Demystifying Public Cloud Auditing for IT Auditors

Jacques Putters, Jalal Bani Hashemi, and Ayhan Yavuz

1 Introduction

Over the course of the past decade, cloud computing has become the underpinning infrastructure that supports trends such as the Internet of Things, data analytics, and artificial intelligence. It is giving organisations a competitive advantage in digital transformation in terms of innovation, agility, resilience, and skills. As more organisations become more aware of these prospects, adoption of public cloud is taking place at a fast pace. In addition, 'The economic, organizational and societal impact of the pandemic will continue to serve as a catalyst for digital innovation and adoption of cloud services', said Henrique Cecci, senior research director at Gartner. 'This is especially true for use cases such as collaboration, remote work, and new digital services to support a hybrid workforce'. As a result, global cloud adoption will continue to expand rapidly. Gartner forecasts end-user spending on public cloud services to grow from $396 billion in 2021 to reach $482 billion in 2022 (Gartner, 2021). Additionally, by 2026, Gartner predicts public cloud spending will exceed 45% of all enterprise IT spending, up from less than 17% in 2021.

The financial services industry was initially hesitant to adopt public cloud technology. Primarily security and compliance concerns in addition to an unclear regulatory position prevented them from migrating regulated workloads into the public cloud and made many of them instead choose for private cloud implementations (Association for Financial Markets in Europe, 2019). These concerns usually pertained to data compromise or exploitation by Cloud Service Providers (CSPs), other CSP clients or law enforcements offices, vendor lock-in, inability to perform control and audit activities, and the loss of physical control. In

J. Putters · J. B. Hashemi · A. Yavuz (✉)
Group Audit, ABN AMRO Bank, Amsterdam, The Netherlands
e-mail: jacques.putters@nl.abnamro.com; ayhan.yavuz@nl.abnamro.com

E. Berghout et al. (eds.), *Advanced Digital Auditing*, Progress in IS,
https://doi.org/10.1007/978-3-031-11089-4_8

addition, the financial services industry is heavily regulated, causing financial institutions to be very—sometimes overly—cautious.

As financial institutions have become increasingly aware that—to stay competitive—the adoption of public cloud technology is a bare necessity, they have been trying to address the aforementioned concerns. In parallel, several guidelines have been published by authorities such as the European Banking Authority (EBA) (2019) and the European Securities and Markets Authority (ESMA) (2020) to ensure that the financial services industry and its regulators have a clear set of standards that say how to address these concerns.

The purpose of this article is to provide a conceptual framework that can be used for auditing operational public cloud systems. We will do this by first describing some of the characteristics of cloud computing in Sect. 2. In Sect. 3 we will outline the journey we—as IT auditors—made to address the challenges that the introduction of public cloud technology brought about. This will include a description of the different frameworks and audit programs we used as a basis for our audit activities. Section 4 contains the case description: It outlines the IT/Cloud transformation that our organisation—ABN AMRO Bank—has been going through since 2012. This transformation initially related to the implementation of a private cloud and was later followed by both the rollout of DevOps and the large-scale migration of applications to Microsoft Azure, but we will limit ourselves to the audit on the operational Microsoft Azure environment and exclude the change program and the DevOps transformation from the scope. Section 5 contains a description of the several audits we have done on public cloud since the start of the bank's journey to the cloud. Based on the knowledge and experience that were gained during the execution of the different audits, we designed a conceptual framework that can be used to organise and define IT audits for public cloud systems. This framework is described in Sect. 6. It does not specifically address the audit of Software as a Service, although some elements of the framework apply to SaaS as well. In Sect. 7, we outline a few discussions regarding auditing public cloud systems and the presented framework. This article ends with our conclusions in Sect. 8.

2 Cloud Computing

There are various definitions of cloud computing. Amazon Web Services[1] (Amazon. (n.d.)) defines cloud computing as 'The on-demand delivery of IT resources over the Internet with pay-as-you-go pricing'. A frequently used definition has been published by the National Institute of Standards and Technology (NIST) (Mell & Grance, 2011): 'Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly

[1] https://aws.amazon.com/what-is-cloud-computing/

provisioned and released with minimal management effort or service provider interaction'. While these definitions give a first idea of what cloud computing is, it can be further illustrated by its characteristics as defined by NIST (Mell & Grance, 2011). These have been universally adopted/accepted and are being referred to in many publications.

The first characteristic is the *on-demand delivery of services*. This means that cloud-based services/resources are provisioned without any human interaction with the cloud service provider. They are delivered automatically whenever and wherever they are needed. These services could be virtual machines, databases, storage, etc. The second characteristic is *broad network access*. It means that access is enabled to whatever resource you want, whenever needed, from any location, if you have (Internet) network access. Network bandwidth and latency are key factors to consider because they will determine the quality of service. The third characteristic is *multi-tenancy/resource pooling*. Multi-tenancy means that software and the associated infrastructure serves multiple customers (tenants), at the same time ensuring data privacy and security/isolation on a logical level. In addition, resource pooling means that different physical and virtual resources are dynamically assigned and reassigned according to customer demand across the client base. The fourth characteristic is *scalability and elasticity*: having the ability to quickly provision/scale up or decommission/scale down resources in the cloud whenever required. The fifth characteristic—*measured service*—means that the usage of resources is measured and reported by the cloud service provider and that clients pay in line with resource usage.

There are four main types of cloud deployment models, i.e. public, private, hybrid, and community cloud computing. Cloud deployment models classify cloud environments based on several criteria, such as ownership, purpose, and scale. As every model has its benefits and disadvantages, companies should choose a model—or a combination of models—that best meets their needs.

Public clouds are owned, managed, and controlled by cloud service providers. They are aimed at making cloud services available to multiple customers (tenants) and they offer extremely high scalability, performance, and throughput thanks to the enormous size of public cloud technology. Some examples of large public cloud service providers are Amazon Web Services, Microsoft Azure, and Google.

A *private cloud* is usually owned, controlled, and used by one single company, but it might be managed/operated by a third party. It offers the owner a much higher level of control as compared to a public cloud, but it comes at a price: considerably higher costs are incurred, as they include the costs of traditional data centre ownership as well as the costs of managing the related infrastructure. In addition, although the technical differences between a public and private cloud are small, private clouds will usually be much smaller in size than their public counterparts. Many public cloud service providers also offer solutions that can be used to implement and support a private cloud environment for customers that need to control their whole IT infrastructure.

Community clouds are quite similar to private clouds. The main difference is that—instead of one company—several companies will own, control, and share the infrastructure and related community cloud resources. Usually, these companies have similar backgrounds and shared interests and—consequently—similar requirements, e.g. in the areas of compliance, privacy, or security.

