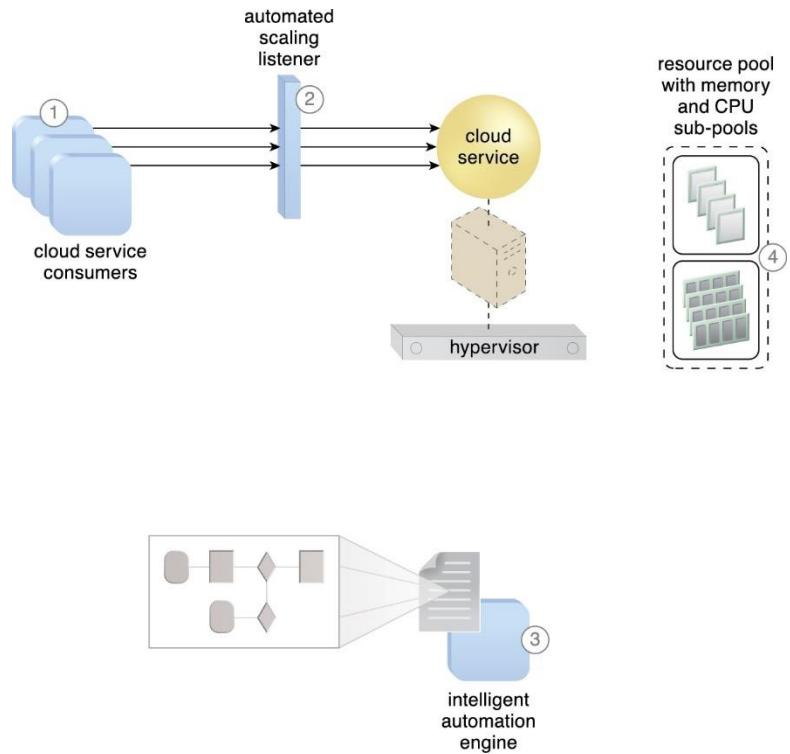


Figura 9.8 Los consumidores de servicios en la nube envían activamente solicitudes a un servicio en la nube (1), que son monitoreados por un oyente de escalado automatizado (2). Se implementa un motor de script de automatización inteligente con lógica de flujo de trabajo (3) que es capaz de notificar al grupo de recursos mediante solicitudes de asignación (4).

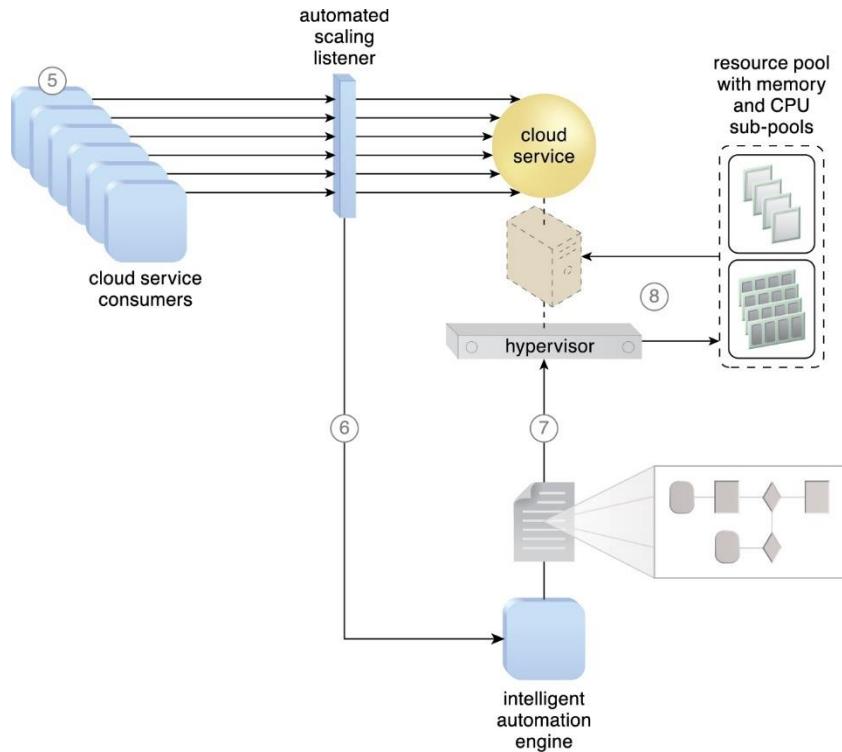


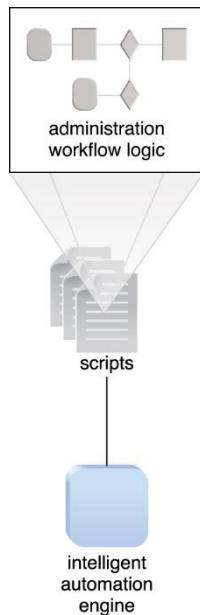
Figura 9.9 Las solicitudes de los consumidores de servicios en la nube aumentan (5), lo que hace que el oyente de escalado automatizado indique al motor de automatización inteligente que ejecute el script (6). El script ejecuta la lógica del flujo de trabajo que indica al hipervisor que asigne más recursos de TI de los grupos de recursos (7). El hipervisor asigna CPU y RAM adicionales al servidor virtual, lo que permite manejar el aumento de la carga de trabajo (8).

Los pools de recursos se utilizan mediante tecnología de escalado que interactúa con el hipervisor y/o VIM para recuperar y devolver recursos de CPU y RAM en tiempo de ejecución. El procesamiento en tiempo de ejecución del servidor virtual se supervisa para que se pueda aprovechar la potencia de procesamiento adicional del pool de recursos a través de la asignación dinámica, antes de que se alcancen los umbrales de capacidad. El servidor virtual y sus aplicaciones alojadas y los recursos de TI se escalan verticalmente en respuesta.

Este tipo de arquitectura en la nube se puede diseñar para que el script del motor de automatización inteligente envíe su solicitud de escalado a través del VIM en lugar de al hipervisor directamente. Es posible que sea necesario reiniciar los servidores virtuales que participan en sistemas de asignación de recursos elásticos para que la asignación dinámica de recursos surta efecto.

Motor de automatización inteligente

El motor de automatización inteligente automatiza las tareas de administración mediante la ejecución de secuencias de scripts que contienen la lógica de flujo de trabajo.



Algunos mecanismos adicionales que se pueden incluir en esta arquitectura de nube son los siguientes:

- *Monitor de uso de la nube*: Los monitores de uso de la nube especializados recopilan información sobre el uso de los recursos de TI antes, durante y después del escalado, para ayudar a definir los umbrales de capacidad de procesamiento futuros de los servidores virtuales.
- *Monitor de pago por uso*: El monitor de pago por uso es responsable de recopilar información sobre el costo del uso de los recursos a medida que estos fluctúan con el aprovisionamiento elástico.
- *Replicación de recursos*: Este modelo arquitectónico utiliza la replicación de recursos para generar nuevas instancias de los recursos de TI escalados.

9.5. Arquitectura del servicio de balanceo de carga

La arquitectura del servicio de balanceo de carga se puede considerar una variación especializada de la arquitectura de distribución de carga de trabajo que está diseñada específicamente para escalar implementaciones de servicios en la nube. Se crean implementaciones redundantes de servicios en la nube, con un sistema de balanceo de carga agregado para distribuir dinámicamente las cargas de trabajo.

Las implementaciones de servicios en la nube duplicados se organizan en un pool de recursos, mientras que el balanceador de carga se posiciona como un componente externo o integrado para permitir que los servidores host balanceen las cargas de trabajo por sí mismos.

Dependiendo de la carga de trabajo anticipada y la capacidad de procesamiento de los entornos de servidores host, se pueden generar múltiples instancias de cada implementación de servicios en la nube como parte de un grupo de recursos que responde a los volúmenes de solicitudes fluctuantes de manera más eficiente.

El balanceador de carga se puede colocar independientemente de los servicios en la nube y sus servidores host (Figura 9.10) o integrado como parte del entorno de la aplicación o del servidor. En el último caso, un servidor principal con la lógica de balanceo de carga puede comunicarse con los servidores vecinos para equilibrar la carga de trabajo (Figura 9.11).

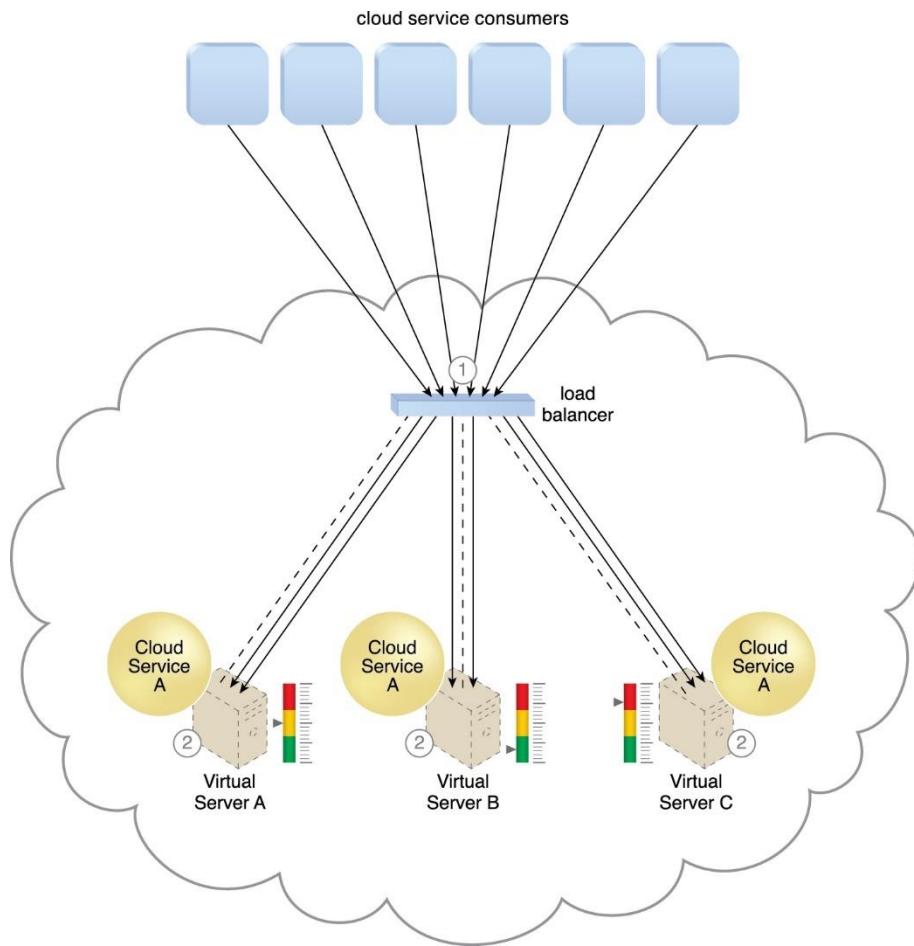


Figura 9.10 El balanceador de carga intercepta mensajes enviados por los consumidores del servicio cloud (1) y los envía a los servidores virtuales de modo que el procesamiento de la carga de trabajo es escalada horizontalmente (2).

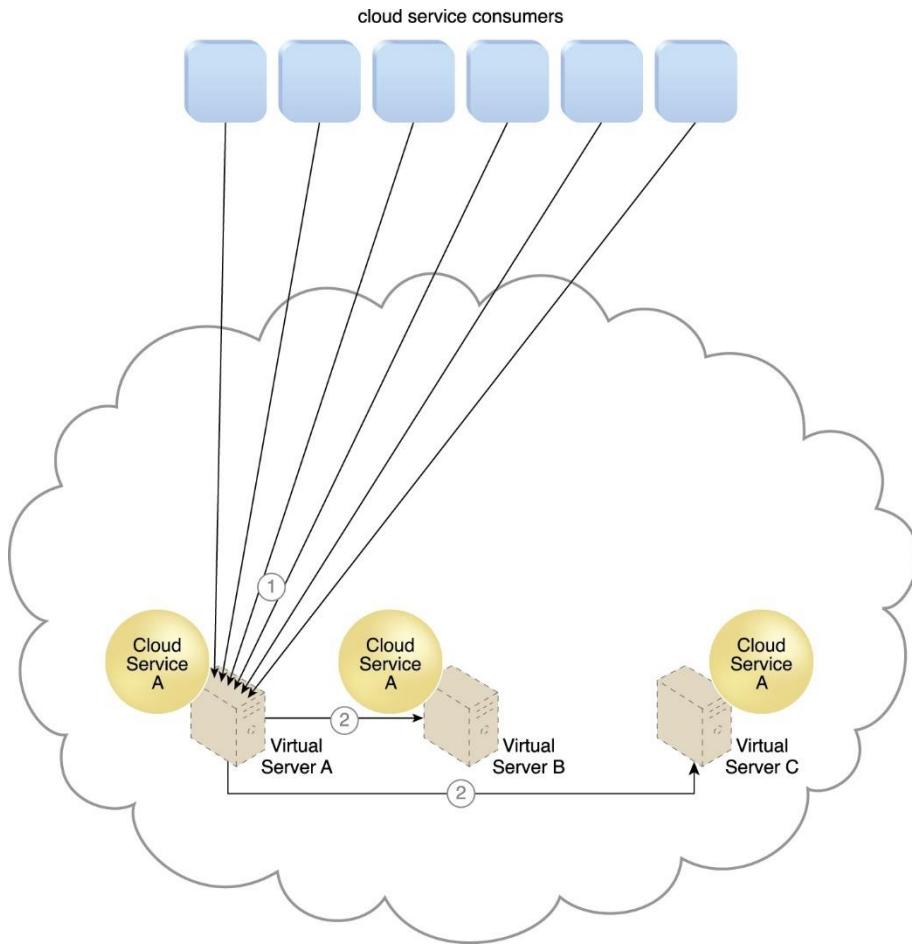


Figura 9.11 Las solicitudes de los consumidores de servicios en la nube se envían al servicio en la nube A en el servidor virtual A (1). La implementación del servicio en la nube incluye una lógica de balanceo de carga integrada que es capaz de distribuir solicitudes a las implementaciones vecinas del Servicio en la nube A en los Servidores virtuales B y C (2).

La arquitectura del servicio de balanceo de carga puede involucrar los siguientes mecanismos además del balanceador de carga:

- *Monitor de uso de la nube*: Los monitores de uso de la nube pueden estar involucrados en el monitoreo de las instancias del servicio en la nube y sus respectivos niveles de consumo de recursos de TI, así como varios datos de uso y monitoreo en tiempo de ejecución y uso de datos colectados de las tareas.
- *Clúster de recursos*: Los grupos de clúster activo-activo se incorporan en esta arquitectura para ayudar a balancear las cargas de trabajo entre los diferentes miembros del clúster.
- *Replicación de recursos*: El mecanismo de replicación de recursos se utiliza para generar implementaciones de servicios en la nube en apoyo de los requisitos de balanceo de carga.

9.6. Arquitectura de Cloud Bursting

La arquitectura de cloud bursting (nube saturada) establece una forma de escalamiento dinámico que escala o “bursts out” (expande) nuestros recursos de TI locales (on-premise) en una nube cada

vez que se alcanzan los umbrales de capacidad predefinidos. Los recursos de TI basados en la nube correspondientes se implementan previamente de forma redundante, pero permanecen inactivos hasta que se rebasa la capacidad de la nube. Una vez que ya no son necesarios, los recursos de TI basados en la nube se liberan y la arquitectura "burst in"(contrae) de nuevo en el entorno local (on-premise).

Cloud bursting es una arquitectura de escalamiento flexible que provee a los consumidores de la nube con la opción de usar recursos de TI basados en la nube solo para satisfacer demandas de mayor uso. La base de este modelo arquitectónico se basa en el escucha de escalado automatizado y los mecanismos de replicación de recursos.

El escucha de escalado automatizado determina cuándo redirigir las solicitudes a los recursos de TI basados en la nube, y la replicación de recursos se usa para mantener la sincronía entre los recursos de TI locales y los basados en la nube en relación con la información de estado (Figura 9.12).

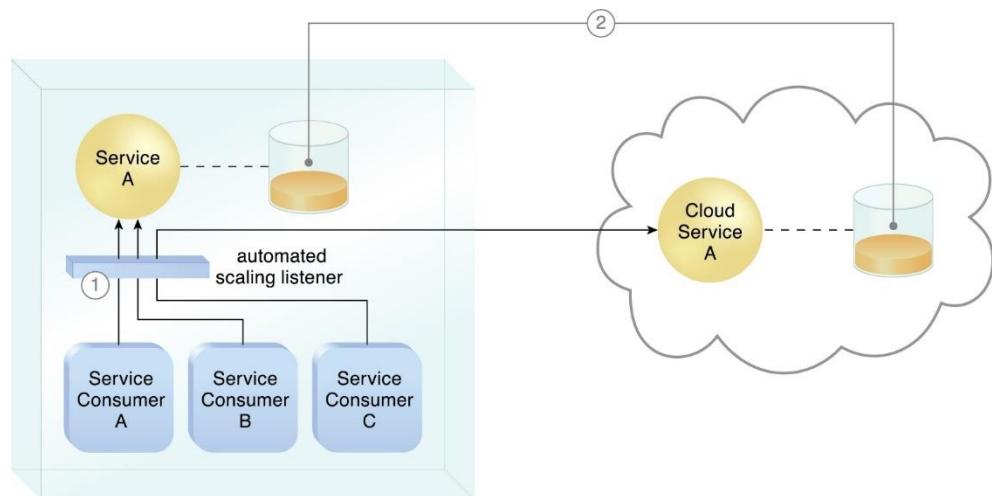


Figura 9.12 Un escucha de escalado automatizado supervisa el uso del Servicio A local y redirige la solicitud del Consumidor del servicio C al Servicio A como implementación redundante en la nube (Cloud Service A) una vez que se ha excedido el umbral de uso del Servicio A (1). Se utiliza un sistema de replicación de recursos para mantener sincronizadas las bases de datos de gestión de estado (2).

Además del escucha de escalado automatizado y la replicación de recursos, se pueden usar muchos otros mecanismos para automatizar la dinámica de burst in y burst out de esta arquitectura, dependiendo principalmente del tipo de recurso de TI que se escala.

9.7. Arquitectura de aprovisionamiento de discos elásticos

A los consumidores de la nube se les suele cobrar por el espacio de almacenamiento basado en la nube en función de la asignación de almacenamiento de disco fijo, lo que significa que los cargos están predeterminados por la capacidad del disco y no se alinean con el consumo real de almacenamiento de datos. La Figura 9.13 demuestra esto al ilustrar un escenario en el que un consumidor de la nube aprovisiona un servidor virtual con el sistema operativo Windows Server y tres discos duros de 150 GB. Al consumidor de la nube se le factura el uso de 450 GB de espacio de almacenamiento después de instalar el sistema operativo, aunque aún no haya instalado ningún software.

