Guía de implementación y estrategia de Azure

Tercera edición

Información actualizada para organizaciones que usan Azure por primera vez

Peter De Tender, Greg Leonardo y Jason Milgram

www.packt.com

Guía de implementación y estrategia de Azure

Tercera edición

Información actualizada para organizaciones que usan Azure por primera vez

Peter De Tender

Greg Leonardo

Jason Milgram

BIRMINGHAM - BOMBAY

Guía de implementación y estrategia de Azure

Tercera edición

Editor de adquisiciones: Ben Renow-Clarke

Editor de adquisiciones – Reseñas de pares: Suresh Jain

Editor de desarrollo de contenido: Ian Hough

Editora de proyectos: Janice Gonsalves

Editor técnico: Aniket Shetty

Corrector de estilo: Safis Editing

Revisor: Safis Editing

Indexador: Pratik Shirodkar

Diseñador de producción: Sandip Tadge

Primera publicación: enero de 2020

Referencia de producción: 1160120

Publicado por Packt Publishing Ltd.
Livery Place
35 Livery Street
Birmingham B3 2PB, Reino Unido.

ISBN 978-1-83898-668-1

`www.packt.com`

Colaboradores

Acerca de los autores

Peter De Tender es un conocido experto de Azure, un instructor apasionado y dedicado, que siempre logra ofrecer talleres técnicos profundos e inspiradores sobre la plataforma Azure, entretenidos y repletos de demostraciones.

Antes de que Peter se uniera al prestigioso equipo de instructores técnicos de Azure de Microsoft, ocupó un cargo similar en su propia empresa durante los últimos 6 años. Ahora, continúa haciendo lo que más le gusta: perfeccionar las habilidades de los clientes y socios en el maravilloso mundo y las funcionalidades de Azure.

Peter ha sido un **Microsoft Certified Trainer** (**MCT**) por más de 10 años y MVP de Microsoft desde 2013, inicialmente en Windows IT Pro, pero pasó a la categoría Azure en 2015.

Además de la coautoría de este libro, Peter ha publicado otro material orientado a Azure con Packt Publishing, Apress y a través de la autopublicación.

Puede seguir a Peter en Twitter como `@pdtit` o `@007ffflearning`, o revisar su sitio web, `http://www.007ffflearning.com`, para mantenerse al día en sus aventuras de Azure.

Quisiera agradecer a Ben Renow-Clarke del equipo de Packt Publishing por su confianza y dedicación para publicar este libro y acercarse a mí como autor. Lo que inicialmente comenzó como un pequeño trabajo de escritura rápida se amplió a un proyecto completo de creación de alta calidad y una gran cantidad de información. Él también administró el control de calidad para asegurarse de que todo el contenido estuviera lo más actualizado posible durante todo el proceso de escritura.

También me gustaría agradecer a mi esposa Els y a mis dos hijas Kaylee y Kitana. Han tenido que extrañarme muchas veces en los últimos años, porque estaba afuera de viaje en otra ubicación para realizar otro taller de Azure en algún lugar del mundo. E incluso, en el poco tiempo libre que tuve se conformaban con verme ocupando tiempo en otro proyecto paralelo. Todo porque saben lo feliz que me hace poder inspirar a otras personas a usar esta plataforma increíble llamada Azure. Ahora que este proyecto está completo, puedo enfocarme un poco más en todas ustedes nuevamente. Gracias por su comprensión, ¡son tres mujeres increíbles!

Greg Leonardo es actualmente arquitecto de la nube y ayuda a las organizaciones en la adopción de la nube y la innovación. Ha trabajado en la industria de la TI desde su época como militar. Es un veterano, arquitecto, profesor, orador y usuario pionero. También es un arquitecto certificado experto en soluciones de Azure y MVP de Microsoft Azure. A lo largo de su carrera, ha trabajado en muchas facetas de la TI. Es presidente de TampaDev, un encuentro comunitario que organiza #TampaCC, Azure User Group, Azure Medics y diversos eventos tecnológicos en Tampa.

Greg también es el autor del libro *Hands-On Cloud Solutions with Azure* de Packt Publishing.

Quisiera agradecer a mi esposa Kate y a mis dos hijos Maddux y Lucas por apoyarme mientras escribía los capítulos de este libro.

Jason Milgram es el vicepresidente principal y arquitecto de soluciones en la nube del Banco Nacional de la Ciudad de Florida con sede en Miami, Florida. Anteriormente, fue vicepresidente de Arquitectura de Plataformas e Ingeniería de Champion Solutions Group en Boca Ratón, Florida.

Jason se educó en la Universidad de Cincinnati y en el Instituto Tecnológico de Massachusetts, Escuela de Administración y Dirección de Empresas Sloan. También fue sargento en la Reserva del Ejército de los Estados Unidos, en servicio de 1990 a 1998.

Como MVP de Microsoft Azure (de 2010 al presente), Jason ha ofrecido más de 100 presentaciones de Azure y habitualmente escribe artículos sobre temas de Azure.

Acerca del revisor

Steve Buchanan es director y jefe de Midwest Containers Services en el equipo de transformación de la nube/DevOps con Avanade, la sección de Microsoft de Accenture. Ha sido ocho veces MVP de Microsoft y es autor de seis libros técnicos. Se ha presentado en eventos técnicos, incluidos **Midwest Management Summit** (**MMS**), Microsoft Ignite, BITCon, Experts Live Europe, OSCON y grupos de usuarios.

Steve se centra en transformar la posición de TI en un impulsor de la transformación digital a través de ITSM, DevOps y CloudOps. Se mantiene activo en la comunidad técnica y disfruta crear blogs sobre sus aventuras de TI en `www.buchatech.com`.

Créditos

Adam Harbour

Ahmed Sabbour

Alessandro Segala

Enrico Fuiano

Guillermo Gómez

Joachim Hafner

Michael Leworthy

Ori Zohar

Shankar Sivadasan

Simon Schwingel

Índice

1
Comprensión de la nube de Azure

Introducción

Azure es una plataforma eficaz que ofrece una gran cantidad de servicios y funcionalidades para organizaciones de cualquier tamaño que migren a una estrategia de nube. Ya sea que se trate de una empresa nueva que necesita solo la infraestructura básica para ejecutar su sitio web o una multinacional que opera en todo el mundo, cualquier tipo de organización hoy en día puede comenzar a implementar y migrar cargas de trabajo a Azure. El primer paso para aprovechar las diversas funcionalidades que ofrece Azure es una planificación cuidadosa. Podría ser tentador simplemente aplicarnos a la tarea y llevar a cabo la implementación de recursos de Azure. Sin embargo, se debe subrayar que la creación de un entorno de nube puede requerir un nivel de planificación y detalle similar a la creación de sus propios servicios de centro de datos.

En este capítulo, le proporcionamos una visión general clara de las funcionalidades de Azure, sus beneficios y cómo empezar a utilizar la plataforma de forma correcta. También se analiza cómo migrar las cargas de trabajo existentes a la nube y abarca las herramientas que Azure proporciona para agilizar y simplificar este proceso.

Luego, en este capítulo, se presentan asuntos relacionados con el negocio en que la nube puede ayudar en la innovación y la transformación digital, así como en el manejo de la identidad y la seguridad en la nube. Por último, abarca temas relacionados con la infraestructura de TI sobre cómo crear una red preparada para la empresa, cómo migrar los servicios de archivos y qué herramientas están disponibles en Azure para la supervisión y las operaciones diarias.

A partir de la fase de evaluación, en que se le guiará en la preparación para la nube de su organización, este capítulo y, de hecho, todo el libro, lo ayudará a entender la estrategia principal de modernización de aplicaciones con Azure.

Una vez que decida comenzar a implementar y utilizar recursos en Azure, examinaremos los procedimientos recomendados en torno a la planificación de la migración, las herramientas de migración y los procesos que Microsoft pone a su disposición para que este proceso sea lo más fluido posible.

Además de la migración de la carga de trabajo, necesita comprender la implementación y ejecución de los servicios de infraestructura de Azure. Piense en recursos como redes, almacenamiento y máquinas virtuales de Azure, y cómo administrarlos. No olvide que Azure le permite hacer mucho más que solo ejecutar su centro de datos virtual en la nube. Cada vez más organizaciones buscan soluciones de nube pública para hospedar cargas de trabajo basadas en Plataforma como servicio. Esto significa que aún ejecuta las aplicaciones empresariales, pero que ya no las implementa en máquinas virtuales ni administra la mayor parte de la infraestructura, como las redes y el almacenamiento. Además de ejecutar servicios de infraestructura y plataforma, también podría pensar en migrar sus cargas de trabajo a soluciones sin servidor y de microservicios con contenedores. Tener esta flexibilidad con respecto al entorno que usa en la nube pública, ya sean servicios de infraestructura tradicionales, servicios de plataforma o cargas de trabajo en contenedores, es realmente uno de los principales beneficios de la nube y una de las formas principales en que la innovación técnica de Azure puede apoyar las necesidades de su empresa.

Además, debe pensar en los requisitos de gobernanza y cumplimiento de la organización. Incluso en su propio centro de datos, no quiere que cualquier persona entre e implemente hardware nuevo, implemente nuevos servidores, expanda el almacenamiento, etc. La actitud en torno a la gobernanza, la seguridad y el control permanece, incluso si empieza a usar los servicios en la nube. La buena noticia es que Azure incluye una amplia lista de funcionalidades de gobernanza y cumplimiento, varias incluso como parte fundamental de la plataforma subyacente. Otras se ofrecen como servicios flexibles y configurables en que puede asumir el control.

Si bien en los primeros párrafos de esta sección de introducción se mencionó la nube pública como una estrategia general para ejecutar sus cargas de trabajo de TI, a partir de aquí pasamos a Azure como solución de nube pública de Microsoft. Antes de abordar asuntos y aspectos más técnicos sobre lo que se necesita para migrar e implementar sus aplicaciones en Azure, hablemos de algunos escenarios en los que Azure puede ayudar en la innovación empresarial.

Innovación empresarial con Azure

Si bien los entornos de informática en la nube como Azure han existido por alrededor de 10 años, el punto de inflexión para la adopción de la nube por parte de las empresas se observó alrededor del año 2016, cuando por primera vez una encuesta de IDG descubrió que más de la mitad de los entornos de TI de las empresas encuestadas se hospedaban en la nube[1].

1 `https://bit.ly/2N8QVo4`

La "primera generación" de la adopción de la nube se caracterizó principalmente por la implementación de centros de datos virtuales. En esta primera generación, las organizaciones implementaron cargas de trabajo de máquina virtual existentes nuevas o migradas en Azure por diversos motivos: algunas migraron a la nube pública para ahorrar en costos de ejecución de centros de datos, mientras que otras querían aprovechar el método más fácil y rápido de implementación de la infraestructura. Otras organizaciones buscaron a Azure para optimizar sus procesos empresariales, usarlo como una configuración de prueba o usar la nube como una solución de recuperación ante desastres asequible. Para otros, el enorme potencial de rendimiento y escala, además de la flexibilidad (en especial, durante el uso máximo), fueron las principales razones para adoptar Azure.

La "segunda generación" de la nube pública llegó cuando, en lugar de simplemente ver el valor de ejecutar y administrar máquinas virtuales, algunas organizaciones vieron beneficio e innovación en la migración a los servicios de plataforma. Esto elimina principalmente el enfoque y la dependencia de las máquinas virtuales, las redes y el almacenamiento, y cambia a un nuevo enfoque concentrado en la aplicación en sí. Dado que los servicios de plataforma no necesitan demasiada infraestructura, son más fáciles de administrar. Además, se dedica menos tiempo a las revisiones o al mantenimiento de servidores, lo que normalmente también se traduce en un mejor tiempo de actividad de las aplicaciones.

En la actualidad, existe una "tercera generación" en sus primeras etapas, en que las organizaciones adoptan tecnologías sin servidor y microservicios, además de usar servicios nativos en la nube para desarrollar soluciones cognitivas y de inteligencia artificial. Azure facilita la implementación de estos tipos de servicios de back-end, en los que se necesita cada vez menos conocimiento para crear la infraestructura subyacente. Para la mayoría de estos servicios, no hay mucho, incluso a veces nada, que administrar en la infraestructura.

Azure, en el fondo, es una plataforma de nube pública, pero hoy en día, hay una variedad de diferentes modelos de nube disponibles en la industria. Analicemos los principales.

Modelos de nube pública, nube híbrida y de varias nubes

En un nivel alto, muchas organizaciones están evaluando la implementación o adopción de uno (o más) de los siguientes modelos de nube.

Nube pública

Esta es la plataforma de nube que normalmente ofrece un proveedor de servicios. Estos proveedores de servicios incluyen Azure, Amazon Web Services (AWS), Google Cloud Platform (GCP), Rackspace y Digital Ocean. En palabras simples, el centro de datos es administrado por el proveedor y usted consume la parte de hospedaje como un servicio.

Además, no hay ninguna dependencia o integración con su centro de datos local existente. Normalmente, este modelo lo utilizan empresas emergentes, clientes de **pequeñas y medianas empresas** (**PYMES**) o empresas más grandes, que desean crear un entorno independiente fuera de lo que ejecutan de forma local.

Nube híbrida

En un modelo de nube híbrida, se crea una integración entre los centros de datos locales existentes y un entorno de nube pública. La mayoría de las veces, esto se debe a que se desean expandir las capacidades del centro de datos o no se quiere realizar una migración completa a un modelo que solo sea de nube pública. La creación de una nube híbrida normalmente comienza con la integración de la red física (en Azure ofrecida por ExpressRoute o una VPN de sitio a sitio), seguida de la implementación de la Infraestructura como servicio o Plataforma como servicio. Otro aspecto de la nube híbrida es la identidad. Azure le ofrece Azure Active Directory como la solución de identidad. En el caso de las nubes híbridas, las organizaciones sincronizan (todos o seleccionan) a los usuarios locales y agrupan objetos de dominios de Active Directory en un solo inquilino de Azure Active Directory. Esto permite la optimización de la administración de los usuarios y la seguridad, lo que ofrece a los usuarios un procedimiento de autenticación fácil, pero altamente seguro para las cargas de trabajo que se ejecutan en la nube.

Varias nubes

Cada vez más clientes (empresariales) analizan utilizar o actualmente usan una estrategia de varias nubes. Varias nubes significa usar varias nubes públicas o híbridas de forma conjunta. El beneficio es usar lo que está disponible. Imagine que su aplicación empresarial se basa en un servicio que no está disponible en la nube pública de su preferencia, pero que ya está disponible en Azure. Siempre y cuando pueda integrar ambos mundos en todos los aspectos, como la seguridad, la compatibilidad, los empleados calificados, entre otros, no hay razón para no ir en esa dirección. Otro impulsor podría ser considerar las ventajas económicas. En lugar de ejecutar todas las cargas de trabajo con el mismo proveedor de nube pública, podría ser rentable dividir las cargas de trabajo entre diferentes proveedores de nube. Por último, adoptar características como DevOps e Infraestructura como código también lo ayudará a adoptar una estrategia de varias nubes. Herramientas como Jenkins, Terraform, Ansible y muchas otras proporcionan API de REST que pueden comunicarse con diferentes back-ends de nube. Por lo tanto, sus equipos de TI no tienen que aprender diferentes plantillas específicas de la nube, sino que pueden centrarse en las capacidades de las herramientas, en lugar de centrarse en las capacidades de la nube como tal. Al mismo tiempo, se debe mencionar que una estrategia de varias nubes también implica varios desafíos. Probablemente, las preocupaciones más críticas que debe considerar son la compatibilidad, los requisitos de un conjunto de habilidades mixtas de su personal de TI y la complejidad general debido a la necesidad de administrar diferentes entornos.

Arquitecturas de nube pública de Azure

Ahora que conoce mejor los diferentes modelos de nube, nos concentraremos más en las arquitecturas de nube pública.

Infraestructura como servicio (IaaS)

Parte de este primer capítulo está dedicado a la migración y ejecución de sus aplicaciones empresariales en un modelo de **Infraestructura como servicio** (**IaaS**), a través de conceptos similares a los relacionados con su centro de datos local, con las redes virtuales, el almacenamiento virtual y las máquinas virtuales como los principales bloques de creación de la arquitectura.

Sin embargo, esa no es la única forma en que puede ejecutar sus aplicaciones en Azure. Como transición a otros capítulos de este libro, describiré brevemente dónde Azure puede ayudar en la innovación empresarial, o la transformación digital de sus cargas de trabajo, mediante otras arquitecturas, además de las máquinas virtuales.

Opciones de plataforma como servicio (App Service, SQL Database, Azure Container Instances y Azure Kubernetes Service)

Plataforma como servicio (**PaaS**) se refiere a ejecutar sus cargas de trabajo en Azure sin implementar máquinas virtuales. Por ejemplo, podría ejecutar una aplicación web en Azure App Service sin implementar primero la máquina virtual subyacente. Esto aporta eficiencia y optimización, puesto que tiene que preocuparse de menos administración operativa. Como alternativa, podría usar un servicio como Azure SQL Database o Cosmos DB, lo que le permite ejecutar exactamente el mismo tipo de solución de base de datos que ya tiene, pero una vez más sin tener que considerar ninguna máquina virtual. Como se mencionó antes, tener esta flexibilidad de arquitectura a la vez que no necesita implementar y administrar sus propias dependencias de infraestructura ofrece espacio para la innovación. Además, las nuevas funcionalidades son más fáciles de adoptar en un modelo de PaaS que en un modelo de IaaS, porque la implementación del servicio es mucho más rápida, gracias a que no depende del sistema operativo y a la compatibilidad con el lenguaje de desarrollo, por nombrar solo un par de razones. El modelo de PaaS también puede ser rentable, puesto que la mayoría de los servicios de PaaS son más económicos que sus alternativas de IaaS (por ejemplo, usar aplicaciones web de Azure es más económico que implementar una máquina virtual web de Azure con las mismas características de rendimiento).

Sin servidor (Functions, Cosmos DB, Logic Apps y Cognitive Services)

Además de estos dos conceptos estándar de IaaS y PaaS, también tiene las arquitecturas **sin servidor** y de **microservicios**. Si consideramos los servicios sin servidor de Azure, una opción es usar Azure Functions, un servicio de informática sin servidor que le permite ejecutar código a petición sin tener que aprovisionar o administrar explícitamente la infraestructura. Puede usar Azure Functions para ejecutar una secuencia de comandos o un fragmento de código en respuesta a una variedad de eventos. Otro servicio sin servidor ofrecido por Azure es Azure Logic Apps, un motor de flujo de trabajo empresarial que proporciona conectores a más de 200 aplicaciones empresariales, como Dropbox, OneDrive, SAP, DocuSign y Adobe. Con Logic Apps, puede crear un flujo de trabajo paso a paso con inteligencia lógica para reemplazar su cadena actual de cargas de trabajo complejas (en su mayoría) basadas en máquinas virtuales.

Los microservicios son un enfoque de desarrollo de aplicaciones en el que una arquitectura de aplicaciones más compleja, conocida como aplicación monolítica, se divide en varios componentes más pequeños. Cada componente tiene un solo propósito, como administrar pedidos de productos, pagos de pedidos o el seguimiento de envíos, en una solución de plataforma de comercio electrónico más amplia. Los microservicios son una estrategia muy popular para migrar aplicaciones (heredadas) a la nube pública. Los microservicios también se conocen como contenedores o aplicaciones en contenedores. En la actualidad, Docker (`http://www.docker.com`) es el formato de contenedor estándar para los microservicios y es totalmente compatible con Azure.

Azure le permite colocar sus contenedores en Azure y ejecutarlos con **Azure Container Instances** (**ACI**) como una carga de trabajo de contenedor independiente. O bien, si busca funcionalidades de orquestación más avanzadas, también puede implementar **Azure Kubernetes Service** (**AKS**), que proporciona un entorno de clúster de Kubernetes como servicio. Además de la ventaja de la compatibilidad con aplicaciones heredadas, los contenedores también son muy interesantes porque son ligeros, ejecutan una tarea específica y se pueden mover fácilmente en diferentes entornos. La misma imagen de contenedor se puede ejecutar en el Mac de un desarrollador, insertar en un registro de contenedor de Azure, ejecutar en Azure Web App para contenedores, ejecutar como una instancia de contenedor de Azure o ejecutar en su propio centro de datos, además de la infraestructura de RedHat OpenShift o cualquier otra infraestructura de nube pública. Esta es la razón principal por la cual los contenedores son tan populares hoy en día.

¿Cómo pueden estas diferentes arquitecturas de nube llevar a la innovación empresarial? Durante mucho tiempo, las empresas se esforzaron por optimizar la TI y mejorar los procesos de TI, mediante la búsqueda de formas de alinear las operaciones y los equipos de desarrollo, no siempre con gran éxito. Sin embargo, al cambiar de la infraestructura (IaaS) a los servicios de plataforma o sin servidor (PaaS/SaaS), se elimina gran parte de esta dependencia.
La flexibilidad de usar la nube pública y los diferentes modelos operativos que proporciona definitivamente ayudará en la innovación empresarial.

Aparte del lado técnico de la optimización, esto también podría llevar a liberar más recursos y ahorrar dinero, que la empresa puede invertir en más innovación. Examinemos más razones para migrar a la nube.

Creación de estrategias para la modernización de aplicaciones con Azure

Encontrar las razones correctas para migrar cargas de trabajo a la nube o implementar nuevas cargas de trabajo directamente en la nube es una parte esencial del éxito de sus proyectos de migración. Si lo examinara a través de los anteojos técnicos de su administrador del sistema o desarrollador, lo más probable es que encontraría muchas buenas razones para migrar. Los sistemas se pueden implementar más rápido, las operaciones y la administración se vuelven más fáciles, y la nube le ofrece escala, alta disponibilidad y rentabilidad. Pero tal vez la alta dirección tiene una opinión diferente al respecto, puesto que acaban de aprobar una extensión de su centro de datos físico. O tal vez el sector en el que se encuentra tiene que seguir varias normativas de cumplimiento, lo que hace que la nube pública sea una plataforma difícil de aprobar.

¿Por qué migrar a Azure?

Intentemos ser lo más abiertos posible respecto a la migración a la nube y examinemos tanto los beneficios como los riesgos. Note que estos se basan principalmente en la experiencia adquirida al trabajar como arquitecto y entrenador de nube de Azure por varios años. Además, tenga en cuenta que lo que constituye un beneficio para una organización podría ser una preocupación para otra.

Beneficios de la nube

Existen varios desafíos empresariales comunes que hacen que realizar una migración a la nube valga la pena. Aquí se enumeran algunos escenarios representativos en que la nube es beneficiosa:

- Se le pide que reduzca los costos operativos.
- Sus aplicaciones se enfrentan a aumentos de tráfico y sus sistemas internos no pueden escalar para satisfacer la demanda.
- Su empresa requiere implementaciones más rápidas de sistemas y aplicaciones, en diferentes regiones del mundo, para atender a sus clientes.
- Debe seguir normativas de seguridad que son difíciles de implementar en sus centros de datos locales o que son costosas.
- Su capacidad de almacenamiento está creciendo exponencialmente y es difícil ponerse al día, tanto en el aspecto técnico como en el de ahorro de costos.

- Sus sistemas se están quedando obsoletos, puesto que ejecutan aplicaciones y sistemas operativos heredados que ya no son compatibles.
- El conocido CAPEX a OPEX (gastos de capital y gastos operacionales): su empresa desea cambiar a un modelo basado en el consumo y de pago por uso.
- Usted tiene una empresa nueva que no posee los medios financieros o la certeza para implementar y mantener sus propios centros de datos.

Desafíos potenciales de la migración a la nube

Si bien es posible que reconozca varios de los ejemplos mencionados en su propia organización, lo que le dará excelentes motivos para explorar la migración a la nube, debe saber que este tipo de migraciones también conllevan desafíos potenciales. Aunque estos se consideran cada vez menos como factores de bloqueo, vale la pena mencionar algunos de ellos:

- Su negocio está en un sector específico y no se le permite almacenar datos en la nube pública debido a la confidencialidad u otras normativas de cumplimiento.
- Sus cargas de trabajo sufren de latencia cuando el centro de datos no está cerca de sus usuarios.
- Algunas de sus cargas de trabajo son heredadas y no pueden ejecutarse en un entorno de nube.
- Acaba de realizar una gran inversión en su propio centro de datos.
- No desea estar atado a un proveedor y una estrategia de varias nubes parece tener demasiada sobrecarga.

Asignación de justificaciones y resultados empresariales

Si los beneficios de la nube y los posibles desafíos mencionados que vienen con las migraciones a la nube ya le han hecho pensar en las estrategias de nube, examinemos rápidamente otros temas que debe tener en cuenta.

La nube no es la solución más económica para todos

Cambiar o migrar desde su propio centro de datos a la nube pública es una solución rentable, pero eso no significa que sea más económico para todos. Los cálc ulos de análisis empresarial podrían ejecutarse más rápido debido a la escala de la nube, lo que aporta beneficios para la empresa, pero los costos de ejecución podrían ser tan costosos (o incluso más) como los costos asociados con la compra y ejecución de una carga de trabajo similar en el entorno local. Sin embargo, esa infraestructura demoraría mucho más antes de poder producir resultados similares.

Para obtener una visión adecuada de los costos de consumo de recursos de Azure, siempre comience en la página de precios de Azure: `https://azure.microsoft.com/pricing/`.

Ninguna nube pública garantiza una alta disponibilidad completa

Este es uno de mis temas favoritos para presentar en una conferencia. Obviamente, una plataforma de nube pública se crea teniendo en cuenta la alta disponibilidad, pero debe considerar que no tendrá una disponibilidad total. Por ejemplo, si considera usar IaaS, usted, como cliente, necesita diseñar su alta disponibilidad mediante Conjuntos de disponibilidad de Azure o Zonas de disponibilidad. Migrar a PaaS podría ser una solución para esto, puesto que hace que sea más responsabilidad de Microsoft asegurarse de que sus servicios de aplicaciones, soluciones de datos, etc. se ejecuten en una plataforma de alta disponibilidad. La buena noticia es que los servicios de Azure tienen un Contrato de nivel de servicio (**SLA**) total de un 99,9 % para la mayoría de los servicios, con algunos servicios como Azure SQL Database que tienen una capacidad de SLA del 99,99 % o un 99,999 % en el caso de Azure Cosmos DB.

Puede encontrar toda la información relacionada con los SLA de Azure en este vínculo: `https://azure.microsoft.com/support/legal/sla/summary/`.

La migración mediante "lift-and-shift" de máquinas virtuales puede que no siempre proporcione los mejores beneficios

Si bien una migración mediante "lift-and-shift" de máquinas virtuales a Azure es sin duda un buen paso hacia la migración a la nube, no siempre es la más eficaz, desde una perspectiva técnica y también de costos. Tal vez sus sistemas estén quedando obsoletos y ejecutan aplicaciones heredadas que podrían no funcionar "mejor" en una máquina virtual en la nube. O bien, quizá sus sistemas locales no tienen el tamaño correcto para el servicio que ofrecen. La migración de una carga de trabajo de este tipo a la nube sin cambiar las características del sistema podría generar un enorme aumento de costos para el consumo de Azure.

Encontrará los recursos que pueden ayudar a identificar su **Retorno de la inversión** (**ROI**) y **Costo total de propiedad** (**TCO**) de Azure aquí: `https://bit.ly/2R4wGc6`.

Los contenedores no siempre son la mejor solución para la migración a la nube

En la introducción, hablamos un poco de los contenedores. Los contenedores son realmente increíbles. Sin embargo, no siempre son la mejor plataforma a la que migrar sus aplicaciones y, sin duda, no deben ser el único impulsor de la migración a la nube. Adopte los contenedores como parte de su estrategia de nube general, pero no los vea como la meta final de su migración a la nube.

Para obtener información adicional sobre la ejecución de contenedores en Azure y los servicios que están disponibles, explore este vínculo: `https://bit.ly/2QFTf8h`.

Lo que aprendió en esta sección

Debe conocer lo que la nube puede hacer por su organización y lo que no puede hacer. Decida si alguna de las afirmaciones anteriores es verdadera para su organización y su razonamiento específico para migrar a un escenario de nube pública.

Como está leyendo este libro, estoy seguro de que le interesa obtener más información sobre el aspecto técnico de la implementación y ejecución de las aplicaciones empresariales de Azure. Esto es exactamente de lo que trata este libro.

Cualquier implementación o migración comienza con saber lo que tiene antes de poder saber a dónde se dirige. Esta fase se conoce como la fase de **evaluación**, por lo que tiene sentido que ocupemos tiempo en ella en la siguiente sección.

Enfoque de migración a la nube

Más que nunca, las organizaciones están replanteando su estrategia de centro de datos y adoptando e integrando fuertemente la nube pública en su estrategia de TI a largo plazo. Dos factores clave que motivan esta tendencia son la escalabilidad de la nube y la reducción de las inversiones de capital. Uno de los puntos de partida para hacer que esta migración a la nube pública sea exitosa es saber qué se admite en la nube pública desde la perspectiva de la carga de trabajo y saber cuál es su panorama actual de TI. Cuanto mejor sea la coincidencia entre estos dos aspectos, más fácil será la migración a la nube.

Evaluación de la preparación para la nube de su organización

Esta primera sección se enfoca en evaluar sus centros de datos locales y determinar qué cargas de trabajo, máquinas virtuales, redes, almacenamiento, aplicaciones y soluciones de datos tiene. Tendrá que pensar cuáles son los elementos que desea migrar a la nube pública y cómo migrarlos con el menor tiempo de inactividad posible y el menor impacto en la empresa. Una migración exitosa depende del uso de herramientas para facilitar el trabajo. Analizaremos varias herramientas disponibles de Microsoft que ayudan en este proceso. También realizaremos la identificación y estimación del costo de la migración y, más específicamente, cuál podría ser el costo de consumo de ejecución, una vez que la carga de trabajo se ejecute en Azure.

Herramientas de evaluación

Ayudar a sus clientes a realizar una migración a Azure que sea fácil o, como mínimo, menos compleja es uno de los objetivos clave de Microsoft. Cuanto más fácil sea la migración (es decir, menos disruptiva para el negocio), mayor será el éxito que tendrá la nube. Además de la excelente documentación sobre todos los aspectos de Azure que puede encontrar en `https://bit.ly/30kWG7B`, debería revisar el **Azure Migration Center**: `https://bit.ly/384cHRN`.

Azure Migration Center divide las fases que deberá ejecutar cuando realice la migración a la nube. La primera de ellas es la fase de **evaluación**. Esta debe ser siempre la primera fase de su proyecto de migración más amplio. Conozca lo que tiene hoy para descubrir lo que puede ejecutar mañana.

Hoy en día, puede elegir entre las siguientes herramientas de evaluación gratuitas de Microsoft:

- **Azure Migrate**: realiza una evaluación de las cargas de trabajo basadas en máquinas virtuales
- **Azure Data Migration Assistant**: ejecuta una evaluación para bases de datos de SQL Server, que ayuda en la migración a máquinas virtuales de Azure SQL o bases de datos de Azure SQL
- **Azure Database Migration Service**: aunque no es específicamente una herramienta de evaluación, ayuda a migrar desde diferentes soluciones de datos locales hasta alternativas de PaaS de Azure
- **App Service Migration Assistant**: realiza un análisis de cualquier punto de conexión de la aplicación y es compatible con varios lenguajes y plataformas (como Java, .NET, Node.js y PHP)

Para la mayoría de estas herramientas, existe cierta superposición entre la evaluación y la migración, pero permítanme dedicar un poco de tiempo a cada una de ellas y destacar los beneficios básicos y los argumentos para revisarlas.

Azure Migrate

Azure Migrate es la herramienta principal para ejecutar las migraciones de sus cargas de trabajo a Azure. Proporciona una integración completa desde la evaluación hasta la migración y el seguimiento. La ventaja clave de Azure Migrate es que ayuda a las organizaciones a obtener una visión detallada sobre cómo se ven sus máquinas virtuales del centro de datos desde una perspectiva de las características (CPU, memoria y discos), qué sistema operativo están ejecutando y, lo que es más importante, si son compatibles con la migración a Azure. Es una herramienta viable para organizaciones de todos los tamaños que quieran validar la migración de sus máquinas virtuales a Azure mediante un enfoque de "lift and shift", y que admite tanto VMware como Hyper-V y hasta 35 000 máquinas de origen (10 000 para entornos de Hyper-V).

Una vez que Azure Migrate tiene una buena vista en la máquina virtual de origen, analiza la información recopilada y la asigna con Azure. En primer lugar, identifica si la máquina virtual es compatible con los requisitos de máquina virtual de Azure (si el sistema operativo es compatible, etc.), y recomienda un tamaño de máquina virtual de Azure. Por último, proporciona un cálculo del costo mensual de ejecutar esa máquina virtual tal como está en Azure, que se presenta en vistas de panel claras y fáciles de comprender (*Figura 1*).

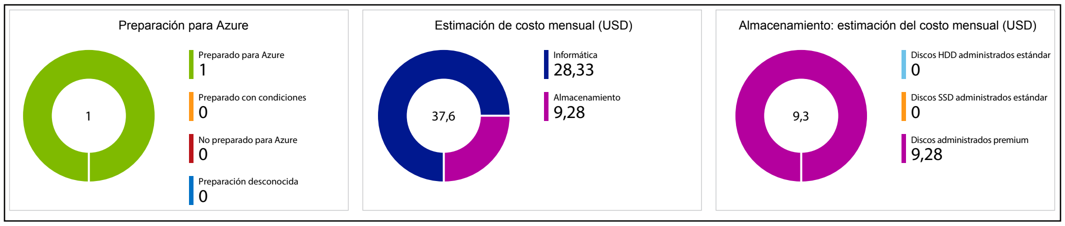

Figura 1: Azure Migrate: detalles de la evaluación

Otra característica interesante es la *visualización de dependencias*, en que Azure Migrate identificará una aplicación u otras dependencias entre las máquinas virtuales a fin de facilitar la migración real. Piense en un servidor de aplicaciones que tenga conectividad con un servidor de base de datos. No ayudaría a la migración si el servidor de aplicaciones se migró, pero el servidor de base de datos no. Tenga en cuenta que la visualización de dependencias requiere la instalación de un agente, aunque es posible que no todas las organizaciones lo acepten, en especial, cuando la decisión de realizar realmente la migración a Azure no se ha confirmado del todo.

Evaluación de la infraestructura de VMware

En caso de que exista una infraestructura de VMware, donde tener vCenter es una dependencia, se debe implementar un *dispositivo de Azure Migrate* que se ejecute junto a las máquinas virtuales de VMware. Para acelerar esta implementación, hay una plantilla de OVA disponible para descargar (alrededor de 15 GB), que le permite realizar la implementación como una plantilla de OVF directamente desde su cliente de VMware vSphere y facilitar la instalación de este dispositivo. Este dispositivo realiza una detección, que puede limitarse a centros de datos de vCenter, clústeres, hosts específicos o máquinas virtuales seleccionadas. Una vez finalizada la detección, administrará la mayor parte de los resultados de la evaluación en el proyecto de Azure Migrate en el portal de Azure, en función de la información de metadatos enviada desde el dispositivo de regreso a Azure Migrate.

Evaluación de la infraestructura de Hyper-V

Si actualmente usa Hyper-V en su centro de datos, puede confiar en Azure Migrate para la evaluación y el descubrimiento. Empezará por descargar un archivo VHD de Hyper-V (alrededor de 10 GB) y, a continuación, lo configurará como una nueva máquina virtual de Hyper-V. De forma similar al escenario de VMware, este dispositivo de detección es responsable de reunir toda la información necesaria de su infraestructura local para las máquinas que defina. Este permite una evaluación selectiva basada en hosts de Hyper-V o máquinas virtuales individuales.

Evaluación de la infraestructura de nube física y de otro tipo

Con la versión más reciente de Azure Migrate, Migrate no solo ayuda en la evaluación basada en Hyper-V, sino que ahora también es compatible con máquinas físicas y otros proveedores de nube (como AWS y GCP) como origen. Al igual que con VMware o Hyper-V, la parte mágica es el *dispositivo de replicación*, a partir de una instancia de Windows Server 2016 que necesita preparar.

Además de usar las herramientas nativas de Microsoft, el nuevo Azure Migrate funciona directamente con varios proveedores, que ofrecen herramientas de evaluación y migración de terceros para diferentes cargas de trabajo. A veces, estas son genéricas y, a veces, admiten una solución única, pero, hoy en día, están bien integradas en el mismo servicio de Azure Migrate.

Puede encontrar más información sobre qué socios y herramientas están disponibles en la página principal de Azure Migration Center:

`https://azure.microsoft.com/pricing/details/azure-migrate/`

Azure Data Migration Assistant

Si tiene soluciones de datos locales basadas en SQL Server que se ejecutan en servidores físicos o virtuales, puede migrarlas mediante el enfoque descrito anteriormente, suponiendo que las máquinas virtuales son compatibles con los requisitos de máquina virtual de Azure. Aunque tal vez no necesite realizar una migración mediante "lift and shift" a las máquinas virtuales de Azure, puesto que sus bases de datos podrían calificar para ejecutarse como instancias de Azure SQL. Para averiguarlo, el enfoque más sencillo es ejecutar una evaluación con **Data Migration Assistant** (**DMA**) de Azure (*Figura 2*).

El proceso de evaluación no podría ser más fácil:

1. Debe ejecutar la herramienta en su entorno local, directamente en la máquina de SQL Server que desea evaluar o en una máquina remota (por ejemplo, la estación de trabajo del administrador de la base de datos SQL).
2. Una vez que se instale la herramienta, pasará a un asistente de evaluación, donde puede especificar el origen (SQL Server) y el destino (Azure SQL Database).
3. Después de hacer clic en la opción **Iniciar evaluación**, el proceso comienza a analizar la base de datos de origen y, por lo general, lo hace en minutos. Los resultados se muestran dentro de la herramienta y también se pueden exportar. Desde una perspectiva de evaluación, analiza dos dominios:
    - **Paridad de características de SQL Server**: aquí es donde puede revisar detalles sobre funcionalidades no admitidas y parcialmente admitidas, una vez que su base de datos se ejecute en Azure SQL. Un ejemplo de algo que descubrí como cliente fue la función de índice de búsqueda.
    - **Problemas de compatibilidad**: esta es la segunda comprobación que tiene lugar. Si se detectan problemas, se enumerarán aquí como un factor de bloqueo para la migración:

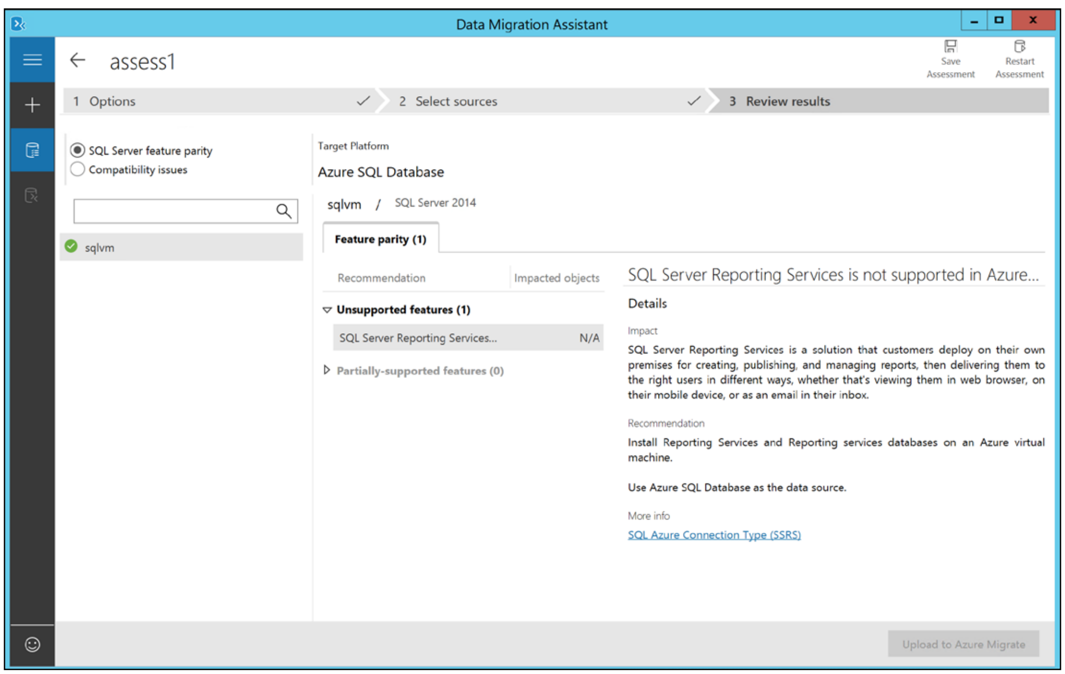

Figura 2: DMA: evaluación

Azure Database Migration Service

Si, además de las soluciones de datos de SQL Server, está ejecutando otras bases de datos en su centro de datos local, Azure ofrece otra herramienta para ayudarlo a migrarlas también a Azure. Esta herramienta permite realizar migraciones sin problemas desde varios orígenes de bases de datos diferentes hasta las soluciones de datos de Azure. Los casos de usos característicos aquí son los siguientes:

- Migración de PostgreSQL a Azure Database for PostgreSQL.
- Migración de Oracle Database a Azure Database for PostgreSQL.
- Migración de MongoDB a Azure Cosmos DB.
- Migración de MySQL a Azure Database for MySQL.

Habitualmente, la migración de una infraestructura de arquitectura de máquina virtual a PaaS de Azure ofrece asistentes fáciles de utilizar que lo ayudarán en este proceso (*Figura 3*).

Observe que esta herramienta permite escenarios de migración sin conexión y en línea:

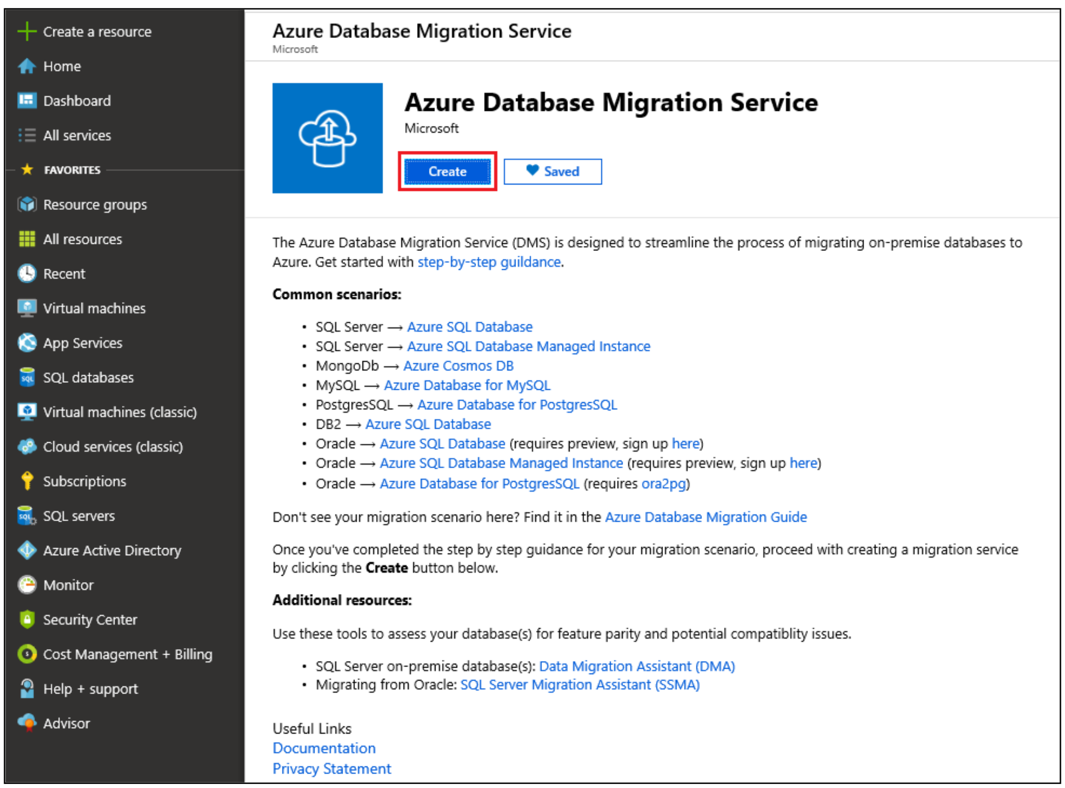

Figura 3: Azure Database Migration Service

App Service Migration Assistant

App Service Migration Assistant es una versión actualizada de la antigua herramienta Movemetothecloud.net. Como su nombre lo indica, la funcionalidad básica de esta herramienta gratuita de Microsoft es ayudar a migrar las aplicaciones (Web) a Azure Web Apps.

