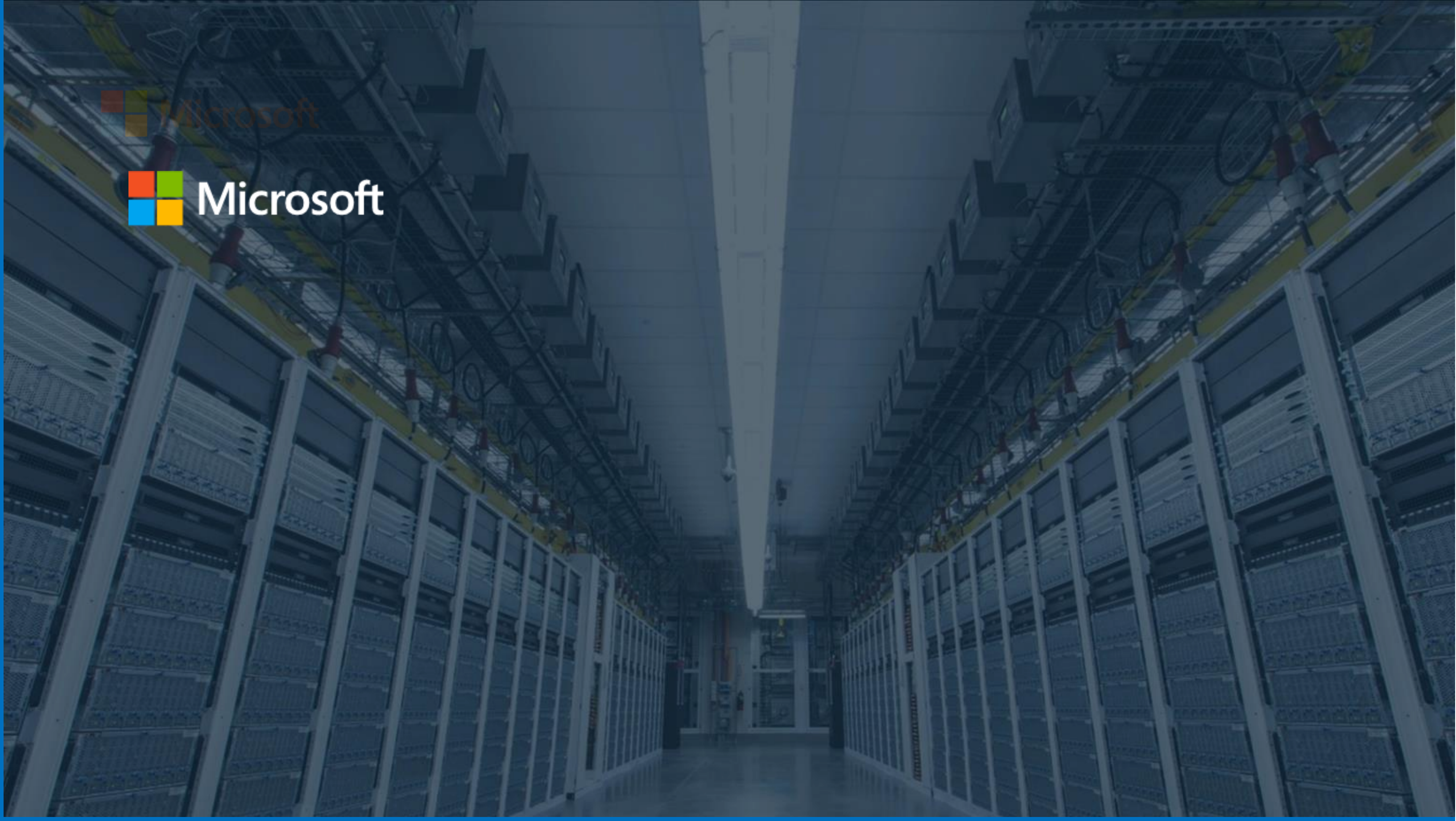


Azure for your Linux and open source components

Migrate, develop, deploy and operate your applications on an open, flexible, secure and trusted platform

By Philippe Beraud (Microsoft France)

Version 1.1, October 2019 (Updated November 2019)

For the latest information about Azure, please see
https://azure.microsoft.com/en-us/overview/

For the latest information about open source on Azure, please see
https://azure.microsoft.com/en-us/overview/choose-azure-opensource/

For the latest information about Cloud-native applications on Azure, please see
https://azure.microsoft.com/en-us/overview/cloudnative/

This page is intentionally left blank.

Table of contents

Notice

This guide for architects and developers is intended to illustrate how you can leverage Azure, as an open, flexible, secure, and trusted platform for your Linux and open source components. For that purpose, and from this standpoint, it outlines some (key) products and services Azure offer in this space.

MICROSOFT DISCLAIMS ALL WARRANTIES, EXPRESS, IMPLIED, OR STATUTORY, IN RELATION WITH THE INFORMATION CONTAINED IN THIS WHITE PAPER. The white paper is provided "AS IS" without warranty of any kind and is not to be construed as a commitment on the part of Microsoft.

Microsoft cannot guarantee the veracity of the information presented. The information in this guide, including but not limited to internet website and URL references, is subject to change at any time without notice. Furthermore, the opinions expressed in this guide represent the current vision of Microsoft France on the issues cited at the date of publication of this guide and are subject to change at any time without notice.

Publication: October 2019 (Updated November 2019)
Version 1.1

Introduction

The cloud emerges as a major disruptive force in shaping the nature of business and plays a major opportunity and allowing dramatic global growth. The cloud brings breakthrough change, and thus represents (for incumbents) a major opportunity and plays a determining role.

An increasing number of people recognize the benefits of locating existing applications or building new ones in the public cloud to reduce infrastructure and ongoing datacenter operational costs, along with the technical debt, maximize availability, simplify management, take advantage of a predictable pricing model – provided that the resource consumption is also predictable, rapidly deliver to the market, elastically scale in respect to the demand, but also to build and leverage key competitive advantages, open multichannel access for their own business, etc.

We fundamentally see digital transformation as serving the organizations' business whatever it is, which is in our view the *sine qua non* of both its ownership by organizations and success in the era of the now so-called "Intelligent Cloud and Intelligent Edge" we are entering into.

Modernizing applications

Gartner predicts that every dollar invested in digital transformation and innovation through to the end of 2020 will require organizations to spend at least three times that to continuously modernize the legacy application portfolio.

Modernization strategies for on-premises hosted applications to the public cloud have been well-theorized notably by Gartner and implemented given the increasingly massive adoption of the cloud by organizations, even though the "lift-and-shift" migration solution reproducing identically the application and its environment remains the easy way.

There is one journey, but each application can take a radically different path to get to the cloud. "Application modernization is not one 'thing,'" says Stefan van der Zijden, research director at Gartner. "If you're faced with a legacy challenge, the best approach depends on the problem you're trying to solve", the workload itself, and its architecture.

At a very high level, applications consist of three layers. The first layer is the application code – functionality and business logic. Then, there's the data that the application consumes and generates – every application works with data, and that data can come from many different sources. Finally, there's the physical or virtualized infrastructure the application runs on – servers or virtual machines, networking and so on.

When you are looking to modernize an application, you will need to look at all these layers individually.

According to the Gartner, seven different modernization approaches can be considered depending on your goals and on the problem to solve : "The key is to understand if your problem is caused by technology, architecture or functionality of the application, and how each modernization approach improves those aspects […]"

1. **Encapsulate**. To leverage and extend an application's features and value, encapsulate data and functions in the application and make them available as services via an application programming interface (API). Implementation specifics and knowledge are hidden behind the interface.
2. **Rehost**. Redeploy an application component to another physical, virtual or cloud infrastructure without recompiling, altering the application code, or modifying features and functions.
3. **Replatform**. Migrate an application component to a new runtime platform. Make minimal changes to code to adapt to the new platform, but don't change the code structure or the features and functions it provides.

4. **Refactor**. Restructure and optimize existing code without changing its external behavior to remove technical debt and to improve the component's features and structure.
5. **Rearchitect**. Materially alter the application code so you can shift it to a new application architecture and fully exploit new and better capabilities of the application platform.
6. **Rebuild**. Rebuild or rewrite the application component from scratch while preserving its scope and specifications.
7. **Replace**. Eliminate the former application component altogether and replace it, taking new requirements and needs into account."

The best choices in the context are probably between:

1. **Rehost (lift-and-shift)**. The application is migrated into the Infrastructure-as-a-Service (IaaS) environment of the cloud provider while avoiding modifications to the system as much as possible. Virtual machines (VMs) that support all features and the infrastructure that hosts them are recreated identically in the cloud. This is a simple VMs-based rehosting that does not involve any changes to the application, but only brings the benefits of the cloud to a limited extent.
2. **Refactor**. The application is modified at a minimum to take advantage of the features of the cloud: for example an on-premises MySQL database will be migrated to Azure Database for MySQL, i.e. an Enterprise-ready, fully managed community MySQL database , PHP web parts will be migrated to Azure App Service, i.e. another fully managed platform (Platform-as-a-Service or PaaS) available on Microsoft Azure, some services will be containerized for easier migration and more efficient managed through the available orchestrators.

Note The Azure Database Migration Service reduces the complexity of your cloud migration by using a single comprehensive service instead of multiple tools. This is a fully managed service designed to enable seamless migrations from multiple database sources to Azure data platforms with minimal downtime (online migrations). This service allows you to migrate on-premise databases such as MySQL, MongoDB, and PostgreSQL to an Azure managed database in the cloud or to your own database running in an Azure VM.

The service uses the Data Migration Assistant (DMA) to generate assessment reports that provide recommendations to guide you through the changes required prior to performing a migration.

3. **Rearchitect**. The application architecture is completely redesigned to take advantage of the cloud capabilities and the provided services' portfolio. For example, some parts of the application may be redistributed as services to match business functions, others will disappear being replaced by cloud services, others will be reviewed or dissociated for better availability or scalability. The new architecture will be based on up-to-date models implementing recent technologies such as microservices.

These three approaches are gradually more complex. *Rehost* (lift-and-shift) will allow to migrate more quickly and with less effort. This constitutes a viable path to the cloud for many applications. Some cloud benefits are quickly unlocked, and you can take advantage of advanced cloud capabilities such as autoscaling or improved resiliency gradually by modernizing afterwards.

Modernizing an application involves some change to application design, but it does not necessarily require wholesale changes to application code. Refactor consists of an adaptation of the major functionalities without reconsidering the complete architecture of the application while limiting the modernization effort. Rearchitect is however more complex, more expensive, as it requires changes to the application source code or/and architecture, but this is the one that will offer maximum benefits in its new cloudified version if we leave aside *Rebuild* (see below).

With the modernization, the application takes advantage of IaaS and potentially PaaS capabilities from a cloud service provider (CSP) while maintaining the existing code strategic to the application's use case. This approach is particularly interesting for organizations looking to unlock advanced cloud benefits even if they are unable or unwilling to change code and rewrite the application. It is also the preferred approach for organizations using multiple cloud providers that are looking for portability across clouds.

In terms of a Cloud maturity model, the first choice, i.e. the migration, can be referred as to "Cloud Ready Infrastructure" while the last two, i.e. the modernization, as to "Cloud Optimized".

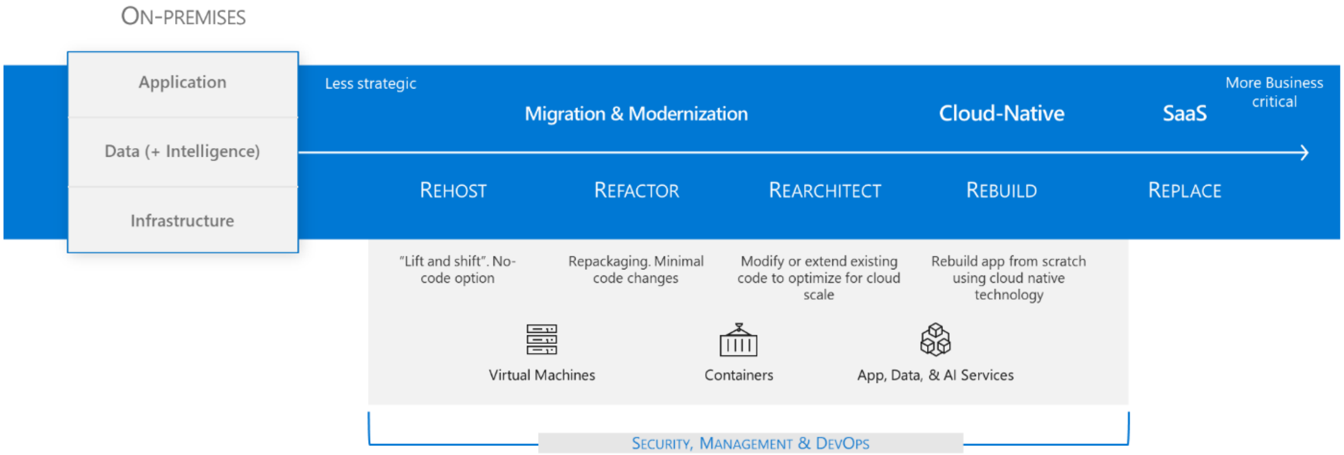

ON-PREMISES
Application
Data (+ Intelligence)
Infrastructure
Less strategic
Migration & Modernization
Cloud-Native
SaaS
More Business critical
REHOST
REFACTOR
REARCHITECT
REBUILD
REPLACE
"Lift and shift". No-code option
Repackaging. Minimal code changes
Modify or extend existing code to optimize for cloud scale
Rebuild app from scratch using cloud native technology
Virtual Machines
Containers
App, Data, & AI Services
SECURITY, MANAGEMENT & DEVOPS

Figure 1 Main approaches for application modernization

Note For more information about the different modernization strategies, see article Contoso migration: Overview/Migration strategies.

Rebuild leads to a complete rethink of the application to develop a new version almost "from scratch", which is outside the scope of this paper since this aims at developing a brand-new application, which will bring at least the features of the previous version.

However, if you are looking to get the most from the cloud and tap into advanced capabilities like improved resiliency, global scale or maximum agility, this route allows to embrace new and more relevant standards, e.g. the Cloud-Native Computing Foundation (CNCF) to have applications be built from the ground up and optimized for cloud scale and performance.

Note Microsoft has joined the CNCF as a Platinum member in 2017. CNCF is a part of the Linux Foundation, which helps govern for a wide range of cloud-oriented open source projects, such as Kubernetes, Prometheus, OpenTracing, Fluentd, Linkerd, containerd, Helm, gRPC, and many others.

CLOUD NATIVE
COMPUTING FOUNDATION

In other words, options range from migration (moving the application, its infrastructure and data as-is to the cloud) through modernization (where an application is modified to better take advantage of the cloud) to re-building where the application is recreated using a cloud-native approach.

In addition, and in this context, we cannot stress enough that the success of the digital transformation of organizations of all size reside notably in their ability to make the most of their data.

We will have to cope with all the above as part of your journey to Microsoft Azure.

Objectives of the document

Considering the above, the purpose of this paper aims at discussing the considerations that pertain to Azure for your Linux and open source components in order to seamleslly and smoothly migrate, develop, deploy and operate your applications.

For that purpose, this document will introduce the key IaaS services to consider along with their core associated concepts, principles, and considerations.

In addition, wherever relevant, additional managed PaaS services will be also highlighted.

Non-objectives of the document

For the wide range of Azure products and services outlined in this document, no in-depth will be provided. You can instead refer to the extensive documentation available on docs.microsoft.com. As such, the various links provided as resources constitute a starting point to deeper dive into that documentation.

This said, the modernization of an application heavily depends as a whole on its architecture, complexity, constraints and targeted objectives.

The choice for the best suited modernization route(s) and the impact on the target architecture will strongly depend on the considered application and the goals that sustain the modernization. The route(s) to follow may involve more or less deep and costly changes. To adapt to the constraints of budget or time, you will now have to think about possible iterations to deliver functional versions but with incremental changes and benefits.

This document will not cover the design principles, or architecture options that your application(s) may follow as part of a modernization effort. **We strongly advise to review the Azure Application Architecture Guide along with the one for Structured review of Azure architectures.**

Moreover, this does not prevent to define a number of steps from the modernization vision (along with its the triggers, goals, the success factors, and related key performance indicators (KPIs)) to the effective deployment that you will be able to follow to realize the full potential of this modernization effort, which supposes at first glance to consider the expected outcomes. This document will not discuss these steps either. This is also outside the scope of this document.

Organization of the document

Beside starting by considering some of the key reasons you may have to choose Microsoft Azure as a platform for your applications (and data), the document discusses the following topics:

- Rehosting your applications.
- Building new Cloud-native applications.
- Connecting your applications with data.
- Securing your applications.
- Deploying your Azure services.
- Implementing (secure) DevOps practices for your applications.
- Monitoring your applications.

- Keeping your applications up and running.

Each above topic is covered in order by a dedicated section.

Audience of the document

This document is intended for architects, developers or anyone wishing to understand the benefits of Microsoft Azure to develop, deploy and operate their applications based on Linux and open source components.

Let’s jump into the first topic!

Why choosing Azure?

Microsoft Azure is an open, flexible, enterprise-grade cloud computing platform that provides an ever growing collection of integrated IaaS and PaaS cloud services - compute, storage, networking, data lake, not only SQL (NoSQL) and relational (SQL) database, advanced analytics, Machine Learning, Internet of Things (IoT), mobile, web, API, etc. - that allow Microsoft customers to move faster, achieve more, and save money.

> **Note** Azure updates allows customers to stay up to date on what product features are planned and what's coming next. Customers can learn about important Azure product updates, roadmap and announcements, and subscribe to notifications to stay informed.

Azure serves as a development, service hosting, and service management environment, providing customers with on-demand IaaS and PaaS resources, and content delivery capabilities to host, scale, and manage applications on the Internet.

Microsoft Azure has established itself in recent years as a great leader in the cloud market. In fact, Azure is considered by Gartner[1] in 2019 to be a leader in a number of magic quadrants (MQ) in the cloud: Cloud Infrastructure as a Service, Disaster Recovery as a Service, Access Management, Public Cloud Storage Services, etc.

As such, Microsoft Azure provides:

- An open platform to support your choices and preferences for your applications,
- A globally available infrastructure for your applications,
- Clearly stated cloud principles of trust for your applications and data.

Let's consider the above in the next sections.

An open platform to support your choices and preferences for your applications

Microsoft Azure offers a feature rich environment incorporating the latest cloud innovations to help customers and partners increase efficiency and unlock insights into their operations and performance.

As a direct translation of our strategy that consists in:

- Meeting where customers are - devices, operating systems (OS), languages, and hybrid.
- Delivering the most productive and trustworthy cloud services platform that enables developers using any framework on any device or OS to create and power the world's applications and services that run anywhere.

[1] Gartner does not endorse any vendor, product or service depicted in its research publications, and does not advise technology users to select only those vendors with the highest ratings or other designation. Gartner research publications consist of the opinions of Gartner's research organization and should not be construed as statements of fact. Gartner disclaims all warranties, expressed or implied, with respect to this research, including any warranties of merchantability or fitness for a particular purpose.

- Providing the best cloud platform for partners to succeed.

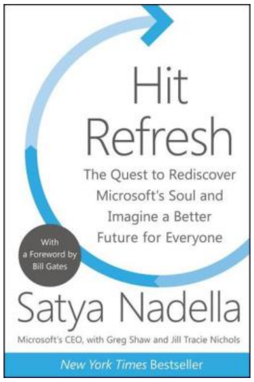
Microsoft Azure has a strong track record with customers for providing strong support, service, and security, across a wide breadth of technology stacks, and a commitment to being open and flexible, which is especially important for heterogeneous environments.

Microsoft Azure supports Linux and the open source technologies millions of users already rely on and trust. Customers thus have suitable choices that help them maximize their existing investments or the ones they target for the future.

Support for IaaS on Linux, Java, and PHP Web application platforms is amongst many others provided. Azure offers Enterprise-grade support for all popular Linux distros: CentOS, CoreOS, Debian, Red Hat, SUSE Enterprise Linux Server, Ubuntu. There's a little-known fact that about 50%+ of servers in Azure are running Linux and Azure is running containerized workloads. Besides that, the microservices are all using open source programming languages and interfaces.

As Satya Nadella, Microsoft CEO, outlines in his Hit Refresh book:

> ***We are investing in and improving these new products, building new muscle as a service provider, and embracing Linux and other open-source efforts, all while keeping focus on our customers.***
>
> ***Microsoft had long held that the open-source software from Linux was the ennemy. We couldn't afford to cling to that attitude any longer. We had to meet the customers where they are and, more importantly, we needed to ensure that we viewed our opportunity not through a rearview mirror, but with a more future-oriented perspective.***


Hit
Refresh
The Quest to Rediscover Microsoft's Soul and Imagine a Better Future for Everyone
With a Foreword by Bill Gates
Satya Nadella
Microsoft's CEO, with Greg Shaw and Jill Tracie Nichols
New York Times Bestseller

Today, an increasing number of customers are choosing to build open source solutions on top of Azure.

You can seamlessly and easily develop and test your Linux and open source components in Azure. Microsoft Azure enables every developer and organization to more easily adopt open source in the cloud, without having to be an expert.

Note For more information about open source software on Azure, see page Open source on Azure.

You can bring the tools you love and skills you already have, and run virtually any application, using your data source, with your OS. You can even install and run Microsoft SQL Server on Linux.

As such, you can use Microsoft Azure to deploy a variety of existing and new (business-critical) workloads and benefit from rapid feature growth, resiliency, and the cost-effective operation of the hyperscale public cloud while still obtaining the levels of isolation, security, compliance, and confidence required to handle your workloads.

In addition, you can complement what you've already built by using Azure, augment your open source workloads, and add values to them with technologies and fully managed services that work well with each other.

For that purpose, you can tap into a growing ecosystem of solutions, including open source, available from the Azure Marketplace that enable rapid deployment in the cloud.

Note Azure Marketplace is a service on Azure that helps connect you with offerings, virtual (network) appliances and services, which are optimized to run on Azure. Azure Marketplace allows you to find, try, purchase, and provision applications and services from hundreds of leading service providers, all certified to run on Azure.

> The solution catalog spans several industry categories, including but not limited to: open-source container platforms, VM images, databases, application build and deployment software, developer tools, threat detection, and blockchain. Using Azure Marketplace, you can provision end-to-end solutions quickly and reliably, hosted in your own Azure environment. At the time of writing, this includes over 8,000 listings.

For testing purposes, you can leverage Azure Test Drives. Azure Test Drives are free ready-to-go environments that allow you to experience a product or a technology for free without needing an Azure subscription at all. One can for example test drive Ansible Tower by Red Hat that helps organizations scale IT automation and manage complex deployments across physical, virtual, and cloud infrastructures.

An additional benefit with a Test Drive is that it is pre-provisioned - you don't have to download, set up or configure the product or the technology and can instead spend your time on evaluating the user experience, key features, and benefits of the product or the technology.

Open source as part of a day-to-day approach to cloud innovation

At Microsoft, open source is a part of our day-to-day approach to cloud innovation.

In 2014, Satya Nadella directed all Microsoft engineers to "open source internally" so that anyone else at the company can see anyone else's code and use it as needed. This vision is now a day-to-day reality for Microsoft engineers. In October of the same year, he even announced at a Microsoft Conference in San Francisco that Microsoft loves Linux!

These are only two examples amongst many many others.

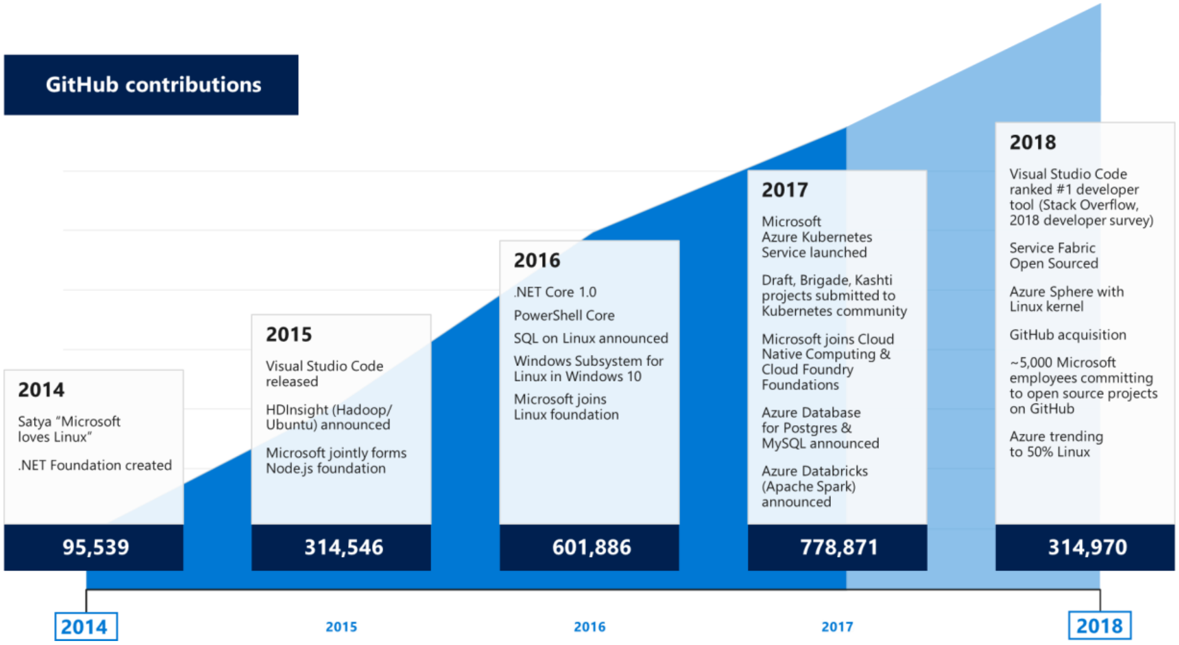


Figure 2 Microsoft actions, along with Microsoft contributions on GitHub.

As illustrated in the above Figure 2, **Microsoft joined the Linux Foundation organization in 2016 as a Platinum member to confirm the steadily increasing interest and engagement in the open source development.** Microsoft is also a member of the Cloud-Native Computing Foundation (CNCF) as already outlined, the Cloud Foundry Foundation, the Apache Software Foundation, the .NET Foundation, and recently MariaDB.

Microsoft is very serious about open source and open sourced .NET, PowerShell, and many other technologies and products. Microsoft has invested heavily in hiring the best open source talent. Microsoft is now home to some of the founders and top developers of open source technologies.

As far as Azure is concerned, technical documentation, SDKs, and examples are all open source and available on GitHub: https://github.com/Azure.

Satya Nadella said back in 2018:

> ***Judge us by the actions we have taken in the recent past, our actions today and in the future.***

Microsoft is working together with open source projects and vendors and is also a major contributor of code to many open source projects. Microsoft has the most GitHub contributors, second most projects.

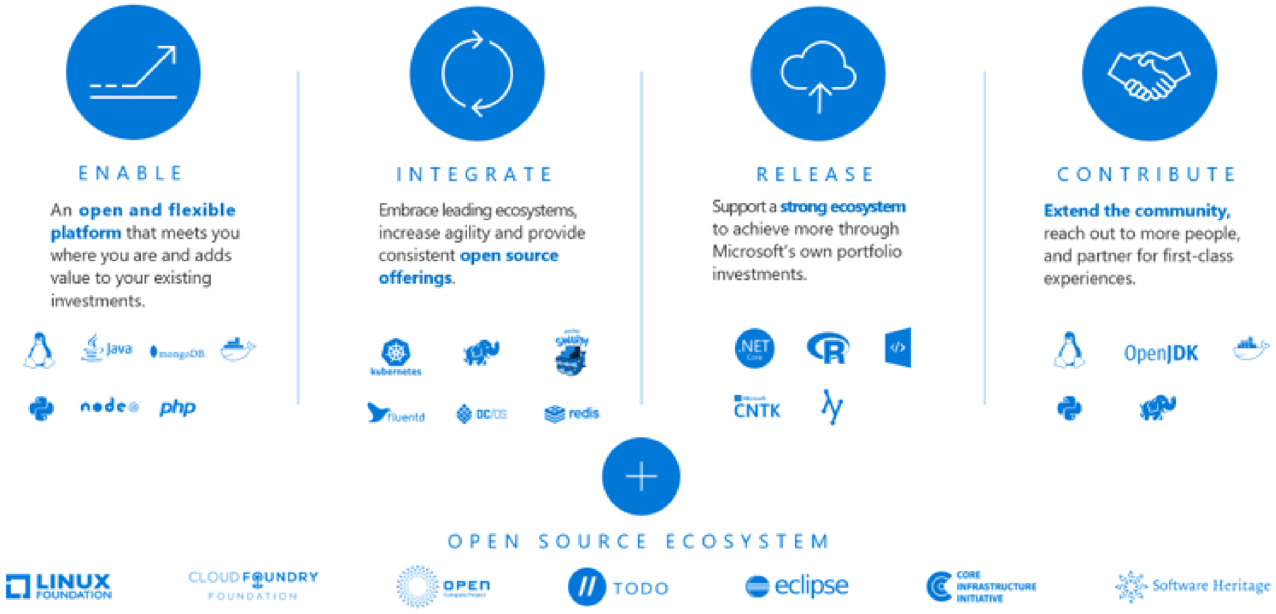

ENABLE
An open and flexible platform that meets you where you are and adds value to your existing investments.
Java
mongoDB
node
php
INTEGRATE
Embrace leading ecosystems, increase agility and provide consistent open source offerings.
kubernetes
fluentd
DC/OS
redis
RELEASE
Support a strong ecosystem to achieve more through Microsoft's own portfolio investments.
.NET Core
CNTK
CONTRIBUTE
Extend the community, reach out to more people, and partner for first-class experiences.
OpenJDK
OPEN SOURCE ECOSYSTEM
LINUX FOUNDATION
CLOUD FOUNDRY FOUNDATION
OPEN Compute Project
TODO
eclipse
CORE INFRASTRUCTURE INITIATIVE
Software Heritage

Figure 3 Microsoft approach to open source in the Cloud

For example, Microsoft is also a key contributor to the Kubernetes project. To make Kubernetes easier for organizations to adopt - and easier for developers to use - Microsoft has tripled the number of employees who participate in the open source project in just three years. Now the third-leading corporate contributor, Microsoft works to make Kubernetes more enterprise-friendly and accessible by bringing the latest learnings and best practices from working with diverse customers to the Kubernetes community. As an illustration, we are delivering a simplified end-to-end experience for Kubernetes and adding new container capabilities with Docker and serverless Kubernetes integration. (See section § "Scaling and orchestrating containers" below).

All these investments that pertain to Kubernetes are led by the following people:

- Brendan Burns, Kubernetes cofounder, Director of Engineering at Microsoft leading the Azure Container Service and Azure Resource Manager teams, and now Microsoft Distinguished Engineer for containers and DevOps.

- Gabe Monroy, CTO and creator of Deis from which have originated the Helm, Draft, Brigade projects, and now Director of Program Management in Azure Compute responsible for Azure products spanning containers, functions, messaging, eventing, etc.

  Gabe Monroy manages the teams responsible for open source infrastructure software in and around the Cloud Native Computing Foundation (CNCF) including Kubernetes, Helm, Cloud Native Application Bundles (CNAB), Draft, Brigade, Virtual Kubelet, Service Mesh Interface (SMI), and more.

Likewise, we're also constantly looking for ways to improve developer and user experiences with SDKs for open source languages and an open API. Early this month, we're announcing two new open source projects:

1. The Dapr project. Dapr is an event-driven, portable runtime that takes some of the complexity out of building microservices, makes it easy for developers to build resilient, microservice stateless and stateful applications that run on the cloud and edge and embraces the diversity of languages and developer frameworks.
2. The Open Application Model (OAM) project under the Open Web Foundation. OMA is a specification that allows developers to define the resources their applications need to run on Kubernetes clusters and which Microsoft developed in cooperation with Alibaba Cloud.

Note For more information about the above announcements, see blog posts Announcing Distributed Application Runtime (Dapr), an open source project to make it easier for every developer to build microservice applications and Announcing the Open Application Model (OAM), an open standard for developing and operating applications on Kubernetes and other platforms.

Plus, we're committed to sharing our cloud learnings with you and for your datacenters, thanks to Linux and open source support in Azure Resource Manager (ARM) and Azure Stack.

Note For more information about open source trends in Microsoft Azure, see the Open Source Blog.

Partnerships with the top tier open source technology providers

In addition to the above, Microsoft Azure has partnerships with the top tier open source technology providers, including Red Hat, SuSE, Pivotal, Hashicorp, Docker, Mesosphere, Cloudera, and many more, ensuring customer success.

Let's take Red Hat as an illustration.

Back in November 2015, Red Hat and Microsoft entered into a joint, strategic partnership to offer Red Hat solutions on Microsoft Azure. A partnership was formed in response to what mutual customers demanded - the ability to deploy, run, and scale workloads on Red Hat products across a hybrid mix of cloud and on-premise environments.

As Microsoft's Azure strategy was evolving, it was clear that Red Hat's open source solutions would play a vital role in meeting the needs of our customers, developers, and partners. This partnership included initially:

- Integrated enterprise-grade support spanning hybrid environments.
- Red Hat solutions available natively to Microsoft Azure customers.
- Collaboration on .NET for a new generation of application development capabilities.
- Unified workload management across hybrid cloud deployments.

Due to the success of this partnership, it was extended to other products. Microsoft SQL Server on Red Hat Enterprise Linux was for example formally added to the partnership in July 2017 and the Generally Available (GA) version made available to the public in October 2017.

And later, in May 2018 Microsoft and Red Hat expanded their alliance to empower enterprise developers to run container-based applications across Microsoft Azure and on-premises. With this collaboration, the companies introduced the first jointly managed OpenShift offering in the public cloud, combining the power of Red Hat OpenShift and Azure.

Azure Red Hat OpenShift, a flexible, self-service deployment of fully managed OpenShift clusters, has been jointly engineered and designed to reduce the complexity of container management for customers. You can focus on your application development, while your master, infrastructure, and application nodes are patched, updated, and monitored by both Microsoft and Red Hat.

A globally available infrastructure for your applications

Microsoft Azure is a hyperscale public cloud to adequately sustain your projects and accommodate their scale and footprint on the planet. For that purpose, Microsoft Azure provides a global infrastructure to achieves both the global reach and the local presence you (may) need.

As of this writing, Azure is available in 140 countries with an energy-efficient infrastructure spanning more than 100 highly secure facilities worldwide, linked by one of the largest networks on earth.

Azure divides the world into geographies that are defined by geopolitical boundaries or country borders. An Azure geography is a discrete market typically containing two or more regions that preserve data residency and compliance boundaries. This division provides several benefits:

- Geographies allow customers with specific data residency and compliance needs to keep their data and applications close.
- Geographies ensure that data residency, sovereignty, compliance, and resiliency requirements are honored within geographical boundaries.
- Geographies are fault-tolerant to withstand complete region failure through their connection to dedicated high-capacity networking infrastructure (see section § “What can Azure do for high-availability?” below).

Note Data residency refers to the physical or geographic location of an organization's data or information. It defines the legal or regulatory requirements imposed on data based on the country or region in which it resides. See eponym section § “Data residency” below.

Geographies are broken up into the following areas:

- Americas,
- Europe,
- Asia Pacific,
- Middle East and Africa.

Each region belongs to a single geography and has specific service availability, compliance, and data residency/sovereignty rules applied to it.

As of this writing, Azure has 54 regions (as of 2019) across the globe with up to 1.6 Pbps of bandwidth in a region. Azure offers a global footprint, including China and Russia deployments.

Note China regions are operated through a partner called 21Vianet. Microsoft Azure operated by 21Vianet (Azure China 21Vianet) is a physically separated instance of cloud services located in mainland China, independently operated and transacted by Shanghai Blue Cloud Technology Co., Ltd. ("21Vianet"), a wholly owned subsidiary of Beijing 21Vianet Broadband Data Center Co., Ltd..