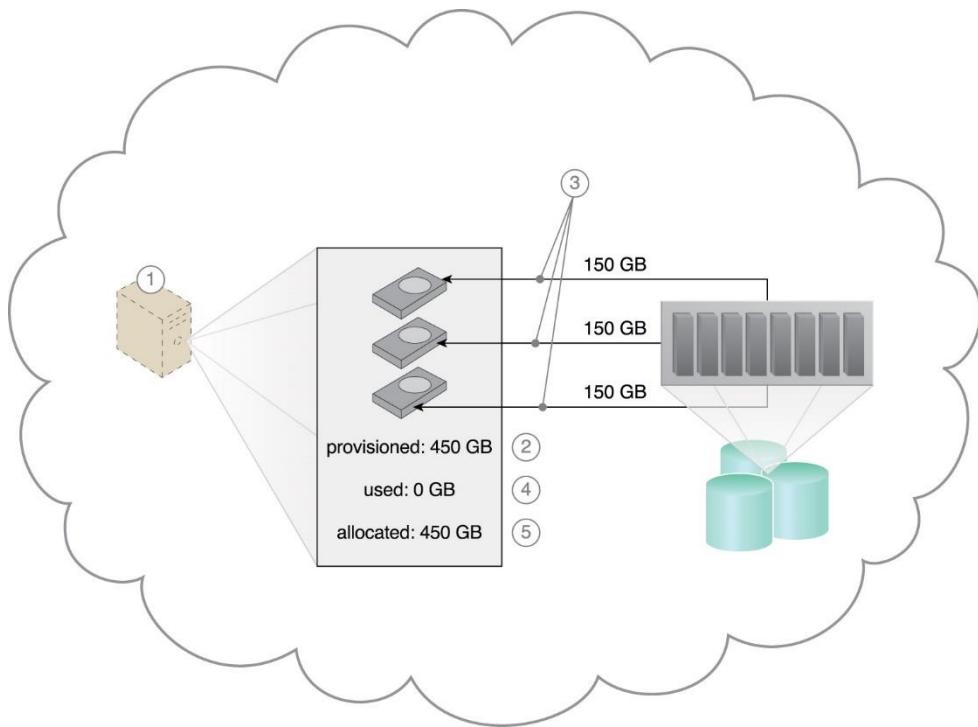


Figura 9.13 El consumidor de la nube solicita un servidor virtual con tres discos duros, cada uno con una capacidad de 150 GB (1). El servidor virtual se aprovisiona según la arquitectura de aprovisionamiento de discos elásticos, con un total de 450 GB de espacio en disco (2). El proveedor de la nube asigna los 450 GB al servidor virtual (3). El consumidor de la nube aún no ha instalado ningún software, lo que significa que el espacio utilizado actualmente es de 0 GB (4). Debido a que los 450 GB ya están asignados y reservados para el consumidor de la nube, se le cobrará por 450 GB de uso de disco a partir del punto de asignación (5).

La arquitectura de aprovisionamiento de discos elásticos establece un sistema de aprovisionamiento de almacenamiento dinámico que garantiza que el consumidor de la nube reciba una factura granular por la cantidad exacta de almacenamiento que realmente utiliza. Este sistema utiliza tecnología de aprovisionamiento ligero para la asignación dinámica de espacio de almacenamiento y, además, cuenta con el respaldo del monitoreo del uso en tiempo de ejecución para recopilar datos de uso precisos con fines de facturación (Figura 9.14).

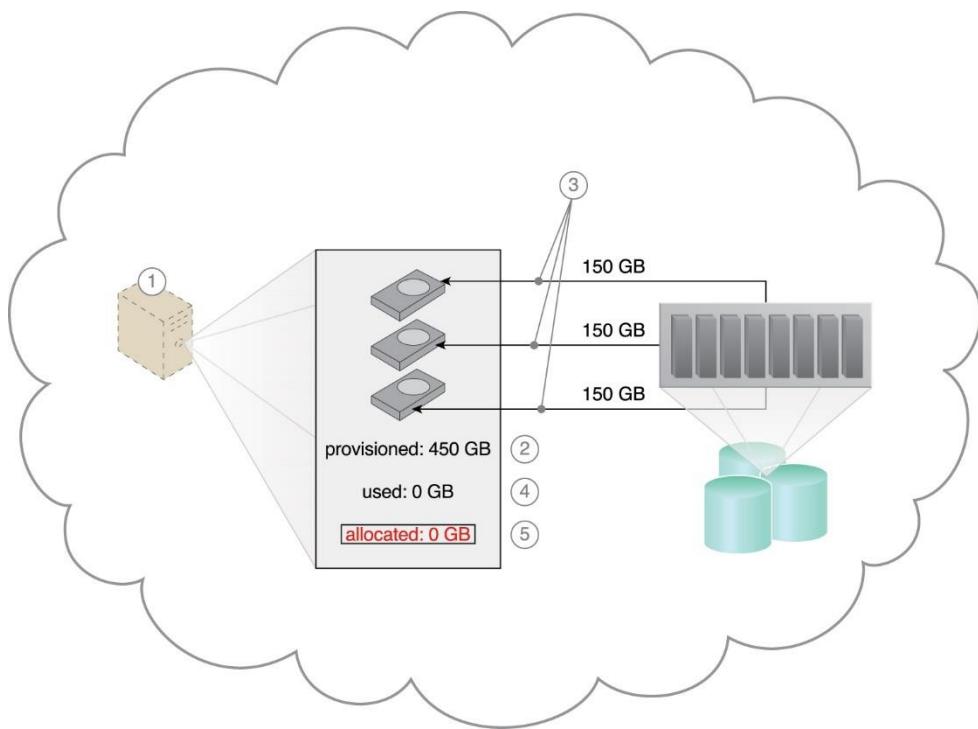


Figura 9.14 El consumidor de la nube solicita un servidor virtual con tres discos duros, cada uno con una capacidad de 150 GB (1). El servidor virtual está aprovisionado por esta arquitectura con un total de 450 GB de espacio en disco (2). Los 450 GB se establecen como el uso máximo de disco permitido para este servidor virtual, aunque todavía no se ha reservado ni asignado espacio en el disco físico (3). El consumidor de la nube no ha instalado ningún software, lo que significa que el espacio utilizado actualmente es de 0 GB (4). Debido a que el espacio en disco asignado es igual al espacio utilizado real (que actualmente es cero), el consumidor de la nube no paga por el uso del espacio en disco (5).

El software de aprovisionamiento delgado (thin) se instala en servidores virtuales que procesan la asignación dinámica de almacenamiento a través del hipervisor, mientras que el monitor de pago por uso rastrea y reporta datos de uso granular del disco para la facturación (Figura 9.15).

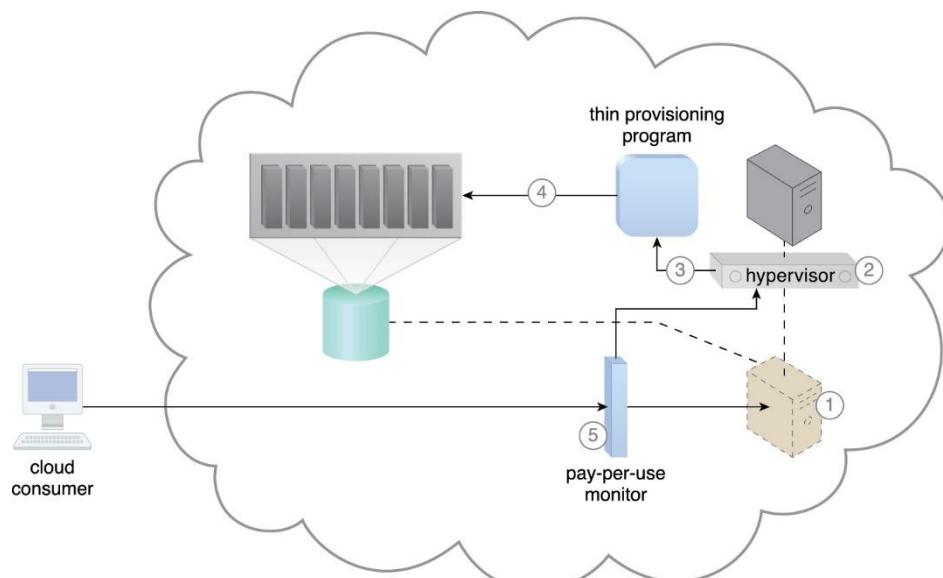


Figura 9.15 Se recibe una solicitud de un consumidor de la nube y comienza el aprovisionamiento de una nueva instancia de servidor virtual (1). Como parte del proceso de aprovisionamiento, los discos duros se eligen como discos dinámicos o de aprovisionamiento thin (2). El hipervisor llama a un componente de asignación dinámica de discos para crear discos delgados para el servidor virtual (3). Los discos de servidores virtuales se crean a través del programa de aprovisionamiento thin y se guardan en una carpeta de tamaño casi cero. El tamaño de esta carpeta y sus archivos crece a medida que se instalan aplicaciones operativas y se copian archivos adicionales en el servidor virtual (4). El monitor de pago por uso realiza un seguimiento del almacenamiento real asignado dinámicamente con fines de facturación (5).

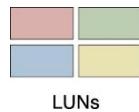
Los siguientes mecanismos se pueden incluir en esta arquitectura además del dispositivo de almacenamiento en la nube, el servidor virtual, el hipervisor y el monitor de pago por uso:

- *Monitor de uso de la nube*: Se pueden usar monitores de uso de la nube especializados para rastrear y registrar las fluctuaciones del uso del almacenamiento.
- *Replicación de recursos*: La replicación de recursos es parte de un sistema de aprovisionamiento de discos elásticos cuando se requiere la conversión de almacenamiento dinámico en disco delgado a almacenamiento estático en disco grueso.

9.8. Arquitectura de almacenamiento redundante

Los dispositivos de almacenamiento en la nube ocasionalmente están sujetos a fallas e interrupciones causadas por problemas de conectividad de la red, fallas generales del hardware o del controlador, o violaciones de seguridad. La confiabilidad de un dispositivo de almacenamiento en la nube comprometido puede tener un efecto dominó y causar fallas de impacto en todos los servicios, aplicaciones y componentes de infraestructura en la nube que dependen de su disponibilidad.

LUN



Un número lógico de unidad (LUN) es un drive lógico que representa una partición en un drive físico.

Storage Service Gateway



Un Storage Service Gateway (puerta de enlace del servicio de almacenamiento) es un componente que actúa como interfaz externa para los servicios de almacenamiento en la nube y es capaz de redirectir automáticamente las solicitudes de los consumidores de la nube cada vez que cambia la ubicación de los datos solicitados.

La arquitectura de almacenamiento redundante introduce un dispositivo de almacenamiento en la nube duplicado secundario como parte de un sistema de failover que sincroniza sus datos con los datos en el dispositivo de almacenamiento en la nube principal. Una puerta de enlace de servicio de almacenamiento desvía las solicitudes de los consumidores de la nube al dispositivo secundario cada vez que falla el dispositivo principal (Figuras 9.16 y 9.17).

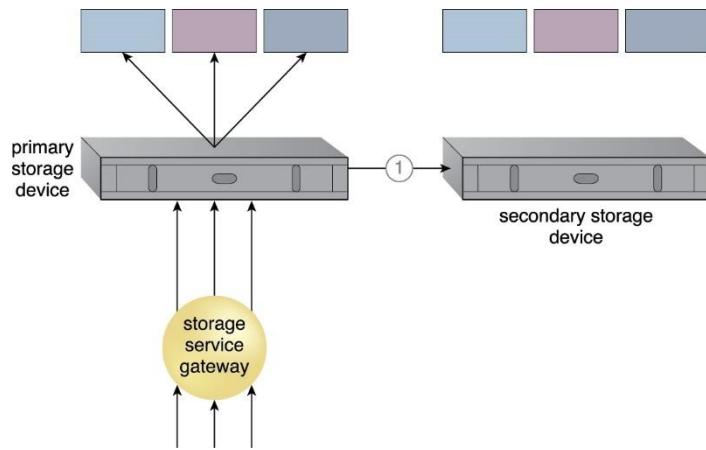


Figura 9.16 El dispositivo de almacenamiento primario en la nube es rutinariamente replicada al dispositivo de almacenamiento secundario en la nube (1).

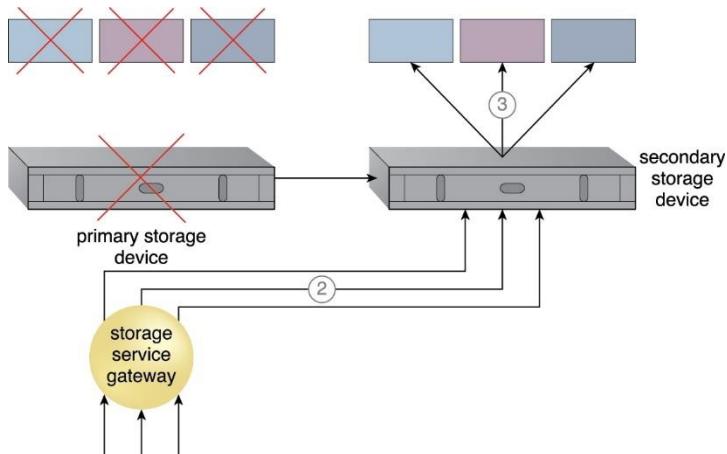


Figura 9.17 El almacenamiento principal deja de estar disponible y la puerta de enlace del servicio de almacenamiento reenvía las solicitudes del consumidor de la nube al dispositivo de almacenamiento secundario (2). El dispositivo de almacenamiento secundario reenvía las solicitudes a los LUNs, lo que permite a los consumidores de la nube continuar accediendo a sus datos (3).

Esta arquitectura de nube se basa principalmente en un sistema de replicación de almacenamiento que mantiene el dispositivo de almacenamiento en la nube principal sincronizado con sus dispositivos de almacenamiento en la nube secundarios duplicados (Figura 9.18).

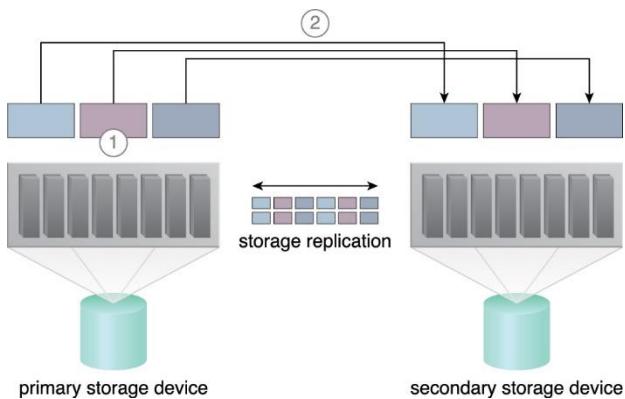
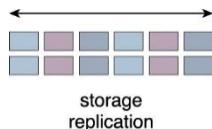


Figura 9.18 La replicación de almacenamiento se utiliza para mantener el dispositivo de almacenamiento redundante sincronizado con el dispositivo de almacenamiento principal.

Replicación de almacenamiento

La replicación de almacenamiento es una variación de los mecanismos de replicación de recursos utilizados para replicar datos de forma síncrona o asíncrona desde un dispositivo de almacenamiento principal a un dispositivo de almacenamiento secundario. Se puede utilizar para replicar LUNs parciales y completos.



Los proveedores de la nube pueden ubicar dispositivos de almacenamiento en la nube secundarios en una región geográfica diferente a la del dispositivo de almacenamiento en la nube principal, generalmente por razones económicas. Sin embargo, esto puede presentar problemas legales para algunos tipos de datos. La ubicación de los dispositivos de almacenamiento en la nube secundarios puede dictar el protocolo y el método utilizado para la sincronización, ya que algunos protocolos de transporte de replicación tienen restricciones de distancia.

Algunos proveedores de la nube usan dispositivos de almacenamiento con matriz doble y controladores de almacenamiento para mejorar la redundancia de los dispositivos y colocan los dispositivos de almacenamiento secundarios en una ubicación física diferente para el balanceo de la nube y la recuperación ante desastres. En este caso, es posible que los proveedores de nube necesiten alquilar una conexión de red a través de un proveedor de nube externo para establecer la replicación entre los dos dispositivos.

9.9. Ejemplo de Estudio de Caso

Una solución in-house que ATN no migró a la nube es el Remote Upload Module, un programa que utilizan sus clientes para cargar documentos contables y legales a un archivo central diariamente. Los picos de uso se producen sin previo aviso, ya que la cantidad de documentos que se reciben día a día es impredecible.

Actualmente, el módulo de carga remota rechaza los intentos de carga cuando está funcionando al máximo de su capacidad, lo que es problemático para los usuarios que necesitan archivar ciertos documentos antes del final de un día hábil o antes de una fecha límite.

ATN decide aprovechar su entorno basado en la nube mediante la creación de una arquitectura cloud-bursting en torno a la implementación del servicio del Remote Upload Module on-premise. Esto le permite hacer burst out en la nube siempre que se excedan los umbrales de procesamiento on-premise (Figuras 9.19 y 9.20).

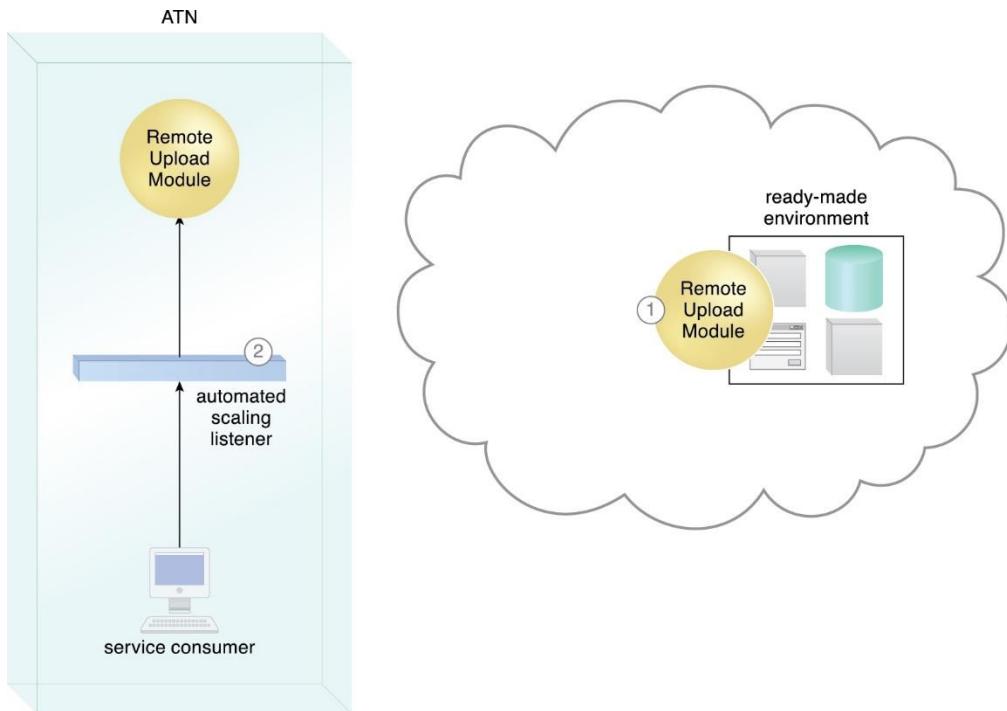


Figura 9.19 Se implementa una versión basada en la nube del servicio del Remote Upload Module on-premise en el entorno prefabricado alquilado de ATN (1). El escucha de escalado automatizado supervisa las solicitudes de los consumidores del servicio (2).