A *hybrid cloud* consists of a combination of two or more interconnected cloud deployment models (public, private, or community) that allows companies to choose the cloud environment that best meets the needs of the applications and associated data, even on a per case basis. It offers companies a good compromise between costs and control.

In addition to the distinction between the various cloud deployment models, a distinction can also be made between the three distinct types of cloud computing (Jones, E. (2021): (1) Infrastructure as a Service (IaaS), (2) Platform as a Service (PaaS), and (3) Software as a Service (SaaS). Just like with the cloud deployment models, companies should choose the types of cloud computing services that best meet their requirements as to the level of control, management effort, flexibility, and costs.

IaaS gives companies internet access to processing power, storage, and network facilities, which they can use to deploy and run all kinds of software. IaaS offers the highest level of control and flexibility, but it also requires the most management effort. The cloud service provider manages and controls the underlying cloud infrastructure, but the companies using IaaS services have control over the software deployed on top of the IaaS services (e.g. operating systems, applications).

PaaS provides companies with access to all cloud services that are needed to manage the entire lifecycle of applications, without the burden of having to manage the underlying infrastructure. It includes all software needed to design, program, test, deploy, and run applications. The service provider controls and manages the cloud infrastructure, operating systems, middleware, storage, etc., and the company controls and manages the applications.

SaaS provides companies with internet access to applications that are controlled and managed by the service provider (although some application settings might be configurable). SaaS offers the lowest level of control to the companies using these applications, but also the lowest management effort. In many cases, the SaaS provider will use PaaS or IaaS services from other service providers. The following figure (Fig. 1) shows how the three cloud service types compare as to the level of control, flexibility, management effort, and costs/efficiency.

3 Audit Programs for Public Cloud Audits

Although virtualisation has been around since the 1960s and the first cloud (SaaS) applications became available in the late 1990s, public cloud only really took off when Amazon launched Amazon Web Services in 2006. And although public cloud is being adopted at an accelerating pace, most IT auditors are still quite unfamiliar

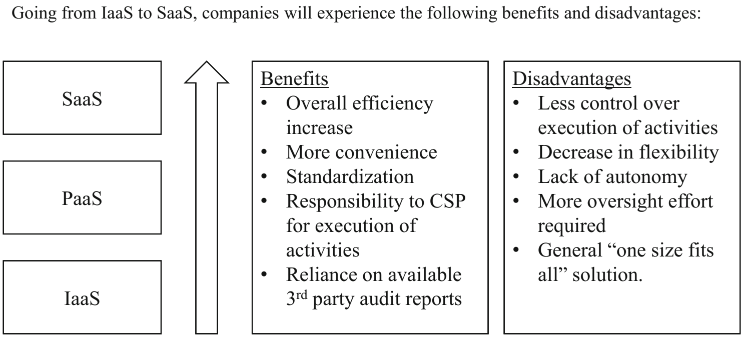

Going from IaaS to SaaS, companies will experience the following benefits and disadvantages:
SaaS
PaaS
IaaS
Benefits
• Overall efficiency increase
• More convenience
• Standardization
• Responsibility to CSP for execution of activities
• Reliance on available 3rd party audit reports
Disadvantages
• Less control over execution of activities
• Decrease in flexibility
• Lack of autonomy
• More oversight effort required
• General "one size fits all" solution.

Fig. 1 Comparison of the benefits and disadvantages of the different cloud service types

with the subject matter. Over the course of the past years, some auditors who were being confronted with public cloud on a professional level have resorted to the Certified Cloud Security Professional certification from ISC2 to obtain the required knowledge to audit public cloud developments and systems. Only recently the 'Certificate of Cloud Auditing Knowledge™' (CCAK™) was introduced by the Cloud Security Alliance® (CSA), a global leader in cloud security research, training, and credentialing and ISACA® a global leader in training, education, and certification for IS/IT professionals.

In addition, although there are some audit programs available to help IT auditors figure out how to audit public cloud system(s), processes, and organisation that are included in the scope of their engagement, a holistic view on auditing public cloud subjects is still missing. This is exacerbated by the fact that in a public cloud world there are many variables to consider when defining the scope and objectives of the engagement. What type of cloud computing deployment model is the object of our audit (e.g. public cloud, private cloud, hybrid cloud)? And, which type of service model (e.g. IaaS, PaaS or SaaS or a combination of these)? To what extent are outsourcing controls relevant? Can we rely on available assurance reports? Does the audit relate to a BAU (Business as Usual) system, or do we need to take the migration of a system and associated data to the public cloud into account?

Regardless of the type/subject of audit, it is crucial to plan ahead. Audit programs or plans can be helpful and having them before starting an audit engagement is in most cases mandatory. The audit procedures included in audit programs are to ensure that auditors meet the specific criteria for an audit assignment. Furthermore, the audit program looks to create a framework that can provide auditors with guidelines. The following sections give a brief overview of (1) the shared responsibility model, (2) frameworks/sets of best practices, and (3) work programs that are currently available and that can help auditors design a suitable audit program, including the required audit procedures.

3.1 Shared Responsibility Model

Cloud service providers explicitly communicate the shared responsibility model to their clients. It explains their view on the responsibilities of management and security of their cloud services as managed by the organisation/consumer as it deploys workloads in the cloud, versus those managed by them as the provider of those services. The line between the organisation's responsibilities and those of the provider is also the demarcation between the assets, processes, functions, and associated controls that the provider owns and is responsible for, and the ones of the organisation/consumer. Please note that the views on the shared responsibility model vary between the different CSPs. The following tables show the shared responsibility model as used/published by Microsoft and Amazon Web Services, respectively (Tables 1 and 2):

In a traditional data centre/on-premises model, the organisation/consumer is responsible for management and security across its entire operating environment, including applications, physical servers/hardware, network configuration, user controls, and even physical and environmental security/control. In a cloud environment, the service provider takes on a share of the operational burden. By working together with the CSP and by sharing portions of the security responsibilities, it is possible to maintain a secure environment with less operational costs. When CSPs speak of 'shared responsibility', it is important to understand that the user and CSP never

Table 1 The shared responsibility model according to Microsoft (2022c)

	Responsibility	**SaaS**	**PaaS**	**IaaS**	**On-Prem**
Responsibility always retained by the customer	Information and data	■	■	■	■
	Devices (Mobile and PCs)	■	■	■	■
	Accounts and identities	■	■	■	■
Responsibility varies by type	Identity and directory infrastructure	■	■	■	■
	Applications	□	■	■	■
	Network controls	□	■	■	■
	Operating system	□	□	■	■
Responsibility transfers to cloud provider	Physical hosts	□	□	□	■
	Physical network	□	□	□	■
	Physical datacenter	□	□	□	■