Además de realizar la migración de aplicaciones reales, la otra funcionalidad importante es realizar una evaluación antes de la migración real. La evaluación comienza con la conexión a un punto de conexión público o a un servidor web interno, que se analiza en busca de varios detalles de tecnologías de servidor web para identificar cualquier problema de migración, como los siguientes:

- Enlaces de puerto (puesto que Azure solo admite 80/443 para los enlaces de puerto)
- Protocolos (HTTP y HTTPS)
- Certificados (comprueba si el sitio usa certificados y CA autofirmada o pública)
- Dependencias en el archivo `applicationhost.config`
- Grupos de aplicaciones
- Tipo de autenticación
- Cadenas de conexión

Desde una perspectiva de plataforma de servidor web y lenguaje, es compatible con muchas más aplicaciones basadas en Windows Server IIS/.NET, como las siguientes:

- Ruby
- Node.js
- Java
- PHP

El resultado de la evaluación y el inventario se presenta en un formulario web detallado, que se puede exportar con fines de documentación (*Figura 4*):

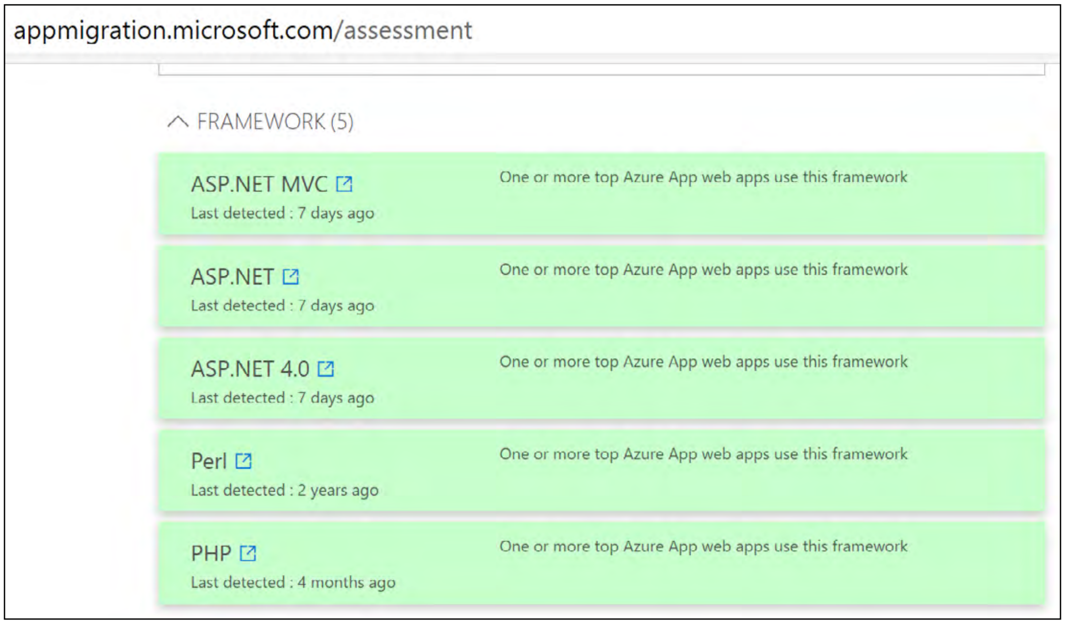

Figura 4: Evaluación de Azure App Service Migration

Resumen de la sección

En esta sección, se proporcionó una introducción y una descripción del marco de migración de aplicaciones de Microsoft, mediante el posicionamiento y la descripción de varias herramientas de Microsoft disponibles para ayudar en el proceso de evaluación.

En la siguiente sección, se explicará el tema de la identidad y el control de acceso, como otro aspecto importante que considerar antes y durante la realización de una migración a la nube en Azure.

Identidad y control de acceso

Hasta ahora, nos hemos centrado en la evaluación y la justificación empresarial de sus proyectos de migración a la nube, principalmente desde la perspectiva de las soluciones de aplicaciones y datos, pero otro aspecto importante que analizar es cómo administrar la identidad y control de acceso en la nube. Esto es importante tanto para los usuarios finales como para los administradores.

La identidad es el componente principal de toda la seguridad en la nube de Azure: cada vez que cualquier administrador desea "hacer" algo en la plataforma de Azure, debe autenticarse y obtener autorización. No importa si utiliza el portal de Azure, las herramientas de línea de comandos de Azure, como PowerShell o la CLI de Azure, o utiliza las API de REST. Los usuarios finales también pueden beneficiarse en gran medida de Azure Active Directory. Las soluciones como el autoservicio de restablecimiento de contraseña, la unión a un dominio para la administración de dispositivos de Azure AD, el acceso condicional, el riesgo de usuario y muchas más optimizarán en gran medida la forma en que los usuarios inician sesión en las aplicaciones en la nube y qué tan seguro es este inicio de sesión.

Azure Active Directory como una solución de identidad en la nube

Desde una perspectiva de identidad, no hay manera de eludir Azure Active Directory. Esta solución de identidad en la nube viene en diferentes tipos:

- **Azure Active Directory**: el componente de identidad central de Azure, que ofrece a los usuarios de la nube, grupos, aplicaciones y objetos de entidad de servicio
- **Azure Active Directory Domain Services**: un servicio emulado de Active Directory, que ofrece Kerberos y NTLM, similares a los controladores de dominio de Active Directory locales
- **Azure Active Directory B2B**: concepto negocio a negocio, mediante el cual las organizaciones pueden invitar a los usuarios desde sus inquilinos de Azure AD
- **Azure Active Directory B2C**: concepto de negocio a consumidor, mediante el cual las organizaciones permiten la autenticación de usuarios desde proveedores de identidad de redes sociales (como Facebook, Twitter, LinkedIn, etc.)

Además de los diferentes tipos mencionados aquí, Azure Active Directory también viene en diferentes ediciones:

Edición de Azure Active Directory	Características y funciones principales
EDICIÓN GRATUITA	• Proporciona servicios de identidad básicos, almacenando usuarios, grupos, aplicaciones y objetos de entidad de servicio • Puede sincronizarse con su Active Directory local mediante Azure AD Connect • Proporciona informes seguridad básicos
EDICIÓN BÁSICA	• Todas las características de la edición gratuita + • Identidad empresarial • Proxy de aplicación hacia aplicaciones web locales • Autoservicio de restablecimiento de contraseña • Administración de grupos
EDICIÓN PREMIUM P1	• Todas las características de la edición básica + • Administración de grupos de autoservicio • Reescritura de contraseñas local • Reescritura de dispositivos bidireccional • Acceso condicional para seguridad optimizada
EDICIÓN PREMIUM P2	• Todas las características de la edición Premium PI + • Protección de la identidad • Privileged Identity Management

Tabla 1: Niveles de Azure Active Directory

Solo en función del enriquecido conjunto de características y funciones de seguridad avanzadas que incluye la solución, las organizaciones deberían considerar *Azure AD Premium P1* para la mayoría de sus usuarios habilitados para la nube, ampliada con *Azure AD Premium P2* para usuarios clave, como la administración de alto rango, administradores, encargados de seguridad y otras personas clave dentro de la organización con alta visibilidad.

Autenticación de nube con Azure Active Directory

La mayoría de las organizaciones ya tienen una solución de identidad implementada en su centro de datos local, la que, a menudo, es Microsoft Active Directory. En este escenario, la topología recomendada sería crear una arquitectura de identidad híbrida, a partir del entorno de origen de Active Directory. Azure AD Connect sincroniza los objetos de usuario y de grupo (todos o selecciona los que se basan en los filtros que defina). Como tal, una cuenta de usuario con el **Nombre principal de usuario** (**UPN**) `peter@company.com` desde el Active Directory local, usará el mismo alias para la autenticación en Azure Active Directory.

Sin embargo, hay tres escenarios de autenticación distintos:

- Sincronización de hash de contraseña (PHS) de Azure AD
- Federación de Azure AD con ADFS o federación de terceros (ADFS)
- Autenticación de paso (PTA) de Azure AD

El enfoque más fácil (y el más recomendado) es **PTA de Azure AD**. En este escenario, los objetos de Active Directory se sincronizan con Azure AD mediante AD Connect, incluido el hash de contraseña del dominio. Esto permite a los usuarios iniciar sesión en aplicaciones en la nube con sus credenciales de Azure AD, que son idénticas a las credenciales locales.

Lamentablemente, el almacenamiento de contraseñas (o el hash de contraseñas) no es viable para una gran cantidad de organizaciones, que desean mantener el control de las credenciales desde una perspectiva local. En este escenario, debe implementar una infraestructura de federación, que puede ser del tipo **Servicios de federación de Active Directory** (**ADFS**) o una alternativa que no sea de Microsoft (Okta es una opción popular). Aunque aún necesita sincronizar los objetos de AD con Azure AD, la contraseña nunca se almacena en el directorio de la nube. Después de la autenticación del usuario, Azure AD reenvía la solicitud a la infraestructura de ADFS, que normalmente se ejecuta en el centro de datos local. ADFS envía las credenciales recibidas a Active Directory para validación. Si se aceptan, el usuario puede autenticarse.

Aunque ADFS es el diseño "típico" que se debe seguir cuando se implementa la identidad en un modelo de nube híbrida, también viene con algunos inconvenientes. Los servidores de ADFS se ejecutan de forma local, lo que significa que existe una dependencia de la conectividad a Internet, debido a que se necesita una topología altamente disponible para garantizar que los usuarios siempre puedan iniciar sesión en las aplicaciones en la nube cuando sea necesario. ADFS también es complejo de administrar, y su servidor proxy de ADFS en la DMZ está orientado a la Internet pública todo el tiempo.

Para adaptar las ventajas y la facilidad de uso de la sincronización de hash de contraseñas, junto con la necesidad de mantener la administración de credenciales en el Active Directory local, Microsoft creó un tercer escenario, la PTA. Una vez más, se empieza por sincronizar a los usuarios y los grupos con AD Connect. A continuación, en lugar de implementar una infraestructura de ADFS compleja, implemente agentes de Acceso directo en los Controladores de dominio de Active Directory locales. Estos escuchan en el puerto 443, pero solo para direcciones IP públicas de puntos de conexión de servicios de Azure AD. Se denegarán otras solicitudes. Cuando un usuario inicia sesión en Azure AD, la solicitud se transfiere al agente de PTA, que envía las credenciales al Active Directory local, que sigue siendo responsable de validar las credenciales.

Revise el siguiente vínculo para conocer todos los detalles sobre la documentación de administración de identidades y acceso de Azure:

`https://azure.microsoft.com/product-categories/identity/`

Gobernanza de Azure

La gobernanza de Azure es una combinación de diferentes servicios y funcionalidades de Azure, que permiten la administración de todos sus recursos de Azure a escala y siguen las directrices de control. La gobernanza de Azure funciona en varias suscripciones y grupos de recursos, y se basa en una combinación de Azure Identity, control de acceso basado en roles (RBAC), directivas de Azure y grupos de administración. También podría ampliar el concepto con Azure Resource Graph. Algunos clientes también consideran el control de costos como parte de los procesos de gobernanza y los procedimientos recomendados. Si su organización tiene un Centro de operaciones de seguridad (SOC), este departamento probablemente se responsabilizará de este proceso o, al menos, (debería) estar muy involucrado en él.

A continuación, describiré cada uno de los diferentes servicios de Azure que permiten la gobernanza.

Grupos de administración

Durante mucho tiempo, una suscripción de Azure se consideró el límite entre la administración y el control. Esto permitía a las organizaciones utilizar varias suscripciones de Azure para "separar" los recursos entre sí. Algunas organizaciones se suscribieron a un nivel geográfico, otras usaron una suscripción dedicada para una carga de trabajo específica de la aplicación y otras incluso se separaron en función del desarrollo, la prueba y la producción.

Este modelo cambió hace poco con la introducción de **grupos de administración** (*Figura 5*). Cuando la directiva y la iniciativa de Azure eran (y siguen siendo) de verdad excelentes fuentes de control de gobernanza, estaban vinculadas a una única suscripción de Azure, que era difícil de administrar en entornos de Azure más grandes, en que los administradores querían replicar la configuración de directivas en varias suscripciones. Eso es exactamente lo que proporcionan los grupos de administración de Azure: una asignación de suscripción cruzada de la directiva y la iniciativa de Azure.

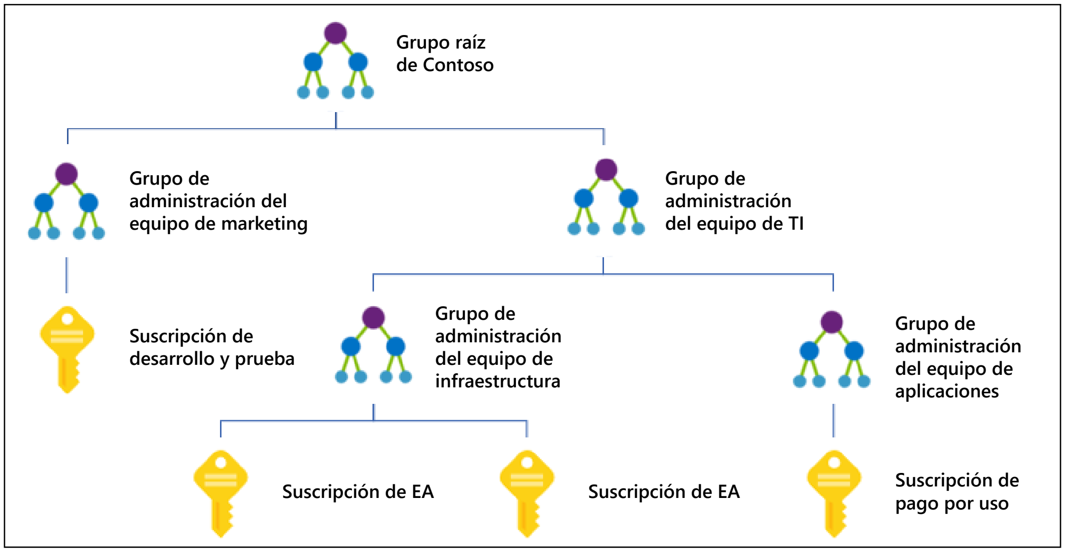

Figura 5: Grupos de administración de Azure

Identidad y control de acceso basado en roles

Una vez más, la identidad es clave en una plataforma de nube pública como Azure. Los ejemplos que vimos anteriormente ya deberían haberlo dejado claro, pero aún hay otro ejemplo que puedo compartir: Azure depende en gran medida de RBAC para identificar quién puede hacer qué en la plataforma.

Este "quién" puede ser un usuario o grupo de su Azure Active Directory, un usuario de otro inquilino de Azure Active Directory, o una aplicación registrada o entidad de servicio.

RBAC en Azure ofrece más de 75 roles diferentes entre los que elegir, y si no puede encontrar la asignación de roles específica para la necesidad particular de su organización, también puede crear sus propios roles personalizados a partir de Azure PowerShell.

Directiva de Azure

Otra fuente de control está disponible a través de **Azure Policy**. Este es un verdadero mecanismo de administración y control de gobernanza de Azure. Como organización, usted define las directivas de Azure: archivos JSON en los que se especifican los requisitos de recursos de Azure que desea aplicar antes de que la implementación de los recursos de Azure pueda realizarse correctamente. Por ejemplo, está la obligación de utilizar ciertas regiones de Azure debido a las regulaciones de cumplimiento, o permitir solo ciertos tamaños de máquinas virtuales de Azure en su suscripción para mantener el control de los costos o, tal vez, podría tener ciertos estándares de nomenclatura que quiera exigir para los recursos de Azure, que optimicen la administración de activos y las regulaciones de CMDB. Un último ejemplo de algo que muchas empresas encuentran útil es la aplicación del uso de etiquetas. Una etiqueta es como un rótulo que se puede agregar a un grupo de recursos o a recursos individuales, por ejemplo, un centro de costos o una unidad de negocio. Es principalmente gracias a estas etiquetas que un administrador de facturación de Azure puede tener una visión clara de para qué se utiliza un recurso de Azure o, al menos, a qué unidad de negocio o centro de costos pertenece este recurso.

Las directivas de Azure se pueden agrupar en las denominadas **iniciativas de directivas de Azure**. Esto ayuda a aplicar varias directivas a la vez. Después de definir las directivas y las iniciativas de directivas de Azure, estas deben asignarse a un ámbito. Este ámbito podría ser una suscripción, un grupo de recursos o recursos individuales de Azure.

Azure Blueprints

Otro mecanismo disponible en Azure actualmente para ayudar con el control de la gobernanza es **Blueprints**. Azure Blueprints (*Figura 6*) permite a los arquitectos de la nube y a los equipos de TI definir una estructura de instrucciones reutilizables y repetibles para la implementación y configuración, de acuerdo con los estándares, las regulaciones y los requisitos de la empresa.

A través de la dependencia de una combinación de roles, controles e infraestructura como código, Azure Blueprints organiza el ciclo de vida completo de la implementación de los recursos de Azure.

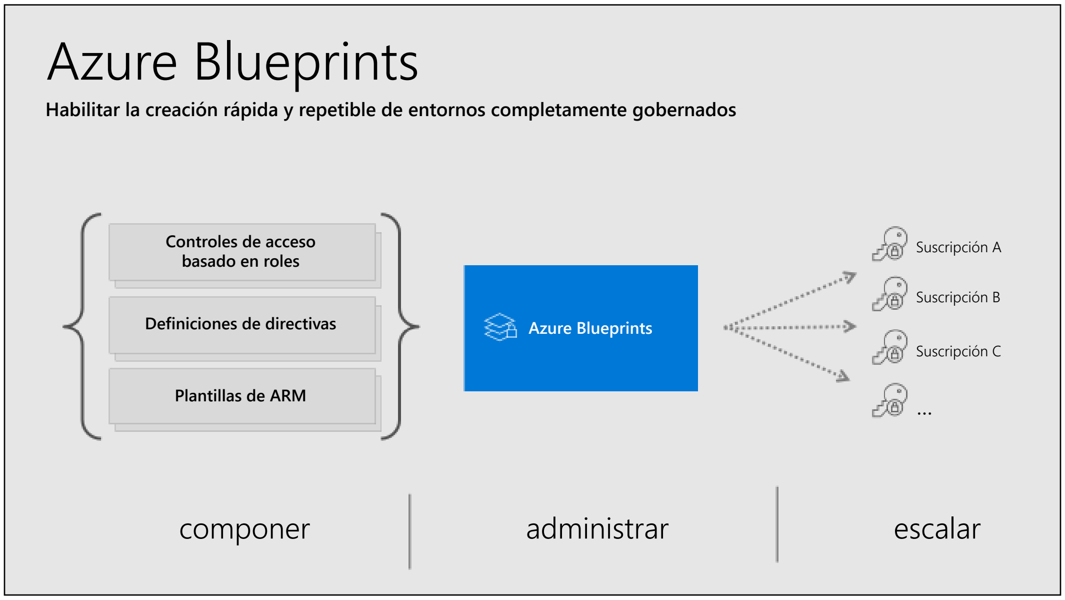

Figura 6: Azure Blueprints

Blueprints se basa en artefactos, que son un conjunto de configuraciones, parámetros, plantillas de implementación de infraestructura de Azure como código y plantillas de directivas.

Estándares de nomenclatura

Otro aspecto fundamental de su estrategia de migración a una nube pública como Azure es conocer bien los estándares de nomenclatura. En Azure todo se basa en los recursos de Azure. Muchos de estos usan nombres dinámicos que no puede cambiar. Otros servicios se implementan en un dominio de espacio de nombres fijo (`azurecri.io` para Azure Container Registry, `blob.core.windows.net` para blobs de Cuentas de Azure Storage, `azurefd.net` para Azure Front Door, etc.).

Además, muchos recursos de Azure tienen requisitos (y limitaciones) en relación con el uso de ciertos caracteres, mayúsculas, valores numéricos y caracteres complejos.

Hay documentación muy buena disponible sobre este tema específico:

`https://bit.ly/2FyxKjh`

Grupos de recursos

Otro elemento al que quiero referirme como parte de la información previa a la migración son los grupos de recursos. Si bien no es tan difícil entender lo que hacen (son grupos de recursos de Azure), hay mucha confusión en torno a ellos, en concreto, con respecto a cómo organizarlos o cómo organizar sus recursos en ellos.

Como punto de partida, realmente depende de su organización. Microsoft no impone la ubicación de los recursos ni la forma de organizar los grupos de recursos (con algunas excepciones). Algunas organizaciones tienen un grupo de recursos por carga de trabajo; otras definen los grupos de recursos en función de los tipos de recursos (un grupo de recursos de red, un grupo de recursos de almacenamiento, etc.). Se puede ver un ejemplo de este enfoque en el diagrama de la *Figura 7*. Esto podría ayudar a asignar el RBAC, manteniendo la misma estructura en capas que su centro de datos local. Otras organizaciones utilizan ubicaciones geográficas de centros de datos como orientación (West-EuropeRG, East-USRG, etc.).

Y respecto al tema de los grupos de recursos y las ubicaciones de recursos de Azure, especificar la ubicación es un requisito difícil para cualquier recurso de Azure, puesto que la mayoría de los recursos de la plataforma son específicos de la región. La complejidad surge cuando tiene un grupo de recursos en una ubicación que contiene recursos en una ubicación diferente. Aunque técnicamente está bien, esto podría causar interrupciones cuando ya no se puede acceder a la región de Azure en la que se encuentra el grupo de recursos. Los recursos permanecerían (por ejemplo, una máquina virtual seguiría ejecutándose), pero no se podrían realizar cambios en la máquina virtual (puesto que los metadatos de la información no se pueden escribir en el grupo de recursos).

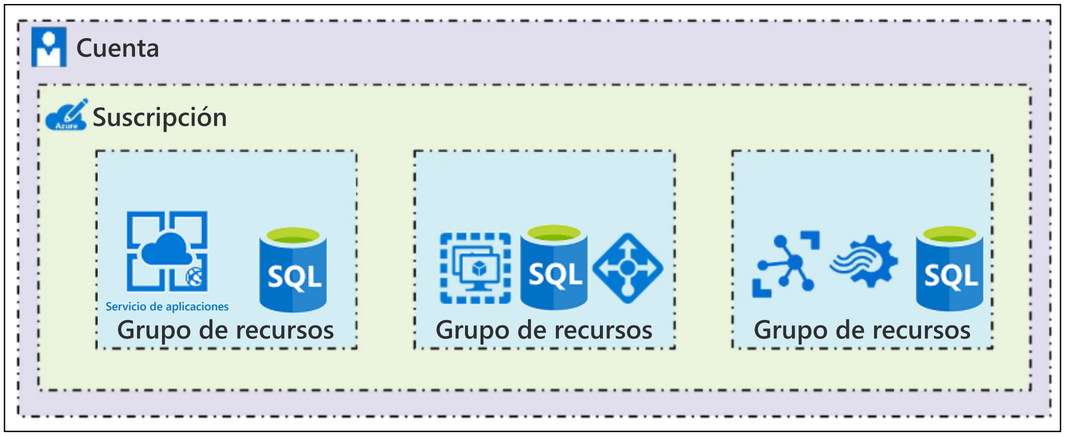

Figura 7: Grupos de recursos de Azure

Ahora debería quedar claro que la identidad y el control son temas importantes que se deben abordar antes de iniciar la migración (o implementación) real de sus cargas de trabajo empresariales en Azure.

Mediante la alineación de los arquitectos de la nube con las necesidades empresariales, el otorgamiento de los roles y permisos correctos, la optimización de la seguridad en la nube a través de características de identidad de Azure, como la autenticación multifactor (MFA), el acceso condicional, la administración de identidades privilegiadas y Azure Identity Protection, puede optimizar drásticamente su seguridad en la nube. En la mayoría de las situaciones, eso significa de inmediato que también optimizará la seguridad de sus centros de datos locales, por lo que este es un verdadero beneficio de nube híbrida.

Azure Resource Graph

Aunque no se creó específicamente como un servicio de gobernanza, Azure Resource Graph definitivamente puede ayudar a obtener una mejor visualización de los recursos de Azure implementados por una organización. Resource Graph es un servicio de Azure diseñado para proporcionar una forma rápida y fácil de administrar y explorar todos los recursos dentro de una sola suscripción, o incluso en varias suscripciones.

Azure Resource Graph le permite ejecutar consultas de filtrado, lo que reduce los resultados de lo que está buscando.

Aunque Azure Resource Manager también le permite recopilar recursos de Azure filtrados, esta herramienta comienza por los proveedores de recursos individualmente. Si desea obtener una vista de las redes virtuales de Azure, podría "llamar" al proveedor de recursos de red. Luego, debería conectarse al proveedor de recursos de la máquina virtual para obtener información sobre las máquinas virtuales.

Azure Resource Graph hace esto de forma diferente y de una manera que le permite reunir información en todos esos recursos, sin comunicarse con cada uno de los proveedores de recursos individualmente.

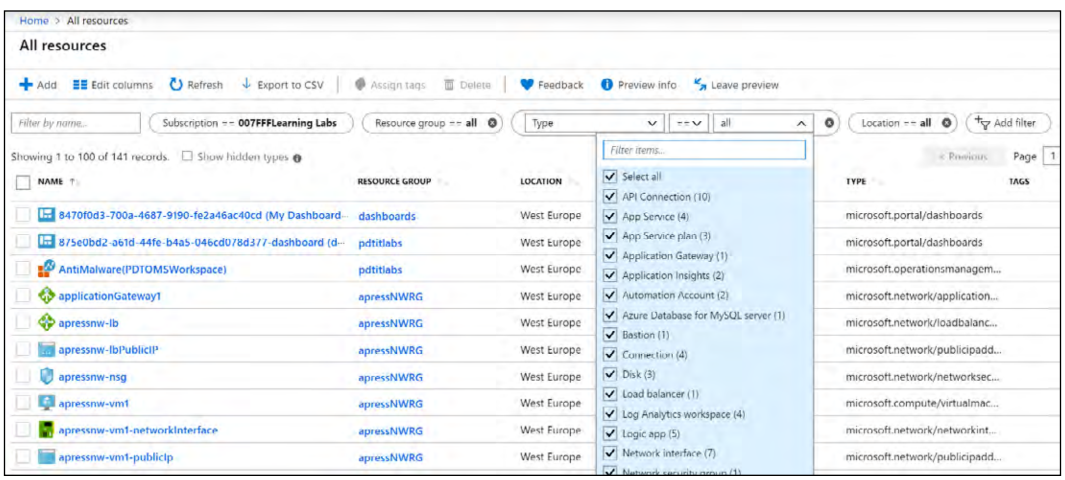

Figura 8: Azure Resource Graph

En la *Figura 8*, se muestra un resultado de esta herramienta con el portal de Azure. Además del portal de Azure, Resource Graph también se puede usar desde Azure PowerShell y la CLI de Azure, con el poderoso y rápido lenguaje de consulta KUSTO.

Control y administración de costos

Un último elemento que se ajusta a la perfección en el tema de la gobernanza de Azure es la administración de costos. Microsoft adquirió recientemente Cloudyn, una herramienta de informe de costos para varias nubes. El servicio de Cloudyn permite a cualquier organización abrir paneles detallados, que muestran el consumo de costos de cualquier recurso o grupo de recursos de Azure, en función del tipo de recurso, la región o las etiquetas asociadas a los recursos de Azure en sí.

Microsoft ahora ha integrado completamente la experiencia de Cloudyn en el portal de Azure, en un servicio específico llamado Cost Management, que le ofrece paneles de informes (consulte la *Figura 9*) y que tiene varias opciones para elegir.

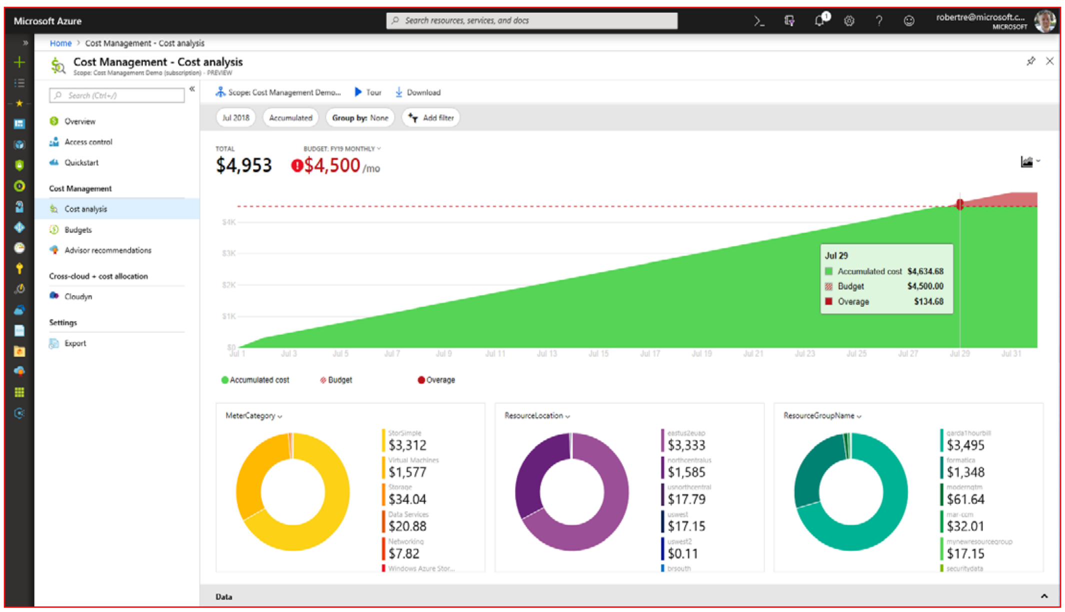

Figura 9: Azure Cost Management: análisis de costos

Otra característica de Administración de costos lanzada recientemente son los *Presupuestos de costos*. Esta es una configuración flexible, que le permite definir un límite máximo de consumo de costos para un recurso o grupo de recursos de Azure específico. Una vez que se alcance el monto del presupuesto (o cualquier porcentaje, como el 80 %), los administradores de Azure pueden ver un informe de los resultados en el panel (consulte la *Figura 10*) o recibir una notificación de alerta por correo electrónico, por ejemplo.

Tenga en cuenta que la característica de presupuesto no detiene el consumo de Azure como tal, ni se eliminará el recurso de Azure, pero por lo menos es una ayuda útil en la gobernanza de costos.

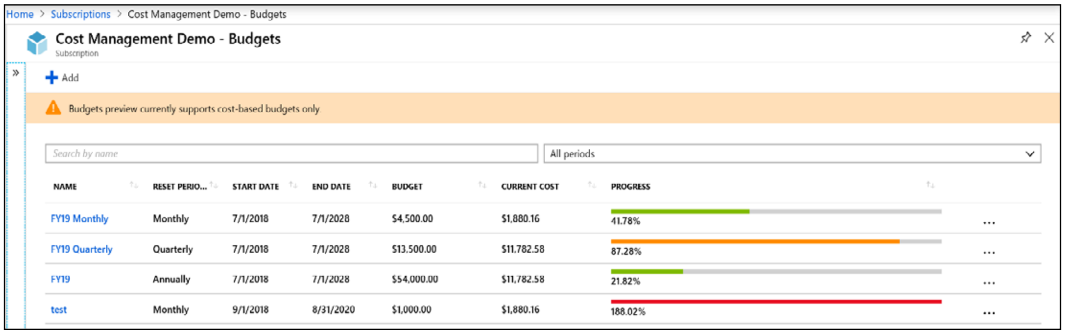

Figura 10: Presupuesto de costos

Resumen de la sección

En esta sección, lo guiamos por varios servicios y funcionalidades de gobernanza de Azure que puede implementar en sus suscripciones de Azure. Comenzando por los Grupos de administración, que le permiten asignar directivas a varias suscripciones a la vez, aprendió sobre las directivas de Azure y Azure Blueprints. También hablamos de Azure Identity como un mecanismo de gobernanza, que ofrece RBAC. En la última parte, abarcamos el nuevo servicio, Cost Management, como otro instrumento de gobernanza.

Ahora tiene un conocimiento apropiado de las capas fundamentales de la migración a la nube. Demos un vistazo a algunas de las herramientas y procesos de migración que Microsoft tiene disponibles hoy para ayudar a optimizar esta operación.

Herramientas y procesos de migración

Después de realizar las evaluaciones de su entorno de origen y preparar el camino hacia un entorno de nube sólido que considere la identidad, el cumplimiento y la gobernanza, está aquí para dar el siguiente paso: ejecutar sus migraciones. Ya hablamos brevemente acerca de varias de ellas, pero en gran medida desde la perspectiva de las funcionalidades de evaluación que las acompañan. En esta sección, lo guiaré por los enfoques de migración propiamente tales. Hay varios enfoques posibles y el que elija dependerá de diferentes factores.

Migraciones manuales

La primera opción en la que pensamos es ejecutar una migración manual. A partir de su carga de trabajo de origen existente, debe crear un entorno similar en Azure y copiar los datos. Examinemos algunos ejemplos.

Migración de discos VHD

Si su entorno de origen es Hyper-V y su aplicación tiene suficiente tiempo de inactividad, podría considerar realizar una migración manual mediante la copia de discos VHD en Azure Storage. Actualmente, Azure es compatible con máquinas Hyper-V Gen1 y Gen2, con un tamaño máximo de disco VHD de 1023 GB.

Si el entorno de origen es VMware, aunque hay un poco más de trabajo, no debería ser un factor de bloqueo para copiar la VM en Azure. Las herramientas como Microsoft Virtual Machine Converter pueden ayudarlo a transformar el archivo VMDK en un formato de archivo VHD.

Si solo desea migrar un único VHD y ejecutar una máquina virtual de Azure fuera de él, no tiene que generalizar la imagen del VHD; en su lugar, puede configurarlo como un disco especializado. Sin embargo, si desea utilizar este VHD de origen como un disco de plantilla para varias implementaciones de VM de Azure, primero debe generalizar el disco. Esto se realiza con `sysprep` desde dentro de la misma VM de origen.

Una vez que el disco esté listo para copiarse, utilice el cmdlet `Add-azVHD` de PowerShell para cargar el VHD a una cuenta de Azure Storage. Otras opciones disponibles incluyen usar AzCopy y la herramienta gratuita Explorador de Azure Storage.

A continuación, defina una imagen de este VHD cargado mediante los cmdlets `New-AzImageConfig` y `New-AzImage` de PowerShell.

Por último, cuando tenga su imagen disponible, puede continuar con la implementación de una nueva máquina virtual de Azure en función de esta imagen de origen.

En el siguiente vínculo, se documenta orientación detallada paso a paso de cómo lograrlo:

`https://bit.ly/35zF8FK`

Migración de bases de datos de SQL con bacpac

Si su solución de base de datos de origen es una base de datos de SQL Server, podría estar familiarizado con la solución de copia de seguridad integrada de SQL Server y almacenar su base de datos en un archivo BACPAC. Esta es la forma perfecta de migrar su base de datos a Azure SQL, si se permite el tiempo de inactividad.

Después de implementar una nueva instancia de Azure SQL Database en Azure (*Figura 11*), copie el archivo bacpac en Azure Storage. Vaya a la opción de importación de base de datos en Azure SQL Database y listo. Realmente así de fácil puede ser una migración.

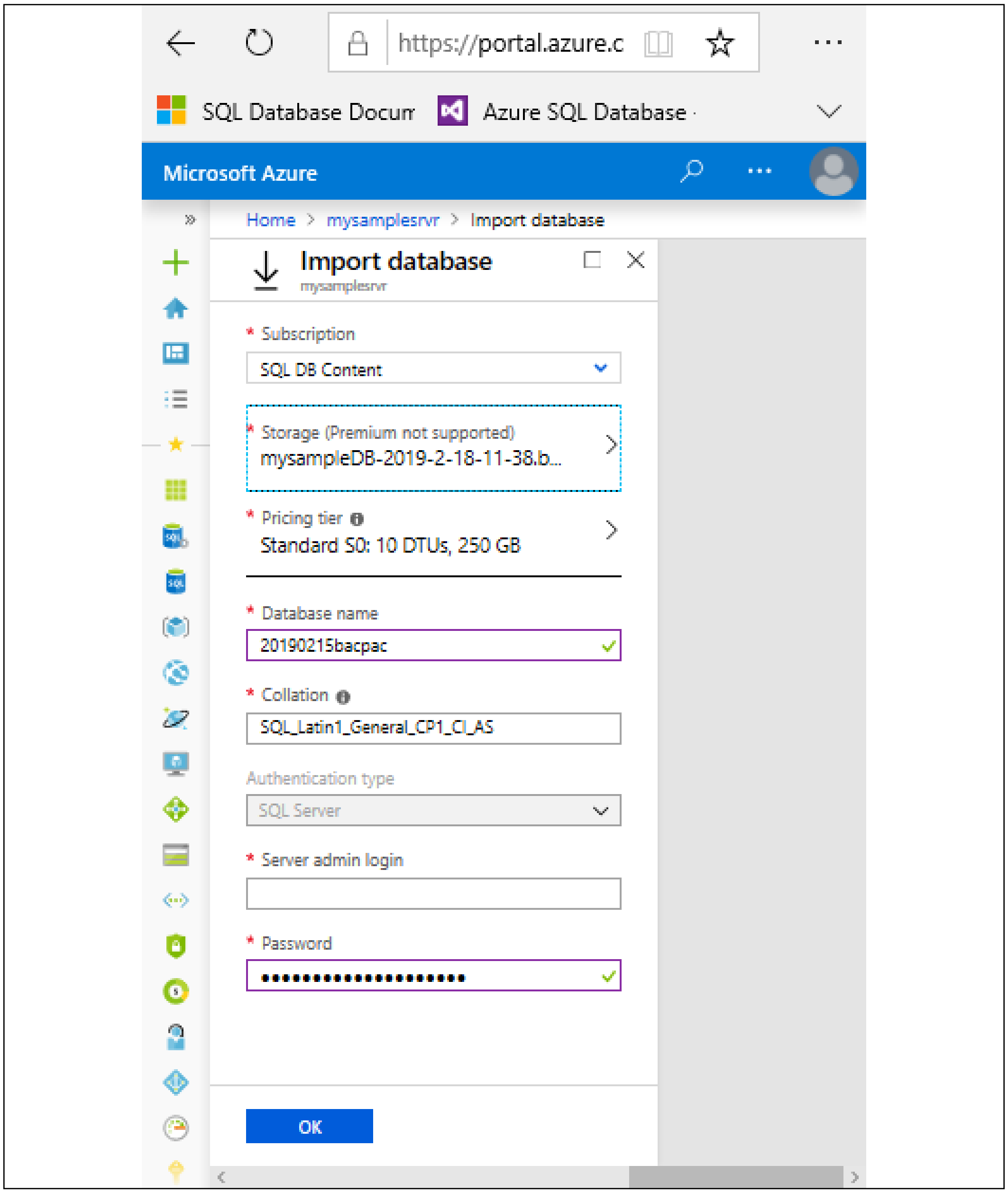

Figura 11: Importación de SQL Azure desde un archivo bacpac

Migración de sitios web a Azure Web Apps

La otra carga de trabajo común que migrar a Azure es una aplicación web o un sitio web. Web Apps de Azure App Service es compatible con los servidores web de Windows y los servidores web Apache de Linux como entornos de origen subyacentes, además de una lista completa de lenguajes y marcos de desarrollo (.NET, Java, Python, Node.js, Ruby y más).

Si tiene su código fuente disponible, lo más probable es que lo inserte directamente en una aplicación web de Azure y ejecute su sitio desde Azure. Las herramientas como Visual Studio y Visual Studio Code proporcionan este mecanismo de publicación de inmediato. Si utilizó la herramienta de Evaluación de migración de App Service para una carga de trabajo local, también puede usar la misma herramienta para realizar el paso de migración de contenido web real. La *Figura 12* muestra una captura de pantalla de cómo se ve el proceso de migración desde la herramienta.

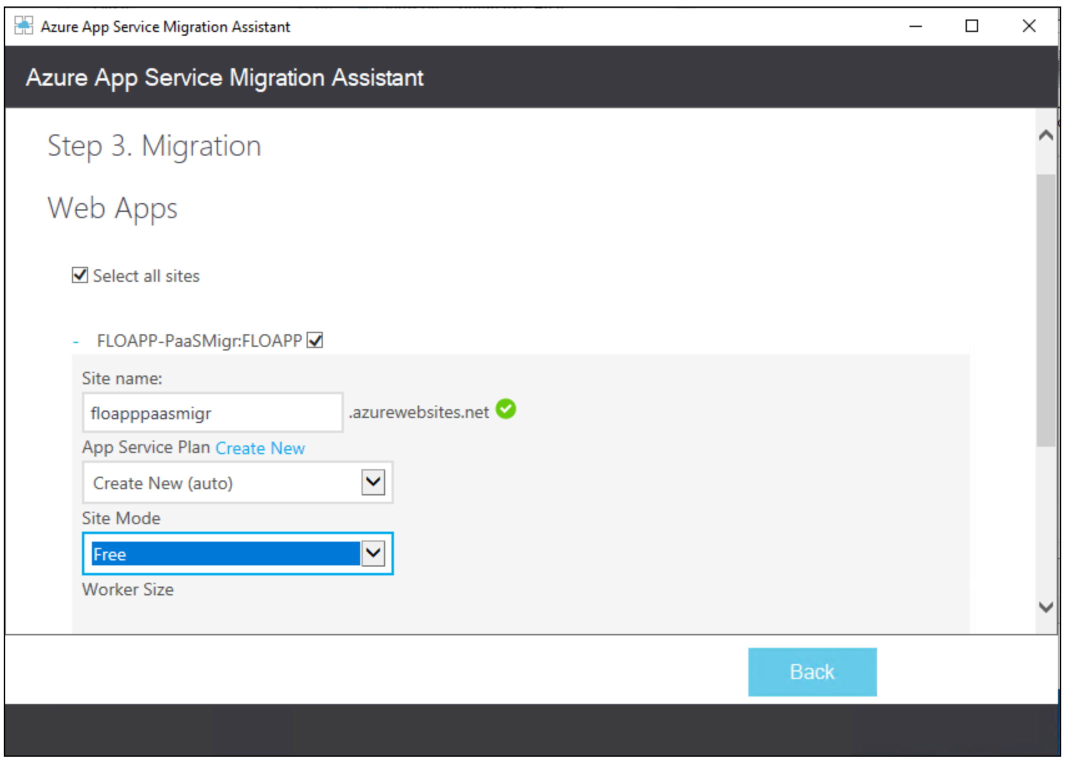

Figura 12: App Service Migration Assistant

Azure Migration Center

Ya describimos el nuevo Azure Migration Center en términos de su compatibilidad y orientación en la realización de evaluaciones de cargas de trabajo de origen. Sin embargo, obviamente, también puede guiarlo a través de la migración real de estos sistemas. Gracias a las últimas actualizaciones, solo publicadas en julio de 2019, Azure Migration Center se ha convertido en el lugar principal que debe revisar cuando necesita migrar sistemas de soluciones de datos o aplicaciones a Azure. Por lo tanto, Azure Migration Center ahora se ha convertido en un núcleo central para iniciar, ejecutar y realizar un seguimiento de sus proyectos de migración, ya sea que desee migrar a máquinas virtuales de Azure, soluciones de datos de Azure o Azure App Service.

Azure Data Box

Por ahora, debe quedar claro que tiene un conjunto completo de opciones para migrar sus cargas de trabajo a Azure, sin importar si el entorno de destino es IaaS o PaaS. Los escenarios mencionados antes probablemente cubren alrededor del 80 % al 90 % de los escenarios de migración.

Hay un último escenario sobre el que me gustaría hablar aquí, que es la migración de grandes volúmenes de datos (piense en cientos de terabytes o incluso petabytes). Estos volúmenes de datos más grandes no son los mejores candidatos para la migración manual, tampoco migrará esos entornos de almacenamiento a máquinas virtuales de Azure con Azure Site Recovery. La buena noticia es que hay otra forma más flexible de hacerlo: **Azure Data Box**.

Azure Data Box permite la migración de datos sin conexión, en función de un dispositivo físico que introduce en su centro de datos. Los datos se descargan en este dispositivo físico, que se transporta de forma segura a la región de Azure más cercana, donde los datos se copian en el entorno de Azure. Todos los datos en tránsito y en reposo están cifrados con AES para seguridad y cumplimiento avanzados.

Según el tamaño del volumen de datos, puede elegir cualquiera de los tres modelos que se muestran en la *Tabla 2*:

	**Azure Data Box:** • Un dispositivo resistente, que admite una capacidad de hasta 100 TB, que ofrece protocolos de área de almacenamiento estándares y se integra con herramientas de copia de datos estándar (como Robocopy). Esta unidad ofrece cifrado de 256 bits.
	**Azure Data Box Disk:** • Un disco SSD robusto con una interfaz SATA/USB, que admite hasta 8 TB por disco. Este disco ofrece cifrado de 128 bits.
	**Azure Data Box Heavy:** • Esto es para cantidades masivas de datos, que admiten hasta 1 PB de volumen de datos.

Tabla 2: Modelos de Azure Data Box para la migración de grandes volúmenes de datos sin conexión

Además de estas soluciones de migración de datos **sin conexión** , la oferta de Azure Data Box se extendió con dos soluciones de migración **en línea** adicionales, Azure Data Box Edge y Azure Data Box Gateway, que se muestran en la *Tabla 3*:

	**Azure Data Box Edge:** • Mediante la creación de un vínculo activo entre el entorno local y Azure, Data Box Edge proporciona un método sencillo y en línea para cargar o descargar datos desde y hacia Azure. El chasis, que parece un servidor físico de IU, admite hasta 12 TB de capacidad de datos.
	**Azure Data Box Gateway:** • Data Box Gateway es un dispositivo virtual que se implementa en el hipervisor (VMware o Hyper-V), que actúa como un gateway de transferencia de datos desde el entorno local hasta Azure, y es compatible con los protocolos NFS y SMB.