Note For Russia, you can get Azure services through services providers offerings based on the Azure Stack portfolio (see below). Regarding our enterprise cloud services, we are following the Russia Data Localization Law. The Federal Law No. 242-FZ amendments effective from September 1st, 2015, imposes a data localization requirement on data "operators" in Russia that collect "personal data" of Russian citizens.

Microsoft has one of the world's 3 largest networks (bandwidth, latency, etc.) to ensure interconnection between all regions. A region consists of a set of datacenters deployed in a perimeter with defined latency and connected through a dedicated regional network with low latency.

To build, develop and lead this global network, we rely on three guiding principles:

1. Being as close as possible to our customers for optimal latency.

Note Latency is therefore a function of the distance (in the sense of the network path) between the client and the data center. Microsoft uses innovative software to optimize network routing and build and deploy as direct network paths as possible between customers and their data and services. This reduces latency to the limits imposed by the speed of light. You can measure this latency between your current location and our data centers with the Azure Speed online tool.

2. Stay in control of capacity and resilience to ensure the network can survive multiple failures.
3. Proactively manage network-wide traffic through a software-defined network (SDN) approach.

Customer traffic enters our global network via strategically placed Microsoft Edge nodes, our points of presence. These Edge nodes are directly interconnected to more than 2,500 unique Internet partners through thousands of connections in more than 150 locations. Our rich interconnect strategy optimizes the paths taken by data traveling on our global network. With all of that, you get a better network experience with less latency, packet loss and more throughput. Direct interconnections give customers better quality of service over transit links because there are fewer transitions, fewer intermediaries, and better network paths.

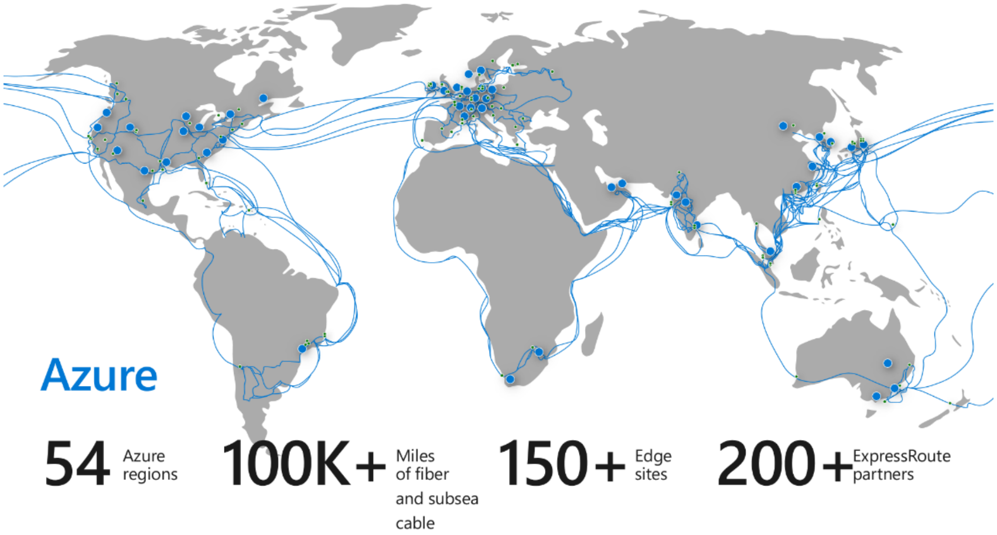

Azure
54 Azure regions
100K+ Miles of fiber and subsea cable
150+ Edge sites
200+ ExpressRoute partners

Figure 4 Microsoft Azure global infrastructure

Where in a specific country Azure may not suit the customer's requirements for whatsoever reason, the Azure Stack portfolio can represent along with other alternatives an enabling technology that allow you to process your data using a private or hybrid cloud and pursue your strategy leveraging Microsoft intelligent cloud and intelligent edge approach.

For many customers, enforcing data sovereignty, addressing custom compliance requirements are the primary driving factors behind these efforts along with applying maximum available protection to highly sensitive customer data.

The recently announced Azure Stack portfolio constitutes an extension of Azure to consistently build and run hybrid applications across datacenters, edge locations, remote offices, and cloud. As its name indicates, it's a portfolio of products consisting of notably Azure Stack Hub (previously Azure Stack), and Azure Stack Edge (previously Azure Data Box Edge).

Azure Stack Hub is a Cloud-native integrative integrated system of software and validated hardware that you can purchase from Microsoft hardware partners, deploy in your own datacenter, and then operate entirely on your own or with the help from a managed service provider. With Azure Stack, you are always fully in control of access to your data. Azure Stack can accommodate up to 16 physical servers per Azure Stack scale unit. It represents an extension of Azure, enabling customers to provision a variety of IaaS and PaaS services and effectively bring multi-tenant cloud technology to on-premises and edge environments. You can run many types of virtual machine (VM) instances, containers, Azure Functions (for (event-driven) serverless computing), Azure Event Hubs, and other services while using the same development tools, APIs, and management processes they use in Azure.

> **Note** For more information, see Azure Stack Hub user documentation, and more specifically article Using Services or Building Apps for Azure Stack Hub.

Azure Stack Hub is not dependent on connectivity to Azure to run deployed applications and enable operations via local connectivity.

Azure Stack Hub can help deploying an application or a full solution differently depending on the country or region. You can develop and deploy applications in Azure, with full flexibility to deploy on-premises with Azure Stack Hub based on the need to meet data sovereignty or custom compliance requirements. You can leverage Azure Stack Hub architecture for data sovereignty, e.g., transmit data from Azure Virtual Network (VNET, see section § "Leveraging network virtualization capabilities" below) to Azure Stack Hub VNET (or vice versa) over private connection and thus making technology placement decisions based on business needs - simplifying meeting custom compliance, sovereignty, and data gravity requirements. You can use Azure Stack Hub to accommodate even more restrictive requirements such as the need to deploy solutions in a completely disconnected environment managed by security cleared, in-country personnel.

Azure Stack Edge is a Cloud managed and AI-enable edge appliance that brings the compute power and intelligence of Azure right to where you need it—whether that's your corporate data center, your branch office, or your remote field asset.

Azure Stack Edge runs containers to analyze, transform, and filter data at the edge locations or datacenters. Aside the Azure IoT Edge container platform that is currently used to provision and manage containers, Azure Stack Edge will also soon support VMs and Kubernetes clusters so that you have a single platform to run most of your edge compute workloads, be it net-new container based applications or the existing virtual machine (VM) based applications. This capability is part of the recently announced Azure Arc.

For customers who want to simplify complex and distributed environments across on-premises, edge and multicloud, Azure Arc, currently in preview, enables deployment of Azure services anywhere and extends Azure management to any infrastructure.

With Azure Arc, you can deploy Azure SQL Databases on any Kubernetes cluster. Azure Arc and Azure Stack portfolio are complementary. You can combine the benefits of Azure Arc with Azure Stack portfolio where Azure Arc can manage VMs, containers, and run Azure Data Services on Azure Stack portfolio of validated and integrated systems while leveraging the compute and cloud capabilities of Azure Stack.

Clearly stated cloud principles of trust for your applications and data

In addition to the above, Microsoft Azure provides services that can help address the security and compliance requirements of customers. In addition, Microsoft works with customers to understand their assurance concerns, and to help define their responsibilities as well as its own with regard to protecting customer data and environmental infrastructure after services are provisioned. Such infrastructure includes (microservices-based) applications, data content, virtual machines, access credentials, and compliance requirements.

As a customer, you may have several legitimate typical questions to ask like:

- *Does Azure potentially meet your (specific) security and compliance requirements?*
- *Where is data stored and who can access it in the light of your security* and *compliance requirements?*
- *What is Microsoft doing to protect data in accordance to your security* and *compliance requirements?*
- *How can you verify that Microsoft is doing what it says?*

Protecting the security, privacy, and integrity of sensitive customer data is one of Microsoft's highest priorities.

Microsoft is committed to earn your trust and we take seriously our commitment to safeguard our customers' data, to protect their right to make decisions about that data, and to be transparent about what happens to that data. As outlined by Brad Smith, President and Chief Legal Officer, Microsoft Corporation:

> ***People will not use technology they do not trust. And they cannot trust technology they do not understand.***

We are guided by a set of "Trusted Cloud Principles," that articulate our vision of what Enterprise organizations are entitled to expect from their Cloud Service Provider (CSP) if any.

Such a vision and its day-to-day translation in our investments, practices and operations, etc. allow our Azure services to deliver enterprise-grade security at every layer, helping ensure your data is safe. We operate them with high ethical standards that provide transparency on how we design our solutions and protect your data. We eventually collaborate with global security experts and proactively invest in technology, policy and regulations that enhance the public security ecosystem.

The Microsoft Trust Center lists the four underlying foundational principles that guide the way Microsoft Azure is built and operated for a Trusted Cloud:

1. Security,
2. Privacy,
3. Transparency,
4. Compliance.

Note For more information, see whitepaper Trusted Cloud: Microsoft Azure Security, Privacy, Compliance, Reliability/Resiliency, and Intellectual Property.

Security

You must be able to count on the security of your data. Security is built into Microsoft Azure from the ground up, starting with the Microsoft Security Development Lifecycle (SDL), i.e. a mandatory development process this is part of the Microsoft Operational Security Assurance (OSA) framework, and that embeds security requirements into every phase of the development process for all the engineering and development projects. (see section § "Microsoft Security standards and related practices" in the Appendix)

Microsoft engineers help ensure that Microsoft Azure is protected at the physical, network, host, application, and data layers so that all services are resilient to attack. Continuous proactive monitoring, penetration testing, and the application of rigorous security guidelines and operational processes further increase the level of detection and protection throughout Microsoft Azure.

So, unsurprisingly, certification to the following two ISO/IEC standards is at the forefront of Microsoft's approach to implementing and managing information security, considering both the international acceptance and applicability:

- The ISO/IEC 27001:2013 "Information technology - Security techniques - Information security management systems - Requirements". Microsoft's achievement of ISO/IEC 27001:2013 certification points up its commitment to making good on customer promises from a business, security compliance standpoint. Currently, Azure is audited once a year for ISO/IEC 27001:2013 compliance by a third-party accredited certification body, providing independent validation that security controls are in place and operating effectively.

Note You can review the Azure ISO/IEC 27001 certificate, assessment report, and statement of applicability on the Service Trust Portal. For more information, see article ISO/IEC 27001:2013 Information Security Management Standards on the Microsoft Trust Center.

Note The Service Trust Portal provides independent, third-party audit reports and other related documentation. You can use the portal to download and review this documentation for assistance with your own regulatory requirements. For more information on how to use the Service Trust Portal, see article Get started with the Microsoft Service Trust Portal on the Microsoft Trust Center.

- The ISO/IEC 27017:2015 "Information technology - Security techniques - Code of practice for information security controls based on ISO/IEC 27002 for cloud services".

Note You can review the Azure ISO/IEC 27017 certificate, assessment report, and statement of applicability on the Service Trust Portal. For more information, see article ISO/IEC 27017:2015 Code of Practice for Information Security Controls on the Microsoft Trust Center.

The Microsoft Security Policy for Microsoft Azure is written according this international standard, and provides additional controls to address cloud-specific information security threats and risks referring to clauses 5 to 18 in ISO/IEC 27002:2013 "Information technology — Security techniques — Code of practice for information security controls" for controls, implementation guidance, and other information. Specifically, this standard provides guidance on 37 controls in ISO/IEC 27002, and it also features 7 new controls that are not duplicated in ISO/IEC 27002. These new controls address the following important areas:

  - Shared roles and responsibilities within a cloud computing environment.
  - Removal and return of cloud service customer assets upon contract termination.
  - Protection and separation of a customer's virtual environment from that of other customers.
  - Virtual machine (VM) hardening requirements to meet business needs.
  - Procedures for administrative operations of a cloud computing environment.

- Enabling customers to monitor relevant activities within a cloud computing environment.
- Alignment of security management for virtual and physical networks.

The Microsoft Security Policy undergoes a formal management review and update process at a regularly scheduled interval not to exceed 1 year. Changes to business or regulatory requirements, emerging technologies, or responses to security incidents or newly identified threats may also result in ad hoc reviews and updates.

Moreover, as part of the Cloud Security Alliance (CSA) STAR Self-Assessment, Microsoft publishes both a Cloud Control Matrix (CCM)-based report and a Consensus Assessments Initiative Questionnaire (CAIQ) for Azure to indicate its compliance with CSA best practices.

Note You can review the Azure standard response for request for information, CAIQ, and responses to the CSA CAIQ v3.0.1 on the Service Trust Portal. For more information, see article Cloud Security Alliance (CSA) STAR Self-Assessment on the Microsoft Trust Center.

Microsoft establishes and institutionalizes contact with selected groups and associations within the security community to facilitate ongoing security education and training for organizational personnel.

Microsoft Azure partners with the Microsoft Trustworthy Computing (TwC) Group to maintain contact with external parties such as regulatory bodies, service providers, and industry forums to ensure appropriate action can be quickly taken and advice obtained when(ever) necessary.

Privacy

You must be able to trust that the privacy of your data will be protected and that it will be used only in ways that are consistent with your expectations. The Microsoft Privacy Statement describes the specific privacy policy and practices that pertain to customer data in Microsoft Azure. Microsoft was also the first major CSP to adopt the first international code of practice for cloud privacy, ISO/IEC 27018:2014 "Information technology -- Security techniques -- Code of practice for protection of personally identifiable information (PII) in public clouds acting as PII processors".

Note You can review the Azure ISO/IEC 27018 certificate, assessment report, and statement of applicability on the on the Service Trust Portal. For more information, see article ISO/IEC 27018 Code of Practice for Protecting Personal Data in the Cloud on the Microsoft Trust Center.

Furthermore, since May 25, 2018, a European privacy law, the General Data Protection Regulation (GDPR) takes effect. The GDPR imposes new rules on companies, government agencies, non-profits, and other organizations that offer goods and services to people in the European Union (EU), or that collect and analyze data tied to EU residents. The GDPR applies no matter where you are located.

Microsoft believes the GDPR represents an important step forward for individual privacy rights. It gives EU residents more control over their "personal data" (which is precisely defined by the GDPR). The goals of the GDPR are consistent with Microsoft's long-standing commitment to principles we are discussing here.

You can find Microsoft's contractual commitments with regard to the GDPR in the Online Services Terms (OST). The GDPR Terms commit Microsoft to the requirements on processors in GDPR Article 28 and other Articles of GDPR.

Note The GDPR Terms are in Attachment 4 to the Online Services Terms, at the end of the document.

The GDPR accountability documentation provides information about the capabilities of Microsoft Azure you can use to address specific requirements of the GDPR and to support your GDPR accountability for Data Subject

Requests (DSRs), data breach notification, and Data Protection Impact Assessments (DPIAs). It will help to your GDPR accountability, and to your understanding of the technical and organizational measures Microsoft has taken to support the GDPR.

Note For more information, see article The General Data Protection Regulation (GDPR) on the Microsoft Trust Center.

Transparency

As an hyperscale cloud, Microsoft Azure provide a global infrastructure as already introduced. Most Azure services enable you to specify the region where your data will be stored. This has key cloud implications for data residency and data sovereignty, as well as the fundamental principles guiding Microsoft's handling of worldwide law enforcement requests for customer data, including CLOUD Act provisions.

Microsoft defines customer data as all data, including text, sound, video, or image files and software that customers provide to Microsoft to manage on customer's behalf through customer's use of Microsoft Azure.

Data residency

Microsoft may replicate to other regions for data resiliency, but Microsoft will not replicate or move customer data outside the Geo. Customers and their end users may move, copy, or access their customer data from any location globally.

Microsoft provides strong customer commitments regarding cloud services data residency and transfer policies:

- **Data storage for regional services.** Most Azure services are deployed regionally and enable the customer to specify the region into which the service will be deployed, e.g., Europe. Microsoft will not store customer data outside the customer-specified Geo except for Azure Databricks (managed Spark), Cloud Services, Cognitive Services, and Preview services. This commitment helps ensure that customer data stored in a given region will remain in the corresponding Geo and will not be moved to another Geo for the majority of regional services, including virtual machines (VMs), storage, etc.
- **Data storage for non-regional services.** Certain Azure services do not enable the customer to specify the region where the services will be deployed.

Resources:

- Microsoft Azure - Where is my customer data? for details about how Microsoft treats customer data.
- Services by region for a complete list of non-regional services.

Data sovereignty

Data sovereignty implies data residency. However, it also introduces rules and requirements that define who has control over and access to customer data stored in the cloud. In many cases, data sovereignty mandates that customer data be subject to the laws and legal jurisdiction of the country in which data resides. These laws can have direct implications on data access even for service troubleshooting or customer-initiated support requests.

You can use Azure public multi-tenant cloud in combination with Azure Stack or other solutions for on-premises and edge solutions to meet your data sovereignty requirements. These additional products can be deployed to put you solely in control of your data, including storage, processing, transmission, and remote access. This is fully aligned with our so-called "Intelligent Cloud, Intelligent Edge" strategy.

Government requests for customer data

Government requests for customer data follow a strict procedure. Microsoft takes strong measures to help protect customer data from inappropriate access or use by unauthorized persons. This includes restricting access by Microsoft personnel and subcontractors and carefully defining requirements for responding to government requests for customer data. Microsoft ensures that there are no back-door channels and no direct or unfettered government access to customer data. Microsoft imposes special requirements for government and law enforcement requests for customer data.

As stated in the Online Services Terms (OST), Microsoft will not disclose customer data to law enforcement unless required by law. If law enforcement contacts Microsoft with a demand for customer data, Microsoft will attempt to redirect the law enforcement agency to request that data directly from the customer. If compelled to disclose customer data to law enforcement, Microsoft will promptly notify the customer and provide a copy of the demand unless legally prohibited from doing so.

Government requests for customer data must comply with applicable laws. A subpoena or its local equivalent is required to request non-content data and a warrant, court order, or its local equivalent is required for content data. Every year, Microsoft rejects a number of law enforcement requests for customer data. Challenges to government requests can take many forms. In many of these cases, Microsoft simply informs the requesting government that it is unable to disclose the requested information and explains the reason for rejecting the request. Where appropriate, Microsoft challenges requests in court.

To verify that Microsoft meets the standards it sets for itself, the Law Enforcement Requests Report that Microsoft publishes twice a year provides extensive information and statistics about how Microsoft has responded to law enforcement requests, US national security orders, and content removal requests.

The Clarifying Lawful Overseas Use of Data Act or CLOUD Act (H.R. 4943) is a United States law that was enacted in March 2018. You should refer to the following blog for more information, as well as the follow-up posting that describes Microsoft's call for principle-based international agreements governing law enforcement access to data. Key points of interest to customers procuring Azure services are captured below.

- The CLOUD Act enables governments to negotiate new government-to-government agreements that will result in greater transparency and certainty for how information is disclosed to law enforcement agencies across international borders.
- The CLOUD Act is not a mechanism for greater government surveillance; it is a mechanism toward ensuring that customer data is ultimately protected by the laws of each customer's home country while continuing to facilitate lawful access to evidence for legitimate criminal investigations. Law enforcement in the U.S. still needs to obtain a warrant demonstrating probable cause of a crime from an independent court before seeking the contents of communications. The CLOUD Act requires similar protections for other countries seeking bilateral agreements.
- While the CLOUD Act creates new rights under new international agreements, it also preserves the common law right of cloud service providers to go to court to challenge search warrants when there is a conflict of laws – even without these new treaties in place.
- Microsoft retains the legal right to object to a law enforcement order in the United States where the order clearly conflicts with the laws of the country where customer data is hosted. Microsoft will continue to carefully evaluate every law enforcement request and exercise its rights to protect customers where appropriate.

- For legitimate enterprise customers, U.S. law enforcement will, in most instances, now go directly to the customer rather than Microsoft for information requests.

Microsoft does not disclose additional data as a result of the CLOUD Act. This law does not practically change any of the legal and privacy protections that previously applied to law enforcement requests for data – and those protections continue to apply. Microsoft adheres to the same principles and customer commitments related to government demands for user data.

Azure offers an unmatched variety of public, private, and hybrid cloud deployment models to address each customer's concerns regarding the control of their data. Customers worldwide expect to be fully in control of protecting their data in the cloud. Azure enables customers to protect their data through its entire lifecycle whether in transit, at rest, or in use (see section § "Data security" below).

Compliance

Microsoft includes in its Online Services Terms (OST) a specific section § "Compliance with Laws":

> "Microsoft will comply with all laws and regulations applicable to its provision of the Online Services, including security breach notification law. However, Microsoft is not responsible for compliance with any laws or regulations applicable to Customer or Customer's industry that are not generally applicable to information technology service providers. Microsoft does not determine whether Customer Data includes information subject to any specific law or regulation. All Security Incidents are subject to the Security Incident Notification terms below.
>
> Customer must comply with all laws and regulations applicable to its use of Online Services, including laws related to privacy, Personal Data, biometric data, data protection and confidentiality of communications. Customer is responsible for determining whether the Online Services are appropriate for storage and processing of information subject to any specific law or regulation and for using the Online Services in a manner consistent with Customer's legal and regulatory obligations. Customer is responsible for responding to any request from a third party regarding Customer's use of an Online Service, such as a request to take down content under the U.S. Digital Millennium Copyright Act or other applicable laws."

As an hyperscale CSP, Microsoft must be able to comply with many regulatory and industry obligations as Microsoft Azure is adopted by many industries around the world, Microsoft's compliance programs as well as its ability to share third party reviews of its capabilities are key to meeting this challenge. Microsoft is committed to respecting and accommodating regional regulatory standards.

To address the needs of customers across regulated markets worldwide, Azure maintains a comprehensive compliance portfolio based on formal third-party certifications and other types of assurance documents to help you meet your own compliance obligations. As of this writing, this portfolio includes 90+ compliance offerings spanning globally applicable certifications, US Government specific programs, industry assurances, and regional / country specific offerings to help you.

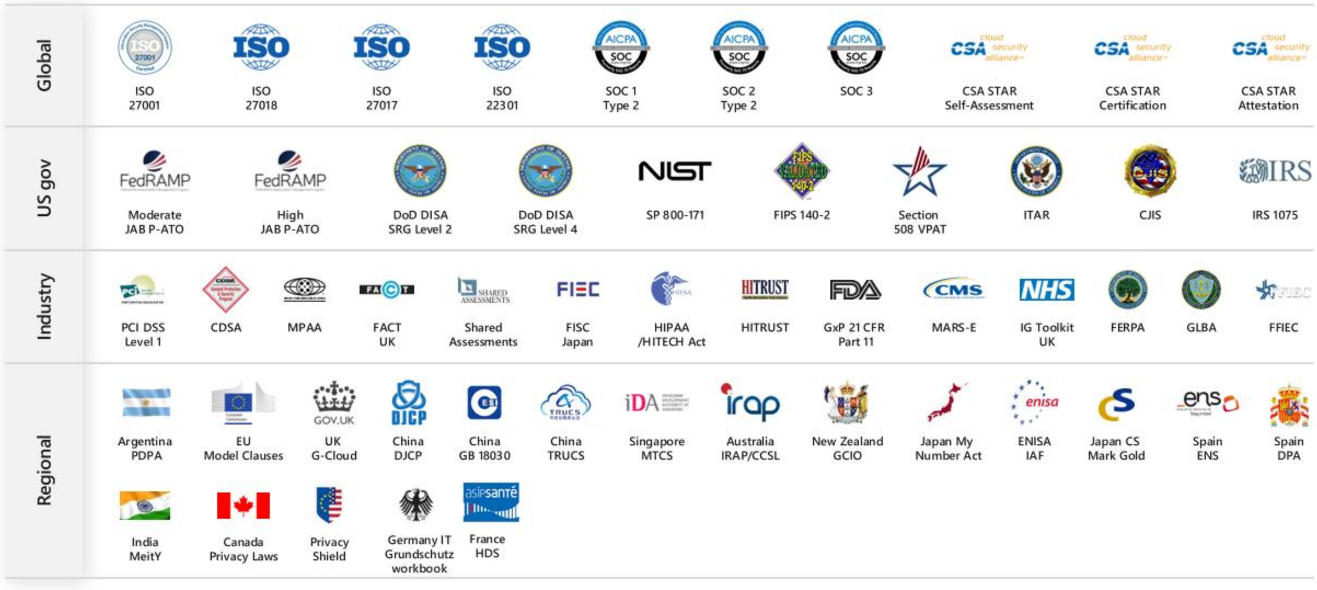


Figure 5 Microsoft offers the most comprehensive set of certifications and attestations of any cloud service provider (as of this writing)

Microsoft offers the most comprehensive set of certifications and attestations of any CSP for a wide range of international, industry and local standards, regulations, legislation and policies. When deploying applications to Azure that are subject to regulatory compliance obligations, you seek assurances that all cloud services comprising the solution be included in cloud service provider's audit scope. Azure offers industry leading depth of compliance coverage judged by the number of cloud services in audit scope for each Azure certification.

A new International Data Corporation (IDC) white paper based on original research by IDC (and sponsored by Microsoft) on Azure customers who are using Azure as a platform to meet regulatory compliance needs stresses that:

> ***Study participants reported use of Azure as a compliance platform helped them carry out their day–to-day compliance responsibilities more effectively. Azure helped them better manage spikes in the workload, enabled faster access to (and analysis of) data during audits, and reduced exposure to risk based on the strong internal controls of Azure.***

Significant IDC findings of research include:

- Five-year return on investment (ROI) of 465 percent, worth an average of $4.29 Million.
- Six-month payback on investment.
- 47 percent reduction in unplanned downtime.
- 35 percent reduction in compliance-related penalties.
- A 24 percent increase in productivity for regulatory compliance teams.

Note Read more about the IDC findings by visiting the article.

You can build and deploy realistic applications and benefit from extensive compliance coverage provided by Azure independent third-party audits. Azure compliance and certification resources are intended to help customers address their own compliance obligations with various regulations. Current certifications and accreditations are listed in the Microsoft Trust Center under the Compliance section. This place aims at helping organization to address a wide range of international, country, and industry-specific regulatory requirements.

> **Note** Azure Stack also provides compliance documentation to help you integrate Azure Stack into solutions that address regulated workloads. As an illustration, the Azure Stack - Cloud Security Alliance (CSA) Cloud Controls Matrix (CCM) v. 3.0.1 Assessment Report includes Azure Stack control mapping to CCM domains and controls.

Compliance Manager is an online tool available from the Microsoft Service Trust Portal that you can use to get insight into how Microsoft implements controls specified in leading standards such as ISO/IEC 27001:2013, ISO/IEC 27018:2014, NIST SP 800-53 Rev 4, and others. For example, Compliance Manager provides insight into Microsoft control implementation and test details for controls that are part of Microsoft responsibility. Moreover, you can use this interactive tool to track progress for control implementations that you own.

Azure Compliance and Security Blueprints is a set of reference architectures with Industry-specific overview and guidance, supporting deployment automation and guidance, security control mappings, customer responsibility matrices to assist you with deploying applications to Azure that meet established compliance standards. Customer responsibility matrices outline which controls are part of customer's responsibility (see section § "Understanding the shared responsibilities' model for your applications" below).

To offer such an industry leading depth of compliance coverage, Microsoft has implemented a common controls framework, which maps and aligns control domains and activities across services, requirements, and operations for each audit, certification, and accreditation.

This mechanism allows to build a 1,000+ controls backbone and is regularly maintained and updated with new controls when new services or standards are incorporated into Microsoft's continuous cloud compliance program.

> **Resources**:
> - Overview of Microsoft Azure compliance.
> - 2-minute video to introduce key Compliance Manager features.
> - Azure Compliance and Security Blueprints.

While security compliance shares many activities with managing security risk, the measure of success for compliance is quite different as illustrated hereafter.

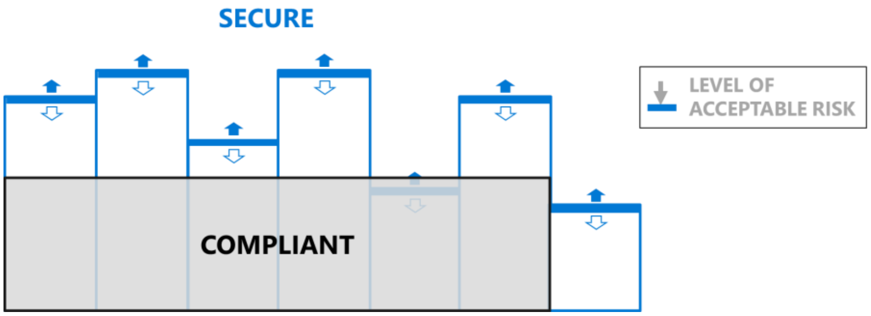


Figure 6 Compliant <> Secure

Security compliance is focused on satisfying regulators and auditors and is typically focused on meeting a very specific set of standards that don't change frequently. While many controls in security standards are relevant to current threats at any point, the standards may also include many requirements that have no effect on current threats and techniques.

In contrast, managing security risk requires mitigating actual and anticipated risks to a specific organization. This is frequently a very dynamic endeavor as there may be frequent changes in the adversaries of concern, the techniques they use, the controls available, and the effectiveness of those controls. See section § "Microsoft Security standards and related practices" in the Appendix.

Enough considerations regarding Azure! Let's dive in and see what you can do by taking advantage of the always growing capabilities of Azure.

Rehosting your applications

The "Lift-and-shift" (*Rehost*) corresponds to the situation where your application is migrated into the Azure IaaS while avoiding modifications to the system as much as possible.

What can Azure do for the "Lift-and-shift" of your existing applications?

When you're not planning to redevelop or rearchitect your current applications but are instead setting up a "lift-and-shift" migration to the cloud, the Azure Migrate service could be especially useful.

As such, the service helps you streamline your migration journey to the Azure public cloud. It helps you iscover, assess, and migrate all your on-premises applications, and related infrastructure, and data.

It not only maps the current environment to the Azure Virtual Machine instances (see nection section) - this mapping helps in figuring out the expected costs of running the infrastructure in the Azure cloud -, but also reports on potential compatibility issues, with guidelines for remediating them.

After Azure Migrate provides its assessment, you can use other services, such as Azure Site Recovey (see section § "Disaster recovery" below) and Azure Database Migration Service, to migrate your applications to Azure.

Beyond this service, **let's consider the various virtualization capabilities provided by Azure, starting by the compute ones.**

Leveraging compute virtualization capabilities

Azure supports a wide range of computing solutions for development and testing, running your applications as part of a "Lift-and-shift" effort, and/or for extending your on-premises datacenter.

Azure Compute is an on-demand computing service for running cloud-based applications. It provides computing resources such as disks, processors, memory, networking and operating systems. The resources are available on-demand and can typically be made available in minutes or even seconds.

Virtual machines (VMs) are one of the three common service types for performing compute with Azure Compute: virtual machines, the other service types being containers (see section § "Leveraging containerization" below), Azure App Services, and Serverless Computing (see section § "Going serverless" below).

Note For a full list of compute services available with Azure and the context on when to use them, see page Compute.

Virtual machines (VMs) are software emulations of physical computers. They include a virtual processor, memory, storage, and networking resources. They host an operating system (OS), and you are able to install and run software just like a physical computer. The use and the control of a VM is typically done through a remote connection client such as SSH a client for Linux-based VMs.

Virtual machines (VMs) support in Azure comprises the following services and capabitilites:

- Azure Virtual Machines.
- Azure Virtual Machine Scale Set.

- Azure Batch.

The next sections depict each of them in order.

Azure Virtual Machines

Hosting your application in a (series of) VM(s) in Azure Virtual Machines provides you with a lot of control over how you host your application. However, You're however responsible for maintaining the environment, including patching the operating system (OS), a Linux distro of your choice in the context of this paper, and keeping antivirus/antimalware programs up to date.

Azure Virtual Machines provide various types, sizes and options for the VMs you may use to run your Linux applications and workloads.

Type	Sizes	Description
General purpose	B, Dsv3, Dv3, Dasv3, Dav3, DSv2, Dv2, Av2, DC	Balanced CPU-to-memory ratio. Ideal for testing and development, small to medium databases, and low to medium traffic web servers.
Compute optimized	Fsv2	High CPU-to-memory ratio. Good for medium traffic web servers, network appliances, batch processes, and application servers.
Memory optimized	Esv3, Ev3, Easv3, Eav3, Mv2, M, DSv2, Dv2	High memory-to-CPU ratio. Great for relational database servers, medium to large caches, and in-memory analytics.
Storage optimized	Lsv2	High disk throughput and IO ideal for Big Data, SQL, NoSQL databases, data warehousing and large transactional databases.
GPU	NC, NCv2, NCv3, ND, NDv2 (Preview), NV, NVv3	Specialized virtual machines targeted for heavy graphic rendering and video editing, as well as model training and inferencing (ND) with deep learning. Available with single or multiple GPUs.
High performance compute	HB, HC, H	Our fastest and most powerful CPU virtual machines with optional high-throughput network interfaces (RDMA).

Note Support for generation 2 virtual machines (VMs) is now available in preview in Azure. Generation 2 VMs support key features that aren't supported in above generation 1 VMs. These features include increased memory, Intel Software Guard Extensions (Intel SGX), and virtualized persistent memory (vPMEM). Generation 2 VMs use the new UEFI-based boot architecture rather than the BIOS-based architecture used by generation 1 VMs. Compared to generation 1 VMs, generation 2 VMs might have improved boot and installation times

Resources:

- Sizes for Linux virtual machines in Azure for information about available sizes and options for the various VM type.
- Virtual Machines Pricing for information about pricing of the various sizes.
- Products available by region for availability of VM sizes in Azure regions.
- Azure subscription and service limits, quotas, and constraints to see general limits on Azure VMs.
- Azure compute units (ACU) to compare compute performance across Azure SKUs.
- Support for generation 2 VMs (preview) on Azure.

Shared Image Gallery provides an Azure-based solution to make the custom management of VM managed images easier in Azure - A managed image is a copy of either a full VM (including any attached data disks) or just the OS disk, depending on how the image has been created -. As such, Shared Image Gallery provides a simple way to share applications with others in your organization, within or across regions, enabling to expedite regional expansion or DevOps processes, simplify a cross-region HA/DR setup and more. Shared Image Gallery also enables to quickly deploy thousands of VMs concurrently from a custom image.

Moreover, Azure Virtual Machines provide you with a lot of control over how you host your application. However, you're responsible for maintaining the environment, including patching the OS and keeping antivirus programs up to date.

Azure AD authentication can be used to improve the security of Linux VMs in Azure, see section § "Identity and access management" below. There is then no need to create local administrator accounts and manage credential lifetime. When you integrate with Azure AD, you centrally control and enforce policies that allow or deny access to the VMs.

With Role-Based Access Control (RBAC), you can specify who can sign in to a given VM as a regular user or with administrator privileges. When users join or leave your team, you can update the RBAC policy for the VM to grant access as appropriate. This experience is much simpler than having to scrub VMs to remove unnecessary SSH public keys.

Moreover, Azure Bastion is a newly introduced fully platform-managed PaaS service that you provision inside your VNET. Azure Bastion provides secure and seamless SSH access to your VMs directly through the Azure Portal. Azure Bastion is provisioned directly in your VNet and supports all VMs in your virtual network (see below) using SSL without any exposure through public IP addresses.

In addition, listed below are key enabling technologies and services that you may find helpful when deploying sensitive data and workloads on VM in Azure:

- Azure Dedicated Hosts via Azure E64is v3, E64i v3, GS5, G5, DS15 v2, and D15 v2 VM instances allow customers to be deployed on hardware dedicated to a single customer. Using isolated VMs guarantees that customer VM will be the only one running on that specific server node.
- Azure Confidential Computing offers encryption of data while in use, ensuring that data is always under customer control. Data is protected inside a Trusted Execution Environment (TEE) and there is no way to view data or operations from outside the enclave (see section § "Encryption in use" below). Azure DC-series virtual machine (VM) enable the latest generation of Intel Xeon Processors with Intel SGX technology to the Azure cloud. Using these new VMs to build applications that protect data and code in use.
- Accelerated FPGA networking based on Azure SmartNICs enables you to offload host networking to dedicated hardware, enabling tunneling for VNETs, security, and load balancing. Offloading network traffic to a dedicated chip guards against side-channel attacks on the main CPU.
- Just-in-time (JIT) virtual machine (VM) access can be used to lock down inbound traffic to Azure VMs by creating network security group (NSG) rules (see section § "Network security" below). Customer selects ports on the VM to which inbound traffic will be locked down and when a user requests access to a VM, Azure Security Center checks that the user has proper role-based access control (RBAC) permissions.