■ Microsoft, □ Customer, ▨ Shared

Table 2 The shared responsibility model according to Amazon Web Services (Amazon, 2021)

CUSTOMER responsibility for security 'in' the cloud	Customer Data			
	Platform, applications, Identity & Access Management			
	Operating system, Network & Firewall configuration			
	Client-side data	Server-side encryption (file system and/or data)	Networking traffic protection (encryption, integrity, identity)	
	Encryption & Data Integrity			
	Authentication			
AWS responsibility for security 'of' the cloud	SOFTWARE			
	Compute	Storage	Database	Networking
	HARDWARE/AWS GLOBAL INFRASTRUCTURE			
	Regions	Availability zones	Edge locations	

really share responsibility for a single aspect of operations. The parts of the application and infrastructure stack that a consumer can configure, are solely managed by the consumer of the services, and the CSP does not dictate how the service consumer should secure his parts. Likewise, the user/consumer has no control over how the CSP secures their portions of the application and infrastructure stack. The user/consumer usually has the ability and right to access the CSP's certifications and related reports (e.g. SOCI, SOCII, SOCIII, FedRamp, ISO) to verify that their systems are secure and that they are adhering to the agreed terms and conditions. CSPs publish these reports regularly and freely, and the most current reports are always accessible to their clients. Please note that not all CSPs offer one or more of these reports as it can be costly to produce them/obtain these certifications.

In our cloud audits, we have used the Microsoft Azure Shared Responsibility Model to make clear demarcations of the in-scope and out-of-scope elements in our audit engagements. Moreover, we have also used the model in our audit planning process to find gaps in our audit coverage.

3.2 *Frameworks*

Some existing frameworks give a solid foundation for the creation of work programs for audits on public cloud systems, these are described below:

- *ISACA:* COBIT (Control Objectives for Information and Related Technologies) (Haes et al. (2015))—framework for enterprise governance of IT. The framework defines a set of generic processes for the management of IT, with each process defined together with process inputs and outputs, key process-activities, process objectives, performance measures, and an elementary maturity model.
- *AXELOS:* ITIL (Information Technology Infrastructure Library) (Axelos, (2020))—a library of best practices for managing IT services and improving IT support and service levels. One of the most essential parts of ITIL is the

configuration management database (CMDB), which provides the central authority for all components—including services, software, IT components, documents, users, and hardware—that must be managed to deliver an IT service.

- *The National Institute of Standards and Technology* (NIST) Information Technology Laboratory regularly publishes research, standards, and guidelines on information systems and security. NIST Special publication SP 800-53 outlines the standards and guidelines for Security and Privacy Controls for Information Systems and Organizations. This publication lists the controls that will enable companies to protect themselves against a diverse set of threats and risks. The controls cover 20 areas, including access control, awareness and training, audit and accountability, contingency planning, and incident response. The classification of the information system (low, medium, or high) will determine the controls that must be implemented and monitored. SP 800-53 is widely used by cloud service providers as the set of reference controls that they have their audit or compliance teams audit them against.
- *ABN AMRO IT Organisation: Standards for Cloud Risk Control.* In the proof-of-concept phase of the Microsoft Azure and the Amazon Web Services platforms, the ABN AMRO IT organisation defined the Standards for Cloud Risk Control to guide the implementation of workloads on these platforms. The standards were created in close collaboration with cloud specialists from Azure and AWS and representatives from both the Corporate Information Security Office and the Corporate Technology Office. These standards define which controls need to be implemented to adopt and use the platform services securely. The focus of these requirements is on which controls teams must implement. The standards apply to IaaS, PaaS and SaaS services.

3.3 Audit Programs

There are several audit programs that can be used for auditing cloud service providers or implementations of public cloud in organisations.

First, there is the Cloud Controls Matrix (CCM) of the Cloud Security Alliance. CCM is composed of 197 control objectives that are structured in 17 domains (shown below in REF), covering key aspects of cloud technology. The controls in the Cloud Controls Matrix (CCM) are mapped against industry-accepted security standards, regulations, and control frameworks including but not limited to ISO 27001/27002/27017/27018, NIST SP 800-53, AICPA TSC, German BSI C5, PCI DSS, ISACA COBIT, NERC CIP, FedRamp, CIS, and many others (Table 3).

It can be used as a tool for the systematic assessment of a cloud implementation and provides guidance on which security controls should be implemented by which actor within the cloud supply chain and is considered a de-facto standard for cloud security assurance and compliance (CSA, 2021).

Second, there is ISACA's Cloud Computing Management Audit Program (ISACA, 2020–2021), which focuses on the governance affecting cloud computing,

Table 3 Overview of 17 domains of the Cloud Controls Matrix (CCM)

Audit and Assurance	Identity and Access Management
Application and Interface Security	Interoperability and Portability
Business Continuity Management & Operational Resilience	Infrastructure & Virtualization Security
Change Control and Configuration Management	Logging and Monitoring
Cryptography, Encryption, and Key Management	Sec. Incident Management, E-discovery & Cloud Forensics
Datacentre Security	Supply Chain Management, Transparency, and Accountability
Data Security and Privacy	Threat and Vulnerability Management
Governance, Risk Management, and Compliance	Universal Endpoint Management
Human Resources Security	

Table 4 Processes of ISACA's Cloud Computing Management Audit Program

Governance of Cloud Computing Services	Incident response, notification, and remediation
Enterprise Risk Management	Application Security
Information Risk Management	Compliance
Third-party Management	Tools and Services
Legal and Electronic Discovery	Application Functionality
Legal Compliance	Data Security and Integrity
Right to Audit	Key Management
Auditability	Identity and Access Management
Compliance Scope	Virtualization
Certifications	Standards and Best Practices
Service Transition Planning	

Table 5 Areas of ISACA's Azure Audit Program

Governance	Logging and monitoring
Network configuration and management	Security incident response
Identity and access management	Data encryption controls
Resource security	

contractual compliance between the service provider and customer, and control issues specific to cloud computing. Controls and test-steps are included (with mapping to COBIT5) and cover the following processes (Table 4):

Third, there is ISACA's Azure Audit Program (ISACA, 2020–2022), which helps auditors in their assessments of whether the enterprise's use of Azure services supports achievement of strategic goals through covering the following areas (Table 5):

The Cloud Computing Management Audit Program is agnostic to the cloud platform being used, while the Azure Audit Program holds specific details and is tailored towards the Azure environment. For our specific use case, these two programs were complementary to each other.