Tabla 3: Modelos de Azure Data Box para la migración de grandes volúmenes de datos en línea

Después de analizar Azure Data Box, pasemos a la implementación de Azure.

Implementación de un entorno nuevo de Azure

Otra estrategia de migración que debemos revisar es implementar un entorno nuevo en Azure, lo que significa implementar un centro de datos virtual desde cero. En lugar de realizar una migración mediante "lift and shift", como se recomienda en los escenarios anteriores, podría ser beneficioso comenzar con un entorno totalmente nuevo y solo migrar lo mínimo (lo que probablemente serían solo datos, como código fuente, recursos compartidos de archivos y soluciones de datos).

Con el fin de que este enfoque tenga éxito, debe entender bien cómo diseñar y crear un entorno de este tipo, asignándolo con sus requisitos empresariales desde una perspectiva técnica y no técnica.

Por supuesto, lo mismo es válido para los otros escenarios de migración descritos antes.

Para el resto de este capítulo, supondremos que usará IaaS y confiará en las diversas funcionalidades de centro de datos virtual disponibles en Azure.

Conceptos básicos de la administración de IaaS de Azure

Como se describió antes, IaaS se refiere a la creación de un centro de datos virtual. Esencialmente, esto se refiere a lo siguiente:

- Redes
- Almacenamiento
- Proceso

Ampliado con funcionalidades generales de seguridad, gobernanza y supervisión.

Cada uno de estos bloques de creación de la arquitectura se describirán con más detalle en esta sección.

Redes

La base central de cualquier centro de datos, físico o virtual, son las redes. Aquí es donde se definen los intervalos de direcciones IP y la configuración de comunicación del firewall, además de definir qué máquinas virtuales pueden conectarse entre sí. A continuación, también definirá cómo se establecerá la conectividad del centro de datos híbrido entre los centros de datos locales y los centros de datos de Azure.

Debe pensar e implementar los siguientes servicios de Azure desde cero:

- Conectividad híbrida del centro de datos mediante VPN de sitio a sitio de Azure o ExpressRoute.
- Creación de las redes virtuales de Azure y las subredes correspondientes.
- Implemente funcionalidades similares al firewall con grupos de seguridad de red de Azure, grupos de seguridad de aplicaciones, Firewall de Azure o dispositivos virtuales de red de terceros.
- Considere cómo realizará la administración remota de las máquinas virtuales. En cuanto a la seguridad avanzada, el acceso "just-in-time" a máquinas virtuales desde Azure Security Center o el nuevo Azure Bastion (versión preliminar pública) podrían ser buenas opciones. Si no tiene ninguna de estas funciones, asegúrese de que sus sesiones de RDP/SSH estén protegidas por firewall y que nunca se exponga directamente el host de administración o las máquinas virtuales a la Internet pública.

- Al igual que en el centro de datos de su empresa, Azure admite capacidades de equilibrio de carga en su red virtual. Puede elegirlas de Azure Load Balancer, un equilibrador con capacidad de 4 capas, que admite tráfico TCP y UDP en todos los puertos. Si desea equilibrar la carga de los protocolos de aplicación web (HTTP/HTTPS), podría ser interesante implementar Azure Application Gateway, un servicio de equilibrio de carga de 7 capas, que se puede ampliar con un Firewall de aplicaciones web (WAF) para seguridad avanzada y detección de amenazas. La última opción para el equilibrio de carga es la implementación de un dispositivo de equilibrio de carga de terceros de proveedores de confianza, como Kemp, F5 o Barracuda.
- En caso de tener varios centros de datos de Azure para escenarios de carga de trabajo altamente disponibles, implemente Azure Traffic Manager o el nuevo servicio Azure Front Door, que permite la detección y el redireccionamiento en varias regiones de Azure para la conmutación por error y evita problemas de latencia.

Además de esto, ejecutar una infraestructura de red virtual de Azure (consulte la *Figura 13* para ver un diagrama de ejemplo de cómo podría verse este diseño de red) es similar a lo que sucede en sus propios centros de datos. Estas son algunas de las características y funcionalidades principales:

- **Incorpore su propia red**: esto se refiere al aspecto de definir los intervalos de IP de su red virtual interna de Azure (VNet). Azure es compatible con el direccionamiento IP estándar de clase A, clase B y clase C.
- **Direcciones IP públicas**: cada suscripción de Azure viene con una cantidad predeterminada de cinco direcciones IP públicas que se pueden asignar a los recursos de Azure (firewall, equilibrador de carga y máquina virtual). Estas direcciones IP se pueden definir como dinámicas o estáticas. Además, tenga en cuenta que nunca será un intervalo de direcciones IP, sino más bien direcciones independientes. Tampoco es posible incorporar su propio intervalo de direcciones IP públicas a un centro de datos de Azure.
- **Direcciones IP internas**: cuando implementa una VNet y subredes de Azure, tiene el control del intervalo de IP (basado en CIDR). Preferentemente, Azure asigna direcciones IP dinámicas a recursos como las máquinas virtuales (a diferencia de las direcciones IP fijas más tradicionales que usted asigna a los servidores de su centro de datos). En mi opinión, los únicos servidores que requieren una dirección IP fija en Azure serían clústeres de software, como SQL u Oracle, o cuando la máquina virtual ejecuta servicios DNS para la subred en la que se implementa.
- **Azure DNS**: Azure viene con un DNS completamente operativo como un servicio que puede aprovechar para su propia resolución de nombres. Cuando se implementa una red virtual y una subred de Azure, se hace referencia a este DNS de Azure de forma predeterminada. Sin embargo, puede cambiar esta configuración para hacer referencia a su propia solución de DNS, que puede ser una máquina virtual en ejecución en Azure o una solución de DNS local (siempre y cuando tenga implementada una conectividad híbrida).

- **IPv4/IPv6**: las redes virtuales de Azure son compatibles con IPv4, pero la compatibilidad con IPv6 actualmente está en versión preliminar. Cada vez más servicios internos, como las máquinas virtuales y la Internet de las Cosas (IoT), son compatibles con IPv6 para su conectividad de red, tanto entrante como saliente.

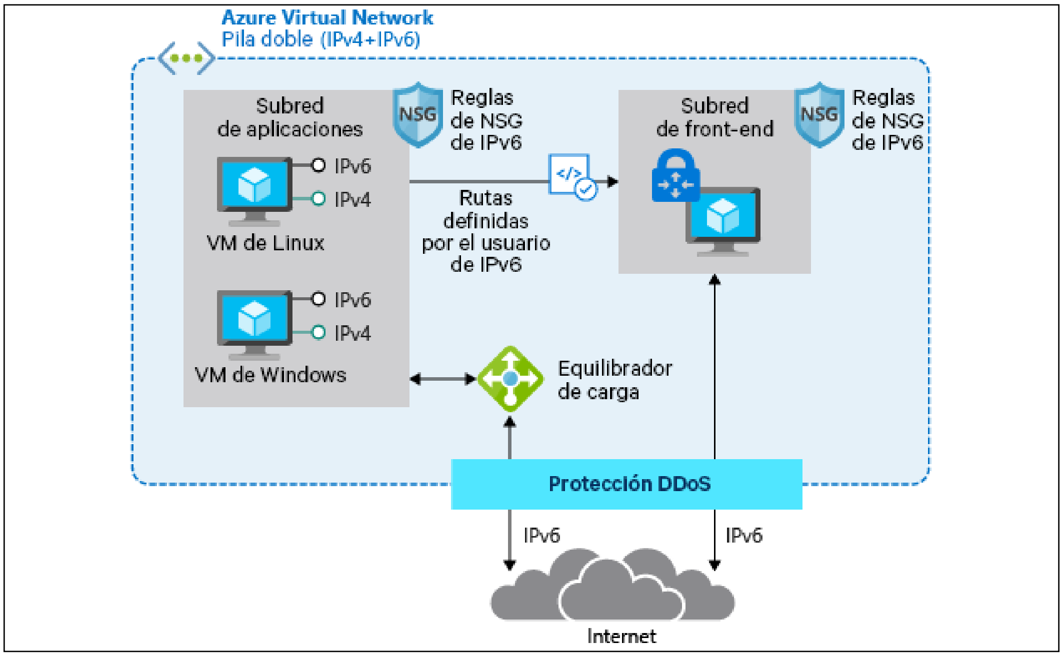

Figura 13: Arquitectura de red virtual de Azure

Puede encontrar información más detallada sobre los servicios de red de Azure, sus funcionalidades y cómo implementarlos y administrarlos en el siguiente vínculo:

`https://docs.microsoft.com/azure/virtual-network/`

Almacenamiento

La siguiente capa de la arquitectura de centro de datos virtual que destacaré es Azure Storage. Al igual que su escenario de centro de datos local, Azure ofrece varios servicios de almacenamiento diferentes:

- Azure Storage considera blobs, tablas, colas y filas
- Discos administrados de Azure para discos de máquinas virtuales
- Azure File Sync le permite sincronizar recursos compartidos de archivos locales con Azure
- Soluciones de big data de Azure

A continuación, describiré las características básicas de cada uno de estos servicios.

Cuentas de Azure Storage

Las cuentas de Azure Storage son la forma más fácil de consumir almacenamiento en Azure. Una cuenta de Azure Storage es muy similar a la solución NAS o SAN local, en la que se definen volúmenes o recursos compartidos.

Cuando implementa una cuenta de Azure Storage en una región de Azure, esta ofrece cuatro casos de uso diferentes:

- **Almacenamiento de blobs**: probablemente el tipo de almacenamiento más común. Los blobs le permiten almacenar conjuntos de datos más grandes (archivos VHD, imágenes, documentos, archivos de registro, etc.) dentro de contenedores de almacenamiento.
- **Archivos**: recursos compartidos de archivos de Azure a los que puede conectarse mediante el protocolo de recurso compartido de archivo SMB. Esta característica es compatible con los puntos de conexión de Windows (SMB) a Linux (montaje).
- **Tablas y colas**: estos servicios son compatibles principalmente con el entorno de la aplicación. Las tablas son una alternativa rápida para almacenar datos, mientras que las colas se pueden usar para enviar información de telemetría.

Las cuentas de Azure Storage ofrecen varias opciones con respecto a la alta disponibilidad:

- **LRS (almacenamiento con redundancia local)**: todos los datos se replican tres veces dentro del mismo edificio de centro de datos de Azure.
- **ZRS (almacenamiento con redundancia de zona)**: todos los datos se replican tres veces en diferentes edificios de Azure en la misma región de Azure.
- **GRS (almacenamiento con redundancia geográfica)**: todos los datos se replican tres veces en el mismo centro de datos de Azure y se replican tres veces más en una región diferente de Azure (se considera el cumplimiento geográfico, para garantizar que los datos nunca abandonen los límites regionales).
- **GRS (acceso de lectura)**: similar a GRS, pero los datos replicados en la otra región de Azure se almacenan en un formato de solo lectura. Solo Microsoft puede accionar un interruptor para hacer que esta sea una copia modificable.

Discos administrados de Azure

Durante mucho tiempo, las **cuentas de Azure Storage** fueron la única opción de almacenamiento en Azure para crear discos virtuales destinados a sus máquinas virtuales y se remontan a 10 años con Azure Classic. Aunque las cuentas de almacenamiento eran buenas, también venían con algunas limitaciones, y las más importantes eran el rendimiento y la escalabilidad.

En ese momento fue cuando Microsoft lanzó una nueva arquitectura de almacenamiento de disco virtual, alrededor de 3 años atrás, llamada discos administrados. En el escenario de los **discos administrados**, Azure Storage realiza la mayor parte del trabajo por usted. No tiene que preocuparse de crear una cuenta de almacenamiento o de que existan problemas de rendimiento o escalabilidad; solo tiene que crear los discos y listo. A propósito del rendimiento, los discos administrados incluyen una lista completa de SKU para elegir, que ofrece de todo, desde discos con rendimiento promedio (500 IOPS-P10) hasta discos de alto rendimiento (IOPS 20 000-P80), y también hay tipos de disco Ultra SSD disponibles, que van hasta 160 000 IOPS con este modelo.

Es importante tener en cuenta que el rendimiento del subsistema de disco también depende en gran medida del tamaño real de la VM que asigna a la máquina virtual de Azure. Lo mismo ocurre con la capacidad de almacenamiento, puesto que no todas las VM de Azure admiten un gran número de discos. Además de tener volúmenes de disco más grandes disponibles desde la perspectiva del sistema operativo, también se podría usar una cantidad mayor de discos para configurar la fragmentación de disco en un subsistema de disco de máquina virtual de Azure. Esto también generaría un mejor rendimiento de IOPS.

Azure File Sync

Si actualmente tiene varios servidores de archivos, es muy probable que tenga una solución para mantenerlos sincronizados. En el mundo de Windows Server, esto podría hacerse con DFS (Sistema de archivos distribuido) que ha sido un servicio básico desde Windows Server 2012. Cuando se migran aplicaciones a la nube, es posible que también se deban migrar las dependencias del recurso compartido de archivos. Una opción sería implementar servidores de archivos basados en máquinas virtuales de Azure para esto. Pero eso podría ser excesivo, en especial, si esas máquinas solo ofrecen servicios de uso compartido de archivos. Una alternativa válida es implementar Azure File Sync.

Con Azure File Sync (consulte la *Figura 14* para ver un ejemplo de la arquitectura), puede centralizar sus recursos compartidos de archivos en Azure Files (como parte de las cuentas de Azure Storage) y utilizarlos de la misma manera que los servidores de archivos de Windows locales, pero sin la capa intermedia de Windows Server. A partir de un Servicio de sincronización de almacenamiento de Azure, se crea un grupo de sincronización. Dentro de este grupo de sincronización, se configuran los servidores registrados. Una vez que se registra este servidor, se implementa en él el agente de Azure File Sync, que se encarga del proceso de sincronización de los recursos compartidos de archivos.

Azure File Sync también proporciona funcionalidad de almacenamiento en niveles, que le permite ahorrar en costos de almacenamiento cuando se almacenan datos de archivo que no consulta con frecuencia, pero que debe conservar debido al cumplimiento de datos.

La desduplicación de datos es otra ventaja para estos volúmenes habilitados para niveles de nube en Windows Server 2016 y 2019.

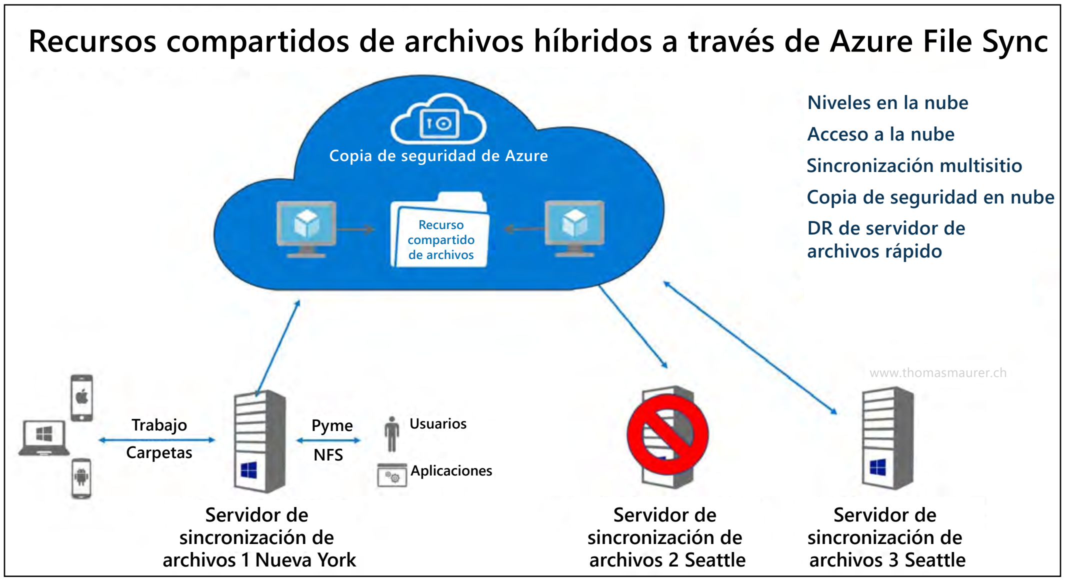

Figura 14: Azure File Sync

Proceso

Esto nos lleva a la siguiente capa lógica del centro de datos virtual: la implementación de máquinas virtuales. Este es probablemente uno de los casos de uso más comunes de la nube pública.

Como se mencionó en los párrafos introductorios al principio de este capítulo, la virtualización ha cambiado drásticamente la forma en que las organizaciones implementan y administran su infraestructura de TI. Gracias a soluciones como VMware e Hyper-V, los sistemas se pueden consolidar con una superficie más pequeña de servidor físico, son fáciles de implementar, fáciles de recuperar y proporcionan otros beneficios cuando se trata de migrarlos a través de entornos (de desarrollo y prueba a producción, por ejemplo).

La mayoría de los aspectos y características que conoce de la ejecución de máquinas virtuales por su cuenta se pueden asignar a las máquinas virtuales en ejecución en Azure. A menudo, describo esto como "solo otro centro de datos". (Sin embargo, obviamente Azure es mucho más que eso…)

Al igual que con la capa de redes, proporcionaré una visión general de los diversos beneficios que conlleva la implementación de cargas de trabajo de máquinas virtuales en Azure:

- La implementación de máquinas virtuales de Azure permite obtener **agilidad y escalabilidad** y ofrece diversos procesos administrativos para hacerlo. Aproveche su experiencia con PowerShell para implementar y administrar máquinas virtuales, simplemente como las implementaría y administraría en su centro de datos. O bien, extienda su centro de datos a la infraestructura como código, lo que le permite implementar máquinas virtuales desde las plantillas de Azure. Esto no solo permite la implementación de recursos de Azure, sino que también se puede ampliar con herramientas de administración de configuración, como Desired State Configuration de PowerShell, Chef, Puppet y más.
- Las máquinas virtuales de Azure vienen con un SLA predeterminado de 99,9 %, para una sola máquina virtual implementada con discos premium. Si sus aplicaciones empresariales requieren un SLA aún mejor, implemente sus máquinas virtuales como parte de un **conjunto de disponibilidad de máquinas virtuales**. Esto garantiza un SLA del 99,95 %, en el que las diferentes VM del conjunto de disponibilidad nunca se ejecutarán en el mismo bastidor físico en el centro de datos de Azure. O bien, implemente las VM en una **zona de disponibilidad de máquinas virtuales**, lo que aumentará el nivel del SLA al 99,99 %. En esta arquitectura, las VM (dos o más instancias) se propagarán entre diferentes edificios de centros de datos físicos de Azure en la misma región de Azure.
- Es posible que aplique requisitos empresariales, lo que le obliga a implementar máquinas virtuales en diferentes ubicaciones, a cientos o miles de millas de distancia entre sí. O bien, tal vez desee ejecutar las aplicaciones tan cerca del cliente o usuario final como sea posible, para evitar cualquier problema de latencia. Azure puede adaptar este escenario exacto, puesto que todas las regiones del centro de datos de Azure están interconectadas entre sí mediante **cableado de red troncal de Microsoft**. Desde una perspectiva de conectividad, puede usar el emparejamiento de VNet de Azure o una VPN de sitio a sitio para crear centros de datos de varias regiones para **alta disponibilidad lista para empresas**.
- Si no necesita alta disponibilidad en tiempo real, sino que evalúa un escenario de recuperación ante desastres rápido y sólido, Azure tiene un servicio integrado en la plataforma conocida como **Azure Site Recovery**. En función de la replicación del cambio de estado del disco de máquina virtual, las VM de Azure se mantendrán sincronizadas en varias regiones de Azure (manteniendo los límites de cumplimiento y soberanía de datos). En caso de que ocurra un error o un desastre con una de las máquinas virtuales, el administrador de TI puede comenzar un proceso de conmutación por error manual (o con scripts), lo que garantiza el tiempo de actividad de las cargas de trabajo de la aplicación en la otra región del centro de datos. El principal beneficio, además de la conmutación por error rápida, es el ahorro de costos. Solo paga por el almacenamiento subyacente, siempre y cuando las máquinas virtuales de recuperación ante desastres estén sin conexión. Durante el período de desastre, solo paga por el costo real de consumo de las VM en ejecución durante la vigencia del escenario de desastre.

- Aunque los centros de datos de Azure están administrados por Microsoft y son de su propiedad, esto no significa que solo se limitan a ejecutar sistemas operativos de Windows Server y aplicaciones de Microsoft Server. **El 60 % de las máquinas virtuales de Azure** ejecutan hoy un sistema operativo Linux.
- Muchas aplicaciones empresariales, como SAP, Oracle y Citrix, están disponibles en **Azure Marketplace**, lo que permite una implementación más fácil, al igual que la mayoría de las demás cargas de trabajo de máquinas virtuales de Azure. A partir de imágenes preconfiguradas, cualquier organización puede implementar una arquitectura de aplicaciones empresariales rápidamente. Microsoft, junto con el proveedor, proporcionan soporte técnico para estas cargas de trabajo de soluciones de terceros.
- Azure ofrece **más de 125 tamaños distintos de máquinas virtuales**, cada uno con diferentes características y capacidades. A partir de una máquina virtual de núcleo único de CPU con 0,75 GB de memoria, puede aumentar a máquinas virtuales con 256 núcleos virtuales y más de 430 GB de memoria. A continuación, también tiene familias de máquinas virtuales específicas, que admiten cargas de trabajo específicas (por ejemplo, la familia de la serie N está equipada con un conjunto de chips Nvidia para aplicaciones de proceso gráfico de alta gama, la serie H se recomienda para infraestructura de **informática de alto rendimiento** (**HPC**) y también tiene tamaños de máquinas virtuales específicos para SAP y SAP HANA).

Para obtener una descripción general de los tipos y tamaños de máquinas virtuales de Azure, use este vínculo a la documentación de Azure:

`https://docs.microsoft.com/azure/virtual-machines/windows/sizes`

Para obtener una visión más amplia sobre las funcionalidades de procesos de máquinas virtuales de Azure en Azure, use este vínculo:

`https://azure.microsoft.com/product-categories/compute/`

La última característica de proceso de Azure que quiero destacar aquí es **Azure Confidential Computing**. A principios de mayo de 2018, Microsoft Azure se convirtió en la primera plataforma en la nube, lo que permitió nuevas funcionalidades de seguridad de datos. Basándose principalmente en tecnologías como Intel SGX y la seguridad basada en la virtualización (VBS), ofrece entornos de ejecución de confianza (TEE) en una nube pública. Cada vez más empresas migran sus cargas de trabajo críticas para el negocio a la nube, por lo que la seguridad se vuelve aún más crucial. Azure Confidential Computing tiene como objetivo ofrecer seguridad y protección de primer nivel para los datos en la nube. El concepto se basa en los siguientes dominios clave:

- **Hardware**: conjunto de chips Intel SGX desde una perspectiva de seguridad de hardware
- **Proceso**: la plataforma de procesos de Azure permite instancias de VM con TEE habilitada.

- **Servicios**: las plataformas seguras habilitan cargas de trabajo muy seguras, como blockchain.
- **Investigación**: el departamento de investigación de Microsoft está trabajando estrechamente con Azure PGs para mejorar continuamente las capacidades de esta plataforma de confianza.

Administración de la infraestructura de Azure (y más)

La última parte del proceso de migración y adopción de la nube, en relación con IaaS, es ofrecer una solución que permita una administración de su entorno de Azure preparada para la empresa. Y cuando sea posible, esta se debe extender a una solución de administración híbrida, en especial, durante un período de migración más largo, cuando tenga sistemas que se ejecuten en su centro de datos local, pero también en Azure.

Azure viene con una diversidad de herramientas de administración, supervisión y operaciones. A continuación, describiré algunas de ellas, las que probablemente comenzará a usar de inmediato de forma cotidiana.

Azure Monitor

Azure Monitor proporciona una solución de supervisión unificada para Azure, que ofrece un solo lugar para extraer información de métricas, como el consumo de CPU, los registros, incluidos los registros de eventos y aplicaciones, y cualquier otra información de telemetría generada por los servicios de Azure. Esto se puede ampliar para supervisar también las cargas de trabajo y las soluciones locales en ejecución, lo que genera una herramienta de supervisión única y eficaz.

Azure Monitor también ofrece análisis y diagnósticos avanzados, con tecnología de "machine learning". Estos se exponen en los servicios de Azure, como Azure Monitor para máquinas virtuales, Azure Monitor para contenedores, Azure Advisor y Azure Security Center, por nombrar solo algunos. Todos y cada uno de estos escenarios canónicos ofrecen recomendaciones e información sobre la forma en que se realiza la implementación de los recursos de Azure.

La supervisión en Azure normalmente se divide en dos categorías: fundamentos de supervisión, que son componentes que están disponibles de forma automática en la plataforma de Azure, sin tener que habilitar ningún servicio, y supervisión específica del escenario, que incluye servicios dentro de la plataforma de Azure que se pueden utilizar para la supervisión, pero que requerirán una configuración adicional o pueden incluir costos adicionales. Ejemplos de esta categoría serían Azure Monitor Log Analytics, que tiene un nivel gratuito y precios basados en el consumo, o Azure Security Center, que también tiene un nivel gratuito y un nivel de pago. Azure Monitor forma parte de la solución de supervisión general de Microsoft Azure. Este servicio lo ayuda a hacer seguimiento del rendimiento, mantener la seguridad e identificar tendencias.

También le permite consumir telemetría para tener visibilidad del rendimiento y el estado de sus cargas de trabajo en Azure. El tipo más importante de datos de telemetría de Azure son las métricas (que también se denominan contadores de rendimiento) emitidas por la mayoría de los recursos de Azure. Azure Monitor proporciona varias formas de configurar y consumir estas métricas para la supervisión y solución de problemas.

La información de Azure Monitor se basa en dos tipos principales de información de registro: las métricas y los registros. Las métricas son valores numéricos, que habitualmente se utilizan para obtener comentarios en tiempo real. Los registros, por otro lado, contienen mucho más detalle y normalmente se utilizan para recuperar o identificar correlaciones entre eventos y actividades. Estos también se almacenarían durante un período más largo, a menudo, según los requisitos de cumplimiento de una organización.

Las métricas características de un recurso de Azure se publican en la sección de **información general** de un recurso de Azure (consulte la *Figura 15* para ver un ejemplo relacionado con las aplicaciones web de Azure).

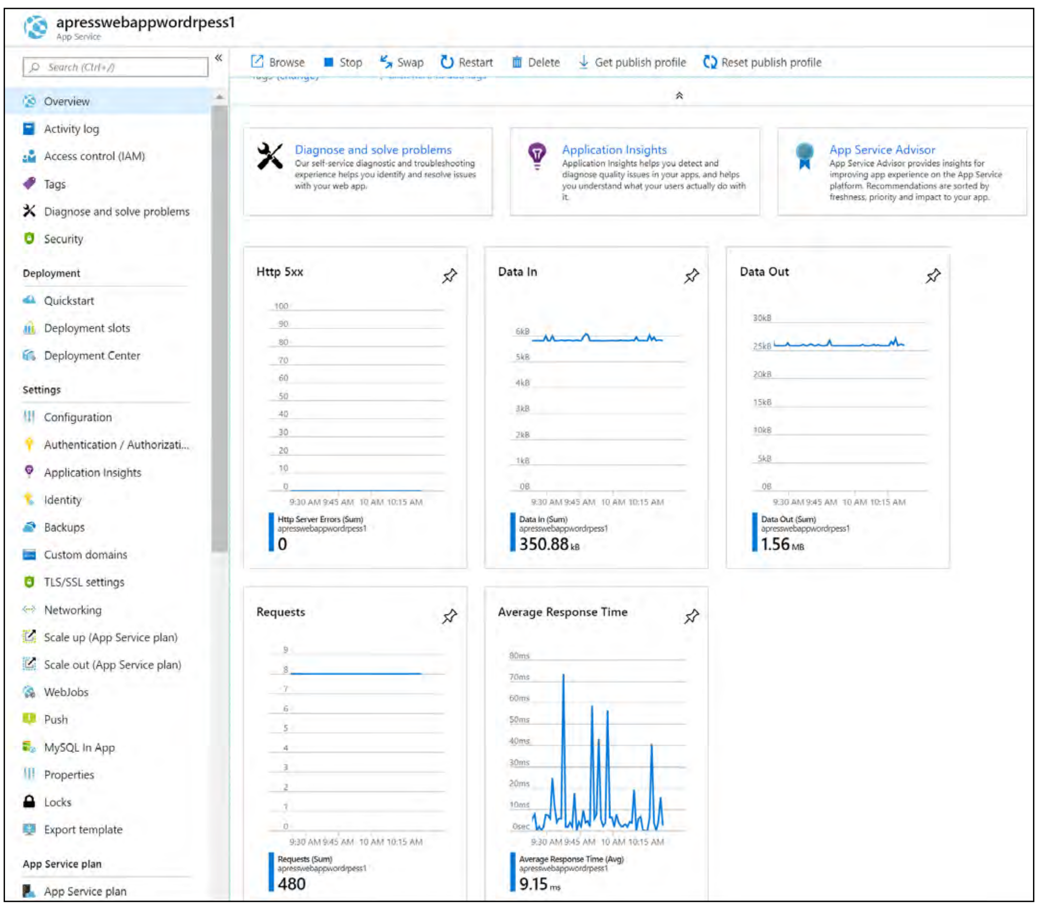

Figura 15: Métricas de recursos de la aplicación web de Azure

La recuperación de datos más complejos de los registros almacenados se realiza mediante Azure Monitor Log Analytics, donde se necesita usar consultas basadas en lenguaje de consulta Kusto. En la *Figura 16*, se muestra un ejemplo de lo que vería en una consulta de este tipo.

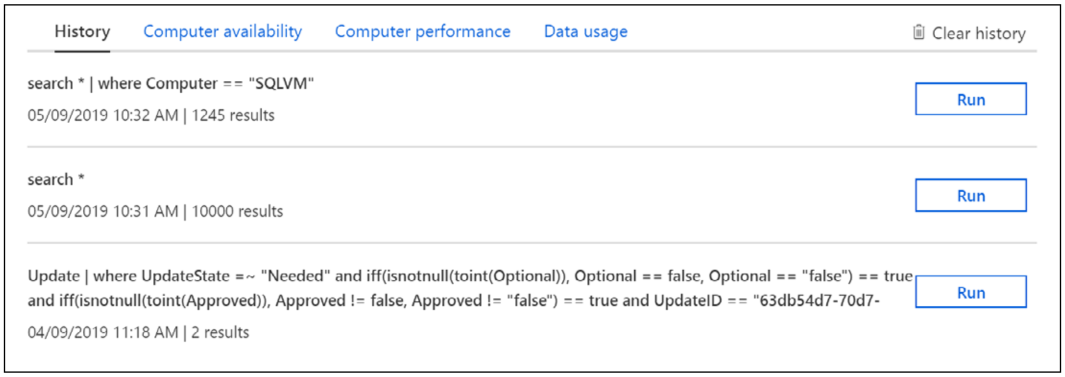

Figura 16: Ejemplo de consulta de Azure Monitor Log Analytics

Consulte `https://bit.ly/2QFzeyJ` para obtener más información y detalles.

Azure Monitor Log Analytics

Azure Monitor Log Analytics se trató anteriormente como un servicio aparte en Azure, conocido como Log Analytics de Operations Management Suite (OMS). Ahora se considera parte de Azure Monitor y se centra en el almacenamiento y el análisis de los datos de registro con su lenguaje de consulta. Las características que se consideraban parte de Log Analytics, como los agentes de Windows y Linux para la recopilación de datos, las vistas para visualizar los datos existentes y las alertas para notificarle de forma proactiva los problemas, no han cambiado, pero ahora se consideran parte de Azure Monitor.

Necesita una consulta de registro para recuperar los datos de Azure Monitor Log Analytics (*Figura 17*). Ya sea que esté analizando datos en el portal, configurando una regla de alerta para recibir una notificación de una condición específica o recuperando datos mediante la API de Log Analytics, usará una consulta para especificar los datos que desea.

Puede crear alertas basadas en sus consultas y, como puede incluir datos de métricas de Azure en Log Analytics, incluso puede realizar consultas en las métricas y los datos almacenados en el servicio de Log Analytics.

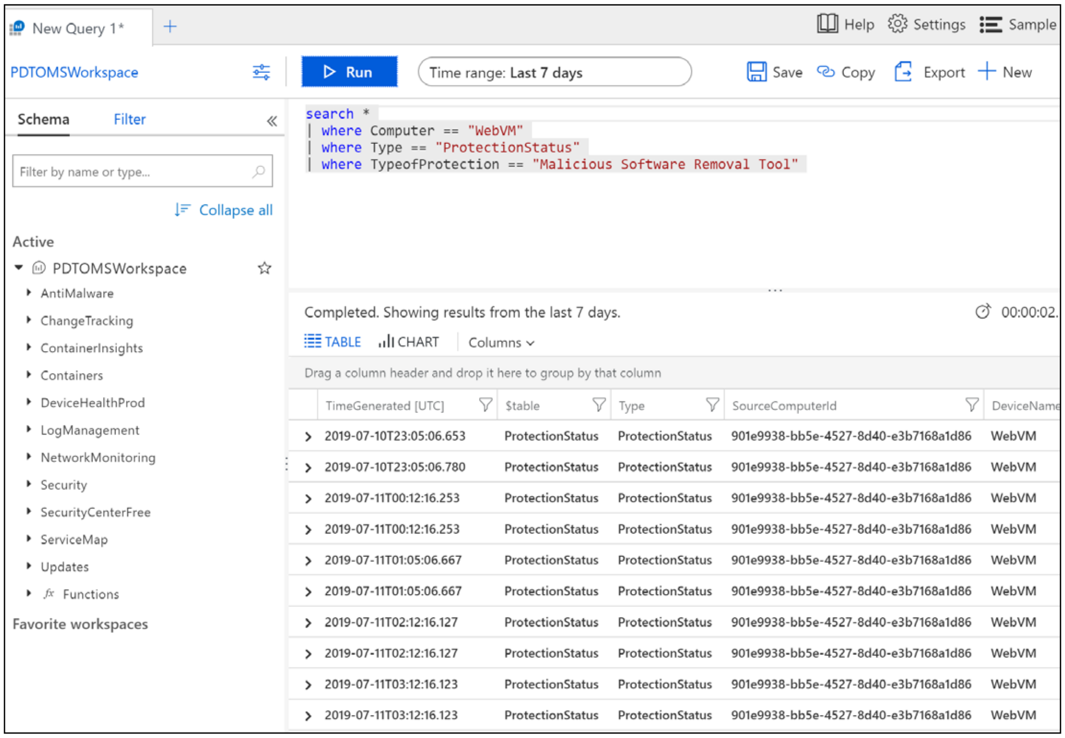

Figura 17: Consulta de Azure Log Analytics

Los orígenes de registro también se ingieren a partir de varios servicios, al igual que las métricas. De hecho, las métricas, junto con los registros de actividad y diagnóstico, pueden servir como origen de datos para Log Analytics. La telemetría de las máquinas virtuales puede ser más rica, puesto que Log Analytics incluye tipos de datos textuales y numéricos, lo que permite que los registros de aplicaciones, los registros de eventos y las métricas de rendimiento adicionales se escriban en Log Analytics. Lo mismo sucede con los datos de aplicación para las aplicaciones personalizadas. Las aplicaciones se pueden instrumentar con Application Insights para proporcionar conocimientos profundos sobre el rendimiento y el estado de la aplicación. Azure Security Center también aprovecha Log Analytics como parte de su análisis de las máquinas virtuales en sus suscripciones. Cuando se incorpora una VM a Security Center, de hecho, se está integrando esa máquina a un área de trabajo de Log Analytics donde Security Center almacena su telemetría para análisis. También hay soluciones de Marketplace que amplían un área de trabajo de Log Analytics, llevan puntos de datos adicionales al servicio para una consulta, así como nuevas visualizaciones basadas en esos puntos de datos.

Por último, puede interactuar con los eventos y responder a ellos en función de los datos de Log Analytics con servicios como Azure Automation, en que puede hacer que un evento active un runbook o, incluso, un webhook en una función de Azure.

Azure Security Center

Azure Security Center (*Figura 18*) proporciona administración de seguridad unificada y protección avanzada contra amenazas en cargas de trabajo de nube híbrida. Muchas organizaciones están migrando cargas de trabajo a la nube o implementando nuevas cargas de trabajo en la nube para optimizar su posición de seguridad.

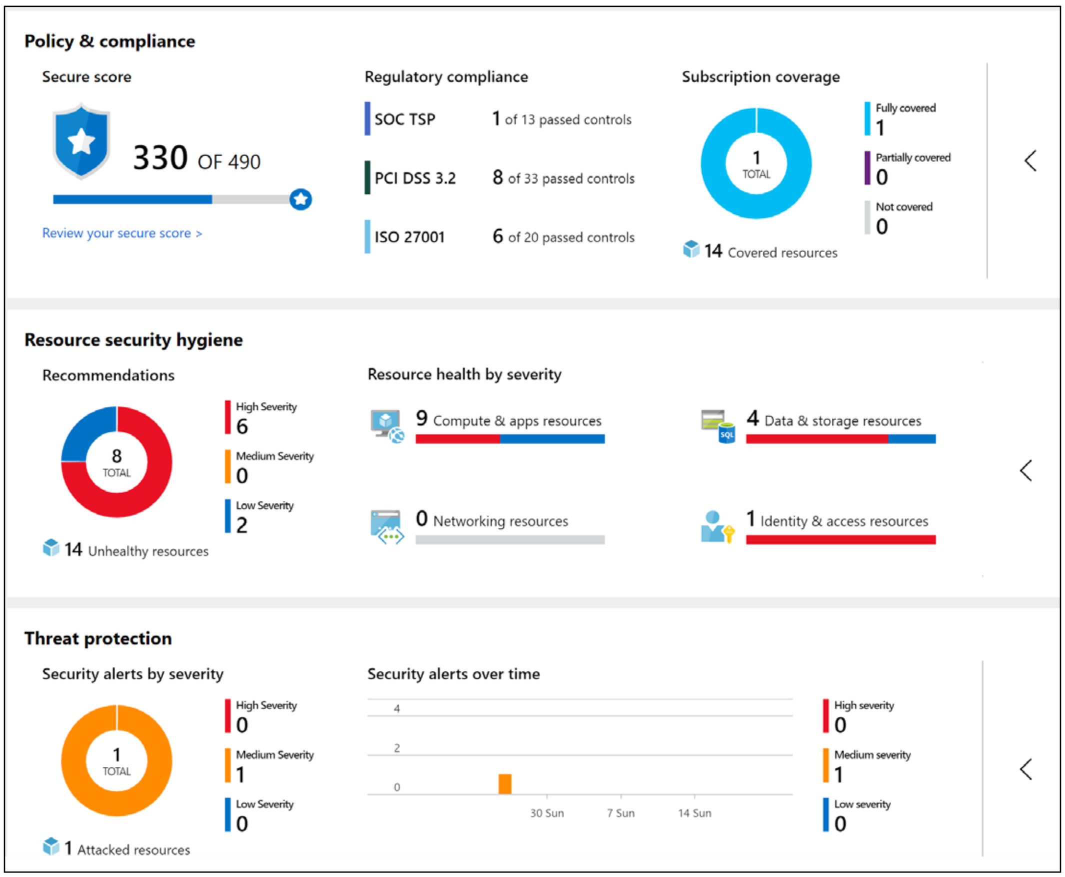

Figura 18: Azure Security Center

Azure Security Center ofrece funcionalidades poderosas en tres áreas principales:

1. **Administración de la posición de seguridad en la nube**: Azure Security Center le proporciona una vista de posición de seguridad vertical en su entorno de Azure, lo que le permite supervisar y mejorar su posición de seguridad con la Puntuación de seguridad de Azure. El Centro de seguridad puede ayudarlo a identificar y llevar a cabo procedimientos recomendados de seguridad y a fortalecer las tareas e implementarlas en sus máquinas, servicios de datos y aplicaciones. Esto incluye administrar y aplicar las directivas de seguridad y asegurarse de que las máquinas virtuales de Azure, los servidores que no son de Azure y los servicios de PaaS de Azure cumplan con las normativas. Con las funcionalidades de IoT recién agregadas, también puede reducir la superficie de ataque para su solución Azure IoT y solucionar los problemas antes de que puedan ser aprovechados. Además de facilitar la visibilidad completa de la posición de seguridad de su entorno, ASC también proporciona visibilidad del estado de cumplimiento de su entorno de Azure en comparación con los estándares normativos comunes.
2. **Protección de cargas de trabajo en la nube**: la protección contra amenazas de Azure Security Center le permite detectar y prevenir amenazas en la capa de IaaS, así como en los recursos de PaaS de Azure, tales como IoT y App Service, y finalmente con máquinas virtuales locales. Las características clave de la protección contra amenazas de Azure Security Center incluyen la supervisión de la configuración, EDR de servidor, control de aplicaciones y segmentación de red. Azure Security Center también admite cargas de trabajo de contenedor y sin servidor.
3. **Seguridad de los datos**: Azure Security Center incluye funcionalidades que identifican infracciones y actividades anómalas contra sus bases de datos de SQL, instancias de almacenamiento de datos y cuentas de almacenamiento, con soporte para otros servicios en el camino. Además, Security Center le ayuda a realizar la clasificación automática de sus datos en Azure SQL Database.

La seguridad forma parte de todas las capas del entorno de la nube pública. La buena noticia es que Azure Security Center tiene las funcionalidades para abordar e informar sobre cada una de estas capas en un entorno de Azure, así como en un escenario de nube híbrida.

Azure Sentinel

Aunque Azure Security Center se utiliza principalmente como una herramienta de operaciones reactivas, muchas organizaciones tienen dificultades para cambiar a un enfoque reactivo. Aquí es donde **Azure Sentinel** puede ser valioso. Azure Sentinel es una herramienta de **Información de seguridad y administración de eventos** (**SIEM**) en la nube. Si consideramos las enormes cantidades de datos relacionados con la seguridad, es importante tener una visión clara y centrarse en las preocupaciones clave y cómo solucionarlas. Muchos de los distintos servicios de Microsoft, como Azure, Office 365 y también servicios que no son de Microsoft, pueden notificar la información a Sentinel.

Toda esta información pasa a través de un motor de "machine learning" y análisis de datos masivo, que ayuda a identificar las amenazas. Además, cuenta con más de 100 reglas integradas y se pueden configurar reglas de alerta propias.

Cuando se producen uno o más incidentes (casos), se centralizarán para permitir una mayor investigación y control. Con una vista gráfica eficaz, resulta más fácil presentar y detectar la correlación entre ataques y vulnerabilidades independientes. Y todo esto está disponible en varias plataformas, ya sea comprobando los recursos de Azure, locales o híbridos.

Además de las vistas y detecciones de paneles, Azure Sentinel también puede ayudar en la corrección. Con los Cuadernos de estrategias de seguridad, respaldados por Azure Logic Apps, crear un enfoque lógico paso a paso de las medidas solo toma unos cuantos clics.

Por último, una enorme comunidad de expertos y organizaciones de seguridad de Microsoft y externas ayudan a mantener y optimizar las detecciones de seguridad, centralizadas en el repositorio de GitHub de Sentinel, `https://bit.ly/302Yfqq`, lo que genera un servicio de seguridad más eficaz.

Azure Network Watcher

Una actividad básica como parte de la administración de su propio centro de datos es obtener una visión clara de su tráfico de red, no solo desde una perspectiva de seguridad, sino también en relación con el ancho de banda, la latencia y otros aspectos similares. Las herramientas de proveedores, como Wireshark y Fiddler, han sido parte de la caja de herramientas de solución de problemas de todo administrador de TI durante años. Si bien estas herramientas siguen siendo útiles en un entorno de Azure, no siempre proporcionan la misma experiencia enriquecida y detallada. El motivo principal de esto es que usted no posee ni controla la pila de red de Azure como lo hace en su centro de datos local. No hay firewalls ni interruptores a los que pueda conectar un cable de consola serie para capturar el tráfico de red o leer la información de syslog.