You can run single VMs for testing, development, or minor tasks, or group VMs together to provide high availability, scalability, and redundancy (see section § "What can Azure do for high-availability?" below). Azure has several features so that no matter what uptime requirements are, Azure can meet them. These features notably include virtual machine scale sets and Azure Batch.

Azure Virtual Machine Scale Set

Azure Virtual Machine Scale Sets are an Azure compute resource that you can use to deploy and manage a set of identical VMs. With all VMs configured the same, VM scale sets are designed to support true auto-scale - no preprovisioning of VMs is required - and as such makes it easier to build large-scale services targeting big compute, big data, and containerized workloads.

So, as demand goes up more VM instances can be added, and as demand goes down VM instances can be removed. The process can be manual, automated, or a combination of both.

Resource:

- What are virtual machine scale sets?.

Azure Batch

If you need to run large-scale batch or high-performance computing (HPC) applications on Azure, you can use Azure Batch.

Azure Batch creates and manages a collection of thousands of VMs, installs the applications you want to run, and schedules jobs on the VMs. They don't need to deploy and manage individual VMs or server clusters; Batch schedules, manages, and auto- scales your jobs so you use only the VMs you need.

Batch is well suited to run parallel workloads at scale.

Resource:

- What is Azure Batch?.

Leveraging network virtualization capabilities

Azure networking components offer a range of functionality and services that can help you design and build cloud infrastructure services that meet your requirements.

Note For a full list of networking services available with Azure, and context on when to use them, see page Networking.

Azure Virtual Network

Azure Virtual Network (VNET) is the fundamental building block for your private network in Azure. VNET enables many types of Azure resources, such as Azure Virtual Machines (VM) (see section § "Leveraging compute virtualization capabilities" above), to securely communicate with each other, the internet, and on-premises networks. Subnets allow to segment a VNet address space into segments that are appropriate for the organization's internal network.

VNET is similar to a traditional network that you'd operate in your own datacenter but brings with its additional benefits of Azure's infrastructure such as scale, availability, and isolation.

In a VNET, customer can run your favorite network virtual appliances (NVAs) - WAN optimizers, load balancers, and application firewalls - and define traffic flows, allowing them to design your network with a greater degree of control.

Network security capabilities of NVAs include:

- Firewalling,
- Intrusion detection/intrusion prevention,
- Vulnerability management,
- Application control,
- Network-based anomaly detection,

- Web filtering,
- Antivirus,
- Botnet protection.

These Industries best-of-breed NVAs allow you to bring the networking, security, and other functions of your favorite provider to Azure for a familiar experience using skills your team already has. NVAs support network functionality and services in the form of VM images from the Azure Marketplace.

A10
Array Networks
aviatrix
Barracuda
BROCADE
Check Point Software Technologies Ltd.
cisco
CITRIX
COATS
f5
FORTINET
IMPERVA
Infoblox
KEMP
NetApp
Netgate
paloalto networks
riverbed Think fast.
TREND MICRO

VNET peering enables to seamlessly connect VNETs. Once peered, the VNETs appear as one, for connectivity purposes. The traffic between VMs in the peered VNETs is routed through the Microsoft backbone infrastructure, much like traffic is routed between VMs in the same VM, through private IP addresses only.

Azure supports:

- **VNet peering**. Connecting VNets within the same Azure region.
- **Global VNet peering**. Connecting VNets across Azure regions.

Resource:

- What is Azure Virtual Network?.
- Virtual network peering.

Azure Load balancer

Azure Load Balancer can provide scale for applications and create high availability for your services. Load Balancer supports inbound and outbound scenarios, provides low latency and high throughput, and scales up to millions of flows for all Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) applications.

You can use Load Balancer with incoming internet traffic, internal traffic across Azure services, port forwarding for specific traffic, or outbound connectivity for VMs in your VNETs.

Resource:

- What is Azure Load Balancer?.

Azure Private DNS

Azure Private DNS provides a reliable, secure DNS service to manage and resolve domain names in a VNET without the need to add a custom DNS solution.

By using private DNS zones, you can use your own custom domain names rather than the Azure-provided names available today. Using custom domain names helps you to tailor your virtual network architecture to best suit your organization's needs. It provides name resolution for VMs within a VNET and between VNETs. Additionally, you can configure zones names with a split-horizon view, which allows a private and a public DNS zone to share the name.

Resource:

- What is Azure Private DNS?.

Azure DNS

Azure DNS is a hosting service for DNS domains that provides name resolution by using Microsoft Azure infrastructure. As such, Azure DNS uses a global network of name servers to provide fast responses to DNS queries. Microsoft uses Anycast networking, so DNS queries automatically route to the closest name servers to give you the best possible performance.

In addition, by hosting your domains in Azure, you can manage their DNS records by using the same credentials, APIs, tools, and billing as your other Azure services. Azure DNS seamlessly integrates Azure-based services with corresponding DNS updates and streamline your end-to-end deployment process.

Resource:

- What is Azure DNS?.

Connecting your virtual networks to your on-premises resources

A VNET also provides a means for Azure Virtual Machines to act as part of a your internal (on-premises) network. It expands the nature of intranet connectivity beyond a single cloud service to include any set of internal addresses of other cloud services on Azure or other machines on a customer's own network (presumably behind the customer's datacenter firewall).

With VNET, you choose the address ranges of non-globally routable IP addresses to be assigned to the VMs so that they will not collide with addresses you are using elsewhere. A cryptographically protected "tunnel" is established between Azure and your internal network, allowing the VM(s) to connect to your back-end resources as though it was directly on that network.

Azure Express Route

Most customers will extend their on-premises networks to Azure using Azure ExpressRoute, though Site-to-Site (S2S) IPsec virtual private network (VPN) may also be used (see next section).

Note For more information about connecting an on-premises network to Azure, see article Choose a solution for connecting an on-premises network to Azure.

ExpressRoute is a streamlined solution for establishing a secure private connection facilitated by a connectivity provider between customer infrastructure and Azure datacenters.

Connectivity can be from an any-to-any (IP VPN) network, a point-to-point Ethernet network, or a virtual cross-connection through a connectivity provider at a co-location facility. ExpressRoute connections do not go over the public Internet. This allows ExpressRoute connections to offer more reliability, faster speeds, consistent latencies, and higher security than typical connections over the Internet.

The termination point of ExpressRoute private peering can affect firewall capacity, scalability, reliability, and network traffic visibility:

- **Terminate outside the firewall** (DMZ Paradigm). If you require visibility into the traffic, you can continue an existing practice of isolating datacenters, or if they are solely putting extranet resources on Azure.
- **Terminate inside the firewall** (Network Extension Paradigm). Azure is treating as a N^th datacenter.

Resources:
- What is ExpressRoute?.
- ExpressRoute connectivity models.

Azure VPN Gateway

VPN connectivity can instead use the Azure VPN Gateway, an Industry-standard Site-to-Site (S2S) VPN gateway and Point-to-Site (P2S) VPN access from anywhere to connect customer's infrastructure to the cloud.

An Azure VPN Gateway can also be referred to as a VNET gateway, but a VPN gateway is a specific type of VNET gateway that is used to send encrypted traffic between an VNET and an on-premises location over the public internet.

It connects your on-premises networks to Azure through Site-to-Site (S2S) VPNs in a similar way that you set up and connect to a remote branch office. The connectivity is secure and uses the industry-standard protocols Internet Protocol Security (IPsec) and Internet Key Exchange (IKE).

It also allows to connect to VNETs from anywhere: Point-to-Site (P2S) VPN lets connect to VMs on VNETs from virtually anywhere.

Resources:
- What is VPN Gateway?
- About VPN devices and IPsec/IKE parameters for Site-to-Site VPN Gateway connections. (This article provides the list of IPsec/IKE parameters for Azure VPN gateways, and a list of validated VPN devices connecting to Azure VPN gateways)

Azure Hybrid Connections

Azure Hybrid Connections can be used to access application resources in other networks. Those other networks can be in Azure, on-premises or networks in other CSPs.

Each hybrid connection correlates to a single TCP host and port combination. This means that the hybrid connection endpoint can be on any operating system and any application, provided you are accessing a TCP listening port. The Hybrid Connections feature does not know or care what the application protocol is, or what you are accessing. It is simply providing network access between other networks and Azure.

Resource:
- Azure App Service Hybrid Connections.

Leveraging storage virtualization capabilities

Note For a full list of storage services available with Azure, and context on when to use them, see page Storage.

Azure Disk Storage

Azure Disk Storage is specifically meant for high I/O performance and provides disks for VMs, applications, and other services to access and use as they need, similar to how they would in on-premises scenarios. Disk storage allows data to be persistently stored and accessed from an attached virtual hard disk. The disks can be managed or unmanaged by Azure, and therefore managed and configured by the user.

> **Note** Managed disks provide better reliability for availability sets (see section § "What can Azure do for high-availability?" below) by ensuring that the disks of VMs in an availability set are sufficiently isolated from each other to avoid single points of failure. It does this by automatically placing the disks in different storage fault domains (storage clusters) and aligning them with the VM fault domain. If a storage fault domain fails due to hardware or software failure, only the VM instance with disks on the storage fault domain fails.

Typical scenarios for using disk storage are if a customer want to "lift-and-shift" applications that read and write data to persistent disks, or they you are storing data that is not required to be accessed from outside the VM to which the disk is attached.

Disks come in many different sizes and performance levels, from solid-state drives (SSDs) to traditional spinning hard disk drives (HDDs), with varying performance abilities.

> **Resources**:
> - What disk types are available in Azure?.
> - Introduction to Azure managed disks.

Azure Files

Azure Files offers fully managed file shares in the cloud that are accessible via the industry standard Server Message Block (SMB) protocol. Azure file shares can be mounted concurrently by cloud or on-premises deployments of Linux.

Applications running in Azure Virtual Machine (see above eponym section) or cloud services in Azure can mount a file storage share to access file data, just as a desktop application would mount a typical SMB share. Any number of VMs or roles in Azure can mount and access the file storage share simultaneously. Typical usage scenarios would be to share files anywhere in the world, diagnostic data, or application data sharing.

> **Resources**:
> - What is Azure Files?.
> - Use Azure Files with Linux.

Building new Cloud-native applications

Given the rate with which technology is changing, we believe microservices adoption will continue to grow briskly because of its promise of speed and scalability.

IDC predicted last year that "**by 2022, 90% of all apps will feature microservices architectures that improve the ability to design, debug, update, and leverage third-party code; 35% of all production apps will be cloud-native**. The digital economy's requirement to deliver high-quality applications at the speed of business is driving the shift to "hyperagile apps" – highly modular, distributed, continuously updated, and leveraging cloud-native technologies such as containers and serverless computing."

In 2019, a lot of customers are already going to be scrambling to figure out how to get their virtual machine (VM) infrastructure refactored for the realities of supporting microservices.

What can Azure do for your new applications?

A microservices-based application separates functionality into smaller services called microservices. According to the definition from James Lewis et Martin Fowler, "the microservices architectural style is an approach to developing a single application as a suite of small services, each running in its own process and communicating with lightweight mechanisms, often an HTTP resource API. These services are built around business capabilities and independently deployable by fully automated deployment machinery. There is a bare minimum of centralized management of these services, which may be written in different programming languages and use different data storage technologies".

In other words, microservices architecture means breaking large software projects into smaller, independent, and loosely coupled modules. When using microservices, each service is independent, and each service is a new project that can be developed on its own schedule, using any language or stack that best fits current requirements. Microservices make application maintenance easier: developers work on individual services, so the code a developer needs to handle is smaller, easier to manage and easier to understand.

The microservices approach scales out by deploying each service independently, creating instances of these services across servers, virtual machines (VMs) or containers.

Microsoft Azure helps you implement a microservices-based, cloud-native architecture that makes it easier to develop and scale your applications.

Leveraging containerization

"Containerization" is one of those technology buzzwords flying around in the news. But containers are more than just buzz - they're actually very useful for running your applications.

Containers are a virtualization environment. However, unlike VMs, they do not include an operating system (OS). Instead, they reference the OS of the host environment that runs the container. Containers are meant to be lightweight and are designed to be created in a few seconds, scaled out, and stopped dynamically. This allows you to respond to changes on demand and quickly restart in case of a crash or hardware interruption.

Containers also offer tremendous portability, which makes them ideal for developing an application locally on your machine and then hosting it in the cloud, in test, and later in production.

As such, containers are often used to create solutions using a microservice architecture. This is where you break solutions into smaller, independent pieces. This allows you to separate portions of your applications into logical sections that can be maintained, scaled, or updated independently. (see above)

There are many technologies for running containers, including Docker. Azure can run and manage containers with Azure Container Instances and Azure Kubernetes Service. (You can also run containers in Web Apps for Containers, Azure Service Fabric, and in Azure Batch (see eponym section above)).

Azure Container Instances

Azure Container Instances (ACI) offer the fastest and simplest way to host and run a container in Azure. You don't have to manage any VMs or configure any additional services. It is a PaaS offering that allows them to upload your containers and execute them directly. ACI provides fast, isolated compute to meet traffic that comes in spikes, without the need to manage servers.

The ACI service is billed per second, per virtual CPU, per gigabyte, or by memory.

Resource:

- What is Azure Container Instances?.

Azure Kubernetes Service

Azure Kubernetes Service (AKS) is a variation of the Kubernetes Orchestrator as a managed service in Azure. It is a complete orchestration service for containers with distributed architectures and large volumes of containers.

AKS makes it simple easy to deploy a Kubernetes infrastructure by delegating to Azure the management tasks of the cluster masters. All you have to do is focus on managing the agent nodes. The creation of an AKS cluster can be done simply at the portal or from the command line, configuration of Kubernetes masters and nodes being automatically supported.

This means you can use your existing skills to manage and deploy (microservices-based) applications that run in containers on Azure.

See section § "Scaling and orchestrating " below.

Resource:

- Introduction to Azure Kubernetes Service.

Azure Container Registry

Azure Container Registry (ACR) allows you to store images for all types of container deployments including Kubernetes, Docker Swarm, and Azure services such as App Service, Batch, and others. Your DevOps team can manage the configuration of apps isolated from the configuration of the hosting environment.

This is a highly available and secure storage service, specifically built to store container images. This is great for storing a customer's private Docker images. Images can be pushed to and pulled from an ACR instance. From there, Helm can use it to compile the image into a package that can be deployed on AKS.

You can also use ACR for your existing container development and deployment pipelines.

Resource:

- Introduction to private Docker container registries in Azure.

Azure Service Fabric

Another way to run containers in Azure is with Azure Service Fabric. Azure Service Fabric is a distributed systems platform that makes it easy to package, deploy, and manage scalable and reliable microservices and containers.

Service Fabric is Microsoft's container orchestrator for deploying microservices across a cluster of machines. Service Fabric benefits from the lessons learned during its years running services at Microsoft at massive scale. It powers all the core Azure infrastructure. This is actually the service that runs many of the Azure services inside Microsoft, like above Azure App Service.

Service Fabric is an open source project on GitHub. It provides an application model in which a container represents an application host in which multiple service replicas are placed. Service Fabric also supports a guest executable scenario in which you don't use the built-in Service Fabric programming models but instead package an existing application, written using any language or framework, inside a container. This scenario is the common use-case for containers. You can also run Service Fabric services inside a container.

You can also use Azure Service Fabric Mesh to run containers on a Service Fabric cluster that Microsoft manages for them as a service.

Resource:

- Service Fabric and containers.
- Create your first Service Fabric container application on Linux.
- What is Service Fabric Mesh?.

Azure App Service

Azure App Service is a fully managed compute platform that is optimized for hosting websites and web applications. You can use Azure App Service on Linux to host web apps natively on Linux for supported application stacks.

As part of it, Web App for Containers helps you easily deploy and run containerized web apps at scale. Just pull container images from Docker Hub or a private ACR instance, and Web App for Containers will deploy the containerized application along with your preferred dependencies to production in seconds.

The platform automatically takes care of OS patching, capacity provisioning, and load balancing.

Eventually, you can use an Azure App Service Environment to both afford you a very high scale and gives you control over isolation and network access.

Resource:

- Introduction to Azure App Service on Linux.
- Making it easier to bring your Linux based web apps to Azure App Service.

Scaling and orchestrating containers

In terms of containers orchestration, Kubernetes (K8s) has become the de facto standard over the last few years.

Note For more information on Kubernetes support in Azure, see page Kubernetes.

Leveraging fully managed services

Azure Kubernetes Service (AKS) is a variation of the Kubernetes orchestrator as a highly available, secure, and fully managed Kubernetes service in Azure.

AKS is a complete orchestration service for containers with distributed architectures and large volumes of containers, and makes it simple to create, configure, and manage a cluster of VMs that are preconfigured to run containers. This means you can use your existing skills to manage and deploy applications that run in containers on Azure.

AKS reduces the complexity and operational overhead of managing a Kubernetes cluster by offloading much of that responsibility to Azure. **As a hosted Kubernetes service, Azure handles critical tasks like health monitoring and maintenance.**

You pay only for the agent nodes within your clusters, not for the masters. **As a managed Kubernetes service, AKS provides automated Kubernetes version upgrades and patching, easy cluster scaling, a self-healing hosted control plane (masters), and cost savings, since you only pay for running agent pool nodes.**

With Azure handling the management of the nodes in your AKS cluster, there are many tasks that you don't have to perform manually, such as cluster upgrades. Because Azure handles these critical maintenance tasks for you, AKS does not provide direct access (such as with SSH) to the cluster.

In addition, from a security standpoint, a user-defined network policy feature in AKS enables secure network segmentation within Kubernetes and allows cluster operators to control which pods can communicate with each other and resources outside the cluster.

Network policy is generally available through the Azure native policy plug-in or through the community project Calico.

Note For additional information, see the article Integrating Azure CNI and Calico: A technical deep dive.

Note Already available on Azure, AKS is expected later this calendar year 2019 on Azure Stack. As of this writing, you can install Kubernetes using ARM templates (see section § "Are you saying "Infrastructure as Code"?" below) generated by the Azure Container Service (ACS)-Engine on Azure Stack, but any certified Kubernetes distribution will work. (Kubeadm for example is a simple tool to deploy a Kubernetes cluster and is supported by Kubernetes). For more information, see article Azure Kubernetes Service (AKS) on Azure Stack.

Resource:

- Introduction to Azure Kubernetes Service.

Kubernetes applications' connection to Azure Services with Open Service Broker API

You need an easy way to connect your containers to Azure services. Microsoft open sourced the Open Service Broker for Azure (OSBA) built using the Open Service Broker API.

Note The Open Service Broker API is an industry-wide effort to meet that demand, simply and securely. The Open Service Broker API provides a standard way for service providers to expose backing services to applications running in cloud native platforms like Kubernetes.

OPEN SERVICE BROKER API™

In a multi-cloud, multi-platform world, developers want a standard way to connect their applications to the wealth of services available in the marketplace.

OSBA is the simplest, most flexible way to connect Kubernetes applications to a suite of the most popular Azure services, from standard offerings like Azure Database for MySQL, a fully-managed service for MySQL, to unique Azure solutions like Azure Cosmos DB, a globally distributed multi-model database, that offers an API for MongoDB is compatible with the MongoDB's wire protocol (see section § "Using (short term) storage for your data" below). Microsoft currently offer 14 Azure services and intend to support most others over the next year.

With OSBA and the Kubernetes Service Catalog, you can manage these SLA-backed Azure data services via the Kubernetes API, making it easy for you to use Azure's data stores in a Kubernetes-native way. To showcase OSBA, we've adapted some of the most popular Helm charts to leverage Azure services.

Additionally, Microsoft contributed on a release of a Command Line Interface (CLI) for the Kubernetes Service Catalog. This helps cluster administrators and application developers request and use services exposed through the Kubernetes Service Catalog.

Resource:

- Integrate with Azure-managed services using Open Service Broker for Azure (OSBA). (This article is part of the AKS documentation)

Kubernetes applications' test and iterative development without replicating or mocking dependencies

Developing a Kubernetes application can be challenging. You need Docker and Kubernetes configuration files. You need to figure out how to test your application locally and interact with other dependent services. You might need to handle developing and testing on multiple services at once and with a team of developers.

Azure Dev Spaces provides a rapid, iterative Kubernetes development experience for teams in AKS clusters. You can collaborate with your team in a shared AKS cluster. Azure Dev Spaces also allows you to test all the components of your application in AKS without replicating or mocking up dependencies. You can iteratively run and debug containers directly in AKS with minimal development machine setup.

Azure Dev Spaces helps teams to focus on the development and rapid iteration of their microservice application by allowing teams to work directly with their entire microservices architecture or application running in AKS. Azure Dev Spaces also provides a way to independently update portions of your microservices architecture in isolation without affecting the rest of the AKS cluster or other developers. Azure Dev Spaces is for development and testing in lower-level development and testing environments and is not intended to run on production AKS clusters.

Azure Dev Spaces provides tooling to generate Docker and Kubernetes assets for your projects. This tooling allows you to easily add new and existing applications to both a dev space and other AKS clusters.

Resource:

- What are Azure Dev Spaces?.
- How Azure Dev Spaces works and is configured.

Serverless containers management using Kubernetes with the virtual Kubelet

Back in 2017, Microsoft released the Azure Container Instances (ACI) Connector for Kubernetes, an experimental open-source project to extend Kubernetes with ACI, a serverless container runtime that provides per-second billing and no VM management, see section § "Leveraging containerization" above.

A new version of the above Kubernetes connector, i.e. the Virtual Kubelet, is available and can be used by customers to target ACI or any equivalent runtime.

The Virtual Kubelet open source project from the Cloud Native Computing Foundation (CNCF) is an open source Kubernetes kubelet implementation that masquerades as a kubelet for the purposes of connecting Kubernetes to other APIs. It features a pluggable architecture that supports a variety of runtimes, and uses existing Kubernetes primitives, making it much easier to build on.

Resources:
- Virtual kubelet introduction.
- Use Virtual Kubelet with Azure Kubernetes Service (AKS).

Cloud Native Application Bundle (CNAB) packaging

One of the critical problems solved by containers is the hermetic packaging of a binary into a package that is easy to share and deploy around the world. But a Cloud-native application is more than a binary, and this is what led to the co-development, with HashiCorp and others, of the Cloud Native Application Bundle (CNAB) specification to reduce the complexity of running multiple services packaged together.

Note Cloud-native applications are based on the above microservices architecture, use managed services, and take advantage of continuous delivery to achieve reliability and faster time to market. They offer you greater agility, resilience, and portability across clouds. Cloud-native is a way of approaching the development and deployment of applications in a way that takes full account of the characteristics and nature of the cloud – resulting in applications and workflows that unlock all cloud benefits.

If you are looking to get the most from the cloud and tap into advanced capabilities like improved resiliency, global scale or maximum agility, this route allows to embrace new and more relevant standards, e.g. the Cloud-native computing foundation (CNCF) to have applications be built from the ground up and optimized for cloud scale and performance.

CNAB can greatly simplify simplifies container-style deployments for distributed application.

“Today if you’re using just container-based applications maybe you’re building Helm artifacts or for Azure you’re targeting an ARM artifact or something like Terraform,” says Gabe Monroy. “The problem comes when the app you’re building is a mix of these things, so it’s got, say, Terraform and containers and functions, because we’re starting to see that diversity in different runtimes and cloud APIs emerge today. How do you wrap your hands around that and turn it into something you can manage like a simple application? Can we offer that familiar experience around repeatability, immutability and cryptographic assurances that the workload hasn’t been modified in a world that’s containers plus… or that doesn’t even include containers at all?”

CNABs allow you to package images alongside configuration tools like Terraform and other artifacts to allow seamlessly deploying an application from a single package (see section § “Are you saying “Infrastructure as Code” below).

Resource:
- Cloud Native application bundles (CNAB).

Implemeting service mesh

After containers and Kubernetes, one of the most important innovations in microservices has been the development of the concept of a service mesh.

In a microservices architecture with multiple discrete services, all of them are communicating over a network. This brings multiple issues to address, but two major ones are service discovery and security. Service Mesh helps with discovering services, advertising address/port to communicate to and routing inside the different services. Service Mesh security also includes service authentication and traffic encryption by the use of certificates.

Earlier this year, Microsoft partnered with HashiCorp and others to announce the release of Service Mesh Interface (SMI), a collaborative, implementation agnostic API for the configuration and deployment of service mesh technology.

We collaborated with HashiCorp to produce a control rules implementation of the traffic access control (TAC) using Consul Connect.

HashiCorp Consul is an open-source service mesh that provides a key set of functionality across the microservices in a Kubernetes cluster. These features include service discovery, health checking, service segmentation, and observability. In short, it provides a solution to simplify and secure service networking.

Note For more information about Consul, see the official What is Consul? documentation.

HashiCorp Consul can be fully installed into a Kubernetes cluster on AKS. Furthermore, as of this writing, you can take advantage of HashiCorp Consul Services on Azure powered by the Azure Managed Applications platform. With this managed offering, you can focus on the value of Consul while being confident that the experts at HashiCorp are taking care of the management of the service, de facto reducing complexity for you and enabling you to focus on cloud native innovation.

Istio is an open-source service mesh that provides a key set of functionalities across the microservices in a Kubernetes cluster. These features include traffic management, service identity and security, policy enforcement, and observability.

Note For more information about Istio, see article What is Istio?.

Istio can be fully installed into a Kubernetes cluster on AKS. One should also note the availability of an Application Insights adapter for Istio Mixer project's GitHub.

Resources:

- HashiCorp announcement regarding the HashiCorp Consul Service offering on Azure.
- Install and use Consul Connect in Azure Kubernetes Service (AKS).
- Install and use Istio in Azure Kubernetes Service (AKS).

Leveraging a comprehensive set of services

API management

The API management gathers APIs in a central place and offers services like API documentation, authorization, throttling, metering and API facading. One can see a possible overlap with service mesh but service mesh is more about consuming services in an East-West direction, i.e. inside the local network. OSAPI management, on the other, side is more about managing North-South service consumption, i.e. requests coming from the outside, e.g the Internet Edge. This applies for APIs you create yourself as well as those from third-party vendors.

In Azure, you can use Azure API Management. This is basically a proxy you put in front of APIs that adds features like caching, throttling, and authentication or authorization.

With Azure API Management, you secure an API by requiring users to create a subscription to it. This way, applications need to authenticate before they can use your API. You can use various authentication methods like access tokens, basic authentication, and certificates. Additionally, you can track who’s calling your API and block unwanted callers.

While security is critical, Azure API Management offers other capabilities that can help streamline your development and testing workflow, such as test data response mocking, publishing multiple API versions, introducing non-breaking changes safely with revisions, and giving developers access to your API's auto-generated documentation, catalog, and code samples.

Resources:

- About API Management.
- Feature-based comparison of the Azure API Management tiers.

In-memory caching

Every modern application works with data. When you retrieve data from a data store like a database, this typically involves scanning multiple tables or documents in some distant server, weaving the results together, and then sending the result to the requestor.

To eliminate some of these "roundtrips," you can cache data that doesn't change often. This way, instead of querying the database every time, you could retrieve some of the data from a cache, like Azure Cache for Redis, a fully managed, open source compatible in-memory data store to power fast, scalable applications.

As its name indicates, Azure Cache for Redis is based on the popular software Redis and is backed by industry-leading SLAs (see section § "What can Azure do for high-availability?" below).

It is typically used as a cache to improve the performance and scalability of systems that rely heavily on backend data-stores. The benefit of the cache is that it stores data in a simple format, such as key-value. You don't need to run a complex query to get this data. Instead, you just need to know the key to retrieve the value.

Performance is improved by temporarily copying frequently accessed data to fast storage located close to the application. With Azure Cache for Redis, this fast storage is located in-memory with Azure Cache for Redis instead of being loaded from disk by a database.

Azure Cache for Redis can also be used as an in-memory data structure store, a distributed non-relational database, and a message broker. Application performance is improved by taking advantage of the low-latency, high-throughput performance of the Redis engine. Azure Cache for Redis has advanced options like clustering and geo-replication.

Azure Cache for Redis provides you access to a secure, dedicated Redis cache. Azure Cache for Redis is managed by Microsoft, hosted within Azure, and accessible to any application within or outside of Azure. Azure provides Cache-as-a-Service with Redis Cache.

Resource:

- Azure Cache for Redis description.

Message bus

Modern, globally distributed applications often must deal with large amounts of messages coming in, so they need to be designed with decoupling and scaling in mind. As such, a message bus offers a platform for services and applications to exchange messages. This enables asynchronous consumption of services and reduce their mutual dependence, hence bringing decoupling.

Note There are numerous debates on when sending a message is more appropriate than directly consuming a REST API, so really it all depends on the type of application architecture we want to implement. Even though message bus offers a buffer in-between two services, some REST implementations may offer something similar through specific implementations.

Azure provides several services to help with event ingestion and analysis as well as messaging patterns. The core of messaging is the Azure Service Bus that provides reliable cloud messaging as a service (MaaS) and simple hybrid integration.

Microsoft Azure Service Bus is a fully managed enterprise integration message broker. Azure Service Bus supports standard Advanced Message Queueing Protocol (AMQP) 1.0 and HTTP/REST protocols.

Azure Service Bus is most commonly used to decouple applications and services from each other and is a reliable and secure platform for asynchronous data and state transfer. Data is transferred between different applications and services using messages.

> **Note** A message is raw data produced by a service to be consumed or stored elsewhere. The message contains the data that triggered the message pipeline. A message is in binary format, which can contain JSON, XML, or just text. The publisher of the message has an expectation about how the consumer handles the message. A contract exists between the two sides. For example, the publisher sends a message with the raw data, and expects the consumer to create a file from that data and send a response when the work is done.

Some common messaging scenarios are:

- **Messaging**. transfer business data.
- **Decouple applications**. improve reliability and scalability of applications and services (client and service do not have to be online at the same time).
- **Topics and subscriptions**. enable 1:*n* relationships between publishers and subscribers.
- **Message sessions**. implement workflows that require message ordering or message deferral.

As such, Azure Service Bus encompasses a collection of services that you can use for messaging patterns. The most important services are Azure Service Bus queues and topics.

Azure Service Bus queues decouple systems from one another. By decoupling the systems, each one can work at a different speed, and can be scaled individually to the application’s needs.

A Service Bus queue is a simple mechanism. Multiple applications can put messages on the queue, but a queue message can be processed by only one application at a time. There are some clever features to work with messages on the queue, like duplicate detection and a dead-letter queue (DLQ) to which messages are moved when they fail to be processed correctly.

Just like Service Bus queues, Azure Service Bus topics are a form of application decoupling with the following noticeable difference between them:

- With a queue, multiple applications write messages to the queue, but only one application at a time can process a message.
- With a topic, multiple applications write messages to the topic, and multiple applications can process a message at the same time.

In terms of security, Azure Service Bus supports security protocols such as Shared Access Signatures (SAS), Role Based Access Control (RBAC) (see eponym section below), and Managed identities for Azure resources (see eponym section below).

> **Resources**:
> - What is Azure Service Bus?.
> - Compare Azure messaging services.

Going serverless

Serverless computing is a cloud-hosted execution environment that runs your code but abstracts the underlying hosting environment. You create an instance of the service and you add your code. No infrastructure configuration or maintenance is required, or even allowed.

They configure their serverless apps to respond to events. An event could be a REST endpoint, a periodic timer, or even a message received from another Azure service. The serverless app runs only when it's triggered by an event.

Scaling and performance are handled automatically, and you don't even need to reserve resources.

Azure provide a range of serverless execution environments

As such, some of the most common serverless service types in Azure are Azure Functions, Azure Logic Apps, and Azure Event Grid.

Note For a full list of serverless services available with Azure, see page Azure serverless.

Azure Functions

When you are concerned only about the code running your service and not the underlying platform or infrastructure, Azure Functions are ideal.

They're commonly used when you need to perform work in response to an event (often via a REST request), timer, or message from another Azure service, and when that work can be completed quickly, within seconds or less.

Azure Functions scale automatically, and charges accrue only when a function is triggered, so they're a solid choice when demand is variable. They can scale out to accommodate these busier times.

Furthermore, Azure Functions are stateless; they behave as if they're restarted every time they respond to an event. This is ideal for processing incoming data. And if state is required, they can be connected to an Azure storage service.

Durable Functions is an extension of Azure Functions that lets customer write stateful functions in a serverless compute environment. The extension lets them define stateful workflows by writing orchestrator functions and stateful entities by writing entity functions using the Azure Functions programming model. Behind the scenes, the extension manages state, checkpoints, and restarts for you, allowing you to focus on your business logic.

With Azure Functions, it’s possible to pay only for functions that run, rather than having to keep compute instances running all month. This is also called serverless because it only requires you to create your application - you don’t have to deal with any servers or even scaling of servers.

The Functions runtime is open-source and available on GitHub.

Resources:

- An introduction to Azure Functions.
- What are Durable Functions?.

Azure Logic Apps

Azure Logic Apps is a cloud service that helps automate and orchestrate tasks, business processes, and workflows when applications, data, systems, and services need to be integrated across enterprises or organizations. Azure Logic Apps simplifies how you design and build scalable solutions - whether in the cloud, on premises, or both - for app integration, data integration, system integration, enterprise application integration (EAI), and business-to-business (B2B) integration.

Logic Apps are designed in a web-based designer and can execute logic triggered by Azure services without writing any code. To build enterprise integration solutions with Azure Logic Apps, you can choose from a growing gallery of over 200 connectors, including most popular enterprise apps. You can also build custom connectors and workflow steps if the service you need to interact with isn't covered. You then use a visual designer to link connectors and blocks together, passing data through the workflow to do custom processing - often all without writing any code.

Sometimes, Logic apps and integration accounts need access to secured resources, such as VMs and other systems or services, that are inside a VNET. To set up this access, you can create an integration service environment (ISE) where you can run your logic apps and create your integration accounts (see section § "Options for accessing Azure services and beyond" below).

Resource:

- What is Azure Logic Apps?.

Azure Event Grid

Azure Event Grid allows to easily build applications with event-based architectures. It's a fully managed, intelligent event routing service that uses a publish-subscribe model for uniform event consumption.

Azure Event Grid has built-in support for events coming from Azure services, such as storage blobs, for example, a new blob is added, and resource groups.

In addition, you can use Azue Event Grid to support your own non-Azure-based events in near-real time, using custom topics. You can use filters to route specific events to different endpoints, and ensure your events are reliably delivered.

Azure Event Grid detects these events and makes them available to event handlers and services that subscribe to the events as illustrated hereafter.

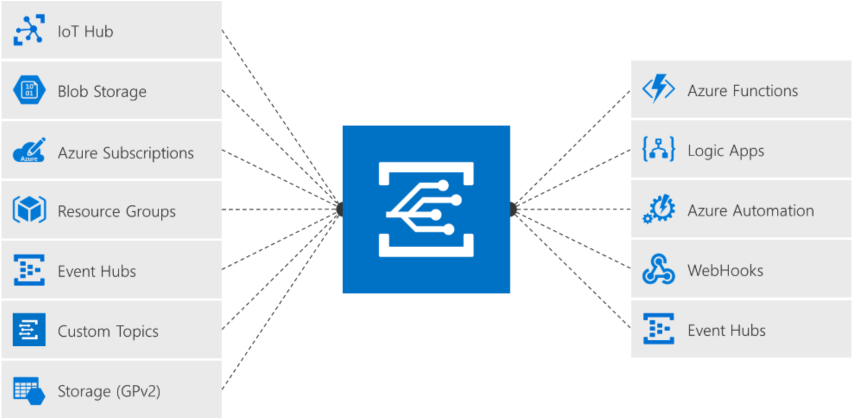


Figure 7 Managing all events in one place

Azure Event Grid is a fully managed, intelligent event routing service that uses a publish-subscribe model for uniform event consumption. Azure Event Grid hooks into almost every service in Azure as well as into custom publishers and subscribers, and automatically pushes messages to subscribers, making it a real-time, reactive event service.