3.4 Suitability of the Available Frameworks and Work Programs

The shared responsibility model and the available frameworks and work programs will be of added value for IT auditors when deciding how to audit public cloud implementations, system(s), processes, and organisation(s). The shared responsibility model provides IT auditors with a reference for deciding what to expect from the user organisation versus what to expect from the service provider per type of cloud computing. This is especially important for scoping purposes. In addition, the frameworks and work programs mentioned in the previous paragraphs give a basis for auditing specific aspects of public cloud implementations such as encryption & key management, governance & risk management, infrastructure & virtualisation, third party (risk) management, etc. However, there are three disadvantages to the use of these work programs. First, they lack the holistic perspective, as they do not show or explain the relative importance and interrelationships between the individual components of the work programs. Second, although the level of detail differs between these work programs, none of them are sufficiently specific and give the required guidance for more experienced IT auditors if they aim to do a more in-depth audit of public cloud implementations. And third, these frameworks and work programs do not distinguish between the platform and the workloads running on the platform, although this is a relevant distinction when auditing public cloud implementations.

4 Case Description: The ABN AMRO IT/Cloud Transformation

4.1 ABN AMRO Bank

Headquartered in Amsterdam and employing some 18,000 people, ABN AMRO is the third largest bank in the Netherlands. The foundation of the current bank was laid when the ‘Algemene Bank Nederland’ (ABN) merged with the ‘AMRO Bank’ in 1991, thereby creating the largest bank in the Netherlands, the 6th bank in Europe and the 16th bank worldwide. A period of domestic and international mergers and acquisitions followed. By 2007, ABN AMRO was the second-largest bank in the Netherlands and the eighth largest in Europe by assets. It had operations in 63 countries, with over 110,000 employees.

In 2007, a consortium that consisted of the Royal Bank of Scotland (RBS), Fortis, and Banco Santander under the name RFS Holdings, made an offer on the shares of ABN AMRO. This offer was accepted by the shareholders in September 2007 and ABN AMRO was split up by its new owners.

When in 2008 the global financial crisis hit the financial service industry, the Belgian-Dutch Fortis Group that had taken over the ABN AMRO Business Units Netherlands, Asset Management and Private Banking had to be bailed out by the Dutch and Belgian governments. The Dutch government bought the Dutch activities of Fortis Bank, Fortis' insurance activities, and Fortis' share in the ABN AMRO Bank. The Dutch government later decided that these parts would be integrated in the new ABN AMRO Bank which eventually took place on 1 July 2010.

The current ABN AMRO Bank is aiming to become a personal bank in the digital age. This strategy rests on three pillars:

1. Reinvent the customer experience: Getting closer to clients and offering them a fully digital experience with best-in-class services and products.
2. Support our clients' transition to sustainability.
3. Building a future-proof bank.

Information Technology is at the heart of the bank's strategic goals. To improve the productivity and lower the costs of IT, senior IT management decided to transform from an agile into a DevOps organisation. This transformation would be strengthened by also moving to the public cloud and away from a managed IT service provider. The bank applies a cloud-first strategy and has chosen Azure as its strategic cloud platform and AWS as its challenger cloud platform (Monterie, 2020).

4.2 IT Within ABN AMRO Bank

IT within ABN AMRO has its foundation in the IBM Mainframe systems that have been used since the 1960s. But over the course of the last 30 years, a wide variety of platforms had been added to the environment, especially at the end of the 1990s when large-scale client-server system implementations took place. This resulted in an overly complex, expensive, and difficult-to-control situation. A large variety of platforms was used: Open VMS, HP Unix, AIX, Linux, Solaris, AS400, Windows server, Tandem, etc. As the need for a reduction of complexity and cost grew, a virtualised platform was identified as a means to accomplish this. ABN AMRO decided to select one of the private cloud offerings from IBM to become the platform of choice for the years to come. It also enabled the organisation to explore cloud technology and to experience how to make applications cloud-ready. In 2016, this on-premises dedicated cloud platform went live, and a program was started to migrate hundreds of applications from the legacy/midrange platforms to this private cloud environment.

It was already clear at the time of implementation that the functionality offered by the private cloud would not be able to compete with the ones from large cloud

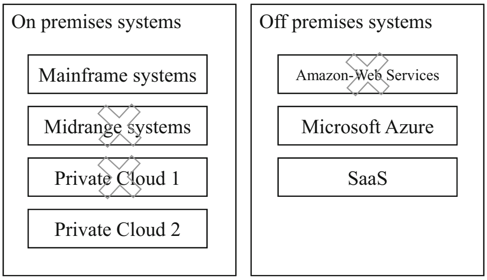


Fig. 2 ABN AMRO platform landscape after the cloud transformation

service providers such as Amazon Web Services, Microsoft Azure, and Google. Consequently, in 2017 two proofs of concept were started to experiment with Microsoft Azure as well as with Amazon Web Services. Secondly, an alternative private cloud solution was explored and implemented. The proof of concept of the two public cloud platforms was so successful that a multi-platform strategy was finally adopted where both Microsoft Azure and Amazon Web Services had their place. The two private cloud solutions were maintained, next to the traditional Mainframe environment.

In 2019 IT began to realise that—even although steps forward were being made—a drastic strategic shift was needed for the bank to become more efficient. Although the many midrange systems had now to a substantial extent been migrated to the private cloud environment, a further reduction of complexity had to take place. Based on the experiences with the two public cloud platforms (AWS and Microsoft Azure), it was decided to use public cloud as a strategic platform next to the Mainframe that would remain to run many core systems. The choice was made to select one public cloud platform—Microsoft Azure—instead of using the two platforms available and to migrate all private cloud and AWS hosted workloads to Microsoft Azure (Rosa, J., & Dee, M. (2020)). This will result in the platform landscape as depicted in Fig. 2.

To achieve the IT transformation, a program organisation was put in place that had three main aims. Migrating all private cloud systems and AWS workloads to Microsoft was one of them. The other two pertained to the introduction and rollout of DevOps and the optimisation and consolidation of vendor relations.

5 Internal Audit Activities on Public Cloud

In this section, a description is given of the activities that enabled the internal audit department of ABN AMRO to opine on the public cloud environments. These activities were primarily focused on educating the auditors, creating the audit universe for cloud, and keeping a close eye on the transformation process.

5.1 Bringing the IT Auditors Up to Speed

One of the best practices of the IT Audit section of Group Audit ABN AMRO has always been the pro-active involvement in projects and programs. As Benjamin Franklin said: 'An ounce of prevention is worth a pound of cure'. This saying has driven IT auditors to get involved in programs/projects as early as possible to provide the program/project with audit feedback at a moment that fixing shortcomings is possible without affecting timelines and budget too much.