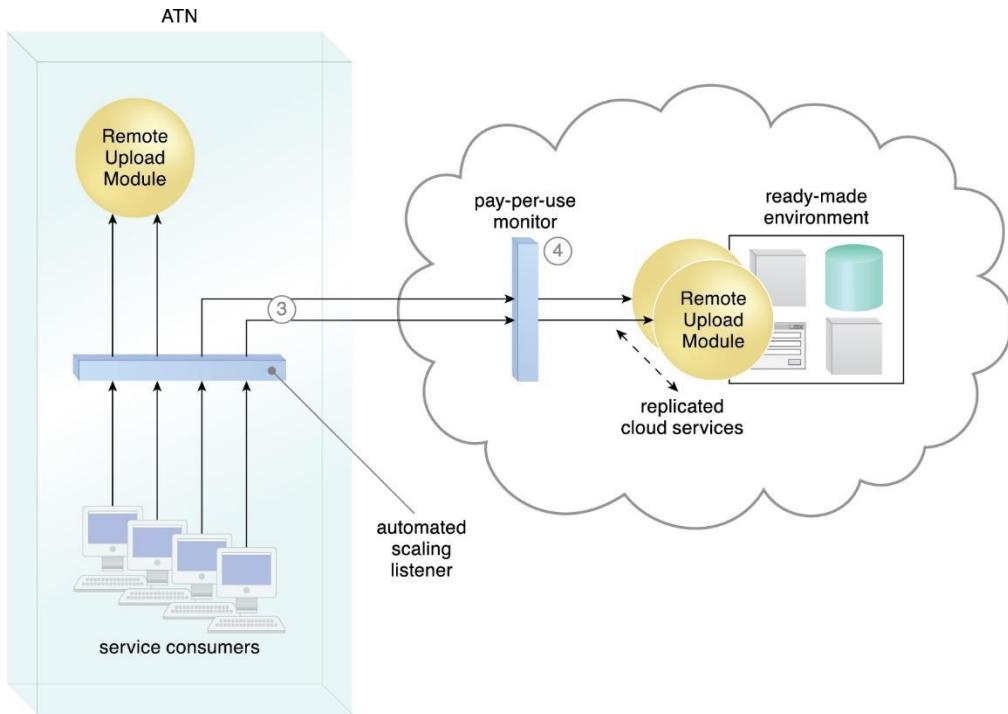


Figura 9.20 El oyente de escalado automatizado detecta que el uso del consumidor del servicio ha excedido el umbral de uso del servicio del Remote Upload Module local y comienza a desviar el exceso de solicitudes a la implementación del Remote Upload Module basado en la nube (3). El monitor de pago por uso del proveedor de la nube realiza un seguimiento de las solicitudes recibidas del oyente de escalado automatizado on-premise para recopilar datos de facturación, y las instancias del servicio en la nube del Remote Upload Module se crean bajo demanda a través de la replicación de recursos (4).

Se invoca un sistema de “burst in” después de que el uso del servicio ha disminuido lo suficiente como para que la implementación del Remote Upload Module on-premise pueda procesar nuevamente las solicitudes de los consumidores del servicio. Instancias de los servicios cloud son liberadas, y no se incurre en honorarios adicionales relacionados con el uso de la nube.

10 Arquitecturas de nube avanzadas



Las arquitecturas de tecnología de nube exploradas en este capítulo representan capas arquitectónicas distintas y sofisticadas, varias de las cuales pueden construirse sobre los entornos más fundamentales establecidos por los modelos arquitectónicos cubiertos en el capítulo anterior.

10.1. Hypervisor Clustering Architecture

Los hipervisores pueden ser responsables de crear y hospedar múltiples servidores virtuales. Debido a esta dependencia, cualquier condición de falla que afecte a un hipervisor puede repercutir en cascada en sus servidores virtuales (Figura 10.1).

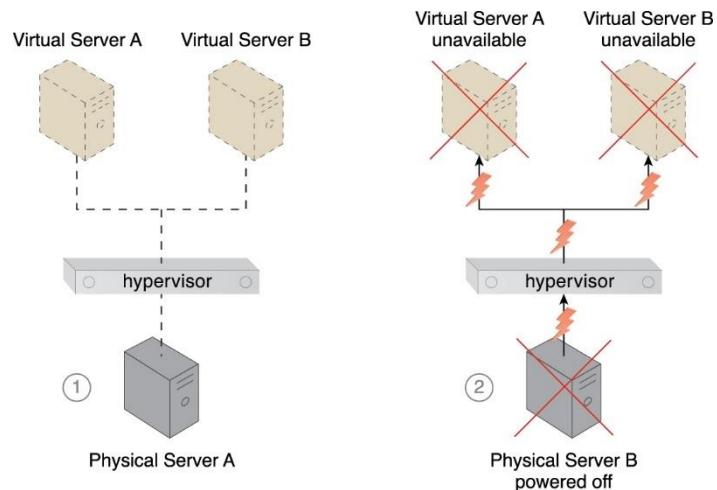


Figura 10.1 El servidor físico A aloja un hipervisor que alberga los servidores virtuales A y B (1). Cuando el servidor físico A falla, el hipervisor y dos servidores virtuales también fallan (2).

Heartbeats



Los heartbeats (latidos del corazón) son mensajes a nivel de sistema que se intercambian entre hipervisores, hipervisores y servidores virtuales, e hipervisores y VIMs.

La Hypervisor Clustering Architecture (arquitectura de clústeres de hipervisores) establece un clúster de hipervisores de alta disponibilidad en varios servidores físicos. Si un hipervisor determinado o su servidor físico subyacente deja de estar disponible, los servidores virtuales alojados se pueden mover a otro servidor físico o hipervisor para mantener las operaciones de tiempo de ejecución (Figura 10.2).

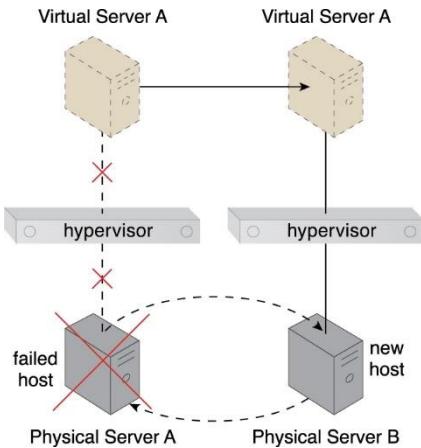
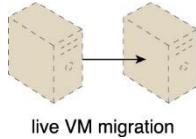


Figura 10.2 El servidor físico A deja de estar disponible y hace que su hipervisor falle. El servidor virtual A se migra al servidor físico B, que tiene otro hipervisor que forma parte del clúster al que pertenece el servidor físico A.

El clúster de hipervisores se controla a través de un VIM central, que envía mensajes de latidos regulares a los hipervisores para confirmar que están en funcionamiento. Los mensajes de latido no reconocidos hacen que el VIM inicie el programa de migración de VM en vivo para mover dinámicamente los servidores virtuales afectados a un nuevo host.

Migración de máquinas virtuales en vivo

La migración de máquinas virtuales en vivo es un sistema que es capaz de reubicar servidores virtuales o instancias de servidores virtuales en tiempo de ejecución.



El clúster de hipervisor utiliza un dispositivo de almacenamiento en la nube compartido para migrar servidores virtuales en vivo, como se ilustra en las Figuras 10.3 a 10.6.

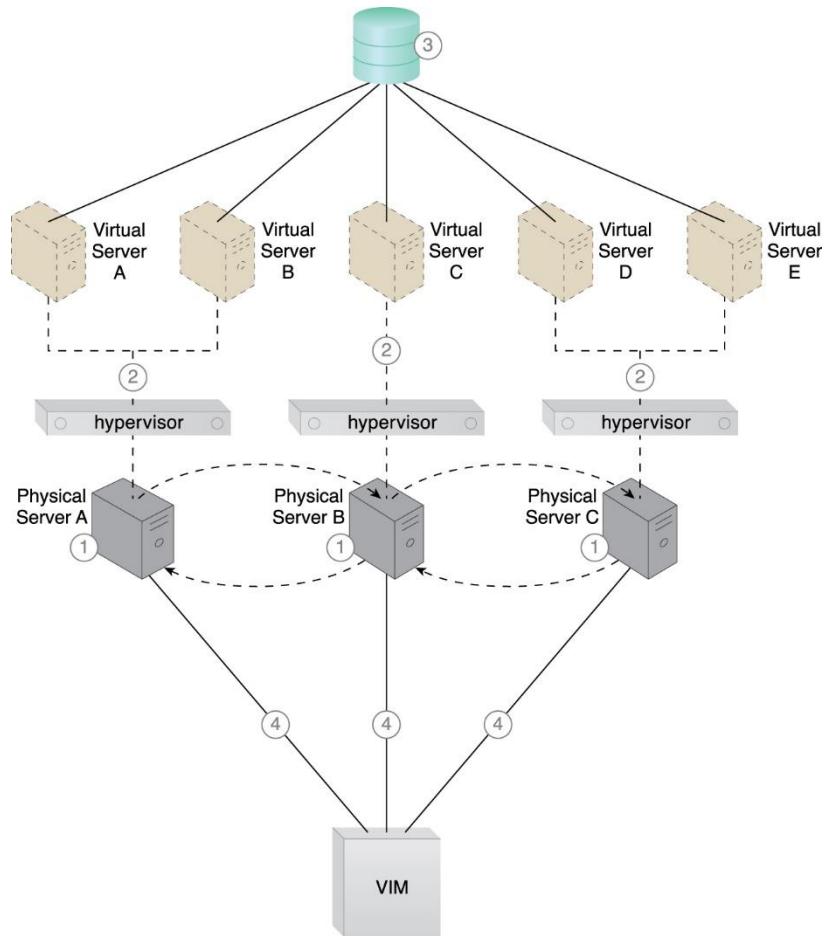


Figura 10.3 Los hipervisores están instalados en los servidores físicos A, B y C (1). Los servidores virtuales son creados por los hipervisores (2). Un dispositivo de almacenamiento en la nube compartido que contiene archivos de configuración del servidor virtual se coloca en un dispositivo de almacenamiento en la nube compartido para el acceso de todos los hipervisores (3). El clúster de hipervisor está habilitado en los tres servidores físicos a través de un VIM central (4).

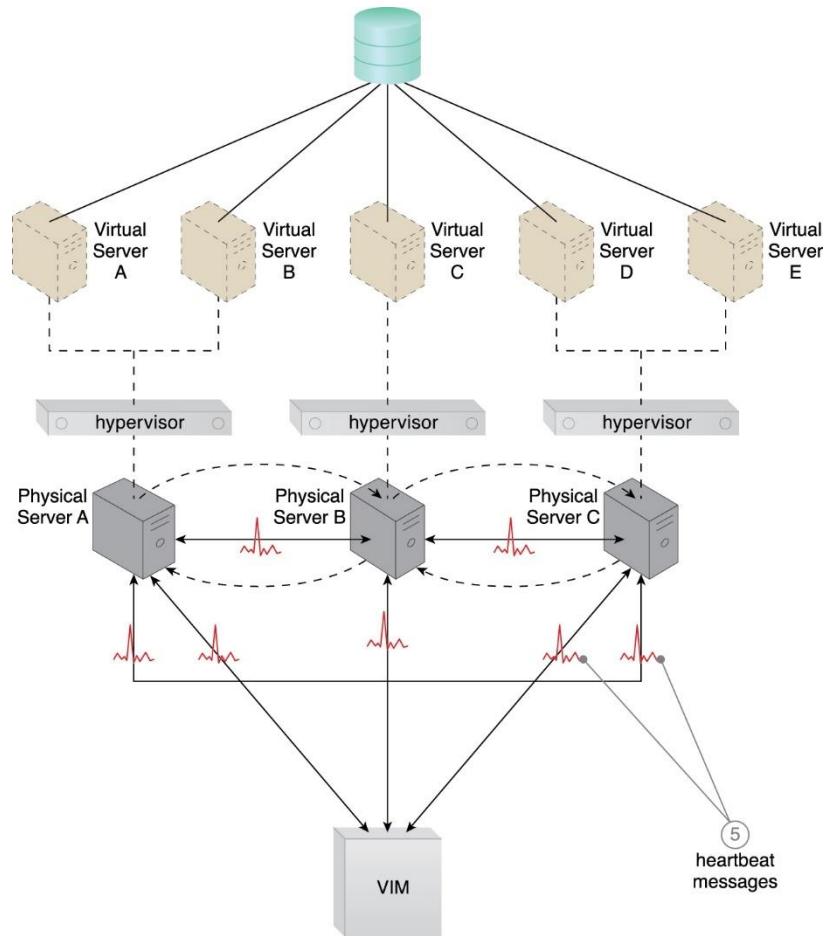


Figura 10.4 Los servidores físicos intercambian mensajes de latidos entre sí y con el VIM de acuerdo con una calendarización predefinida (5).

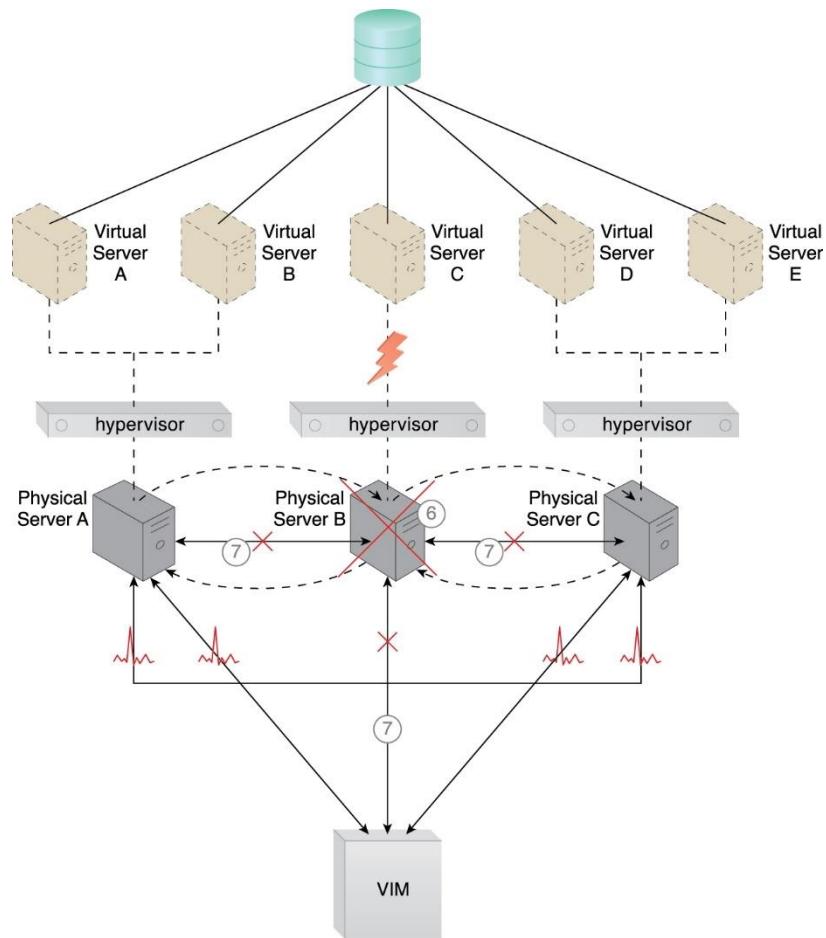


Figura 10.5 El servidor físico B falla y deja de estar disponible, lo que pone en peligro el servidor virtual C (6). Los demás servidores físicos y el VIM dejan de recibir mensajes de latido del servidor físico B (7).

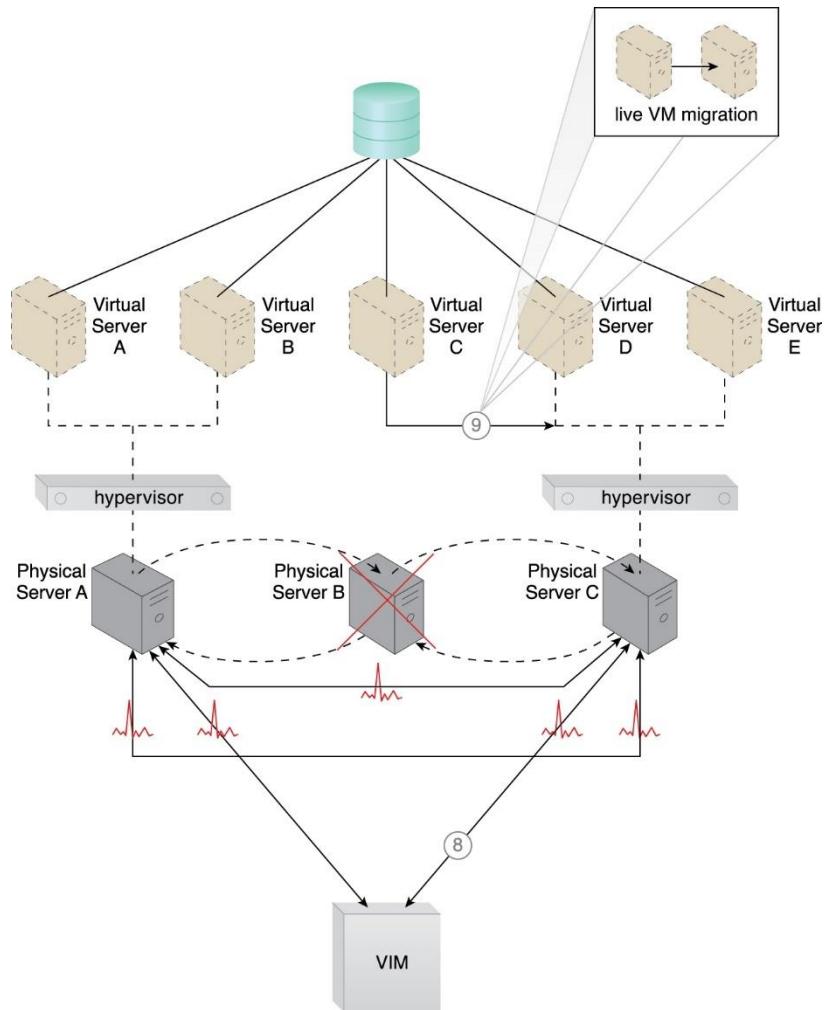


Figura 10.6 El VIM elige el Servidor físico C como el nuevo host para tomar posesión del Servidor virtual C después de evaluar la capacidad disponible de otros hipervisores en el clúster (8). El servidor virtual C se migra en vivo al hipervisor que se ejecuta en el servidor físico C, donde puede ser necesario reiniciar antes de que se puedan reanudar las operaciones normales (9).