Y ahí es donde entra en acción Azure Network Watcher (la *Figura 19* ofrece una vista de panel de la integración de Network Watcher con Azure Monitor). Debido a la incorporación de muchas características representativas de seguimiento de red, es una herramienta imprescindible si se toma en serio la supervisión de red de Azure:

- **Comprobación de flujo de IP**: le permite obtener una visión clara sobre el asunto si se permite o deniega un paquete a una máquina virtual o desde ella, mediante la utilización de información de cinco tuplas
- **Próximo salto**: proporciona el próximo salto desde la máquina virtual de destino hasta la dirección IP de destino
- **Reglas de seguridad eficaces**: muestra las reglas eficaces, basadas en diferentes reglas de Grupo de seguridad de red (NSG), configuradas en diferentes niveles (NIC, subred, etc.), lo que facilita la solución de problemas y detecta el verdadero motivo por el que se permite o deniega el tráfico

- **Solución de problemas de VPN**: diagnostica el estado de los gateways de VPN de sitio a sitio y ExpressRoute, mediante la captura de detalles en un archivo de registro de diagnóstico
- **Captura de paquetes**: al igual que Wireshark, ejecuta capturas de paquetes para analizar todos los detalles de la pila de red

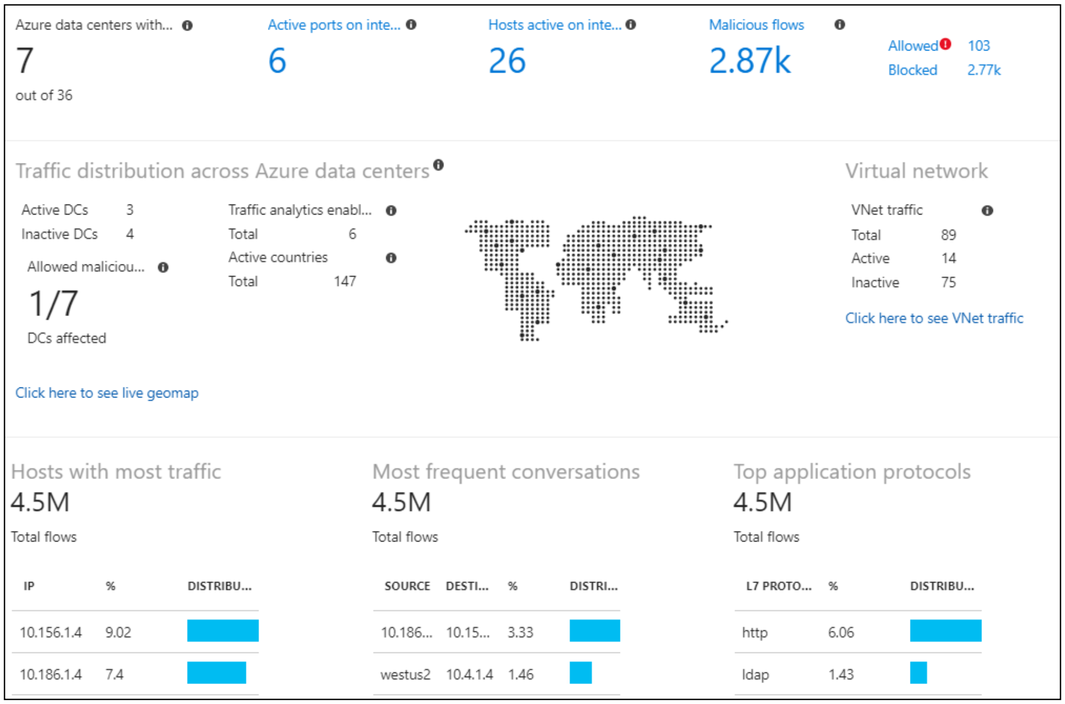

Figura 19: Monitor de conexión de Azure Network Watcher

Azure Service Health

A veces, usted enfrenta un problema o tiempo de inactividad con una de sus cargas de trabajo de Azure en ejecución. Sin embargo, esto no siempre es provocado por los administradores. Como cualquier otro centro de datos, Azure experimenta actualizaciones continuas y requiere revisiones de su infraestructura. Los componentes físicos fallan todo el tiempo, especialmente si se considera la cantidad de servidores físicos, almacenamiento, redes y bastidores que se ejecutan dentro de todas y cada una de las regiones de Azure. Para ayudarlo en la solución de problemas, además de identificarlo legalmente e informarle sobre cualquier problema que enfrente el proveedor de PaaS, Azure le ofrece Azure Service Health (*Figura 20*). Desde aquí, puede obtener una vista en tiempo real del estado general de tiempo de actividad de cualquier centro de datos de Azure en el que tenga servicios en ejecución, además de vistas históricas. Si detecta un problema con una carga de trabajo o un servicio cuya configuración usted no ha cambiado, debería abrir inmediatamente Azure Service Health y validar que todo está bien en la pila física de Azure.

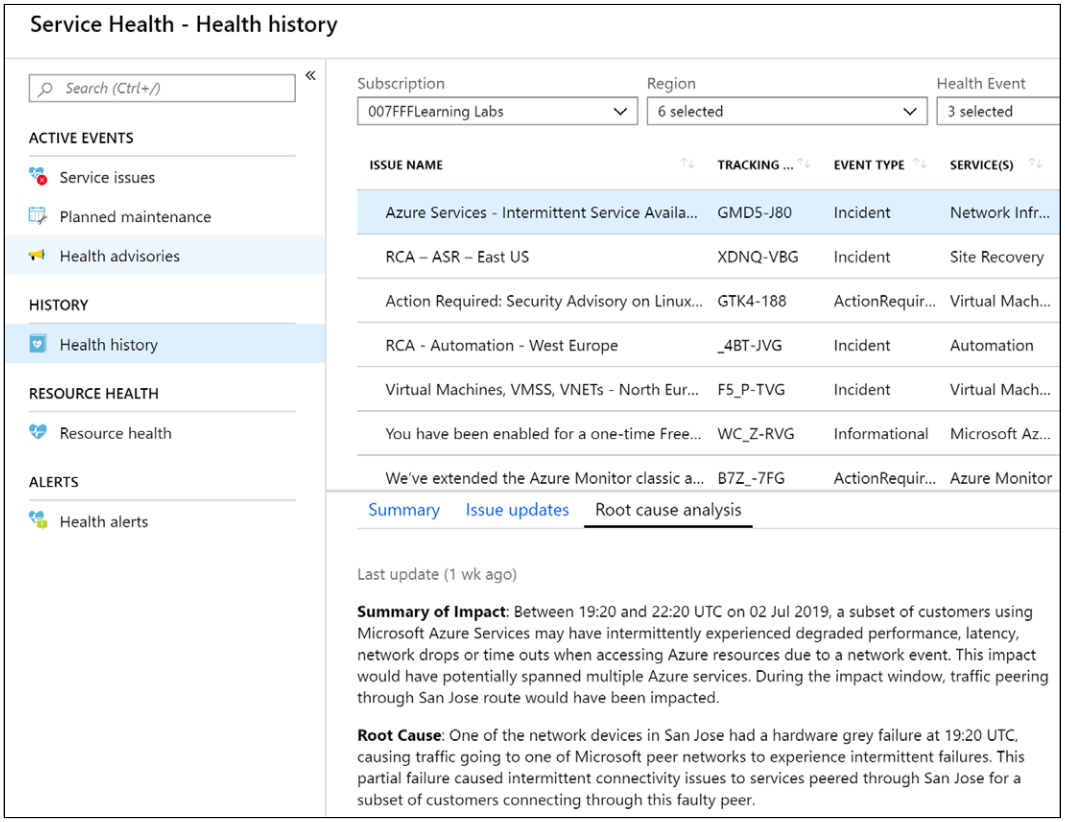

Figura 20: Azure Service Health

Azure Advisor

Azure Advisor es una excelente herramienta de supervisión de Azure. En primer lugar, es gratis, pero esa no es la razón principal por la que cualquier cliente de Azure debería habilitarla y usarla en sus suscripciones.

Azure Advisor proporciona información y recomendaciones en cuatro dominios:

- Alta disponibilidad
- Seguridad
- Rendimiento
- Costo

Todo esto se presenta en un panel de Azure (*Figura 21*), que le permite profundizar para ver más detalles sobre las recomendaciones.

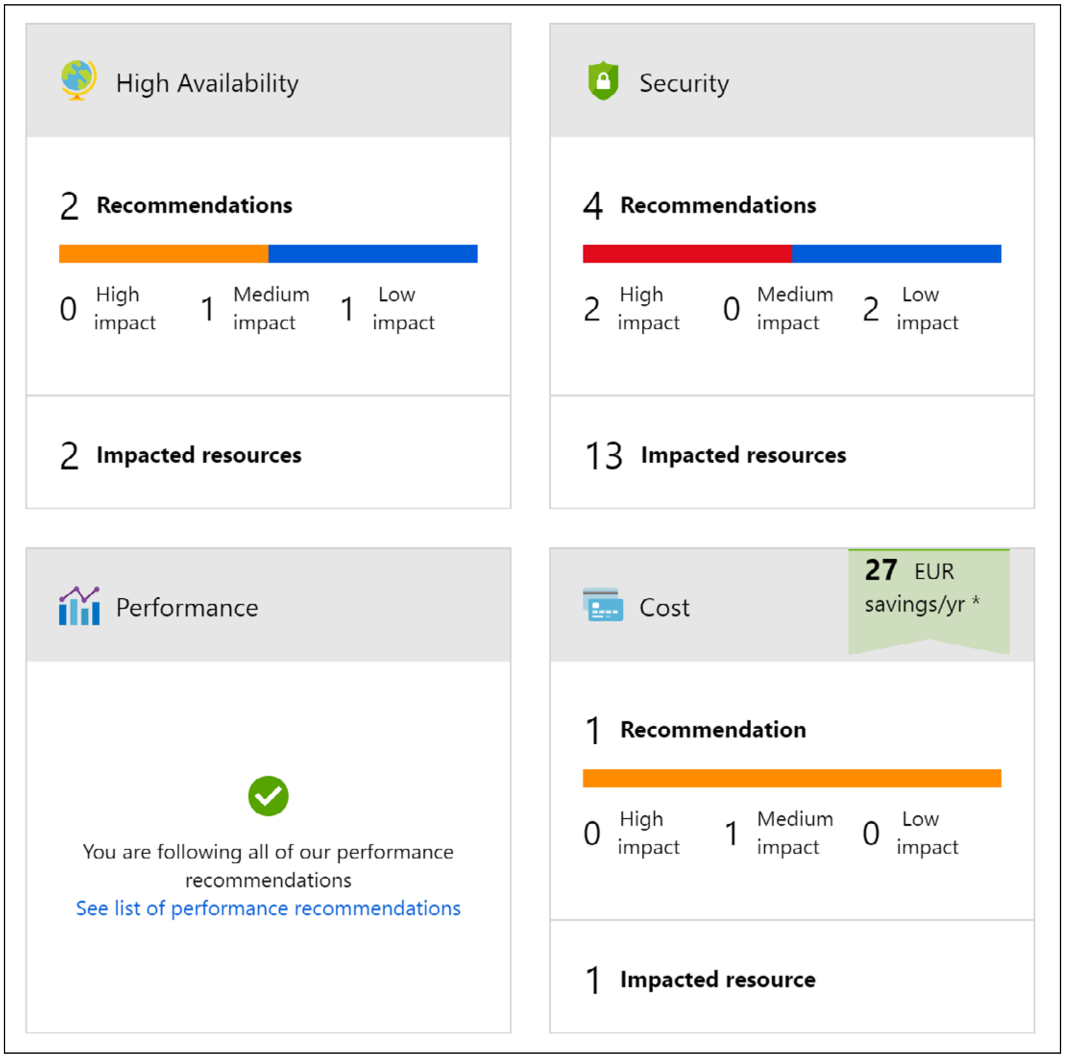

Figura 21: Azure Advisor

La idea principal de Azure Advisor es obtener orientación sobre el uso de los procedimientos recomendados de Azure, lo que le ayuda a optimizar las cargas de trabajo y los servicios en Azure. Se basa en "machine learning" en el back-end y depende de la información de telemetría y la configuración de sus entornos reales de Azure para proporcionar recomendaciones destinadas a la optimización.

Si bien sería posible buscar esa información por su cuenta, es probable que sea mucho más lento y difícil descubrir lo que Azure Advisor presenta bien y casi en tiempo real, sin ningún tipo de molestia o agente que necesite implementarse.

Azure Monitor Application Insights

Azure Monitor Application Insights es otra herramienta de supervisión de Azure, con un enfoque central en la supervisión de sus entornos de aplicaciones (web), sin importar donde se ejecuten. Y esa es inmediatamente una de las principales razones por las que debe darle un vistazo. Aunque se ejecutan en Azure, las aplicaciones web supervisadas por el cliente no necesitan ejecutarse en Azure App Service. Es una herramienta muy detallada, que captura la mayor parte de la información que un desarrollador busca cuando ejecuta aplicaciones web. Detecta información sobre el rendimiento de la aplicación web, pero también ayuda a analizar el tráfico de aplicaciones web. Desde una perspectiva de lenguaje, se admiten los entornos y lenguajes de desarrollo más populares (.NET, Java, Node.js, Python, Ruby, y más).

Se instala un pequeño paquete de instrumentación en su aplicación y se configura un recurso de Application Insights en el portal de Microsoft Azure. La instrumentación supervisa su aplicación y envía datos de telemetría a Azure Monitor. (La aplicación puede ejecutarse en cualquier lugar, no es necesario que se hospede en Azure).

La información que Azure Monitor Application Insights puede rastrear y presentar incluye la siguiente:

- Tiempos de respuesta y solicitudes con errores de la aplicación web
- Excepciones en el tráfico o el uso
- Vistas de página y rendimiento de carga
- Métricas personalizadas, si están configuradas

Le permite trabajar con cuatro dominios de hospedaje de aplicaciones web:

- Supervisión (disponibilidad, rendimiento e integración con otros servicios, como las bases de datos)
- Detección y diagnóstico de problemas (errores HTTP, dar seguimiento, etc.)
- Transacciones sintéticas (ejecución de pruebas web)
- Información de telemetría

Las diferentes secciones supervisadas por Azure Monitor Application Insights se pueden presentar en paneles claros y personalizables (la *Figura 22* muestra un ejemplo), que se pueden compartir entre los administradores de Azure.

Figura 22: Azure App Insights

Resumen del capítulo

En este primer capítulo, describimos los procedimientos recomendados de Microsoft en torno a las migraciones de Azure, desde las evaluaciones hasta las herramientas que proporciona Microsoft para ayudar en esta fase. A continuación, analizamos Azure Migrate y el Azure Migration Center más reciente. Examinamos de qué forma proporciona las herramientas necesarias y útiles para realizar migraciones reales de cargas de trabajo desde entornos locales hasta Azure y abarcamos diferentes arquitecturas. Luego, hablamos de los diferentes aspectos fundamentales de la ejecución de un centro de datos virtual en Azure, destacando la identidad y el control, las redes preparadas para la empresa, las funcionalidades de Azure Storage y las arquitecturas de máquinas virtuales de Azure disponibles hoy en día.

Proporcionamos información sobre la innovación empresarial y la transformación digital de las cargas de trabajo tradicionales con máquinas virtuales, y qué otras capacidades y servicios de Azure están disponibles para hospedar sus cargas de trabajo críticas para el negocio con PaaS, sin servidor y microservicios.

También compartimos información sobre la supervisión y las operaciones de Azure, incluida la forma de administrar eficazmente el entorno de nube mediante las principales herramientas integradas de Azure.

Si bien la mayoría de las organizaciones han utilizado una gran cantidad de herramientas similares para administrar y operar centros de datos locales, no es necesario preocuparse por la complejidad adicional debido a que se están agregando más servicios y herramientas de Azure. En función de la estrategia de nube, una organización puede comenzar por usar los servicios de operación que proporciona Azure para supervisar y operar solo las cargas de trabajo de Azure. Sin embargo, la mayoría de los servicios de supervisión a los que se hace referencia aquí también se extienden a su centro de datos local y, a menudo, también a los escenarios híbridos. Por lo tanto, un enfoque adecuado podría ser aprender cómo integrar estas herramientas de Azure en su entorno general de TI y aprovechar su verdadero poder. No se quede atascado pensando que solo puede usar Azure.

Si desea probar Azure y utilizar algunas de las funciones mencionadas en esta guía, puede registrarse para obtener una cuenta gratuita en el siguiente vínculo: `https://azure.microsoft.com/free/services/virtual-machines/`.

2

Opciones de arquitectura y principios de diseño

A medida que las organizaciones trabajan en aras de modernizar sus aplicaciones, ya sea para ellas o sus clientes, tienen como objetivo manejar sus aplicaciones a fin de lograr escalabilidad, resistencia y alta disponibilidad. La nube y los dispositivos móviles están cambiando la forma en que las organizaciones abordan el diseño de aplicaciones. Vemos que las grandes aplicaciones monolíticas son reemplazadas por servicios más fragmentados, pequeños y descentralizados. Estos servicios proporcionan comunicación a través de API de microservicios, o mensajes o eventos asincrónicos. Este cambio ha creado nuevos obstáculos que las organizaciones deben superar, como el paralelismo, las operaciones asincrónicas y la distribución del estado de la aplicación. También hay consideraciones básicas que se deben tener en cuenta, como el diseño para errores o el escalamiento, a la vez que se adopta la automatización de la administración y la implementación.

En este capítulo se muestra un enfoque para diseñar soluciones en la nube que abarcan una variedad de tecnologías y temas. Examinaremos algunas tecnologías populares de la nube y revisaremos los procedimientos recomendados y algunas consideraciones respecto a los precios, pero primero examinemos algunos aspectos básicos del desarrollo de aplicaciones en la nube antes de profundizar en las tecnologías.

Fundamentos de la aplicación para la nube

Comenzaremos con algunas conclusiones clave sobre las prácticas de desarrollo en la nube en comparación con las prácticas locales. En la nube, hay muchas maneras de resolver un problema, así que esfuércese para simplificarlo, trate de no forjar su camino hacia la complejidad. Tenga en cuenta los aspectos básicos detrás de la arquitectura de la aplicación, que se muestran en la *Figura 1*, y esfuércese por resolver las capas de la forma más sencilla que pueda.

A todo el mundo en su recorrido hacia la nube le ha picado el bicho de la complejidad y ha aprendido esa lección:

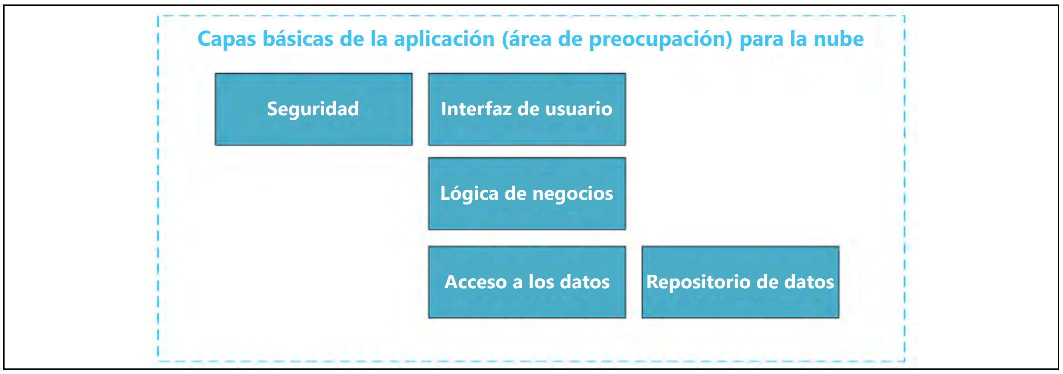

Figura 1: Las capas básicas de la arquitectura de la aplicación

Abordar estas capas y definirlas lo antes posible es una parte e sencial de la arquitectura de la nube y permitirá que cada parte de la organización contribuya a la aplicación en su conjunto, lo que significa que la seguridad, el desarrollo, las operacione s y las pruebas ocuparán desde el principio un lugar importante. Ser un arquitecto de nube exitoso requiere, en gran parte, aprender a facultar a las personas que lo rodean para que tengan éxito en sus roles y sientan que son parte del proceso. Tenga en cuenta que estas capas no necesitan estar en el mismo proyecto y se pueden desglosar y separar, pero, al final del día, las aplicaciones tienen límites marcados. Recuerde no combinar las capas, por ejemplo, colocar la lógica de negocios en su IU o repositorio de datos. Las protecciones de la arquitectura son una parte clave de la arquitectura de nube. Abarcaremos las estructuras del proyecto con más profundidad en el *Capítulo 3*, *Azure DevOps*.

Tener en mente las capas básicas de la aplicación conducirá a una gobernanza general de la nube, que ayudará a ubicar las protecciones y facultar a los arquitectos a fin de que elijan los recursos adecuados para ofrecer las mejores soluciones. Las directivas corporativas de la nube deben incluir definiciones en torno al riesgo empresarial que ay udarán a definir las directivas y el cumplimiento, y crearán el proceso para supervisar la observancia de estas directivas definidas. Las directivas corporativas de la nube deben seguir e stos principios:

- Deben documentar lo que se ha identificado y entendido como el riesgo corporativo.
- Este riesgo debe traducirse en declaraciones de directiva.
- Se debe probar y supervisar la adherencia a estas declaraciones.

A medida que las organizaciones comienzan su recorrido hacia la nube, es importante adquirir un conocimiento general adecuado de los estilos, el diseño, las opciones tecnológicas de la arquitectura y el uso de los cinco pilares de la calidad de software (resistencia, seguridad, disponibilidad, escalabilidad y administración). Como nota al margen, creo que es útil conservar este vínculo a `https://docs.microsoft.com/` para revisar cualquier actualización de arquitectura de Microsoft, con respecto al desarrollo de aplicaciones nativas de la nube. Comprobar su nueva arquitectura en relación con los cinco pilares ayudará a que sus aplicaciones tengan éxito. Junto con una práctica arquitectónica sólida, también es importante comprender las áreas de responsabilidad de la nube. A veces, la solución de problemas en la nube puede perjudicar si no se tiene esta comprensión de las áreas de responsabilidad, porque es posible que no tenga acceso a los registros subyacentes o que los registros se hayan transmitido y sean difíciles de seguir. Esto hace que sea extremadamente importante entender las áreas de responsabilidad y la forma en que los cinco pilares se ven afectados por ellas. Demos un vistazo rápido a estas responsabilidades en la *Figura 2*:

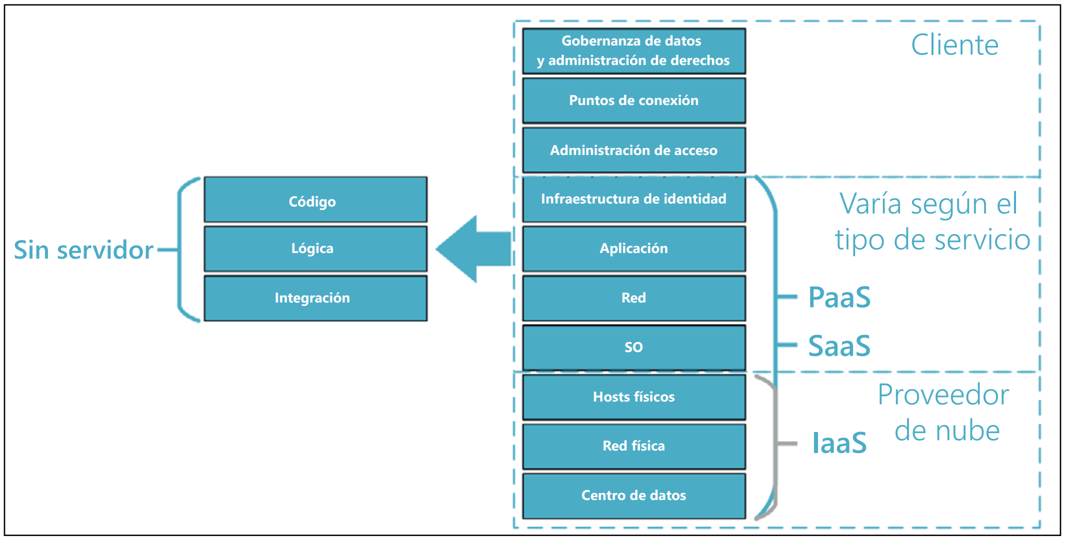

Figura 2: Áreas de responsabilidad de la nube

La Figura 2 muestra una matriz de responsabilidades de la nube. Mientras se encuentra en una configuración local, usted o su organización son responsables de todas las capas; en una configuración de nube, la responsabilidad se comparte con el proveedor de nube. Cuando realiza una migración desde un hardware local o dedicado en un centro de datos, hay algunos límites nuevos respecto a dónde recaen las responsabilidades. Este cambio en las responsabilidades puede generar temor indebido dentro del equipo de operaciones de una organización, pero se debe comprender que estos cambios ayudarán a mejorar su operación al final del día. También crea responsabilidades financieras diferentes, puesto que la autonomía del costo ahora refleja el tiempo de proceso o el espacio de almacenamiento, y es más un costo de la utilidad que un costo irrecuperable o capitalizado.

Siempre debe tratar de tener en cuenta los aspectos básicos de lo que significa migrar de un entorno local a la nube. La *Figura 3* muestra cómo una sencilla aplicación almacenada en una **máquina virtual** (**VM**) se asigna entre el entorno local y Azure:

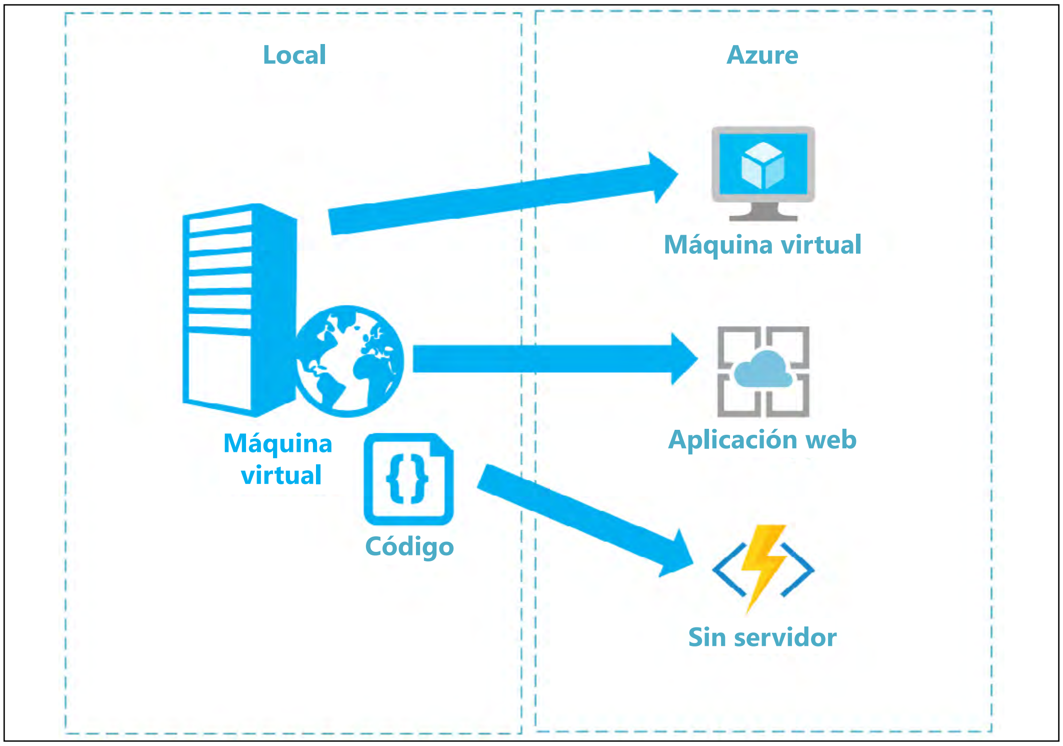

Figura 3: Entorno local en comparación con Azure

Como puede ver, es relativamente fácil entender una asignación de VM a VM: migrar la VM como un todo a la nube. Este modelo, que se conoce como "lift and shift", es el más sencillo, pero también es el más costoso. Esta es la forma más rápida de migrar a la nube sin realizar modificaciones de código. Este modelo se basa en las Redes virtuales de Azure (VNets), los dispositivos de red virtuales y los servicios de dominio.

Puede elegir migrar solo las aplicaciones de IIS a una aplicación web, lo que es un poco más complejo, porque es posible que necesite modificar el código para completar esta migración. Este tipo de migración ayuda a su modelo de costos y sus líneas de responsabilidad, puesto que solo tiene que preocuparse por la aplicación y no por los recursos subyacentes necesarios para esa aplicación. Por ejemplo, las aplicaciones que se alojan en IIS o en el administrador de tareas se pueden migrar a Azure Web Apps o a WebJobs si cambia el punto de conexión de implementación.

Puede ir incluso más allá con un modelo sin servidor. Aquí, puede migrar solo la estructura de clase de código subyacente a un modelo sin servidor con Azure Functions. Esto requiere un poco más de pruebas y modificaciones, pero es el modelo más barato desde una perspectiva de consumo y soporte técnico. Debe considerar la forma en que las diversas partes de su empresa se asignarán desde el entorno local hasta Azure y tomarlo en cuenta cuando elija su arquitectura.

Ahora demos un vistazo a algunos de los desafíos y consideraciones que deberá tener en cuenta a medida que tratemos las diversas opciones de arquitectura en este capítulo.

- Asegúrese de que los componentes estén diseñados para ser coherentes, administrables y reutilizables, sin descuidar los aspectos de administración e implementación. Las decisiones y las protecciones que se pongan en marcha durante la fase de diseño tendrán un gran impacto en el resultado general y en su capacidad para cambiar su diseño cuando sea necesario.
- Las aplicaciones en la nube existen en el dominio público fuera de su organización, por lo que la seguridad se convierte en un principio fundamental para restringir el acceso y proteger los datos. Es conveniente entender cómo delegar la autenticación o federar las identidades a proveedores externos. Por ejemplo, puede usar una cuenta de almacenamiento que utilice una "clave auxiliar", que es un token o clave para restringir el acceso a los recursos importantes. El uso de un gateway o gatekeeper para aislar sus servicios y aplicaciones ayuda a proteger y negociar las solicitudes con los clientes que los consumen.
- Algo importante que necesitará en la nube, como con cualquier modelo compartido, es resistencia: ser capaz de manejar los problemas que se produzcan y recuperarse de ellos. Debe ser capaz de detectar problemas y reaccionar a ellos de forma rápida y eficiente. Colocar "cierres" o aislar los servicios para asegurarse de que las fallas no se propaguen es una táctica esencial y debe utilizarse con un "interruptor" que aísle aún más los servicios. El patrón de "interruptor" ayuda con el aislamiento del servicio hasta que este vuelva a encontrarse en buen estado, al igual que un interruptor de circuito en su hogar. También debe tener en cuenta cómo realizar la reversión de un estado de error en el sistema desconectado. Investigue las "transacciones compensatorias" que le permiten retroceder en un proceso hasta que pueda continuar con una operación coherente. También le recomiendo utilizar la supervisión de comprobación de estado de su punto de conexión y asegurarse de tener un proceso de reintento que funcione con estos tipos de fallas.
- La disponibilidad es un aspecto fundamental en la nube, que le permite propagar cargas de trabajo y limitar el consumo de sus recursos. Normalmente, se mide en tiempo de actividad, al igual que se hace desde una perspectiva local. La mayor diferencia entre la nube y el entorno local es el modelo de recursos a petición que proporciona la nube, donde los recursos se pueden escalar automáticamente hacia arriba y hacia abajo con poca o ninguna intervención.

Esto significa que para escalar una solución local, necesita el espacio virtual o máquinas nuevas para escalar. Esto también incluye escalar toda la máquina y no solo los componentes más pequeños, y requiere realizar un ajuste a la granja de servidores o a los dispositivos de red para agregar máquinas nuevas.

- Otro elemento clave en la nube es cómo manejar la administración de datos, puesto que los datos normalmente se hospedan en varias ubicaciones en varios servidores. Debe asegurarse de que la coherencia de los datos esté bien mantenida y sincronizada en varias ubicaciones. Es esencial asegurarse de tener una estrategia de almacenamiento en caché para el contenido estático o los elementos de datos solicitados con frecuencia, puesto que pueden desempeñar un papel importante en el rendimiento de las aplicaciones.
- A fin de mantener el rendimiento y la capacidad de respuesta, debe tener un proceso para administrar las variaciones o los cambios en el procesamiento de cargas de trabajo y para lidiar con los tiempos de uso máximos y las cargas del sistema. Uno de los grandes beneficios de migrar a la nube es la forma en que este escalado y la asignación de recursos se manejan en gran parte de forma automática. Cuando se diseña una arquitectura de sistema, se deben utilizar lo más posible tipos de recursos que admitan este escalado automático.
- Con toda esta complejidad, la capacidad de administrar los recursos y la supervisión aparecen en el primer plano de su recorrido hacia Azure. Si bien la administración de recursos y la supervisión eran importantes en los modelos locales, estas son fundamentales para la nube, donde cambian los límites de las responsabilidades, tal como se muestra en la *Figura 2*. Estos cambios alejan su enfoque del hardware en el que se ejecuta la aplicación y lo coloca en áreas que son importantes para ella.
- La mensajería en la nube se utiliza para separar la dependencia del servicio de la misma manera en que los buses de servicio se utilizaban en el entorno local, pero esta estructura asincrónica desconectada tiene sus propios desafíos, como el pedido, la administración de mensajes y la idempotencia.
- Una de las piezas más importantes para que todo funcione bien cuando se implementa una estrategia de nube es implementar una convención de nomenclatura apropiada. Esto es importante para solucionar problemas y crear sistemas que las personas puedan entender. También es difícil y requiere mucho tiempo hacer cambios en los sistemas de nomenclatura de archivos más adelante, por lo que vale la pena hacerlo bien desde el principio. Puede encontrar recomendaciones de procedimientos de Microsoft en `https://bit.ly/37Kj0tI`.

También debemos reconocer que no todo el mundo quiere o necesita migrar toda su carga de trabajo de la organización a la nube, por lo que, en este capítulo, también indicaremos modelos híbridos que mezclan soluciones en la nube y locales. En la *Figura 4*, se muestra un ejemplo rápido de un escenario híbrido:

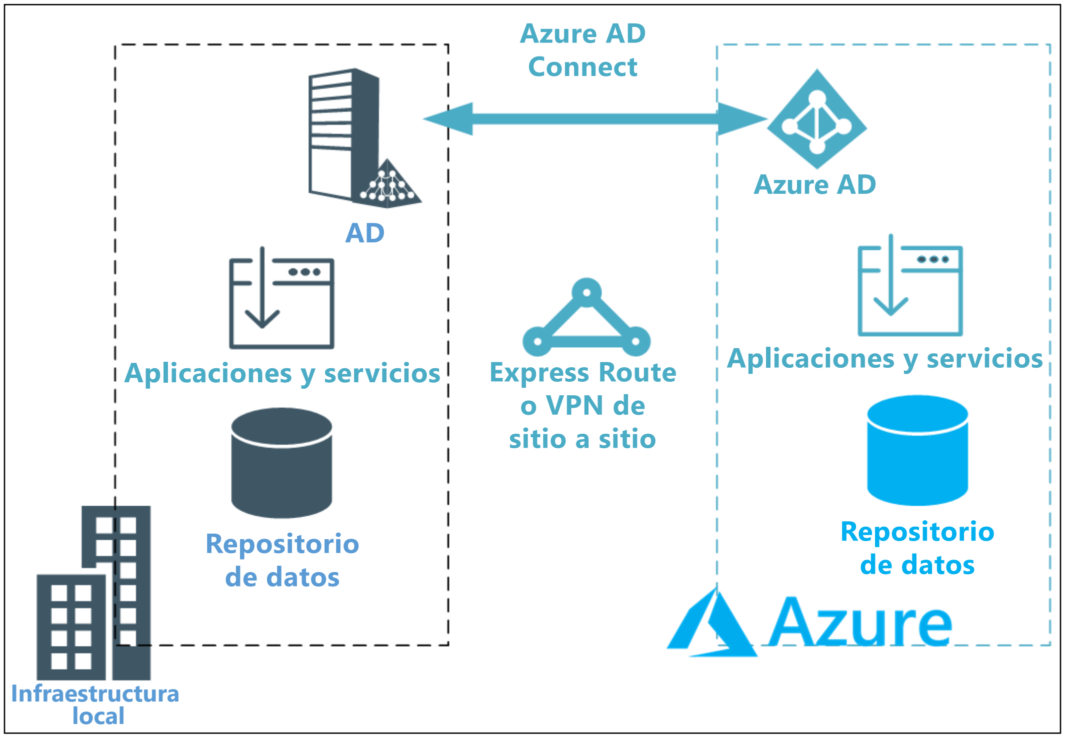

Figura 4: Modelo híbrido sencillo de nube

Como puede ver, la conexión puede realizarse a través de una VPN ExpressRoute o de sitio a sitio sencilla para conectar la red local a Azure, y Azure AD Connect se utiliza para sincronizar la red AD con Azure AD, de modo que la seguridad y el **inicio de sesión único** (**SSO**) se conserven desde el entorno local hasta la nube. Este sería el mismo enfoque básico para formar una configuración de nube a nube o local a nube. Existirán diferencias entre las tecnologías de nube a nube que utiliza, y estas tendrán que resolverse mediante VM o contenedores para garantizar el cumplimiento del código y la compatibilidad. Si bien la mayoría de los proveedores de nube comparten parte de la misma funcionalidad, Azure se destaca en la provisión de muchos servicios nativos diferentes para ayudar a resolver problemas. Sin embargo, los contenedores proporcionan una mejor experiencia multiplataforma sin necesidad de rediseñar su base de código. Trataremos esto un poco más adelante en este capítulo.

Otro aspecto clave que se debe señalar antes de profundizar más en la arquitectura es que DevOps tiene una gran sinergia con su recorrido hacia la nube. Esta sinergia se centra en la implementación coherente y la administración de recursos de Desired State Configuration (DSC). También tiene sinergia con sus directivas de nube para definir el riesgo y tener un proceso para administrar este riesgo. Esto se abordará con más detalle en el *Capítulo 3*, *Azure DevOps*, pero debería considerarse en la etapa de planificación.

Las arquitecturas clave de la aplicación

Sin más preámbulos, abordemos algunos de los principales enfoques arquitectónicos para diferentes ecosistemas de aplicaciones. Para cada una de estas arquitecturas, también analizaremos los cinco principios que debe admitir en la gobernanza de su nube:

- La administración de costos de la solución
- Cómo definir una línea base de seguridad
- Cómo definir la coherencia de los recursos
- Cómo definir una línea base de identidad
- Cómo acelerar la implementación de la solución

También puede encontrar una gran cantidad de información en las páginas de documentación de Azure de Microsoft: `https://bit.ly/35Dnzo7`.

Diseño de un ecosistema de microservicios

Los microservicios se han convertido en un estilo arquitectónico popular en los ecosistemas de aplicaciones, puesto que proporcionan un modelo de implementación dividido, muy escalable, resistente y sencillo, que puede evolucionar rápidamente cuando sea necesario. El mundo actual de bloques funcionales más pequeños, en lugar de niveles de aplicación monolíticos grandes, ha encontrado un aliado en la nube. Esta relación se basa en la capacidad de un recurso de ser independiente y escalar por sí solo. La agilidad para implementar y realizar pruebas rápidamente es otra gran ventaja que se proporciona mediante la migración a la nube. El mayor problema es que estas piezas autónomas más pequeñas pueden crear una pesadilla operativa para las organizaciones. Teniendo en cuenta eso, demos un vistazo a cómo abordamos estos tipos de ecosistemas en la nube.

Una de las ventajas de la estrategia de microservicios es que permite que los servicios se desarrollen e implementen de forma independiente, lo que significa que los errores se pueden aislar en un servicio en lugar de en la aplicación en su conjunto. Estos servicios no se limitan a una pila de tecnología y ofrecen la posibilidad de escalarlos de forma separada de la aplicación. Sin embargo, esto crea una cantidad significativa de complejidad para el desarrollo y las pruebas, a la vez que se carece de una gobernanza general y una latencia de servicio. Con estos servicios que dependen otros servicios dentro del ecosistema, la administración y el control de versiones con control de cambios pueden ser engorrosos, incluso para las organizaciones más experimentadas.

En la Figura 5, se muestra cómo podría funcionar un ecosistema como este:

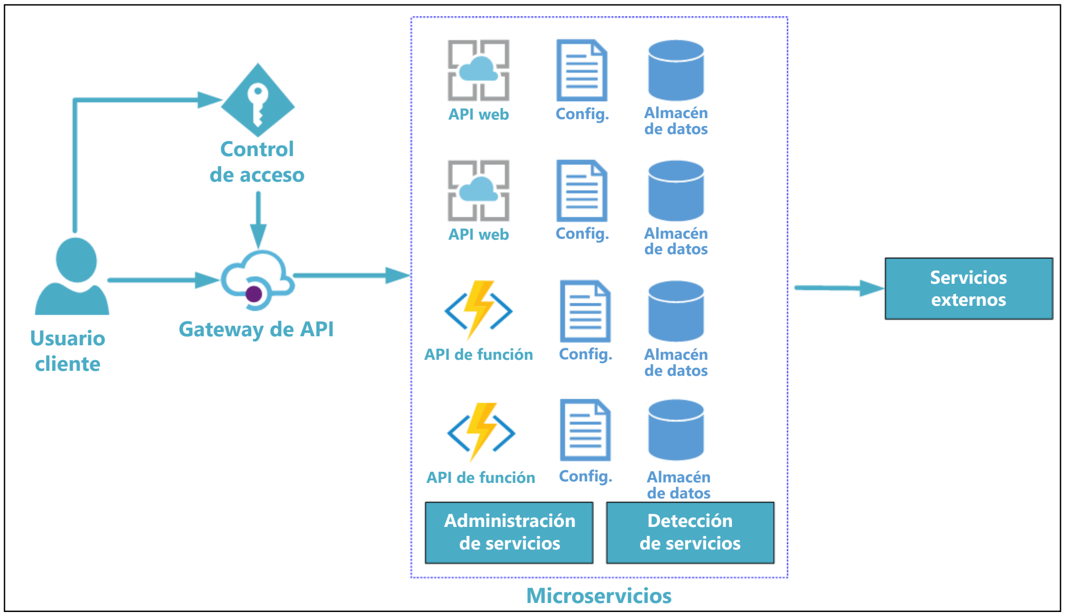

Figura 5: Ejemplo de arquitectura de microservicio

Como puede ver, las características de una arquitectura de microservicio son las siguientes:

- Cada microservicio contiene código, configuración y un repositorio de datos en el ámbito del servicio, lo que proporciona un espacio más pequeño.
- Cada componente de este modelo de aplicación jerárquica se puede escalar, versionar y actualizar de forma independiente.
- Estos servicios se pueden escribir con diferente código y pueden ser administrados por diferentes equipos.
- Los servicios se dividen en función de las capacidades empresariales, lo cual los hace más pequeños y tener responsabilidad única.

Los servicios de administración y detección son una parte importante de la arquitectura; los servicios de administración están a cargo del equilibrio y la mitigación de errores, mientras que los servicios de detección ayudan con la detectabilidad. Usamos la API o el servicio de Front Door para aislar a los clientes de los microservicios, lo que permite un mejor control de versiones, terminación de SSL, registro, autenticación centralizada y equilibrio de carga.

Antes de analizar los beneficios y los procedimientos recomendados, examinemos los cinco principios que mencionamos al comienzo de esta sección:

- La administración de costos en torno a los microservicios dependerá de la dirección que elija para hospedar sus servicios. Si usa un modelo sin servidor, tendrá el mayor potencial de ahorro de costos debido a su precio de consumo. Si usa una aplicación web como estrategia de implementación o requiere una función respaldada por un plan de servicios de aplicación, su estructura de precios será razonable y más predecible, pero los costos aumentarán. Si decide usar Docker o un enfoque de contenedor, existirá un aumento en los precios de consumo. Puede elegir usar Azure Container Instances (ACI), Azure Kubernetes Service (AKS) o Service Fabric, que también pueden requerir algunos componentes de administración adicionales y configuraciones de seguridad que debe planificar.
- La línea de base de seguridad radica en la seguridad necesaria para acceder al recurso que eligió anteriormente, además del acceso al punto de conexión expuesto para el microservicio, es decir, HTTP/HTTPS. Siempre recomendamos que, si su servicio maneja *cualquier* información confidencial, use HTTPS para proteger el punto de conexión.
- La coherencia del recurso se basa en la estrategia que seleccionó anteriormente. No debe elegir varias formas de implementar y administrar su recurso, así que seleccione un enfoque predecible que pueda proporcionar protecciones para su organización, que cubran todos los riesgos de desarrollo, implementación y administración que se hayan identificado.
- La línea de base de la identidad será la forma en que alguien obtendrá acceso al punto de conexión del microservicio, por ejemplo, con la administración de API para controlar y administrar el acceso. Puede usar claves compartidas o autenticación de AD para administrar el acceso al punto de conexión del microservicio. Independientemente de lo que elija, debe esforzarse por conservar este enfoque para la estrategia en su conjunto. Es justo señalar que, si bien las claves compartidas son fáciles de usar, no se puede hacer un seguimiento individual correcto de ellas y, cuando se comprometen, las claves en cuestión deben rotarse y todas las aplicaciones o servicios que las utilizan también. Por este motivo, siempre recomendaría usar Azure AD, en lugar de las claves compartidas, y debido al seguimiento adicional que puede hacer a nivel individual para los usuarios con comportamiento erróneo.
- La aceleración de la implementación debe cumplir con un flujo de Azure DevOps y permitir una actualización fácil. Aprovechar una solución con ranuras como funciones o Web Apps, o servicios de actualización administrados, como AKS o Service Fabric, es extremadamente útil para minimizar el tiempo de inactividad del servicio, si no lo elimina.