So, as soon as it became clear that IT was going to run a program aimed at exploring public cloud as a potential replacement or alternative for on-premise systems, the decision was made to dedicate one full time IT Auditor to the program. He had to get acquainted with the program, but also with the subject matter. A basic understanding of cloud computing was needed, so the Certified Cloud Security Professional (CCSP) course was done. Initially, the IT auditor closely monitored the program while gaining knowledge on cloud computing. One of the first things that needed his attention were the new cloud policies and standards that had to be put in place. In addition, the program was audited, covering both program governance and its deliverables. After that first period, the IT auditor made sure that public cloud was included in the multi-year audit plan and that specific audit activities were planned for the next year.

Gradually, the IT Audit department started to realise that public cloud was here to stay and would gain relevance in the years to come. With increasing cloud adoption, the audit workload would also increase. Sharing the acquired cloud knowledge and experience was needed for the other IT auditors to become proficient and remain relevant in an organisation with a substantial number of workloads running in the public cloud. A start was made by organising and providing internal training to the rest of the IT auditing community. This proved to be a very cost-effective method.

Given that CCSP certification is cloud-agnostic, the training material does not include the details that are specific for a cloud service provider. Once it became clear that the bank would be focused on Azure, the choice was made for the Azure Fundamentals course. As there was sufficient online training material on Azure, in-house training was needed, and staff was able to follow this course online at their own pace.

5.2 Audits Performed

The audits that were done on the implementation of public cloud, took place during three distinctive phases. The first phase was characterised by the ad hoc nature of the audits. Through continuous business monitoring, it became clear that the IT organisation intended to put in place two public cloud platforms. From the moment the two proofs of concept were started to experiment with Microsoft Azure as well as with

Amazon Web Services, until the time that the IT organisation decided that both Azure and AWS would be strategic platforms, the following audits were done:

- *Initial program and cloud platform set-up:* An audit was done from the very start of the two proofs of concept to ensure that no critical mistakes were made, and that cloud computing was being used in a controlled fashion. This included auditing the program organisation, but also whether the products being used were secure enough and compliant with the bank's policies and standards. One of the focus areas of the audit was the set of standards that was drawn up by the IT organisation to act as a basis for the configuration of cloud services and associated workloads. This all resulted in two audit reports: One for each platform.
- *Cloud Service Provider audit:* At the end of 2017, ABN AMRO Group Audit started taking part in the Collaborative Cloud Audit Group (CCAG) and carried out several pooled audits on cloud service providers as part of this group. As one of the early members of the CCAG, we have been actively involved in setting up, organising, and executing these audits. We have shared our knowledge, journey, and experience regarding CCAG in two articles mentioned under References (Pooled audits on cloud service providers—Parts 1 and 2).
- *Cloud Maturity Assessment:* The IT organisation found shortcomings in several areas that impeded an accelerated adoption of public cloud. Consequently, a high priority initiative was launched across the organisation to improve public cloud maturity. The audit aimed at assessing to what extent the initiative resulted in the required improvements to support a controlled acceleration in public cloud adoption. The areas covered included governance, security, architecture, operations, financials, cloud native development, and technical skills.

With the decision to use Microsoft Azure as a strategic platform next to the Mainframe, and to migrate all private cloud and AWS hosted workloads to Microsoft Azure, also the next audit phase started. This audit phase was characterised by the efforts to audit the IT transformation program organisation, the Azure migration organisation, and the Azure platform in a more detailed manner. This resulted in the following engagements:

- *Audit on the IT Transformation program:* This audit was aimed at assessing the readiness of the organisation for a large-scale DevOps implementation and migration to Azure. This audit covered the five principal areas of the program: (1) governance, (2) organisational design, (3) execution and migration organisation, (4) strategic sourcing, and (5) the Azure foundation design and delivery.
- *Azure migration organisation and deliverables:* The program responsible for the migrations of workloads to Microsoft Azure was audited to assess whether it could migrate existing workloads to the Azure cloud environment in a safe and prompt fashion.
- *Migration factory and tooling:* This audit covered the migration workflow 'factory' and associated tooling being used for the migration of workloads to ensure a standardised, controlled approach when migrations take place. Tooling was of special interest because of the prominent level of automation involved.

- *Cloud landing zone:* The cloud shell chosen as the private space of the bank was audited to ensure isolation from the public space, isolation between different workloads, and for the separation of development and production environments.

As more workloads had been migrated to Azure, the emphasis of the audit activities shifted to auditing applications running on Azure in addition to the ongoing audits on the Azure platform, now being a more balanced mix in the audit plan. This characterised the third and final audit phase. During this phase, the following audits were done:

- *Cloud platform products:* Platform products/services can be used as building blocks to set up the infrastructure for applications: e.g. Windows/Linux Virtual Machines, Storage Accounts, SQL databases. Using a risk-based approach we selected and audited the most critical components to ensure that these building blocks are designed and implemented correctly/securely.
- *Cloud security/directory services:* Our cloud platform relies on Azure AD for Identity and Access Management, and Azure Sentinel for security analytics and threat intelligence. We have performed audits on these crucial components as many products and services are depending on them.
- *Cloud applications:* Using a risk-based approach, we selected applications running on the Azure platform and performed an examination of all underlying cloud products and services to assess whether control processes were suitably designed and operating effectively.
- *Deployment pipelines:* By auditing pipelines we wanted to verify whether these essential components were (technically) sufficiently secured to ensure separation of environments and segregation of duties.
- *Software-As-A-Service (SAAS):* Next to the platform and application audits, we also performed audits on the usage of the riskiest SaaS applications. The focus here was primarily on the user-organisation controls and the integration with the banks internal processes (e.g. incident/problem/change management) and shared services (e.g. IAM, SIEM, CMDB).

A key element in all our internal engagements was the arrangement of read-only access to resources (e.g. products, groups, subscriptions) in the cloud environments. Our Infrastructure Managed Services department (refer to Sect. 6.2) created a new role for us with almost tenant-wide access (excluding the LogAnalytics workspaces) in Azure and as eligible role ‘security reader’ for AAD analysis. This provided us with uninterrupted access to (technical) audit evidence and streamlined our audit work in terms of efficiency.