Además del hipervisor y los mecanismos de clúster de recursos que forman el núcleo de este modelo arquitectónico y los servidores virtuales que están protegidos por el entorno en clúster, se pueden incorporar los siguientes mecanismos:

- **Perímetro de red lógica:** Los límites lógicos creados por este mecanismo aseguran que ninguno de los hipervisores de otros consumidores de la nube se incluya accidentalmente en un clúster determinado.
- **Replicación de recursos:** Los hipervisores en el mismo clúster se informan entre sí sobre su estado y disponibilidad. Las actualizaciones sobre cualquier cambio que ocurra en el clúster, como la creación o eliminación de un comutador virtual, deben replicarse en todos los hipervisores a través del VIM.

10.2. Arquitectura de instancias de servidor virtual con equilibrio de carga

Mantener las cargas de trabajo entre servidores balanceadas uniformemente entre servidores físicos cuya operación y administración están aisladas puede ser un desafío. Un servidor físico fácilmente puede terminar alojando más servidores virtuales o recibir cargas de trabajo más grandes que sus servidores físicos vecinos (Figura 10.7). Tanto la sobreutilización como la infrautilización del servidor físico pueden aumentar drásticamente con el tiempo, lo que genera desafíos continuos de rendimiento (para servidores sobreutilizados) y desperdicio constante (para el potencial de procesamiento perdido de servidores infrautilizados).

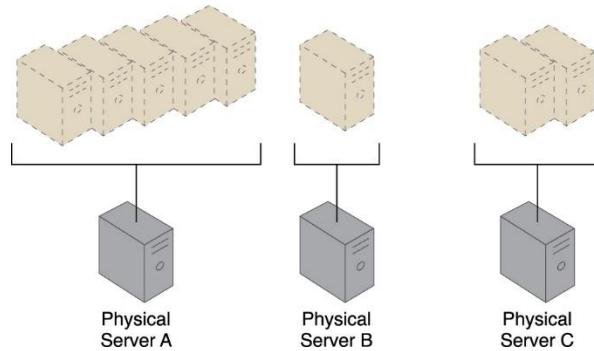


Figura 10.7 Tres servidores físicos tienen que albergar diferentes cantidades de instancias de servidores virtuales, lo que genera servidores sobreutilizados e infrautilizados.

La arquitectura de instancias de servidores virtuales con equilibrio de carga establece un sistema de vigilancia de la capacidad, que calcula dinámicamente las cargas de trabajo asociadas a las instancias de servidores virtuales, antes de distribuir el procesamiento entre los hosts de servidores físicos disponibles (Figura 10.8).

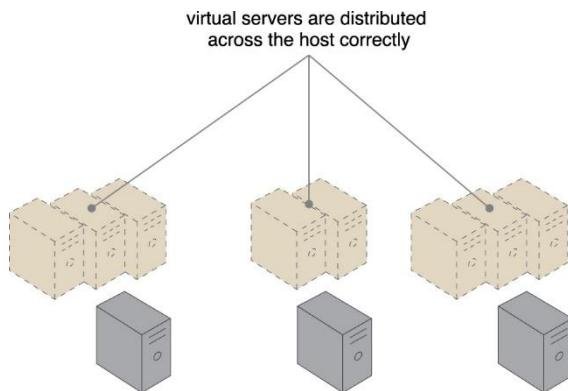


Figura 10.8 Las instancias del servidor virtual se distribuyen de manera más uniforme entre los hosts del servidor físico.

El capacity watchdog system (sistema de vigilancia de la capacidad) se compone de un watchdog cloud usage monitor, el programa de migración de máquinas virtuales en vivo y un planificador de capacidad. El capacity watchdog rastrea el uso del servidor físico y virtual e informa cualquier fluctuación significativa al planificador de capacidad, que es responsable de calcular dinámicamente las capacidades informáticas del servidor físico contra los requisitos de capacidad del servidor virtual. Si el planificador de capacidad decide mover un servidor virtual a otro host para distribuir la

carga de trabajo, se indica al programa de migración de VM en vivo que mueva el servidor virtual (Figuras 10.9 a 10.11).

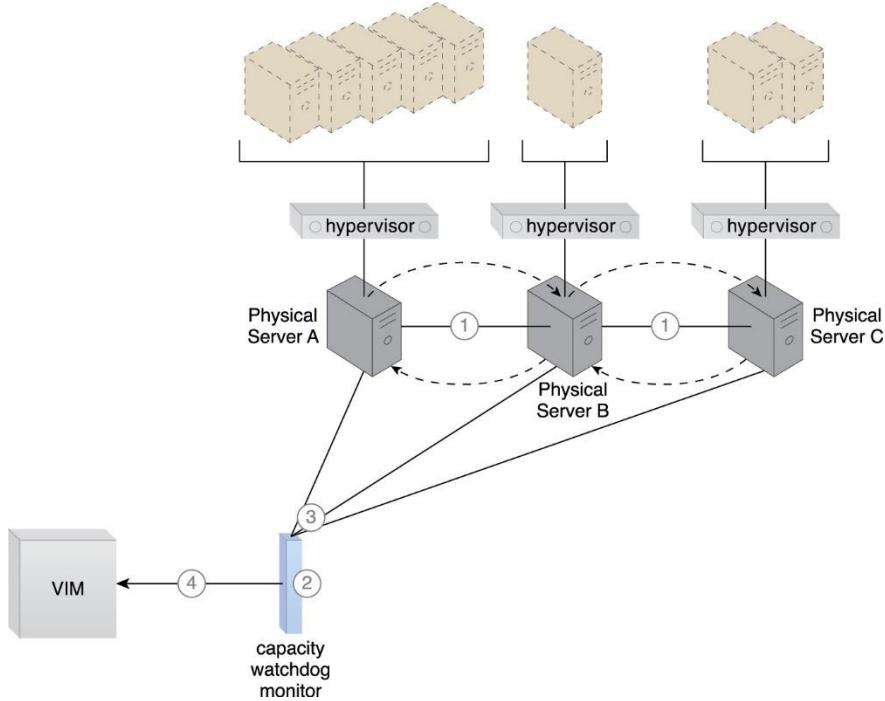


Figura 10.9 La arquitectura del clúster del hipervisor proporciona la base sobre la cual se construye la arquitectura del servidor virtual con balanceo de carga (1). Las políticas y los umbrales se definen para el capacity watchdog (2), el cual compara las capacidades del servidor físico con el procesamiento del servidor virtual (3). El watchdog de capacidad informa una sobreutilización al VIM (4).

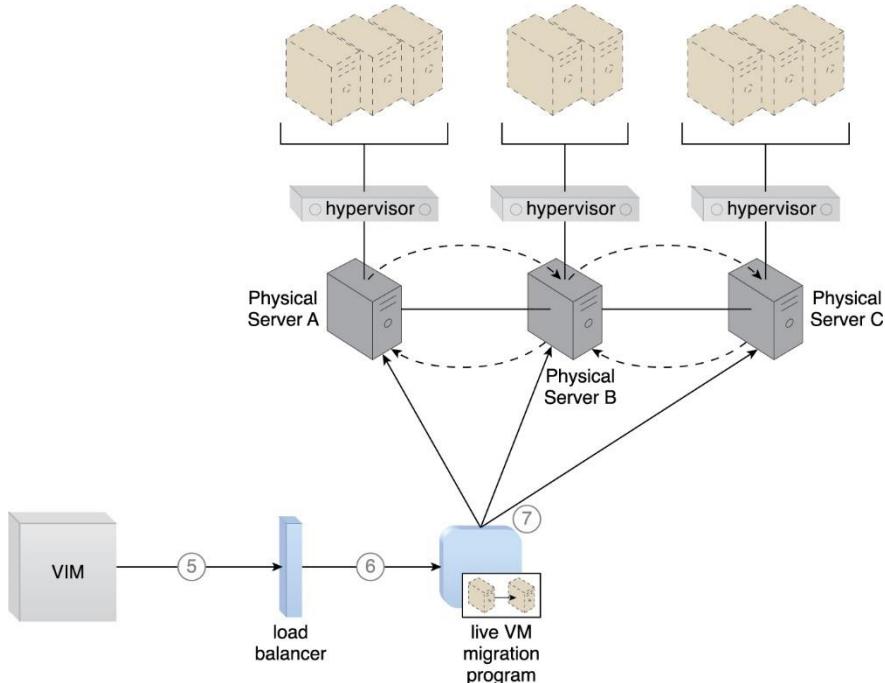


Figura 10.10 El VIM indica al balanceador de carga que redistribuya la carga de trabajo en función de los umbrales predefinidos (5). El equilibrador de carga inicia el programa de migración de VM en vivo para mover los servidores virtuales (6). La migración de máquinas virtuales en vivo mueve los servidores virtuales seleccionados de un host físico a otro (7).

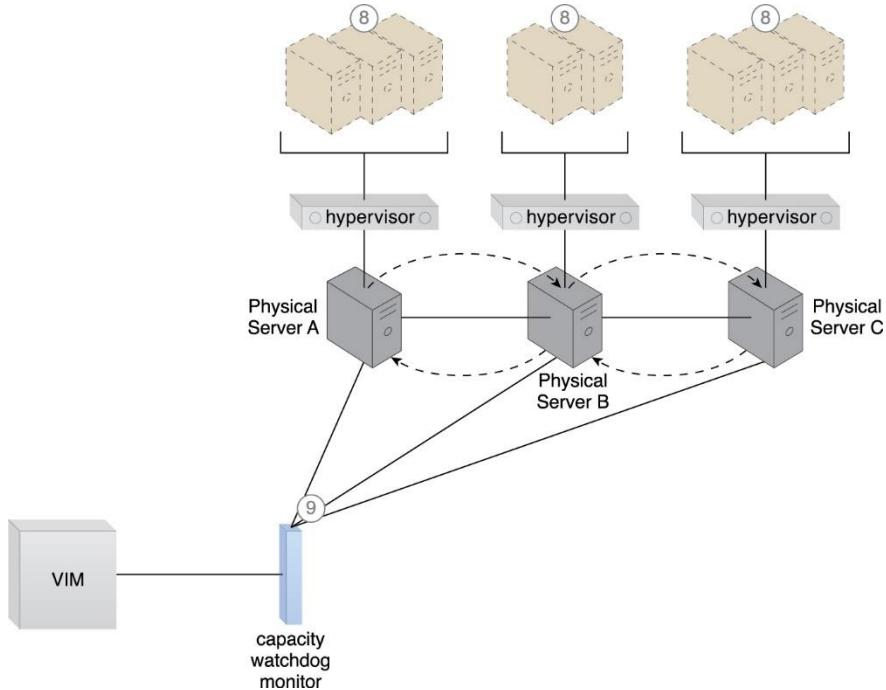


Figura 10.11 La carga de trabajo está balanceada entre los servidores físicos del clúster (8). El capacity watchdog sigue supervisando la carga de trabajo y el consumo de recursos (9).

Los siguientes mecanismos se pueden incluir en esta arquitectura:

- *Automated Scaling Listener* - La escucha de escala automática se puede usar para iniciar el proceso de balanceo de carga y monitorear dinámicamente la carga de trabajo que llega a los servidores virtuales a través de los hipervisores.
- *Load Balancer* - El mecanismo del balanceo de carga es responsable de distribuir la carga de trabajo de los servidores virtuales entre los hipervisores.
- *Logical Network Perimeter* - Un perímetro de red lógica garantiza que el destino de un servidor virtual reubicado cumpla con los SLA y las normas de privacidad.
- *Resource Replication* - Es posible que se requiera la replicación de instancias de servidores virtuales como parte de la funcionalidad de balanceo de carga.

10.3. Arquitectura de reubicación de servicios no disruptiva

Un servicio en la nube puede dejar de estar disponible por varios motivos, como:

- Demandas de uso del tiempo de ejecución que exceden su capacidad de procesamiento
- Una actualización de mantenimiento que exige una interrupción temporal
- Migración permanente a un nuevo servidor físico host

Las solicitudes de los consumidores del servicio en la nube generalmente se rechazan si un servicio en la nube deja de estar disponible, lo que puede resultar potencialmente en condiciones de excepción. No es deseable hacer que el servicio en la nube no esté disponible temporalmente para los consumidores de la nube, incluso si la interrupción está planificada.

La arquitectura de reubicación de servicios sin interrupciones establece un sistema mediante el cual un evento predefinido desencadena la duplicación o migración de una implementación de servicios en la nube en tiempo de ejecución, evitando así cualquier interrupción. En lugar de escalar los servicios en la nube in o out con implementaciones redundantes, la actividad del servicio en la nube se puede desviar temporalmente a otro host en tiempo de ejecución agregando una implementación duplicada en el nuevo host. De manera similar, las solicitudes de los consumidores de servicios en la nube se pueden redirigir temporalmente a una implementación duplicada cuando la implementación original necesita sufrir una interrupción de mantenimiento. La reubicación de la implementación del servicio en la nube y cualquier actividad del servicio en la nube también puede ser permanente para adaptarse a las migraciones del servicio en la nube a nuevos hosts de servidores físicos.

Un aspecto clave de la arquitectura subyacente es que se garantiza que la implementación del nuevo servicio en la nube recibirá y responderá con éxito a las solicitudes de los consumidores del servicio en la nube antes de que se desactive o elimine la implementación del servicio en la nube original. Un enfoque común es que la migración de máquinas virtuales en vivo mueva toda la instancia de servidor virtual que aloja el servicio en la nube. Los mecanismos de balanceo de carga y/o oyente de escalado automatizado se pueden usar para desencadenar una redirección temporal de las solicitudes de los consumidores de servicios en la nube, en respuesta a los requisitos de escalamiento y distribución de la carga de trabajo. Cualquiera de los mecanismos puede comunicarse con el VIM para iniciar el proceso de migración de la VM en vivo, como se muestra en las Figuras 10.12 a 10.14.

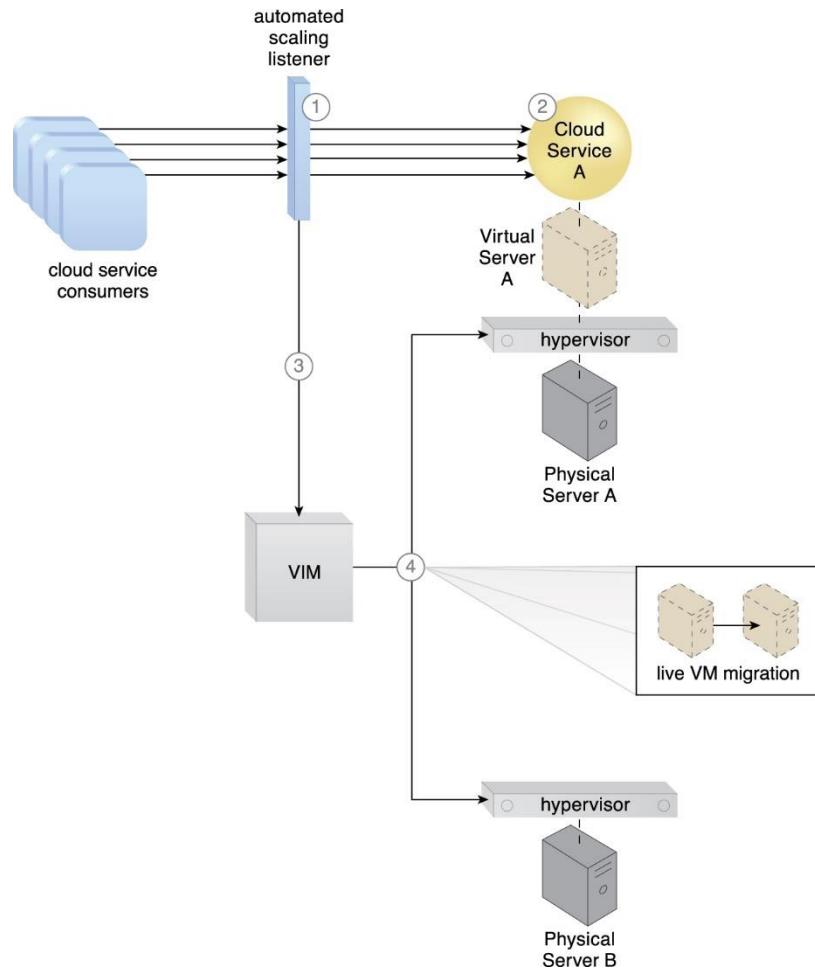


Figura 10.12 El oyente de escalado automatizado supervisa la carga de trabajo de un servicio en la nube (1). El umbral predefinido del servicio en la nube se alcanza a medida que aumenta la carga de trabajo (2), lo que hace que el agente de escucha de escalado automatizado indique al VIM que inicie la reubicación (3). El VIM usa el programa de migración de VM en vivo para indicar a los hipervisores de origen y de destino que lleven a cabo la reubicación en tiempo de ejecución (4).

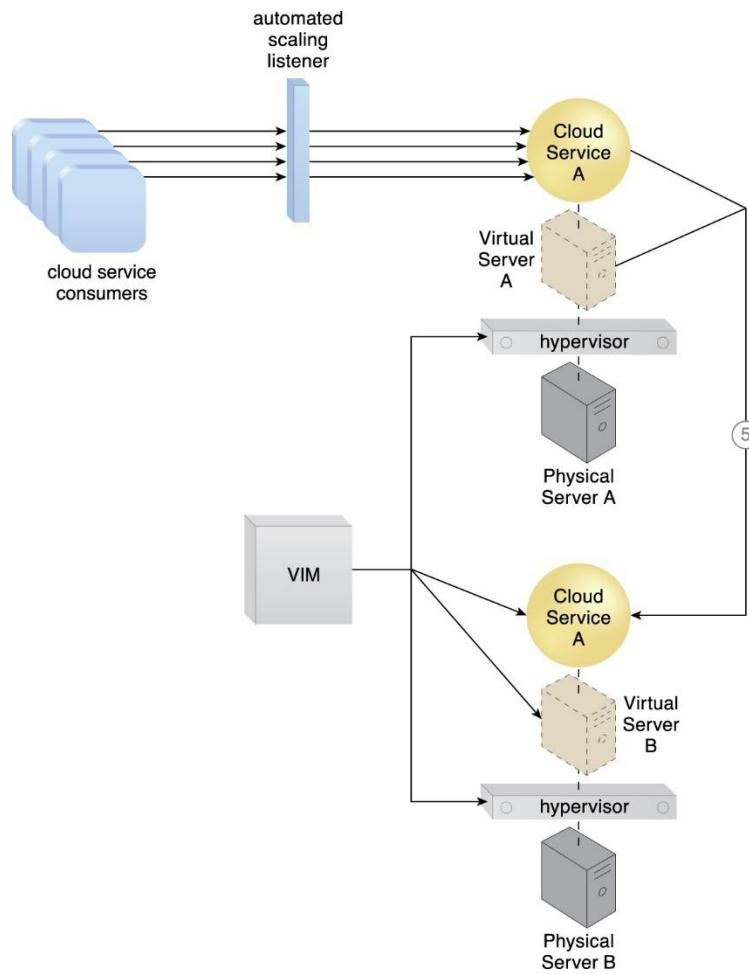


Figura 10.13 Se crea una segunda copia del servidor virtual y su servicio de nube alojado a través del hipervisor de destino en el servidor físico B (5).