In addition to its default event schema, Azure Event Grid natively supports events in the CloudEvents JSON schema. CloudEvents is an open specification for describing event data. As such, CloudEvents simplifies interoperability by providing a common event schema for publishing, and consuming cloud-based events. This schema allows for uniform tooling, standard ways of routing & handling events, and universal ways of deserializing the outer event schema. With a common schema, you can more easily integrate work across platforms. As of this writing, CloudEvents is being built by several contributors, including Microsoft, through the Cloud Native Computing Foundation (CNCF). It's currently available as version 0.1.

Event handlers can be Azure Functions or Azure Logics Apps, which can then act on the data in the event. Like Azure Functions or Azure Logics Apps, Azure Event Grid scales automatically and doesn't need an instance of it to be deployed. You just configure it and use it, and you pay only when it's used.

Resource:

- What is Azure Event Grid?.

Connecting your applications with data

What can Azure do for your data?

Whatever your data is, wherever your data is, Azure will help you unlock its potential. Support rapid growth and save more time for innovation with a portfolio of secure, enterprisegrade services that support open source frameworks, platforms, and not only SQL (NoSQL) and relational (SQL) database engines.

Beside the ability to instantiate the aboves on top of VMs (see section § "Leveraging compute virtualization capabilities" above), Azure provides a vast range of fully managed services, freeing up valuable time so you can focus on new ways to unlock opportunities rather than spending that time managing the core required plumbing.

Enterprise-grade performance with built-in high availability means you can scale quickly and reach global distribution without worrying about costly downtime.

Let's dive in.

Ingesting/streaming data

Data streaming consists of ingesting data from various sources in real time, with the objective of making it available to various consumers for further processing or storage. When streaming data, one usually refers to very large quantity of data, produced at a highly rate and in some sort of steady and continuous way.

This usually shapes big data architecture. A big data architecture is designed to handle the ingestion/streaming, processing, and analysis of data that is too large or complex for traditional database systems.

> **Note** The threshold at which organizations enter into the big data realm differs, depending on the capabilities of the users and their tools. For some, it can mean hundreds of gigabytes of data, while for others it means hundreds of terabytes. As tools for working with big data sets advance, so does the meaning of big data. More and more, this term relates to the value customers can extract from their data sets through advanced analytics, rather than strictly the size of the data, although in these cases they tend to be quite large.

A typical big data pipeline has four stages, i.e. ingest/stream, process, store, and analysis/reporting. To perform the real-time data ingestion/streaming, you are provided on Azure with a number of technical options, and amongst the technical possibilities, Apache Kafka in the context of this paper.

Apache Kafka is featured in the Azure IoT Reference Architecture Guide. This guide aims to accelerate customers building IoT solutions on Azure by providing a (Cloud-native) proven production ready architecture, with links to Solution Accelerator reference architecture implementations such as Remote Monitoring or Connected Factory.

> **Note** As such, technical content covers topics such as microservices, containers, orchestrators (e.g. Kubernetes), serverless usage, etc. with proven technology implementation recommendations per subsystems and options in terms of technology. Each of these options provide different levels of control, customizability/extensibility, and simplicity. These attributes have different levels of importance for different customers; e.g. some customers need a solution that is highly customized while others might be able to use what is "in the box" and nothing more.
>
> Consequently, primary options customers choose from range from a Microsoft SaaS offering, i.e. Azure IoT Central, abstracting all technical choices, to the use of open source components (e.g. Apache Kafka, Apache Spark, etc.) to bootstrap their system and host it on IaaS VMs/VM scale sets, containers and/or run it on top of fully managed service(s), e.g. Azure

HDInsight. (Azure (IoT) PaaS components (e.g. Azure IoT Hub, Azure Event Hub, Cosmos DB, Redis Cache, etc.) are featured in the middle.).

Apache Kafka

Apache Kafka is an open source distributed streaming platform that can be used to build real-time streaming data pipelines and applications. Microsoft Azure offers a number of options to benefit from Apache Kafka starting from running Apache Kafka on top of Azure IaaS capabilities (see section § "Leveraging compute virtualization capabilities" above).

Beyond this core capabilities, you can leverage fully managed PaaS services in Azure, and more especially Azure HDInsight and Azure Event Hubs as depicted hereafter.

Apache Kafka on Azure HDInsight

Azure HDInsight is a platform within Azure that you can use to run open source data analytics services. You can use it to run specialized clusters of your favorite open source data analytics tools and frameworks, such as Apache Kafka, Apache Hadoop, Apache Spark, Apache Hive (and Live Long And Process (LLAP)), Apache Storm, etc.

Note To see available Hadoop technology stack components on HDInsight, see Components and versions available with HDInsight.

The advantage of running these tools and frameworks in Azure HDInsight is that they're managed, which means you don't have to maintain VMs or patch operating systems. Plus, they can scale and easily connect to one another, other Azure services, and on-premises data sources and services.

Azure HDInsight and some of its benefits:

- It is a fully managed service that provides a simplified configuration process. The result is a configuration that is tested and supported by Microsoft.
- Microsoft provides a 99.9% Service Level Agreement (SLA) on Kafka uptime.
- HDInsight allows you to change the number of worker nodes (which host the Kafka-broker) after cluster creation. Scaling can be performed from the Azure portal, Azure PowerShell, and other Azure management interfaces. For Kafka, you should rebalance partition replicas after scaling operations. Rebalancing partitions allows Kafka to take advantage of the new number of worker nodes.

In addition, Azure HDInsight integrates seamlessly with other Azure services, including Azure Data Factory (see section § "Building data pipelines for data movement, transformation, and analytics" below) and Azure Data Lake Storage (see section § "Using long term storage for your data" below) for building comprehensive analytics pipelines.

Lastly, with the Enterprise Security Package (ESP), you can enable role-based access control (RBAC) by integrating HDInsight clusters with your own Azure AD Domain Services. Azure AD Domain Services (Azure AD DS) is a feature of Azure AD (see section § "Identity and access management" below) that provides managed domain services, such as domain join, group policy, LDAP, and Kerberos / NTLM authentication that is fully compatible with Windows Server Active Directory; Azure AD DS replicates identity information from Azure AD.

Resources:

- What is Azure HDInsight?.
- What is Apache Kafka in Azure HDInsight?.

Azure Event Hubs

Azure Event Hubs is a fully managed and massively scalable distributed streaming platform designed for a plethora of use cases and for ingesting millions of event messages per second. (As an internal illustration, Azure Event Hubs are used by the Xbox One Halo team, as well as powers both Microsoft Teams and Microsoft Office client application telemetry pipelines.) The captured event data can be processed by multiple consumers in parallel.

Azure Event Hubs has been immensely popular with Azure's largest customers and even more so with the release of Event Hubs for Apache Kafka.

Along with the native support of the Advanced Message Queuing Protocol (AMQP) 1.0, Azure Event Hubs also provides a binary compatibility layer that allows existing applications, including Apache Kafka MirrorMaker, using Apache Kafka protocol 1.0 and later to process events using Azure Event Hubs with no application changes.

With this powerful new capability, you can stream events from Kafka applications seamlessly into Azure Event Hubs without having to run Zookeeper or manage Kafka clusters, all while benefitting from a fully managed PaaS service with features like auto-inflate and geo-disaster recover.

Resources:

- What is Azure Event Hubs?.
- Data streaming with Event Hubs using the Kafka protocol.
- Use Azure Event Hubs from Apache Kafka applications.
- Integrate Apache Kafka Connect support on Azure Event Hubs (Preview).
- Connect your Apache Spark application with Kafka-enabled Azure Event Hubs.
- Use Kafka MirrorMaker with Event Hubs for Apache Kafka.

Azure IoT Hub

As part of the above-mentioned Azure IoT platform, Azure IoT Hub provides a cloud-hosted managed solution for bi-directional communication between Internet-connected devices it manages, and a scalable message queue that can handle millions of simultaneously connected devices. You can use Azure IoT Hub to build your solutions with reliable and secure communications between millions of devices and a cloud-hosted solution backend. You can connect virtually any device to Azure IoT Hub.

Note Azure IoT Hub allows devices to use the following protocols for device-side communications: Message Queuing Telemetry Transport (MQTT), MQTT over WebSockets, Advanced Message Queuing Protocol (AMQP) 1.0, AMQP over WebSockets, and HTTPS.

Features of IoT Hub include:

- Multiple options for device-to-cloud and cloud-to-device communication. These options include one-way messaging, file transfer, and request-reply methods.
- Message routing to other Azure services.
- Queryable store for device metadata and synchronized state information.
- Secure communications and access control using per-device security keys or X.509 certificates.
- Monitoring of device connectivity and device identity management events.

In terms of message ingestion, Azure IoT Hub is similar to Azure Event Hubs (see previous section). However, it was specifically designed for managing IoT device connectivity, not just message ingestion.

Resource:

- What is Azure IoT Hub?.

Using (short term) storage for your data

(Short term) storage is needed by all kind of applications to persist data for (interim) processing and when there is no need to keep it for a (very) long time.

Azure offers a comprehensive portfolio of NoSQL and relational database services on Azure that are grounded in choice and flexibility, providing the right tool for maximizing productivity, efficiency, and return on investment for every use case that you encounter.

Whether migrating on-premises (open source) databases at scale, developing new, Cloud-native applications, or even building multi-tenant software-as-a-service (SaaS), Azure databases comprise the range of relational and non-relational, community-based (and proprietary) engines that provide a variety of deployment options and support an array of application types. All the Azure databases are managed by Microsoft (DBaaS), so you can focus more on building great apps and growing your business.

Leveraging NoSQL databases

CosmosDB

Azure Cosmos DB represents a new kind of fully managed database service made for the cloud. Its key features include:

- A 99.99 percent SLA (99.999% for read operations) that includes low latencies (less than 10 milliseconds on reads and less than 15 milliseconds on writes).
- Turnkey global distribution and transparent multi-master replication geo-replication, which replicates data to other geographical regions in real time.
- Tunable data consistency levels so you can enable a truly globally distributed data system. You can choose from a spectrum of data consistency models, including strong consistency, session consistency, and eventual consistency.
- Traffic Manager, which sends users to the service endpoint to which they are closest.
- Limitless global scale, so you pay only for the throughput and storage that you need.
- Automatic indexing of data, which removes the need to maintain or tune the database.

In addition to all these features, Azure Cosmos DB offers different APIs with which you can store and retrieve data. Aside SQL, implements wire protocols for common NoSQL APIs including Not only SQL (NoSQL) APIs:

- MongoDB API compatible with version 3.2 of the MongoDB's wire protocol.

  Features or query operators added in version 3.4 of the wire protocol are currently available as a preview feature. Any MongoDB client driver that understands these protocol versions should be able to natively connect to Cosmos DB.

  Apache Cassandra API, allowing the use of Cassandra client drivers compliant with the Cassandra Query Language (CQL) v4, and Cassandra-based tools.
- Gremlin API based on the Apache TinkerPop graph database standard, and uses the Gremlin query language.
- Etcd API allowing o scale Kubernetes state management on a fully managed cloud native PaaS service.

- Etc.

Different APIs handle data in different ways and you can use the API that fits your needs, for example MongoDB, and Azure Cosmos DB takes care of the rest. This allows you to use your familiar NoSQL client drivers and tools to interact with your Cosmos database as noticed above.

> **Note** Azure Cosmos DB named a leader in the Forrester Wave^{TM}: Big Data NoSQL, Q1 2019 report. Azure Cosmos DB received the highest scores in the Strategy and Roadmap criteria and among the highest scores in the Ability to Execute, Install Base, Market Awareness, Partnerships, Reach, Professional Services and Technical Support criteria.
>
> The Forrester report notably stresses that "Microsoft starts to get strong traction with Azure Cosmos DB […] Customer references like its resilience, low maintenance, cost effectiveness, high scalability, multi-model support, and faster time-to-value. They use Cosmos DB for operational apps, real-time analytics, streaming analytics, and internet-of-things (IoT) analytics."

Resources:

- What is Azure Cosmos DB?.
- Azure Cosmos DB's API for MongoDB.
- Global data distribution with Azure Cosmos DB - overview.

Leveraging relational databases in Azure

Azure SQL Database

Azure SQL Database is a general-purpose relational database, provided as a fully managed service.

With it, you can create a highly available and high-performance data storage layer for your applications and solutions in Azure. SQL Database can be the right choice for a variety of modern applications because it enables you to process both relational data and non-relational structures, such as graphs, JSON, spatial, and XML.

As such, it provides the broadest SQL Server engine compatibility and up to a 212% return on investment.

SQL Database offers several service tiers that are geared toward specific scenarios.

- **General purpose/standard**. This tier offers budget-oriented, balanced, and scalable compute and storage options. This tier is the best option for most business workloads.
- **Business Critical/Premium**. This tier offers the highest resilience to failures using several isolated replicas. With consistently high IO, it includes a built-in availability group for high availability. This is the best option for your critical Online Transactional Processing (OLTP) (normal CRUD operations) business applications with consistently high IO requirements.
- **Hyperscale**. This tier offers very large database (VLDB) support without the headaches. With a built-for-the-cloud architecture of highly scalable storage and a multilayer cache optimized for very large and demanding workloads, it provides low latency and high throughput regardless of the size of data operations. This is the best tier for your very large and demanding workloads with highly scalable storage and read-scale requirements.

Resource:

- What is the Azure SQL Database service?.

Azure databases for MySQL, PostgreSQL, and MariaDB

For your low-latency scenarios, Azure provides community-based Azure Database for MySQL, Azure Database for PostgreSQL, and Azure Database for MariaDB databases as Enterprise-ready, managed databases, which means that you just spin them up and don't have to worry about any of the underlying infrastructure. Just like above Azure Cosmos DB and Azure SQL Database, these databases are universally available, scalable, highly secure, and fully managed (DBaaS).

> **Note** Forrester has named Microsoft as a leader in the Forrester Wave^{TM}: Database-as-a-service, Q2 2019. This decision is based on their evaluation of Azure relational and non-relational databases. According to the Forrester report, customers "like Microsoft's automation, ease of provisioning, high availability, security, and technical support." We believe Microsoft's position as a leader is further underscored by its standing in the recent Forrester Wave^{TM}: Big Data NoSQL, Q1 2019 report.

Each of these databases is suited for slightly different use cases, but in general their functionality overlaps a lot. You would typically use Azure databases for MySQL, PostgreSQL, and MariaDB when you've already been using one of their on-premises community versions and want the advantage of having it run fully managed in the cloud.

Resources:

- What is Azure Database for MySQL?.
- What is Azure Database for PostgreSQL?.
- What is Azure Database for MariaDB?.

Leveraging Azure Storage

Azure Storage is Microsoft's cloud storage solution for modern data storage scenarios. This is is one of the oldest, most reliable, and most performant services in Azure.

As such, Azure Storage offers a massively scalable object store for data objects, a file system service for the cloud, a messaging store for reliable messaging, and a NoSQL store.

Azure Storage includes these data services:

- Azure Blobs. A massively scalable object store for text and binary data.
- Azure Queues. A messaging store for reliable messaging between application components.
- Azure Tables. An inexpensive, extremely fast NoSQL key-value store for schemaless storage of structured data.

Each service is accessed through a storage account.

All data written to Azure Storage is encrypted by the service (See section § "Encryption at rest" below). Azure Storage provides you with fine-grained control over who has access to your data.

Azure Storage is designed to be massively scalable to meet the data storage and performance needs of today's applications.

Furthermore, Microsoft Azure handles hardware maintenance, updates, and critical issues for you.

Finally, data in Azure Storage is accessible from anywhere in the world over HTTP or HTTPS. Microsoft provides client libraries for Azure Storage in a variety of languages, including .NET, Java, Node.js, Python, PHP, Ruby, Go, and others, as well as a mature REST API. Azure Storage supports scripting in Azure PowerShell or Azure CLI. And the Azure portal and Azure Storage Explorer offer easy visual solutions for working with your data.

Resource:

- Introduction to Azure Storage.

Using long term storage for your data

Although conversaly "long" doesn’t refer to anything specific, something longer than 12 months would probably make sense here.

If your applications indeed rely on a big data architecture, there is often a need for an “analytical data store” that serves processed data in a structured format that can be queried using analytical tools. The “Analytical data store” corresponds to the third stage of a typical big data pipeline that comprises four stages, i.e. ingest, process, store, and analysis/reporting as already mentioned.

Analytical data stores that support querying of both hot-path and cold-path data are collectively referred to as the serving layer, or data serving storage.

> **Note** The serving layer deals with processed data from both the hot path and cold path. In the well-known lambda architecture, the serving layer is subdivided into a speed serving layer, which stores data that has been processed incrementally, and a batch serving layer, which contains the batch-processed output. The serving layer requires strong support for random reads with low latency. Data storage for the speed layer should also support random writes, because batch loading data into this store would introduce undesired delays. On the other hand, data storage for the batch layer does not need to support random writes, but batch writes instead.

There is no single best data management choice for all data storage tasks. Different data management solutions are optimized for different tasks. Most real-world cloud apps and big data processes have a variety of data storage requirements and often use a combination of data storage solutions. Unsurprisingly, you are provided on Azure with a number of technical options to prepare data for analysis and then serve the processed data in a structured format that can be queried using analytical tools. However, most of them if not all (will) rely on Azure Data Lake Storage Gen 2.

Azure Data Lake Storage

Azure Data Lake Storage is a set of capabilities dedicated to big data analytics, built on Azure Blob storage (see section § “Using (short term) storage for your data” above). It combines the power of a Hadoop compatible file system with integrated hierarchical namespace with the massive scale and economy of Azure Blob Storage to help speed your transition from proof of concept to production.

Azure Data Lake Storage Gen2 is the result of converging the capabilities of our two existing storage services, Azure Blob storage and Azure Data Lake Storage Gen1. Features from Azure Data Lake Storage Gen1, such as file system semantics, directory, and file level security and scale are combined with low-cost, tiered storage, high availability/disaster recovery capabilities from Azure Blob storage.

> **Important note** Blob Storage APIs are disabled to prevent feature operability issues that could arise because Blob Storage APIs aren't yet interoperable with Azure Data Lake Gen2 API. With the public preview of multi-protocol access on Data Lake Storage, blob APIs and Data Lake Storage Gen2 APIs can operate on the same data. For more information, see article Multi-protocol access on Azure Data Lake Storage (preview).

> **Note** With multi-protocol access on Data Lake Storage, you can work with all of your data by using the entire ecosystem of tools, applications, and services. This includes Azure services such as Azure HDInsight, Azure Event Hubs, Azure IoT Hub, Azure Data Factory, Azure Stream Analytics, Power BI, and many others. For a complete list, see article Integrate Azure Data Lake Storage with Azure services.

Data Lake Storage Gen2 makes Azure Storage the foundation for building enterprise data lakes on Azure. Designed from the start to service multiple petabytes of information while sustaining hundreds of gigabits of throughput, Data Lake Storage Gen2 allows you to easily manage massive amounts of data.

In the past, cloud-based analytics had to compromise in areas of performance, management, and security. Data Lake Storage Gen2 addresses each of these aspects in the following ways:

- Performance is optimized because you do not need to copy or transform data as a prerequisite for analysis. The hierarchical namespace greatly improves the performance of directory management operations, which improves overall job performance.
- Management is easier because you can organize and manipulate files through directories and subdirectories.
- Security is enforceable because you can define POSIX permissions on directories or individual files.
- Cost effectiveness is made possible as Data Lake Storage Gen2 is built on top of the low-cost Azure Blob storage. The additional features further lower the total cost of ownership for running big data analytics on Azure.

Key features of Data Lake Storage Gen2 include:

- **Hadoop compatible access**. Data Lake Storage Gen2 allows you to manage and access data just as you would with a Hadoop Distributed File System (HDFS). The ABFS driver is available within all Apache Hadoop environments, including aforementioned Azure HDInsight, Azure Databricks services, as well as Azure SQL Data Warehouse to access data stored in Data Lake Storage Gen2.
- **A superset of POSIX permissions**. The security model for Data Lake Gen2 supports ACL and POSIX permissions along with some extra granularity specific to Data Lake Storage Gen2. Settings may be configured through Azure Storage Explorer or through frameworks like Apache Hive and Apache Spark.
- **Cost effective**. Data Lake Storage Gen2 offers low-cost storage capacity and transactions. As data transitions through its complete lifecycle, billing rates change keeping costs to a minimum via built-in features such as Azure Blob storage lifecycle.
- **Optimized driver**. The ABFS driver is optimized specifically for big data analytics. The corresponding REST APIs are surfaced through the endpoint dfs.core.windows.net.

Resources:

- Choosing an analytical data store in Azure.
- Introduction to Azure Data Lake Storage Gen2.
- Using Azure Data Lake Storage Gen2 for big data requirements.
- Integrate Azure Data Lake Storage with Azure services.

Azure SQL Data Warehouse

When you need a traditional data warehousing solution that is completely managed, scalable in size, and performant and secure, Azure SQL Data Warehouse can provide the solution. Store data

Azure SQL Data Warehouse is a cloud-based Enterprise Data Warehouse (EDW) that combines SQL relational databases with massively parallel processing (MPP) to quickly run complex queries across petabytes of data.

You can use SQL Data Warehouse as a key component of a big data solution.

Resource:

- What is SQL Data Warehouse?.

Building a metadata catalog of all your data

The metadata catalog contains all the information (a.k.a. metadata: data about data). It is very important to have a central place where the data structure is documented; this information is used when transforming or querying data. A metadata catalog is particularly important when dealing with unstructured data.

Azure Data Catalog is an enterprise-wide metadata catalog that makes data asset discovery straightforward. It's a fully managed cloud service that lets users register, enrich, discover, understand, and consume data sources.

With Azure Data Catalog, any user - from analyst to data scientist to data developer – can discover the data sources they need and understand the data sources they find. Data Catalog includes a crowdsourcing model of metadata and annotations.

It is a single, central place for all of an organization's users to contribute their knowledge and build a community and culture of data.

Resource:

- What is Azure Data Catalog?.

Archiving your data

Archiving addresses the business need of keeping and protecting data for a very long time; we're talking about decades. This type of storage needs to be cheap and must ensure that data cannot be altered. This latter is often backed by some data lifecycle management capabilities, see section § "Managing the lifecycle of your data" below.

Azure Archive Storage provides a storage facility for data that is rarely accessed. It allows you to archive legacy data at low cost to what it would traditionally have cost to create and maintain archives.

Archive storage is available as a tier of the Azure Blob storage (see section § "Leveraging Azure Storage" above) that offers different access tiers, which allow you to store your data in the most cost-effective manner. Available access tiers include:

- **Hot**. Optimized for storing data that is accessed frequently.
- **Cool**. Optimized for storing data that is infrequently accessed and stored for at least 30 days.
- **Archive**. Optimized for storing data that is rarely accessed and stored for at least 180 days with flexible latency requirements (on the order of hours).

With the latter, data is stored offline and offers the lowest storage costs. However, it also has the highest access cost, hence it is suited for archival data that is rarely accessed.

Archive storage is intended for data that can tolerate several hours of retrieval latency and will remain archived for at least 180 days.

While a blob is in archive storage, the blob data is offline and can't be read, copied, overwritten, or modified. You can't take snapshots of a blob in archive storage. However, the blob metadata remains online and available, allowing you to list the blob and its properties. For blobs in archive, the only valid operations are GetBlobProperties, GetBlobMetadata, List Blobs, SetBlobTier, and DeleteBlob.

Example usage scenarios for the archive access tier include:

- Long-term backup, secondary backup, and archival datasets.

- Original (raw) data that must be preserved, even after it has been processed into final usable form.
- Compliance and archival data that needs to be stored for a long time and is hardly ever accessed.

All archive operations are consistent with the other tiers so you can seamlessly move your blob data among tiers programmatically or using lifecycle management policies. Archive is supported by a broad and diverse set of storage partners.

As such, Azure Archive Storage allows (small and large) customers from all industries to significantly reduce their storage bill, improve data durability, and meet legal compliance.

Note Azure Archive Storage provides an extremely cost-effective alternative to on-premises storage for cold data as highlighted in the Forrester Total Economic Impact (TEI) study, a study commissioned By Microsoft to evaluate the value customers achieved by moving both on-premises and existing data in the cloud to Archive Storage. Customers can significantly reduce operational and hardware expenses to realize an ROI of up to 112 percent over three years by moving their data to the Archive tier.

Resources:

- Azure Blob storage: hot, cool, and archive access tiers.
- Rehydrate blob data from the archive tier.

Managing the lifecycle of your data

Not all data are meant to stay in the same state or place forever. Data sets have unique lifecycles. Early in the lifecycle, people access some data often. But the need for access drops drastically as the data ages. Some data stays idle in the cloud and is rarely accessed once stored. Some data expires days or months after creation, while other data sets are actively read and modified throughout their lifetimes.

Data lifecycle management is where you configure what happens to data once particular conditions have been met. These conditions will most likely relate to the age of the data, but not necessary only that. What you actually do with the data depends on the lifecycle policy, but could be anything like deleting, moving to different storage tier, etc.

Azure Blob storage (see section § "Using (short term) storage for your data" above) lifecycle management offers a rich, rule-based policy for General Purpose v2 (GPv2) and Blob storage accounts, and Premium Block Blob storage accounts. In the Azure portal, you can upgrade an existing General Purpose (GPv1) account to a GPv2 account.

Note This features set is available to accounts that have a hierarchical namespace only if you enroll in the public preview of multi-protocol access on Data Lake Storage (see section § "Azure Data Lake Storage" above). To review limitations, see article Known issues with Azure Data Lake Storage Gen2.

You can use the policy to transition your data to the appropriate access tiers or expire at the end of the data's lifecycle.

The lifecycle management policy lets you:

- Transition blobs to a cooler storage tier (hot to cool, hot to archive, or cool to archive) to optimize for performance and cost.
- Delete blobs at the end of their lifecycles.
- Define rules to be run once per day at the storage account level.
- Apply rules to containers or a subset of blobs (using prefixes as filters).

As an illustration, one can consider a scenario where data gets frequent access during the early stages of the lifecycle, but only occasionally after two weeks. Beyond the first month, the data set is rarely accessed. In this scenario, hot storage is best during the early stages. Cool storage is most appropriate for occasional access. Archive storage (see previous section) is the best tier option after the data ages over a month. By adjusting storage tiers in respect to the age of data, you can design the least expensive storage options for your needs. To achieve this transition, lifecycle management policy rules are available to move aging data to cooler tiers.

Resources:

- Manage the Azure Blob storage lifecycle.
- Rehydrate blob data from the archive tier.

Building data pipelines for data movement, transformation, and analytics

In the world of (big) data, raw, unorganized data is often stored in relational, non-relational, and other storage systems as per above sections. On its own, raw data doesn't have the proper context or meaning to provide meaningful insights to analysts, data scientists, or business decision makers. To make that happens, data needs to be moved in-between locations, from one place to another, internally but also possibly externally, all of that with possibly transformation/computation happening at the same time. This capability is often referred to as Extract-Transform-Load (ETL), or even Extract-Load-Transform (ELT), depending on where the transformation happens.

Moving and transforming data is not a trivial task, but Azure Data Factory (Azure Data Integration Service) can help you to do just that. Azure Data Factory is a managed cloud service that's built for these complex hybrid ETL, ELT, and data integration projects via the definition of suitable pipelines, i.e. data-driven workflows.

Within Azure Data Factory, you can create one or more comprehensive pipelines that perform your complete ETL/ELT processes. A pipeline is a logical grouping of activities that performs a unit of work. Together, the activities in a pipeline perform a task. For example, a pipeline can contain a group of activities that ingests data from an Azure Blob Storage, and then runs a Hive query on an Azure HDInsight cluster to partition the data.

The benefit of this is that the pipeline allows you to manage the activities as a set in lieu of managing each one individually. The activities in a pipeline can be chained together to operate sequentially, or they can operate independently in parallel. Activities represent a processing step in a pipeline.

As such, pipelines in Azure Data Factory typically perform the following four steps:

1. **Connect and collect**. Enterprises have data of various types that are located in disparate sources on-premises, in the cloud, structured, unstructured, and semi-structured, all arriving at different intervals and speeds.

   With Azure Data Factory, you can use the Copy activity in a data pipeline to move data from both on-premises and cloud source data stores to a centralization data store in the cloud for further analysis.

   For example, you can collect data in Azure Data Lake Storage (see section § "Azure Data Lake Storage" above) and transform the data later by using an Azure Data Lake Analytics compute service - Azure Data Lake Analytics is an on-demand analytics job service that, in lieu of deploying, configuring, and tuning hardware, allows you writing queries to transform your data and extract valuable insights -. You can also collect data in Azure Blob storage and transform it later by using an Azure HDInsight Hadoop cluster.

   After you copy the data, you can use other activities to further transform and analyze it.

2. **Transform and enrich**. After data is present in a centralized data store in the cloud, you want to reliably produce transformed data on a maintainable and controlled schedule to feed production environments with trusted data.

   For that purpose, you can process or transform the collected data thanks to transformation activities running on a variety of compute services such as Azure HDInsight (Hadoop, Spark), Azure Databricks (see section § "Leveraging Azure data analytics solutions" below), Azure Data Lake Analytics, and Azure Machine Learning (see section § "Building intelligence with your data" below). an activity defines the action to be performed. For example, a Spark activity can be used with an on-demand Azure HDInsight linked service to transform some data.

   You can also use the Copy activity to publish transformation and analysis results for business intelligence (BI) and application consumption.

3. **Publish**. After the raw data has been refined into a business-ready consumable form, load the data into:

   - Azure Cosmos DB (see section § "Using (short term) storage for your data" above).
   - Azure SQL Database (see section § "Using (short term) storage for your data" above).
   - Azure SQL Data Warehouse, a fully managed cloud data warehouse for enterprises of any size that combines lightning-fast query performance with industry-leading data security.
   - Or whichever analytics engine your business users can point to from their business intelligence tools.

4. **Monitor**. After you have successfully built and deployed your data integration pipeline, providing business value from refined data, monitor the scheduled activities and pipelines for success and failure rates. Azure Data Factory has built-in support for pipeline monitoring via Azure Monitor (see section § "Managing your logs" below) logs, and health panels on the Azure portal.

Resource:

- What is Azure Data Factory?.

Exporting your data

Data Export typically exposes the output of a query in such a way that it can be consumed by external parties in the form of a file.

The data movement in Azure Data Factory allows to copy data across data stores in public network and data stores in private network (on-premises or VNET). It provides support for built-in connectors, format conversion, column mapping, and performant and scalable data transfer.

The integration runtime (IR) is the technical compute infrastructure that Azure Data Factory uses underneath to provide data-integration capabilities across different network environments.

Azure Data Factory offers different types of integration runtime, and you should choose the type that best serve the data integration capabilities and network environment needs you are looking for.

Beside the Azure integration runtime that provides a fully managed compute to natively perform data movement and dispatch data transformation activities to compute services like Azure HDInsight (see above section), a self-hosted integration runtime can run copy activities between a cloud data store and a data store in a private network, and it can dispatch transform activities against compute resources in an on-premises network or an Azure VNET. The installation of a self-hosted integration runtime needs on an on-premises machine or a virtual machine (VM) inside a private network.

Beyond this capability, one should note that many organizations need to be accountable for the data that they have shared. In addition to accountability, many organizations would like to be able to control, manage, and monitor all of their data sharing in a simple way. In today world, where data is expected to continue to grow at an exponential pace, organizations need a simple way to share big data. Customers demand the most up-to-date data to ensure that they are able to derive timely insights.

Azure Data Share enables organizations to simply and securely share data with multiple customers and partners. In just a few clicks, you can provision a new data share account, add datasets, and invite your customers and partners to your data share. Data providers are always in control of the data that they have shared. Azure Data Share makes it simple to manage and monitor what data was shared, when and by whom. Azure Data Share is currently in preview.

Resources:

- Integration runtime in Azure Data Factory.
- What is Azure Data Share Preview?.

Leveraging Azure data analytics solutions

Almost as important as ingesting/streaming, storing and moving data is analyzing it to get insights. Azure provides many services for data analytics scenarios, enabling you to get valuable and actionable insights from your data - no matter how large or small or complex it is.

Processing large data set

Processing large sets of data, either in real-time or in batch, is needed to start turning raw data into more meaningful or aggregated information.

You are provided on Azure with a number of technical options to best accommodate your own specific requirements and needs.

Apache Storm on Azure HDInsight

Apache Storm is an open source framework for stream processing that uses a topology of spouts and bolts to consume, process, and output the results from real-time streaming data sources. You can use Apache Storm to process streams of data in real time with Apache Hadoop. Storm solutions can also provide guaranteed processing of data, with the ability to replay data that was not successfully processed the first time.

Customer can provision Apache Storm in an Azure HDInsight cluster, and implement a topology in Java or C#. As already introduced, Azure HDInsight is a managed, full-spectrum, open source analytics service in the public cloud.

Azure HDInsight allows to easily run popular open source frameworks, including Apache Storm, Apache Spark, etc., to effortlessly process massive amounts of data, and to ultimately get all the benefits of the broad open source ecosystem with the global scale of Azure.

Apache Storm on Azure HDInsight comes with full enterprise-level continuous support. Apache Storm on Azure HDInsight also provides an SLA of 99.9 percent. In other words, Microsoft guarantees that an Apache Storm cluster has external connectivity at least 99.9 percent of the time.

Resources:

- What is Azure HDInsight?.
- What is Apache Storm on Azure HDInsight?.
- Process events from Event Hubs with Apache Storm on HDInsight.
- Create and monitor an Apache Storm topology in Azure HDInsight.

Apache Spark Streaming on Azure HDInsight

Apache Spark is an open source distributed platform for general data processing. Apache Spark provides the Apache Spark Streaming API, in which you can write code in any supported Spark language, including Java, Scala, and Python. Apache Spark 2.0 introduced the Spark Structured Streaming API, which provides a simpler and more consistent programming model.

Interestingly enough, Spark 2.0 is available in an Azure HDInsight cluster -Azure HDInsight is a managed, full-spectrum, open source analytics service in the public cloud -.

Apache Spark Streaming provides data stream processing on HDInsight Spark clusters, with a guarantee that any input event is processed exactly once, even if a node failure occurs.

A Spark Stream is a long-running job that receives input data from a wide variety of sources.

While Apache Spark already has connectors to ingest data from many sources like Apache Kafka, Apache Flume, ZeroMQ, TCP sockets, etc. Apache Spark in Azure HDInsight adds first-class support for ingesting data from Azure Event Hubs (see section § "Ingesting/streaming data" above). Azure Event Hubs is the most widely used queuing service on Azure. Having an out-of-the-box support for Azure Event Hubs may make Azure Spark clusters in Azure HDInsight for some you an ideal platform for building real-time analytics pipeline.

Resources:

- What is Azure HDInsight?.
- What is Apache Spark in Azure HDInsight?.

Azure Databricks with Spark Streaming

Azure Databricks is an Apache Spark-based analytics platform optimized for the Microsoft Azure cloud services platform. Designed with the founders of Apache Spark, Databricks is integrated with Azure to provide one-click setup, streamlined workflows, and an interactive workspace that enables collaboration between data scientists, data engineers, and business analysts.

Azure Databricks is a fast, easy, and collaborative Apache Spark-based analytics service, and is fully integrated with Azure AD, which gives you the ability to implement granular security (see section § "Identity and access management" below).

For a big data pipeline, the data (raw or structured) is ingested into Azure through Azure Data Factory (see section § "Building data pipelines for data movement, transformation, and analytics" above) in batches, or streamed near real-time using Apache Kafka (on Azure HDInsight), Azure Event Hub, or Azure IoT Hub (see section § "Ingesting/streaming data" above).

Azure Databricks comprises the complete open source Apache Spark cluster technologies and capabilities. Within Databricks, you can run optimized versions of Apache Spark to do advanced data analytics, and notably Spark Streaming for real-time data processing and analysis for analytical and interactive applications. This integrates with Integrates with HDFS, Apache Flume, and Apache Kafka.