6 Conceptual Framework

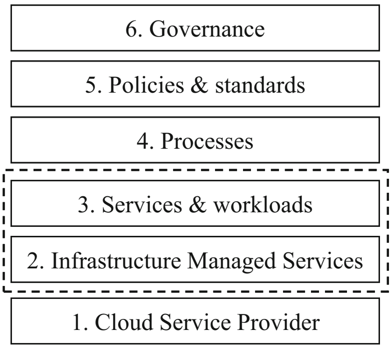
The audits discussed in Sect. 5.2 served as the inspiration for the design of the framework that we will present in this section. This conceptual framework can be used for planning and executing audits on public cloud implementations. While elaborating on the various components of this framework, we will refer to several products and services that are used by ABN AMRO and/or that are offered by Microsoft Azure. Figure 3 depicts the complete conceptual framework to audit public cloud implementations. However, most of the concepts that will be elaborated on will apply to other public cloud implementations as well. We will start our outline with the section on Cloud Service Providers (1. Cloud Service Provider) and work upwards to the Governance section (6. Governance).

The focus of our explanation will be on Infrastructure Managed Services (component 2 of the framework) and the Services and Workloads (component 3 of the framework). In our opinion, this has the most added value, given the fact that there currently is hardly any concrete guidance for IT auditors available on these two topics. We will supply both the contextual information and risk-control descriptions that will help IT auditors gain a better understanding of the subject matter and that will aid them in designing the audit programs they can use to audit public cloud implementations. As there are several publications and work programs that adequately cover the other components (Cloud Service Provider, Processes, Policies & Standards, and Governance), we will only explain what the specific attention points for these topics are in the context of public cloud. We will refer to relevant articles and audit programs when covering these topics.

6.1 Cloud Service Provider

Cloud service providers supply the basic services that their customers can use to build, run, and support their applications. These services pertain at the very least to

Fig. 3 Conceptual framework for auditing public cloud implementations



the physical data centres, physical networks, and physical hosts, on top of which virtualisation software runs. Customers could use a wide range of added services, depending on their specific needs.

To get assurance on the services that are outsourced to the CSP, there are three complementary approaches: The first approach is aimed at assessing the way the retained organisation manages the outsourcing arrangement. This is usually done by collecting information as to the outsourced services, inspecting this information to verify that performance is in line with expectations and contractual obligations, and finally to—whenever applicable—contact the CSP to request for corrections. The auditor will need to assess these processes to come to an overall conclusion on the level of control over the outsourced services. With the second approach, the retained organisation uses the available assurance reports and certifications made available by the CSP to get the assurance that is needed. The scope and applicability of the assurance reports will need to be assessed in addition to the proficiency of the external auditor, the quality of the report, etc. The third approach is aimed at carrying out audits at the CSP, possibly in collaboration with other clients. By carrying out these audits, auditors will be able to provide assurance on the scope of the audit. A combination of these three approaches is highly encouraged as they are complementary.

As regulatory guidelines (European Banking Authority, 2019; European Securities and Markets Authority, 2020) and other publications (e.g. Institute of Internal Auditors, 2018) are already available on how to audit outsourced activities or how to use a pooled audit approach to audit cloud service providers (Akdeniz et al., 2020; Bani Hashemi et al., 2020), we will refrain from elaborating on these topics in this section.

6.2 Infrastructure Managed Services

Although cloud service providers such as Microsoft Azure and Amazon Web Services offer many possibilities for DevOps teams to utilise ready to use (or customise) services, several aspects will be mostly the same for all DevOps teams within a company. For a start, there is one Active Directory for Identity Management for on-premise usage with centralised management and there will be only one enterprise Azure Active Directory (AAD) for usage in Azure. The same holds for Azure policy management: At the highest level (tenant management group), the Azure policies will be managed by a central department. At lower levels, DevOps teams may specify their own specific policies, if they do not contradict the central policies. At ABN AMRO, we decided to centralise functions that should be the same for all DevOps teams into Infrastructure Managed Services. These are:

1. Identity management
2. Policy management
3. Product development

4. Subscription management/Secure landing zones
5. Network management
6. Support for implementing security event monitoring

These functions will be discussed into more detail below.

6.2.1 Identity Management

Identity management (IdM), also known as identity and access management (IAM) ensures that only authorised people have access to the technology resources they need to perform their job functions. It includes policies and technologies that encompass an organisation-wide process to properly identify, authenticate, and authorise people, groups of people, or software applications through attributes including user access rights and restrictions based on their identities.[2] Most companies with a large IT landscape will recognise the need for central administration of identities and access rights: it makes it easier to block all access to systems at once of staff leaving the company and role-based access can be implemented across different IT systems. The ideal situation would be that all local user administrations of IT systems (such as e.g. Linux, Oracle and Windows Active Directory) and applications are/can be onboarded to central solutions like Ping Identity or SailPoint.

When an organisation starts using Azure services, it can only manage identities and access rights by using the Azure Active Directory (AAD) SaaS service. There are four offerings: Free, Office 365 apps, Premium P1, and Premium P2. The free version has an object limitation, and the Office version comes with added features to work with the functionality on the Microsoft collaboration platform. The premium editions offer more advanced access control capabilities and for heavily regulated industries like government and finance P2 is recommended. Obviously, making changes to the Azure AD directly/interactively must be restricted to a limited set of (tier 0) administrators. Analogous to Microsoft administrators who under specific conditions can make changes to production systems, they must use dedicated hardware to maintain the AAD. Complementary to their laptop for day-to-day tasks they use a separate specifically hardened device that can be used for AAD maintenance only.

As a detective measure, all actions on the AAD must be logged and monitored. We refer to the Monitoring section for further detail.

By using Azure in addition to your traditional IT landscape (hybrid situation), a new identity and access management system is introduced. Luckily, using the AAD Connect sync service, information held in the on-premises Active Directory (e.g. users/identities and user groups) can be synchronised towards Azure AD (one way), the central IAM system is so to say, ‘in the lead’.

[2] https://www.vmware.com/topics/glossary/content/identity-management.html

For a new application or supporting DevOps team in Azure, the following steps for onboarding are typical: A few generic (company/organisation) roles typically apply to almost every application and Azure service, such as administrator, developer, and operator. For each (new) team these roles can be defined centrally and the DevOps team members are added/assigned to the appropriate role(s) centrally as well. The identities of the team members can be added to the on-premises AD the moment they join the company. So only their group membership needs to be synchronised from the central IAM solution to the on-prem AD. Next, via AAD Connect, group memberships are synchronised to the Azure AD. The synchronisations are of course automated, i.e. they do not require administrator intervention.

When performing an audit, it is important to understand the structure of access rights assignments in Azure to identify anomalies or undesired implementations.