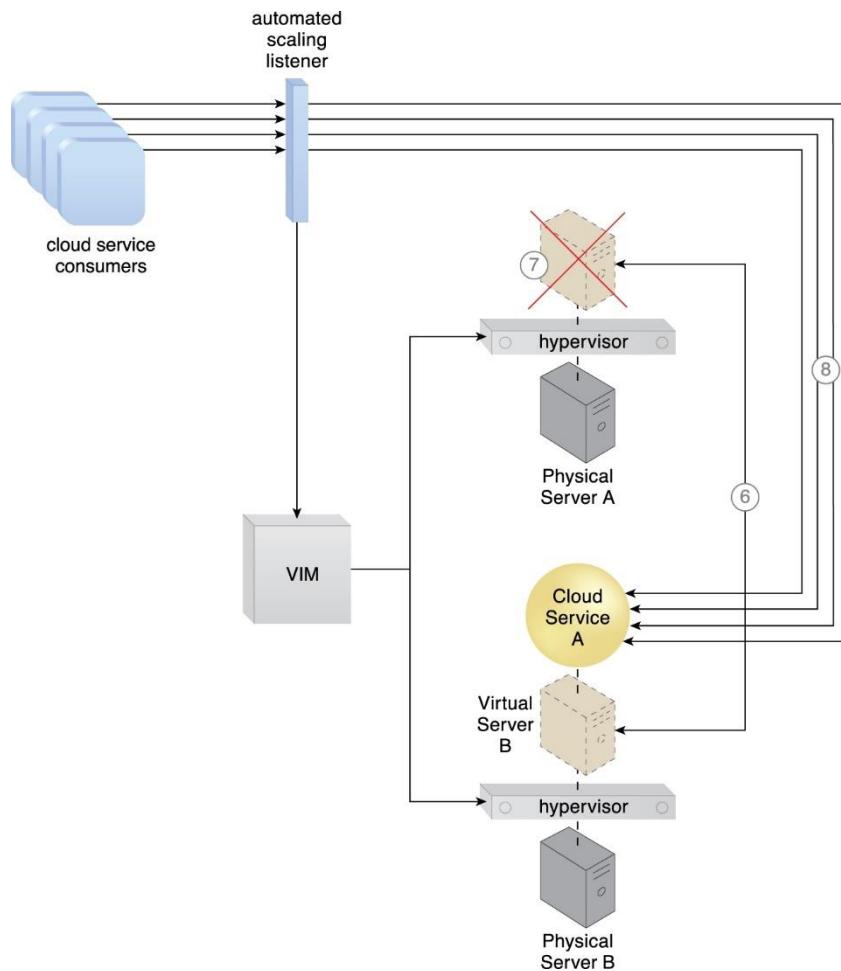


Figura 10.14 El estado de ambas instancias del servidor virtual está sincronizado (6). La primera instancia de servidor virtual se elimina del servidor físico A después de que se confirme que las solicitudes de los consumidores del servicio en la nube se intercambiaron correctamente con el servicio en la nube en el servidor físico B (7). Las solicitudes de los consumidores del servicio en la nube ahora solo se envían al servicio en la nube en el servidor físico B (8).

La migración del servidor virtual puede ocurrir de una de las dos maneras siguientes, según la ubicación de los discos y la configuración del servidor virtual:

- Se crea una copia de los discos del servidor virtual en el host de destino, si los discos del servidor virtual se almacenan en un servidor local, dispositivo de almacenamiento o dispositivos de almacenamiento remotos no compartidos conectados al host de origen. Una vez creada la copia, ambas instancias del servidor virtual se sincronizan y los archivos del servidor virtual se eliminan del host de origen.
- No es necesario copiar los discos del servidor virtual si los archivos del servidor virtual se almacenan en un dispositivo de almacenamiento remoto que se comparte entre los hosts de origen y de destino. El contexto del servidor virtual simplemente se transfiere desde el origen al host del servidor físico de destino, y el estado del servidor virtual se sincroniza automáticamente.

Esta arquitectura puede ser compatible con la arquitectura de configuraciones de red virtual persistente, de modo que las configuraciones de red definidas de los servidores virtuales migrados se conservan para mantener la conexión con los consumidores del servicio en la nube.

Además del escucha de escalado automatizado, el equilibrador de carga, el dispositivo de almacenamiento en la nube, el hipervisor y el servidor virtual, otros mecanismos que pueden formar parte de esta arquitectura incluyen lo siguiente:

- *Monitor de uso de la nube* - Se pueden usar diferentes tipos de monitores de uso de la nube para realizar un seguimiento continuo de TI uso de recursos y actividad del sistema.
- *Monitor de pay-per-use* - El monitor de pago por uso se utiliza para recopilar datos para los cálculos de costos de uso de servicios para los recursos de TI en las ubicaciones de origen y destino.
- *Replicación de recursos* - El mecanismo de replicación de recursos se usa para instanciar la instantánea del servicio en la nube, en su destino.
- *Sistema de gestión de SLA* - Este sistema de gestión es responsable de procesar los datos de SLA proporcionados por el monitor de SLA para obtener garantías de disponibilidad del servicio en la nube, tanto durante como después de la duplicación o reubicación del servicio en la nube.
- *Monitor de SLA* - Este mecanismo de monitoreo recopila la información de SLA requerida por el sistema de gestión de SLA, que puede ser relevante si las garantías de disponibilidad se basan en esta arquitectura.

[10.4. Arquitectura Zero Downtime](#)

Un servidor físico actúa naturalmente como un único punto de falla para los servidores virtuales que aloja. Como resultado, cuando el servidor físico falla o se ve comprometido, la disponibilidad de cualquiera (o todos) los servidores virtuales alojados pueden verse afectados. Esto hace que la emisión de garantías de zero downtime (tiempo de inactividad cero) por parte de un proveedor de la nube para los consumidores sea un desafío.

La arquitectura de Zero Downtime establece un sofisticado sistema de failover que permite que los servidores virtuales se muevan dinámicamente a diferentes hosts de servidores físicos, en caso de que falle su host de servidor físico original (Figura 10.15).

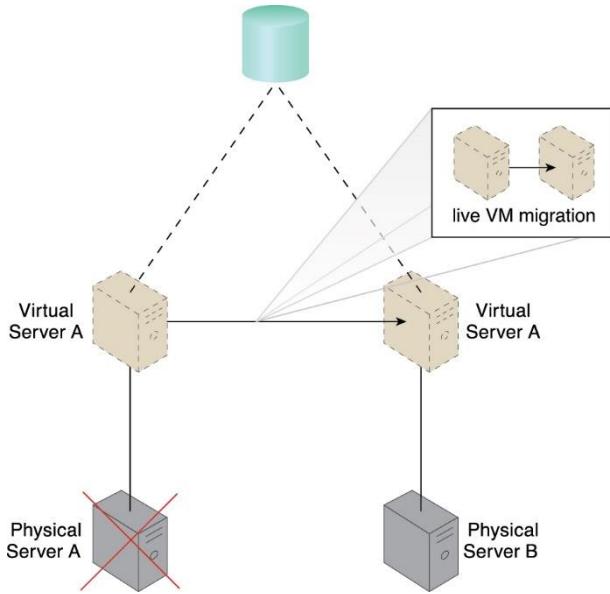


Figura 10.15 El servidor físico A falla y dispara el programa de migración de VM en vivo para mover dinámicamente el servidor virtual A al servidor físico B.

Varios servidores físicos se ensamblan en un grupo que está controlado por un sistema de failover capaz de conmutar la actividad de un servidor físico a otro, sin interrupción. El componente live VM migration suele ser una parte central de esta forma de arquitectura de nube de alta disponibilidad.

La tolerancia a fallas resultante asegura que, en caso de falla del servidor físico, los servidores virtuales alojados se migrarán a un servidor físico secundario. Todos los servidores virtuales se almacenan en un volumen compartido (según la arquitectura de configuración de red virtual persistente) para que otros servidores físicos del mismo grupo puedan acceder a sus archivos.

Además del sistema de failover, el dispositivo de almacenamiento en la nube y los mecanismos del servidor virtual, los siguientes mecanismos pueden formar parte de esta arquitectura:

- *Monitor de auditoría* - Este mecanismo puede ser necesario para verificar si la reubicación de los servidores virtuales también reubica los datos alojados en ubicaciones prohibidas.
- *Monitor de uso de la nube* - Las incorporaciones de este mecanismo se utilizan para monitorear el uso real de los recursos de TI de los consumidores de la nube para ayudar a garantizar que no se excedan las capacidades del servidor virtual.
- *Hipervisor* - El hipervisor de cada servidor físico afectado aloja los servidores virtuales afectados.
- *Perímetro de red lógica* - Los perímetros de red lógica brindan y mantienen el aislamiento necesario para garantizar que cada consumidor de nube permanezca dentro de su propio límite lógico luego de la reubicación del servidor virtual.
- *Clúster de recursos* - El mecanismo de clúster de recursos se aplica para crear diferentes tipos de grupos de clústeres activo-activo que colaborativamente mejoran la disponibilidad de los recursos de TI alojados en servidores virtuales.

- *Replicación de recursos* - Este mecanismo puede crear nuevas instancias de servidor virtual y servicio en la nube en caso de falla del servidor virtual principal.

10.5. Arquitectura de balanceo en la nube

La arquitectura de balanceo en la nube establece un modelo arquitectónico especializado en el que los recursos de TI se pueden balancear a través de múltiples nubes. El equilibrio entre nubes de las solicitudes de los consumidores de servicios en la nube puede ayudar a:

- Mejorar el rendimiento y la escalabilidad de los recursos de TI
- Aumentar la disponibilidad y confiabilidad de los recursos de TI
- Mejorar el balanceo de carga y la optimización de los recursos de TI

La funcionalidad de balanceo en la nube se basa principalmente en la combinación del escucha de escalamiento automatizado y los mecanismos del sistema failover (Figura 10.16).

Muchos más componentes (y posiblemente otros mecanismos) pueden formar parte de una arquitectura de balanceo en la nube completa.

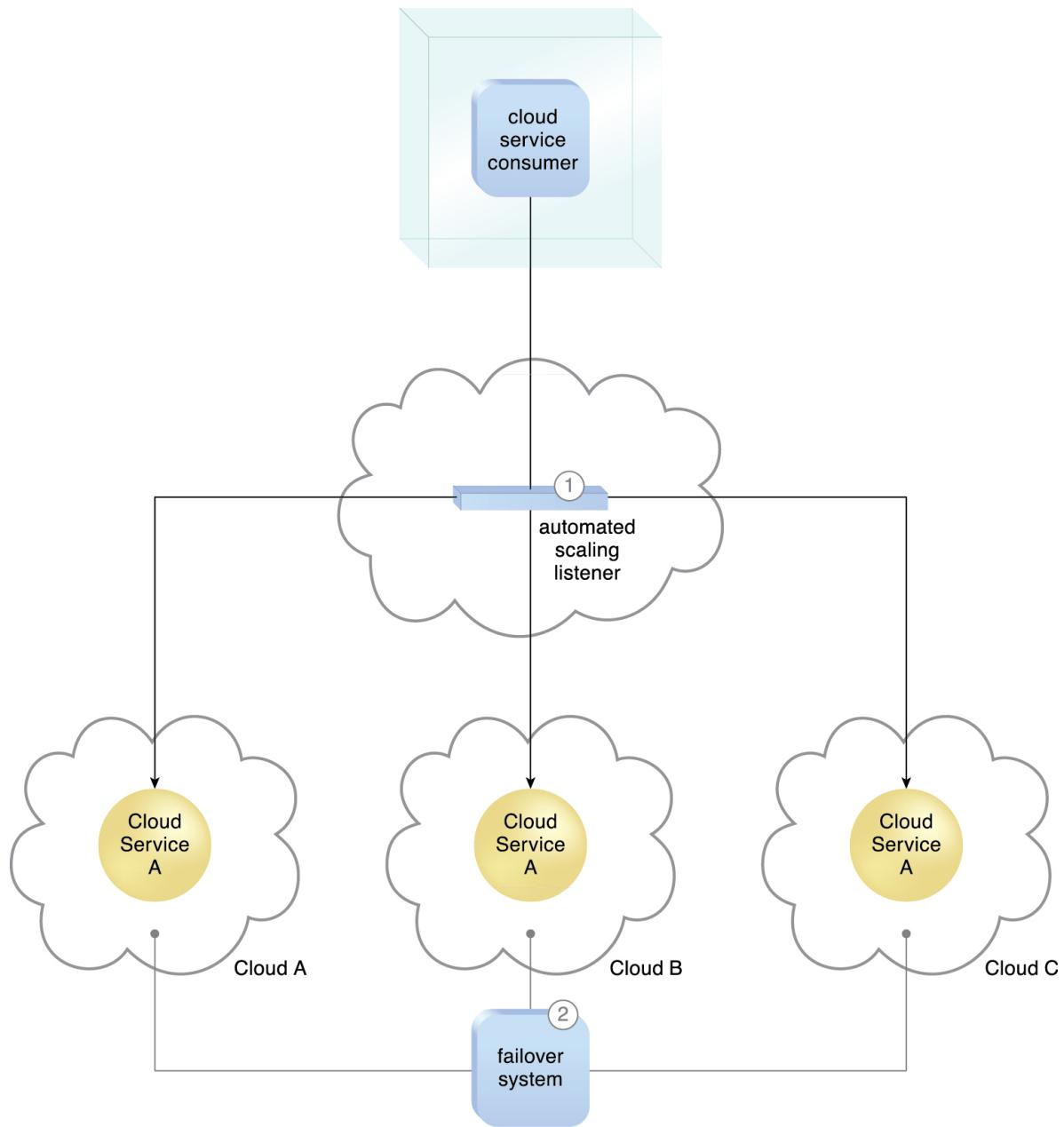


Figura 10.16 Un escucha de escalado automatizado controla el proceso de balanceo de la nube al enrutar las solicitudes de los consumidores del servicio en la nube a implementaciones redundantes del Servicio en la nube A distribuidas en varias nubes (1). El sistema de failover incopora resiliencia dentro de esta arquitectura al proporcionar failover entre nubes (2).

Como punto de partida, los dos mecanismos se utilizan de la siguiente manera:

- El escucha de escalado automatizado redirige las solicitudes de los consumidores de servicios en la nube a una de varias implementaciones de recursos de TI redundantes, según los requisitos actuales de escalamiento y rendimiento.
- El sistema de failover garantiza que los recursos de TI redundantes sean capaces de realizar un failover entre nubes en caso de falla dentro de un recurso de TI o su entorno de alojamiento subyacente. Las fallas de los recursos de TI se anuncian para que el oyente de escalado automatizado pueda evitar enrutar inadvertidamente las solicitudes de los consumidores del servicio en la nube a recursos de TI inestables o no disponibles.

Para que una arquitectura de balanceo en la nube funcione de manera efectiva, el oyente de escalado automatizado debe conocer todas las implementaciones de recursos de TI redundantes dentro del alcance de la arquitectura balanceada en la nube.

Tenga en cuenta que, si la sincronización manual de las implementaciones de recursos de TI entre nubes no es posible, es posible que deba incorporarse el mecanismo de replicación de recursos para automatizar la sincronización.

10.6. Arquitectura de reserva de recursos

Según cómo se diseñen los recursos de TI para el uso compartido y según sus niveles de capacidad disponibles, el acceso simultáneo puede dar lugar a una condición de excepción en tiempo de ejecución denominada *restricción de recursos*. Una restricción de recursos es una condición que se produce cuando se han asignado dos o más consumidores de la nube para compartir un recurso de TI que no tiene la capacidad para adaptarse a los requisitos de procesamiento total de los consumidores de la nube. Como resultado, uno o más de los consumidores de la nube encuentran un rendimiento degradado o pueden ser rechazados por completo. El servicio en la nube en sí puede dejar de funcionar, lo que provocaría el rechazo de todos los consumidores de la nube.

Pueden ocurrir otros tipos de conflictos de tiempo de ejecución cuando diferentes consumidores de servicios en la nube acceden simultáneamente a un recurso de TI (especialmente uno que no está diseñado específicamente para permitir el uso compartido). Por ejemplo, los pools de recursos anidados y hermanos introducen la noción de préstamo de recursos, mediante el cual un pool puede tomar prestados temporalmente recursos de TI de otros pools. Se puede desencadenar un conflicto de tiempo de ejecución cuando el recurso de TI prestado no se devuelve debido a un uso prolongado por parte del consumidor del servicio en la nube que lo está tomando prestado. Esto puede conducir inevitablemente a la aparición de limitaciones de recursos. La arquitectura de reserva de recursos establece un sistema mediante el cual uno de los siguientes se reserva exclusivamente para un consumidor de nube determinado (Figuras 10.17 a 10.19):

- recurso único de TI
- parte de un recurso de TI
- varios recursos de TI

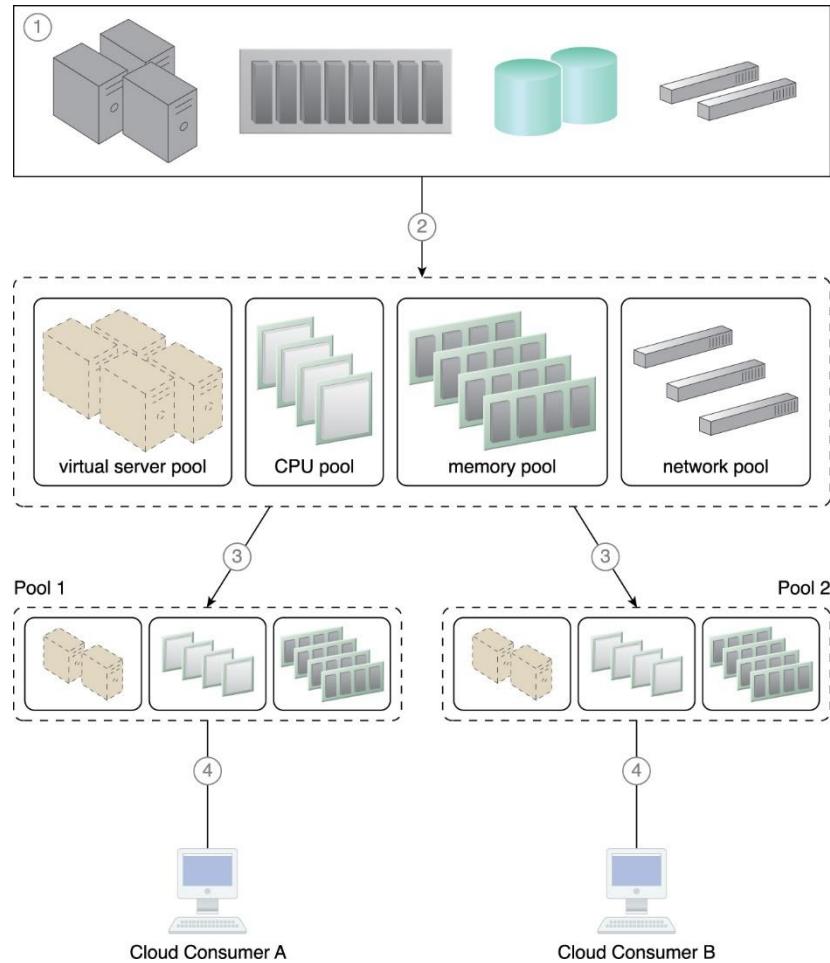


Figura 10.17 Se crea un grupo de recursos físicos (1), a partir del cual se crea un pool de recursos principal según la arquitectura de pooling de recursos (2). Se crean dos pools secundarios más pequeños a partir del pool de recursos principal y los límites de recursos se definen mediante el sistema de gestión de recursos (3). Los consumidores de la nube tienen acceso a sus propios pools de recursos exclusivos (4).