Apache Spark in Azure Databricks also includes the following components:

- **Spark SQL and DataFrames**. Spark SQL is the Spark module for working with structured data. A DataFrame is a distributed collection of data organized into named columns. It is conceptually equivalent to a table in a relational database or a data frame in R/Python.

- **MLib**. Machine Learning library consisting of common learning algorithms and utilities, including classification, regression, clustering, collaborative filtering, dimensionality reduction, as well as underlying optimization primitives.
- **GraphX**. Graphs and graph computation for a broad scope of use cases from cognitive analytics to data exploration.
- **Spark Core API**. Includes support for R, SQL, Python, Scala, and Java.

Resources:

- What is Azure Databricks?.
- Work with Spark Clusters.

Azure Stream Analytics

Azure Stream Analytics is a real-time analytics and complex event-processing engine that is designed to analyze and process high volumes of fast streaming data from multiple sources simultaneously.

Azure Stream Analytics is a fully managed serverless (PaaS) offering on Azure. You don't have to provision any hardware or manage clusters to run your jobs. Azure Stream Analytics fully manages your job by setting up complex compute clusters in the cloud and taking care of the performance tuning necessary to run the job.

Integration with Azure Event Hubs and Azure IoT Hub (see section § "Ingesting/streaming data" above) allows your job to ingest millions of events per second coming from a number of sources, to include connected devices, clickstreams, and log files.

An Azure Stream Analytics job consists of an input, query, and an output. Stream Analytics can connect to Azure Event Hubs and Azure IoT Hub for streaming data ingestion, as well as Azure Blob storage (see section § "Using (short term) storage for your data" above) to ingest historical data.

Job output can be routed to many storage systems such as Azure Blob storage, Azure CosmosDB (see section § "Using (short term) storage for your data" above), and Azure Data Lake Store (see section § "Using long term storage for your data" above).

The query, which is based on SQL query language, can be used to easily filter, sort, aggregate, and join streaming data over a period of time. You can also extend this SQL language with JavaScript user defined functions (UDFs). You can easily adjust the event ordering options and duration of time windows when preforming aggregation operations through simple language constructs and/or configurations.

Resource:

- What is Azure Stream Analytics?.

Querying your data

The ability to query any data stored is used to explore data and possibly prepare work for other activities like data visualization (see next section) or data export (see section § "Resource:

What is Azure Data Factory?.

Exporting your data" above).

As already introduced, Azure Databricks is a fast, easy, and collaborative Apache Spark-based analytics service for big data pipeline (raw or structured data). Through a collaborative and integrated environment, Azure Databricks streamlines the process of exploring data, prototyping, and running data-driven applications in Spark.

Azure Databricks has the ability to query any data easily with one of those components, i.e. the Spark Core API (that includes support for R, SQL, Python, Scala, and Java).

Following capabilities are notably provided:

- Determine how to use data with easy data exploration.
- Document your progress in notebooks in R, Python, Scala, or SQL.
- Visualize data in a few clicks, and use familiar tools like Matplotlib, ggplot, or d3.
- Use interactive dashboards to create dynamic reports.
- Use Spark and interact with the data simultaneously.

With Azure Databricks, you can perform Spark-based data analytics on data that comes from many places, including Azure Storage and Azure Data Lake Store (see section § "Using (short term) storage for your data" above). Databricks also works with data from Azure SQL Database, and Azure Cosmos DB (see section § "Using (short term) storage for your data" above). Additionally, you can plug Databricks into Power BI to create and show powerful dashboards (see next section).

In addition, Azure Databricks provides interactive notebooks and integrated workflows and workspaces you can use to collaborate with the entire data team, including data scientists, data engineers, and business analysts - all of whom have access to specialized tools for their specific needs.

Resource:

- What is Azure Databricks?.

Visualizing your data

This abaibility to data in a consumable way by the business includes charts, tables, gauges and interactive reporting that can help with decision making.

You can use a number of interactive data visualization BI tools on Azure. Amongst the various possibilities, one should outline Power BI.

Power BI is a collection of business analytics software services, applications, and connectors that work together to turn your unrelated sources of data into coherent, visually immersive, and interactive insights. Your data may be an Excel spreadsheet, or a collection of cloud-based and on-premises hybrid data warehouses. The latter requires the on-premises data gateway. The on-premises data gateway can be deployed centrally to enable cloud to on-premises data connections.

Note The on-premises data gateway acts as a bridge to provide quick and secure data transfer between on-premises data (data that isn't in the cloud) and several cloud services in Azure, Power BI being one of them. By using a gateway, organizations can keep databases and other data sources on their on-premises networks, yet securely use that on-premises data in cloud services.)

Power BI lets you easily connect to your (cloud-based and on-premises) data sources, visualize and discover what's important, and share that with anyone or everyone you want. For example, Power BI offers, Azure Databricks integrates with Power BI to allows you to discover and share your impactful insights quickly and easily. (You can also use Tableau Software via JDBC/ODBC cluster endpoints.)

The Microsoft Power BI service (app.powerbi.com), sometimes referred to as Power BI online, is the SaaS part of Power BI. In the Power BI service, dashboards help you keep a finger on the pulse of your business. Dashboards display tiles, which you can select to open reports for exploring further. Dashboards and reports connect to datasets that bring all of the relevant data together in one place.

Resources:

- [What is Power BI?](#).
- [What is the Power BI service?](#).
- [What is an on-premises data gateway?](#).

Building intelligence with your data

Machine Learning is often thought to mean the same thing as Artificial Intelligence (AI), but they aren't actually the same. AI involves machines that can perform tasks characteristic of human intelligence. AI can also be implemented by using machine learning, in addition to other techniques.

Machine Learning itself is a field of computer science that gives computers the ability to learn without being explicitly programmed. Machine Learning can be achieved by using one or multiple algorithm technologies, like neural networks, Deep Learning, and Bayesian networks.

In order to create data analytics algorithms with open-source tools like Python and the Azure CLI, you can use Azure Machine Learning service. You can create whatever algorithm, a.k.a. model, you want, providing flexibility for a variety of scenarios, like predictive analytics, data recommendations, and data classification. With Azure Machine Learning service, you create custom Machine Learning algorithms from scratch. Azure Machine Learning service fully supports open-source technologies like Google TensorFlow, PyTorch, and scikit-learn.

Azure Machine Learning is a complete service that offers start-to-finish capabilities. You can create your model, prepare your data, train the model on it, test and deploy the model at scale in production, and track and manage it when it's running. You can indeed use Azure Machine Learning to manage the complete lifecycle of your models. Azure Machine Learning uses a Machine Learning Operations (MLOps) approach, which improves the quality and consistency of your machine learning solutions.

Azure Machine Learning provides the following MLOps capabilities:

- **Deploy ML projects from anywhere.** Converting your model to the Open Neural Network Exchange (ONNX) open format may improve performance. On average, converting to ONNX can yield a 2x performance increase.

Note ONNX comes with the ONNX Runtime, an optimized ML inference engine for ONNX models. For more information on ONNX with Azure Machine Learning, see article Create and accelerate ML models.

ONNX

Note At the time of this writing, ONNX is joining the LF AI Foundation, an umbrella foundation of the Linux Foundation supporting open source innovation in Artificial Intelligence (AI), Machine Learning, and Deep Learning

- **Monitor ML applications for operational and ML related issues**, including comparing model inputs between training and inference, exploring model-specific metrics and providing monitoring and alerts on your ML infrastructure in production.
- **Capture the data required for establishing an end to end audit trail of the ML lifecycle**, including who is publishing models, why changes are being made, and when models were deployed or used in production.
- **Automate the end to end ML lifecycle with Azure Machine Learning and Azure DevOps** to frequently update models, test new models, and continuously roll out new ML models alongside your other applications and services.

Azure Machine Learning works with many Azure services that can help you create, train, and run algorithms. You can, for instance, create your algorithm in Jupyter Notebook or Azure Notebooks in for online Jupyter Notebooks, train it using Azure Databricks, and deploy it on a Kubernetes container cluster in Azure Kubernetes Service. (see section § "Leveraging compute virtualization capabilities" above).

Resources:

- What is Azure Machine Learning?.
- What can I do with Azure Machine Learning?.
- MLOps: Manage, deploy, and monitor models with Azure Machine Learning..
- What is Azure Databricks?.
- Machine Learning Guide for machine learning capabilities in Azure Databricks.

Securing your applications

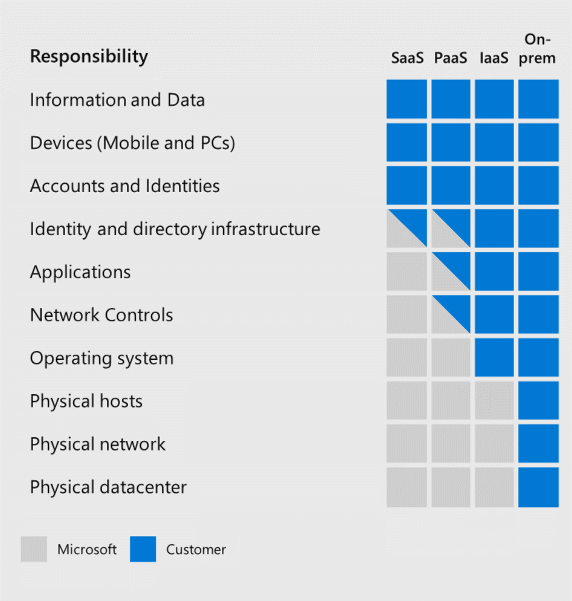
Microsoft takes a defense-in-depth approach to security in Azure. Consequently, security is integrated into every aspect of Microsoft Azure.

Before diving into the various considerations that pertain to this suject, let's start discussing the share responsibilities that rule the environment.

Understanding the shared responsibilities' model for your applications

Because cloud services split the operational responsibilities of workloads between the CSP and the customer tenant, it is critical to have a common understanding of the shared responsibility model and what security tasks will be handled by the CSP and which ones will be handled by your organization. The workload responsibilities vary depending on whether the workload is hosted on Software-as-a-Service (SaaS), Platform-as-a-Service (PaaS), Infrastructure-as-a-Service (IaaS), or in an on-premises datacenter.

In Figure 8 below, the left-most column shows ten responsibilities (defined in the sections that follow) that organizations should consider, all of which contribute to the security and privacy of a computing environment.



Responsibility	SaaS	PaaS	IaaS	On-prem
Information and Data	🟦	🟦	🟦	🟦
Devices (Mobile and PCs)	🟦	🟦	🟦	🟦
Accounts and Identities	🟦	🟦	🟦	🟦
Identity and directory infrastructure	◩	◩	🟦	🟦
Applications	⬜	◩	🟦	🟦
Network Controls	⬜	◩	🟦	🟦
Operating system	⬜	⬜	🟦	🟦
Physical hosts	⬜	⬜	⬜	🟦
Physical network	⬜	⬜	⬜	🟦
Physical datacenter	⬜	⬜	⬜	🟦

⬜ Microsoft 🟦 Customer

Figure 8 Shared responsibilities for operations and security

After an Information Security Management System (ISMS) foundation is set and best practices are adopted, there are additional areas to evaluate and understand to determine an enterprise organization's risk posture and keys for mitigating its risks. To do this, organizations need to understand which areas are the cloud service provider's responsibility and which are the organization's responsibility.

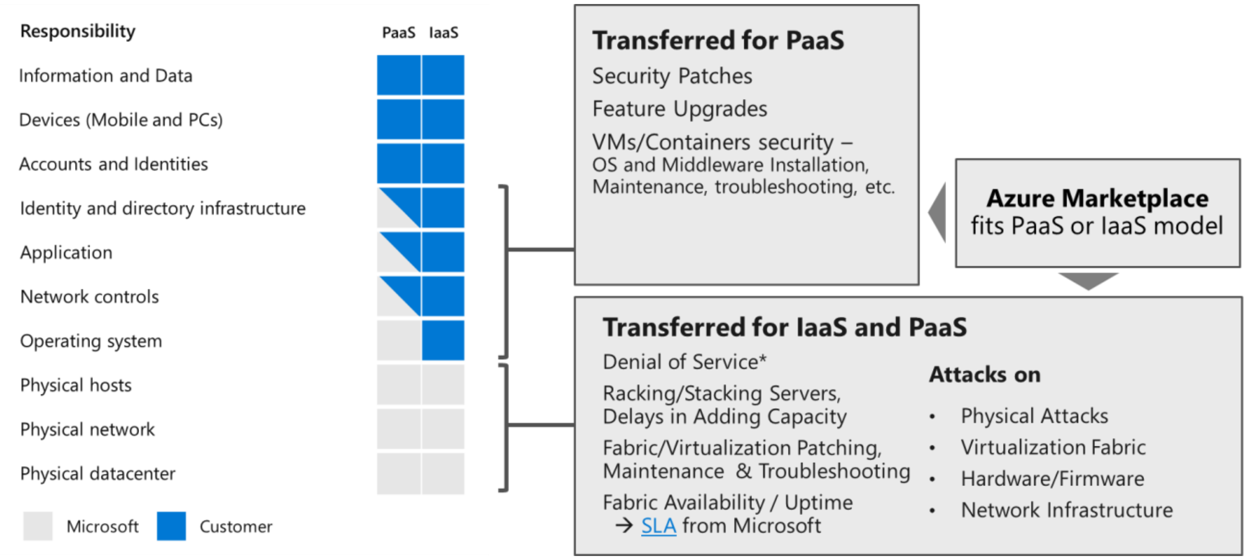
Above Figure 8 makes it clear that responsibilities are driven by the cloud service model: SaaS, PaaS, IaaS, or on-premises.

Except for *Information and Data*, *Devices (Mobile and PCs)*, and *Accounts and Identities* that pertain to the organizations' complete responsibilities, there are operational and security responsibilities that:

- Are shared between the customer and the CSP and require to be managed and administered together by both the customer and the CSP, including auditing of their domains.

  For example, consider *Identity and directory infrastructure* for Identity & Access Management (IAM), Microsoft Azure uses Azure Active Directory (Azure AD) to manage users and to provide (password less and multi-factor) authentication, (privileged) identity management, and role-based access control (RBAC), and more, see section § "Identity and access managementIdentity and access management" below. The configuration of certain services such as password less and multi-factor authentication is up to the customer, but ensuring effective functionality is the responsibility of Microsoft Azure.

- Transfer to the CSP for IaaS and PaaS workloads as illustrated hereafter. Azure Marketplace offerings fit into one of these two models depending on how they are architected.



Responsibility	PaaS	IaaS
Information and Data	🟦	🟦
Devices (Mobile and PCs)	🟦	🟦
Accounts and Identities	🟦	🟦
Identity and directory infrastructure	◩	🟦
Application	◩	🟦
Network controls	◩	🟦
Operating system	⬜	🟦
Physical hosts	⬜	⬜
Physical network	⬜	⬜
Physical datacenter	⬜	⬜

⬜ Microsoft 🟦 Customer

Transferred for PaaS

Security Patches

Feature Upgrades

VMs/Containers security – OS and Middleware Installation, Maintenance, troubleshooting, etc.

Azure Marketplace fits PaaS or IaaS model

Transferred for IaaS and PaaS

Denial of Service*

Racking/Stacking Servers, Delays in Adding Capacity

Fabric/Virtualization Patching, Maintenance & Troubleshooting

Fabric Availability / Uptime → SLA from Microsoft

Attacks on

- Physical Attacks
- Virtualization Fabric
- Hardware/Firmware
- Network Infrastructure

Figure 9 Security responsibilities transfer to the cloud

Note See the Shared responsibility in cloud computing white paper to learn more about the responsibility for each cloud based solution whether it's an IaaS, a PaaS, or a SaaS solution.

One should mention that the aforementioned ISO/IEC 27017:2015 standard is unique in providing guidance for both CSPs and cloud service customers. It also provides cloud service customers with practical information on what they should expect from CSPs. Customers can benefit directly from ISO/IEC 27017:2015 by ensuring they understand the shared responsibilities in the cloud.

How can Azure help secure your applications?

Physical security

Physical security ensures Azure infrastructures are physically secured wherever they are located. This capability ensures that datacenters or any other facilities are under enough physical protection to secure them against intruders.

As per above section, the physical security falls into the responsibility of the CSP. It procures it and manages it.

When a customer chooses Microsoft Azure, Microsoft Cloud Infrastructure and Operations (MCIO) manages the physical security of datacenters for Microsoft Azure. It takes responsibility for designing, building datacenters, provisioning and maintaining hardware and securing both the physical datacenters and the services in a way that strictly controls physical access to the areas where customer application instances and customer data reside.

MCIO has years of experience in delivering the world's largest online services with 24 x 7 continuity.

Microsoft has hundreds of Azure datacenters in 54 regions (as of 2019), and each of these has extensive multilayered protections to ensure unauthorized users cannot gain physical access to your customer data. Layered physical security measures at Microsoft datacenters include access approval:

- At the facility's perimeter
- At the building's perimeter.
- Inside the building.
- On the datacenter floor.

Physical security reviews of the facilities are conducted periodically to ensure the datacenters properly address Azure security requirements.

These geographically dispersed datacenters comply with a broad set of international as well as regional and industry-specific compliance standards, and notably ISO/IEC 27001:2013 and NIST SP 800-53 Rev 4, for security as already outlined.

Resources:

- Azure facilities, premises, and physical security.
- Microsoft Datacenter Security video.

Security design and operations

As already outlined, Microsoft makes Azure security a priority at every step, including code development that follows the Microsoft Security Development Lifecycle (SDL), a company-wide, mandatory process based on a rigorous set of security controls that govern operations, as well as robust incident response strategies. The Microsoft Operational Security Assurance (OSA) framework makes Microsoft business cloud services more resilient to attack by decreasing the amount of time needed to prevent, detect, and respond to real and potential internet-based security threats. See section § "Microsoft Security standards and related practices" in the Appendix.

Infrastructure protection

Infrastructure protection ensures customers' assets exposed externally (e.g. Internet or through any partner network connection) are secured. This encompasses any security control that secures network flows between customers' assets and external network as well as any security control that identifies attacks on the traversed networks. Typically this embrasses the following security controls: network firewalls, intrusion detection systems (IDS)/intrusion prevention systems (IPS), etc.

When a customer chooses Microsoft Azure, Microsoft Cloud Infrastructure and Operations (MCIO) takes responsibility and delivers the Azure infrastructure and production network for Microsoft Azure that support the Azure infrastructure where customer application instances and customer data reside, as per section § "Understanding the shared responsibilities' model for your applications".

Secure hardened infrastructure

The Microsoft Azure production network structured such that publicly accessible system components are segregated from internal resources. Physical and logical boundaries exist between web servers providing access to the public-facing Microsoft Azure management portal and the underlying Azure virtual infrastructure where customer application instances and customer data reside.

You do not access to the Azure physical infrastructure and only see the above Azure virtual infrastructure through software defined network (SDN) capabilities. The main logical construct is the Azure Virtual Network (VNET) (see section § "Leveraging network virtualization capabilities" above) where customer can define subnets.

Multiple techniques are used to control information flows, including but not limited to:

- **Physical separation**. Network segments are physically separated by routers that are configured to prevent specific communication patterns.
- **Logical separation**. Virtual LAN (VLAN) technology is used to further separate communications (see below).
- **Firewalls**. Firewalls and other network security enforcement points are used to limit data exchanges with systems that are exposed to the Internet, and to isolate systems from back-end systems managed by Microsoft.
- **IDS/IPS** detect and identify suspicious or undesirable activities that indicate intrusion, proactively drop packets that are determined to be undesirable, and disconnect unauthorized connections.
- **Protocol restrictions.**
- **All traffic to and from customers** that are transmitted over encrypted connections.

Microsoft Azure implement boundary protection through the use of controlled devices at the network boundary and at key points within the network. The overarching principle of network security is to allow only connection and communication that is necessary to allow systems to operate, blocking all other ports, protocols and connections by default.

Access Control Lists (ACLs) are the preferred mechanism through which to restrict network communications by source and destination networks, protocols, and port numbers. Approved mechanisms to implement networked-based ACLs include:

- Tiered ACLs on routers managed by MCIO,
- IPSec policies applied to hosts to restrict communications (when used in conjunction with tiered ACLs),
- Firewall rules,
- Host-based firewall rules.

In addition, the guiding principle of our security strategy is to "assume breach", see section § "Vulnerability risk assessment" in the Appendix. The Microsoft global incident response team works around the clock to mitigate the effects of any attack against Azure, see section § Microsoft Cyber Defense Operations Center in the Appendix.

Resources:

- Inside Azure datacenter architecture with Mark Russinovich video.
- Azure information system components and boundaries.
- Isolation in the Azure Public Cloud.
- Azure network architecture.
- The Azure production network.
- Microsoft Azure Network Security.

One of the primary benefits of cloud computing is concept of a shared, common infrastructure across numerous customers simultaneously, leading to economies of scale. This concept is called multi-tenancy.

Multi-tenancy for the services

Microsoft Azure is a multi-tenant cloud services platform that you can use to deploy a vast variety of solutions. A multi-tenant cloud platform implies that multiple customer applications and data are stored on the same physical hardware.

Microsoft Azure was designed to help identify and counter risks inherent in a multitenant environment, and Microsoft works continuously to ensure that the multi-tenant architecture of its Microsoft Azure supports enterprise-level security, confidentiality, privacy, integrity, and availability standards to ultimately provide a secure hardened infrastructure.

Azure is designed with the assumption that all tenants are potentially hostile to all other tenants, and we have implemented security measures to prevent the actions of one tenant from affecting the security or service of another tenant, or accessing the content of another tenant.

The two primary goals of maintaining tenant isolation in a multi-tenant environment are:

1. Preventing leakage of, or unauthorized access to, customer content across tenants.
2. Preventing the actions of one tenant from adversely affecting the service for another tenant.

To accommodate (highly) sensitive data in the Azure public multi-tenant cloud, you can deploy additional technologies and services on top of those used for confidential data and limit provisioned services to those that provide sufficient isolation. These services offer isolation options at run time and support data encryption at rest using customer managed keys in dedicated single tenant Hardware Security Modules (HSMs) that are solely under your control.

Logical segregation for your applications

Azure uses logical isolation to segregate each customer's applications and data from those of others.

This approach provides the scale and economic benefits of multi-tenant cloud services while rigorously preventing customers from accessing one another's data or applications.

In the exposed software defined network (SDN), you have the ability to create virtual networks (see section § "Leveraging network virtualization capabilities" above). A virtual network (VNET) is a logical construct built on top of this SDN logical infrastructure. This helps ensure that network traffic in one customer deployments is not accessible to other Azure customers.

Fundamental to any shared cloud architecture is indeed the isolation provided for each customer to prevent one malicious or compromised customer from affecting the service or data of another. In Azure, one customer's subscription can include multiple deployments, and each deployment can contain multiple VMs. Azure provides network isolation at several points:

- Each deployment is isolated from other deployments. Multiple VMs within a deployment are allowed to communicate with each other through private IP addresses.
- Multiple deployments (inside the same subscription) can be assigned to the same VNET, and then allowed to communicate with each other through private IP addresses. Each VNET is isolated from other virtual networks.
- Traffic between VMs always traverses through trusted packet filters.
- Protocols such as Address Resolution Protocol (ARP), Dynamic Host Configuration Protocol (DHCP), and other OSI Layer-2 traffic from a VM are controlled using rate-limiting and anti-spoofing protection.

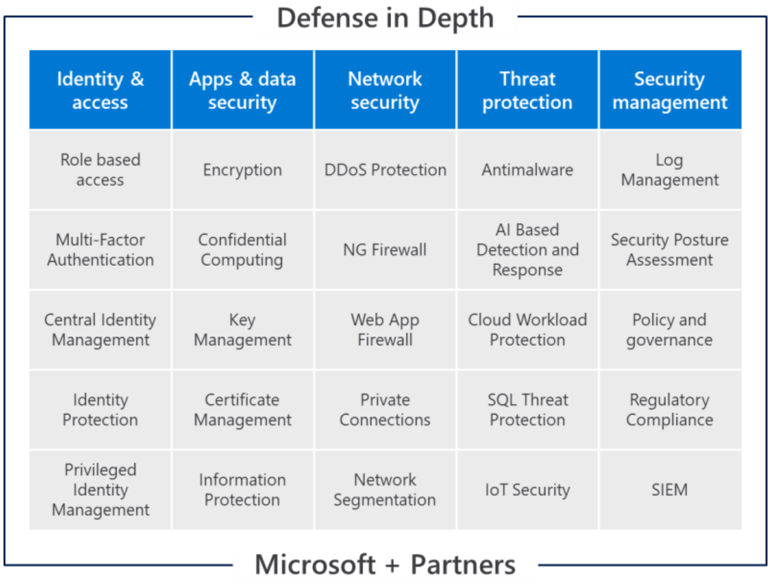
- VMs cannot capture any traffic on the network that is not destined for them.
- Customer VMs cannot send traffic to Azure private interfaces, or other customers' VMs, or Azure infrastructure services themselves (see above). Customer VMs can only communicate with other VMs owned or controlled by the same customer and with Azure infrastructure service endpoints meant for public communications.
- When you put VMs on a VNET, those VMs get their own address spaces that are completely invisible, and hence, not reachable from VMs outside of a deployment or virtual network (unless configured to be visible via public IP addresses). Your environments are open only through the ports that you specify for public access; if the VM is defined to have a public IP address, then all ports are open for public access.

Microsoft Azure separates customer VM-based computation resources from storage as part of its fundamental design. The separation allows computation and storage to scale independently, making it easier to provide multi-tenancy and isolation. Consequently, Azure Storage (See section § "Leveraging storage virtualization capabilities" above) runs on separate hardware with no network connectivity to Azure Compute except logically.

Beyond the above considerations, when you run your applications on Azure in virtual machines (VM), containers or as platform, adding security and operations management services is highly recommended so that you can control and customize how you manage your environment.

Building defense in depth with Azure built-in controls and partners' solutions

Microsoft Azure not only offers you unique security advantages derived from global security intelligence, a secure hardened infrastructure, BUT also sophisticated customer facing built-in security controls along with partner solutions to help you get protected faster across identity, network, and data, as well as providing tools to help them with security management and threat protection.



Defense in Depth

Identity & access	Apps & data security	Network security	Threat protection	Security management
Role based access	Encryption	DDoS Protection	Antimalware	Log Management
Multi-Factor Authentication	Confidential Computing	NG Firewall	AI Based Detection and Response	Security Posture Assessment
Central Identity Management	Key Management	Web App Firewall	Cloud Workload Protection	Policy and governance
Identity Protection	Certificate Management	Private Connections	SQL Threat Protection	Regulatory Compliance
Privileged Identity Management	Information Protection	Network Segmentation	IoT Security	SIEM

Microsoft + Partners

Figure 10 Defense in Depth

Network security

Network security ensures that assets homed in the various network segments are secured from malicious activity. In contrast to the perimeter security, network security deals with threats related to networks not directly exposed to external entities.

Typically, the following security controls would fall under this capability: network access controls, distributed denialof service (DDoS) protection, IDS/IPS on internal (virtual) networks, network segment firewalling, out-of-band administration networks, etc.

Security controls for Internet Edge security (North-South)

Azure provide a number of sophisticated customer facing controls that are native Azure controls or third-party network virtual appliances (NVAs) for Internet Edge security (North-South). (Nothing prevents to opt to a hybrid configuration where some VNets use advanced 3rd party controls and others use native controls.)

Native Azure controls

Following are some of the native Azure services to consider in this space:

- Azure Firewall, a managed, cloud-based network security service that provides centralized outbound and inbound (non-HTTP/S) network and application (L3-L7) filtering. It's a fully stateful firewall as a service with built-in high availability and unrestricted cloud scalability.
- Azure DDoS Protection, a managed, cloud-based network security service that, combined with application design best practices, provides extensive Distributed Denial of Service (DDoS) protection to help you protect your Azure resources from attacks. It provides the following service tiers:
    - **Basic**. Automatically enabled as part of the Azure platform. Always-on traffic monitoring, and real-time mitigation of common network-level attacks, provide the same defenses utilized by Microsoft Azure. The entire scale of Azure's global network can be used to distribute and mitigate attack traffic across regions. Protection is provided for IPv4 and IPv6 Azure public IP addresses.
    - **Standard**. Provides additional mitigation capabilities over the Basic service tier that are tuned specifically to Azure Virtual Network (VNET) resources (see below)

      The Standard service tier is simple to enable and requires no application changes. Protection policies are tuned through dedicated traffic monitoring and Machine Learning algorithms. Policies are applied to public IP addresses associated to resources deployed in VNETs, such as Azure Load Balancer and Azure Application Gateway. Application layer protection can be added through the Azure Application Gateway Web Application Firewall (WAF) (see below), or by installing a third-party firewall from Azure Marketplace. Protection is provided for IPv4 and IPv6 Azure public IP addresses.
- Azure Application Gateway Web Application Firewall (WAF), a web application firewall (WAF) that provides centralized inbound web application protection from common exploits and vulnerabilities. It's based on Core Rule Set (CRS) 3.0 or 2.2.9 from the Open Web Application Security Project (OWASP). The WAF automatically updates to include protection against new vulnerabilities, with no additional configuration needed.

- [Azure Front Door Service](), a scalable and secure entry point for fast delivery of globally distributed (microservice-based) applications that provides SSL offload and application acceleration at the edge close to end users, WAF and DDoS protection, etc. Azure Front Door works at OSI layer 7.

Following Figure 11 provides a depiction of a core services segment using native Azure controls (depicted within a single subscription). The Azure Firewall and WAF (in Application Gateway) are designed natively with high availability and don't require you to configure load balancers. WAFs functionality is depicted here with a public IPs, but you can also use private IPs in a VNET as a frontend as well.

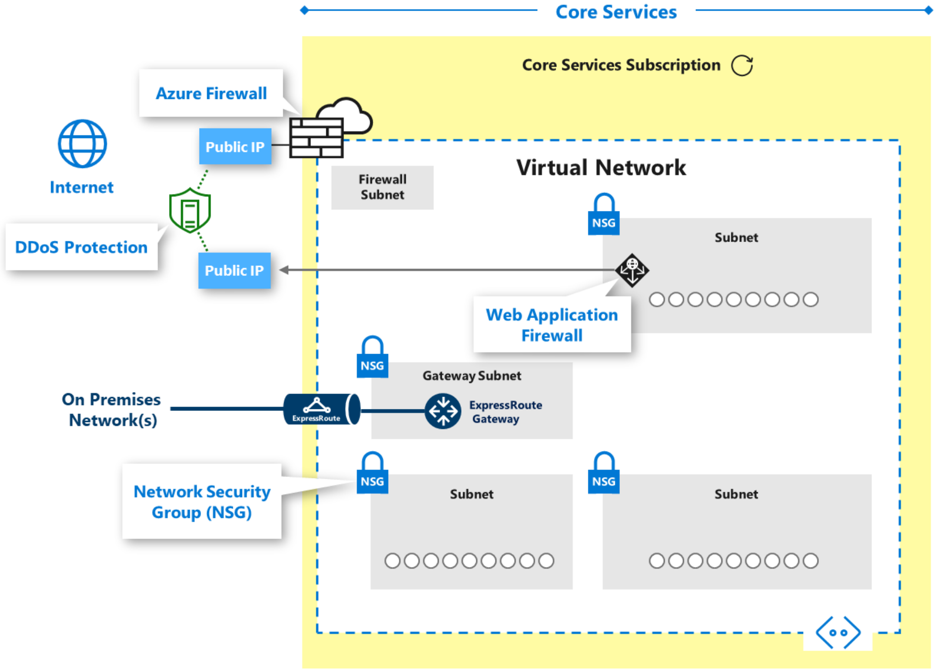


Figure 11 Core services segment using native Azure controls (depicted within a single subscription)

A Network Security Group (NSG) allows to express security rules for distributed inbound and outbound network (L3-L4) traffic filtering on VM, container or subnet, see section § "Security controls for East-West traffic" below.

Resources:

- What is Azure Firewall? .
- Azure DDoS Protection Standard overview.
- Azure DDoS Protection: Best practices and reference architectures.
- Web application firewall for Azure Application Gateway.
- What is Azure Front Door Service?.

The above services are notably in-scope services for the certification against the Service Organization Control (SOC) 1, SOC 2, and SOC 3 standards. Azure has been audited against the Service Organization Control (SOC) reporting framework and has achieved SOC 1 Type 2, SOC 2 Type 2, and SOC 3 reports:

- The SOC 1 Type 2 audit report attests to the design and operating effectiveness of Azure controls.
- The SOC 2 Type 2 audit included a further examination of Azure controls related to security, availability, and confidentiality.
- SOC 3 report is an abbreviated version of the SOC 2 Type 2 audit report.

Azure is audited annually against the SOC reporting framework by independent third-party auditors to ensure that security controls are maintained.

Resource:

- SOC Reports on the Service Trust Portal.

Third-party security appliances

The third-party security virtual appliances provide security controls such as Network Intrusion Detection/Prevention Systems (NIDS/NIPS), Next Generation Firewalls (NGFWs), encryption and more to your existing skillsets, processes, and licenses. These technologies are available as network virtual appliances (NVAs), see section § "Azure Virtual Network" above.

The Azure platform filters malformed packets and most classic NIDS/NIPS solutions are typically based on outdated signature-based approaches which are easily evaded by attackers and typically produce high rate of false positives.

Thus, you may want to deprecate and then discontinue some legacy security approaches as you move to Microsoft Azure. However, you can continue to use these technologies in Azure if you see value, but many organizations are not migrating these solutions to Azure.

Following Figure 12 is a depiction of a core services segment using a NGFW with built in Web Application Firewall (WAF) capabilities. Customers frequently choose this configuration to utilize their existing licensing and skillsets in Azure.

Note These are instantiated as VMs running the appliance (not a service like the native capabilities), so you need to configure the appropriate subnets, routing, network virtual applications (NVAs), and load balancers for a resilient architecture.

A public Azure Load Balancer (see eponym section above) enables scalability and availability while Azure DDoS Protection Standard can be applied to public IP addresses.

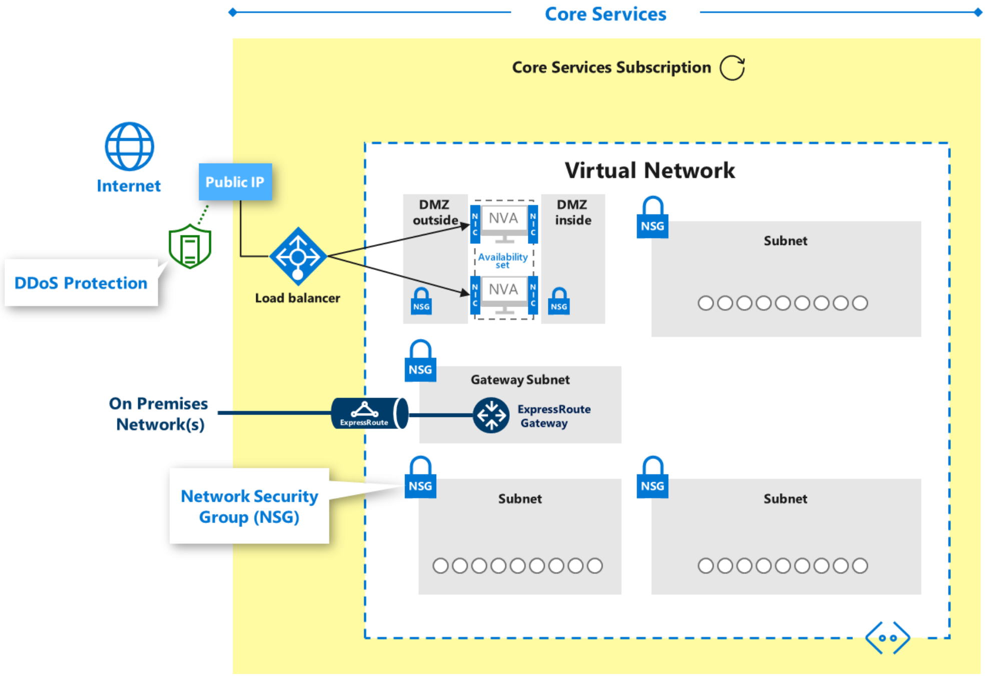


Figure 12 Next Generation Firewall with Integrated WAF/Proxy

Resources:

- Hub-spoke network topology with shared services in Azure.