Hay bastantes beneficios para elegir una arquitectura de microservicios, como puede ver a continuación. Sin embargo, esto puede complicar su modelo de soporte si no planifica su orientación.

Podemos resumir los beneficios de una arquitectura de microservicios de la siguiente manera:

- Servicios pequeños e independientes
- Acoplados libremente
- Base de código independiente
- Implementaciones más pequeñas
- Límites de dependencia interna

Algunos procedimientos recomendados sencillos cuando se opera según esta arquitectura incluyen los siguientes:

- Descentralizar todo
- Utilizar la mejor tecnología para cada servicio
- Definir bien la comunicación de la API
- No acoplar servicios
- Minimizar las preocupaciones transversales
- Aislar los errores del servicio
- Evitar la locuacidad de los servicios

Si bien la arquitectura de microservicios se diseñó en torno a las capacidades empresariales, uno de los mayores desafíos que debe superar es definir los límites del servicio. La regla general es una acción empresarial por microservicio independiente, lo que requerirá una gran cantidad de planificación. Tendrá que precisar sus requisitos, su dominio empresarial y sus objetivos generales con el fin de producir el diseño adecuado. Aprovechar algo como el diseño impulsado por el dominio puede ayudar con el diseño estratégico y táctico, con el diseño estratégico que proporciona la estructura del sistema y el diseño táctico que proporciona las entidades, agregados y servicios de dominio del modelo de dominio.

Si desea leer más sobre este tema, consulte la documentación en línea: `https://bit.ly/37WqZDR`.

Diseño de un entorno basado en eventos

La arquitectura basada en eventos no es un concepto nuevo en Azure, y ha existido durante muchos años en el entorno local dentro de aplicaciones basadas en servidor, como BizTalk, donde se le conoce comúnmente como "desarrollo centrado en el mensaje". Como puede ver en la *Figura 6*, el evento tiene una construcción sencilla. Consiste en un productor, una ingesta o bombeo, y un consumidor, que básicamente se traduce a esto: se genera un evento, hay un consumidor que escucha ese evento y hay algo de pegamento en el medio.

Piense en esto como un teléfono: la persona que llama es el productor, la respuesta es el consumidor y el teléfono es el agente (ingesta). Es importante saber que el productor y el consumidor están separados en este modelo sencillo:

Figura 6: Plantilla de la arquitectura basada en eventos

Existen dos tipos de arquitectura basada en eventos:

- **Pub/sub**: un evento se publica y se envía a un suscriptor, tipo agente, por lo que el evento no se vuelve a reproducir.
- **Transmisión de eventos**: se trata de un bombeo de mensajes donde los eventos se escriben en un registro y están estrictamente ordenados y son duraderos. El cliente puede leer cualquier parte de la transmisión y es responsable de administrar su posición en ella, lo que significa que los clientes pueden unirse en cualquier momento y los eventos pueden volver a reproducirse.

Los siguientes son casos de uso adecuados para la arquitectura basada en eventos:

- Cuando varios servicios o subsistemas utilizan los mismos eventos
- Si se requiere un procesamiento casi en tiempo real
- Si hay una necesidad de procesamiento complejo, como la agregación o la coincidencia de patrones
- Alta velocidad de datos

Antes de analizar los beneficios y los procedimientos recomendados, examinemos los cinco principios que mencionamos al comienzo de este capítulo:

- La administración de costos de una arquitectura de eventos se basa en Event Grid y en la comprensión del usuario de la conexión de las secuencias de datos que se crean.
- La línea de base de seguridad depende de los requisitos del publicador del evento, que, por lo general, no son extremadamente altos.
- La coherencia de los recursos es fácil, puesto que no hay demasiadas opciones, además de Event Grid, Service Bus, IoT Hub y Event Hub. Estos recursos reciben un mensaje y es responsabilidad del consumidor recibir y manejar el mensaje.
- La línea de base de la identidad no es demasiado aplicable.
- La aceleración de la implementación debe seguir un flujo de Azure DevOps y permitir una actualización fácil.

Demos un vistazo a algunos beneficios de usar una arquitectura basada en eventos:

- Sistemas altamente escalables y distribuidos
- Separación de los productores y los consumidores
- Los consumidores tienen un procesamiento independiente de eventos

Estos son algunos de los desafíos que podría enfrentar con este tipo de arquitectura:

- Procesamiento de pedidos, especialmente en varias instancias de consumidor, a medida que se transmiten los eventos y es responsabilidad del consumidor administrar su lugar en la transmisión.
- La entrega garantizada no forma parte de esta estructura arquitectónica porque el productor no se preocupa por la capacidad del consumidor secundario de recibir o procesar los eventos.

Puede encontrar más documentación para la arquitectura basada en eventos en `https://bit.ly/39UR0Fr`.

Diseño de un ecosistema sin servidor

El término "sin servidor" puede ser un poco engañoso, puesto que podría pensar que estas piezas de código se ejecutan mágicamente sin un servidor. Al contrario de lo que indica su nombre, sin servidor no significa que estas sean piezas mágicas de código, sino más bien que los recursos requeridos se han abstraído de su vista, lo que significa que solo tiene que preocuparse por el código, la lógica y la integración de su aplicación, y no por la infraestructura que lo ejecuta, como puede ver en la *Figura 7*:

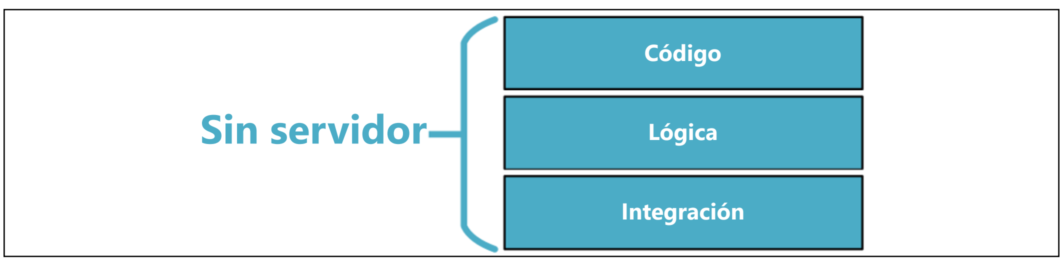

Figura 7: Plantilla de la arquitectura sin servidor

Básicamente, la estructura sin servidor le permite compilar aplicaciones en la nube con el mínimo de código, sin la necesidad de crear, hospedar, supervisar o mantener recursos o infraestructura complicados.

Actualmente, hay tres tipos de recursos sin servidor en Azure:

- Functions, que son pequeños fragmentos de código que se ejecutan en respuesta a una variedad de eventos
- Logic Apps, que son pasos de proceso para ejecutar el proceso (a través de flujos de trabajo)
- Azure Event Grid, que se utiliza para ejecutar su infraestructura sin servidor, puesto que ayuda a detener el sondeo de recursos con una arquitectura basada en eventos

Los siguientes son casos de uso adecuados para una arquitectura sin servidor:

- Cuando desea utilizar un enfoque de microservicio para entregar su capa de servicio. Esto seguiría la arquitectura de microservicios que se analizó antes
- Necesidad de ejecutar pequeños procesos basados en el tiempo
- Cuando se requiere una escalabilidad flexible, por lo general, si se usa un modelo de consumo o un plan de App Service
- Cuando necesita una facturación basada en la utilidad; pague por lo que usa

Examinemos la arquitectura sin servidor con respecto a los cinco principios:

- La administración de costos de la estructura sin servidor es única, puesto que admite un modelo de consumo de precios de utilidad y precios más predecibles con el respaldo del plan de App Service para las funciones. La mayoría de las soluciones sin servidor utilizan precios de ejecución para consumo, como ejecuciones de 1 millón por un precio pequeño para Logic Apps y Azure Functions. Para Logic Apps, también debe incluir el costo del conector en función de los conectores que utilice.
- La línea de base de seguridad depende de la seguridad necesaria para acceder al recurso que eligió anteriormente, además del acceso al punto de conexión expuesto para el microservicio, es decir, HTTP/HTTPS. Recomendación: si su servicio maneja CUALQUIER información confidencial, use HTTPS para proteger el punto de conexión.
- La coherencia del recurso se basa en la estrategia que seleccionó anteriormente. No debe elegir varias formas de implementar y administrar su recurso, así que seleccione un enfoque predecible que pueda proporcionar protecciones para su organización, que cubran todos los riesgos de desarrollo, implementación y administración que se hayan identificado.
- La línea de base de la identidad será la forma en que alguien obtendrá acceso al punto de conexión del microservicio, por ejemplo, con la administración de API para controlar y administrar el acceso. Puede usar claves compartidas o autenticación de AD para administrar el acceso al punto de conexión del microservicio. Independientemente de lo que elija, debe esforzarse por conservar este enfoque para la estrategia en su conjunto.

- La aceleración de la implementación debe seguir un flujo de Azure DevOps y permitir una actualización fácil. Aprovechar una solución con ranuras como funciones o Web Apps, o servicios de actualización administrados, como AKS o Service Fabric, es extremadamente útil para minimizar el tiempo de inactividad del servicio, si no lo elimina.

Ahora demos un vistazo a algunos de los beneficios y desafíos que se presentan con la estructura sin servidor. Uno de los mayores desafíos que enfrentará es que las soluciones sin servidor suelen ser específicas del proveedor y pueden requerir la reescritura del código antes de migrarlas entre el entorno local u otras nubes.

Los beneficios de usar la arquitectura sin servidor incluyen:

- Funciones lógicas independientes basadas en la nube
- Sin estado por diseño
- Desencadenado por evento
- Totalmente administrado por el proveedor de nube
- Sin administración del sistema

Los desafíos que enfrentará con la arquitectura sin servidor incluyen:

- Reducción del control general aparte del código específico necesario para ejecutar las aplicaciones sin servidor.
- Es específica del proveedor tanto en la estructura de código como en los recursos subyacentes.
- El costo puede ser impredecible puesto que paga por lo que usa.
- Debe tener disciplina para ayudar a combatir la expansión del servicio o del código.
- Tienen un límite de ejecución y deben diseñarse para procesos pequeños y de ejecución rápida.

Con la introducción de la arquitectura sin servidor, también se introdujo una estructura de facturación basada en la utilidad o microfacturación. Esta fue una desviación única del modelo actual basado en suscripción de Azure y se creó en torno al consumo.

Para Azure Functions, hay tres tipos de facturación:

- Según el consumo o de pago por uso
- Plan Premium
- Plan de App Service

Para Logic Apps, hay dos tipos de facturación:

- Acciones
- Conexiones

Para Event Grid, hay un tipo de facturación:

- Por número de eventos procesados

A continuación, indicamos cómo la arquitectura sin servidor se relaciona con nuestros cinco principios:

- La administración de costos para la arquitectura sin servidor es única, puesto que admite un modelo de consumo de precios de utilidad. Casi todas las arquitecturas sin servidor utilizan precios de ejecución para consumo, como ejecuciones de 1 millón por un precio pequeño para Logic Apps y Azure Functions. Para Logic Apps, también debe incluir el costo del conector en función de los conectores que utilice.
- La línea de base de seguridad radica en la seguridad necesaria para acceder al recurso que eligió anteriormente, además del acceso al punto de conexión expuesto para el microservicio, es decir, HTTP/HTTPS. Recomendación: si su servicio maneja CUALQUIER información confidencial, use HTTPS para proteger el punto de conexión.
- La coherencia del recurso se basa en la estrategia que seleccionó anteriormente. No debe elegir varias formas de implementar y administrar su recurso, así que seleccione un enfoque predecible que pueda proporcionar protecciones para su organización, que cubran todos los riesgos de desarrollo, implementación y administración que se hayan identificado.
- La línea de base de la identidad será la forma en que alguien obtendrá acceso al punto de conexión del microservicio, por ejemplo, con la administración de API para controlar y administrar el acceso. Puede usar claves compartidas o autenticación de AD para administrar el acceso al punto de conexión del microservicio. Independientemente de lo que elija, debe esforzarse en conservar este enfoque para la estrategia en su conjunto.
- La aceleración de la implementación debe seguir un flujo de Azure DevOps y permitir una actualización fácil. Aprovechar una solución con ranuras como funciones o Web Apps, o servicios de actualización administrados, como AKS o Service Fabric, es extremadamente útil para minimizar el tiempo de inactividad del servicio, si no lo elimina.

Dediquemos un momento a revisar algunas de las características, procedimientos recomendados y consideraciones de los tres recursos sin servidor de Azure.

Funciones sin servidor Azure

Aquí se indican algunos procedimientos recomendados para usar Azure Functions:

- Evite las funciones grandes de larga duración: refactorice las funciones grandes en conjuntos de funciones más pequeños que trabajen en conjunto y devuelvan respuestas rápidas, a menos que se usen funciones duraderas.
- Comunicación entre funciones: cuando se integran varias funciones, generalmente es mejor usar colas de almacenamiento para la comunicación entre funciones.

- Escribir funciones sin estado: debe asociar cualquier información de estado requerida con sus datos.
- Escribir funciones defensivas: debe suponer que la función podría encontrar una excepción en cualquier momento, así que diseñe sus funciones con la capacidad de continuar desde un punto de error anterior durante la próxima ejecución.
- No mezcle el código de prueba y de producción en la misma aplicación de función.
- Use código asincrónico, pero debe evitar el bloqueo de llamadas. Para ello, no haga referencia a la propiedad `Result` o al método `wait` en la instancia de la tarea.

Logic Apps

Aspectos que debe tener en cuenta antes de usar Logic Apps:

- Use el patrón de desarrollo "centrado en el mensaje". Se pueden encontrar algunos patrones en `https://bit.ly/2t2WPjH`.
- Logic Apps usa el patrón arquitectónico "If This Then That" (IFTTT), lo que significa que usa declaraciones condicionales y desencadenadores de eventos.

Logic Apps proporciona una variedad de desencadenadores y acciones con herramientas de administración para ayudar a centralizar el desarrollo de la API. Consisten en los siguientes:

- **Flujos de trabajo**: una forma gráfica de modelar los procesos empresariales como una serie de pasos o un flujo de trabajo.
- **Conectores administrados**: estos conectores se crean específicamente para ayudarlo cuando se conecte y trabaje con sus datos.
- **Desencadenadores**: algunos conectores administrados también pueden actuar como un desencadenador. Un desencadenador inicia una nueva instancia de un flujo de trabajo basado en un evento específico, como la llegada de un correo electrónico o un cambio en su cuenta de Azure Storage.
- **Acciones**: cada paso después del desencadenador en un flujo de trabajo se denomina acción. Cada acción normalmente se asigna a una operación en su conector administrado o aplicaciones de API personalizadas.
- **Enterprise Integration Pack**: para escenarios de integración más avanzados, Logic Apps incluye capacidades de BizTalk, la plataforma de integración de Microsoft. Los conectores de Enterprise Integration Pack le permiten incluir fácilmente validación, transformación y mucho más en los flujos de trabajo de Logic Apps.

Event Grid

Event Grid es un servicio de Azure para administrar el enrutamiento de eventos dentro de su sistema. Es especialmente útil en las arquitecturas sin servidor, donde puede hacer el difícil trabajo de conectar sus orígenes de datos con los controladores de eventos. Vale la pena comprender algunos de los conceptos y términos que sustentan Event Grid a fin de que pueda evaluarlo para sus necesidades:

- "Qué ocurre" es el evento. Un ejemplo es si Event Grid está conectado a una suscripción y se agrega un rol de seguridad; a continuación, se generaría un evento para Event Grid.
- "Dónde se realizó esto" es el origen del evento. Por ejemplo, esto podría ser una suscripción. Es el objeto que tiene el evento.
- Se hace referencia a "Punto de conexión del publicador" como el tema. Este es el punto de conexión para el que se publica el evento.
- "Filtrado de eventos entrantes o enrutamiento de evento" es la suscripción al evento, el consumidor del evento.
- "Lo que reacciona a un evento" es el controlador del evento. Esta es la acción del consumidor para el evento.
- La seguridad se controla a través de claves de autenticación o tokens de SAS.
- Se integra el reintento para confirmar la recepción de eventos.
- Para procesar eventos de publicación en lote, se pueden aceptar lotes de hasta 1 MB y se recomienda que cada evento sea inferior a 64 KB.

Ahora que tratamos los conceptos básicos de una arquitectura de sistema sin servidor, demos un vistazo a una arquitectura de aplicaciones móviles en Azure.

Diseño de aplicaciones móviles

Si bien el desarrollo móvil en sí no se relaciona con la nube, tiene sentido agregar una descripción rápida de cómo se vería el desarrollo de una solución móvil. Cuando se desarrolla una aplicación móvil, se debe considerar la compatibilidad con varias plataformas, por lo que se recomienda usar una plataforma como Visual Studio con Xamarin. Esto permite crear experiencias nativas con una sola base de código. Tenga en cuenta que aún necesitará hardware de Apple, como un MacBook, para compilar y cargar archivos en la Apple App Store, pero un único entorno de desarrollo realmente simplifica el proceso.

Esto le dará la oportunidad de aprovechar todos sus servicios en la nube favoritos en un entorno al que está acostumbrado, como puede ver en la *Figura 8*:

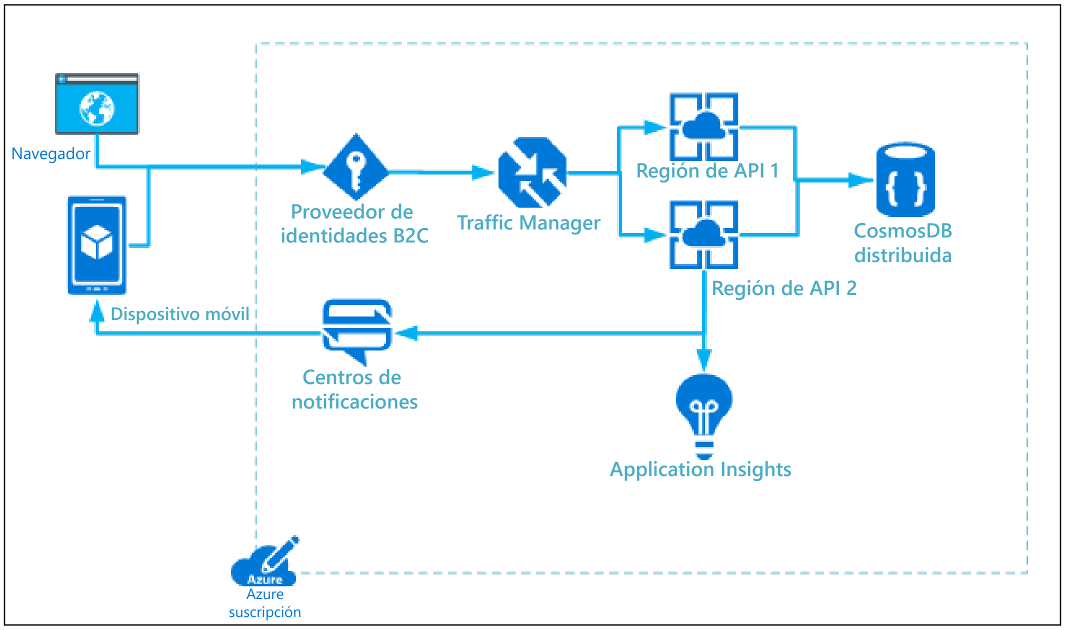

Figura 8: Arquitectura móvil simple

Consideremos la arquitectura móvil con los cinco principios que tratamos anteriormente:

- La administración de costos para el desarrollo de aplicaciones móviles abarca su modelo de seguridad, **Azure Active Directory** (**AAD**), B2C, etc., y sus servicios de notificación. Tendrá que factorizar los servicios de back-end y los repositorios de datos.
- La línea de base de seguridad radica en la seguridad necesaria para acceder al recurso que eligió anteriormente, además del acceso al punto de conexión expuesto para el microservicio, es decir, HTTP/HTTPS. Recomendación: si su servicio maneja CUALQUIER información confidencial, use HTTPS para proteger el punto de conexión.
- La coherencia del recurso se basa en la estrategia que seleccionó anteriormente. No debe elegir varias formas de implementar y administrar su recurso, así que seleccione un enfoque predecible que pueda proporcionar protecciones para su organización, que cubran todos los riesgos de desarrollo, implementación y administración que se hayan identificado.

- La línea de base de la identidad será la forma en que alguien obtendrá acceso a AAD o B2C desde un dispositivo y usará la administración de API para controlar y administrar el acceso. Independientemente de lo que elija, debe esforzarse por conservar este enfoque para la estrategia en su conjunto.
- La aceleración de la implementación debe seguir un flujo de Azure DevOps y permitir una actualización fácil. Aprovechar una solución con ranuras como funciones o Web Apps, o servicios de actualización administrados es útil para minimizar el tiempo de inactividad del servicio, si no lo elimina.

Si bien las soluciones móviles no se consideran realmente relacionadas con la nube, sino que en su lugar son nativas del dispositivo, dependen de la nube para sus servicios de procesamiento de back-end y administración. Es agradable ver el entorno de desarrollo de servicios de back-end y desarrollo móvil alineado dentro del ecosistema de Visual Studio. Puede encontrar la documentación en línea de Xamarin en `https://bit.ly/2FOlIm8`. Pasemos ahora a otro caso de uso de arquitectura de nube común, la IoT.

Diseño de un ecosistema de IoT

Las aplicaciones de la Internet de las Cosas (IoT) normalmente se describen como las cosas o los dispositivos que envían datos para generar información que desencadene acciones a fin de realizar un proceso. Un buen ejemplo es el uso de un sensor para saber si la puerta de un congelador está abierta. Cuando el sensor detecta que la puerta está abierta, envía información que genera una acción de notificar a alguien que la puerta está abierta.

IoT Central, en `https://docs.microsoft.com/azure/iot-central/`, es una solución basada en SaaS administrada para controlar sus dispositivos de IoT. No permite tanta personalización como una solución basada en PaaS, pero es fácil de usar. Los dispositivos de IoT tienen que ver con telemetría, básicamente leer sensores para medir y obtener información sobre el dispositivo de IoT en sí, como puede ver en la *Figura 9*. Hay dos formas de procesar la telemetría a medida que se envía:

- El almacenamiento a corto plazo, o "ruta de acceso activa", es casi telemetría en tiempo real y, por lo general, se controla mediante un motor de procesos de flujo, que habitualmente crea la acción de alertas o registro de consultas para herramientas de análisis.
- El almacenamiento a largo plazo, o "ruta de acceso en frío", usa el procesamiento basado en intervalos y normalmente se ocupa de grandes volúmenes de datos, que, a menudo, serán procesados por "machine learning" para descubrir acciones de procesos empresariales más definidas.

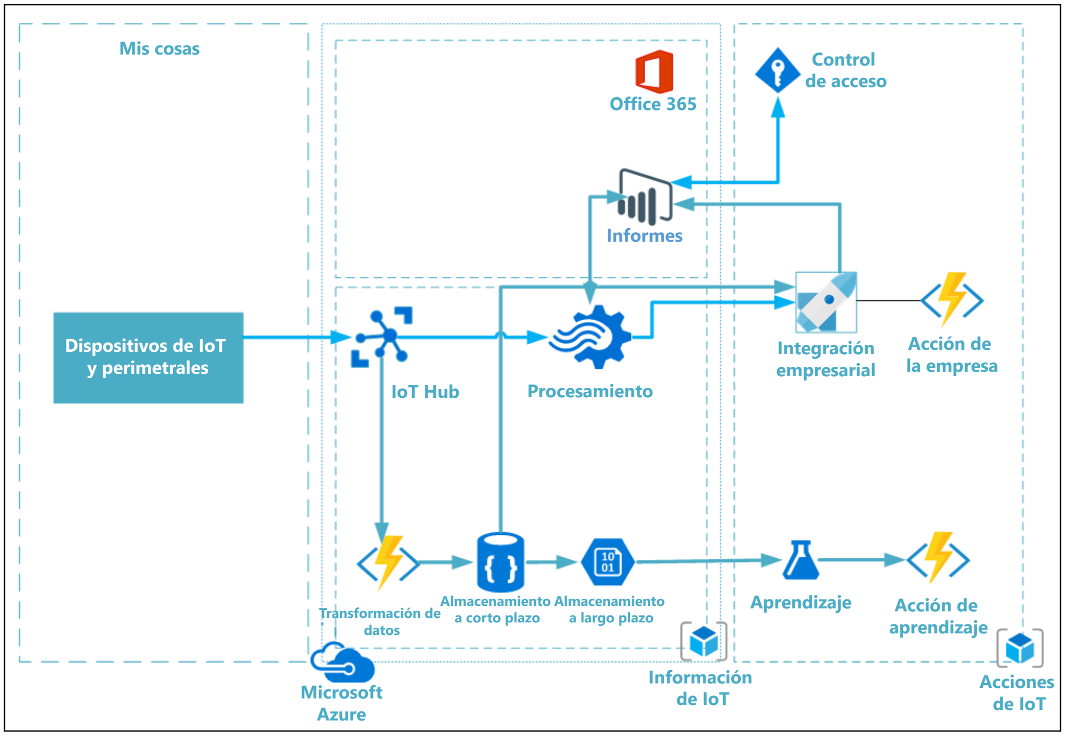

Figura 9: Arquitectura de IoT sencilla

Examinemos la lista de las cosas que usamos en nuestra arquitectura sencilla:

- Dispositivos de IoT, que son las cosas que conectamos de forma segura a la nube.
- IoT Hub o gateway es la conexión segura para la ingesta de datos.
- El procesamiento de transmisiones es cuando los datos se analizan a medida que llegan a través del gateway.
- El almacenamiento a corto plazo o ruta activa es una entrada manuscrita pendiente a corto plazo para los datos entrantes.
- El almacenamiento a largo plazo o la ruta de acceso en frío es una entrada manuscrita pendiente a largo plazo para datos históricos.
- Las transformaciones de datos ayudan a proporcionar una forma de datos coherente para los motores de almacenamiento y procesamiento.
- El procesamiento empresarial es la acción que se realiza en la información.
- La administración de usuarios se refiere a las acciones de control de acceso que se ejecutan en Azure y en los dispositivos de IoT, como las actualizaciones.

Estos son algunos de los aspectos que debe entender y que afectan el escalado de una arquitectura de IoT:

- Recuerde sus cuotas diarias: `https://bit.ly/2FPYZ8Y`.
- Conozca la cuota de dispositivo de los dispositivos conectados.
- Vigile de cerca el rendimiento de la ingesta y del proceso.

Use la paralelización en Stream Analytics para dividir la carga de trabajo entre nodos. Hay una cantidad máxima de instancias de función por partición de IoT Hub, por lo que se recomienda procesar el mensaje en lotes.

Existen algunas consideraciones de seguridad específicas que debe tener en cuenta con los dispositivos de IoT:

- Implemente una directiva de cifrado de datos segura.
- Proporcione una directiva de cifrado de datos firmada digitalmente.
- Asegúrese de que sea compatible con TLS 1.2 y DTLS 1.2.
- Asegúrese de que el almacén de claves y las claves del dispositivo se puedan actualizar.
- Asegúrese de que la actualización del dispositivo, como el firmware y el software de la aplicación, sea parte del proceso.

Para obtener ayuda con la solución de problemas, habilite la supervisión y el registro tanto como pueda con los subsistemas. La supervisión y el registro pueden proporcionar respuestas a las siguientes preguntas operativas:

- ¿Existe una condición de error?
- ¿Hay algo que esté mal configurado?
- ¿Los datos son precisos?

Los sistemas de supervisión ayudan a proporcionar información sobre el estado, la seguridad, la estabilidad y el rendimiento de una solución de IoT. Algunas de las métricas que se recopilarán podrían ser las siguientes:

- Subsistemas y dispositivos de IoT que informen de cambios de configuración
- Rendimiento de lectura y escritura del repositorio de datos, cambios de esquema, registros de auditoría de seguridad, bloqueos o interbloqueos, rendimiento de índices, CPU, memoria y uso de disco
- Informes de estado de servicios administrados y cambios de configuración que afectan a los sistemas dependientes

Puede encontrar más información sobre las arquitecturas de IoT con Azure en `https://bit.ly/2Tb5Ulg`.

Pasemos ahora a nuestra arquitectura final para examinarla: aplicaciones basadas en web.

Diseño de aplicaciones basadas en la web

Siempre que comience a usar soluciones basadas en la web debe pensar en el mantenimiento. También debe comenzar desde una perspectiva táctil, puesto que los dispositivos táctiles son cada vez más populares, lo que debería llevar a un diseño más receptivo, en general. Por lo tanto, comencemos por revisar una lista de las características del diseño de las aplicaciones web modernas:

- Capacidad multiplataforma
- Hospedado en la nube
- Escalable (se prefiere autoescalable)
- Diseño modular (piezas más pequeñas)
- Acoplados libremente
- Comprobación fácil (se prefiere la automatización)
- Diseño con capacidad de respuesta
- Fácil de implementar

Ahora revisemos algunos principios de diseño comunes que ayudarán a lograr esas características:

- **Encapsulación**: limitar la cantidad de acceso exterior al estado interno de un objeto. Por lo tanto, si una persona desea cambiar el estado de un objeto, debe hacerlo a través de una función bien definida o un establecedor de propiedad, y no a través del acceso directo al estado privado de ese objeto.
- **Separación de inquietudes**: el principio básico de que las aplicaciones deben separarse en función del trabajo que se realice. Esto generalmente se reduce a separar el comportamiento de negocios principal de la lógica de la interfaz de usuario: en palabras simples, no ponga lógica de negocios en la interfaz de usuario.
- **Inversión de dependencias**: principio de OOP (programación orientada a objetos) según el cual los módulos de alto nivel no deben depender de módulos de bajo nivel; ambos deben depender de abstracciones. Primero céntrese en las abstracciones (las interfaces), en lugar de en los detalles de la implementación.
- **Omisión de persistencia**: las clases solo se deberían diseñar para solucionar el problema de negocios del momento, y no ser atenuadas por las inquietudes con la persistencia.
- **Contexto acotado**: un patrón del diseño impulsado por el dominio, en que los modelos se separan en diferentes contextos de dominio con una relación estricta. Esto permite que diferentes contextos evolucionen de manera distinta y tengan diferentes lexicones a la vez que mantienen una relación.

Recuerde que a medida que diseña su aplicación web debe tener en mente la creación de capas simples de la aplicación para ayudar a la separación de inquietudes, como puede ver en la *Figura 10*:

Figura 10: Capas básicas de la aplicación

Estos son algunos de los procedimientos recomendados para las aplicaciones web:

- La coubicación debe estar en la misma región para ayudar a reducir la latencia y el costo monetario por las transferencias entre regiones.
- Habilite la característica de recuperación automática de App Service a fin de que su aplicación web siga en buen estado.
- Considere usar el escalado automático para combatir la presión de la CPU.
- Realice pruebas de carga de su aplicación web.
- Utilice ranuras de implementación cuando pueda.
- Habilite el registro de diagnósticos.
- Habilite la supervisión.
- Desacople la lógica de la aplicación.
- Automatice su infraestructura con plantillas de ARM.

Encontrará documentación en línea para las arquitecturas de la aplicación web en `https://bit.ly/2R5VRLk`.

Ahora que abordamos los principales tipos de arquitectura de aplicaciones que usará con Azure, nos concentraremos en algunos procedimientos recomendados de diseño que se aplican a todas las arquitecturas.

Procedimientos recomendados de diseño de arquitectura

Para iniciar nuestro análisis de los procedimientos recomendados, tenemos que empezar con la seguridad. Como vimos en nuestra matriz de responsabilidades en la *Figura 2*, independientemente del tipo de servicio implementado, siempre mantendrá las responsabilidades de seguridad respecto a:

- Cuentas
- Acceso
- Puntos de conexión
- Datos

La seguridad es un elemento fundamental en su recorrido hacia la nube. Comprender sus datos y clasificarlos en categorías en función de la confidencialidad y el impacto empresarial ayudará a proporcionar información sobre las optimizaciones que es probable que su empresa no haya notado cuando los datos no estaban clasificados. Si bien la clasificación de datos no es un remedio milagroso que resolverá todos sus problemas, puede obtener beneficios, como mejoras en el cumplimiento, y mejorar las formas de administrar estos recursos. Esto también ayuda a su organización a mitigar el riesgo a través de la administración de derechos, el cifrado y la prevención de pérdida de datos, que pueden tener un gran costo, en función de las directivas de retención, si su organización no tiene un plan para ello. Estos tipos de directivas se pueden implementar en Azure a través de Blueprints y el Administrador de cumplimiento, lo que puede ayudar a las organizaciones a establecer un enfoque repetitivo y estandarizado. A través de los grupos de administración, Azure Blueprints proporciona una forma de ayudar a implementar y gobernar las suscripciones y los recursos dentro de esas suscripciones, mientras que el Administrador de cumplimiento proporciona una visión más integral de su posición de cumplimiento y de la protección general de los datos. Los grupos de administración también permiten la administración de directivas en todos los niveles dentro de la estructura de su grupo de administración. Estos son conceptos clave que actuarán como la base para su recorrido hacia la nube.

Profundicemos rápidamente en Blueprints y los grupos de administración con un poco más de detalle para proporcionar fundamentos respecto al motivo y el momento en que los usaría. Los grupos de administración proporcionan un enfoque centralizado para administrar la seguridad, las implementaciones y las directivas que existen en un nivel superior a las suscripciones.

Tenga en cuenta que los grupos de administración comienzan en el inquilino, que es la raíz, y pueden contener una cantidad limitada de árboles o grupos anidados. Estos límites y estructuras se pueden revisar en `https://bit.ly/2sa6MLI`. Conviene señalar que se debe crear su estructura para los grupos de administración a fin de eliminar la necesidad de administrar el control de acceso basado en roles (RBAC) en el nivel de recursos o de grupos de recursos.

Demos un vistazo a cómo se vería una estructura de grupo de administración sencilla:

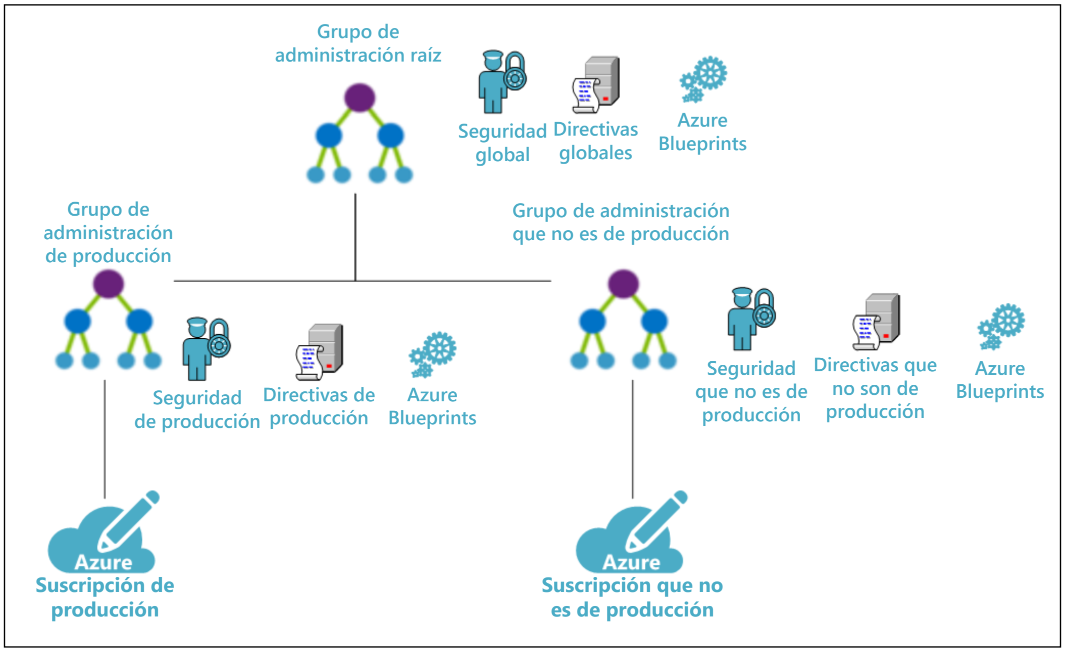

Figura 11: Una estructura de grupo de administración sencilla

Como puede ver en la *Figura 11*, hay un grupo de administración raíz y dos ramas que separan la producción de la no producción. Si tiene un grupo de seguridad, una directiva o un proyecto que abarque ramas, colóquelo en la rama más alta posible. Por ejemplo, si tengo un grupo de arquitectura que requiere acceso de propietario a todas las ramas, entonces, lo agregaría al inquilino raíz. Esto se debe a que los grupos de administración heredan un modelo primario/secundario.

Como puede ver, los grupos de administración le permiten definir su seguridad, directivas y proyectos (implementaciones) en cualquier nivel. Por ejemplo, puedo agregar el grupo de arquitectura en el nivel raíz porque su acceso abarca todas las suscripciones, pero solo podría agregar el grupo de desarrolladores a la rama que no es de producción. Esto también se convierte en directivas, puesto que podría limitar el tamaño de las VM en la rama que no es de producción para ayudar a controlar los costos. Se podrían aplicar Blueprints en los grupos de administración para ayudar a implementar entornos y costos mediante la asignación de la versión de desarrollo a la rama que no es de producción y una versión de producción a la rama de producción. Ahora que abarcamos los conceptos de por qué es posible que desee usar grupos de administración en su seguridad de la nube, enumeremos algunos procedimientos recomendados de seguridad general:

- Utilice grupos de administración para administrar la seguridad, las implementaciones iniciales y las directivas.

- Utilice Security Center Standard, por lo menos, para sus suscripciones de producción.
- Utilice Azure Key Vault para almacenar secretos y claves.
- Utilice **Web Application Firewall** (**WAF**) para controlar ataques y vulnerabilidades.
- Utilice la **Autenticación multifactor** (**MFA**) para proteger los métodos de autenticación de la organización.
- Utilice el cifrado en todos los recursos de almacenamiento, SQL, archivos, blobs, archivos de disco de VM, entre otros.
- Utilice las redes virtuales para aislar las VM, las aplicaciones y otros dispositivos.

Con esos procedimientos recomendados de seguridad básicos en mente, ahora analizaremos algunos procedimientos recomendados más específicos para las bases de datos, la Plataforma como servicio (PaaS) y la Infraestructura como servicio (IaaS) en Azure.

Estos son algunos de los procedimientos recomendados de seguridad para bases de datos:

- Utilice las reglas de IP del firewall para limitar el acceso y desactive la marca que permite el acceso a todos los recursos de Azure.
- Utilice mecanismos de autenticación como AAD para demostrar la identidad de un usuario; limite el uso de la autenticación de SQL tanto como sea posible.
- Aproveche un mecanismo de autorización para limitar el acceso y use el principio de privilegio mínimo.
- Utilice el Cifrado de datos transparente (TDE) para cifrar sus archivos de base de datos.
- Habilite la auditoría y la detección de amenazas para los datos confidenciales.

Puede encontrar documentación en línea para seguridad de bases de datos en `https://bit.ly/2NbGrnW`.

Estos son algunos procedimientos recomendados para PaaS:

- Utilice Key Vault para secretos.
- Utilice la Configuración de aplicación para la configuración de la aplicación, siempre que sea posible.
- Proteja y supervise todos los puntos de conexión, internos o externos.
- Proteja todos los puntos de conexión y las aplicaciones mediante una autenticación y autorización sólidas.
- Utilice siempre la codificación defensiva y cree tolerancia a errores.
- Utilice restricciones de IP o WAF para proteger los puntos de conexión.

Puede encontrar documentación en línea para seguridad de PaaS en `https://bit.ly/37QX53X`.

Procedimientos recomendados de IaaS:

- Controle el acceso a la VM y aplique directivas para el cumplimiento.
- Utilice plantillas de ARM de DevOps a fin de simplificar la configuración y la implementación para las repeticiones.
- Utilice un enfoque de privilegios mínimos para el acceso privilegiado.
- Utilice Security Center e instale software antimalware.
- Utilice conjuntos de disponibilidad para una alta confiabilidad.
- Mantenga siempre actualizadas las VM.
- Siempre cifre los discos de las VM.
- Supervise el rendimiento de las VM y busque amenazas.

Puede encontrar documentación en línea para seguridad de IaaS en `https://bit.ly/2T9T1b6`.

Como puede ver, la mayoría de los procedimientos recomendados se relacionan con la seguridad. Esto se debe a que con la mayoría de los recursos en la nube, algunos elementos se abstraen de su vista, lo que crea una oportunidad para centrarse más en sus soluciones de seguridad, en lugar de distraerse con las fuentes subyacentes. La excepción a esto es IaaS, donde todavía queda bastante a la vista.

Eso resume nuestro repaso de los tipos de arquitectura de aplicaciones en Azure. Puede revisar el conjunto completo de documentación en `https://bit.ly/2QFJFlE`. A continuación, demos un vistazo a la escalabilidad y la administración de aplicaciones en Azure.

Principios de diseño para aplicaciones escalables y administrables en Azure

Las aplicaciones en la nube deben ser capaces de responder a problemas o fallas, como la falta de disponibilidad, la pérdida de datos o red, los tiempos de espera o la transición del servicio. Algunos de estos problemas pueden ser temporales y un reintento básico puede superarlos, mientras que otros demandarán más esfuerzo. Lo primero que hay que hacer es crear sus aplicaciones pensando en la resistencia y la recuperación automática.

Los fundamentos de un sistema de recuperación automática son los siguientes:

- Detección automática del problema

- Tomar medidas para responder al problema detectado
- Auditar toda la información relevante sobre el problema

Las aplicaciones de recuperación automática se basan en diseñar las aplicaciones para resistencia, lo que significa que necesita una planificación ante errores con un tiempo de inactividad y pérdida de datos mínimos.

Diseño para resistencia

Hay dos características principales de las aplicaciones resistentes:

- Pueden recuperarse fácilmente de los errores y experimentar un tiempo de inactividad mínimo.
- Se ejecutan en buen estado sin tiempo de inactividad significativo.

La mayor diferencia con la resistencia en la nube es que puede escalar tanto vertical como horizontalmente.

Estas son algunas maneras de ayudarlo a crear resistencia:

- Defina sus requisitos según la descomposición de la carga de trabajo y las necesidades empresariales:
    - Documente el uso de sus cargas de trabajo.
    - Planifique el uso para garantizar el tiempo de actividad.
    - Defina una matriz de disponibilidad y recuperación para sus SLA.
- Utilice los procedimientos recomendados arquitectónicos para satisfacer los requisitos de su empresa:
    - Realice el análisis del modo de error (FMA) para identificar los tipos de errores que podría experimentar.
    - Cree un plan de redundancia para cada carga de trabajo, incluidos los costos.
    - Planifique la escala, tenga en cuenta los límites.
    - Planifique una suscripción y la asignación de recursos.
    - Planifique una estructura de grupo de administración.
    - Administre los datos a través de todo el almacenamiento, las copias de seguridad y las replicaciones.
- Realice muchas pruebas y conmutaciones por error forzadas:
    - Pruebe escenarios comunes de error.
    - Realice pruebas de carga.
    - Ejecute simulacros de problemas de aplicaciones.
    - Pruebe sondeos de estado.

    - Pruebe los sistemas de supervisión.
    - Pruebe los servicios externos y la forma en que se supervisan.

- Implemente de manera coherente y aproveche la automatización de forma anticipada:
    - Automatice la implementación de la aplicación y la infraestructura al principio de su proceso.
    - Registre y audite implementaciones.
    - Documente procesos y versiones.
- Supervise el estado para detectar errores y enviar alertas:
    - Implemente funciones de control y monitores de estado.
    - Mantenga registros de la aplicación.
    - Compruebe los flujos de trabajo de larga ejecución.
    - Realice seguimiento de excepciones y reintentos transitorios.
- Tenga un plan de recuperación:
    - Planifique la interacción de soporte con el proveedor de la nube.
    - Tenga un plan de recuperación ante desastres.
    - Prepárese para los errores de la aplicación.
    - Cuente con planes para:
        a. Recuperarse de la corrupción de datos
        b. Recuperarse de los errores del servicio
        c. Recuperarse de errores que afecten a toda la región
        d. Recuperarse de una interrupción de la red

Información general y consideraciones sobre la arquitectura

Como ya lo analizamos, la seguridad es el fundamento de los principios de la nube, y la clave para esto en Azure es comprender cómo funciona el RBAC. Como mostramos antes en el capítulo, aprovechar los grupos de administración también es extremadamente útil en la administración de acceso a la suscripción y, con RBAC, solo proporciona a los usuarios la cantidad mínima de acceso necesaria para completar sus trabajos. Esto también se puede combinar con **Privileged Identity Management** (**PIM**) si surge la necesidad de que un miembro aumente su acceso o necesite un acceso "Just-in-Time" en un entorno específico por un tiempo determinado, lo que ayuda con las aprobaciones y la auditoría.