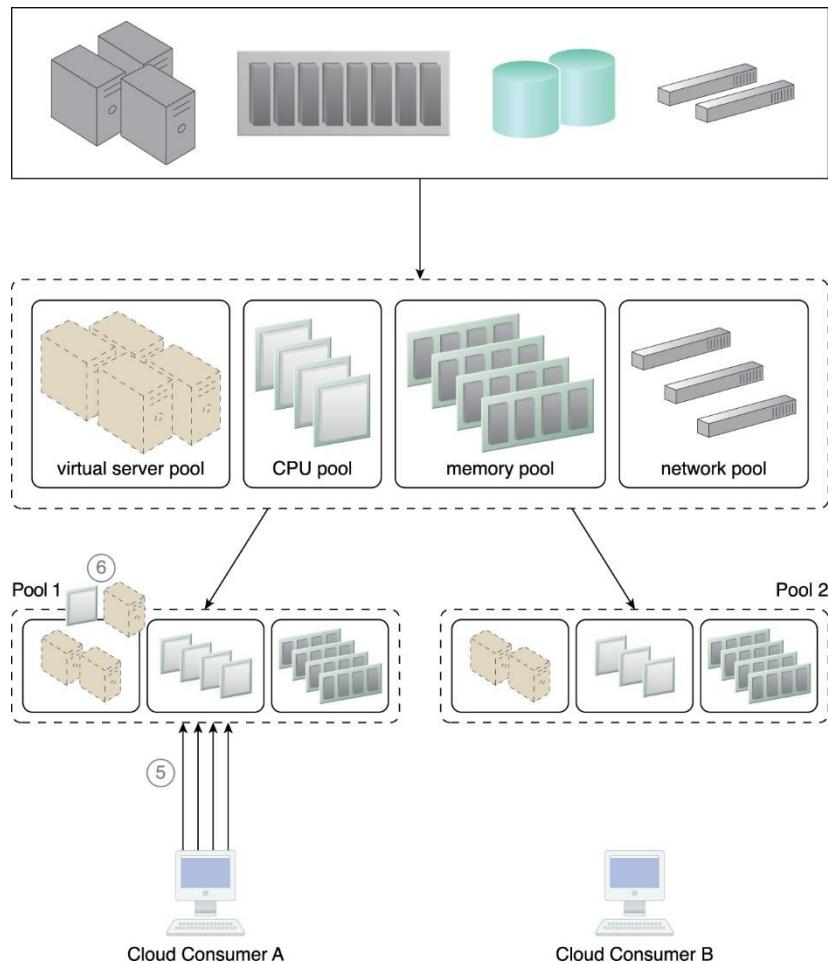


Figura 10.18 Un aumento en las solicitudes del Consumidor de la nube A da como resultado que se asignen más recursos de TI a ese consumidor de la nube (5), lo que significa que algunos recursos de TI deben tomarse prestados del Pool 2. La cantidad de recursos de TI prestados está limitada por el límite de recursos que se definió en el Paso 3, para garantizar que el Consumidor de la nube B no enfrente ninguna restricción de recursos (6).

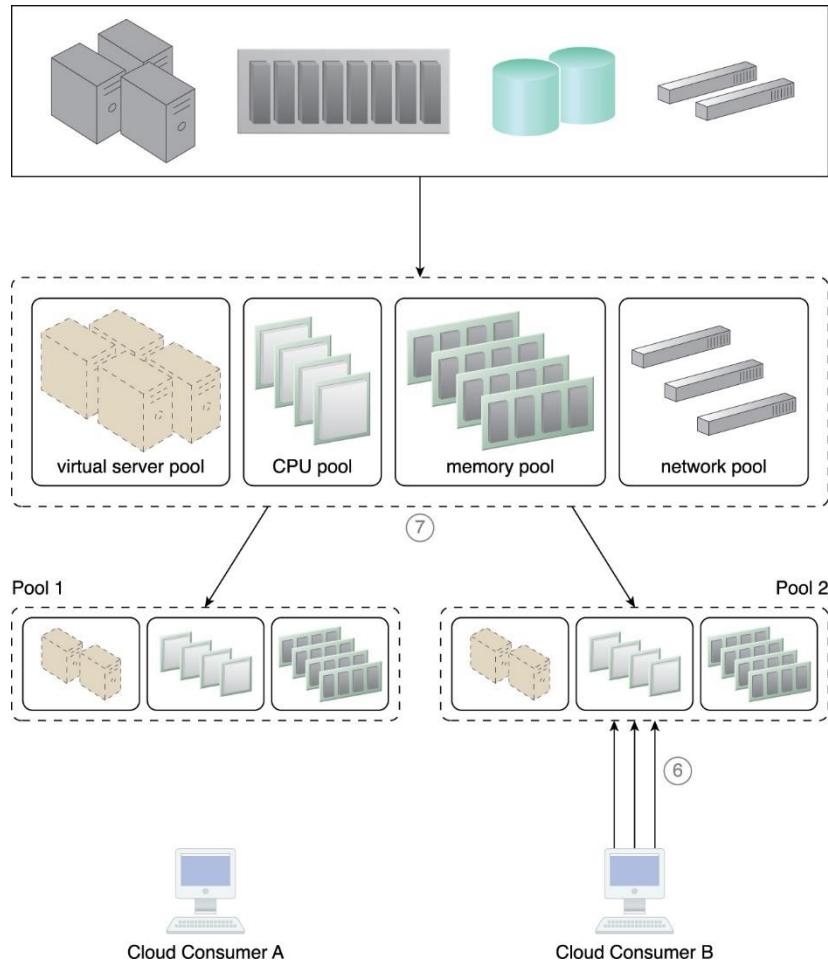


Figura 10.19 El Consumidor de nube B ahora impone más solicitudes y demandas de uso y es posible que pronto necesite utilizar todos los recursos de TI disponibles en el pool (6). El sistema de administración de recursos fuerza al Pool 1 a liberar los recursos de TI y los vuelve a mover al Grupo 2 para que estén disponibles para el Consumidor de la nube B (7).

Esto protege a los consumidores de la nube entre sí al evitar la restricción de recursos y las condiciones de préstamo de recursos antes mencionadas.

La creación de un sistema de reserva de recursos de TI puede requerir la participación del mecanismo del sistema de gestión de recursos, que se utiliza para definir los umbrales de uso para recursos de TI individuales y pools de recursos. Las reservas bloquean la cantidad de recursos de TI que cada pool necesita mantener, con el saldo de los recursos de TI del pool aún disponible para compartir y tomar prestado. El mecanismo del sistema de administración remota también se utiliza para permitir la personalización mediante front-end, de modo que los consumidores de la nube tengan controles de administración para la gestión de sus asignaciones de recursos de TI reservados.

Los tipos de mecanismos que comúnmente se reservan dentro de esta arquitectura son los dispositivos de almacenamiento en la nube y los servidores virtuales. Otros mecanismos que pueden ser parte de la arquitectura pueden incluir:

- *Audit Monitor* - El monitor de auditoría se usa para verificar si el sistema de reserva de recursos cumple con la auditoría del consumidor de la nube, la privacidad y otros requisitos reglamentarios. Por ejemplo, puede rastrear la ubicación geográfica de recursos de TI reservados
- *Monitor de uso de la nube* - Un monitor de uso de la nube puede supervisar los umbrales que activan la asignación de recursos de TI reservados.
- *Hipervisor* - El mecanismo del hipervisor puede aplicar reservas para diferentes consumidores de la nube para garantizar que se asignen correctamente a sus recursos de TI garantizados.
- *Perímetro de red lógica* - Este mecanismo establece los límites necesarios para garantizar que los recursos de TI reservados estén disponibles exclusivamente para los consumidores de la nube.
- *Replicación de recursos* - Este componente debe mantenerse informado sobre los límites de cada consumidor de nube para el consumo de recursos de TI, a fin de replicar y aprovisionar nuevas instancias de recursos de TI de manera conveniente.

10.7. Arquitectura de recuperación y detección dinámica de fallas

Los entornos basados en la nube pueden estar compuestos por grandes cantidades de recursos de TI a los que acceden simultáneamente numerosos consumidores de la nube. Cualquiera de esos recursos de TI puede experimentar condiciones de falla que requieren más que una intervención manual para resolverse. Administrar y resolver manualmente las fallas de los recursos de TI generalmente es ineficiente y poco práctico.

La arquitectura dinámica de detección y recuperación de fallas establece un sistema watchdog resiliente para monitorear y responder a una amplia gama de escenarios de falla predefinidos (Figuras 10.20 y 10.21). Este sistema notifica y escala las condiciones de falla que no puede resolver automáticamente. Se basa en un monitor de uso de la nube especializado llamado monitor de vigilancia inteligente para rastrear activamente los recursos de TI y tomar acciones predefinidas en respuesta a eventos predefinidos.

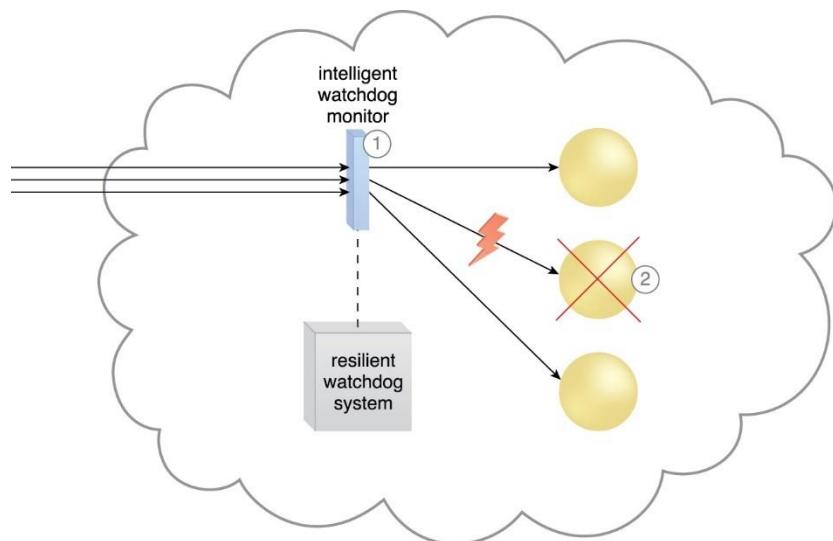


Figura 10.20 El monitor de watchdog inteligente realiza un seguimiento de las solicitudes de los consumidores de la nube (1) y detecta que un servicio de la nube ha fallado (2).

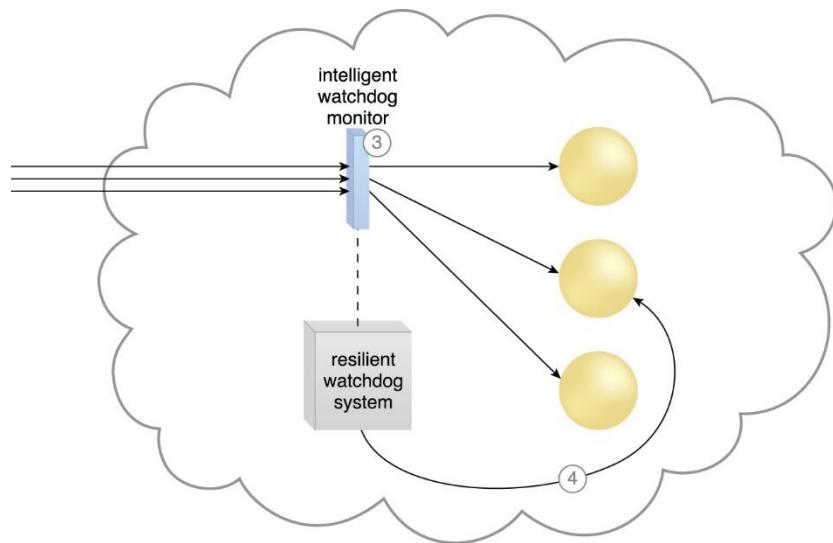


Figura 10.21 El monitor de watchdog inteligente notifica al sistema de watchdog (3), que restaura el servicio en la nube según políticas predefinidas. El servicio en la nube reanuda su funcionamiento en tiempo de ejecución (4).

El sistema de watchdog resiliente realiza las siguientes cinco funciones principales:

- observar
- decidir sobre un evento
- actuar sobre un evento
- generar informes
- escalar

Se pueden definir políticas de recuperación secuencial para cada recurso de TI para determinar los pasos que debe seguir el monitor de watchdog inteligente cuando ocurre una condición de falla. Por ejemplo, una política de recuperación puede establecer que se debe realizar automáticamente un intento de recuperación antes de emitir una notificación (Figura 10.22).

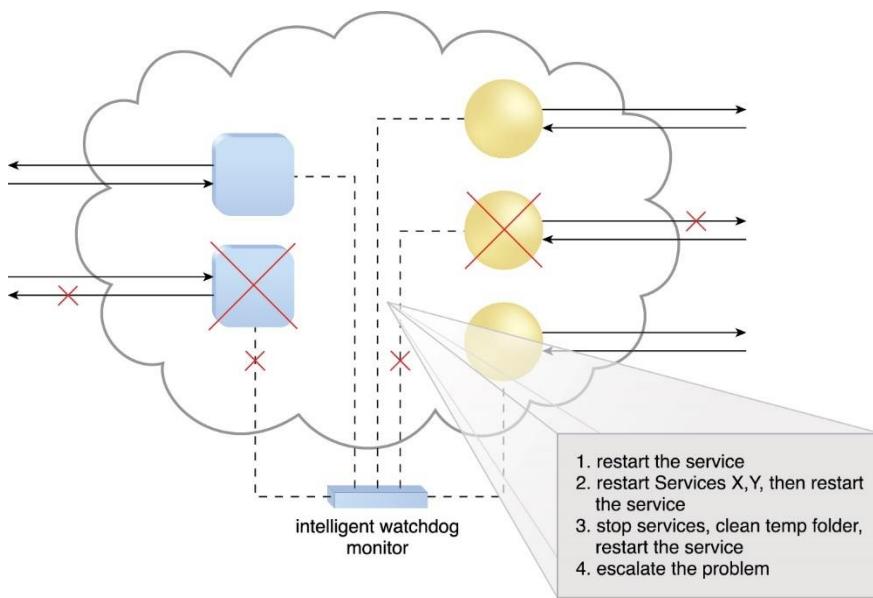


Figura 10.22 En caso de falla, el monitor de watchdog inteligente recurre a sus políticas predefinidas para recuperar el servicio en la nube paso a paso, escalando el proceso cuando el problema resulta ser más profundo de lo esperado.

Algunas de las acciones que el monitor de vigilancia inteligente toma comúnmente para escalar un problema incluyen:

- ejecutar un archivo batch
- enviar un mensaje de consola
- enviar un mensaje de texto
- enviar un mensaje de correo electrónico
- enviar un trap SNMP
- registrar un ticket

Hay una variedad de programas y productos que pueden actuar como monitores de vigilancia inteligentes. La mayoría se puede integrar con sistemas estándar de gestión de eventos y ticketing.

Este modelo arquitectónico puede incorporar además los siguientes mecanismos:

- *Monitor de auditoría* - Este mecanismo se utiliza para rastrear si la recuperación de datos se lleva a cabo de conformidad con los requerimientos legales o de políticas.
- *Sistema Failover* - El mecanismo del sistema failover se suele utilizar durante los intentos iniciales de recuperar los recursos de TI fallidos.
- *Sistema de Gestión SLA y Monitor SLA* - Dado que la funcionalidad lograda mediante la aplicación de esta arquitectura está estrechamente asociada con las garantías SLA, el sistema comúnmente se basa en la información que es administrada y procesada por estos mecanismos.

10.8. Arquitectura de aprovisionamiento Bare-Metal

El aprovisionamiento remoto de servidores es habitual porque el software de gestión remota suele ser nativo del sistema operativo de la mayoría de los servidores físicos. Sin embargo, el acceso a los programas de administración remota convencionales no está disponible para los servidores *bare-metal* (servidores físicos que no tienen sistemas operativos preinstalados ni ningún otro software).

La mayoría de los servidores físicos contemporáneos brindan la opción de instalar soporte de administración remota en la ROM del servidor. Esto lo ofrecen algunos proveedores a través de una tarjeta de expansión, mientras que otros tienen los componentes ya integrados en el conjunto de chips. La arquitectura de aprovisionamiento bare-metal establece un sistema que utiliza esta característica con agentes de servicio especializados, que se utilizan para descubrir y aprovisionar de manera efectiva sistemas completamente operativos de forma remota.

El software de administración remota que está integrado con la ROM del servidor está disponible al iniciarse el servidor. Una interfaz de usuario patentada o basada en la web, como el portal proporcionado por el sistema de administración remota, generalmente se usa para conectarse a la interfaz de administración remota nativa de los servidores físicos. La dirección IP de la interfaz de administración remota se puede configurar manualmente, a través de la IP predeterminada, o alternativamente establecer a través de la configuración de un servicio DHCP. Las direcciones IP en las plataformas IaaS se pueden reenviar directamente a los consumidores de la nube para que puedan realizar instalaciones bare-metal del sistema operativo de forma independiente.