After the relevant AD groups have been defined, within Azure the access rights per (organisation) role need to be determined. An important feature of Azure is inheritance. Access rights in Azure can be granted at the following levels: management group, subscription, and resource group. They are in hierarchical order, which means that rights granted at management group level are inherited to all lower-level subscriptions and resource groups. Access rights granted at subscription level are inherited to all lower-level resource groups. When a subscription is shared between several teams, one would expect limitation of access to their specific resources/resource groups. When the subscription is for one team, then assignment of access rights at subscription level can be expected.

For a central team that manages Azure at enterprise/company level, access rights are expected to be granted/assigned at tenant/management group level. In practice, it is possible to have different management groups in a hierarchical relation and access right inheritance will follow that order.

At practically all levels combinations of built-in and custom Azure roles are to be expected. These Azure roles contain the permissions. Built-in roles like owner, contributor, reader, and user access administrator are widely used and set at management group level. On subscription level, the built-in role Support Request Contributor is assigned additionally.

In addition, custom roles can be built regarding e.g. role management, cost management, and policy management. These Azure roles are assigned at management group level to members of central teams managing Azure. Roles can be assigned to groups which in our case correspond to groups in the AAD and ultimately to the groups in the central identity and access management system. But Azure roles can also be assigned to applications and users which only exist in Azure and AAD. Now is a good moment to pause at the question which Azure roles should be assigned to which generic organisation roles. The actual assignments depend on the Azure service, but as a rule of thumb regarding the built-in roles the following can be configured (Table 6):

The owner role is very powerful and should therefore not be assigned permanently to members of the DevOps team. It is good practice to assign the owner role to a non-personal account like the service principal which is created (in our case) when the subscription or resource group is created. The service principal account is used

Table 6 Assignment of Azure built-in roles. Adopted from Microsoft (2022d)[a]

Azure built-in role	Permissions	Assign to
Owner	Full access to all resources	Service Principal
	Delegate access to others	
Contributor	Create and manage all types of Azure resources	Developer
	Cannot grant access to others	
Reader	View Azure resources	Central team and Auditor
User access administrator	Manage user access to Azure resources	Central team and Privileged Identity Manager (PIM)

[a] The table was extended with column 'Assign to'

by the CI/CD pipeline and used to deploy services and changes to the appropriate environments via a service connection. The contributor role is better suited to grant to developers on a permanent basis. The other roles are assigned outside the DevOps team. In roles mentioned above, no built-in roles are assigned to administrator and operator. In certain environments, administrators typically have the highest access rights, which in this case would be the built-in owner role. That is however undesired from a control perspective and in practice we do not expect to see many assignments of built-in roles to the generic administrator role. Only for a part of the Azure services does a built-in operator role apply (e.g. backup operator, Cosmos database operator, and site recovery operator). Obviously, when these services are not used, no assignment to the generic operator role is required.

The Azure AD P2 edition contains the Privileged Identity Manager (PIM) that can be used to assign/elevate privileges of Azure identities temporarily. For example, when the contributor role of a specific service is assigned to be eligible for a developer in the production environment, then he can only obtain that role after approval of a peer (i.e. someone in the group of identities that are also eligible for that role). PIM makes sure that the access rights are withdrawn after the pre-defined time window for usage has elapsed. The performed actions are logged in an activity log file.

There is more to say about non-personal accounts like the service principal account and managed identities, and their relation to Azure KeyVault and the fact that not all Azure services support Azure AD authentication. The auditor is recommended to be aware of the additional details while performing an audit. Microsoft does have online documentation that can be consulted.

6.2.2 Policy Management

Policy management is the process of creating, implementing/enforcing and maintaining policies within an organisation. Enterprise-wide policies could apply to all business processes, and some could be IT related. The IT-related policies can be generic in order to apply to different environments. However, regarding public cloud environments a dedicated framework can be useful where the company policies are translated into cloud security controls that are in principle cloud-agnostic. The challenge is how to enforce these controls?

Azure Policy is a service in Azure which allows organisations to create policy definitions which enforce and control the properties of a resource. Requests to create or update a resource are evaluated by Azure Policy (it is a little bit more complex). Each policy definition has a single effect (e.g. audit, deny, disabled). That effect determines what happens when the policy rule is evaluated to match.

Azure has many built-in policies that can be viewed via the Azure portal. Most are disabled or in audit mode by default. By putting them in deny mode, they are enforced. Additional considerations regarding migration to enforced policies will follow below.

Industry experts are divided on the topic of using built-in policies: the majority is not in favour of using built-in policies because Microsoft can change these definitions at any time, which can lead to operational problems. For example, when you enforce geo-redundant backups to be enabled and Microsoft changes the default to disabled, then from the time the change is active there will be no more backups. Therefore, companies that use built-in policy definitions, need to closely monitor policy definition changes made by the cloud service provider, and apply timely life cycle management and testing when changes occur to guarantee continuity of service delivery (or in this case to guarantee the ability to restore data when required). Given data privacy restrictions, an obvious built-in policy to use for EU based companies is Allowed Locations. You can restrict the locations to which resources are deployed to e.g. North Europe (Dublin) or West Europe (Amsterdam) by adding a custom policy. Policies can be assigned at distinct levels: management group, subscription, resource group, and individual resource.

Another attention point is the exceptional case that a specific resource is not available in the desired locations but still necessary for (specific) application teams. To enable this, the policy will typically not be enforced, giving all the other application teams within that subscription too many choices. As a compensating control an alert can be triggered when undesired locations are configured. The risk can be remediated by migrating the application requiring the geographic location to a separate subscription and enforcing the policy on management group level with an exemption for the separate subscription.

In the first stages of cloud adoption, it can be expected that by default practically all policies are in ‘audit’ mode, which means that they are evaluated, but not enforced. For the DevOps teams this may look convenient, but from control perspective it is far from ideal. Putting the policies in ‘deny’ mode would enforce them, but the policies out-of-the-box are not customised to the organisation’ needs. There are over 600 built-in policies in Azure. These are grouped into categories that are partly Azure service specific (e.g. compute, Cosmos DB, and Data Factory) and more generic (like general policies, tags policy, backup, and monitoring). The built-in policies are developed by Microsoft, based on their worldwide experience. But that does not mean that they are good enough or directly applicable for organisations.

Of course, Microsoft realised that and offers the possibility of defining custom policies.

In practice it is not unusual that Azure policies are maintained by different parties within the same organisation. For example, product teams (refer to next section) that are responsible for providing customised versions of the Azure services by using custom policies at resource level. And a central department that maintains the not-service specific policies like the General policies (including Allowed Locations) and the Tags policy. To distinguish the custom policies from the built-in policies a naming convention can help.