Vale la pena obtener más detalles sobre la siguiente lista de herramientas y recursos disponibles en Azure para ayudar con la seguridad:

- RBAC
- Antimalware
- MFA
- PIM
- ExpressRoute
- VPN (también denominada Gateway de red virtual)
- Protección de la identidad
- Security Center
- Intelligent Security Graph

Administración identidad y Azure AD

AAD es la solución de Azure para la administración de identidades y acceso que respalda a un inquilino. AAD se diseñó como una solución multiinquilino que se basa en la nube y proporciona servicios de administración de identidades. Combina servicios de directorio básicos, administración de acceso a aplicaciones y protección de la identidad en una solución única.

Los siguientes son algunos procedimientos recomendados para la administración de identidades en AAD y seguridad del control de acceso:

- Trate la identidad como el perímetro principal para la seguridad.
- Centralice la administración de identidades mediante una sola instancia de AAD que se sincronice con directorios locales y asegúrese de habilitar la sincronización de hash de contraseñas cuando lo haga.
- Aproveche AAD en todos los nuevos proyectos de desarrollo que requieren identidades.
- Administre los inquilinos conectados mediante la evaluación del riesgo.
- Habilite SSO para los usuarios.
- Active PIM para la elevación de privilegios.
- Habilite la administración de contraseñas para sus usuarios.
- Aplique MFA para los usuarios a fin de mitigar el robo de identidad.
- Utilice RBAC junto con los grupos de administración.

Examinemos cómo podría funcionar un modelo simple en la *Figura 12*. Como puede ver, tenemos un Active Directory (AD) local y un AAD. Utilizan AAD Connect para sincronizar las identidades de manera que los usuarios puedan aprovechar el SSO para acceder a la red corporativa y a la aplicación basada en la nube:

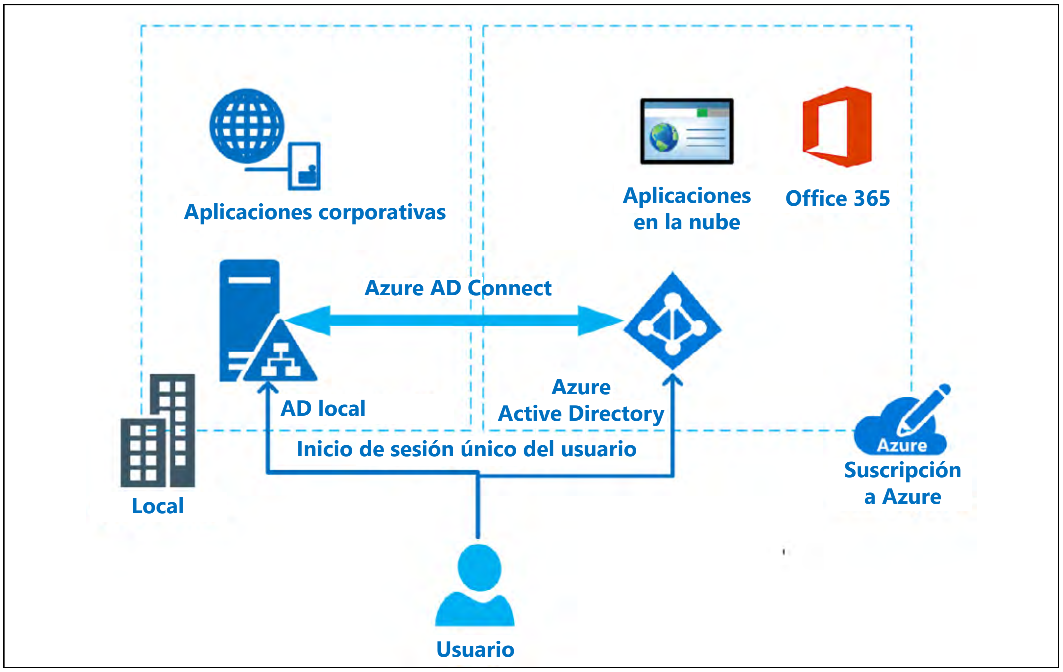

Figura 12: Modelo híbrido sencillo de sincronización AD

¿Cuándo usar B2C y B2B?

Business to Customer (B2C) y Business to Business (B2B) de AAD se crearon para funcionar con identidades externas dentro de AAD. B2C maneja aplicaciones orientadas al cliente con inicios de sesión sociales y de autoservicio, y B2B es para acceso orientado a los negocios. Estas funcionan casi como unidades organizativas (UO) dentro de un AD local y separan a estos tipos de usuarios de su AD corporativo principal. Aquí hay una forma sencilla de decidir cuándo usarlas:

- Cuando necesite autenticar a los usuarios en una organización externa y aprovechar las cuentas del trabajo o la escuela, use B2B.
- Cuando necesite invitar a los clientes a su aplicación web o móvil y le gustaría permitir también inicios de sesión sociales, use B2C.

Protección de los datos

En primer lugar, debemos entender qué proporciona de forma predeterminada el proveedor de nube y los requisitos de control de acceso establecidos por la directiva de seguridad de Azure, que establece lo siguiente:

- No hay acceso a los datos del cliente, de forma predeterminada.
- No hay cuentas de usuario o administrador en las máquinas virtuales del cliente.
- Conceda el mínimo acceso privilegiado para completar la tarea y la auditoría.

La protección de los datos se proporciona a través de la segregación de datos, que aísla lógicamente a los clientes entre sí. TDE proporciona mayor seguridad mediante el cifrado de los datos en reposo, que abarca desde las bases de datos SQL hasta las VM. Los datos en tránsito también se cifran del cliente a la nube y, además, entre los sistemas en la nube. También tiene la capacidad de proteger aún más sus datos a través de la redundancia de datos, tanto dentro como fuera del país, y, a menor escala, dentro y fuera de las regiones, según su cumplimiento. Esto significa que los datos se replicarán con los rangos de cumplimiento definidos. Se ofrecen tres tipos de redundancia:

- **Almacenamiento con redundancia local (LRS)**: esta opción hace tres copias de sus datos en una sola instalación en una única región.
- **Almacenamiento con redundancia de zona (ZRS)**: esta opción hace tres copias de sus datos en dos o tres instalaciones con una sola región o en dos regiones, lo que proporciona durabilidad dentro de una sola región.
- **Almacenamiento con redundancia geográfica (GRS)**: esta opción hace seis copias de sus datos con tres copias dentro de la región primaria y tres copias dentro de una región secundaria, lo que proporciona durabilidad dentro de dos regiones separadas.

La destrucción de datos en la nube tiene directivas estrictas para garantizar la sobrescritura de los recursos de almacenamiento antes de que se reutilicen. Los datos del cliente nunca se inspeccionan, aprueban, supervisan ni se consideran propiedad de un proveedor de nube; esta es responsabilidad de los clientes. Esto también significa que el cliente es propietario de la retención de los datos.

Redes

Las redes en la nube no difieren demasiado de las redes locales o de centros de datos, salvo por el aislamiento, la disponibilidad y el escalado. No olvide que los espacios de direcciones se componen de subredes que están contenidas dentro de las suscripciones. Las redes virtuales son buenas para aislar recursos en Azure, así como para proporcionar un lugar de llegada para VPN o ExpressRoute destinadas a redes de proveedores locales u otras redes de proveedor de nube.

Destinemos un momento a revisar algunos conceptos de VNet:

- Los espacios de direcciones IP son grandes depósitos de direcciones IP aceptables.
- Las subredes son partes del espacio de direcciones de IP que segmentan las IP en fragmentos más pequeños, que pueden ser controlados por diferentes Grupos de seguridad de red (NSG).
- Las redes virtuales solo pueden residir en una sola región y se extienden a la suscripción, que es un límite difícil.

Aquí se indican algunos procedimientos recomendados para usar VNets:

- Trate de no superponer los espacios de direcciones.
- Las subredes no deben abarcar todo el espacio de direcciones.
- Mantenga las VNets al mínimo.
- Use los NSG para proteger su subred en sus VNets. Puede usar Grupos de seguridad de aplicación (ASG) dentro de los NSG para ayudar a controlar aún más el flujo de red a los recursos dentro de la subred.

Cuando se comunique con recursos en la nube, puede conectar los recursos directamente a las VNets, creando un límite aislado. También puede utilizar un punto de conexión de servicio en VNets para proteger los recursos sin necesidad de una conexión directa. La conexión de VNets entre sí se puede hacer a través del emparejamiento. La conexión a la nube del entorno local a las VNets también es una opción. Puede conectarse desde una VNet hasta un único recurso en una red corporativa a través de una conexión VPN de punto a sitio. También puede establecer una VPN de sitio a sitio completa para conectarse a una red corporativa o utilizar un ExpressRoute, que es un circuito entre la nube y su red. La *Figura 13* muestra una red híbrida que utiliza un servicio ExpressRoute que requiere una conexión subyacente de terceros a la nube. La red también usa una VPN de sitio a sitio como una copia de seguridad si el circuito principal deja de funcionar:

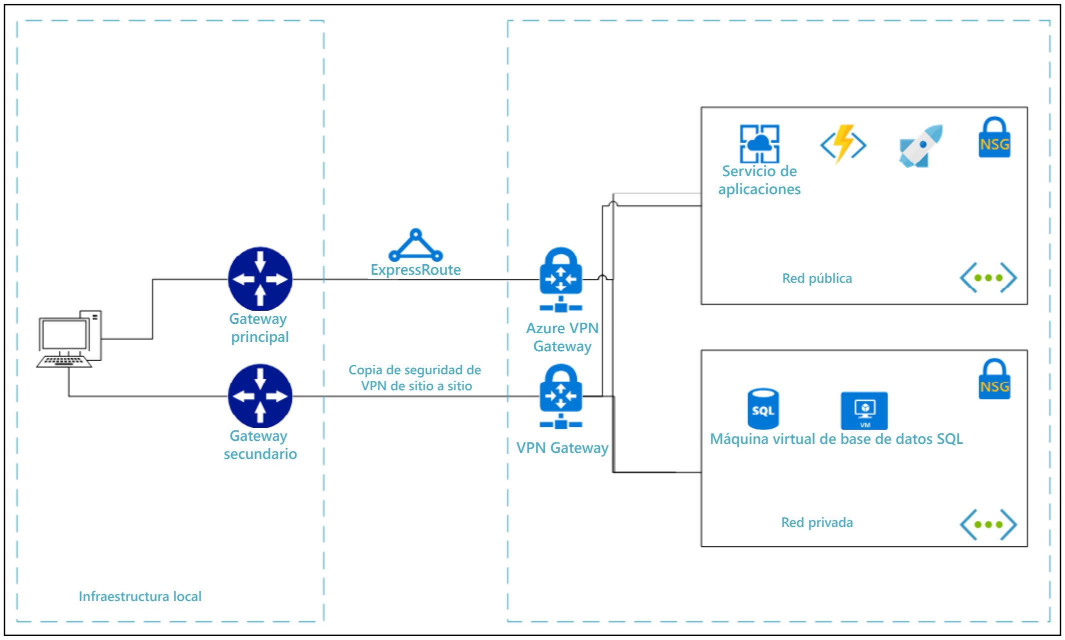

Figura 13: Una red híbrida sencilla entre el entorno local y la nube

Puede aprovechar los NSG que filtrarán el tráfico para las reglas de seguridad entrantes y salientes. Esto también se puede hacer con dispositivos como los firewalls. En nuestro ejemplo, usamos un NSG sencillo para impedir que las VM de VNet privadas accedan a Internet.

Azure para aplicaciones en contenedores

La primera pregunta que nos hacemos cuando analizamos los contenedores es, ¿por qué deberíamos preocuparnos por ellos? Esta suele responderse de dos maneras. En primer lugar, los contenedores proporcionan la libertad de trasladar su aplicación de una instalación local a la nube o dentro de la nube a otro proveedor de nube sin cambiar el código a su aplicación. En segundo lugar, cada aplicación es autónoma, lo que significa que todos los elementos de la aplicación y sus versiones se encuentran dentro de los límites del contenedor, por lo que cambiar una biblioteca para una aplicación no afectará negativamente ni provocará una nueva implementación de todas las aplicaciones que compartían la biblioteca.

Entonces, ¿qué son los contenedores? Para describir los contenedores, me gusta usar la analogía de la caja de zapatos. Una caja de zapatos es bastante estandarizada y pequeña. No puede colocar muchas cosas en ellas y suelen tener una sola función, como almacenar sus fotos. Puede almacenar esta caja de zapatos y moverla con bastante facilidad. Sin embargo, se requiere un estante o un piso para apilarlas. Los contenedores, como estas cajas de zapatos, son un paquete estandarizado de software que contiene todo lo que necesita, desde código, tiempo de ejecución, herramientas, bibliotecas y configuración. También se pueden mover fácilmente, pero al igual que las cajas de zapatos, requieren algo de infraestructura para poder apilarlas. Esto podría parecer similar a nuestro análisis de los microservicios; sin embargo, los contenedores pueden albergar un poco más. La *Figura 14* es un ejemplo de una aplicación de contenedor sencilla:

Figura 14: Aplicación de contenedor sencilla

Una de las tecnologías de contenedores más estandarizadas que se utilizan en la nube es Docker, que existe desde el año 2008. Es open source y le permite empaquetar, enviar y ejecutar sus aplicaciones en todos los entornos de nube actuales. Ahora, antes de que se precipite a usar los contenedores, debemos analizar algunos puntos clave. ¿Recuerda que dijimos que necesita algo de infraestructura? Esto se conoce como organización u organizador, y aquí es donde entra en juego Kubernetes en la nube. Como nota al margen, solo usaría instancias de contenedor en una prueba de concepto (POC) o una aplicación rápida, puesto que no contienen un organizador. Esta tecnología puede escalar sus contenedores de forma vertical y horizontal según el uso. El organizador puede implementar cambios y puede revertir las instancias que fallan o no pasan las comprobaciones de estado. También ayuda a simplificar la administración y el equilibrio de carga para los contenedores o un conjunto de Docker.

Herramientas y servicios de contenedor de Azure

Azure proporciona varias plataformas compatibles con contenedores:

- AKS
- Azure Red Hat OpenShift
- ACI
- Web App for Containers

Visual Studio y Visual Studio Code también han agregado compatibilidad para crear contenedores de código de .NET Core. Docker también proporcionó soporte para contenedores de Windows a fin de mejorar el ecosistema de .NET. Una vez que la carga de trabajo esté activa, podrá crear contenedores de código .NET Core. Para ello, simplemente debe hacer clic con el botón secundario y seleccionar **Agregar contenedor de Docker** (Add Docker Container).

Azure Red Hat OpenShift

Esta plataforma proporciona un único panel sencillo que amplía Kubernetes mediante herramientas y recursos adicionales para ayudar con las imágenes, el almacenamiento, las redes, el registro y la supervisión. En otras palabras, le da la capacidad de elegir su propio almacenamiento, registro de imágenes, redes y otras herramientas para ayudar con la automatización. También ayuda a proporcionar una mejor integración con AAD y Kubernetes RBAC junto con la supervisión del estado de sus recursos y clústeres. Puede obtener más información en `https://docs.microsoft.com/azure/openshift/`.

Azure Container Instances (ACI)

Las instancias de contenedor son una forma muy simple de crear un contenedor aislado para aplicaciones pequeñas o tareas que no requieren una organización completa. Estas instancias tienen un tiempo de inicio rápido y permiten la asignación de nombres DNS y direcciones IP públicas. Los contenedores se pueden agrupar y aprovechan las redes virtuales. Debe revisar y comprender los límites que se encuentran en `https://bit.ly/30hsme5`.

Las instancias de contenedor son la forma más sencilla de comenzar a trabajar con contenedores. Además, son una excelente forma de aplicar pruebas de concepto para las aplicaciones desde la perspectiva de la arquitectura.

Puede encontrar más información sobre las ACI en `https://bit.ly/2Tabwfj`.

Azure Kubernetes Service (AKS)

AKS es un clúster de Kubernetes sencillo administrado en Azure. Elimina la necesidad de saber cómo configurar Kubernetes con el fin de crear una infraestructura de contenedor. También se beneficiará de las capacidades de Kubernetes, como el mantenimiento y la supervisión de estado. Azure administra los maestros de Kubernetes y son gratuitos; solo paga por los nodos de agente dentro de los clústeres. También puede integrar AKS con AAD, lo que permite el uso de RBAC de Kubernetes. Los datos de supervisión se almacenan en el área de trabajo de Azure Log Analytics, que puede supervisar el rendimiento del clúster y el estado de las cargas de trabajo dentro de AKS.

Estos son los componentes clave que se deben conocer:

- Acceso, seguridad y supervisión: AKS usa RBAC para ayudar a controlar los recursos y espacios de nombres. Se puede conectar a AAD. El acceso se puede configurar en función de la identidad existente o la pertenencia a un grupo.
- Clúster y nodos: los nodos de AKS se ejecutan en VM de Azure y los clústeres ejecutan varios grupos de nodos, por lo que se admiten sistemas operativos mixtos. Puede usar los autoescaladores de pod y clúster, que pueden escalar automáticamente en función de la demanda.
- Los nodos de AKS admiten la integración con VNets en la mayoría de las configuraciones.
- Herramientas de desarrollo: hay una extensión de Kubernetes para Visual Studio Code, así como otras herramientas como Helm y Draft.
- Soporte de Docker
- Controlador de ingreso: este ayuda a controlar el enrutamiento de tráfico, el proxy inverso y la terminación de TLS.

Puede encontrar documentación en línea de AKS en `https://bit.ly/37TKSvv`.

También se pueden aprovechar los contenedores de Docker para Web Apps. Es similar a una instancia de contenedor. Puede encontrar la documentación en línea en `https://bit.ly/2QFR4BB`.

Resumen

Tratamos muchos temas en un breve espacio de tiempo, pero esperamos que haya obtenido una visión general de las diversas arquitecturas disponibles para crear aplicaciones con Azure. También analizamos los procedimientos recomendados de diseño de aplicaciones, que se centran principalmente en la seguridad, y terminamos con una revisión del trabajo con contenedores en Azure. En el siguiente capítulo, analizaremos DevOps y cómo su metodología se integra con la nube.

3
Azure DevOps

Introducción

DevOps es una forma de ayudar a proporcionar una mejor experiencia de entrega de software. En este capítulo, analizaremos qué es DevOps y por qué debe usarlo en la nube. Descubrirá que hay bastantes beneficios una vez que comprenda cómo implementarlo correctamente. Además, Azure tiene mucho que ofrecer a las empresas que adoptan un enfoque de DevOps.

¿Por qué DevOps?

Muchas organizaciones que realizan el recorrido hacia la nube se esfuerzan por crear un proceso repetible para implementar sus aplicaciones y el control de cambios. Tienen que esforzarse debido a las líneas de responsabilidad que se analizaron en el *Capítulo 2*, *Opciones de arquitectura y principios de diseño*, justo aquí es donde DevOps y las adopciones de la nube tienen una gran sinergia. DevOps puede mejorar esa implementación, a la vez que reduce los costos en infraestructura, código, supervisión e implementaciones de directivas. Antes de que analicemos cómo funciona esto, primero conoceremos más sobre DevOps y por qué es importante en este recorrido.

DevOps comenzó como una metodología de desarrollo de software que acelera la creación, prueba y lanzamiento de aplicaciones de software al reunir a dos grupos principales, los **desarrolladores** (**Dev**) y las **operaciones** (**Ops**), para trabajar de forma más eficaz. Su objetivo no es ser un sustituto de cualquier metodología Agile o Lean, sino más bien un complemento para ellas. Llena los vacíos de las empresas tecnológicas para desglosar las especificaciones funcionales y no funcionales, a la vez que también trabaja para automatizar el proceso tanto como sea posible, en aras de la velocidad y la calidad, y perfecciona el proceso operativo.

DevOps es un método de desarrollo de software que enfatiza la comunicación, la colaboración, la integración, la automatización y la medición de la cooperación entre los desarrolladores de soluciones y los profesionales de TI.

El proceso de DevOps crea un proceso cíclico de planificación, codificación, creación, prueba, lanzamiento, implementación, operación y supervisión de las aplicaciones que desarrolla para su organización, lo que une el desarrollo con las operaciones. Esta es la representación principal del cambio cultural de DevOps, que tiene un proceso más detallado que la herramienta Microsoft Azure DevOps. Puede ver este ciclo ilustrado en la *Figura 1*:

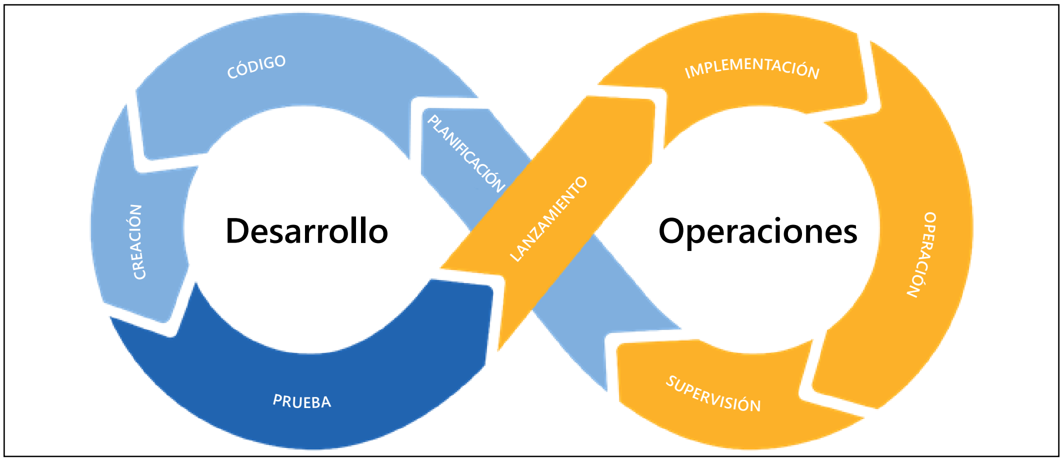

Figura 1: Bucle del proceso de DevOps

DevOps en su conjunto tiene que ver menos con lo que hace y más con cómo lo hace. Gracias a las tecnologías y los procesos en evolución, los principios de DevOps son fundamentales en su viaje hacia la transformación de su organización. Sin embargo, al final del día, se trata de cómo se hace el trabajo y cómo las personas interactúan unas con otras y con la tecnología para impulsar el rendimiento. DevOps no es una solución estándar que simplemente se compra y se ajusta. Microsoft ha lanzado herramientas como Azure DevOps que facilitan la integración de DevOps con Azure. Para Azure DevOps, este paradigma principal en torno a la forma en que DevOps se entregaba a una organización cambió, como puede ver en la *Figura 2* de Azure DevOps de Microsoft en `https://bit.ly/2Fx0OHI`:

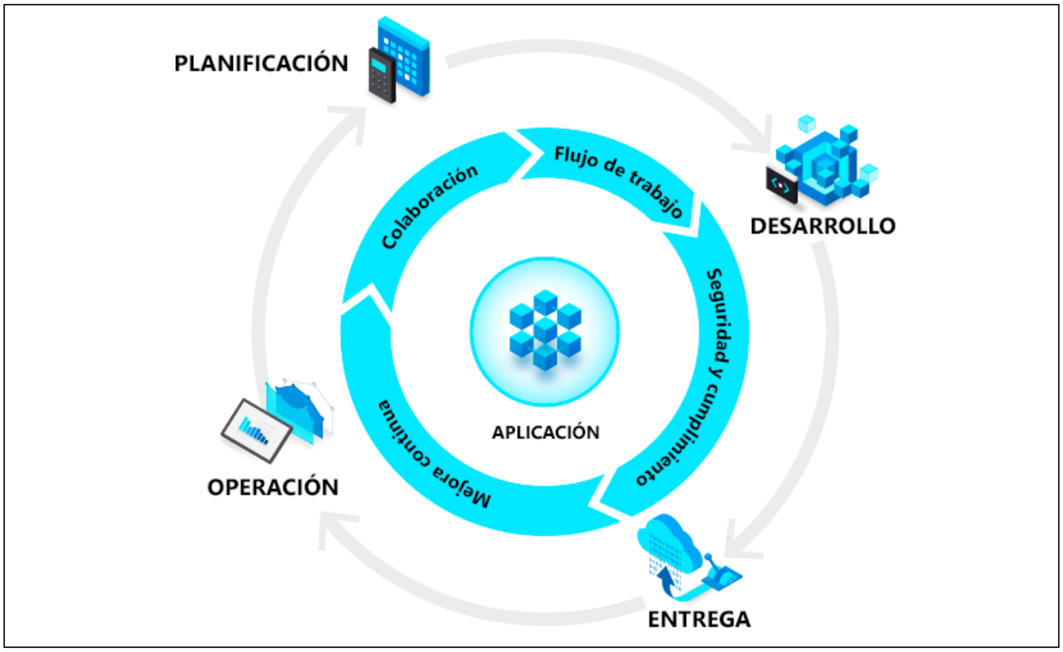

Figura 2: Círculo de procesos de Microsoft Azure DevOps

La principal simplificación está en la reflexión sobre las herramientas proporcionadas, que se analizarán más adelante en este capítulo. Para revisar rápidamente cómo esta simplificación afecta la planificación, el desarrollo, la entrega y las operaciones, la parte de **PLANIFICACIÓN** ayuda al equipo a visualizar su carga de trabajo, a la vez que ayuda a proporcionar una definición del trabajo y realizar un seguimiento de la carga de trabajo.

Esto ayuda a ofrecer un proceso de desarrollo más inteligente y rápido que se integra con Visual Studio Code y Visual Studio, lo que proporciona a los desarrolladores similares la capacidad de compartir y colaborar en su código, a la vez que automatiza las pruebas, compilaciones y versiones para un proceso de **integración continua** (**CI**) optimizada. Esto también lleva a entregar sus aplicaciones a través del proceso de **implementación continua** (**CD**), lo que permite que las variables definidas por el entorno proporcionen un proceso de entrega más estandarizado con significativamente menos problemas de implementación que las implementaciones manuales. Una vez que se entregan estas aplicaciones, debe poner en funcionamiento su entorno con alertas, registros y telemetría, lo que incluye directivas y cumplimiento a través de Azure Blueprints dentro de los grupos de administración.

Los grupos de administración son una forma de estandarizar sus suscripciones en Azure. Puede obtener más información sobre los grupos de administración en `https://bit.ly/2T2wU6m`, que recomiendo encarecidamente. Pero basta de desviarnos de nuestro tema. Volvamos a nuestro análisis de DevOps.

DevOps se trata de adoptar la comunicación mediante la unión de las operaciones con el desarrollador, lo que ofrece a todos espacio para opinar durante todo el ciclo de software. No es el trabajo de un solo individuo, sino más bien el trabajo de todos, y es una forma fundamental de trabajar en conjunto para impulsar el rendimiento. El componente personal del modelo parece ser la parte menos definida. Es importante darse cuenta de que simplemente hacer más cosas no es la solución para tener una transición exitosa de DevOps, o cualquier transición cultural en ese caso, en el ámbito tecnológico.

El objetivo principal de DevOps es permitir que los desarrolladores y las operaciones superen los desafíos a los que se enfrentan. Veamos lo que eso significa. A continuación, se indica cómo se consideran tradicionalmente los roles de desarrollador y operaciones:

Desarrolladores	Operaciones
• Característica entregada después de las pruebas unitarias manuales en los sistemas de desarrollo • Recompensado principalmente en la entrega oportuna • Los sistemas de desarrollo no son los mismos que los de producción • Poca o ninguna preocupación sobre el impacto en la infraestructura/implementación relacionado con los cambios en el código • Poco o ningún comentario sobre los cambios hasta más adelante en el ciclo de software	• Código entregado con poca o ninguna preocupación sobre la infraestructura • Recompensa principalmente por el tiempo de actividad • Debe resolver los problemas de seguridad • Debe solucionar la supervisión de la aplicación con poco o ningún conocimiento de la aplicación • Por lo general, se involucra al final del ciclo de desarrollo de aplicaciones

Como puede ver, los desarrolladores y las operaciones casi se enfrentan entre sí, principalmente en la forma en que se les recompensa. Esto puede causar una lucha cuesta arriba, con cada grupo lanzando granadas al otro en forma de tareas cuando sus objetivos entran en conflicto. Ninguno de los lados realmente entiende las "razones" detrás de la urgencia de lo que se le pide. Algunos atributos importantes que se necesitan para ayudar a reducir esta brecha son:

- La administración de versiones, que proporciona una comprensión más profunda de los riesgos, las dependencias y los problemas de cumplimiento
- La coordinación de versiones e implementación, un mejor seguimiento de actividades discretas, un escalamiento de problemas más coherente y más rápido, un control de procesos documentado y la creación de informes

- Automatización de versiones e implementación, procesos más coherentes y repetibles que se pueden invocar y ejecutar automáticamente con o sin (cuando no son de producción) un proceso de aprobación

Más adelante en el capítulo, veremos cómo algunos de estos atributos se pueden implementar en Azure. También es importante en DevOps definir marcadamente los roles y las responsabilidades. Estos son algunos roles dentro de una estructura de DevOps tradicional, que facilitan los objetivos de acelerar los procesos de la empresa:

- **Evangelista**: el líder que trabaja en la implementación y orquestación de los procesos de DevOps dentro de la organización. Es responsable de mantener el avance del proceso y respaldar la visión del proceso y los roles generales de DevOps.
- **Administrador de versiones**: responsable del proyecto en la producción, donde supervisa todo el desarrollo, las pruebas y la implementación para apoyar la entrega continua. Proporciona visibilidad del proceso a través de la medición e interpretación de las métricas en todas las tareas.
- **Control de calidad**: un rol clave en la entrega exitosa del proyecto, que es responsable de garantizar la experiencia general del usuario y que el producto esté libre de errores.
- **Ingenieros de seguridad y cumplimiento**: se aseguran de que el proyecto cumpla todos los estándares y regulaciones, a la vez que garantizan que el producto esté protegido de los ataques.
- **Desarrolladores**: responsables del desarrollo de la producción para cumplir con los requisitos de la empresa o del cliente.

Entonces, para responder concisamente a la pregunta de ¿por qué DevOps?, podemos definir varios beneficios que son los más importantes para los aspectos técnicos, culturales y comerciales de la operación:

Beneficios técnicos	Beneficios culturales	Beneficios empresariales
• Resolución más rápida de problemas • Menor complejidad • Entrega continua	• Equipos más productivos • Mayores compromisos con el proyecto • Mayor desarrollo profesional	• Entrega rápida de características • Entornos operativos más estables • Colaboración mejorada • Más tiempo de innovación

Ahora que analizamos lo que es DevOps, examinemos la metodología.

Azure DevOps: la metodología

Ocupemos unos minutos para revisar algunos aspectos que debe recordar durante su recorrido hacia DevOps. Se trata de la confianza: los equipos de desarrollo y operaciones suelen estar separados unos de otros y no pueden comunicarse, ni mucho menos colaborar de manera eficaz. DevOps tiene como objetivo resolver estos problemas. Debe entender a las personas, lo que comienza con comprendernos primero a nosotros mismos y, luego, a las personas que nos rodean. Esto llevará a terminar con buscar a un culpable. Espere fallar, fallar con frecuencia y fallar prematuramente, así que acepte el error y aprenda. Trabaje para quitar los obstáculos y mejorar el flujo, y no tema replantearse su proceso ni buscar un nuevo par de ojos para ver las cosas que no puede ver porque está demasiado cerca de los problemas.

Esfuércese por eliminar el trabajo no planificado, porque si la cultura es seguir haciendo las cosas a medias, entonces, es difícil dejar de lado las horas que se ocupan en apagar incendios para hacer que las cosas funcionen o para innovar. Tenga continuidad, adopte la integración continua y la entrega continua, y mantenga las cosas en marcha. Asegúrese de crear equipos multifuncionales y adoptar la transparencia en los equipos y procesos. Desarrolle dominio y propósito y cree objetivos únicos en los que todos puedan enfocarse. Estas son cosas difíciles que no suceden de la noche a la mañana, por lo que debe ser coherente en su enfoque, pero también estar abierto a las necesidades de los demás.

DevOps como filosofía es mucho más que solo herramientas, pero las herramientas adecuadas pueden hacer que todo el proceso sea mucho más fluido. Microsoft lanzó una herramienta que facilita la implementación de DevOps cuando se trabaja en Azure o con otros servicios en la nube y locales: Azure DevOps. Esta herramienta se encuentra disponible como un servicio en la nube administrado, además de un producto autohospedable llamado Azure DevOps Server (antes conocido como Team Foundation Server o TFS). Para comenzar a usar el proceso, debe ir a `https://azure.com/devops`. Demos un vistazo a la metodología del uso de Azure DevOps, para que pueda tomar mejores decisiones cuando la configure por primera vez. En primer lugar, Azure DevOps le permite usar cuatro tipos de metodología de proyecto:

- **Básica** es la más fundamental y ligera de las metodologías. Esta utiliza problemas, epopeyas y tareas para realizar un seguimiento del funcionamiento del proyecto. En la *Figura 3*, se muestra la forma en que la asociación funciona en la plantilla básica:

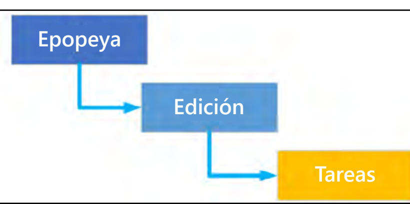

Figura 3: Flujo Básico del proceso del elemento de trabajo

- **Scrum** es una administración de procesos ligera. Tiene más flujo que el método Básico, porque usa epopeya, características, elementos de trabajo pendiente del producto, tareas, impedimentos y tareas de errores para realizar un seguimiento del trabajo del proyecto. En la *Figura 4*, se muestra la forma en que la asociación funciona en la plantilla Scrum:

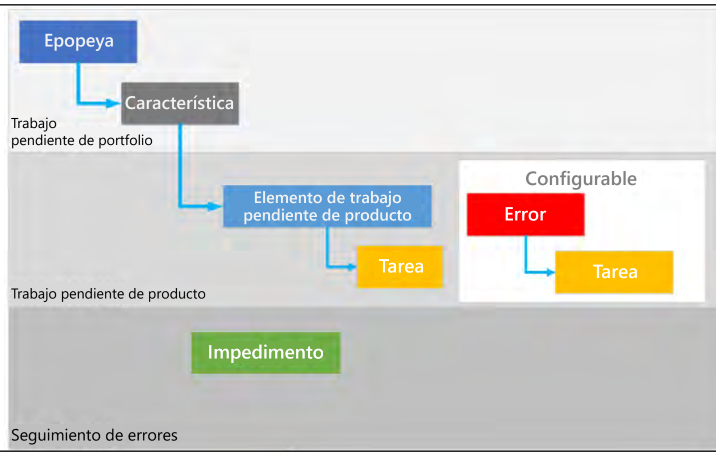

Figura 4: Flujo Scrum del proceso del elemento de trabajo

- **Agile** le permite trabajar a través de guiones gráficos y tiene más elementos de proceso que Scrum. Tiene un trabajo pendiente de portfolio para sus epopeyas y características, con un trabajo pendiente para los casos de usuario con sus tareas y errores, y también existe un sistema de seguimiento de problemas para sus proyectos. En la *Figura 5*, se muestra cómo funciona la asociación con la plantilla Agile:

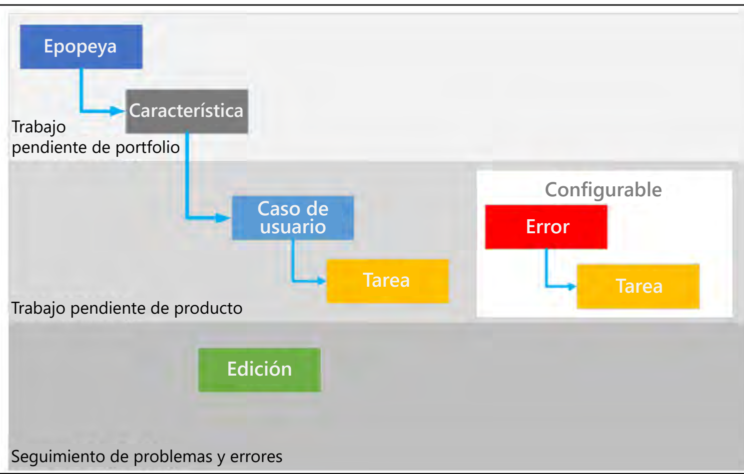

Figura 5: Flujo Agile del proceso del elemento de trabajo

- **CMMI**, que son las siglas de **Capability Maturity Model Integration**, es el método de proyecto más formal. Ayuda a mantener un registro auditable de decisiones. Este proceso lo ayuda a realizar seguimiento de los requisitos, la administración de cambios, el riesgo y las revisiones. Tiene un trabajo pendiente de portfolio para sus epopeyas y características, que realiza un seguimiento de sus requisitos y sus tareas y errores en su trabajo pendiente, a la vez que rastrea las solicitudes de cambio, los problemas, los riesgos y las revisiones en la sección del proyecto de problema, cambio y administración de riesgos. En la *Figura 6*, se muestra cómo funciona la asociación con la plantilla CMMI:

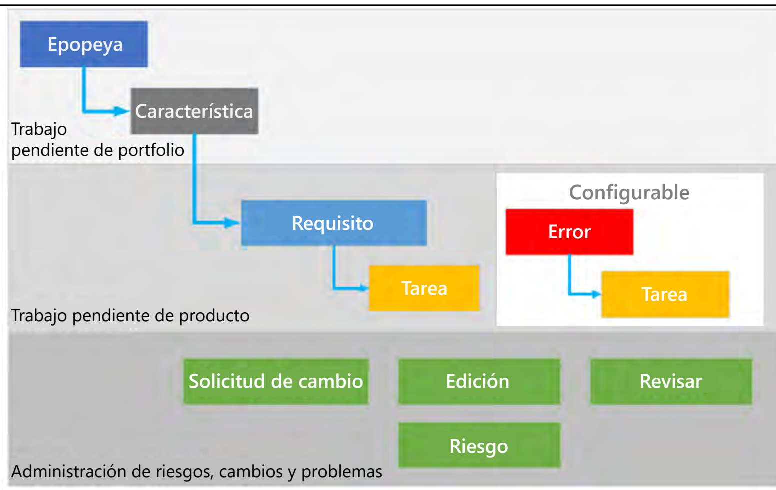

Figura 6: Flujo CMMI del proceso del elemento de trabajo

Como puede ver, hay una amplia gama de metodologías que puede usar para administrar sus proyectos y portfolios de proyecto. Puede usar Azure DevOps como el origen para todos los proyectos de su organización, ya sea que requieran código o no. Puede crear una definición respecto a lo que significa la finalización de la tarea y hacerla coherente en todos los proyectos, de manera que el equipo conozca la definición, independientemente del proyecto.

Con estos procesos, puede automatizar completamente la entrega del software hasta el entorno de producción o configurar procesos semiautomatizados con aprobaciones e implementación a petición para cualquier plataforma. Azure DevOps proporciona cinco áreas principales que ayudan a apoyar estos procesos en un paquete de software. Es igual para cada metodología que elija. Veamos cómo funciona cada una de estas áreas:

Azure Boards	Azure Repos	Azure Pipelines
Ayuda a administrar los requisitos de sus proyectos a través de trabajos pendientes con paneles e informes integrados, y ayuda a realizar un seguimiento rápido de los elementos de trabajo pendiente, los casos de usuario, las tareas, las características y los errores.	Este hospeda los repositorios de código. Actualmente tiene dos variedades, Git and **Team Foundation Version Control (TFVC).** Ayuda con la administración de versiones y la copia de seguridad del código fuente de la aplicación.	Se utiliza para crear y probar los proyectos de código y hacerlos disponibles para otros procesos. Combina CI y CD con pruebas continuas para que el código se pueda enviar.
Azure Artifacts	**Azure Test Plans**	
Este es el concepto de múltiples fuentes que utiliza para controlar el acceso y organizar sus paquetes. Así es como hace que su paquete esté disponible para otros proyectos, como `nuget. org.`	Ayuda a rastrear y administrar sus técnicas de pruebas manuales y exploratorias.	

Como puede ver, Azure DevOps proporciona un amplio conjunto de herramientas para ayudar a completar cualquier proyecto, y proporciona una extensión para la integración en aplicaciones de terceros, como Slack o Trello. Una de las adiciones más importantes fue la mayor integración con GitHub para una plataforma de desarrollo de software más colaborativa, de mente abierta y open source que compró Microsoft. Azure DevOps le permite conectar sus paneles a github.com para administrar las confirmaciones y enviar solicitudes mientras las vincula con elementos de trabajo del sistema, lo que también permite que los problemas se rastreen con GitHub. Esto le otorga la capacidad de usar los paneles de Azure DevOps para administrar su proyecto de GitHub. Ahora profundizaremos en la forma en que se pueden usar para entregar aplicaciones en la nube.

En los siguientes escenarios, analizaremos en profundidad la configuración de Azure DevOps. Daremos un vistazo a las opiniones y los flujos de trabajo personales del autor, y a cómo se pueden aplicar en sus propios proyectos.

Cómo reunir requisitos en Azure Boards

Me gusta comenzar todos mis proyectos reuniendo los requisitos en el trabajo pendiente y usar Azure Boards para administrarlos. Antes de comenzar, tendrá que configurar todo en función de la plantilla de estructura del proyecto que pretende utilizar. Por lo general, elijo Scrum, por lo que le mostraré algunos elementos que necesita configurar antes de crear un trabajo pendiente para la plantilla Scrum.

Utilizo Scrum porque tiene algunos procesos formales, como las actualizaciones y la planificación diarias, y tiene un control un poco más interactiv o, de manera que puede corregir el rumbo cuando sea necesario. Debe comenzar con rutas e iteraciones, que se almacenan en **Configuración del proyecto** (Project settings), como puede ver en la *Figura 7*:

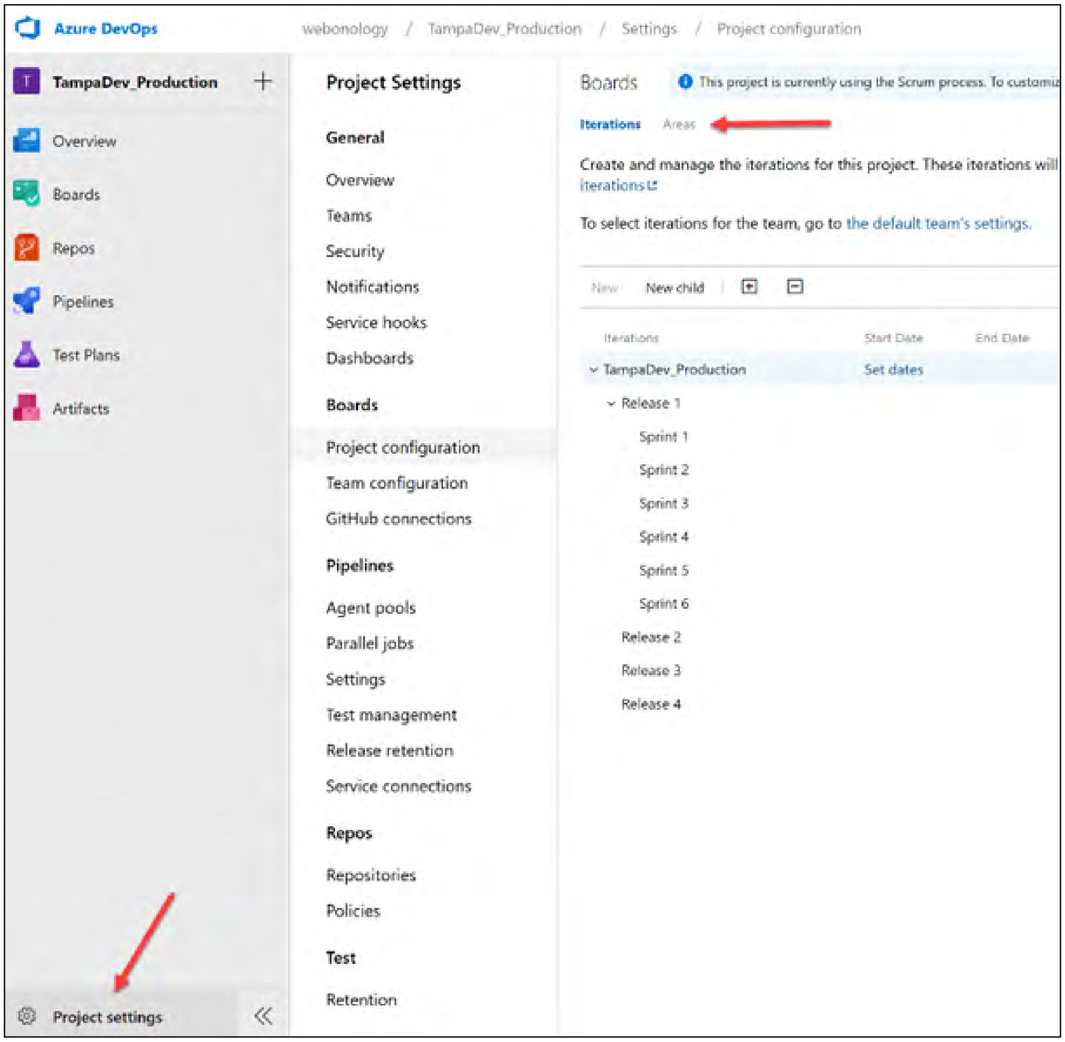

Figura 7: Rutas e iteraciones de Azure DevOps

Las rutas se utilizan para agrupar elementos de trabajo en función de la producción, el área de negocios o las características. Las iteraciones se utilizan para dividir el tiempo de trabajo en segmentos de tiempo de administración, por ejemplo, cada dos semanas en un Scrum. Una vez que complete esto, ingrese los elementos de trabajo pendiente para que su proyecto pueda comenzar.