Aunque el software de administración remota se usa para permitir conexiones a consolas de servidores físicos e implementar sistemas operativos, hay dos preocupaciones comunes acerca de su uso:

- La implementación manual en múltiples servidores puede ser vulnerable a errores humanos y de configuración involuntarios.
- El software de administración remota puede consumir mucho tiempo y requerir un procesamiento de recursos de TI de tiempo de ejecución significativo.

El sistema de aprovisionamiento bare-metal aborda estos problemas mediante el uso de los siguientes componentes:

- *Discovery Agent* - Un tipo de agente de supervisión que busca y encuentra servidores físicos disponibles para asignarlos a los consumidores de la nube.
- *Deployment Agent* - Un agente de administración que se instala en la memoria de un servidor físico, para posicionarse como un cliente para el sistema de implementación de aprovisionamiento básico.
- *Discovery Section* - Un componente de software que escanea la red y localiza los servidores físicos disponibles con los que conectarse.
- *Management Loader* - El componente que se conecta al servidor físico y carga las opciones de administración para el consumidor de la nube.

- *Deployment Component* - El componente responsable de instalar el sistema operativo en los servidores físicos seleccionados.

El sistema de aprovisionamiento bare-metal proporciona una función de implementación automática que permite a los consumidores de la nube conectarse al software de implementación y aprovisionar más de un servidor o sistema operativo al mismo tiempo. El sistema de implementación central se conecta a los servidores a través de sus interfaces de administración y utiliza el mismo protocolo para cargar y operar como agente en la memoria RAM del servidor físico. Luego, el servidor bare-metal se convierte en un cliente raw con un agente de administración instalado y el software de implementación carga los archivos de configuración necesarios para implementar el sistema operativo (Figuras 10.23 y 10.24).

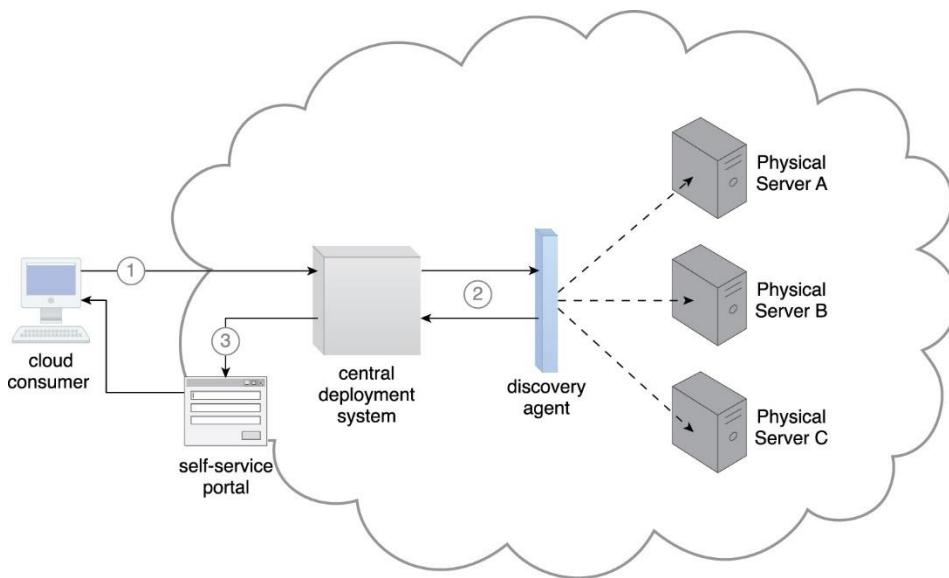


Figura 10.23 El consumidor de la nube se conecta a la solución de implementación (1) para realizar una búsqueda mediante el agente de descubrimiento (2). Los servidores físicos disponibles se muestran al consumidor de la nube (3).

Las imágenes de implementación, la automatización de la implementación del sistema operativo o la implementación desatendida y los scripts de configuración posteriores a la instalación se pueden usar a través del motor de automatización inteligente y el portal de autoservicio para extender esta funcionalidad.

Los siguientes mecanismos adicionales pueden formar parte de esta arquitectura:

- *Dispositivo de almacenamiento en la nube* - Este mecanismo almacena plantillas del sistema operativo y archivos de instalación, así como agentes de implementación y paquetes de implementación para el sistema de aprovisionamiento.
- *Hipervisor* - Puede ser necesaria la implementación de hipervisores en servidores físicos como parte del aprovisionamiento del sistema operativo.
- *Perímetro de red lógica* - Los límites del perímetro de red lógica ayudan a garantizar que solo los consumidores autorizados de la nube puedan acceder a los servidores físicos raw.

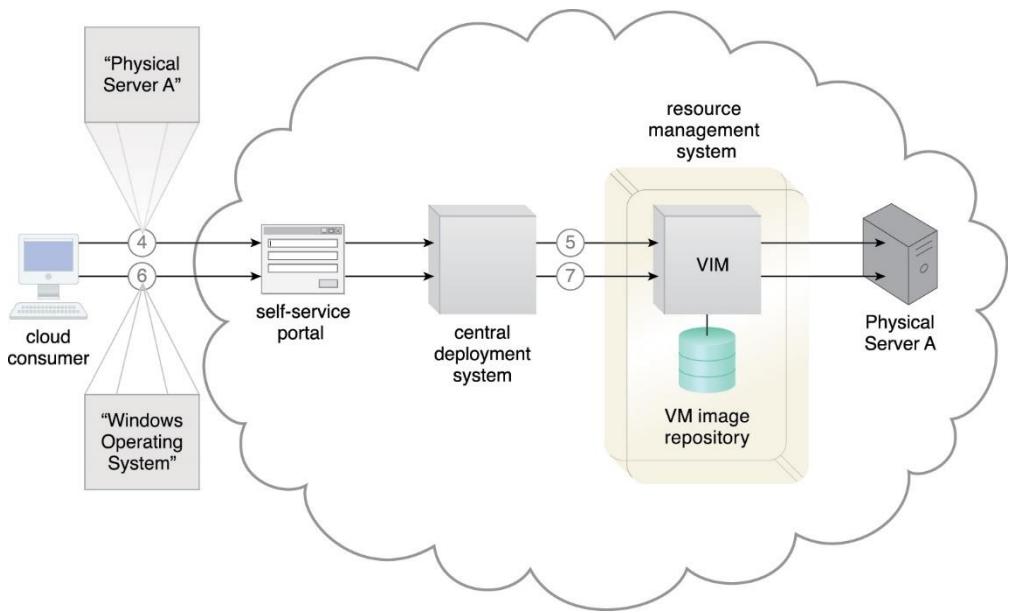


Figura 10.24 El consumidor de la nube selecciona un servidor físico para aprovisionar (4). El agente de implementación se carga en la memoria RAM del servidor físico a través del sistema de administración remota (5). El consumidor de la nube selecciona un sistema operativo y un método de configuración a través de la solución de implementación (6). El sistema operativo está instalado y el servidor se vuelve operativo (7).

- *Replicación de recursos* - Este mecanismo se implementa para la replicación de recursos de TI mediante la implementación de un nuevo hipervisor en un servidor físico para equilibrar la carga de trabajo del hipervisor durante o después del aprovisionamiento.
- *Sistema de gestión de SLA* - Este sistema de gestión garantiza que la disponibilidad de los servidores físicos esté de acuerdo con las estipulaciones de SLA predefinidas.

10.9. Arquitectura de aprovisionamiento rápido

Un proceso de aprovisionamiento convencional puede implicar una serie de tareas que tradicionalmente los administradores y expertos en tecnología realizan manualmente mediante la preparación de los recursos de TI solicitados según las especificaciones pre empaquetadas o las solicitudes personalizadas del cliente. En entornos de nube, donde se atiende a un mayor volumen de clientes y donde el cliente promedio solicita mayores volúmenes de recursos de TI, los procesos de aprovisionamiento manual son inadecuados e incluso pueden generar un riesgo irrazonable debido a errores humanos y tiempos de respuesta inefficientes.

Por ejemplo, un consumidor de nube que solicita la instalación, configuración y actualización de veinticinco servidores Windows con varias aplicaciones requiere que la mitad de las aplicaciones sean instalaciones idénticas, mientras que la otra mitad se personalice. Cada implementación del sistema operativo puede demorar hasta 30 minutos, seguido de tiempo adicional para los parches de seguridad y las actualizaciones del sistema operativo que requieren reiniciar el servidor. Las aplicaciones finalmente deben implementarse y configurarse. El uso de un enfoque manual o semiautomático requiere cantidades excesivas de tiempo e introduce una probabilidad de error humano que aumenta con cada instalación.

La arquitectura de aprovisionamiento rápido establece un sistema que automatiza el aprovisionamiento de una amplia gama de recursos de TI, ya sea de forma individual o colectiva. La arquitectura de tecnología subyacente para el aprovisionamiento rápido de recursos de TI puede ser sofisticada y compleja, y se basa en un sistema compuesto por un programa de aprovisionamiento automatizado, un motor de aprovisionamiento rápido y scripts y plantillas para el aprovisionamiento bajo demanda.

Más allá de los componentes que se muestran en la Figura 10.25, hay muchos artefactos arquitectónicos adicionales disponibles para coordinar y automatizar los diferentes aspectos del aprovisionamiento de recursos de TI, como:

- *Server Templates* - Plantillas de archivos de imágenes virtuales que se utilizan para automatizar la instancia de nuevos servidores virtuales.

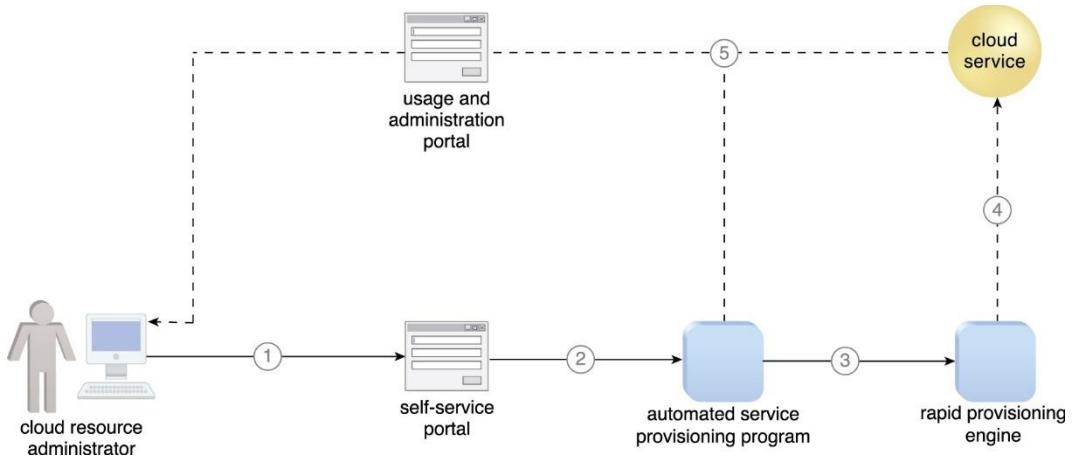


Figura 10.25 Un administrador de recursos en la nube solicita un nuevo servicio en la nube a través del portal de autoservicio (1). El portal de autoservicio pasa la solicitud al programa de aprovisionamiento de servicios automatizado instalado en el servidor virtual (2), que pasa las tareas necesarias a realizar al motor de aprovisionamiento rápido (3). El motor de aprovisionamiento rápido anuncia cuando el nuevo servicio en la nube está listo (4). El programa de aprovisionamiento de servicios automatizado finaliza y publica el servicio en la nube en el portal de uso y administración para el acceso del consumidor en la nube (5).

- *Imágenes de servidor* - Estas imágenes son similares a los plantillas de servidor virtual, pero se utilizan para aprovisionar servidores físicos.
- *Paquetes de aplicaciones* - Colecciones de aplicaciones y otro software que se empaquetan para la implementación automatizada.
- *Application Packager* - El software utilizado para crear paquetes de aplicaciones.
- *Scripts personalizados* - Scripts que automatizan las tareas administrativas, como parte de un motor de automatización inteligente.
- *Administrador de secuencias* - Un programa que organiza secuencias de tareas de aprovisionamiento automatizadas.
- *Registrador de secuencias* - Un componente que registra la ejecución de secuencias de tareas de aprovisionamiento automatizado.

- *Operating System Baseline* – Un template de configuración que se aplica después de instalar el sistema operativo para prepararlo rápidamente para su uso.
- *Application Configuration Baseline* – Un template de configuración con la configuración y los parámetros ambientales que se necesitan para preparar nuevas aplicaciones para su uso.
- *Deployment Data Store* - El repositorio que almacena imágenes virtuales, plantillas, scripts, configuraciones baseline y otros datos relacionados.

La siguiente descripción paso a paso ayuda a proporcionar una idea del funcionamiento interno de un motor de aprovisionamiento rápido, que involucra varios de los componentes del sistema enumerados anteriormente:

- 1.** Un consumidor de la nube solicita un nuevo servidor a través del portal de autoservicio
- 2.** El administrador de secuencias reenvía la solicitud al motor de implementación para la preparación de un sistema operativo.
- 3.** El motor de implementación utiliza las plantillas de servidor virtual para el aprovisionamiento si la solicitud es para un servidor virtual. De lo contrario, el motor de implementación envía la solicitud para aprovisionar un servidor físico.
- 4.** La imagen predefinida para el tipo de sistema operativo solicitado se utiliza para el aprovisionamiento del sistema operativo, si está disponible. De lo contrario, se ejecuta el proceso de implementación regular para instalar el sistema operativo.
- 5.** El motor de implementación informa al administrador de secuencias cuando el sistema operativo está listo.
- 6.** El administrador de secuencias actualiza y envía los registros al registrador de secuencias para su almacenamiento.
- 7.** El administrador de secuencias solicita que el motor de implementación aplique el baseline³⁴ del sistema operativo al sistema operativo aprovisionado.
- 8.** El motor de implementación aplica el baseline del sistema operativo solicitado.
- 9.** El motor de implementación informa al administrador de secuencias que se ha aplicado el baseline del sistema operativo.
- 10.** El administrador de secuencias actualiza y envía los registros de los pasos completados al registrador de secuencias para su almacenamiento.
- 11.** El administrador de secuencias solicita que el motor de implementación instale las aplicaciones.
- 12.** El motor de implementación implementa las aplicaciones en el servidor aprovisionado.

³⁴ Una línea base o baseline según el estándar de la IEEE es una especificación o producto que ha sido revisado formalmente, sobre el que se ha llegado a un acuerdo, y que de ahí en adelante servirá como base para un desarrollo posterior que puede cambiarse solamente a través de procedimientos formales de control de cambios. Fuente: Wikipedia.

13. El motor de implementación informa al administrador de secuencias que se han instalado las aplicaciones.

14. El administrador de secuencias actualiza y envía los registros de los pasos completados al registrador de secuencias para su almacenamiento.

15. El administrador de secuencias solicita que el motor de implementación aplique el baseline de configuración de la aplicación.

16. El motor de implementación aplica el baseline de configuración.

17. El motor de implementación informa al administrador de secuencias que se ha aplicado el baseline de configuración.

18. El administrador de secuencias actualiza y envía los registros de los pasos completados al registrador de secuencias para su almacenamiento.

El mecanismo del dispositivo de almacenamiento en la nube se utiliza para proporcionar almacenamiento para la información básica de la aplicación, las plantillas y los scripts, mientras que el hipervisor crea, implementa y aloja rápidamente los servidores virtuales que se aprovisionan ellos mismos o alojan otros recursos de TI aprovisionados. El mecanismo de replicación de recursos generalmente se usa para generar instancias replicadas de recursos de TI en respuesta a requisitos de aprovisionamiento rápido.

[**10.10. Arquitectura de gestión de carga de trabajo de almacenamiento**](#)

Los dispositivos de almacenamiento en la nube sobreutilizados aumentan la carga de trabajo en el controlador de almacenamiento y pueden causar una variedad de desafíos de rendimiento. Por el contrario, los dispositivos de almacenamiento en la nube que están infrautilizados son un desperdicio debido a la pérdida de procesamiento y capacidad de almacenamiento potencial (Figura 10.26).

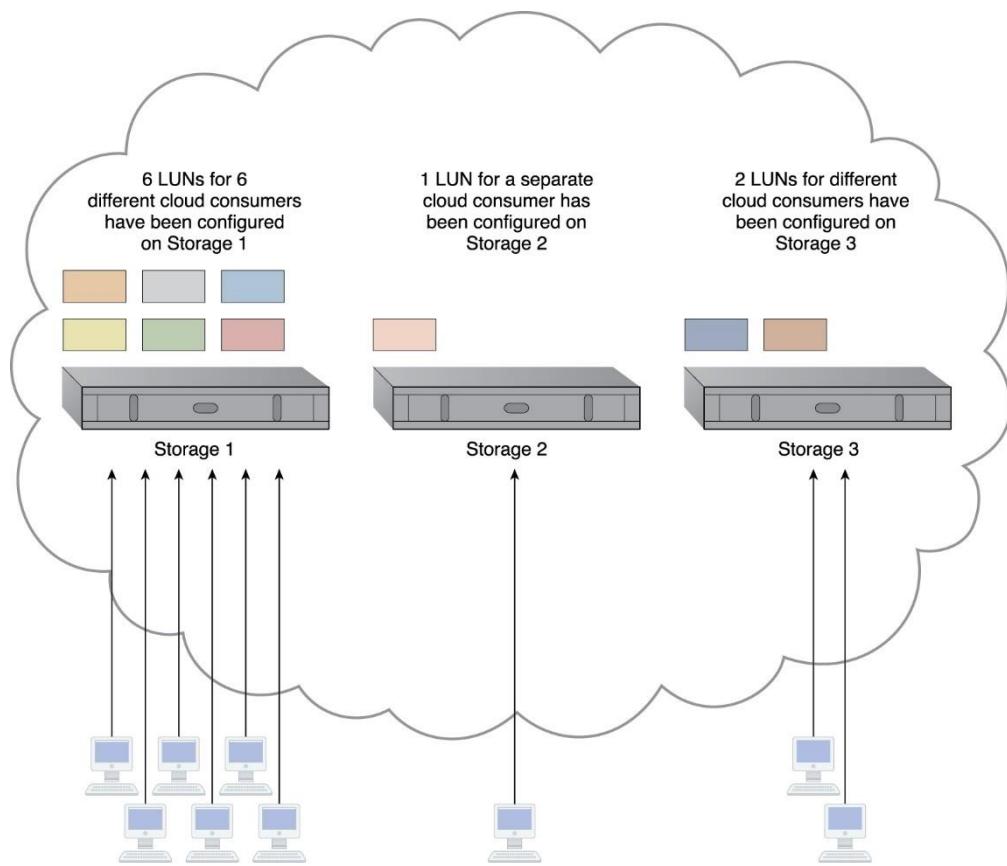
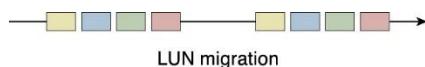


Figura 10.26 Una arquitectura de almacenamiento en la nube desequilibrada tiene seis LUNs de almacenamiento en el Almacenamiento 1 para que los utilicen los consumidores de la nube, mientras que el Almacenamiento 2 aloja un LUN y el Almacenamiento 3 aloja dos. La mayor parte de la carga de trabajo termina en el Almacenamiento 1, ya que aloja la mayoría de los LUNs.