- What is Azure Load Balancer?.

Security controls for East-West traffic

Azure Network security in the exposed virtual infrastructure is very similar to physical network security.

In this software defined network (SDN), subnets can only be created within a virtual network (VNET), see section § "Leveraging network virtualization capabilities" above. Azure requires VMs to be connected to a VNET.

Network security group (NSG) are used to protect against unsolicited traffic into subnets (replaces/supplements East-West traffic controls). A NSG holds a list of security rules that allow or deny inbound network traffic to, or outbound network traffic from, several types of Azure resources connected to VNETs. NSGs can be associated to subnets, individual network interface controllers (NICs) attached to VMs. When an NSG is associated to a subnet, the rules apply to all resources connected to the subnet. You can restrict traffic even further by also associating an NSG to a VM or NIC.

> **Note** While their use is not required, defining application security groups (ASGs) allow to simplify setup and maintenance of NSG rules. An ASG can be defined for lists of IP addresses that may change in the future, be used across many NSGs.

Following Figure 13 is a reference enterprise network design that depicts a core services segment and several example segments aligned. As already presented (see section § "Security controls for Internet Edge security (North-South)" above), the network edge security in the core services segment can use either native Azure controls or third party NVAs.

The illustrated shared services in the core services segment may be hosted in a single VNET or can span across multiple VNETs (e.g. for intranet vs. extranet resources):

- The core services segment includes examples of groupings we typically see in most enterprises.
- Each of the segments is connected to each other by VNET peering configurations.
- A public IP address may be mapped to an application within a segment that may not route through the network edge (depicted if you zoom into example applications). This activity can be restricted with permissions and/or routing.

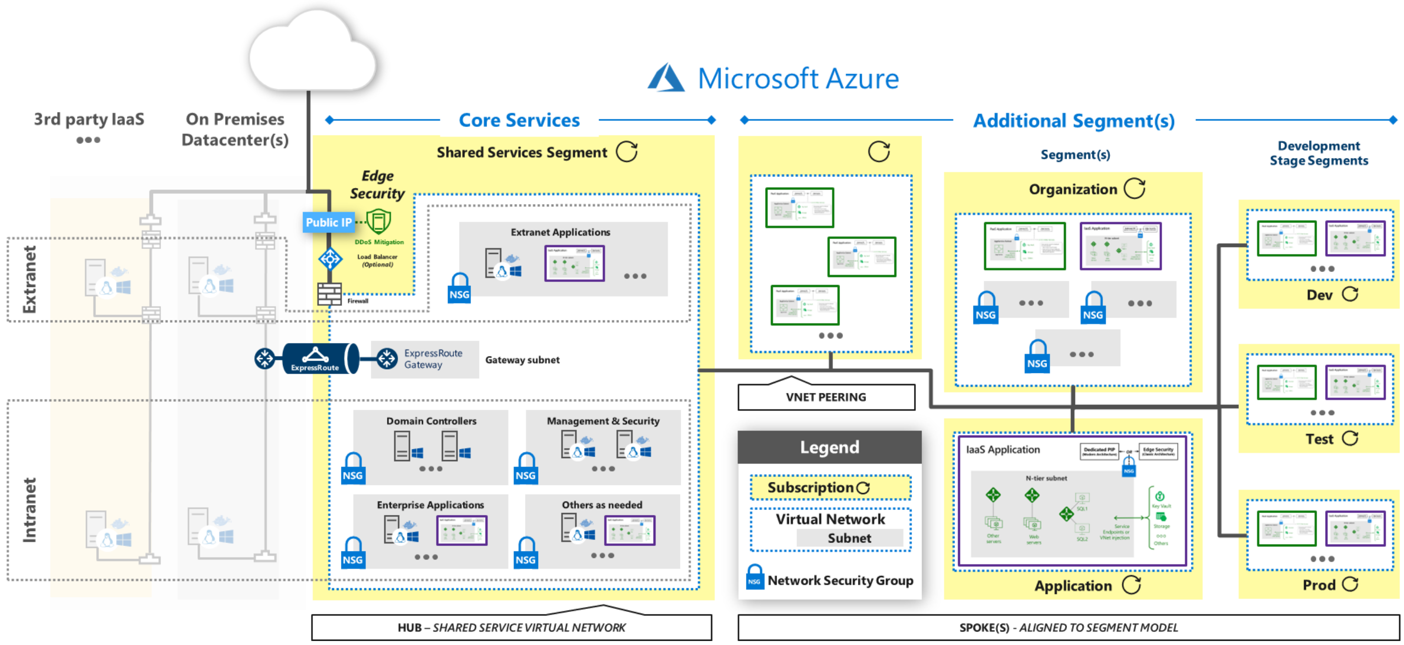


Figure 13 Cloud Infrastructure – Network Architecture

Azure Security Center provides unified security management and advanced threat protection across (hybrid) cloud workloads. It notably provides the ability to continuously monitoring the security status of your network. In this context, one should mention the Adaptive Network Hardening feature.

As such, Adaptive Network Hardening provides recommendations to further harden the NSG rules. It uses a Machine Learning algorithm that factors in actual traffic, known trusted configuration, threat intelligence, and other indicators of compromise, and then provides recommendations to allow traffic only from specific IP/port tuples. For example, let's say the existing NSG rule is to allow traffic from 140.20.30.10/24 on port 22 (SSH). The Adaptive Network Hardening's recommendation, based on the analysis, would be to narrow the range and allow traffic from 140.23.30.10/29 – which is a narrower IP range, and deny all other traffic to that port.

Resources:

- Azure network security overview.
- Azure Network Security Groups.
- Adaptive Network Hardening in Azure Security Center.
- Azure best practices for network security.

Options for VNET network visibility

Azure provides several means to get visibility into VNET network activity.

This includes:

- Getting verbose information during an investigation such as NSG flow logs, i.e. a subset of Azure Network Watcher. Azure Network Watcher is a regional service that enables you to monitor and diagnose conditions at the network level in, to, and from Azure. Its many diagnostic and visualization tools can help you understand and gain deeper insights into your network in Azure.

  Azure NSG flow logs allow you to view information about ingress and egress IP traffic through an NSG. Flow logs can be analyzed to gain information and insights into network traffic and security as well as performance issues related to traffic. While flow logs target NSGs, they are not displayed in the same way as other logs and are stored only within a storage account.

- Using Azure VNET Terminal Access Point (TAP) to continuously stream raw network traffic to a network packet collector or a security analytics tool. This offers the ability to add a virtual span port across a VNET.

  This requires a partner capability to consume this data but allows integration into existing network monitoring program/security analytics capability. You can use VNET Virtual Tap to continuously stream VM network traffic to the following NVAs: Big Switch Big Monitoring Fabric, Flowmon, Gigamon GigaSECURE, Fidelis Cybersecurity, Netscout vSTREAM, RSA NetWitness Platform, and more…

Resources:

- Introduction to flow logging for network security groups
- Virtual network TAP

Options for accessing Azure services and beyond

Azure Services are published to the Azure network via public endpoints by default. While these services are available to the internet, Azure services accessing them only traverse the Azure networks.

Some customers may have a preference or requirement that traffic to/from Azure services doesn't traverse the Azure network. Because of this, Azure provides the following options for accessing Azure services.

- **(VNET) Service endpoints** **(Service tunnel)**. Extend a VNET private address space and the identity of this VNET to the Azure PaaS services, over a direct connection. Endpoints allow you to secure your critical Azure service resources to only your VNETs. Traffic from your VNET to the Azure service always remains on the Microsoft Azure backbone network. This option is currently only available for a subset of Azure services.

  VNET service endpoint policies allow you to filter virtual network traffic to Azure services, allowing only specific Azure service resources, over service endpoints. Endpoint policies provide granular access control for VNET traffic to Azure services. This feature is currently available for a subset of Azure services and regions.

- **(VNet) Integration for Azure services**. Enables private access to the service from VMs or compute resources in the VM. You can integrate Azure services in your VNET with the following options:

  - Deploying dedicated instances of the service into a VNET. The services can then be privately accessed within the VNET and from on-premises networks. This option is currently only available for a subset of Azure services.

  - Using Azure Private Link to access privately an specific instance of the service from a VNET and from on-premises networks.

    Azure Private Link enables you to access Azure PaaS Services and Azure hosted customer/partner services over a private endpoint in a VNET. Traffic between the VNET network and the service traverses over the Microsoft backbone network, eliminating exposure from the public Internet. You can also create your own Private Link Service in your VNET and deliver it privately to your customers.

    The setup and consumption experience using Azure Private Link is consistent across Azure PaaS, customer-owned, and shared partner services.

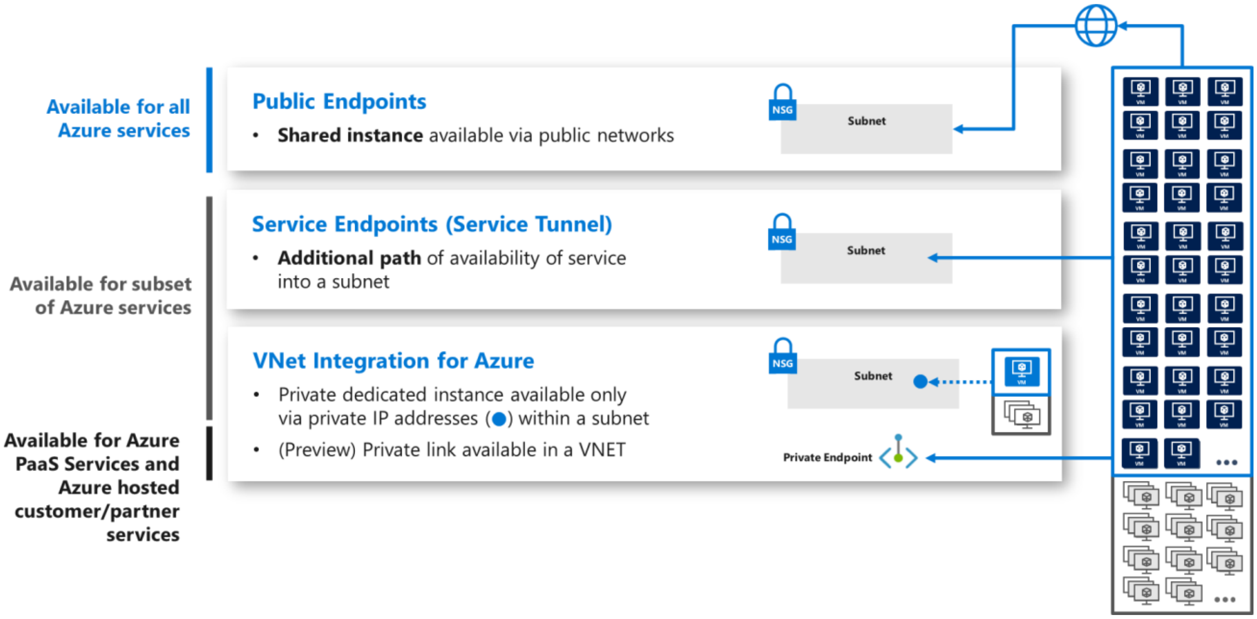

Available for all Azure services
Public Endpoints
• Shared instance available via public networks
NSG
Subnet
Available for subset of Azure services
Service Endpoints (Service Tunnel)
• Additional path of availability of service into a subnet
NSG
Subnet
VNet Integration for Azure
• Private dedicated instance available only via private IP addresses (●) within a subnet
• (Preview) Private link available in a VNET
NSG
Subnet
Private Endpoint
Available for Azure PaaS Services and Azure hosted customer/partner services
VM

Figure 14 Options for accessing Azure service and beyond

Resources:

- [Virtual Network Service Endpoints](#).
- [Virtual network service endpoint policies (Preview)](#).
- [Virtual network integration for Azure services](#).
- [Announcing Azure Private Link](#).
- [What is Azure Private Link? (Preview)](#).

Server/workload security

Server security considerations help ensure compute resources used in the customers' solutions implement the necessary level of controls to be secure. Typically, this comprises the following security controls: antimalware, operating system (OS) hardening, vulnerability management, etc.

As per section § "Understanding the shared responsibilities' model for your applications" above (see Figure 9 Security responsibilities transfer to the cloud), following security responsibilities are transferred to the CSP:

- For IaaS and PaaS workloads: Microsoft Azure Fabric (FC)/Virtualization patching.
- For PaaS workloads:
  - Security patches.
  - VMs/Containers security: Operating System (OS) and middleware installation, hardening, etc.

Note Container technology is causing a structural change in the cloud-computing world. Containers make it possible to run multiple instances of an application on a single instance of an operating system, thereby using resources more efficiently. Because container technology is relatively new, many IT professionals have security concerns about the lack of visibility and usage in a production environment.

The whitepaper [Container Security in Microsoft Azure](#) describes containers, container deployment and management, and native platform services. It also describes runtime security issues that arise with the use of containers on the Azure platform. In figures and examples, this paper focuses on Docker as the container model and Kubernetes as the container orchestrator.

See also section § "Security posture assessment" below.

Data security

Microsoft does NOT classify data uploaded and stored by customers and has no default access to customer data. Each customer will manage and access their data based on resources, access controls groups, identity-based authorization or keys depending on the considered specific resource and will specify the associated controls.

Azure Security Center that has been previously introduced includes capabilities that identify breaches and anomalous activities against storage accounts, SQL databases, data warehouse, and will be extending to other data services.

Access to customer data by Microsoft personnel

Microsoft takes strong measures to protect customer data from inappropriate access or use by unauthorized persons. Microsoft engineers do not have default access to customer data in the cloud. Instead, they are granted access, under management oversight, only when necessary. Using the restricted access workflow, access to customer data is carefully controlled, logged, and revoked when it is no longer needed. For example, access to customer data may be required to resolve customer-initiated troubleshooting requests.

The access control requirements are established by the following policy:

- No access to customer data, by default.
- No user or administrator accounts on customer virtual machines (VMs).
- Grant the least privilege that is required to complete task.
- Audit and log access requests.

Note For more information, see page Who can access your data and on what terms?.

Microsoft engineers can be granted access to customer data using temporary credentials via "just-in-time" (JIT) access. There must be an incident logged in the Azure Incident Management system that describes the reason for access, approval record, what data was accessed, etc. This approach ensures that there is appropriate oversight for all access to customer data and that all JIT actions (consent and access) are logged for audit.

Note Evidence that procedures have been established for granting temporary access for Azure personnel to customer data and applications upon appropriate approval for customer support or incident handling purposes is available from the Azure SOC 2 Type 2 attestation report produced by an independent third-party auditing firm and available on the on the Service Trust Portal.

Azure Customer Lockbox is a service that provides you with the capability to control how a Microsoft Engineer accesses your data. As part of the Support workflow, a Microsoft engineer may require elevated access to customer data. Azure Customer Lockbox puts the customer in charge of that decision by enabling the customer to Approve/Deny such elevated requests. Azure Customer Lockbox is an extension of the JIT workflow and comes with full audit logging enabled. It is important to note that Azure Customer Lockbox capability is not required for support cases that do not involve access to customer data. For the majority of support scenarios, access to customer data is not needed and the workflow should not require Azure Customer Lockbox. Azure Customer Lockbox is available to customers from all Azure public regions.

Resources:

- Azure customer data protection.
- Approve, audit support access requests to VMs using Customer Lockbox for Azure.
- Customer Lockbox for Microsoft Azure.

Customer data sanitization

When you delete data or leave Microsoft Azure, Microsoft follows strict standards for overwriting storage resources before reuse. As part of agreements for cloud services such as Azure Storage (see section § "Leveraging storage virtualization capabilities" above), Azure Virtual Machine (see section § "Leveraging compute virtualization capabilities" above), etc. Microsoft contractually commits to timely deletion of data.

Data destruction techniques vary depending on the type of data object being destroyed, whether it be whole subscriptions themselves, storage, virtual machines, or databases. In a multi-tenant environment such as Microsoft Azure, careful attention is taken to ensure that when a customer deletes data, no other customer (including, in most cases, the customer who once owned the data) can gain access to that deleted data.

(Likewise, in terms of equipment disposal, upon a system's end-of-life, Microsoft operational personnel follow rigorous data handling procedures and hardware disposal processes to help assure that no hardware that may contain customer data is made available to untrusted parties.)

Resource:

- Microsoft Azure Data Security (Data Cleansing and Leakage)

Customer data encryption

Data encryption in the cloud is an important risk mitigation requirement expected by customers worldwide. Microsoft Azure helps you protect your data through its entire lifecycle whether at rest, in transit, or even in use. Azure has extensive support to safeguard customer data using data encryption in transit and at rest, as well as data encryption while in use.

Encryption in transit

Microsoft Azure enforces encryption support for:

- **Data in transit between a user and the service** to protect the user against the interception of their communications and ensure the integrity of operations.
- **Data in transit between datacenters** to protect against mass data interception.
- **Communications from end to end between users** to protect against interception or loss data in transit between users.

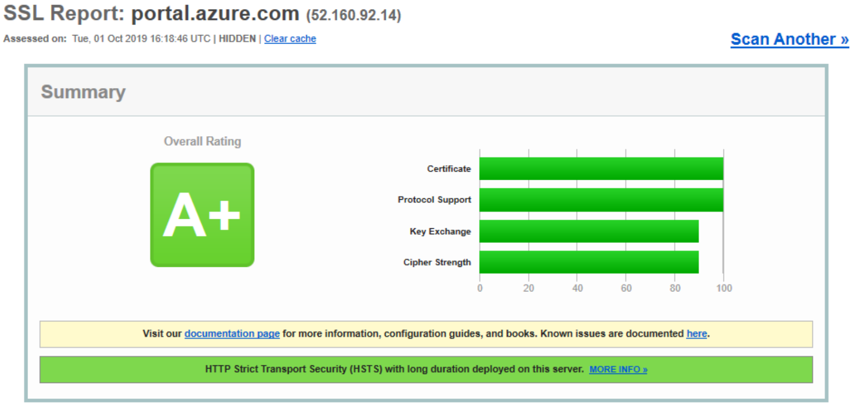
Whenever applicable, Microsoft Azure strives to use industry-accepted standard transport protocols such as TLS between devices and Microsoft datacenters, and within datacenters themselves.

Azure uses the industry-standard Transport Layer Security (TLS) 1.2 protocol with 2048-bit RSA/SHA256 encryption keys, to encrypt communications both between the customer and Azure services, and also internally between Azure systems and data centers.

For example, when administrators use the Azure portal to manage the service for their organization, the data transmitted between the portal and the administrator's device is sent over an encrypted TLS channel. Moreover, you have to be authenticated against your Azure Active Directory (Azure AD) tenant before being in a position to conduct any administrative tasks. (Azure AD provides in this context additional capabilities like conditional access, passwordless or multi-factor authentication (MFA), privileged identity management (PIM), etc., see section § "Identity and access management" below) Same considerations apply for all publicly accessible endpoints.

TLS provides strong authentication, message privacy, and integrity. Perfect Forward Secrecy (PFS) protects connections between customer's client systems and Microsoft Azure services by generating a unique session key for every session a customer initiates. PFS protects past sessions against potential future key compromises. Connections also use RSA-based 2,048-bit encryption key lengths. This combination makes it more difficult to intercept and access data that is in transit.

All implementation details such as the version of TLS being used, whether or not PFS is enabled, the order of cipher suites, etc., are available publicly. One way to see these details consists in using a third-party Web site, such as Qualys SSL Labs. Following is the link to the automated test page from Qualys that display information for the Azure portal: https://www.ssllabs.com/ssltest/analyze.html?d=portal.azure.com&hideResults=on&latest.



SSL Report: portal.azure.com (52.160.92.14)

Assessed on: Tue, 01 Oct 2019 16:18:46 UTC | HIDDEN | Clear cache

Visit our documentation page for more information, configuration guides, and books. Known issues are documented here.

HTTP Strict Transport Security (HSTS) with long duration deployed on this server. MORE INFO »

Figure 15 Qualys SSL report for the Azure portal

Furthermore, Azure offers its customers a range of options for securing their own data and traffic. You can enable for example encryption for traffic between your own virtual machines (VMs) and end-users. The certificate management features built into Azure give administrators flexibility for configuring certificates and encryption keys for management systems, individual services, secure shell (SSH) sessions, VPN connections, remote desktop (RDP) connections, and other functions.

> **Note** You should review Azure best practices for the protection of data in transit and properly configure HTTPS endpoints for your resources provisioned in Azure to help ensure that all traffic going to and from your VMs is encrypted. For key Azure services, data encryption in transit is enforced by default.

Azure optional services helps you safeguard cryptographic keys and secrets by storing them in hardware security modules (HSMs), see section § "Encryption management" below and § "Secret management" below.

Encryption at rest

In most situations, not to say in all of them, data is stored encrypted (i.e. at rest) in Azure and decrypted on the fly when used or computed by a program. This is both a usual and an adapted way to proceed for the most common data.

Microsoft Azure provides extensive options for data encryption at rest to help you safeguard your data and meet your compliance needs using both Microsoft managed encryption keys, as well as customer managed encryption keys, giving you the flexibility to choose the solution that best meets your needs.

To take some example, Azure Cosmos DB requires no action from you - data stored in Azure Cosmos DB in nonvolatile storage (solid-state drives) is encrypted by default (, and there are no controls to turn it on or off).

Transparent data encryption (TDE) helps protect Azure SQL Database (and Azure SQL Data Warehouse). Data and log files are encrypted using industry-standard encryption algorithms. Pages in a database are encrypted before they're written to disk and decrypted when they're read.

SQL Always Encrypted encrypts data within client applications prior to storing it in Azure SQL Database (and Azure SQL Data Warehouse). It allows delegation of on-premises database administration to third parties, and maintains separation between those who own and can view the data and those who manage it but should not access it.

Azure Storage Service Encryption for Data at Rest ensures that customer data is automatically encrypted before persisting it to Azure Storage (see section § "Leveraging storage virtualization capabilities" above) and decrypted before retrieval. All data written to Azure Storage is encrypted through 256-bit AES encryption, and the handling of encryption, decryption, and key management in Storage Service Encryption is transparent to customers. However, you can also use your own encryption keys for Azure Storage encryption at rest and manage your keys in Azure Key Vault. Storage Service Encryption is enabled by default for all new and existing storage accounts and cannot be disabled. (Azure client-side encryption supports encrypting data within client applications before uploading to Azure Storage or other endpoints, and then decrypting data when downloading it to the client.)

Encryption support is in place for your IaaS virtual machines (VMs) with Azure Disk Encryption, enabling you to encrypt your Linux IaaS VM disks (see section § "Leveraging compute virtualization capabilities" above). Azure Disk encryption leverages the DM-Crypt feature of Linux to provide volume encryption for the OS and data disks. The solution is integrated with Azure Key Vault (see section § "Encryption management" below) to help you control and manage the disk encryption keys.

Resources:

- Azure Data Encryption-at-Rest.
- Data encryption in Azure Cosmos DB.
- Transparent data encryption for SQL Azure Database and Data Warehouse.
- Always Encrypted for Azure SQL Database and Data Warehouse.
- Azure Storage encryption for data at rest.
- Configure customer-managed keys for Azure Storage encryption from the Azure portal.
- Azure Disk Encryption overview.

Encryption in use

Sometimes, protecting data in transit and data at rest is not enough. Data may be indeed too sensible to appear in clear in memory (i.e. in use), even if the (virtual) machine and the workload processing data can be considered hardened respectively secured. (In some cases, your sensitive content is the code and not the data. To secure sensitive IP, you may require protect confidentiality and integrity of your code while it's in use.)

Increasing popularity of use cases, such as privacy-preserving multi-party machine learning, has led to secure compute workloads within the confines of Trusted Execution Environments (TEEs).

This concept called Confidential Computing is an ongoing effort to protect data and/or code throughout its lifecycle at rest, in transit and now in use. This means that data can be processed in the cloud with the assurance that it is always under customer control. Confidential Computing ensures that when data is in the clear, which is needed for efficient data processing in memory, the data is protected inside a TEE, (a.k.a. as an enclave). TEE ensures that there is no way to view data or the operations from outside the enclave, even with a debugger. Moreover, TEE ensures that only authorized code is permitted to access data. If the code is altered or tampered with, the operations are denied, and the environment is disabled.

With the introduction of Azure Confidential Computing (ACC) (with the DC-series of virtual machines (VMs) (see section § "Azure Virtual Machines" above) that have the latest generation of Intel Xeon processors with Intel Software Extension Guard (SGX) technology - The Intel SGX instruction extension was introduced with 7th Generation Intel

Core processor platforms and Intel Xeon processor E3 v5 for data center servers back in 2015 -, Azure is the very first cloud environment that allows developers to create TEEs-based applications for deployment in the cloud.

> **Note** For more information on Azure Confidential Computing, see blog post Introducing Azure confidential computing and the webcast Azure Confidential Computing updates with Mark Russinovich | Best of Microsoft Ignite 2018.

> **Note** Intel SGX isolates a portion of physical memory to create an enclave where select code and data are protected from viewing or modification. The protection offered by Intel SGX, when used appropriately by application developers, can prevent compromise due to attacks from privileged software and many hardware-based attacks.
>
> An application leveraging Intel SGX needs to be re-factored into trusted and untrusted components. The untrusted part of the application sets up the enclave, which then allows the trusted part to run inside the enclave. No other code, irrespective of the privilege level, has access to the code executing within the enclave or the data associated with enclave code. Design best practices call for the trusted partition to contain just the minimum amount of content required to protect customer's secrets.

But all of this is still really a low-level work and developing applications above that is really difficult and requires both advanced security expertise and specifics skills. In this context, Microsoft Research, together with partners, has embarked and invested in a way that simplifies TEE-based application development for all audiences from hardcore hardware security experts to edge and cloud software applications developers, regardless of the underlying enclaving technologies. The effort results in the Microsoft Open Enclave SDK, an open source framework available on GitHub over a year ago, which developers can use to build C/C++ enclave applications targeting Intel SGX technology as well as ARM TrustZone (TZ), and embedded Secure Elements using Linux OSs.

The Open Enclave SDK is intended to be portable across enclave technologies, cross platform – cloud, hybrid, edge, or on-premises, and designed with architectural flexibility in mind. As such, the Open Enclave SDK aims at creating a single unified enclaving abstraction for developer to build applications once that run across multiple TEE architectures, and thus was designed to:

- Make it easy to write and debug code that runs inside TEEs.
- Allow the development of code that's portable between TEEs.
- Provide a flexible plugin model to support different runtimes and cryptographic libraries.
- Have a high degree of compatibility with existing code.

This SDK is natively leverage by ACC.

> **Note** Microsoft has recently joined partners and the Linux Foundation to create Confidential Computing Consortium that will be dedicated to defining and accelerating the adoption of confidential computing.
>
> CONFIDENTIAL COMPUTING
> CONSORTIUM
>
> Microsoft will be contributing the Open Enclave SDK to the Confidential Computing Consortium to develop a broader industry collaboration and ensure a truly open development approach. "The Open Enclave SDK is already a popular tool for developers working on Trusted Execution Environments, one of the most promising areas for protecting data in use," said Mark Russinovich, chief technical officer, Microsoft. "We hope this contribution to the Consortium can put the tools in even more developers' hands and accelerate the development and adoption of applications that will improve trust and security across cloud and edge computing." (see blog post New Cross-Industry Effort to Advance Computational Trust and Security for Next-Generation Cloud and Edge Computing).

You can now build, deploy, and run applications that protect data confidentiality and integrity through the entire data lifecycle whether at rest, in transit, or in use. To get started, you can deploy a DC-series VM through the custom deployment flow in Azure Marketplace. The custom deployment flow deploys and configures the VM and installs the Open Enclave SDK for Linux VMs if selected.

Furthermore, as recently announced, you can also now create a Kubernetes cluster on hardware that supports Intel SGX, such as the above DC-series virtual machines running Ubuntu 16.04 or Ubuntu 18.04 and install a confidential computing device plugin into those VMs. The device plugin (running as a DaemonSet) surfaces the usage of the Encrypted Page Cache (EPC) RAM as a schedulable resource for Kubernetes. As a Kubernetes user, you can then schedule pods and containers that use the Open Enclave SDK onto hardware which supports TEEs.

Aside from controls implemented by Microsoft to safeguard customer data, customers deployed in Azure derive considerable benefits from security research that Microsoft conducts to protect the cloud platform. Microsoft global threat intelligence is one of the largest in the industry and it is derived from one of the most diverse sets of threat telemetry sources. It is both the volume and diversity of threat telemetry that makes Microsoft Machine Learning algorithms applied to that telemetry so powerful. See section § "Microsoft Cyber Defense Operations Center" in Appendix.

Encryption management

Azure offers comprehensive encryption (key) management to help you control your keys in the cloud, including key rotation, key deletion, permissions, etc. End-to-end data encryption using advanced ciphers is fundamental to ensuring confidentiality and integrity of customer data in the cloud.

Azure Key Vault is a multi-tenant secrets management service that enables Azure services, applications and users to store and use several types of secret/key data:

- **Secrets**. Provides secure storage of secrets, such as passwords and database connection strings.
- **Cryptographic keys**. Supports multiple key types and algorithms. As of this writing, Azure Key Vault supports RSA and Elliptic Curve keys.
- **Certificates**. Supports X.509 certificates, which are built on top of keys and secrets and add an automated renewal feature.
- **Keys of an Azure Storage**. Can manage keys of an Azure Storage account. Internally, Key Vault can list (sync) keys with an Azure Storage account, and regenerate (rotate) the keys periodically.

Azure Key Vault enables the use of Hardware Security Modules (HSM) for high value keys: you can import or generate keys in HSMs that never leave the HSM boundary to support Bring Your Own Key (BYOK) scenarios. Azure Key Vault HSMs are FIPS 140-2 Level 2 validated, which includes requirements for physical tamper evidence and role-based authentication.

Azure Key Vault is designed, deployed, and operated such that Microsoft and its agents are precluded from accessing, using or extracting any data stored in the service, including cryptographic keys.

> **Note** With Azure Stack, you can store and manage your secrets including cryptographic keys on an external Hardware Secure Module (HSM) by using Thales CipherTrust Cloud Key Manager (available via the Azure marketplace), which allows customers to integrate an HSM with Key Vault service running on Azure Stack.

If you expect to use back-end services integrated with Azure Key Vault (e.g., Azure Storage, Azure Disk Encryption, Azure Data Lake Storage, etc.) for customer managed key support, then you need to use Microsoft provided cryptography and encryption hardware, e.g. HSMs.

For customers who require single-tenant HSMs, Azure provides dedicated HSMs that have FIPS 140-2 Level 3 validation, as well as Common Criteria EAL4+ certification and conformance with eIDAS requirements. Azure Dedicated HSM is most suitable for "lift-and-shift" scenarios where you require full administrative control and sole access to your HSM device for administrative purposes and retain your crypto algorithms. After a Dedicated HSM device is provisioned, only you have administrative or application-level access to the device. Dedicated HSMs are provisioned directly on your virtual network and can also connect to on-premises infrastructure via VPN.

Resources:

- What is Azure Key Vault?.
- About keys, secrets, and certificates in Azure Key Vault.
- How to generate and transfer HSM-protected keys for Azure Key Vault.
- What is Azure Dedicated HSM?.

Secret management

As stated in the above section, Azure Key Vault is a multi-tenant secrets management service that stores and controls access to secrets.

In addition, for customers that already rely on HashiCorp Vault to store and manage their secrets, Vault can also help them manage and eliminate secrets sprawl in Azure.

HashiCorp Vault is a highly scalable, highly available, environment agnostic way to generate, manage, and store secrets. It encrypts data using the Advanced Encryption Standard (AES) 256 bits. (Once data is encrypted it is stored on a variety of storage backends such as HashiCorp Consul.)

Working with Microsoft, HashiCorp launched Vault with a number of features to make secret management easier to automate in Azure cloud. (HashiCorp and Microsoft are longstanding partners in the cloud infrastructure community.)

As a result, you can leverage all of these Vault features to automate your secrets management and retrieval through Azure specific integrations. First and foremost Vault can be automatically unsealed using KMS keys from Azure Key Vault. Next, as already outlined, Azure managed identities can be used to authenticate systems and applications preventing the need to distribute initial access credentials.

Lastly, HashiCorp Vault can dynamically generate Azure AD service principals for apps using its Azure secrets engine feature. "The Azure secrets engine dynamically generates Azure service principals and role assignments. Vault roles can be mapped to one or more Azure roles, providing a simple, flexible way to manage the permissions granted to generated service principals. Each service principal is associated with a Vault lease. When the lease expires (either during normal revocation or through early revocation), the service principal is automatically deleted.

If an existing service principal is specified as part of the role configuration, a new password will be dynamically generated instead of a new service principal. The password will be deleted when the lease is revoked."

This allows users and applications off-cloud an easy method for generating flexible time and permission bound access into Azure APIs.

Note For customers that would like a quick way of testing out Vault in Azure, the hc-demos repo on GitHub contains all the code to create a Vault environment in Azure including all instructions on how to obtain Terraform, run it, connect to their Azure instance and run the Vault commands. This is a great way to learn the concepts covered here with a low barrier to entry.

Note For more information on HashiCorp Vault and Azure integrations, see page Hashicorp/Azure Integrations.

Resources:

- What is Azure Key Vault?.
- How to identify and eliminate secrets sprawl on Azure with HashiCorp Vault.
- Azure Friday: Azure Key Vault Auto Unseal & Dynamic Secrets with HashiCorp Vault on HashiCorp documentation.
- HashiCorp auto-unseal using Azure Key Vault on HashiCorp documentation.

- Vault server configuration with Azure Key Vault on HashiCorp documentation.
- Azure Secrets Engine on HashiCorp documentation.

Identity and access management

Azure Active Directory (Azure AD) provides the identity and access management (IAM) capabilities behind Microsoft Azure for use by the customers of its catalog of services.

Azure AD is an identity repository and cloud service that provides authentication, authorization, and access control for an organization's users, groups, and objects. Azure AD can be used as a standalone cloud directory or as an integrated solution with existing on-premises Active Directory to enable key enterprise features such as directory synchronization and single sign-on (SSO).

Each Azure subscription points to an Azure AD.

Note Azure Stack uses either Azure AD or Active Directory Federation Services (AD FS) as an identity provider.

Azure AD is trusted by millions of organizations serving more than of a billion of identities for access to Azure, as well as hundreds of thousands of other partner applications.

KuppingerCole has named Microsoft as the top overall leader in their Leadership Compass for Identity-as-a-Service (IDaaS) Access Management 2019. Microsoft was identified as the "leading IDaaS AM vendor" for functional strength and completeness of product, with a "focus on constant innovation". The report analyzed 15 vendors across three categories of leadership and Microsoft earned the highest scores across product, innovation, and market leadership.

Note KuppingerCole recognized Microsoft's strength in our "support for popular SaaS app integrations," which is possible through the open support of our many partners. KuppingerCole highlighted the security capabilities of Azure Active Directory including "strong adaptive authentication" and "strong threat analytics capabilities offering real-time threat detection and remediation."

KuppingerCole also acknowledged our accelerating app development support saying we are "increasingly DevOps friendly with strong developer community support." Through open standards, a secure authentication platform, and APIs we're committed to help customers create the next generation of apps and experiences.

For more information, see blog post KuppingerCole names Microsoft the top overall IDaaS leader and complimentary copy of the report.

Recently, Gartner[2] named Microsoft a Leader in the Magic Quadrant for Access Management, Worldwide 2019 for the third year in a row. Additionally, Forrester Research named Microsoft a Strong Performer in its report The Forrester Wave: Identity-As-A-Service (IDaaS) For Enterprise, Q2 2019, with the largest market presence across vendors.

The document Azure Active Directory Data Security Considerations explains the following technical aspects of Azure AD

[2] Gartner does not endorse any vendor, product or service depicted in its research publications, and does not advise technology users to select only those vendors with the highest ratings or other designation. Gartner research publications consist of the opinions of Gartner's research organization and should not be construed as statements of fact. Gartner disclaims all warranties, expressed or implied, with respect to this research, including any warranties of merchantability or fitness for a particular purpose.