Me gusta crear una definición de "listo" para cada tarea a fin de garantizar que se complete un parámetro de medición estándar para los elementos de trabajo o las tareas. Si le dice a alguien cómo se mide su trabajo, obtendrá mejores resultados:

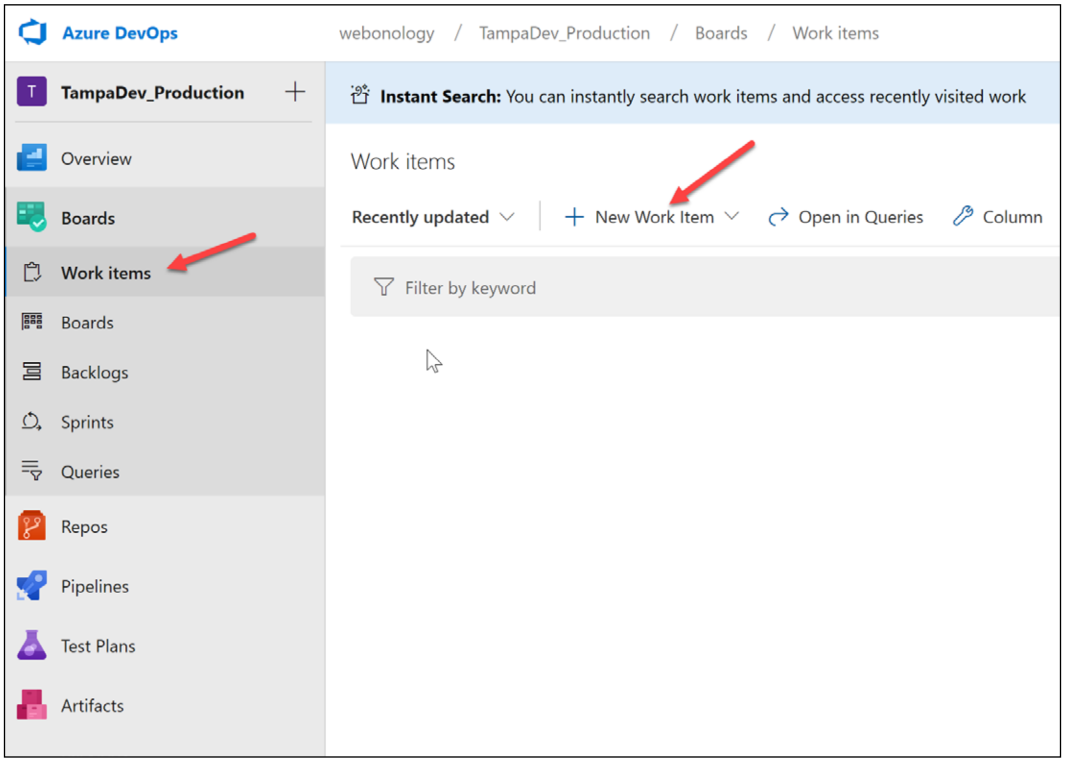

Figura 8: Creación de un elemento de trabajo

Una vez que tiene un trabajo pendiente, puede trabajar con los miembros del equipo para medirlo y dividirlo en partes o tareas. Una vez que lo hace, los miembros del equipo extraerán los elementos del trabajo pendiente en función de su capacidad para ese proyecto. Cada miembro del equipo responsable de un fragmento de código debe comprobar ese código en el elemento de trabajo al cual se destina el código. Por ejemplo, si estoy creando un nuevo elemento de base de datos centrado en el código, verificaría el código para revisar si hay cualquier cambio en el elemento de trabajo que estaba usando para obtener los requisitos.

Siempre creo un área de mejora o de lista de deseos para aquellas cosas que surgen cuando el producto avanza hacia la finalización, que no son parte de los requisitos de un producto mínimo viable. Administrar sus requisitos de trabajo pendiente y seguimiento es realmente lo que funciona para una organización, por lo que investigaría y usaría la plantilla que esté más cerca del estilo de administración que funciona para usted y el proyecto. Recuerde que cada proyecto se puede administrar de manera diferente. Ahora abordaremos la metodología relacionada con la creación y la implementación.

Compilación, implementación y administración

Azure Pipelines maneja las compilaciones e implementaciones del código de su aplicación. Por lo general, le recomiendo comenzar por crear un repositorio de código y, luego, una compilación de CI. Este es un proceso de automatización en que se compila el código y ejecuta una prueba en el código cada vez que un desarrollador realiza una comprobación. Como verá en la *Figura 9*, puede encontrar las compilaciones (**Builds**) y las versiones (**Releases**) en las canalizaciones (**Pipelines**):

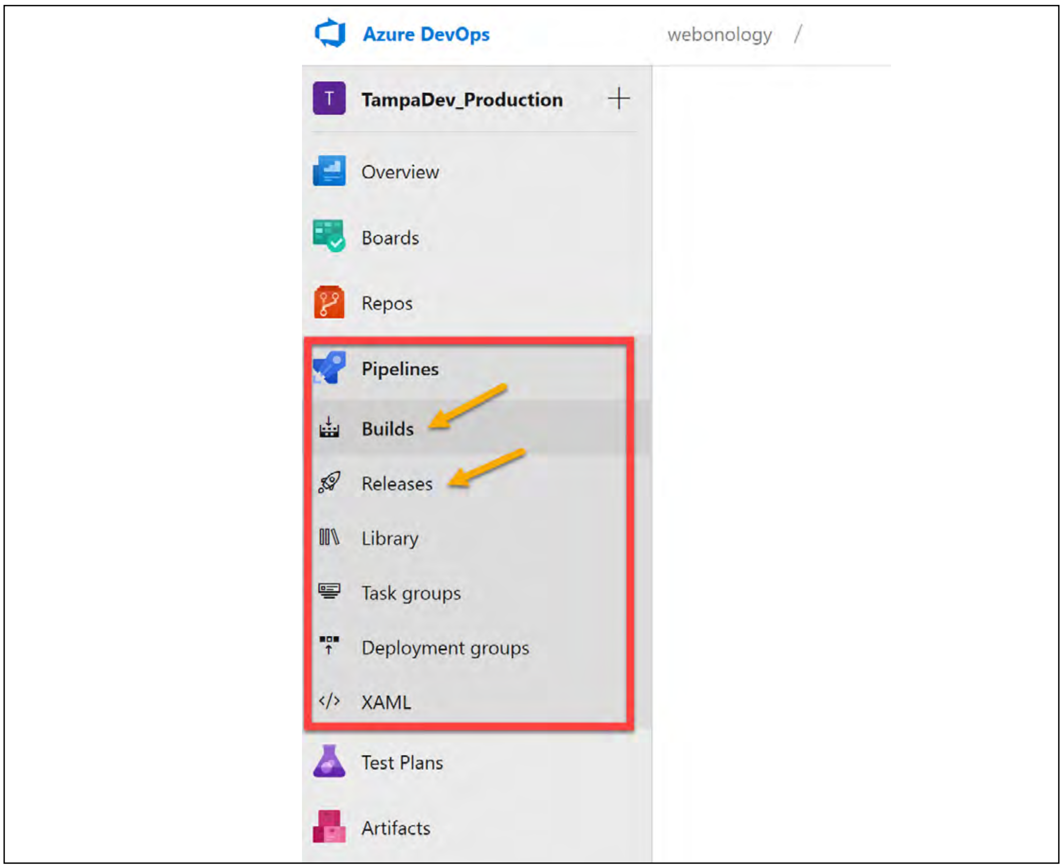

Figura 9: Canalizaciones de compilación y versión

A continuación, puede crear un proceso de versión que use los artefactos de compilación y los implemente en un entorno como la integración de desarrolladores o QA para pruebas de sistema adicionales. En mis procesos de compilación, suelo usar un proceso de NuGet para mis recursos compartidos a fin de ayudar con la reutilización y las implementaciones de cambios de versión. A continuación, profundizaremos en CI/CD.

Uso de CI/CD para el desarrollo de alta productividad

Las canalizaciones de CI y CD ayudan a automatizar los pasos para las compilaciones, las pruebas y las versiones con comentarios rápidos, a la vez que mantienen la aplicación en un estado enviable. Este tipo de proceso ayuda a reducir los errores manuales y a configurar el origen para las configuraciones, puesto que la CI se basa en los cambios y las fusiones de código en un repositorio de código central que no tiene límite en las comprobaciones. Debemos entender de qué forma esto afecta las variaciones de código que otros desarrolladores necesitan fusionar con sus repositorios, lo que normalmente desencadena una compilación y prueba, así como una versión en la mayoría de los casos.

Es conveniente entender las etapas de este proceso, que son origen, compilación, prueba y versión. La etapa de origen es la forma en que los desarrolladores administran la base de código y las inserciones en el repositorio validadas permitidas para las compilaciones. Esas compilaciones son la segunda parte de estas etapas, porque el código debe crearse con nuevos cambios. Luego, ejecutamos pruebas para ayudar con la calidad del código y el cumplimiento de requisitos. La última parte es la versión, que traslada ese código al exterior. La parte de CD del proceso es la entrega: poder entregar una aplicación independientemente del estado del código. Con esto, necesitamos entender algunos procedimientos recomendados de alto nivel.

Implementación de procedimientos recomendados de DevOps

Para implementar los procedimientos recomendados de Azure DevOps, debe recordar lo siguiente:

- Fomente una cultura de DevOps mediante la definición de objetivos compartidos que abarquen a los equipos de desarrollo y operaciones.
- Asegúrese de que sus flujos de trabajo sean flexibles; otorgue al equipo la capacidad de elegir las herramientas que enfaticen su experiencia.
- Adopte más automatización y busque oportunidades para aumentar la automatización a fin de ayudar con la entrega continua.
- Fomente las superficies de proyectos pequeñas mediante el uso de microservicios y contenedores.
- Trate de no personalizar la seguridad ni los grupos de seguridad.
- Cree un proceso de revisión de código sostenible.
- Participe en este proceso temprano y empiece con la automatización desde el principio.
- Tenga continuidad.
- Comuníquese.

Cómo acelerar el proceso de administración del ciclo de vida de la aplicación

Anteriormente en el capítulo, analizamos los pilares de calidad para el desarrollo de software. Aquí examinaremos los pasos a seguir en la creación de la administración del ciclo de vida en Azure DevOps. Hacer que el proyecto sea pequeño y fácil de implementar con su propia plantilla de **Azure Resource Manager** (**ARM**) para los recursos ayudará con la implementación rápida y la independencia de la aplicación. Este es un enfoque:

1. Recopile los requisitos del cliente.
2. Use esos requisitos para crear un trabajo pendiente y cree una definición de "listo".
3. El equipo comienza a dimensionar el trabajo pendiente y dividir las tareas en partes más pequeñas y manejables, además de crear planes de prueba para probar los requisitos.
4. Cree sus interacciones y rutas para que sus sprints se puedan definir como tareas.
5. Use un repositorio de Git y cree un proceso de solicitud de incorporación de cambios para probar la calidad de la comprobación.
6. Active la compilación y versión en el entorno de integración de desarrollo.
7. Pruebe la compilación y verifíquela.
8. Envíe un correo electrónico a QA para obtener la aprobación de la versión del código liberado.
9. Active la versión después de la aprobación y ejecute las pruebas de comprobación de compilación en el entorno de QA.
10. Lance a producción en función del proceso de lanzamiento a un espacio de ensayo, luego, cambie los espacios a un espacio nuevo o ejecute pruebas en la fase de transición a la producción si es necesario.

Entonces, con este conocimiento, examinemos cómo funcionan las etapas y los entornos de Azure DevOps en nuestras canalizaciones de versión.

Explicación de las etapas y los entornos

Azure DevOps tiene el concepto de etapas y entornos, que básicamente son desviaciones dentro de una canalización en función de un elemento, como tareas para completar (etapas) o lugares en los que terminan (entornos). Una etapa es una forma de organizar su canalización en divisiones importantes. Son piezas de la canalización que se pueden pausar y pueden realizar varias comprobaciones. La etapa puede contener varias etapas o etapas paralelas en la finalización de la canalización.

Los entornos son los puntos de conexión de los recursos, es decir, el lugar y el elemento en que se implementan los artefactos. Puede tener varios entornos, además de procesos, como las aprobaciones, antes del lanzamiento. Ahora que tiene una comprensión rápida de las etapas y los entornos, analicemos los servicios de implementación y administración.

Servicios de implementación y administración

En primer lugar, es importante comprender que los recursos en Azure se "implementan". Este es el proceso de aprovisionamiento para Azure, pero normalmente no se le conoce como aprovisionamiento. Con Azure, se produjo una transición desde un enfoque de desarrollo hasta la implementación de recursos, que es una representación JSON del recurso que se está implementando. Esto se conoce como **Infraestructura como código** o **IaC** y se controla a través de un sistema llamado plantillas de **Azure Resource Manager** (**ARM**).

Las plantillas de ARM son el servicio de implementación y administración de Azure. Son la capa de administración que proporciona la capacidad de crear, actualizar y eliminar recursos, además del estado deseado de configuración o DSC del recurso. El sistema de plantillas de ARM se puede utilizar a través del portal, PowerShell, CLI de Azure, API de REST o los SDK de cliente, tal como se muestra en la *Figura 10*:

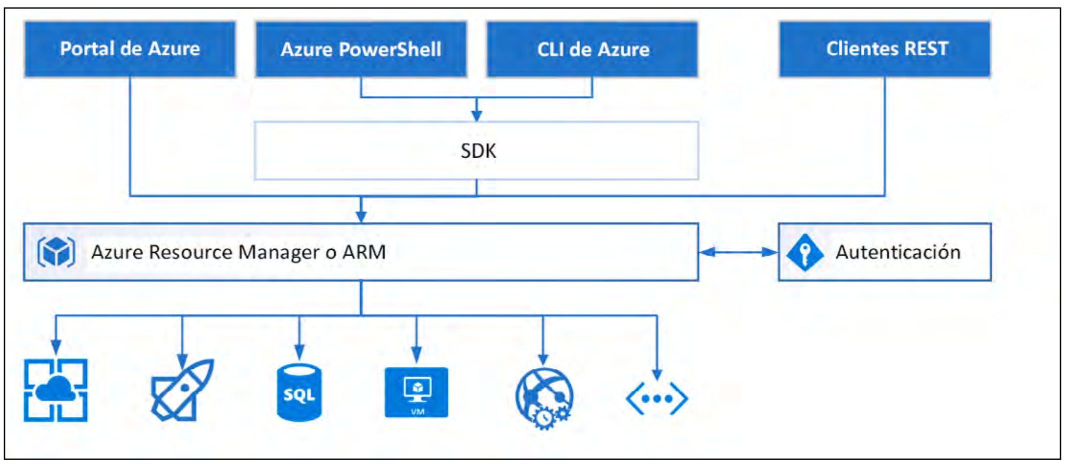

Figura 10: Proceso simplificado de Azure Resource Manager

Como puede ver, el sistema de implementación puede usar una plantilla de ARM o un proceso de PowerShell/CLI para implementar recursos. Es importante entender que el sistema de PowerShell/CLI puede usar plantillas de ARM o implementar recursos sin una plantilla de ARM mediante el uso directo de las API de ARM en Azure. Se recomienda que use las plantillas de ARM tanto como pueda para controlar las implementaciones. Antes de profundizar en las plantillas de ARM o PowerShell/CLI, analicemos la terminología. Lo primero que hay que entender es que todas las implementaciones se producen en un grupo de recursos, que es lo que se utiliza para los límites de la aplicación. Se recomienda que la implementación sea pequeña y fácil de administrar. Los elementos que implementa, como una VM, un servidor Azure SQL (o una base de datos) o una aplicación web, son los recursos que implementa dentro de los grupos de recursos. Cada recurso tiene un proveedor, que se denomina proveedor de recursos, y que le indica a la implementación el tipo de recurso que se proporciona. Dicho esto, profundicemos un poco más.

Comprensión de cómo las plantillas de ARM se usan para implementar artefactos

Antes de comenzar esta sección, asegúrese de que instaló la carga de trabajo de desarrollo de Azure en Visual Studio, porque el proyecto de plantilla de ARM no estará disponible hasta que lo haga. En la barra de búsqueda, escriba `Instalador de Visual Studio`, y se mostrará la pantalla de la *Figura 11*:

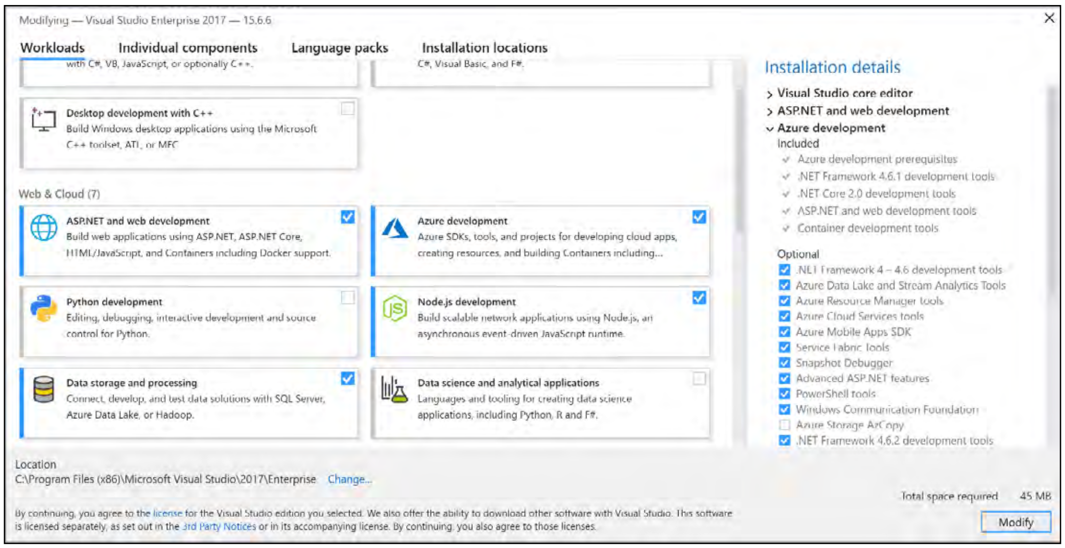

Figura 11: Carga de trabajo de desarrollo de Azure en Visual Studio

Asegúrese de tener la carga de trabajo **Azure development** (desarrollo de Azure) marcada, tal como se muestra en la *Figura 12*:

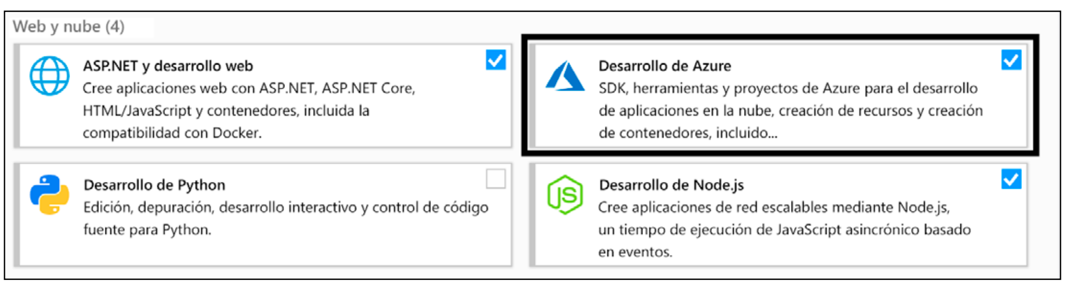

Figura 12: Carga de trabajo de Azure

Una vez que instale la carga de trabajo, podrá crear una plantilla del **Grupo de recursos de Azure** en Visual Studio, como se muestra en la *Figura 13*:

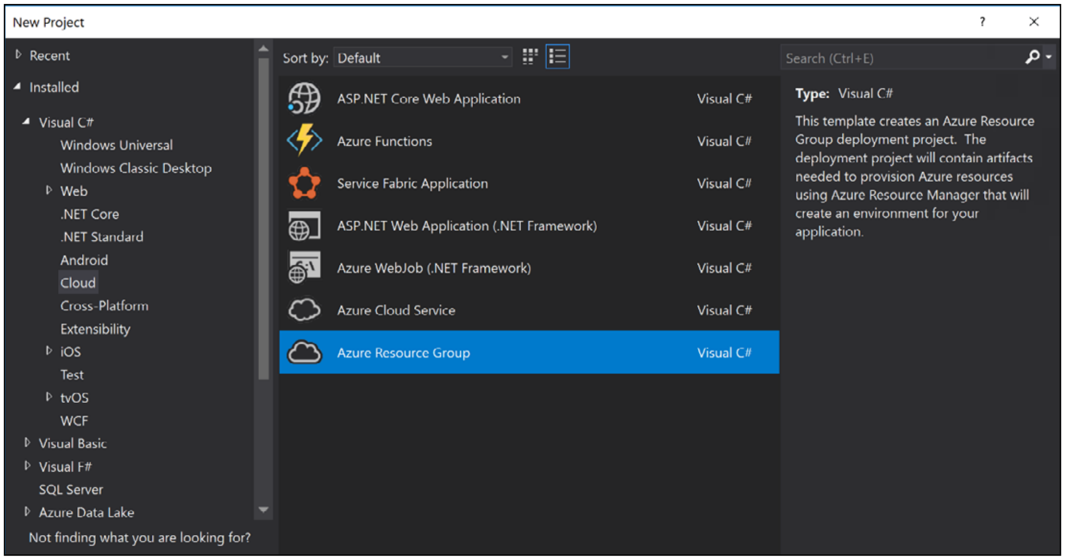

Figura 13: Visual Studio: grupo de recursos de Azure

Ahora que nos aseguramos de que Visual Studio está configurado para usar los grupos de recursos de Azure, podemos profundizar en cómo usarlos. Las plantillas de ARM son los archivos JSON que se utilizan para implementar los recursos en un grupo de recursos. Usan una sintaxis declarativa que utiliza los elementos base que analizamos. Abarcan cuatro niveles dentro de Azure: se pueden usar en proyectos en grupos de administración, contenidos en bibliotecas en suscripciones, contenidos en historiales de implementación (scripts de exportación) en grupos de recursos o directamente en los recursos, como se muestra en la *Figura 14*:

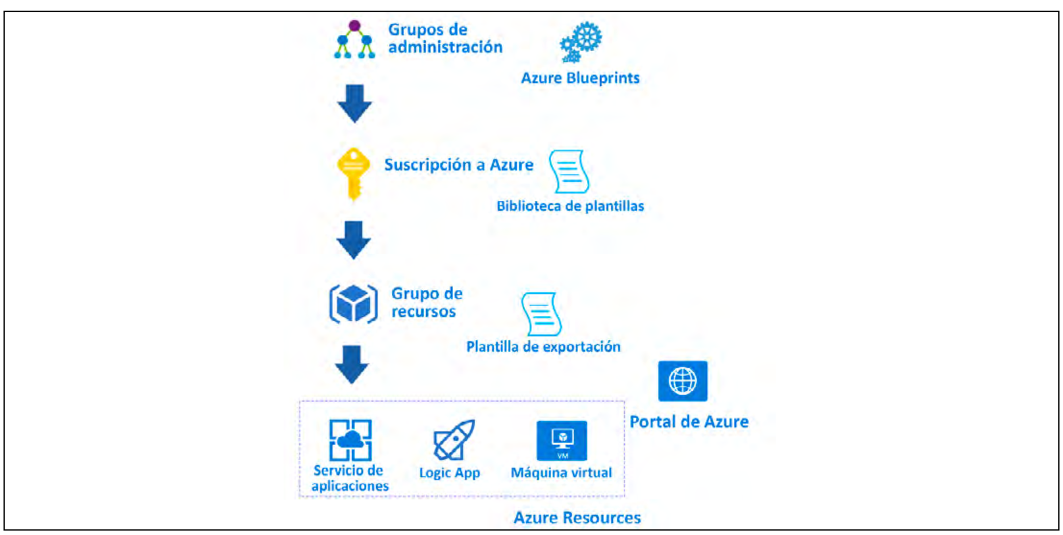

Figura 14: Cuatro niveles de recursos en Azure

Ahora que establecimos un conocimiento básico de cómo se puede usar una plantilla de ARM, examinemos los elementos de la estructura básica del sistema de plantilla que se utiliza para implementar recursos en Azure. Esta es la composición básica del archivo JSON creado para albergar todos los recursos que necesita para su proyecto:

```
{
  "$schema": "http://schema.management.azure.com/schemas/2015-01-
01/deploymentTemplate.json#",    "contentVersion": "",
  "parameters": { },
  "variables": { },
  "functions": { },
  "resources": [ ],
  "outputs": { }
}
```

Veamos lo que significan estos elementos y conozcamos mejor cómo usarlos, además de ver lo que significan para el esquema:

Esquema	**Ubicación del esquema JSON**	**Obligatorio**
contentVersion	Valor de seguimiento de la versión, el predeterminado es 1.0.0.0	Sí
parameters	Los valores que se proporcionan desde un origen externo, como los archivos de parámetro y los parámetros de una canalización de implementación	No
variables	Valores comunes proporcionados para la implementación	No
functions	Funciones definidas por el usuario	No
resources	Tipo de recursos que está implementando	Sí
output	Valores de retorno	No

Nota importante:

ARMVIZ (`http://armviz.io/`) es una herramienta útil para ver sus plantillas de ARM.

Prefiero usar las plantillas de ARM tanto como pueda para implementar mis recursos, pero hay ocasiones en las que no puedo usar ARM y necesito usar algo más, como PowerShell/CLI. También es útil porque puede usar Cloud Shell para ejecutar PowerShell/CLI desde la biblioteca que crea, que me gusta usar porque puedo acceder a Cloud Shell desde cualquier lugar en un navegador. Lo uso para ejecutar runbooks y plantillas de ARM, y para cambiar la seguridad de usuario desde mi teléfono o dispositivo, lo que significa que puedo hacer más cosas sin mi equipo. Examinemos cómo se usa PowerShell para la implementación.

Uso de PowerShell para implementar artefactos

Todos los recursos de Azure se pueden implementar con PowerShell, no solo con plantillas de ARM, y la naturaleza declarativa de la plantilla se utiliza para ejecutar PowerShell a fin de crear los recursos. También hay recursos que no se implementan mediante ARM, como los certificados de Azure App Services o DNS. Estos también se pueden ejecutar en canalizaciones de Azure DevOps, como se muestra en la *Figura 15*:

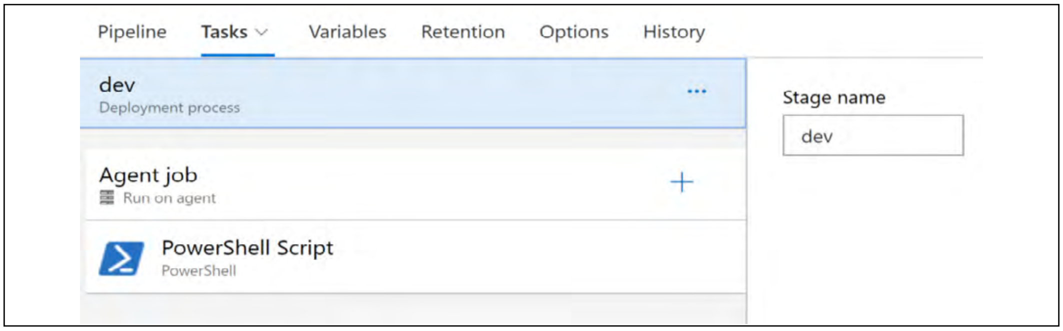

Figure 15: Ejemplo de canalización

Medidas de seguridad y retención poderosas de Azure DevOps para cargas de trabajo

Azure DevOps proporciona seguridad inmediata, así como directivas de retención que se pueden personalizar si es necesario. Trato de mantenerme alejado de demasiada personalización en torno a la seguridad en Azure DevOps, pero revisemos las directivas de seguridad y retención para comprender mejor cómo se aplican de forma predeterminada.

Seguridad

Azure DevOps es una herramienta de entrega segura, disponible y privada para las organizaciones. Los principales aspectos que debe comprender son cómo funcionan la autenticación, la autorización, los grupos de seguridad y los roles, los permisos y los niveles de acceso dentro de la plataforma de Azure DevOps. La autenticación se puede integrar fácilmente a **Azure Active Directory** (**Azure AD**), **Cuenta Microsoft** (**MSA**) o al AD local, con Azure AD y MSA que son soluciones basadas en la nube. Cumplen con la **autenticación de dos factores** (**2FA**), que es un proceso de acceso de dos niveles que utiliza un dispositivo móvil o un código enviado por correo electrónico para acceder a los recursos.

Si bien esto lo llevará a la plataforma, necesitará tener asignada una licencia (básica), ser una parte interesada (una persona de negocios sin registros de código) o un suscriptor de Visual Studio (un usuario de la plataforma de desarrollo) para ver la página de inicio de Azure DevOps. Ahora que inició sesión en la plataforma, veamos cómo funcionan los métodos de autenticación para otros servicios dentro de la plataforma de Azure DevOps.

Una vez que se cree el acceso a la plataforma, necesitará acceso a los repositorios de código si es un desarrollador y los servicios de integración para los recursos de Azure DevOps. Si no desea iniciar sesión cada vez que quiera acceder a los recursos que está utilizando, puede usar uno de los siguientes elementos:

- Tokens de acceso personal, que le permiten acceder a un recurso desde clientes como XCode o NuGet, que realmente no admiten las credenciales de Azure AD.
- OAuth, que permite el acceso a las API de REST de Azure DevOps. Debe señalarse que solo el perfil y las API de cuenta son compatibles con OAuth.
- Autenticación SSH, que es la clave de autenticación para plataformas Linux, macOS o Windows que ejecutan Git para Windows.

Debo señalar que, de forma predeterminada, su colección y la cuenta permiten el acceso a todos los métodos de autenticación enumerados anteriormente. La autorización en Azure DevOps se basa en la asignación individual de usuarios o grupos, que habilita los permisos necesarios para el servicio, la función, el objeto o el método. Azure DevOps tiene algunos grupos de seguridad preconfigurados con los que interactuará, pero examinemos cómo se asignan los permisos a grupos, niveles y estados:

Grupos de seguridad	**Niveles de permiso**	**Estados de permiso**
Se asignan en el nivel de proyecto, de organización o de colección, o en el nivel de servidor para el servidor de Azure DevOps, que es el servidor local.	Estos se asignan en el nivel de objeto, nivel de proyecto, de organización o de colección, o en el nivel de servidor para el servidor de Azure DevOps, que es el servidor local.	Para los estados de permiso, un usuario o un grupo puede tener permiso para **Permitir** o **Permitir heredar** (desde el objeto primario) o **Denegar, Denegar heredar** o **No establecer permisos** en una tarea.

Ahora que tenemos un breve conocimiento de los permisos de configuración, examinemos los grupos integrados y en qué nivel, nivel de servidor, nivel de colección, nivel de proyecto o nivel de objeto, se ubican.

Estos son los grupos de nivel de servidor en Azure DevOps Server (solo en el entorno local):

- Administradores de Team Foundation: estos son como los administradores globales
- Cuenta del servicio de integración de Project Server
- Servicios de aplicaciones web de SharePoint
- Cuenta de servicio de proxy de Team Foundation
- Cuenta de servicio de Team Foundation

Estos son los grupos de nivel de colección:

- Colaboradores
- Administradores de la colección de proyectos, el administrador global de las colecciones
- Administradores de compilación de la colección de proyectos
- Cuentas de servicio de compilación de colecciones de proyectos
- Cuentas de servicio de proxy de la colección de proyectos
- Cuentas del servicio de la colección de proyectos
- Cuentas de servicio de prueba de la colección de proyectos
- Usuarios válidos de la colección de proyectos
- Grupo de servicio de seguridad

Estos son los grupos de nivel de proyecto y de nivel de objeto:

- Administradores de compilación
- Colaboradores
- Lectores, el nivel de acceso más bajo
- Administradores de proyectos, administrador global para el nivel de proyecto o de objeto
- Usuario válido del proyecto
- Administrador de versiones
- `<Team Name>` creado para el equipo en el proyecto de forma predeterminada

Azure DevOps se centra en tres áreas de control de acceso funcional para la administración. Estas son Pertenencia, Permisos y Nivel de acceso. La Pertenencia asigna la cuenta de usuario al grupo predeterminado o a los grupos de seguridad para el acceso al proyecto.

La administración de permisos controla el acceso a tareas funcionales específicas en diferentes niveles del sistema, como el conjunto de permisos de nivel de objeto en una canalización o carpeta. El nivel de acceso está en el nivel del portal y se define como **Básico**, **Parte interesada** o **Visual Studio Enterprise**.

Como puede ver, hay un elemento de seguridad sólido en Azure DevOps.
Puede encontrar estos valores en el portal de Azure DevOps en **Project settings** (Configuración del proyecto) | **General** | **Security** (Seguridad), como se muestra en la *Figura 16*:

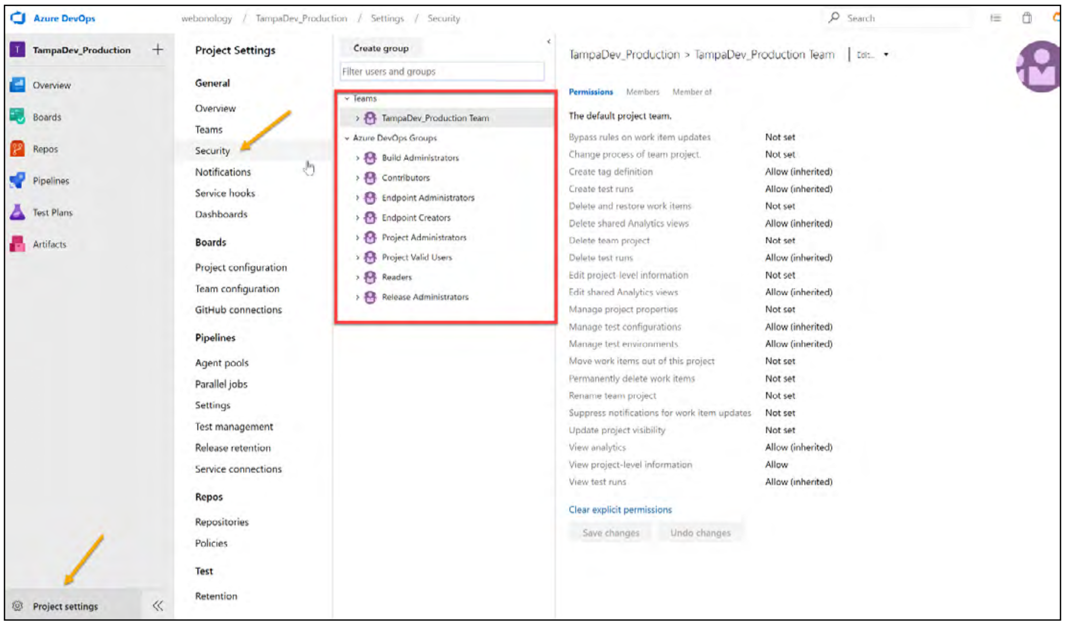

Figura 16: Seguridad del proyecto de Azure DevOps

Ahora que vimos cómo funciona la seguridad de manera inmediata para Azure DevOps, revisemos cómo funciona la retención en sus proyectos.

Retención

Asegúrese siempre de que las directivas de retención estén configuradas correctamente para admitir las necesidades de auditoría de su organización en Azure DevOps.

Estas directivas se aplican a las canalizaciones y los artefactos de prueba para retener el historial de lo que ha sucedido y de los artefactos creados, como se muestra en la *Figura 17*:

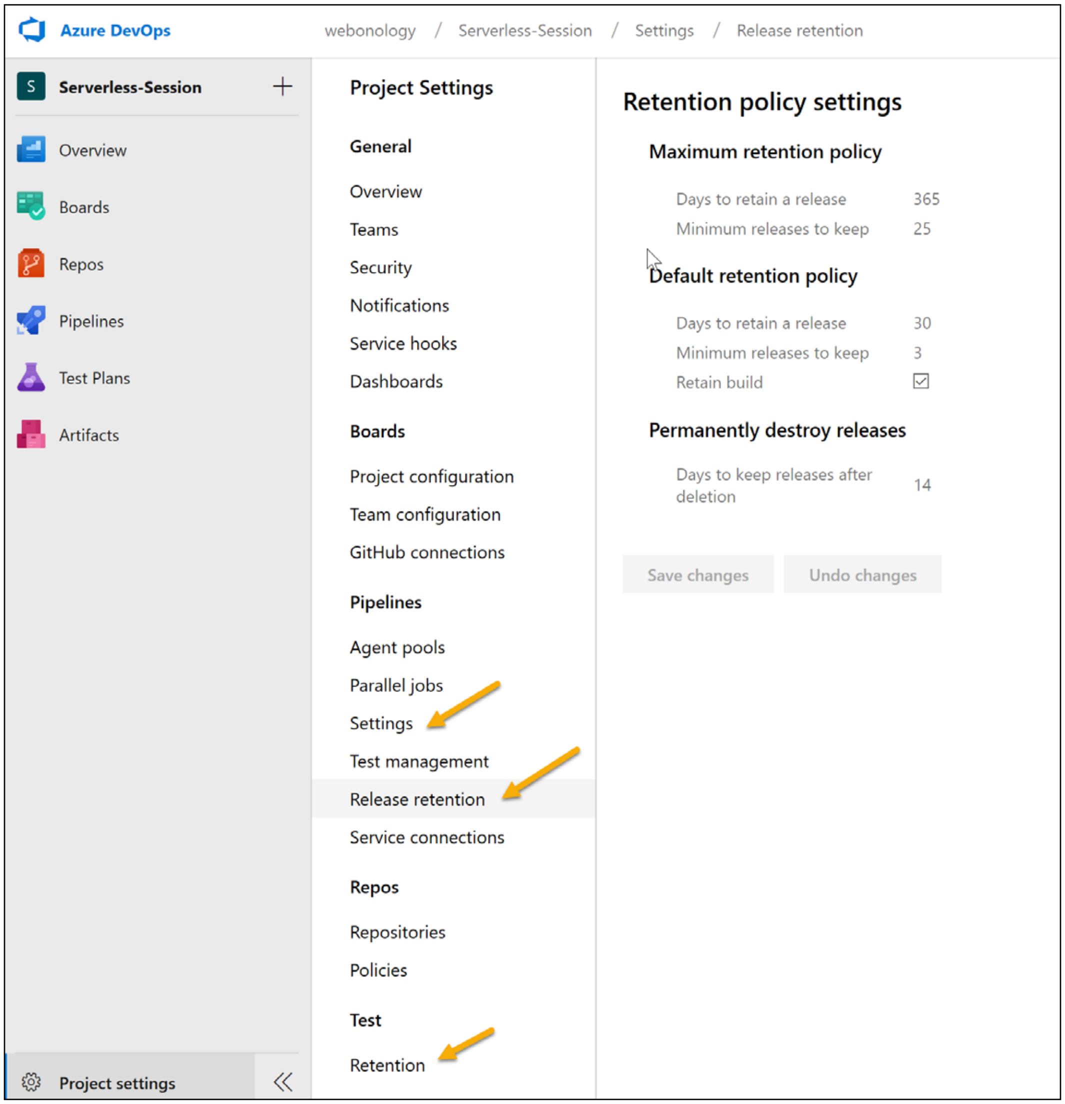

Figura 17: Directivas de retención de Azure DevOps

Resumen

Como puede ver, DevOps y la adopción de la nube van de la mano, puesto que desea automatizar las implementaciones de aplicaciones en la nube tanto como sea posible. Azure DevOps proporciona un excelente conjunto de herramientas de forma inmediata con Azure Boards, Azure Repos, Azure Pipelines, Azure Artifacts y Azure Test Plans. Azure DevOps le permite elegir algunos enfoques estándar para procesar la entrega con marcos como Agile y Scrum. Recuerde que una práctica de DevOps adecuada es responsabilidad de todos y que no se trata de la cantidad de cosas que puede hacer, sino de cómo se hacen las cosas. En el siguiente capítulo, analizaremos los beneficios empresariales que se generan de la migración a la plataforma Azure.

4
Optimización y administración en Azure

Introducción

Las arquitecturas de nube con virtualización crean nuevos desafíos para la administración y la optimización. En lugar de un enfoque tradicional de TI, que consiste en dedicar recursos informáticos específicos a aplicaciones específicas y realizar un aprovisionamiento excesivo para garantizar suficiente potencia de proceso o almacenamiento, la nube comparte recursos dinámicamente. Esto abre la puerta a la flexibilidad, la escalab ilidad y la eficiencia de la infraestructura. Sin embargo, también aumenta la complejidad de supervisar, ajustar, identificar y resolver problemas, y maximizar la rentabilidad.

En este capítulo, analizaremos la administración, primero en recursos solo de Azure y, luego, en entornos híbridos que pueden aprovechar las herramientas de administración de Azure. Continuaremos con un análisis sobre el diagnóstico de problemas de servicio en Azure y cómo obtener soporte. Por último, debido a que la nube y Azure, específicamente, deben cumplir con las promesas de rentabilidad en comparación con otras arquitecturas, terminaremos el capítulo presentando oportunidades de Azure para el ahorro y la optimización de costos.

Administración y optimización de sus recursos de Azure

Cuando elige Azure, abre la puerta a muchas oportunidades diferentes en la nube. **Máquinas virtuales** (**VM**) escalables y de pago por uso, almacenamiento de datos geolocalizado, sitios web, administración de cola e inteligencia artificial son solo algunas de las funcionalidades de Azure. El plano de control de Azure proporciona varias herramientas para ayudarle a aprovechar esto.

Azure Resource Manager

Como componente del panel de control de Azure, Azure Resource Manager es el servicio de implementación y administración de Azure. Le permite crear, actualizar y eliminar recursos en su propio espacio de Azure. Algunos ejemplos de recursos de Azure son las VM, las cuentas de almacenamiento, las aplicaciones web, las bases de datos y las redes virtuales. A medida que los recursos de Azure se multiplican, puede ahorrar tiempo y esfuerzo aplicando acciones de administración en los grupos de recursos, además de los recursos individuales. Estas acciones orientadas a grupos pueden tener varias ventajas:

- Prevención de problemas de seguridad, como conexiones desde varias ubicaciones o desde direcciones IP sospechosas mediante la aplicación de control de acceso a todos los servicios de un grupo.
- Organización lógica de todos los recursos de una suscripción de usuario mediante la aplicación de etiquetas a esos recursos.
- Implementación de recursos en la secuencia correcta mediante la definición de las dependencias entre ellas.
- Explicación de la facturación en el nivel de la organización mediante la visualización de los costos de un grupo de recursos, cada uno con la misma etiqueta.

También puede usar plantillas declarativas que especifiquen lo que desea que ocurra (el resultado) en los recursos, en lugar de deletrear cada paso individual o comando para detallar cómo desea que ocurran esas cosas (el método). Por lo tanto, por ejemplo, puede definir una plantilla declarativa para crear inmediatamente una cuenta de almacenamiento para los discos de VM de forma más rápida, fácil y confiable que intentar enumerar todos los pasos necesarios que realizar y configurar la cuenta.

Azure proporciona cuatro niveles de alcance: grupos de administración, suscripciones, grupos de recursos y recursos. Puede aplicar la configuración de administración en cualquiera de estos niveles de granularidad para determinar qué tan amplio o enfocado es el efecto. Los niveles inferiores heredan la configuración aplicada en los niveles más altos.

Además, las funciones de gobernanza de Azure muestran cómo los proyectos de TI contribuyen a cumplir con los objetivos empresariales. Estas funciones ayudan a su organización a lograr sus objetivos mediante un uso eficaz y eficiente de la TI.

Azure Automation

Azure Automation proporciona un servicio de automatización y configuración basado en la nube para una administración coherente en entornos de Azure y que no son de Azure, como los centros de datos locales. Comprende características de automatización de procesos, administración de actualizaciones y configuración. Azure Automation proporciona un control completo durante la implementación, las operaciones y el retiro de cargas de trabajo y recursos.

Puede utilizar Azure Automation para automatizar las tareas manuales, de ejecución prolongada, propensas a errores y repetitivas que comúnmente se realizan en entornos de nube y empresariales. Esta automatización le permite ahorrar tiempo. También aumenta la confiabilidad de las tareas administrativas normales con la opción de realizarlas automáticamente a intervalos periódicos. Puede automatizar procesos mediante runbooks o automatizar la administración de la configuración mediante **Desired State Configuration** (**DSC**) (consulte la siguiente sección).