Migración de LUN

La migración de LUN es un programa de almacenamiento especializado que se utiliza para mover LUNs de un dispositivo de almacenamiento a otro sin interrupción, sin dejar de ser transparente para los consumidores de la nube.



La *arquitectura de gestión de cargas de trabajo de almacenamiento* permite que los LUN se distribuyan uniformemente entre los dispositivos de almacenamiento en la nube disponibles, mientras que se establece un sistema de capacidad de almacenamiento para garantizar que las cargas de trabajo en tiempo de ejecución se distribuyan uniformemente entre los LUNs (Figura 10.27).

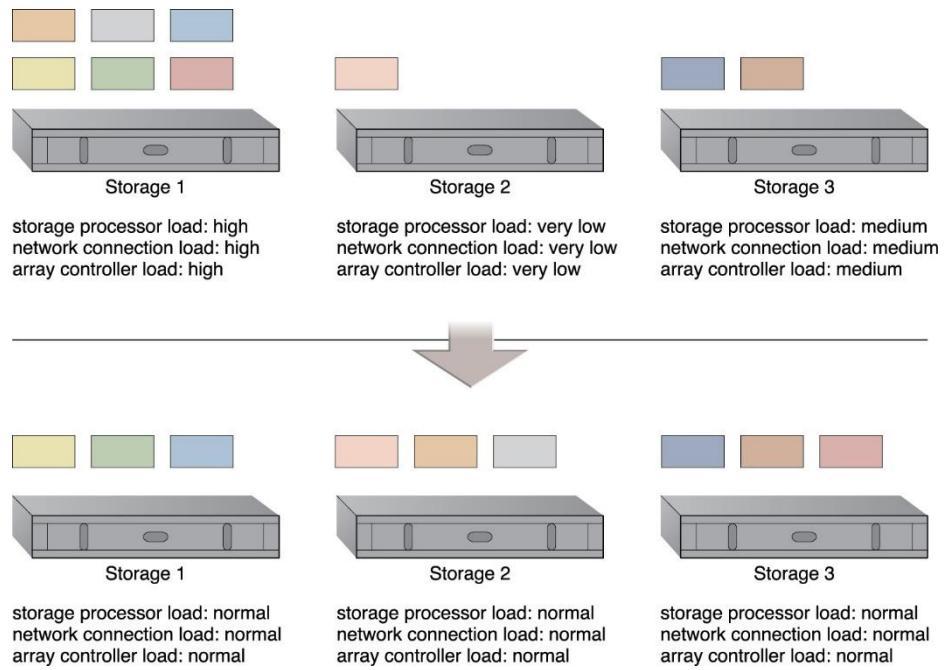


Figura 10.27 Los LUN se distribuyen dinámicamente entre los dispositivos de almacenamiento en la nube, lo que da como resultado una distribución más uniforme de los tipos de cargas de trabajo asociados.

La combinación de dispositivos de almacenamiento en la nube en un grupo permite que los datos de LUN se distribuyan por igual entre los hosts de almacenamiento disponibles. Se configura un sistema de administración de almacenamiento y se coloca un oyente de escalado automatizado para monitorear e igualar las cargas de trabajo en tiempo de ejecución entre los dispositivos de almacenamiento en la nube agrupados, como se ilustra en las Figuras 10.28 a 10.30.

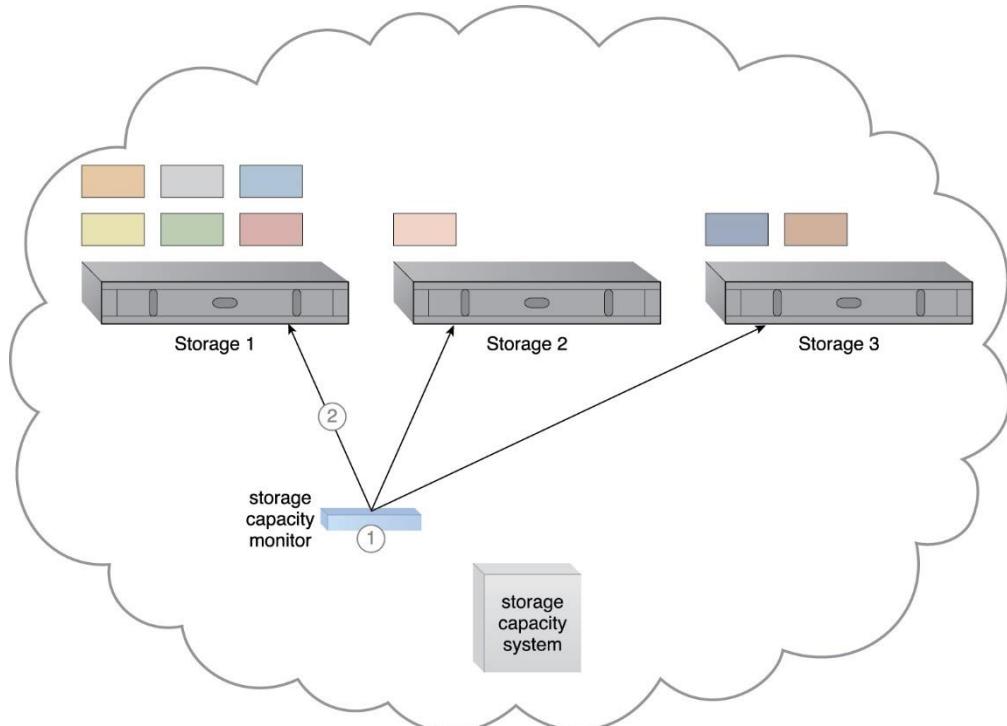


Figura 10.28 El sistema de capacidad de almacenamiento y el monitor de capacidad de almacenamiento están configurados para inspeccionar tres dispositivos de almacenamiento en tiempo real, cuya carga de trabajo y umbrales de capacidad están predefinidos (1). El monitor de capacidad de almacenamiento determina que la carga de trabajo en Almacenamiento 1 está alcanzando su umbral (2).

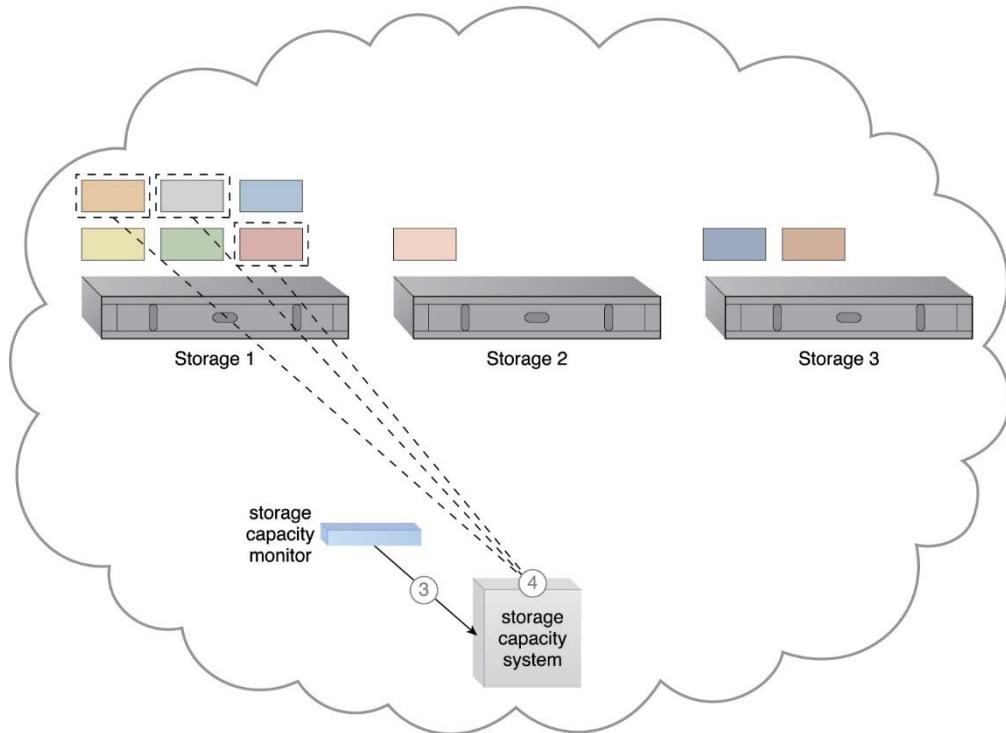


Figura 10.29 El monitor de capacidad de almacenamiento informa al sistema de capacidad de almacenamiento que el Almacenamiento 1 está sobreutilizado (3). El sistema de capacidad de almacenamiento identifica los LUNs que se moverán desde el almacenamiento 1 (4).

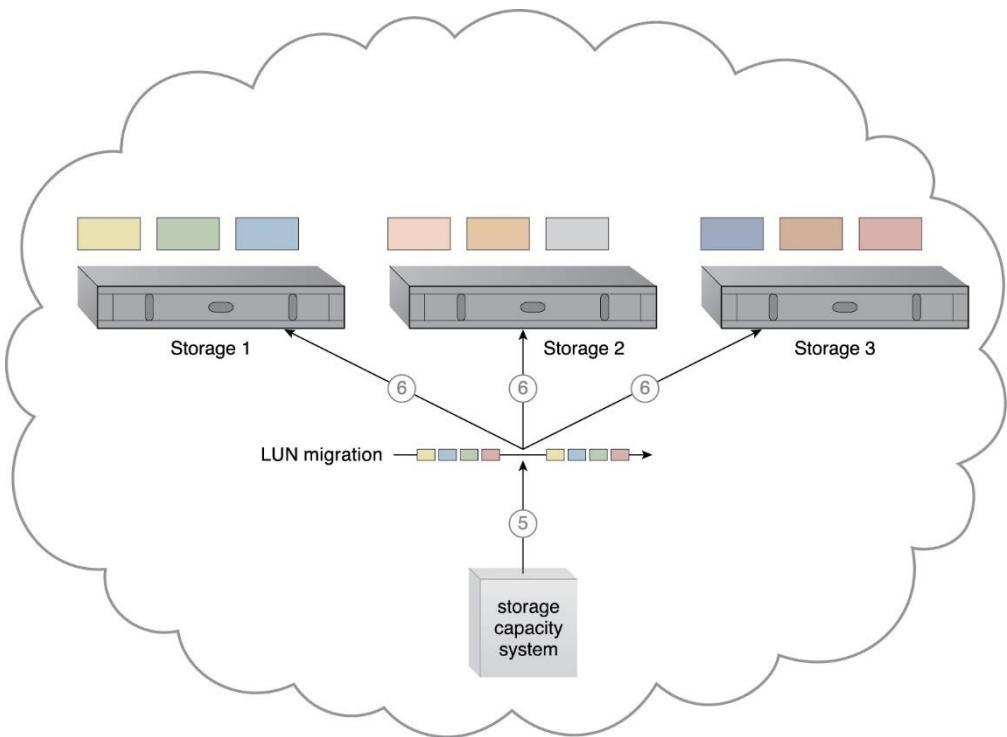


Figura 10.30 El sistema de capacidad de almacenamiento solicita la migración de LUN para mover algunos de los LUNs del almacenamiento 1 a los otros dos dispositivos de almacenamiento (5). La migración de LUN transfiere los LUNs al almacenamiento 2 y 3 para equilibrar la carga de trabajo (6).

El sistema de capacidad de almacenamiento puede mantener el dispositivo de almacenamiento en modo de ahorro de energía durante los períodos en los que se accede a los LUN con menor frecuencia o en algunos momentos específicos.

Algunos otros mecanismos que se pueden incluir en la arquitectura de administración de la carga de trabajo de almacenamiento para acompañar al dispositivo de almacenamiento en la nube son los siguientes:

- *Monitor de auditoría* - Este mecanismo de monitoreo se utiliza para verificar el cumplimiento de los requisitos normativos, de privacidad y de seguridad, ya que el sistema establecido por esta arquitectura puede reubicar datos físicamente.
- *Escucha de escalado automatizado* - La escucha de escalado automatizado se utiliza para observar y responder a las fluctuaciones de la carga de trabajo.
- *Monitor de uso de la nube* - Además del monitor de capacidad de carga de trabajo, se utilizan monitores de uso de la nube especializados para realizar un seguimiento de los movimientos de LUNs y recopilar estadísticas de distribución de la carga de trabajo.
- *Load Balancer* - Este mecanismo se puede agregar para equilibrar horizontalmente las cargas de trabajo entre los dispositivos de almacenamiento en la nube disponibles.

- **Perímetro de red lógica** - Los perímetros de red lógica brindan niveles de aislamiento para que los datos del consumidor de la nube que se reubiquen permanezcan inaccesibles para terceros no autorizados.

Ejemplo de Estudio de Caso

Innovartus está alquilando dos entornos basados en la nube de dos proveedores de nube diferentes y tiene la intención de aprovechar esta oportunidad para establecer un piloto de arquitectura de balanceo de nube para su servicio en la nube Role Player.

Después de evaluar sus requisitos con respecto a las nubes respectivas, los arquitectos de la nube de Innovartus producen una especificación de diseño que se basa en que cada nube tenga múltiples implementaciones del servicio de nube. Esta arquitectura incorpora implementaciones separadas de escucha de escalado automatizado y sistema de failover, junto con un mecanismo de balanceo de carga central (Figura 10.31).

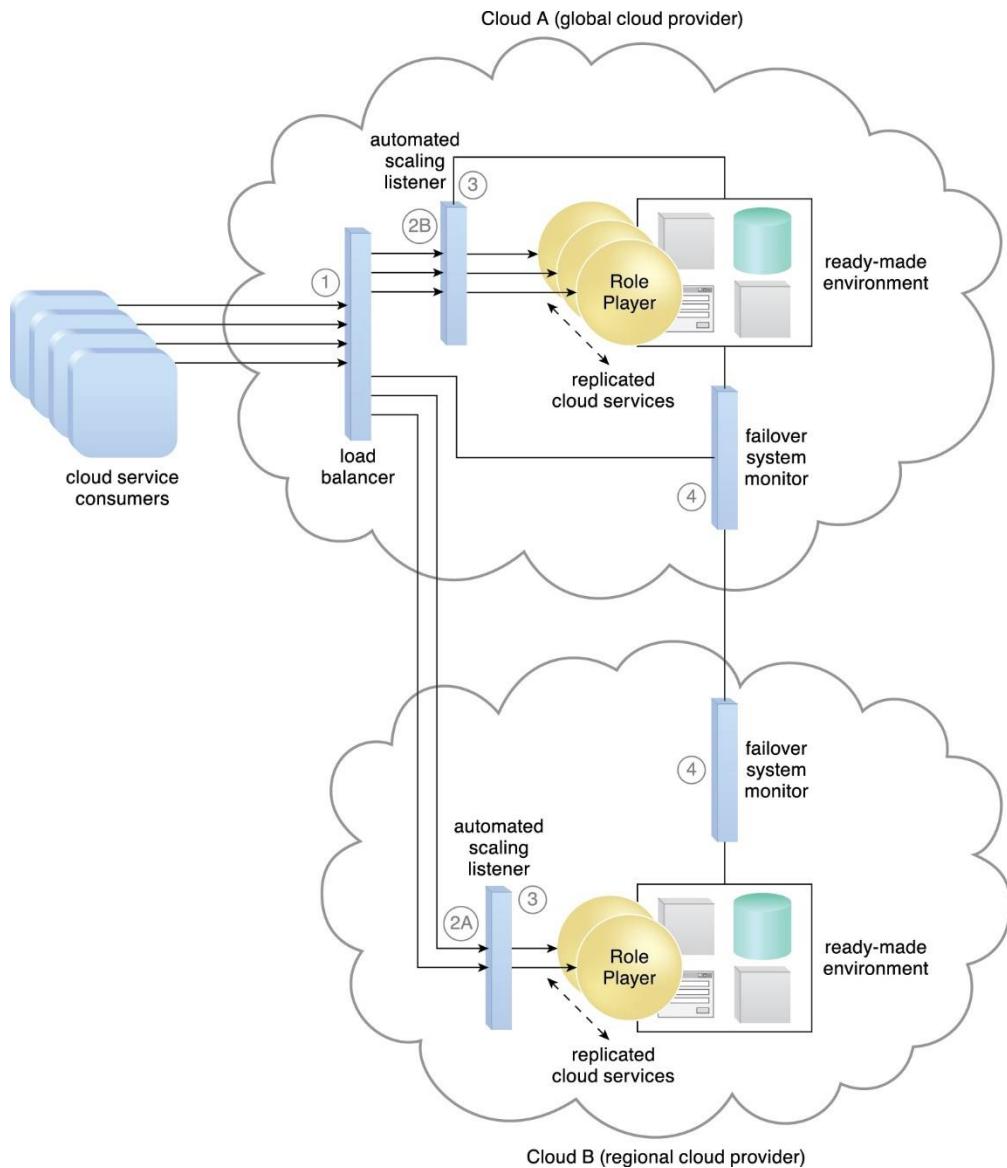


Figura 10.31 Un agente de servicio de balanceo de carga enruta las solicitudes de los consumidores de servicios en la nube de acuerdo con un algoritmo predefinido (1). Las solicitudes son recibidas por el escucha de escalado automatizado local o externo (2A, 2B), que reenvía cada solicitud a una implementación de servicio en la nube (3). Los monitores del sistema de failover se utilizan para detectar y responder a la falla del servicio en la nube (4).

El balanceador de carga distribuye las solicitudes de los consumidores del servicio en la nube a través de las nubes mediante un algoritmo de distribución de la carga de trabajo, mientras que el oyente de escalado automatizado de cada nube enruta las solicitudes a las implementaciones locales del servicio en la nube. Los sistemas de failover pueden realizar la conmutación por error a las implementaciones de servicios en la nube redundantes que se encuentran tanto dentro como entre las nubes. La conmutación por error entre las nubes se lleva a cabo principalmente cuando las implementaciones del servicio de nube local se acercan a sus umbrales de procesamiento, o si una nube se encuentra con una falla grave de la plataforma.

Recomendación:

Parcial 1 - Capítulos 1, 2 y 3

Parcial 2 - Capítulos 4, 5, 6 y 7

Parcial 3 - Capítulos 8, 9 y 10