- **Azure AD Components**. What are the different components of Azure AD.
- **Core Data and Location**. What customer data is used by Azure AD and where is it located.
- **Data Protection**. How is the directory data protected at transit and at rest.
- **Data Flow**. How data from various sources such as on premises directories and applications flows to and from Azure AD.
- **Data and Operations**. What data and operational procedures are used by the Azure AD engineering team to manage the service.

We advise referring to it for the technical details that pertain to Azure AD.

This said, in order to manage access to resources in Microsoft Azure in the context of this document, Azure AD allows to i) assign user accounts to specific built-in Azure AD organizational roles, including (but not limited to) Global Administrator, Billing Administrator, Service Administrator, User Administrator, Password Administrator, etc., ii) manage the (automatic) assignment of user accounts types of user to distinguish external users for B2B collaboration, and iii) manage and control (dynamic) security group memberships.

Optional capabilities provide "just-in-time" and "just-enough" (JIT) administrative access to users for ad hoc tasks for short predetermined periods for Azure, see section § "Privileged identity management" below.

(Microservices-based) application and API developers can integrate their applications and APIs with Azure AD to quickly provide single sign-on (SSO) functionality for their users and to benefit from these capabilities, thanks to Azure AD support of industry identify standards such as Security Assertions Markup Language (SAML) 2.0 or Open ID Connect (OIDC).

If access also requires users, and groups to be provisioned into applications, Azure AD is all-on System for Cross-domain Identity Management (SCIM), a specification that provides a common user schema to help users move into, out of, and around applications. SCIM is becoming the de facto standard for provisioning and, when used in conjunction with federation standards like above SAML 2.0 or OIDC, provides both developers and administrators an end-to-end standards-based solution for IAM. Developers that build an SCIM endpoint can integrate with any SCIM-compliant client without having to do custom work. Microsoft provides a generic SCIM client that can push users and groups from Azure AD into a target application.

Developers can also use the Microsoft Graph API to query directory data for managing entities such as users or groups.

All of the above enable Enterprise-grade applications to be hosted in the cloud and to seamlessly provision users if required, and authenticate them with corporate credentials for your applications and workloads. (It also enables SaaS providers to make authentication and authorization decisions easier for users in Azure AD organizations when accessing to their services.)

Azure AD allows the configuration of passwordless authentication methods and multi-factor authentication (MFA) access rules for any Azure AD connected application.

In addition, Azure AD Conditional Access lets you apply the right access controls by implementing automated access control decisions based on the required conditions.

Conditional Access is at the heart of a Zero Trust approach implementation. Zero Trust is the next evolution in network security.

Note	For more information, see whitepaper Implementing a Zero Trust approach with Azure Active Directory.

The state of cyberattacks drives organizations to take the "assume breach" mindset, but this approach shouldn't be limiting. Zero Trust networks protect corporate data and resources while ensuring that organizations can build a modern workplace by using technologies that empower employees to be productive anytime, anywhere, in any way.

Note “Assume breach” assumes that attackers will be able to get in. If an attack is successful, you must be prepared to mitigate the impact through effective detection and response capabilities. This assumption necessitates greater emphasis on and investment in early detection and rapid response efforts.

An optional capability, i.e. Azure AD Identity Protection, enables organizations to configure automated responses to detected suspicious actions related to user identities. The vast majority of security breaches take place when attackers gain access to an environment by stealing a user’s identity. Discovering compromised identities is no easy task. Azure AD uses adaptive Machine Learning algorithms and heuristics to detect anomalies and suspicious incidents that indicate potentially compromised identities.

Unsurprisingly, Azure AD Identity Protection fully integrate with Azure AD Conditional Access policies.

Resources:

- Azure identity management security overview.
- What is Azure Active Directory?.
- A world without passwords with Azure Active Directory.
- What is passwordless?.
- What is role-based access control (RBAC) for Azure resources?.
- What is Conditional Access?.
- Manage access to Azure management with Conditional Access.
- What is Azure Active Directory Identity Protection?.

Role-based access control (RBAC)

Azure built its Role-Based Access Control (RBAC) capabilities on top of the above to enforce the separation of privileged and non-privileged roles for Azure resources. Using RBAC, users, managed identities, service groups, and applications (a.k.a. service principals) from that directory can be granted access to resources in the Azure subscription at a subscription, resource group, or individual resource level. For example, a storage account (see section § “Leveraging storage virtualization capabilities” above) can be placed in a resource group to control access to that specific storage account using Azure AD - Resource groups in Azure provide a way to monitor, control access, provision and manage billing for collections of Azure assets/resources that are required to run an application -. In this manner, only specific users can be given the ability to access the Storage Account Key, which controls access to storage.

All accesses to Azure resources are based on a prior authentication on Azure AD and use in turn the RBAC mechanism. Azure RBAC comes with the built-in roles that can be assigned to users, groups, and services. You can therefore use predefined roles to give the necessary permissions to users in charge of managing backups or if necessary create your own custom roles to fit your needs, for example if you use custom or third party solutions to manage log events.

Note With Azure Stack, you can similarly use RBAC to grant system access to authorized users, groups, and services by assigning them roles at a subscription, resource group, or individual resource level. Each role defines the access level a user, group, or application has over Azure Stack resources.

The built-in Azure RBAC capabilities allow to list all the roles that are assigned to a specified user under the customer's responsibility and the roles that are assigned to the groups to which the user belongs. Conversely, these capabilities also allow to see all the access assignments for a specified subscription, resource group, or resource. All of this can be achieved notably via the Get-AzureRmRoleAssignment cmdlet.

Resources:

- What is role-based access control (RBAC) for Azure resources?.
- Built-in roles for Azure resources.
- Custom roles for Azure resources.

- Manage Role-Based Access Control with Azure PowerShell.

Managed identities for accessing your Azure services

A common challenge when building cloud applications is how to manage the credentials in code for authenticating to cloud services. Keeping the credentials secure is an important task. Ideally, the credentials never appear on developer workstations and aren't checked into source control. Azure Key Vault (see section § "Encryption management" above) provides a way to securely store credentials, secrets, and other keys, but your code has to authenticate to Key Vault to retrieve them.

The Azure managed identities for Azure resources feature in Azure AD solves this problem. The feature provides Azure services with an automatically managed identity in Azure AD. You can use the identity to authenticate to any service that supports Azure AD authentication, including Azure Key Vault, Azure Container Instances (ACI), Azure Kubernetes Service (AKS) (see section § "Leveraging containerization" above), etc. without any credentials in your code.

Managed identities can also be used for Infrastructure as Code (IaC) (see section § "Are you saying "Infrastructure as Code" below). For example, for customers that rely on HashiCorp Terraform, the Azure Resource Manager module for Terraform can be authenticated using managed identities for Azure resources. Same applies to HashiCorp Vault.

Resources:

- What is managed identities for Azure resources?.
- How to use Azure managed identities with Azure Container Instances.
- How to use Azure managed identities with Azure Kubernetes Services (AKS).
- Azure Provider: Authenticating using managed identities for Azure resources on Terraform documentation.
- Azure Authentication with HashiCorp Vault on vault documentation.
- Using Azure Active Directory Authentication with HashiCorp Vault – Part 1 & Part 2.

Privileged identity management

Azure AD provides a number of optional capabilities for privileged access management and identity governance.

Azure AD Privileged Identity Management (PIM) provides a mechanism for managing and monitoring administrators in Azure AD and granting on-demand, temporary, "just-in-time" (JIT) and "just-enough" administrative access to users for ad hoc tasks for short predetermined periods for Microsoft Azure.

Azure AD Access Reviews enable organizations to efficiently manage group memberships, role assignments, and access to enterprise applications. User's access can be reviewed on a regular basis to make sure only the right people have continued access.

Resources:

- What is Azure AD Privileged Identity Management?.
- What is Azure AD Identity Governance?.
- What are Azure AD access reviews?.

Security management

Log management

Log management allows from a security perspective to identified all alerts from all monitored applications, systems and infrastructure. This includes log analysis, initial diagnosis and where required, dispatching.

Azure Security Center collects data from your Azure virtual machines (VMs), virtual machine scale sets, IaaS containers, and non-Azure (including on-premises) computers to monitor for security vulnerabilities and threats.

Data is collected using the Azure Log Analytics agent, which reads various security-related configurations and event logs from the machine and copies the data to your workspace for analysis. Examples of such data are: operating system type and version, operating system logs, running processes, machine name, IP addresses, and logged in user. The Azure Log Analytics agent also copies crash dump files to your workspace.

Resources:

- What is Azure Security Center?.
- Data collection in Azure Security Center.

Security posture assessment

You can use Azure Security Center to notably analyze the security state of your compute resources and the configurations of the security controls that are in place to protect them. Azure Security Center notably provides capabilities in the following main areas:

1. **Cloud security posture management**. Azure Security Center provides you with a bird's eye security posture view across your Azure environment, enabling you to continuously monitor and improve your security posture using the Azure secure score. Azure Security Center helps you identify and perform the hardening tasks recommended as security best practices and implement them across notably your compute resources (see section § "Azure compute solutions" Azure compute solutions). This includes in particular managing and enforcing your security policies and making sure your Azure Virtual Machine instances (VMs), Azure VM Scale Sets, and non-Azure servers - Azure Security Center integrates with Azure Stack - are compliant.

Note With newly added IoT capabilities, customers can now also reduce attack surface for their IoT devices integrated with the Azure IoT platform) and remediate issues before they can be exploited.

In addition to providing full visibility into the security posture of your environment, Azure Security Center also provides visibility into the compliance state of your Azure environment against common regulatory standards.

2. **Cloud workload protection**. Azure Security Center's threat protection enables customer to detect and prevent threats at the IaaS layer (as well as in platform-as-a-service (PaaS) resources like Azure IoT Hub) and on-premises servers and VMs. Key features of Azure Security Center threat protection include config monitoring, server endpoint detection and response (EDR), application control, network segmentation, and extends to support container and serverless workloads for Cloud-native applications.

   Azure Security Center helps protect Linux servers with behavioral analytics. For every attack attempted or carried out, you receive a detailed report and recommendations for remediation.

   In terms of advanced controls, one can mention the "just-in-time" (JIT) virtual machine (VM) access that can be used to lock down inbound traffic to Azure VMs, thus reducing your surface area exposed to attack - SSH brute-force attack is one of the most common threats with more than 100,000 attack attempts on Azure VMs per month -, while providing easy access to connect to VMs when needed. When JIT is enabled, Azure Security Center locks down inbound traffic to your Azure VMs by creating a Network Security Group (NSG) rule. You select the ports, for example 22 (SSH), on the VM to which inbound traffic will be locked down. These ports are controlled by the JIT solution.

   As you add applications to VMs in Azure, you can block malicious apps, including those not mitigated by antimalware solutions, by using adaptive application controls. Machine learning automatically applies new application whitelisting policies across your VMs.

Continuous assessment helps you discover potential security issues, such as systems with missing security updates or exposed network ports. When Azure Security Center identifies potential security vulnerabilities, it creates recommendations that guide you through the process of configuring the needed controls. A list of prioritized findings and recommendations can trigger alerts or other guided remediation.

In addition, Azure Security Center also supports integration with third-party solutions and can be customized with automation and programming capabilities.

Eventually, you can leverage the various solutions available from the Azure Marketplace. You can use antimalware software from security vendors such as Microsoft, Symantec, Trend Micro, and Kaspersky to protect your VMs from malicious files, adware, and other threats.

Resource:

- What is Azure Security Center?.
- Azure Virtual Machines security overview.
- Security best practices for IaaS workloads in Azure.
- Security Recommendations for Azure Marketplace Images.
- Security controls for Linux Virtual Machines.
- How Azure Security Center helps detect attacks against your Linux machines.
- Leverage Azure Security Center to detect when compromised Linux machines attack.
- Security controls for Azure Virtual Machine Scale Sets.
- Protecting your machines and applications in Azure Security Center.
- Understand Azure Security Center container recommendations.

SIEM

As introduced above, Azure Security Center provides unified security management by identifying and fixing misconfigurations and providing visibility into threats to quickly remediate them. Security Center has grown rapidly in usage and capabilities including a security information and event management (SIEM)-like functionality called investigations.

When it comes to cloud workload protection, the goal is to present the information to users within Security Center in an easy-to-consume manner so that you can address individual threats. Azure Security Center is not intended for advanced security operations (SecOps) hunting scenarios or to be a SIEM tool.

SIEM and security orchestration and automated response (SOAR) capabilities are delivered in Azure Sentinel. Azure Security Center and Azure Sentinel are different capabilities with complementary purposes.

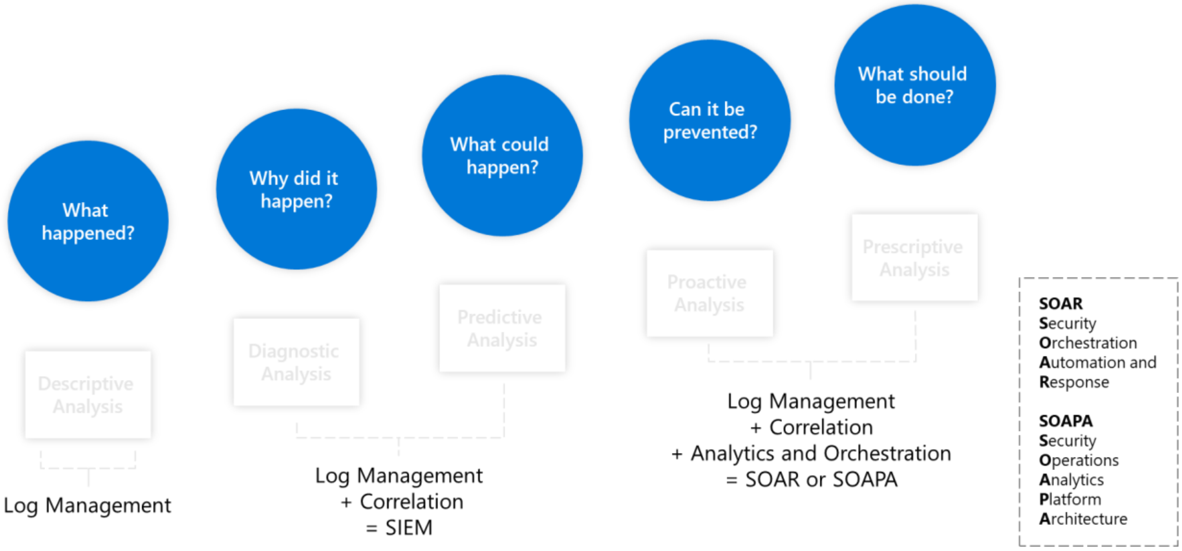

What happened?
Why did it happen?
What could happen?
Can it be prevented?
What should be done?
Descriptive Analysis
Diagnostic Analysis
Predictive Analysis
Proactive Analysis
Prescriptive Analysis
Log Management
Log Management + Correlation = SIEM
Log Management + Correlation + Analytics and Orchestration = SOAR or SOAPA
SOAR Security Orchestration Automation and Response
SOAPA Security Operations Analytics Platform Architecture

Figure 16 How SecOps has evolved

Azure Sentinel delivers intelligent security analytics and threat intelligence across the organization, providing a single solution for alert detection, threat visibility, proactive hunting, and threat response.

Azure Sentinel is your service operations center (SOC) view across the organization, alleviating the stress of increasingly sophisticated attacks, increasing volumes of alerts, and long resolution timeframes. With Azure Sentinel, you can:

- **Collect data at cloud scale** across all users, devices, applications, and infrastructure, both on-premises and in multiple clouds.
- **Integrate curated alerts** from Microsoft's security products like Azure Security Center, and from your non-Microsoft security solutions.
- **Detect previously undetected threats** and minimize false positives using Microsoft Intelligent Security Graph, which uses trillions of signals from Microsoft services and systems around the globe to identify new and evolving threats. Investigate threats with artificial intelligence and hunt for suspicious activities at scale, tapping into years of cyber security experience at Microsoft, see section § "Microsoft Cyber Defense Operations Center" in the Appendix.
- **Respond to incidents rapidly** with built-in orchestration and automation of common tasks.

SIEMs typically integrate with a broad range of applications including threat intelligence applications for specific workloads, and the same is true for Azure Sentinel. SecOps has the full power of querying against the raw data, using Machine Learning models, even building your own model.

As such, Azure Security Center is one of the many sources of threat protection information that Azure Sentinel collects data from, to create a view for the entire organization. Microsoft recommends that customers using Azure use Azure Security Center for threat protection of workloads such as VMs, containers, storage, and IoT as already covered.

In just a few clicks, you can connect Azure Security Center to Azure Sentinel. Once the Security Center data is in Azure Sentinel, you can combine that data with other sources like firewalls, users, and devices, for proactive hunting and threat mitigation with advanced querying and the power of Machine Learning.

Important note To reduce confusion and simplify the user experience, two of the early SIEM-like features in Security Center, namely above mentioned investigation flow in security alerts and custom alerts will be removed in the near future. Individual alerts remain in Security center, and there are equivalents for both security alerts and custom alerts in Azure Sentinel.

Azure Sentinel is designed to simplify the application of advanced technologies like Machine Learning, User and Entity Behavior Analytics (UEBA), to the variety of datasets you monitor and is complemented by other Microsoft Threat Protection solutions that provide specialized investigation of hosts, email, identity attacks, and more.

Resource:

- What is Azure Sentinel?.

Deploying your Azure services

The method used to deploy the modernized application should be reviewed to see if it is still relevant, applicable, etc. depending on the selected modernization route(s) and understand the order of magnitude of what could be reused or adapted from the existing scripts or procedures.

This said, the deployment solutions that are available on the cloud platform should be considered to choose the best deployment option for the application, and related services and configuration.

How can Azure deploy your services?

Azure has an option for every type of organization, including those who need Azure to be in their own datacenter. You can deploy your applications either in the public Azure cloud or on-premises in Azure Stack choose how portable your applications should be.

It's also possible to develop apps in containers to deploy them in containers to deploy them on-premises or in another cloud, or by using Azure Resource Manager templates to script your complete Infrastructure as Code (IaC).

Let's explore these options in more detail.

Are you saying "Infrastructure as Code"?

Infrastructure as Code (IaC) offers the possibility to procure virtual resources through the use of desired-state configuration files. Those configuration files become the infrastructure documentation and can be managed as source code.

In other words, IaC captures environment definitions as declarative code, such as JSON documents, for automated provisioning and configuration. All Azure services are based on Azure Resource Manager (ARM), which you can use to document your environment as IaC thanks to Azure Resource Manager templates. These templates are JSON files that describe what you want to deploy and what the parameters are.

Note For information on Azure Resource Manager (ARM), please see the article Azure Resource Manager.

Note For a primer on ARM template, see article Understand the structure and syntax of Azure Resource Manager templates and whitepaper Getting started with Azure Resource Manager. More in-depth information can be found in the whitepaper World Class ARM Templates Considerations and Proven Practices.

It's easy to create Azure Resource Manager templates in Visual Studio Code, i.e. an open source integrated development environment (IDE) and available on Linux, using Azure Resource Group project templates. You can also generate Azure Resource Manager templates from the Azure portal by clicking the Automation Script button, which is available on the menu bar of every resource in the Azure portal. This creates the Azure Resource Manager template for the given resource and even generates code for building the resource using the Azure CLI, PowerShell, .NET, and others.

After you have an Azure Resource Manager template, you can deploy it to Azure by using PowerShell, the Azure CLI, or Visual Studio Code. Or you can automate its integration, delivery and/or deployment in adequate DevOps pipelines (see section § "Implementing (secure) DevOps practices for your applications" below). A great example of deploying resources to the cloud using Azure Resource Manager is the Deploy to Azure button found in many GitHub repositories.

In addition to using Resource Manager for IaC, you can bring your existing skills and tools such as Ansible, Chef, Puppet, Packer, and Terraform to provision and manage Azure infrastructure directly.

As far as the latter is concerned, the Terraform provider for Azure Resource Manager can be used to configure infrastructure in Microsoft Azure using the Azure Resource Manager API's.

HashiCorp Terraform is a great tool for doing declarative deployment to Azure. We're seeing great momentum with adoption of HashiCorp Terraform on Azure as the number of customers has doubled since the beginning of the year - customers are using Terraform to automate Azure infrastructure deployment and operation in a variety of scenarios.

> **Note** The momentum is fantastic on the contribution front as well with nearly 180 unique contributors to the Terraform provider for Azure Resource Manager. The involvement from the community with our increased 3-week cadence of releases (ensures more coverage of Azure services by Terraform.
>
> Additionally, after customer and community feedback regarding the need for additional Terraform modules for Azure, Microsoft has been working hard at adding high quality modules and now have doubled the number of Azure modules in the terraform registry, bringing it to over 120 modules. We believe all these additional integrations enable you to manage infrastructure as code more easily and simplify managing your cloud environments.

Microsoft and HashiCorp are working together to provide integrated support for Terraform on Azure. Customers using Terraform on Microsoft's Azure cloud are mutual customers, and both companies are united to provide troubleshooting and support services. This joint entitlement process provides collaborative support across companies and platforms while delivering a seamless customer experience. Customers using Terraform Provider for Azure can file support tickets to Microsoft support. Customers using Terraform on Azure support can file support tickets to Microsoft or HashiCorp.

> **Note** HashiCorp and Microsoft are longstanding partners in the cloud infrastructure community. In 2017, Microsoft committed to a multi-year partnership aimed at further integrating Azure services with HashiCorp products. As a result of this collaboration, organizations can rely on tools like Terraform to create and manage Azure infrastructure.
>
> For more information on HashiCorp Vault and Azure integrations, see page Hashicorp/Azure Integrations.

Resources:

- Azure Resource Manager overview.
- Azure Resource Manager templates.
- Azure Quickstart Templates.
- Use infrastructure automation tools with virtual machines in Azure.
- Terraform with Azure.

Leveraging Azure Blueprints

It's easy to use Azure Resource Manager templates, user identities, and access rights and policies (see section § "Identity and access management" above) to design and create a complete infrastructure. But how do you keep all of these things together? And how do you keep track of which environments each piece of infrastructure has been deployed to and which version of the artifact is deployed now?

You can organize all your infrastructure artifacts with Azure Blueprints. Azure Blueprints provides a mechanism that allows you to create and update artifacts, assign them to compliant environments, and define versions. You can store and manage these artifacts as well as manage their versions and relate them to environments.

As such, Azure Blueprints provide templates for quick, repeatable creation of fully governed cloud subscriptions. The azure-blueprints repo on GitHub provides a library of sample Blueprints that can be easily imported via API or PowerShell.

Resource:

- Overview of the Azure Blueprints service.

Doing automation

The idea behind automation is to transform any error prone, repetitive tasks into automation workflows or series of steps. Automatiob addresses the need for the business to implement changes faster and with greater reliability. As such, it serves the need of event management, capacity management, availability management, etc.

Azure Automation delivers a cloud-based automation and configuration service that provides consistent management across your Azure and non-Azure environments. It consists of process automation, update management, and configuration features. Azure Automation provides complete control during deployment, operations, and decommissioning of workloads and resources.

Azure Automation provides you with the ability to automate frequent, time-consuming, and error-prone cloud management tasks. This automation helps you focus on work that adds business value. By reducing errors and boosting efficiency, it also helps to lower your operational costs.

You can integrate Azure services and other public systems that are required in deploying, configuring, and managing your end to end processes. The service allows you to author runbooks graphically, in PowerShell, or Python. By using a hybrid Runbook worker, you can unify management by orchestrating across on-premises environments. Webhooks provide a way to fulfill requests and ensure continuous delivery and operations by triggering automation from ITSM, DevOps, and monitoring systems.

Azure Automation has the ability to integrate with source control which promotes configuration as code where runbooks or configurations can be checked into a source control system.

Resource:

- An introduction to Azure Automation.

Implementing (secure) DevOps practices for your applications

Cutting horizontally across migration, modernization and Cloud-native is DevOps. Indeed, to take full advantage of cloud benefits, a DevOps approach of **Continuous** Delivery (CD) is essential – after all, what sense does an agile, open, and flexible cloud platform like Microsoft Azure make if you still need weeks or months to get changes or bug fixes out there.

The ability to be continually deployed and updated without disrupting the user experience (UX) is at the core of a digitally transformed organization. Those that stick to legacy development models will find that the market, and their customers' expectations, have changed by the time they deploy their application, moving them and their product further behind the competition. The end result is a lightning-fast ability to deliver new business capabilities at rock bottom costs.

DevOps is the union of people, process, and tools to enable Continuous Integration (CI) and Continuous Delivery (CD)/Continuous Deployment (CD) of value to end users. The contraction of *Dev* and *Ops* refers to replacing silo'ed Development and Operations disciplines with multidisciplinary teams that work together with shared and efficient practices and tools. A converged DevOps cycle provides the ability to execute on ideas quickly and iterate on feedback rapidly while at the same time maintaining highest levels of quality.

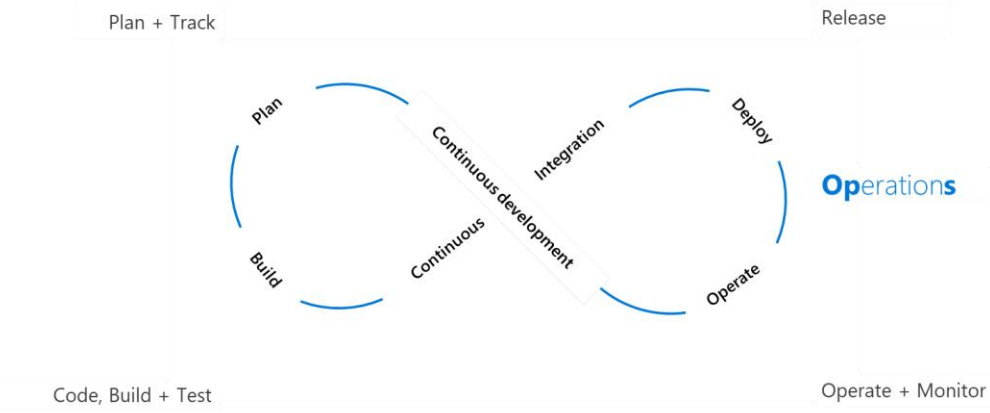


Figure 17 DevOps cycle

Towards (Sec)DevOps practices

For Microsoft, DevOps encompasses the whole application lifecycle, from idea (of the modernization in the context of this document) to application running in production. SecDevOps (a.k.a. DevSecOps and DevOpsSec) is the process of integrating secure development best practices and methodologies into development and deployment processes which DevOps makes possible like the well-defined ones of the Microsoft Security Development Lifecycle (SDL) that are technology agnostic. See section § "An overview of Microsoft SDL" in the Appendix.

Beyond the code for various microservices/components of the workload, the objective of IaC (see previous section) aims at also checking in the above templates for the workload in its target (private) (Git) repo as part of a CI pipeline and thus, ensuring that, for that purpose, it fulfills a series of tests notably in terms of security checks.

For example, such a repo can be typically instantiated in GitHub or Azure Repos.

Note As far as Azure Repos is concerned, the open source Secure DevOps Kit for Azure (AzSK) available on GitHub provides a series of security verification test for the code along with security controls as scripts that can be easily implemented to raise the bar.

This encompasses all the templates for the deployment of the Azure services on top of which these various microservices/components run/are executed or on which these microservices/components rely on. This may also comprise all the configuration files for the above microservices/components if one follows the so-called Twelve Factors App approach (Configuration-as-Code): Store config in the environment.

How can Azure help implement (Sec)DevOps for your applications?

(Sec)DevOps is built into the foundation of Azure, not bolted-on like with other cloud vendors, and Azure DevOps lets you plan smarter, collaborate better, and ship faster with a set of modern developer solutions that can help automate your builds and deployments and automatically test your code and apps before launch.

To help you build, deploy, test, and track your code and applications, Azure DevOps includes:

- Azure Boards to plan, track, and discuss work across teams.
- Azure Repos to collaborate on code development with free Git public and private repositories, pull requests, and code review.
- Azure Pipelines to create Build and Release pipelines that automate CI/CD.
- Azure Test Plans to improve your overall code quality with manual and exploratory testing services for your apps.
- Azure Artifacts to share code packages (like npm packages) across your organization.

With a number of extensions built by the community and available in the Azure Marketplace.

You can use all the above DevOps solutions or choose just what you need to complement your existing workflows.

For example, Azure Pipelines enable you to build, test, and deploy in any language, to any cloud - or event on-premises. Pipelines run in parallel on Linux, and allow to deploy containers to individual hosts or Kubernetes (see section § ʻʻ Leveraging containerization” above).

If the *Build* operations run without any error (and noticeable warning), the code as whole is fully integrated at this stage. It must then be fully qualified in a Quality Assurance (QA) (a.k.a. preproduction) environment that is most akin to production.

A (Sec)DevOps code fabric may automatically instantiate the deployment to the QA environment for a Continuous Deployment (CD) or may require a prior validation(s) depending on the workflow in place along with the policies to enforce as part of a Continuous Delivery (CD) effort.

Release pipelines and related operations are setup for that purpose.

Note One must ensure that the workload fulfill all the requirements and requirements not only from deployment and functional standpoints but also from the operational one in terms of security, privacy, monitoring, performances, stability, manageability, etc. For example, it will be necessary to check that the response times between distributed parts/components/services are correct and respect the expected level of performances. Possible integration problems must be resolved even if minor modifications are required.

In this context, a preproduction environment is indeed intended to best reflects the production environment in which the application will ultimately be deployed. A public cloud platform like Azure greatly helps in the ability to reproduce environment(s) with a configuration that will identically reproduce the target environment(s).

If this quality gate is passed, the workload can be pushed to the production environment(s).

Note The Tutorial: Deploy apps to Azure and Azure Stack illustrates how to deploy an application to Azure and Azure Stack using a hybrid continuous integration/continuous delivery (CI/CD) pipelines.

Resource:

- What is Azure DevOps?.
- What is Azure Boards?.
- What is Azure Repos?.
- What is Azure Pipelines?.
- What is Azure Test Plans?.
- What is Azure Artifacts?.

Monitoring your applications

How can Azure help monitor your applications?

Adequatly monitoring your applications supposes to ensure that you can collect and aggregate logs and metrics from your applications, the services' instances they used and the platform, in order to get sufficient data for event management, performance reporting, capacity management and availability management.

Azure Monitor helps you understand how your applications are performing and proactively identifies issues affecting them and the resources they depend on, thus allowing you to maximize both the availability and performance of your applications

Understanding how your applications are performing

As such, Azure Monitor delivers a comprehensive solution for collecting, analyzing, and acting on telemetry from your cloud and on-premises environments. Azure Monitor also includes several features and tools that provide valuable insights into your applications and other resources that they depend on. (Azure Monitor integrates the capabilities of Log Analytics and Application Insights that were previously branded as standalone services in Microsoft Azure.)

Azure Monitor collects data from each of the following tiers:

- **Application monitoring data**. Data about the performance and functionality of the code you have written, regardless of its platform.
- **Guest OS monitoring data**. Data about the operating system on which customer application is running. The application could be running in Azure, another cloud, or on-premises.
- **Azure resource monitoring data**. Data about the operation of an Azure resource.
- **Azure subscription monitoring data**. Data about the operation and management of an Azure subscription, as well as data about the health and operation of Azure itself.
- **Azure tenant monitoring data**. Data about the operation of tenant-level Azure services, such as Azure AD (see section § "**Error! Reference source not found.**" prior in this document).

All data collected by Azure Monitor fits into one of two fundamental types, metrics and logs:

- Metrics are numerical values that describe some aspect of a system at a particular point in time. They are lightweight and capable of supporting near real-time scenarios.
- Logs contain different kinds of data organized into records with different sets of properties for each type. Telemetry such as events and traces are stored as logs in addition to performance data so that it can all be combined for analysis.

Azure Monitor enables monitoring for workloads and Azure services by collecting metrics, activity logs, and diagnostic logs.

The metrics collected provide performance statistics for different resources, i.e. how a resource is performing and the resources that it's consuming.

The activity log will show you when new resources are created or modified. You can view this data with one of the explorers in the Azure portal and send it to Log Analytics for trending and detailed analysis, or you can create alert rules that will proactively notify you of critical issues.

You can further extend the data you're collecting into the actual operation of the resources by enabling diagnostics and adding an agent to compute resources. This will collect telemetry for the internal operation of the resource and allow you to configure different data sources to collect logs and metrics from Linux and Windows guest operating system (OS).

Moreover, you can enable monitoring for your VM and virtual machine scale set application, Kubernetes-based application, or App Services application to enable Application Insights to collect detailed information about your application including page views, application requests, and exceptions.

Considering the above, just a few examples of what you can do with Azure Monitor include:

- Detect and diagnose issues across applications and dependencies with Application Insights.
- Correlate infrastructure issues with Azure Monitor for VMs and Azure Monitor for Containers.
- Drill into your monitoring data with Log Analytics for troubleshooting and deep diagnostics. (Azure Log Analytics provides real-time insights by using integrated search and custom dashboards to analyze millions of records across all your workloads.)
- Support operations at scale with smart alerts and automated actions.
- Create visualizations with Azure dashboards and workbooks.

With Azure Monitor, you can get a 360-degree view of your applications, infrastructure, and network with advanced analytics, dashboards, and visualization maps. Azure Monitor provides intelligent insights and enables better decisions with AI.

You can analyze, correlate, and monitor data from various sources using a powerful query language and built-in machine learning constructs. Moreover, Azure Monitor provides out-of-the-box integration with popular DevOps, IT Service Management (ITSM), and Security Information and Event Management (SIEM) tools like Azure Sentinel (see below).

Resources:

- What is Azure Monitor?.
- What is Application Insights in Azure Monitor?.
- What is Log Analytics in Azure Monitor?.
- What is Azure Monitor for VMs (preview)?.
- What is Azure Monitor for containers?.
- What is Azure Monitor for storage?.

Managing your logs

As covered above, Azure Monitor (Log Analytics) maximizes the availability and performance of you applications and services by delivering a comprehensive solution for collecting, analyzing, and acting on telemetry from your cloud and on-premises environments. It helps you understand how your applications are performing and proactively identifies issues affecting them and the resources they depend on.

Log data collected by Azure Monitor can be analyzed with queries to quickly retrieve, consolidate, and analyze collected data. As part of Azure Monitor, Log Analytics allow to drill into your log data for troubleshooting and deep diagnostics. You can create and test queries using Log Analytics in the Azure portal and then either directly analyze the data using these tools or save queries for use with visualizations or alert rules.

Azure Monitor uses a version of the Kusto query language used by Azure Data Explorer that is suitable for simple log queries but also includes advanced functionality such as aggregations, joins, and smart analytics. You can quickly learn the query language using multiple lessons. Particular guidance is provided to users who are already familiar with SQL and Splunk.

In addition, the Azure Activity Log is a log that provides insight into the operations that were performed on resources in a customer's subscription. The Activity Log reports control-plane events for customer's subscriptions. Using the Azure Activity Log, you can determine the 'what, who, and when' for any write operations (PUT, POST, DELETE) taken on the resources in your subscription(s). You can also understand the status of the operation and other relevant properties.

Resources:

- Overview of log queries in Azure Monitor.
- Overview of Azure Activity log.

Keeping your applications up and running

Azure helps you to avoid potential disasters and quickly recover if your organization does get hit by disaster. Everything in Azure is built on top of a resilient foundation, which is a key requirement for any application to achieve resiliency. Such a resiliency foundation depends on:

- At the design level, on how Microsoft Azure designs its global fiber network, evolving datacenters, and storage protections built into the platform.
- At the operation level, on how Microsoft rolls out releases into the environment, performs maintenance (planned and unplanned), and uses Machine Learning to predict failures and protect you applications.

When your applications and data are hosted on a cloud service in Azure, you transparently benefit from redundant, distributed implementations of your applications' resources across physical locations (see below).

However, even in the cloud, hardware can fail, the network can have transient failures, and rarely, an entire service or region may experience a disruption. Thus, your CSP's dedication to business continuity (BC) is vital.

As such, Azure provides a comprehensive set of native business continuity solutions that protect you against failures within datacenters and even the failure of entire datacenters. Business continuity (BC) is based on the ability to perform essential business functions during and after adverse conditions, such as a natural disaster or a downed service.