Los siguientes son algunos escenarios comunes para la automatización:

- **Crear/implementar recursos**: implemente VM en un entorno híbrido mediante runbooks y plantillas de Azure Resource Manager, con la posible integración en herramientas de desarrollo, como Jenkins y Azure DevOps.
- **Configurar VM**: evalúe y configure las máquinas Windows y Linux según sea necesario para la infraestructura y la aplicación.
- **Supervisar**: identifique los cambios en las máquinas que pueden causar problemas, luego, realice correcciones o escale a los sistemas de administración.
- **Proteger**: ponga en cuarentena una VM si se genera una alerta de seguridad y establezca los requisitos de los invitados.
- **Gobernar**: implemente el control de acceso basado en roles para los equipos y recupere los recursos no utilizados.

Administración de la configuración

Azure Automation DSC es una solución basada en la nube para PowerShell DSC que proporciona servicios destinados a entornos empresariales. Puede administrar sus recursos de DSC en Azure Automation y aplicar configuraciones a máquinas virtuales o físicas de un servidor de extracción de DSC en la nube de Azure. La solución ofrece una variedad de informes, incluidos informes sobre eventos importantes, como nodos que no cumplen las normas. Puede supervisar y actualizar automáticamente la configuración de la máquina a través de máquinas físicas y virtuales, Windows y Linux, en la nube y el entorno local.

La galería de automatización contiene runbooks y módulos que aceleran la integración y la creación de procesos desde la galería de PowerShell y Microsoft Script Center.

Administración del almacenamiento

El Explorador de Azure Storage permite a los usuarios administrar el contenido de sus cuentas de almacenamiento. Con esta aplicación independiente, puede cargar, descargar y administrar blobs, archivos, colas, tablas y entidades de Cosmos DB, además de administrar discos de VM.

Puede usar el Explorador de Azure Storage para trabajar con Azure Resource Manager o las cuentas de almacenamiento clásicas, para trabajar con los datos de Azure Storage en Windows, macOS y Linux, y para administrar y configurar reglas de **uso compartido de recursos entre orígenes** (**CORS**). Otra posibilidad es la administración de Data Lake.

El Explorador de Azure Storage ofrece varias opciones para la conexión a cuentas de almacenamiento, entre las que se incluyen:

- Conexión a cuentas de almacenamiento asociadas con sus suscripciones de Azure
- Conexión a cuentas de almacenamiento y servicios compartidos desde otras suscripciones de Azure
- Conexión y administración del almacenamiento local mediante el emulador de almacenamiento de Azure

Uso del Explorador de Azure Storage con Azure File Storage

Azure File Storage proporciona recursos compartidos de archivos en la nube mediante el protocolo estándar de **Bloque de mensajes del servidor** (**SMB**) (SMB 2.1 y SMB 3.0). El servicio de Azure File Storage permite la migración rápida y rentable a Azure de aplicaciones heredadas que se basan en recursos compartidos de archivos. Con Azure File Storage, puede hacer que los datos estén disponibles públicamente o almacenarlos de forma privada.

El acceso delegado a los recursos en una cuenta de almacenamiento es posible mediante una **firma de acceso compartido** (**SAS**). Con una SAS y sin necesidad de compartir claves de acceso a la cuenta, puede conceder permiso a un cliente para obtener acceso a objetos específicos en su cuenta de almacenamiento durante un período específico.

Azure Data Studio

Azure Data Studio es una herramienta para la administración de bases de datos de SQL Server, incluidos los sistemas de Azure SQL Database y Azure SQL Data Warehouse. Anteriormente llamado SQL Operations Studio, Azure Data Studio ofrece una experiencia de editor actualizada con IntelliSense, fragmentos de código, integración de control de código fuente (Git), una terminal integrada y gráficos integrados de conjuntos de resultados de consulta y paneles personalizables. En Azure, los fragmentos de código pueden generar la sintaxis SQL correcta para crear bases de datos, tablas, vistas, procedimientos almacenados, usuarios, inicios de sesión, roles, y así sucesivamente, y para actualizar los objetos de base de datos existentes.

Extensión de la administración más allá de Azure

A través de herramientas y tecnología para la administración, automatización y gobernanza, Azure ofrece soluciones para que los usuarios creen nuevas aplicaciones y recreen los entornos existentes en la nube con los mismos niveles de gobernanza y cumplimiento normativo que las implementaciones en el entorno local. Además, los usuarios se benefician de la agilidad de la nube en la expansión y reimplementación de recursos de proceso y almacenamiento, mientras que las herramientas de administración de recursos de Azure les ayudan a seguir siendo eficientes, eficaces y a cumplir las normas.

Sin embargo, los usuarios con frecuencia desean combinar Azure y otros entornos en una solución de nube híbrida que se adapte mejor a sus necesidades. En la próxima sección, veremos cómo las herramientas de administración de Azure pueden abarcar o ser aprovechadas por recursos que no son de Azure, incluida la automatización para la supervisión, la actualización y otras tareas.

Trabajo con su estrategia de nube híbrida

Las configuraciones de nube híbrida pueden ofrecer lo mejor de a mbos mundos, combinando elementos como la escalabilidad de la nube y el ahorro de costos con el control y la seguridad locales. Tiene sentido combinar la administración de recursos en la nube y en el entorno local para evitar los silos de administración y el desperdicio potencial, a la vez que se garantiza una protección adecuada. Puede lograr una productividad y eficacia continuas a través de soluciones resistentes para la administración centralizada, que abarcan sistemas en la nube y locales. Los servidores Windows locales y en la nube de Azure se pueden administrar de esta manera mediante herramientas como Windows Admin Center y Hybrid Runbook Worker.

Uso de servicios de administración locales e híbridos con Windows Admin Center

Windows Admin Center es un conjunto de herramientas de administración basada en navegador para servidores de Windows locales, con acceso a los servicios de Azure. Si bien se puede usar para administrar servidores Windows en redes privadas que no están conectadas a Internet, también tiene varios puntos de integración con los servicios de Azure, como Azure Active Directory, Azure Backup y Azure Site Recovery. Windows Admin Center simplifica y amplía la administración de sistemas Windows en entornos locales y de Azure. También administra diferentes generaciones de sistemas Windows Server y Windows 10 y más, a través del gateway de Windows Admin Center instalado en Windows Server o Windows 10.

Windows Admin Center contiene muchas herramientas con las que los usuarios que administran servidores y clientes de Windows ya deben estar familiarizados. Funciona con soluciones como administración y seguridad de Azure y System Center para realizar acciones de administración de una sola máquina. El gateway puede estar disponible a través de un firewall de la empresa para la administración segura de estos recursos en Internet mediante Microsoft Edge o Google Chrome. El control de acceso basado en roles proporciona un control detallado sobre a qué características de administración pueden acceder los administradores. Los grupos locales y Active Directory basado en el dominio local son opciones para la autenticación del gateway, al igual que Azure Active Directory basado en la nube.

Los servicios híbridos de Azure están disponibles para los servidores de Windows como servidores físicos y VM individuales, además de clústeres. Las implementaciones locales de Windows Server pueden beneficiarse de los servicios en la nube q ue incluyen los siguientes:

- **Azure Site Recovery** para la protección de VM con recuperación ante desastres basada en la nube. Las cargas de trabajo que se ejecutan en VM se pueden replicar para proteger las operaciones críticas para el negocio si ocurre un desastre. Windows Admin Center facilita la configuración y replicación de VM en servidores y clústeres de Hyper-V para aumentar la resistencia a través del servicio de recuperación ante desastres de Azure Site Recovery.
- **Azure Monitor** con análisis avanzado y "machine learning" para hacer seguimiento de eventos en aplicaciones, infraestructura y redes. Con Azure Monitor, los usuarios pueden supervisar el estado, los eventos y el rendimiento del servidor; configurar alertas de correo electrónico; y ver aplicaciones, servicios y sistemas conectados a cualquier servidor.
- **Adaptador de red de Azure** para una conectividad fácil a la red de Azure. Los servidores locales pueden conectarse de forma segura a una red virtual de Azure a través de un Adaptador de red de Azure.
- **Azure Update Management** para mantener actualizadas las VM. Esta herramienta permite la administración de actualizaciones y revisiones para diferentes servidores y VM desde una ubicación, ya sea que estos servidores o VM sean locales, estén en Azure o los hospeden otros proveedores de servicios en la nube. Los usuarios pueden comprobar rápidamente las actualizaciones disponibles, planificar las instalaciones de actualizaciones y revisar y verificar que la instalación de las actualizaciones se realizó de forma correcta.
- **Azure Backup** para realizar copias de seguridad de Windows Server. Para protegerse de los daños provocados por accidentes, corrupción y ataques de datos, los usuarios pueden realizar copias de seguridad de sus máquinas y VM de Windows Server en Azure.
- **Azure File Sync** para sincronizar servidores de archivos locales con la nube. La sincronización de archivos puede evitar la necesidad de realizar una copia de seguridad del servidor local. Los archivos también se pueden sincronizar en varios servidores mediante la sincronización de varios sitios.
- **Azure Active Directory** (**Azure AD**) autenticación para una capa de seguridad adicional para Windows Admin Center. La autenticación de Azure AD también permite el acceso a las características de seguridad de Azure AD, como el acceso condicional y la autenticación multifactor.

Estos servicios ofrecen las ventajas adicionales de la configuración simple y una vista centrada en el servidor para los administradores, que se integran directamente en Windows Admin Center. La herramienta de servicios híbridos de Azure en Windows Admin Center reúne los servicios de Azure integrados en una plataforma centralizada para facilitar la detección de servicios tanto en entornos locales como híbridos.

Cuando los usuarios se conectan con la herramienta de servicios híbridos de Azure a un servidor que ya tiene habilitados los servicios de Azure, se benefician de una experiencia de administración centralizada para todos los servicios habilitados en un servidor. Los usuarios pueden acceder rápidamente a la herramienta necesaria en el conjunto de herramientas de Windows Admin Center, conectarse al portal de Azure para una administración más extensa de esos servicios de Azure y consultar la documentación en línea.

Windows Admin Center también permite la administración de VM de Azure, no solo en servidores locales. Los usuarios pueden administrar las VM en Azure mediante la conexión de su gateway de Windows Admin Center a la VNet de Azure. A continuación, pueden usar las herramientas simplificadas de Windows Admin Center.

Si bien Windows Admin Center satisface muchas necesidades comunes, no está diseñado para reemplazar todas las herramientas heredadas de la **Microsoft Management Console** (**MMC**). Por ejemplo, Windows Admin Center es complementario a las **Herramientas de administración remota del servidor** (**RSAT**), al menos hasta que todas las funcionalidades de administración de RSAT aparezcan en Windows Admin Center. Asimismo, **Windows Admin Center** y **System Center Virtual Machine Manager** (**SCVMM**) son complementarios. Aunque Windows Admin Center puede reemplazar los complementos de MMC y recrear una experiencia de administración de servidor comparable, no está destinada a reemplazar las funcionalidades de supervisión de SCVMM.

Automatización de recursos locales y en la nube mediante Hybrid Runbook Worker

Con Azure Automation, los usuarios pueden automatizar las tareas manuales y repetitivas mediante runbooks. Con Windows PowerShell o Windows PowerShell Workflow, los usuarios pueden programar e implementar la lógica de automatización que elijan en estos runbooks.

Sin embargo, es posible que los runbooks que se ejecutan en la plataforma de nube de Azure no tengan acceso a los recursos que están en el entorno local o en otras nubes. A fin de ampliar el alcance de los runbooks, Azure Automation proporciona la característica Hybrid Runbook Worker para ejecutar runbooks directamente en sistemas que administran recursos locales u otros recursos que no son de Azure. Los runbooks se almacenan y administran en Azure Automation, y se entregan posteriormente a los sistemas designados, que se conocen como Hybrid Runbook Workers.

La estructura de los runbooks en Azure Automation y en Hybrid Runbook Workers es la misma. Lo que diferencia a un tipo de runbook de otro es que los runbooks de Azure Automation administran los recursos en la nube de Azure, mientras que los runbooks de Hybrid Runbook Worker administran los recursos locales para el Hybrid Runbook Worker o en el entorno local para que los recursos se automaticen.

A menudo, la instalación de un runbook se realiza de forma más sencilla y confiable a través de la automatización de la configuración de un sistema Windows. También es posible realizar la instalación y configuración de forma manual. Los usuarios con máquinas Linux pueden ejecutar un script de Python para instalar un agente de runbook en sus sistemas.

En Azure Automation, la opción **RunOn** permite especificar un Grupo de Hybrid Runtime Worker. A continuación, uno de los miembros de este grupo recupera e implementa el runbook en cuestión. Si no se utiliza la opción RunOn, el runbook simplemente se ejecuta en Azure Automation.

Debido a que los runbooks de un Hybrid Runbook Worker acceden a recursos fuera de Azure, su autenticación es distinta de la de los runbooks que se ejecutan y autentican en recursos de Azure. Los runbooks fuera de Azure pueden proporcionar su propia autenticación a los recursos locales. Cuando se ejecutan en un sistema local de Windows, normalmente lo harán en el contexto de la cuenta del sistema local. Cuando el sistema local se basa en Linux, se utiliza la cuenta de usuario especial nxautomation.

Los runbooks de un Hybrid Runbook Worker pueden usar identidades administradas cuando se configura la autenticación a los recursos de Azure. Como alternativa, los usuarios pueden especificar una cuenta **RunAs** a fin de crear un contexto para todos los runbooks. Sin embargo, el uso de identidades administradas para los recursos de Azure ofrece algunas ventajas sobre las cuentas RunAs. Los usuarios no necesitan exportar ni renovar un certificado de RunAs que, luego, se debe importar a Hybrid Runbook Worker. Tampoco es necesario que escriban su código de runbook para controlar el objeto de conexión de runbook.

Hybrid Runbook Worker para actualizaciones y supervisión

Al habilitar la solución Update Management, puede configurar automáticamente un equipo Windows a fin de que admita runbooks para la administración de actualizaciones. Esto se aplica a los equipos Windows conectados al espacio de Azure Log Analytics. Así, estos equipos se transforman en Hybrid Runbook Workers. Esto le permite extender su infraestructura existente y aplicar actualizaciones desde una ubicación confiable, segura y centralizada, lo que crea una infraestructura de nube híbrida/local más integrada. Por otro lado, el equipo de Windows en cuestión no se registra automáticamente con ningún Grupo de Hybrid Worker existente en la cuenta de automatización del usuario.

Se puede instalar Microsoft Monitoring Agent para conectar equipos con registros de Azure Monitor. Cuando el agente se instale en un equipo local y se conecte al espacio de trabajo del usuario, descargará automáticamente los componentes relevantes de Hybrid Runbook Worker. Estos componentes incluyen el módulo de PowerShell `HybridRegistration`, que a su vez contiene el cmdlet `Add-HybridRunbookWorker`. Cuando se ejecuta este cmdlet, se instala el entorno de runbook en el equipo y lo registra con Azure Automation.

Desarrollo y ampliación de las posibilidades de administración de la nube híbrida

La automatización de los procesos en un entorno local o en un entorno de nube que no sea de Azure se puede copiar o modelar en la implementación correcta de Runbook Worker en Azure. Los Hybrid Runbook Workers resultantes solo están restringidos por los límites de los recursos en el propio Hybrid Runbook Worker. Por lo tanto, los Hybrid Runbook Workers están libres de ciertos límites que se imponen a los sandboxes de Azure, como el espacio en disco, la memoria, los sockets de red o el tiempo de ejecución.

Windows Admin Center también se diseñó para ser extensible. Microsoft pone a disposición un **kit de desarrollo de software** (**SDK**) para que los desarrolladores de Microsoft y de terceros creen sus propias herramientas y soluciones de Windows Admin Center y se basen en lo que está disponible actualmente.

¿Qué pasa si algo sale mal?

Cuando las empresas u otras cargas de trabajo clave utilizan Azure, los usuarios deben saber que sus recursos de Azure están disponibles y funcionan correctamente. Por el contrario, se les debe avisar si hay un problema. Deben estar disponibles los datos para verificar que se respetan los **acuerdos de nivel de servicio** (**SLA**). El mantenimiento programado de los recursos de Azure por parte de Azure debe comunicarse e integrarse en la planificación de los clientes.

En la próxima sección, consideramos la amplia variedad de herramientas de soporte disponibles para Azure destinada a recursos globales, personalizados e individuales. También se analizan el estado de los recursos, el estado del servicio, las alertas y la integración con Azure Monitor (descrita anteriormente en este capítulo), así como los aspectos prácticos de cómo obtener soporte directo de Microsoft.

En la siguiente sección de este capítulo, se aborda la optimización de los presupuestos y los ahorros de costos, incluido el análisis y la administración de costos, y las soluciones de Azure para ayudarlo a que su gasto en la nube se aproveche mejor.

Ahorro de costos de Azure: visibilidad, responsabilidad y optimización

A menudo, las empresas consideran los servicios en la nube como una forma de reducir los costos de TI. Sin embargo, cualquier ahorro de costos también dependerá de la forma en que las empresas administren sus costos y optimicen su gasto en la nube. Al igual que con las actividades empresariales, se necesitará la colaboración de diferentes departamentos para lograr una administración eficaz de los costos de la nube. Es probable que los departamentos de TI, finanzas y diferentes niveles de administración participen en el análisis correcto de los costos, su control y la preparación de presupuestos futuros según sea necesario.

Azure ofrece una gama de diferentes herramientas para ayudar a las empresas a administrar sus costos. También es importante tener un enfoque empresarial pragmático y sentido común. Los departamentos de Finanzas deben entender dónde se generan los costos de la nube y cuáles son las tendencias del gasto en la nube, pero también deberían hacerlo los equipos de TI en la nube.

1. **Visibilidad**

   El análisis de costos puede ayudar a los usuarios y las partes interesadas a explorar y desglosar los costos de la nube. La agregación de costos puede mostrarles dónde se ocupa la mayor cantidad de fondos y qué esperar en el futuro. La información sobre los costos acumulados en el tiempo puede permitirles realizar un seguimiento de los costos por mes, trimestre o año en comparación con los presupuestos.

2. **Responsabilidad**

   Identificar entidades específicas que deben financiar el uso de recursos determinados fomenta la eficacia y la rentabilidad, además de ayudar a evitar sorpresas no deseadas en la facturación.

3. **Optimización**

   El análisis de costos puede ayudar a las organizaciones a planificar un mejor uso de los recursos (dimensionar correctamente), resaltar los recursos desperdiciados y mejorar los planes para las estimaciones de costos.

Azure Cost Management

Azure Cost Management lo ayuda a planificar el consumo de recursos en la nube, a la vez que presta atención a los costos. Le permite analizar los costos de manera eficaz y optimizar el gasto en la nube. Con Azure Cost Management, puede ver las tendencias y los patrones que se utilizan y los costos para su organización, con funciones de análisis para explorar estos datos. Esta herramienta pone a su disposición informes sobre los costos, basados en el uso de los servicios y ofertas de Azure y ofertas de Marketplace de terceros.

Los datos de costos de los informes consideran los precios actuales de la organización y otros descuentos de Azure. Los informes también lo ayudan a evaluar posibles anomalías de gastos, mediante grupos de administración, presupuestos y recomendaciones de Azure para ayudarlo a ver posibles oportunidades de reducción de costos. Puede usar el análisis predictivo de Azure Cost Management para administrar o planificar los costos en el futuro. Además, el portal de Azure y las API de Azure le permiten exportar e integrar automáticamente los datos de costos con otros sistemas y procesos con informes periódicos.

Cómo comenzar a optimizar su inversión en la nube

Las herramientas de Azure y un enfoque metódico de la administración de costos pueden ayudarlo a optimizar su gasto en la nube. Aunque Azure ya facilita la construcción y la implementación de soluciones en la nube, sigue siendo importante asegurarse de que esas soluciones estén ajustadas para lograr la rentabilidad.

Una buena administración de costos comienza antes de que se utilicen los presupuestos. Depende de contar con herramientas adecuadas, asignar responsabilidad por los costos y optimizar los gastos. Estos principios deben entenderse y pueden aplicarse por lo menos en tres grupos.

- **Los equipos de TI** que administran los recursos de la nube diariamente deben ajustar sus actividades de manera oportuna a fin de generar el mayor valor para el negocio con los presupuestos que usan para esos recursos en la nube.
- El **departamento de finanzas** debe comparar solicitudes de presupuesto con pronósticos y objetivos financieros de gastos de la nube.
- Por último, los **gerentes** deben asegurarse de que el gasto en la nube y los resultados estén en línea con los objetivos de negocio de la organización.

Uso de ámbitos para la administración de costos de Azure

La mayoría de los recursos de Azure se implementan en grupos de recursos, que forman parte de las suscripciones. Los usuarios autorizados de Azure ven y administran los aspectos de costos de los recursos de Azure a través de los ámbitos, que son nodos de la jerarquía de recursos de Azure. Para la administración de costos, Microsoft define dos roles:

- Administración de datos de facturación (por ejemplo, facturas y pagos)
- Administración de servicios en la nube (por ejemplo, gobernanza de directivas y costos)

La administración de recursos de Azure también se realiza a través de ámbitos, pero utiliza el **Control de acceso basado en roles de Azure** (**RBAC**). Los dos tipos de ámbitos se denominan ámbitos de facturación y ámbitos de RBAC para diferenciarlos. Los ámbitos de RBAC no difieren según el tipo de suscripción de Azure, pero los ámbitos de facturación pueden variar para extenderse desde grupos de recursos individuales hasta cuentas de facturación completas.

Los siguientes ámbitos de Azure se aplican por suscripción para la administración de costos por usuario y grupo:

- **Propietario**: autorizado para crear, cambiar o eliminar presupuestos de suscripción
- **Colaborador y colaborador de administración de costos**: autorizado para crear, cambiar o eliminar sus propios presupuestos, y cambiar el monto del presupuesto para los presupuestos de otros usuarios
- **Lector y lector de administración de costos**: autorizado para ver presupuestos específicos

Ciclo de vida de la administración de costos

La administración de costos es un proceso iterativo que abarca las cuatro actividades que se describen a continuación, con varias etapas que forman un bucle o un círculo virtuoso. Todas las personas o los equipos involucrados en la administración de costos de la nube deben conocer y aplicar este ciclo de vida.

Planificación

Como dice el refrán, "los planes son inútiles, pero la planificación lo es todo". Cualquier plan para usar los recursos de la nube a fin de cumplir con los objetivos del negocio es solo una aproximación, porque las circunstancias pueden cambiar rápidamente. Sin embargo, mediante la consideración continua de los cambios en los objetivos y la situación de la empresa, los planes se pueden actualizar con tanta frecuencia como sea necesario para seguir siendo realistas. Las preguntas clave que se pueden formular con regularidad y frecuencia son:

- ¿Qué objetivos y desafíos empresariales debe satisfacer mi empresa?
- ¿Cuál es la probabilidad de que el uso de la nube evolucione si se cumplen esos objetivos y desafíos?

Las respuestas a estas preguntas lo ayudarán a identificar los recursos y la infraestructura de Azure que mejor se adaptan a su empresa.

Supervisión

A medida que implementa su plan, necesita saber cuánto gasta su empresa y en qué recursos de la nube. Los recursos infrautilizados deben utilizarse mejor o cancelarse, se debe eliminar el desperdicio y se deben maximizar las oportunidades de ahorrar dinero sin afectar los objetivos empresariales.

Responsabilidad

La rendición de cuentas financieras es la responsabilidad de la manera en que se usa y administra el presupuesto. Esta responsabilidad debe asignarse de forma clara y precisa. Los costos incurridos pueden atribuirse a proyectos o departamentos específicos. La eficiencia de los gastos puede controlarse de forma eficaz.

El gasto excesivo se puede identificar hasta el nivel del proyecto o los recursos, y se deben tomar las medidas apropiadas para volver a alinear los costos o para justificar nuevos presupuestos correspondientes al valor ganado para la empresa.

Optimización

La optimización de los gastos se puede lograr de dos formas. La primera es mediante la comparación del rendimiento, el logro de los objetivos u otros resultados relevantes con los desembolsos financieros. Cuanto menor sea el desembolso para un resultado concreto, mejor se optimiza el gasto. La segunda forma es aprovechar la optimización de compras y licencias y los cambios en la infraestructura (ver más adelante).

Análisis y administración de sus costos

Una vez que se implementa una solución de Azure, es importante saber cómo varían los costos en el tiempo.

Organización y etiquetado de los recursos

Las etiquetas son una solución eficaz para la responsabilidad financiera. Le permiten atribuir costos a proyectos o equipos específicos, y agrupar costos para análisis adicionales. Como alternativa, los clientes del Contrato Enterprise pueden definir suscripciones independientes para departamentos o proyectos, lo que también ayuda a promover la responsabilidad y los esfuerzos individuales para reducir los costos. Las suscripciones y los grupos de recursos pueden ser formas útiles de organizar y atribuir costos a diferentes partes de su organización.

Análisis del costo de uso

El análisis periódico de los costos en comparación con el uso puede ser útil para detectar las tendencias de uso y supervisar la evolución de los costos para proyectos y equipos específicos. Los puntos clave pueden incluir los siguientes:

- Costos estimados para el mes y comparación con el presupuesto correspondiente
- Identificación de anomalías en el gasto, que muestre los costos que se encuentran fuera de un rango razonable y cualquier otro costo excepcional
- Conciliación de facturas para resaltar cualquier aumento inesperado de los costos o cambios en las tendencias de gasto
- Devolución de cargo a los consumidores internos con un desglose de los cargos por proyecto, departamento u otra entidad

Los datos de facturación se pueden exportar automáticamente a otras aplicaciones, como un sistema financiero o un panel de visualización de datos. En lugar de recuperar manualmente los archivos, los usuarios pueden configurar exportaciones automatizadas a Azure Storage con integraciones automáticas de los datos de Azure Storage en otros sistemas.

Creación de presupuestos

Buenas estimaciones, análisis de patrones de gasto y pronósticos son los ingredientes de una creación de presupuesto eficaz. Con Azure Budgets, puede definir los presupuestos según los costos o el uso con una amplia gama de límites y alertas. Se puede desencadenar automáticamente una acción cuando se alcanza un umbral de presupuesto específico. Por ejemplo, las VM se pueden apagar o la infraestructura se puede cambiar a un nivel de precios diferente. A medida que se utilizan los presupuestos (consumo presupuestario), se pueden revisar los datos y realizar cambios según sea necesario.

Optimización de costos

La optimización de los costos proviene de maximizar la eficiencia de los recursos y eliminar aquellos que no generan valor para su empresa. Esto incluye recursos que se implementaron para un proyecto de una duración fija y que no se han anulado ni cancelado después de la finalización del proyecto. Por ejemplo, una simulación empresarial o prueba de sistema que se ejecuta durante un fin de semana puede requerir una cantidad considerable de recursos de proceso y almacenamiento. Sin embargo, un equipo de prueba puede suponer que el equipo de operaciones ajustará los niveles de recursos el lunes siguiente, mientras que es probable que el equipo de operaciones no sepa del ejercicio del fin de semana. Azure Cost Management puede ayudar a rectificar esas situaciones.

Azure Advisor

Este servicio ofrece diferentes funcionalidades, incluida la identificación de las VM que utilizan un bajo nivel de recursos de CPU o de red. Cuando sabe qué máquinas son estas, puede detenerlas o cambiar su tamaño de acuerdo con la previsión de costos para mantenerlas en funcionamiento. Si las compras de instancias reservadas pueden ayudarlo a reducir los costos, Azure Advisor puede proporcionar recomendaciones para esas compras en función de los 30 días anteriores de uso de la VM.

Tamaño adecuado de la VM

Es importante seleccionar el tamaño correcto de las VM para sus cargas de trabajo en la nube. El tamaño de la VM es un factor importante para determinar el costo general de Azure. El número de VM que se requieren en Azure también puede ser diferente al número implementado en un centro de datos local. Por lo tanto, el tamaño individual de la VM y la cantidad total deben calcularse a fin de que correspondan a los requisitos de proceso para que las cargas de trabajo se ejecuten en Azure.

Descuentos de Azure

Los descuentos por volumen, ya sea en términos de cantidad o uso, son una característica común de los acuerdos comerciales. Azure adopta el mismo enfoque, lo que ofrece ahorros de costos a los clientes según corresponda.

Reservas de Azure

Reciba un descuento en sus servicios de Azure mediante la compra de Reservas de Azure. El ahorro de costos puede ser significativo en comparación con los precios de pago por uso para las VM, los recursos informáticos de bases de datos de SQL y los servicios adicionales de Azure. Puede mejorar el presupuesto con un solo pago por adelantado, lo que facilita el cálculo de sus inversiones o puede reducir los egresos de efectivo por anticipado con opciones de pago mensuales sin costo adicional. Puede comprar Reservas de Azure por un año o tres años.

Comprar una Reserva de Azure puede ser la opción más rentable para los clientes con VM, Azure Cosmos DB o bases de datos de SQL que están en funcionamiento durante períodos prolongados. Por ejemplo, sin una reserva y con una necesidad de cinco instancias de un servicio, un cliente pagará tarifas estándar de pago por uso. Sin embargo, si compra una reserva para esos recursos, el cliente se beneficia de inmediato del descuento de reserva, con lo que ahorra dinero en comparación con las tarifas de pago por uso.

Beneficio híbrido de Azure

El programa de Beneficio híbrido de Azure ofrece ahorros de costos si ya tiene implementaciones locales con licencias de Windows Server o SQL Server. El beneficio de Windows Server significa que cada licencia incluye el sistema operativo para hasta dos VM. Luego, los usuarios pagan solo los costos de proceso, y la tarifa de proceso básico es igual a la tarifa de Linux para las VM. Del mismo modo, una licencia existente de SQL Server puede aportar ahorros significativos en las opciones de base de datos de SQL basadas en vCore, como SQL Server en las VM de Azure y SQL Server Integration Services.

Azure Reserved VM Instances

Con **Azure Reserved VM Instances** (**RI**), puede reservar VM de forma anticipada para obtener ahorros de costos cuando se combinan Azure RI y el Beneficio híbrido de Azure. Otras ventajas de RI incluyen el intercambio o la cancelación de reservas y la capacidad de proceso priorizada en las regiones de Azure.

RI también proporciona una característica denominada flexibilidad de tamaño de instancia, que aplica automáticamente los ahorros de RI a cualquier VM que use dentro de la misma región y dentro del mismo grupo de VM de Azure RI. La flexibilidad del tamaño de instancia le permite satisfacer las necesidades cambiantes y conocer los ahorros de costos aplicables sin tener que limitarse a un tamaño de VM específico.

La administración automatizada de RI significa que Azure puede aplicar RI automáticamente a otros tamaños de VM en el mismo grupo y región. Ventajas como estas se aplican tanto a VM de Windows como de Linux en Azure.

Cómo aprovechar los beneficios de la administración de costos

Cuando los costos de Azure se administran correctamente, se pueden respetar los límites y objetivos de presupuesto. La responsabilidad financiera puede garantizarse a través de datos de costos correctos y oportunos, que permitan la comparación con los objetivos financieros. La información obtenida a partir de esos datos puede ayudar a identificar la falta de rentabilidad, por ejemplo, recursos infrautilizados. Puede ayudar a identificar opciones para mejorar la eficiencia o cambios para la optimización de costos. Azure permite procesos sólidos de administración de costos con recomendaciones, acciones y la comprobación de que los cambios realizados estén produciendo los beneficios de costo esperados.

Cuando los costos de Azure se administran correctamente, se pueden respetar los límites y objetivos de presupuesto. La responsabilidad financiera puede garantizarse a través de datos de costos correctos y oportunos, que permitan la comparación con los objetivos financieros. La información obtenida a partir de estos datos puede ayudar a identificar la mala rentabilidad, por ejemplo, recursos infrautilizados. Puede ayudar a identificar opciones que mejorarán la eficiencia o sugerirán cambios para la optimización de costos. Azure permite procesos sólidos de administración de costos con recomendaciones, acciones y la comprobación de que los cambios realizados estén produciendo los beneficios de costo esperados.

Diagnóstico de problemas de servicio en Azure y obtención de soporte

Azure ofrece una amplia gama de herramientas para ayudar a los usuarios a supervisar y administrar situaciones de servicio de Azure. Azure Service Health reúne tres servicios para ayudar a los usuarios a ver el estado general de Azure, los informes personalizados sobre los grupos de activos que afectan a los clientes y la información detallada sobre los activos individuales. Los problemas detectados por las herramientas de Azure Service Health pueden activar alertas a través de mensajes de texto o voz, correos electrónicos, respuestas automatizadas con runbooks de Azure o creados por el usuario, o acciones dentro de Azure o dentro de otras aplicaciones preferidas de administración de recursos.

Estado de nivel global (estado de Azure)

La información de estado de Azure ayuda a los usuarios a ver de un vistazo el estado o los impactos en los servicios que usan. Esta información general del estado de todos los servicios de Azure es parte de Azure Service Health. También puede consultarla cualquier persona que visite la página de **estado pública de Azure** de Microsoft.

Estado de servicio personalizado (Azure Service Health)

Naturalmente, los usuarios también quieren saber sobre el estado de los servicios y regiones de Azure, puesto que se aplica específicamente a sus recursos. Los usuarios pueden acceder a Azure Service Health para ver las comunicaciones sobre interrupciones, acciones de mantenimiento programadas y otra información relacionada con los impactos de estado y servicio, de acuerdo con los servicios y recursos utilizados actualmente por ese usuario. Los usuarios pueden configurar alertas de estado del servicio que les notifiquen en sus medios de comunicación preferidos cuando los servicios y las regiones de Azure que utilizan tengan el riesgo de sufrir problemas de servicio, programar el mantenimiento u otros cambios.

Estado de los activos individuales (Azure Resource Health)

Como usuario autorizado de Azure, puede obtener información de Azure Resource Health sobre un recurso específico de la nube, por ejemplo, una VM individual. A través de Azure Monitor, puede configurar alertas que le adviertan de los cambios en la disponibilidad de sus recursos en la nube. La combinación de Azure Resource Health y Azure Monitor puede proporcionarle una mejor información por minuto, lo que le permite determinar rápidamente si un problema está relacionado con un evento en la plataforma de Azure o fue provocado por un problema en su propio entorno.

Actualizaciones de estado e historial de Azure

La página de **estado de Azure** se actualiza de forma dinámica a medida que se producen cambios en el estado de los servicios de Azure. También puede definir la velocidad a la que esta página se actualiza con nuevos datos, con información sobre la última vez que se actualizó la página. La información sobre el estado de Azure y los cambios del estado del servicio también están disponibles a través de una fuente RSS. La página de **Historial de estado de Azure** muestra eventos pasados con una antigüedad de hasta 90 días, con información preliminar de causa raíz, mitigación y próximos pasos.

Información general de Azure Service Health

Azure Service Health ofrece a los usuarios un panel que pueden personalizar para realizar seguimiento del estado de sus servicios de Azure en las regiones donde los usan. Los usuarios pueden supervisar eventos activos, como problemas continuos de servicio, mantenimiento programado a corto plazo u otros avisos de estado que les conciernen. Los usuarios también pueden usar el panel para crear y administrar alertas que les adviertan de forma proactiva de los problemas de servicio que los afecten. Los eventos que están inactivos se almacenan en el historial de estado durante un máximo de 90 días.

Eventos de Azure Service Health

Azure Service Health supervisa tres tipos de eventos de estado que pueden afectar sus recursos:

1. **Problemas de servicio**

   Estos son problemas continuos que pueden afectar a sus recursos en el momento. La vista de ST le muestra cuándo comenzó el problema, y los servicios y las regiones que afecta. Esta vista también muestra la actualización más reciente de las acciones de Azure para resolver el problema.

   La pestaña **Posible impacto** muestra los recursos que posee y a los que podría afectar el problema. Esta información también está disponible como una lista en formato CSV para su descarga y uso compartido.

2. **Mantenimiento programado**

   Este es un mantenimiento a corto plazo que podría tener un impacto en la disponibilidad de sus recursos.

3. **Avisos de estado**

   Estos se refieren a cambios en los servicios de Azure que debe conocer, como la obsolescencia de las características de Azure o una cuota de uso que se ha superado.

Se encuentra disponible un vínculo para un problema específico en el sistema de administración de problemas de su preferencia. Puede compartir información con otras personas que no tienen acceso al portal de Azure a través de la descarga de archivos PDF (y en algunos casos CSV). Los usuarios también pueden anclar un mapa de estado personalizado a su panel para mostrar sus suscripciones, regiones y tipos de recursos críticos para la empresa a través de un filtro en **Service Health**.

La página de **Service Health** proporciona vínculos para soporte de Microsoft, incluidos casos en que un recurso permanece en un estado insatisfactorio incluso después de que se resolvió un problema.

Configuración de alertas de Azure Service Health

Puede utilizar la integración de Service Health con Azure Monitor para recibir alertas de correo electrónico, mensajes de texto y notificaciones de webhook cuando existan cambios o incidentes que afecten sus recursos. A fin de recibir estas alertas, configure una alerta de registro de actividad para el evento de estado de servicio que le interese y, a continuación, utilice un grupo de acciones para enrutar la alerta a las personas que necesitan conocerla. Un grupo de acciones es una definición de las acciones que se deben tomar si se activa una alerta.

Azure Resource Health

Azure Resource Health informa del estado existente e histórico de sus recursos. Estos informes le permiten diagnosticar y obtener soporte para problemas de servicio que afecten sus recursos de Azure. Mientras que el estado de Azure es un informe "general" sobre los problemas de servicio que afectan a los usuarios de Azure, Resource Health ofrece un panel personalizado que le muestra el estado específico de los recursos. Por ejemplo, Resource Health facilita la comprobación de si se respetaron los SLA mediante la visualización de todas las instancias de falta de disponibilidad de sus recursos debido a problemas de servicio de Azure.

Evaluación de estado de los recursos

Azure Resource Health utiliza señales de varios servicios de Azure para evaluar el estado de un recurso. El recurso puede ser una VM, una base de datos de SQL, una aplicación web o cualquier otra instancia de un servicio de Azure. Si Resource Health encuentra que el recurso no está en buen estado, analiza más información para encontrar la causa del problema. Además, informa sobre las acciones de Microsoft para corregir el problema y también sugiere acciones para que el usuario solucione el problema.

Estado de los recursos

Puede mostrarse cualquiera de los siguientes estados de un recurso:

- **Disponible**

  El estado **Disponible** indica que no se vieron afectados los eventos que afectan el mantenimiento del recurso. Una notificación de **Resuelto recientemente** se muestra hasta 24 horas después de que un recurso se recupera de un tiempo de inactividad no programado.

- **No disponible**

  El estado **No disponible** indica que el servicio detectó un evento continuo de plataforma o de otro tipo (consulte la sección *Eventos de plataforma y de otro tipo*) que afecta al estado del recurso.

- **Desconocido**

  Si Azure Resource Health no recibe información sobre un recurso en los últimos 10 minutos, mostrará el estado como **Desconocido**. Este puede ser un evento importante para una solución de problemas posterior. El estado puede cambiar a **Disponible** después de unos minutos, si el recurso funciona según lo esperado. De lo contrario, los problemas con el recurso pueden indicar que este se ve afectado por un evento en la plataforma.

- **Degradado**

  El estado **Degradado** indica la detección de la pérdida de rendimiento de un recurso, aunque el recurso se pueda seguir utilizando. Los recursos individuales tienen sus propios criterios para informar un estado de **Degradado**.

Eventos de plataforma y de otro tipo

Varios componentes de la infraestructura de Azure activan eventos de plataforma, ya sea como eventos programados o incidentes no planificados (por ejemplo, un reinicio inesperado del host). Azure Resource Health proporciona información adicional sobre el evento y la recuperación del evento. También permite a los usuarios ponerse en contacto con el soporte de Microsoft, con o sin un contrato de soporte activo.

Los eventos que no pertenecen a la plataforma son causados por los usuarios, por ejemplo, detener una VM o alcanzar el límite de conexiones de Redis a Azure Cache for Redis.

Cómo informar de un estado incorrecto

Si considera que el estado de un recurso es incorrecto, puede usar **Informar estado incorrecto** para informarlo a Microsoft. También puede ponerse en contacto con el soporte de Microsoft desde Azure Health Monitor si lo afecta un problema de Azure.

Integración con Azure Monitor

Azure Monitor recopila datos de supervisión (telemetría) de diferentes fuentes locales y de Azure. También recibe datos de registro de herramientas de administración, como las de Azure Security Center y Azure Automation, y puede recibir información de estado de Azure Service Health. Azure Monitor agrega y almacena los datos de telemetría en un almacén de datos de registro configurado para un rendimiento y rentabilidad óptimos.

Los usuarios pueden analizar datos, configurar alertas y obtener vistas completas de sus aplicaciones a través de Azure Monitor. Pueden aprovechar la información de "machine learning" para acelerar la identificación y resolución de problemas. Azure Monitor es compatible con .NET, Java, Node.js y otros lenguajes y marcos populares.

De esta forma, Azure Service Health puede integrarse en el sistema de administración centralizado y escalable de Azure Monitor, que unifica la telemetría operativa y proporciona herramientas avanzadas para mejorar la disponibilidad y el rendimiento.

Con las funcionalidades de Azure Monitor, Azure Service Health puede integrarse con procesos de DevOps y herramientas como Azure DevOps, Jira y PagerDuty, además de otras herramientas de administración favoritas como Grafana, IBM Radar, InfluxDB, SignalFx y Splunk.

Obtención de soporte de Microsoft

Los usuarios de Azure pueden crear y administrar solicitudes de soporte a través del portal de Azure. Por ejemplo, haga clic en **?** en la esquina superior derecha y seleccione **Nueva solicitud de soporte** para crear una solicitud de soporte.

La experiencia de solicitud de soporte se diseñó para ser optimizada, integrada y eficaz para los usuarios. Un asistente ayuda a los usuarios mediante la simplificación del procedimiento, el mantenimiento del contexto de recursos (sin necesidad de cambiar a un contexto distinto al recurso) y la recopilación de la información clave necesaria para la resolución eficiente del problema. La información clave permite al asistente enrutar la solicitud de soporte al ingeniero de soporte técnico más adecuado para el problema, de modo que el diagnóstico y la resolución de problemas puedan comenzar lo antes posible.

De acuerdo con la categoría del problema y el tipo seleccionado por el usuario, Microsoft también puede proporcionar información contextual de autoayuda para que los usuarios aborden sus problemas de inmediato. Si las soluciones recomendadas no solucionan el problema, el proceso continúa hasta la creación de una solicitud de soporte y su transmisión al equipo de soporte de Microsoft.

RBAC para solicitudes de soporte

Azure RBAC le permite definir un acceso de administración altamente detallado. El portal de Azure en `portal.azure.com` utiliza este RBAC para autorizar diferentes niveles o ámbitos de creación y administración de solicitudes de soporte. Por ejemplo, los ámbitos pueden extenderse a un recurso, un grupo de recursos o una suscripción completa. Los usuarios, grupos y aplicaciones pueden tener acceso a través del rol de RBAC adecuado para el nivel o ámbito apropiado para ellos.

Por ejemplo, un propietario del grupo de recursos con permisos de lectura en el ámbito de la suscripción puede administrar todos los recursos del grupo de recursos. Estos recursos pueden incluir VM, aplicaciones web y sitios y subredes. Sin embargo, este propietario del grupo de recursos no puede crear una solicitud de soporte para un recurso de VM en el grupo de recursos. Para ello, primero se debe otorgar permiso de escritura al propietario del grupo de recursos en el ámbito de la suscripción. Como alternativa, el rol del administrador de grupo de recursos podría definirse para incluir autorización específica de Soporte de Microsoft en el ámbito de la suscripción.

Eficacia del soporte

En la sección anterior, se mostró cómo los usuarios pueden beneficiarse de una gama de herramientas de Azure para supervisar y administrar los recursos de servicio de Azure. Estas herramientas van desde información global, disponible públicamente (estado de Azure), a través de notificaciones específicas de servicios y recursos (Azure Service Health y Azure Resource Health), hasta posibilidades completas de integración y administración a través de Azure Monitor y otros sistemas de administración de servicios. Los usuarios pueden recibir notificaciones a través de una variedad de canales y las respuestas a las alertas de estado del servicio pueden automatizarse mediante Azure Automation.

El soporte es fácil de usar y eficaz gracias al uso de vínculos de soporte y un asistente de solicitud de soporte. Al mismo tiempo, el acceso a las solicitudes de soporte es flexible y seguro mediante RBAC de Azure para autorizar las acciones del usuario en los niveles apropiados y dentro del ámbito adecuado.

Resumen

En este capítulo, se exploró la administración y optimización de los recursos relacionados con Azure desde diferentes ángulos, incluida la disponibilidad, la eficacia, la seguridad y la rentabilidad. También analizamos los entornos de nube híbrida y mostramos cómo se pueden usar las soluciones de Azure para mejorar la administración de los recursos en instalaciones que no son de Azure. También se abordaron optimizaciones para el ahorro de costos y presupuesto de Azure. Terminamos este capítulo presentando la supervisión y la resolución de problemas desde diferentes puntos de vista, incluido el estado de los recursos.

Índice

A

B

C

D