Azure is the first hyperscale cloud provider to be certified under ISO/IEC 22301:2012 "Societal security — Business continuity management systems — Requirements", the first international standard to demonstrate the ability to prevent, mitigate, respond to, and recover from disruptive incidents.

How can you observe the health status of Azure?

Considering the above, you should be in a position to observe what's happening in your environment(s) in Azure, inform people and systems to make informed decisions before/during issues, and also determine your own availability requirements (see next sections).

We'd like to emphasize here that transparent communication with customers in the event of a service incident is critical to the Microsoft support model and related activities.

In the event of such a service incident, Microsoft aspires to:

- Be transparent with clients,
- Provide coherent communications,
- Bring a quick response.

In this dynamic, Microsoft provides a set of tools and dashboards of service integrity status for viewing, monitoring, and notification of the status of the services Microsoft Azure and whether their availability is affected by a known service incident already being processed by Microsoft teams.

You have a platform availability alert communication system, and can also refer to the Azure Status dashboard. This dashboard indicates the availability of each of the Azure services by region for the different capacities according to the region considered in Europe (and elsewhere).

It is also possible to follow the updates related to a service incident on the blog of the Product group as well as on social networks and in particular on:

- Twitter: https://twitter.com/Azure
- Facebook: https://www.facebook.com/microsoftazure

While the above dashboard is a good reference for incidents with widespread impact, but we strongly recommend that current Azure users leverage Azure Service Health to stay informed about Azure incidents and maintenance.

Azure Service Health indeed gives you a personalized view of the health of your Azure services for your workloads. Azure Service Health notifies you about Azure service incidents and planned maintenance so that you can take action to mitigate downtime. You can configure customizable cloud alerts and use your personalized dashboard to analyze health issues, monitor the impact to your cloud resources, get guidance and support, and share details and updates.

In addition, Azure Resource Health provides information about the health of your individual cloud resources such as a specific VM instance. Using Azure Monitor, you can also configure alerts to notify you of availability changes to your cloud resources (see section § "Log management" above). Azure Resource Health along with Azure Monitor notifications will help you stay better informed about the availability of your resources minute by minute and quickly assess whether an issue is due to a problem on your side or related to an Azure platform event.

Resource:
- What is Azure Service Health?.

What can Azure do for high-availability?

A key aspect of a resilient foundation is availability. Generally speaking, high availability is all about maintaining acceptable continuous performance despite temporary failures in services, hardware, or datacenters, or fluctuations in load.

Highly available systems are consistently operational over long periods of time. Azure uptime, expressed as a rolling 12 month average to June 2019, was 99.996%, or approximately 26 minutes of downtime per year. Availability can never be 100% because hardware and software failures happen, and human error occurs. But the Service Level Agreement (SLA) describes our commitment for uptime and connectivity.

Microsoft maintains its commitment to providing customers with high-quality products and services by adhering to comprehensive operational policies, standards, and practices. Microsoft provides formal SLAs for each individual Azure products and services.

SLAs describe Microsoft's commitment to providing customers with specific performance standards and the guaranteed availability levels. SLAs also specify what happens if a service or product fails to perform to a governing SLA's specification.

Resources:
- Service Level Agreements to learn about our uptime guarantees and downtime credit policies.
- SLA summary for Azure services.

This said, and aside the above SLA's, to achieve resilience, your applications on top of the above resiliency foundation have also to take advantage of the resilient services built on/provided by the foundation as per the shared responsibilities that applies in the public cloud, see section § "Understanding the shared responsibilities' model for your applications" above.

Azure provides in this space support for high availability at the virtual machine, datacenter, and regional levels, through a number of features and functions across the categories of compute, networking, and storage from which your applications can directly benefit. This is the purpose of the next sections to highlight some of them.

Azure compute solutions

Customers can group VMs together to provide high availability, scalability, and redundancy. Azure has several features so that no matter what uptime requirements are, Azure can meet them. These features include availability sets and virtual machine scale sets.

Azure Availability Sets are logical groupings of two or more VMs that help keep an application available during planned or unplanned maintenance.

With an availability set, you get:

- Up to three fault domains that each have a server rack with dedicated power and network resources.
- Five logical update domains which then can be increased to a maximum of 20.

VMs are then sequentially placed across the fault and update domains.

You can also leveraged Azure VM scale sets. See section § "Azure Virtual Machine Scale Set" above.

Resources:

- Manage the availability of Linux virtual machines.
- Create and deploy highly available virtual machines with the Azure CLI.

Azure networking solutions

As suggested above, the Microsoft Azure hyperscale public cloud provides resiliency in time of natural disaster and warfare. The Azure public cloud provides capacity for failover redundancy and empowers you with flexibility regarding global resiliency planning.

Each Azure region (see section § "A globally available infrastructure for your applications" above) is always paired with another region within the same geography at least 300 miles away. This approach allows for the replication of resources (such as virtual machine storage) across a geography that helps reduce the likelihood of interruptions due to events such as natural disasters, civil unrest, power outages, or physical network outages affecting both regions at once.

Since the pair of regions is directly connected and far enough apart to be isolated from regional disasters, you can use them to provide reliable services and data redundancy. Some services offer automatic geo-redundant storage using region pairs.

Additional advantages of region pairs include:

- If there's an extensive Azure outage, one region out of every pair is prioritized to help reduce the time it takes to restore them for applications.
- Planned Azure updates are rolled out to paired regions one region at a time to minimize downtime and risk of application outage.
- Data continues to reside within the same geography as its pair (except for Brazil South) for tax and law enforcement jurisdiction purposes.

Moreover, an increasing number of Azure regions support availability zones. Azure Availability zones are physically separate datacenters within an Azure region.

Each availability zone is made up of one or more datacenters equipped with independent power, cooling, and networking. It is set up to be an isolation boundary. If one zone goes down, the other continues working. Availability zones are connected through high-speed, private fiber-optic networks.

As of this writing, following regions have a minimum of three separate zones to ensure resiliency and support quorum-based workloads.

- Central US,
- East US,
- East US 2,
- West US 2,
- France Central,
- North Europe,
- UK South,
- West Europe,
- Japan East,
- Southeast Asia.

Having a broadly distributed set of datacenters allows Azure to provide a high guarantee of availability. For example, considering the above, a paired region and availability zones within the same data residency boundary provides high availability, backup (see eponym section below), and disaster recovery (see eponym section below).

In addition, to connect to your on-premises resources if needed (see section § "Connecting your virtual networks to your on-premises resources" above), you can deploy ExpressRoute and VPN gateways in Azure Availability Zones. This brings resiliency, scalability, and higher availability to virtual network gateways. Deploying gateways in Azure Availability Zones physically and logically separates gateways within a region, while protecting your on-premises network connectivity to Azure from zone-level failures.

Moreover, Azure Traffic Manager allows to route incoming traffic for high performance and availability. It scales across Azure regions, helping to reduce latency and provide users a performant experience, regardless of where they are. It operates at the DNS layer to quickly and efficiently direct incoming DNS requests based on the routing method of choice.

An example would be sending requests to the closest endpoints, improving the responsiveness of applications. As such it offers four types of DNS-based traffic routing: Failover, performance, geographic, and weighted round-robin. You can choose the one that's right for them or combine, using nested profiles.

Azure Traffic Manager is an intelligent routing mechanism that you put in front of your Web Apps or portals applications. Web Apps acts as endpoints, which Azure Traffic Manager monitors for health and performance.

In addition, Azure Load Balancer can scale your applications and create high availability for your services. Azure Load Balancer automatically scales with increasing application traffic. Azure Load Balancer supports inbound and outbound scenarios, provides low latency and high throughput, and scales up to millions of flows for all Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) applications.

You can use Azure Load Balancer with incoming internet traffic, internal traffic across Azure services, port forwarding for specific traffic, or outbound connectivity for VMs in your VNETs.

You may also consider for your applications a content delivery network (CDN), i.e. a distributed network of servers that can efficiently deliver web content to users. It is a way to get content to users in their local region to minimize latency. CDN can be hosted in Azure or any other location.

With Azure Content Delivery Network, you can cache content at strategically placed physical nodes across the world and provide better performance to end users. Typical usage scenarios include web applications containing multimedia content, a product launch event in a particular region, or any event where you expect a high bandwidth requirement in a region.

Resources:

- What are Availability Zones in Azure?.
- About zone-redundant virtual network gateways in Azure Availability Zones.
- What is Azure Load Balancer?.
- What is Traffic Manager?.
- Azure Load Balancing Solutions: A guide to help you choose the correct option.
- What is a content delivery network on Azure?.

Azure storage options

Customer data in an Azure Storage account is always replicated to help ensure durability and high availability.

Azure Storage (see section § "Leveraging storage virtualization capabilities" above) copies customer data to protect it from transient hardware failures, network or power outages, and even massive natural disasters. You can choose to replicate your data within the same data center, across availability zones within the same region, or across geographically separated regions.

Specifically, when creating a storage account, you can select one of the following redundancy options:

- Locally redundant storage (LRS) that replicates your data within a storage scale unit that is hosted in a datacenter in the region in which you created your storage account.
- Zone-redundant storage (ZRS) that replicates your data synchronously across three storage clusters in a single region, where each storage cluster is physically separated from the others and resides in its own availability zone (see above section).
- Geo-redundant storage (GRS) that replicates your data to a secondary region that is hundreds of miles away from the primary region (see above section)
- Read-access geo-redundant storage (RA-GRS) that provides read-only access to the data in the secondary location, in addition to geo-replication across two regions.

Azure Storage redundancy options can have implications on data residency as Azure relies on paired regions to deliver GRS (and RA-GRS). For example, customers concerned about geo-replication across regions that span country boundaries, may want to choose LRS or ZRS to keep Azure Storage data at rest within the geographic boundaries of the country in which the primary region is located.

Resource:

- Introduction to Azure Storage.

Implementing a business continuity and disaster recovery strategy for your applications

Azure was the first cloud platform to provide a built-in backup and disaster recovery solution.

Backup management

Data backup is a critical part of disaster recovery. If the stateless components of an application fail, you can always redeploy them. If data is lost, the system can't return to a stable state. Data must be backed up, ideally in a different region in case of a region-wide disaster. Backup management comprises all the operations related to executing, securing and restoring backups.

As far as backups are concerned from a resilient foundation standpoint, content is replicated from a primary datacenter to a secondary datacenter within a geography. (To prevent a disaster from affecting the second data center, the distance from the primary data center is at least 250 miles. Traffic between datacenters is encrypted.)

As such, there is not a specific backup schedule as the replication is constant. You can choose to perform your own extractions/backups if necessary. Customer data is stored in a redundant environment with robust backup, restore, and failover capabilities to enable availability, business continuity, and rapid recovery. Multiple levels of data redundancy are implemented, ranging from redundant disks to guard against local disk failure to continuous, full data replication to a geographically dispersed datacenter. Microsoft Azure undergoes an annual validation of backup/recovery practices.

> **Note** "Information back-up" is covered under the ISO/IEC 27001:2013 standard, specifically addressed in Annex A, domain 10.5.1. For more information, review of the publicly available ISO standards we are certified against is suggested.

Data protection services (DPS) backs up data for properties based on a defined schedule and upon request of the properties. Data is retained according to data type identified by property. DPS investigates backup errors and skipped files and follows up appropriately. See SOC 2 A1.2 Environmental protections, software, data backup processes.

Beyond these capabilities, Azure Backup is the Azure-based service you can use to back up and restore your data in Azure. They can back up on-premises machines and workloads, and Azure Virtual Machines (VMs). Azure Backup replaces existing on-premises or off-site backup solutions with a cloud-based solution that is reliable, secure, and cost-competitive to simplify data protection from ransomware and human error.

Azure Backup offers multiple components that are downloaded and deployed on the appropriate computer, server, or in the cloud. All Azure Backup components (no matter whether data to be protected are located on-premises or in the cloud) can be used to back up data to a Backup vault in Azure.

Customers have the responsibility to manage backups of their data. As already introduced, Azure Role-Based Access Control (RBAC) enables fine-grained access management for Azure. Using RBAC, you can segregate duties within your team and grant only the amount of access to users that they need to perform their jobs. Azure Backup provides 3 built-in roles to control backup management operations: Backup Contributor, Backup Operator and Backup Reader.

Every activity is logged with the information about operations taken on the resources in the subscription, who initiated the operation, when the operation occurred and what was the status of the operation. Information can be retrieved the activity logs through the portal, PowerShell, Azure CLI, or Insights REST API.

> **Resource**:
> - What is the Azure Backup service?.

Disaster recovery

Aside Azure Backup, Azure Site Recovery also contributes to a business continuity and disaster recovery (BCDR) strategy.

BCDR consists of two broad aims:

1. Keep business data safe and recoverable when outages occur.
2. Keep apps and workloads up and running during planned and unplanned downtimes.

Azure Site Recovery is a native disaster recovery as a service (DRaaS), and **Microsoft been recognized as a leader in DRaaS based on completeness of vision and ability to execute by Gartner[3] in the 2019 Magic Quadrant for Disaster Recovery as a Service**.

While Azure Backup keeps customer's data safe and recoverable by backing it up to Azure, Azure Site Recovery helps ensuring business continuity by keeping workload running available if the primary site goes down. Site Recovery replicates workloads running on physical and virtual machines (VMs) so that they remain available in a secondary location if the primary site isn't available. It recovers workloads to the primary site when it's up and running again.

To help you comply with national, regional, and industry-specific requirements governing the collection and use of individuals' data, Microsoft offers the most comprehensive set of certifications and attestations of any cloud service provider (see section § "Compliance" above).

Resources:

- What is the Azure Site Recovery service?.
- Business continuity and disaster recovery (BCDR): Azure Paired Regions.

This concludes this whitepaper. We hope you enjoyed the tour!

[3] Gartner does not endorse any vendor, product or service depicted in its research publications, and does not advise technology users to select only those vendors with the highest ratings or other designation. Gartner research publications consist of the opinions of Gartner's research organization and should not be construed as statements of fact. Gartner disclaims all warranties, expressed or implied, with respect to this research, including any warranties of merchantability or fitness for a particular purpose.

Appendix

Microsoft Security standards and related practices

Microsoft Operational Security Assurance (OSA) consists of a set of practices that aim to improve operational security in cloud-based infrastructure. As such, OSA is a framework that incorporates the knowledge gained through a variety of capabilities that are unique to Microsoft, including the Microsoft Security Development Lifecycle (SDL), the Microsoft Security Resource Center (MSRC) program, and deep awareness of the cybersecurity threat landscape by the Microsoft cybersecurity teams. OSA combines this knowledge with the experience of running hundreds of thousands of servers in 120+ data centers around the world.

Note For more information, see page What are the Microsoft OSA practices?.

Security represents $1B+ annual investments over 3,500 security experts.

Secure development policy

Microsoft manages the information system for Microsoft Azure using a system development lifecycle that incorporates information security considerations. Microsoft's implementation of lifecycle support is outlined through the so-called Microsoft Security Development Lifecycle (SDL) process to which adhere all engineering and development projects.

Microsoft SDL consists of a set of practices that support security assurance and compliance requirements. The SDL helps developers build more secure software by reducing the number and severity of vulnerabilities in software, while reducing development cost.

Note For more information, see page What are the Microsoft SDL practices?.

Microsoft SDL is at the core of Microsoft's defense in depth strategy that is fully align the principles of SD3: Secure by Design, Secure by Default and Secure in Deployment.

An overview of Microsoft SDL

Microsoft SDL is a software development model that includes specific security considerations. As such, Microsoft SDL conforms to ISO/IEC 27034-1:2011 "Information technology -- Security techniques -- Application security -- Part 1: Overview and concepts".

In a nutshell, a security requirements analysis must be completed for all system development projects. This analysis document acts as a framework and includes the identification of possible risks to the finished development project as well as mitigation strategies which can be implemented and tested during the development phases.

Critical security review and approval checkpoints are included during the system development lifecycle. All members of software development teams receive appropriate training to stay informed about security basics and recent trends in security and privacy. Individuals who develop software programs are required to attend at least one security training class each year.

Security training helps ensure software is created with security and privacy in mind and helps development teams stay current on security issues. Project team members are strongly encouraged to seek additional security and privacy education that is appropriate to their needs or products.

The Microsoft SDL process includes the following phases:

1. **Phase 1: Requirements**. The *Requirements* phase of the SDL includes the project inception - when the organization considers security and privacy at a foundational level - and a cost analysis - when determining if development and support costs for improving security and privacy are consistent with business needs. This phase also includes defining security roles and responsibilities and identifying individuals with these roles and responsibilities.

2. **Phase 2: Design**. The *Design* phase is when the organization builds the plan for how to take the project through the rest of the SDL process - from implementation, to verification, to release. During the Design phase the organization establishes best practices to follow for this phase by way of functional and design specifications, and by performing risk analysis to identify threats and vulnerabilities in the software. Threat Model Analysis (TMA) is required to define all attack surfaces and their associated risks; all security gaps and risks and documented and analyzed.

   Threat modeling is a team exercise, encompassing the operations manager, program/project managers, developers, and testers, and represents a key security analysis task performed for solution design. (This approach should also be considered by customers developing their own applications to be hosted in cloud services, either using IaaS or PaaS.)

   Microsoft Azure team members use the SDL Threat Modeling Tool to model all cloud-based services and projects, both when they are built and when they are updated with new features and functionality in Azure. Threat models cover all code exposed on the attack surface and all code written by or licensed from a third party and consider all trust boundaries.

   The STRIDE system (Spoofing, Tampering, Repudiation, Information disclosure, Denial of service, and Elevation of privilege) is used to help identify and resolve security threats early in the design process, before they can affect customers.

   This security impact analysis will result in dataflow documentation in order to identify all intended paths for information and potential attack vectors.

3. **Phase 3: Implementation**. The *Implementation* phase is when the organization creates the documentation and tools the customer uses to make informed decisions about how to deploy the software securely. To this end, the Implementation phase is when the organization establishes development best practices to detect and remove security and privacy issues early in the development cycle. Initial testing of elements begins in this phase.

4. **Phase 4: Verification**. During the *Verification* phase, the organization ensures that the code meets the security and privacy tenets established in the previous phases. This is done through security and privacy testing, and a security push - which is a team-wide focus on threat model updates, code review, testing, and thorough documentation review and edit. Additionally, service teams create a Security Incident Response document as part of their SDL requirements that outlines how security-specific incidents are addressed. A public release privacy review is also completed during the Verification phase.

5. **Phase 5: Release**. The *Release* phase is when the organization prepares the software for consumption and prepares for what happens once the software is released. One of the core concepts in the Release phase is response planning - mapping out a plan of action, should any security or privacy vulnerabilities be discovered in the release - and this carries over to post-release, as well, in terms of response execution. To this end, a Final Security Review and privacy review is required prior to release.

Moreover, with Agile development that now rules everything, and the DevOps software engineering approach that thus governs Microsoft Azure, engineering teams leverage the SDL for Agile Development Methodologies process (SDL-Agile) to integrate critical security practices into the Agile methodologies being used on a day-to-day basis with Continuous Integration (CI)/Continuous Delivery (CD).

As its name indicates, SDL-Agile is faithful to both SDL and to Agile, in which teams can innovate and react quickly to changing customer needs but in which the products they create are still more resilient to attack.

Note For more information, see page Secure DevOps.

SDL-Agile reorganizes above security practices into three categories:

1. **Every-Sprint practices**. The first category consists of the SDL requirements that are so essential to security that no software should ever be released without these requirements being met. Every SDL requirement in this category must be completed in each and every sprint, or the sprint is deemed incomplete, and the feature cannot be released.

   Some examples include: run analysis tools daily or per, threat model the features (see Threat Modeling: The Cornerstone of the SDL), and see only strong crypto in new code (AES, RSA, and SHA-256 or better).

2. **Bucket practices**. The second category consists of tasks that must be performed on a regular basis over the lifetime of the project but that are not so critical as to be mandated for each sprint. Currently there are three buckets in the bucket category - verification tasks (mostly fuzzers and other analysis tools), design review tasks, and response planning tasks. Instead of completing all bucket requirements each sprint, product teams must complete only one SDL requirement from each bucket of related tasks during each sprint.

3. **One-Time practices**. There are some SDL requirements that need to be met when you first start a new project with SDL-Agile or when you first start using SDL-Agile with an existing project. These are generally once-per-project tasks that won't need to be repeated after they're complete. The one-time requirements should generally be easy and quick to complete, with the exception of creating a baseline threat model.

Resources:

- Microsoft Security Development Lifecycle (SDL).
- Simplified Implementation of the Microsoft SDL.
- Microsoft Security Development Lifecycle for Agile Development (SDL-Agile).
- SDL for Agile.
- Applying the SDL to Microsoft Azure.

Resulting security benefits for our customers

As shortly described above, a formal review process is implemented to ensure that new or modified source code authored by Microsoft Azure staff is developed in a secure fashion, no malicious code has been introduced into the system, and that proper coding practices are followed. The reviewers' names, review dates, and review results are documented and maintained for audit purposes.

Similarly, a formal security quality assurance process is implemented to test for vulnerabilities to known security exposures and exploits. The process includes the use of automated security testing tools and requires that all high vulnerabilities get remediated before the system will be released to production. Microsoft Azure have implemented information validation through checking of data inputs as part of the SDL process. Thorough code reviews and testing are completed during the above *Verification* phase of the SDL prior to software being put into a production environment. The code reviews and testing check for cases of SQL injection, format string vulnerabilities, cross-site scripting (XSS), integer arithmetic, command injection, and buffer overflow vulnerabilities. This satisfies the lifecycle support control through effective management of the risks associated with failing to implement a system development lifecycle methodology that lacks information security considerations.

After a software program is released, the product development team must be available to respond to any possible security vulnerabilities or privacy issues that warrant a response. In addition, the development team is required to create a response plan that includes preparations for potential post-release issues.

Patches, updates and threat mitigation are all covered by Microsoft SDL, a detailed, robust practice that Microsoft has developed over many years. Part of the Microsoft SDL has been built upon former investments in Microsoft Trustworthy Computing (TwC). We have various patch management release cycles and engagement models that allow us to mitigate new threats as quickly as possible within the service.

Security risk mitigation when using open source software

Modern software projects are increasingly dependent on open source software (OSS), from operating systems through to user interface widgets, from back-end data analysis to front-end graphics, and Microsoft is no exception.

Open source software has led to some amazing benefits, but they are sometimes accompanied by security risks that must be understood and managed. For the most part, these risks can apply when using any third-party software component, whether open source or commercial.

Open source, like any software, can contain security defects, which can become manifest as vulnerabilities in the software systems that use them. Since source code is generally available for open source components, it can often be easier for security researchers to identify new vulnerabilities, and while most researchers will follow responsible disclosure methods when reporting issues to the maintainer, there is a risk that some vulnerabilities will become weaponized and used to attack systems that use them.

Exacerbating this, open source components are generally released as needed, often with little to no advance notice to the user community, so when a vulnerability is fixed and a new release is published, there is often a lag until users can upgrade to the new version; this lag can give adversaries time to create and launch an exploit. As a result, it's very important to update open source components in a timely manner, especially when they contain security fixes.

Microsoft Azure Security Service Engineering has implemented a set of practices, along with tools and techniques to help manage security risks in third-party components.

Note For more information, see page Using Open Source.

This comprises:

- **Inventorying open source**. Properly managing the use of open source software components first consists in understand which components are in use. This obviously requires automation. Fortunately, modern agile development practices (e.g. above mentioned SDL for Agile) already rely heavily on automated tooling, and so are easily adapted to include capabilities in this area. (Without endorsing here a specific tool and service, there are many tools available in this space, including open source tools like OWASP Dependency Check and NPM Audit, and commercial services like WhiteSource Bolt, among many others.)

Note Inventory generation takes place at a natural point in the development lifecycle, such as during pull-request validation or branch merging, with the inventory results being stored centrally and accessible to appropriate personnel (including the Microsoft Security Incident Response Program).

- **Performing security analysis**. All identified components must be validated to ensure they are free of security vulnerabilities, to the level of fidelity required by the policy in place. The following activities are typically considered: checking for public vulnerabilities, using commercial security intelligence, performing static analysis, and performing comprehensive security reviews.

- **Keeping open source up to date**. One of the most effective ways to manage security risk related to the use of open source is to keep components up to date, even in the absence of known vulnerabilities.

Note This can be a security benefit because security vulnerabilities are often fixed without explicit public disclosure, and while the engineering cost of doing this isn't free, benefits extend beyond security (such as engineering agility, taking advantage of new features and bug fixes).

- **Aligning security response processes**. When a vulnerability is found or reported in an open source component, a strategy for managing the process, which should align directly with the organization's overall security response plan. At Microsoft, we use the Microsoft Security Response Center (MSRC) to coordinate response activities related to vulnerabilities in open source components (see next section below).

Technical vulnerabilities management

Vulnerability watch process

Microsoft Azure identifies, reports, and corrects information system flaws through vulnerability management, incident response management, and patch / configuration management processes.

Note For more information, see Azure infrastructure monitoring.

The Microsoft Azure Security Incident Response Program assists with identifying and reporting of information system flaws through a global, 24x7 incident response service that works to mitigate the effects of attacks and malicious activity. The incident response team follows established procedures for incident management, communication, and recovery, and uses discoverable and predictable interfaces with internal and external partners alike.

Vulnerability-related data are received from multiple sources of information, which include: Microsoft Security Resource Center (MSRC), vendor Web sites, other third-party services (e.g., Internet Security Systems) and internal / external vulnerability scanning of services. Microsoft Azure Security Service Engineering will determine which updates are applicable within the Azure environment. Potential changes are tested in advance.

Patching schedules are defined by Microsoft Azure Security Service Engineering as follows:

- 30 days for high vulnerabilities.
- 90 days for medium/moderate vulnerabilities.

In addition, Microsoft works with a variety of different industry bodies and security experts to understand new threats and evolving trends. We constantly scan our systems for vulnerabilities, and we contract with external penetration testers who also constantly scan the systems.

Resource:

- Operational Security for Online Services Overview. Provides insight into how Microsoft applies its resources to online services in ways that extend beyond traditional standards and methodology to deliver industry-leading capabilities.

Vulnerability risk assessment

Microsoft Azure performs risk assessments of its environment to review the effectiveness of information security controls and safeguards, as well as to identify new risks. The risks are assessed annually, and the results of the risk assessment are presented to management through a formal risk assessment report.

The Online Services Security and Compliance (OSSC) team within MCIO manages the Information Security Management System (ISMS) (and was created to ensure that Microsoft Azure is secure, meets the privacy requirements of our customers, and complies with complex global regulatory requirements and industry standards). The OSSC team monitors ongoing effectiveness and improvement of the ISMS control environment by reviewing security issues, audit results, and monitoring status, and by planning and tracking necessary corrective actions.

Identification, assessment, and prioritization of risks are performed as part of Azure's risk management program and verified as part of the ISO/IEC 27001:2013 audit (see section § "Clearly stated cloud principles of trust for your applications and data" above).

In addition, Microsoft employs a method named "Red Teaming" to improve Microsoft Azure security controls and processes through regular penetration testing. The Red Team is a group of full-time staff within Microsoft that focuses on performing targeted and persistent attacks against Microsoft infrastructure, platforms, and applications, but not end-customers' applications or data.

The job of the Red Team is to simulate the kinds of sophisticated, well-funded targeted attack groups that can pose a significant risk to cloud services and computing infrastructures. To accomplish this simulation, the team researches and models known persistent adversaries, in addition to developing their own custom penetration tools and attack methods.

Because of the sensitive and critical nature of the work, Red Team members at Microsoft are held to very high standards of security and compliance. They go through extra validation, background screening, and training before they are allowed to engage in any attack scenarios. Although no end-customer data is deliberately targeted by the Red Team, they maintain the same Access To Customer Data (ATCD) requirements as service operations personnel that deploy, maintain, and administer Microsoft Azure (see section § "Access to customer data by Microsoft personnel" above) The Red Team abides by a strict code of conduct that prohibits intentional access or destruction of customer data, or disruptions to customer Service Level Agreements (SLAs).

A different group, the Blue Team, is tasked with defending Microsoft Azure and related infrastructure from attack, not only from the Red Team but from any other source as well. The Blue Team is comprised of dedicated security responders as well as representatives from Microsoft Azure Engineering and Operations. The Blue Team follows established security processes and uses the latest tools and technologies to detect and respond to attacks and penetration. The Blue Team does not know when or how the Red Team's attacks will occur or what methods may be used - in fact, when a breach attempt is detected, the team does not know if it is a Red Team attack or an actual attack from a real-world adversary. For this reason, the Blue Team is on-call 24x7, 365 days a year, and must react to Red Team breaches the same way it would for any other adversary.

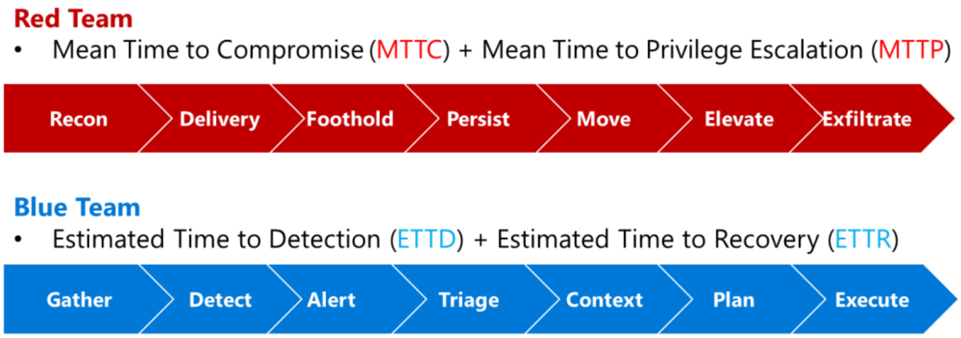

Red Team
Mean Time to Compromise (MTTC) + Mean Time to Privilege Escalation (MTTP)
Recon
Delivery
Foothold
Persist
Move
Elevate
Exfiltrate
Blue Team
Estimated Time to Detection (ETTD) + Estimated Time to Recovery (ETTR)
Gather
Detect
Alert
Triage
Context
Plan
Execute

Figure 18 RED Team vs. BLUE Team

Microsoft understands that security assessment is also an important part of customer application development and deployment. Therefore, Microsoft has established a policy for customers to carry out authorized penetration testing on their applications hosted in Microsoft Azure. Because such testing can be indistinguishable from a real attack, it is critical that customers conduct penetration testing only after notifying Microsoft. Penetration testing must be conducted in accordance with Microsoft terms and conditions.

Resources:

- Microsoft Cloud Red Teaming. Explores the Red Teaming method, how attacks are conducted and defended against, and the history and rationale behind the practice.
- Red vs. Blue - Internal security penetration testing of Microsoft Azure. A brief video explaining the Azure penetration testing approach and discussing the roles of the Red and Blue teams.
- Penetration testing. Explains the process by which customers who wish to formally document upcoming penetration testing engagements against Microsoft Azure are encouraged to fill out the Azure Service Penetration Testing Notification form.

Important note While notifying Microsoft of pen testing activities is no longer required customers must still comply with the Microsoft Cloud Unified Penetration Testing Rules of Engagement.

Important note If, during a penetration testing, you believe you have discovered a potential security flaw related to the Microsoft Cloud or any other Microsoft service, please report it to Microsoft within 24 hours by following the instructions on the Report a Computer Security Vulnerability page. Once submitted, you agree that you will not disclose this vulnerability information publicly or to any third party until you hear back from Microsoft that the vulnerability has been fixed. All vulnerabilities reported must follow the Coordinated Vulnerability Disclosure principle.

Information security incidents

Assessment of information security events and decision-taking

Microsoft follows a 5-step incident response process when managing both security and availability incidents for the Azure services.

Important note The form Microsoft Online Services Security Incident and Abuse Reporting is available to report suspected security issues or abuse of Microsoft Azure. This includes malicious network activity originating from a Microsoft IP address. It also includes distribution of malicious content or other illicit or illegal material through Microsoft Azure.

The goal for both types is to restore normal service security and operations as quickly as possible after an issue is detected, and an investigation is started. The response is implemented using a five-stage process which shows the following activities:

1. **Detect.** First indication of an event investigation.
2. **Assess.** An on-call incident response team member assesses the impact and severity of the event. Based on evidence, the assessment may or may not result in further escalation to the security response team.
3. **Diagnose.** Security response experts conduct the technical or forensic investigation, identify containment, mitigation, and workaround strategies. If the security team believes that customer data may have become exposed to an unlawful or unauthorized individual, parallel execution of the Customer Incident Notification process begins in parallel.
4. **Stabilize, Recover.** The incident response team creates a recovery plan to mitigate the issue. Crisis containment steps such as quarantining impacted systems may occur immediately and in parallel with diagnosis. Longer term mitigations may be planned which occur after the immediate risk has passed.
5. **Close/ Post Mortem.** The incident response team creates a post-mortem that outlines the details of the incident, with the intention to revise policies, procedures, and processes to prevent a reoccurrence of the event.

The Security Incident Response Team may move back and forth between diagnose to stabilize as the investigation progresses.

The detection processes used by Azure are designed to discover events that risks the confidentiality, integrity, and availability (CIA) of Azure services. The *Assess* stage of an incident response is a rapid triage effort in order to execute a preliminary assessment to evaluate the impact and severity of the event. Based on evidence, the assessment may or may not result in further escalation to the security response team.

Then in the *Diagnose* stage, Security response experts conduct the technical or forensic investigation, identify containment, mitigation, and workaround strategies. If the security team believes that customer data may have become exposed to an unlawful or unauthorized individual, parallel execution of the Customer Incident Notification process begins in parallel.

Resource:

- Microsoft Azure Security Response in the Cloud. Describes how Microsoft investigates, manages, and responds to security incidents within Azure.

Response to information security incidents

If during the investigation of a security event, Microsoft becomes aware that customer data has been accessed by an unlawful or unauthorized party, the security incident manager will immediately begin execution of the Customer Security Incident Notification Process. This can occur at any point of the incident lifecycle, but usually begins during the Assess or Diagnose phases. The security incident manager only needs reasonable suspicion that a reportable event has occurred to begin execution of this process. The investigation and mitigation need not be completed before this process begins in parallel.

The goal of the customer security incident notification process is to provide impacted customers with accurate, actionable, and timely notice when their customer data has been breached. Such notices may also be required to meet specific legal requirements.

Microsoft Cyber Defense Operations Center

The Microsoft Cyber Defense Operations Center (CDOC) brings together security response experts from across the company to help protect, detect, and respond to threats in real-time. Staffed with dedicated teams 24x7, CDOC has direct access to thousands of security professionals, data scientists, and product engineers throughout Microsoft to ensure rapid response and resolution to security threats.

Informed by trillions of data points across an extensive network of sensors, devices, authentications, and communications, the Center employs automated software, Machine Learning, behavioral analysis, and forensics to create an intelligent security graph.

The resulting Microsoft Intelligent Security Graph uses advanced analytics to synthesize massive amounts of threat intelligence and security signals obtained across Microsoft products, services, and partners to combat cyberthreats. Millions of unique threat indicators across the most diverse set of sources are generated every day by Microsoft and its partners and shared across Microsoft products and services. Across its portfolio of global services, each month Microsoft scans more than 400 billion email messages for phishing and malware, processes 450 billion authentications, executes more than 18 billion page scans, and scans more than 1.2 billion devices for threats. Importantly, this data always goes through strict privacy and compliance boundaries before being used for security analysis.

The Intelligent Security Graph provides an unparalleled view into the evolving threat landscape and enables rapid innovation to detect and respond to threats. Machine Learning models and Artificial Intelligence reason over vast security signals to identify vulnerabilities and threats.

This threat intelligence insight helps our teams connect the dots, then counter with strong containment and coordinated remediation.

Microsoft France
39 Quai du Président Roosevelt
92130 Issy-Les-Moulineaux

MICROSOFT EXCLUDES ANY EXPRESS, IMPLICIT OR LEGAL GUARANTEE RELATING TO THE INFORMATION IN THIS DOCUMENT.