Most organisation will start in a situation where only a few policies are enforced. For reasons described in the Subscriptions/Secure landing zones paragraph below, gradually more policies can be enforced as the environment matures. It is hard to over-estimate the effort that is required to determine which Azure built-in policies are wanted/needed to be enforced. The cloud controls of our framework range from generic requirements regarding data leakage prevention to specific product/service settings regarding TLS. The built-in policies can be used to enforce part of the controls, but more than likely additional custom policies need to be designed.

Enforcing the policies is another step that should not be taken lightly: the impact will depend on the number of applications/subscriptions, and the maturity of the DevOps teams. When the applications remain in their same subscriptions/resource groups, then obviously a phased approach, starting in audit mode and resolving all non-compliance before turning to deny mode is the best practice. An alternative approach would be to migrate the applications to other/separate subscriptions instead. Organisations needs to consider whether the same policies should apply to the development and test environment as to the acceptance and production environments. And tempting as it may seem to apply a different set of policies to development and test, one should keep in mind that as of consequence the changes required before going to acceptance would be bigger. In our opinion it is better to apply the same policies, albeit that in development and test most policies remain in audit mode. Policy maintenance will be an ongoing effort since it is expected that new Azure services will become available in the future. When DevOps teams require new services, it needs to be determined whether policies need to be changed and/or new policies to be added.

An auditor would expect that all policies are enforced; however, this might not always prove to be practical. First probably not all built-in policies are necessarily relevant (and would hamper application execution when enforced) and second a balance should be made between security and risk appetite. Secure landing zones where more policies are enforced than in the shared subscription and the individual subscription model can help in striking the balance. By no means is reviewing the policies (e.g. what is (not) enforced) going to be an easy task for the auditor. However, he can benefit from using automation in this area, e.g. by using Azure Governance Visualizer. This is a is a PowerShell based script that iterates your Azure Tenant's Management Group hierarchy down to Subscription level. It captures most relevant Azure governance capabilities such as Azure Policy, RBAC and Blueprints and a lot more.[3]

Also important to consider is life cycle management per policy (incl. implementation and compliance) and a rationale for either enforcing the policy, or not. In the case of non-compliance, follow-up depends on the type of non-compliance. When policies that should be in deny mode do not apply or can be circumvented, this should be known to the DevOps team and preferably also with the central oversight department. In addition, there should be an approved policy deviation and a planning/path to compliance. Policy rationales can be reviewed by the auditor for plausibility with support from the DevOps teams and Azure experts. The roles and responsibilities of a likely to be implemented Azure Policy Board can be assessed as well taking into account its composition. Furthermore, the auditor can consider verifying whether the applicable company policies and derived cloud security controls have all been covered effectively by the enforced policies. When company policies and cloud security controls have not been (completely) covered by the policies enforced in Azure, then the gap needs to be determined and compensating controls need to be assessed.

6.2.3 Product Development

Cloud service providers manage a large set of services that cloud customers can deploy in their subscriptions. These services are cloud-based products that include compute, storage, networking, databases, development tools, and management tools. Product development is the process to customise native cloud services (by the cloud customer) to ensure that they meet the organisations security standards.

When an organisation starts its cloud journey with inexperienced DevOps teams, it can be considered necessary to protect the teams against themselves and let them use only customised/approved Azure services that could be deployed from a separate repository (the product catalogue), so not directly from Microsoft. Complaints from the DevOps teams are to be expected from this restrictive approach, as teams somehow always need ‘more exotic’ services. The customisation depends on the service. An easy-to-understand example is the requirement for TLS 1.2 for secure communication to services like Azure SQL server and Azure Data Factory. Another example is selection of encryption at rest for storage and databases like SQL and Cosmos DB.

The preference to protect the teams comes with a price. The Azure services of Microsoft are updated and patched regularly. Using a customised version means that the organisation will have to perform life cycle management and maybe patch management on these services itself. It is not unimaginable to have 3–4 versions per service in the product catalogue which all need to be maintained. So besides customising the services, it is important to manage timely upgrades.

[3] https://github.com/JulianHayward/Azure-MG-Sub-Governance-Reporting

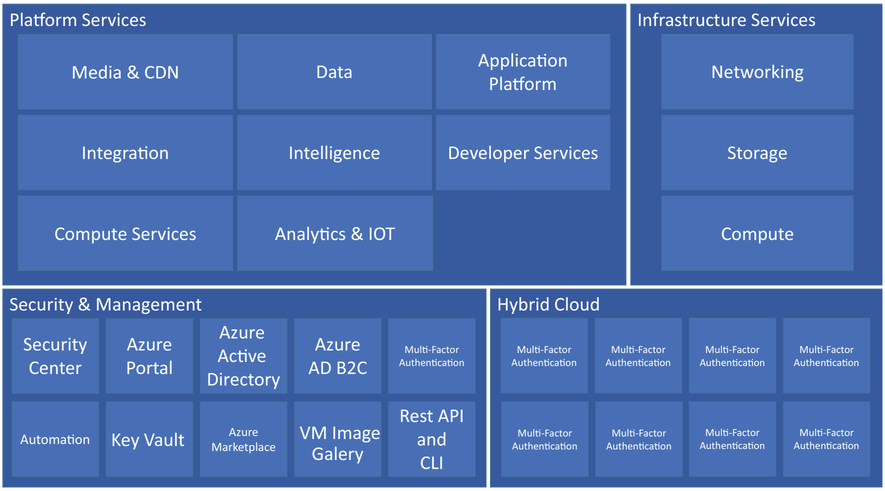


Fig. 4 Overview of Azure services according to Microsoft (n.d.)

Regarding the virtual machines (VMs) of Linux and Windows, an organisation can choose to implement/use CIS (globally recognised standard for secure baselines) hardened images. Attention point is the fact that Microsoft changes the VMs more frequent than CIS changes the hardened image. In other words, there is a delay in CIS hardened images becoming available. In order to keep up the pace with Microsoft, product groups can better build the images themselves, making sure they have the latest copies and all required patches. The images can then be replaced at a higher rate in the product catalogue. Mandatory automated update of deployed VMs should also be considered.

As soon as a DevOps team deploys a service (e.g. from the product catalogue shown below, Fig. 4) into their resource groups, it becomes their responsibility to maintain the lifecycle and to perform vulnerability and patch management, at least for the IaaS services. Of course, tooling (e.g. Microsoft Defender for cloud) is available in Azure to monitor resources, but teams need to be aware of their responsibilities first. When teams are accustomed to support traditional applications where development and operations responsibilities are split, teams do not automatically get the mindset required to maintain public cloud applications.

6.2.4 Subscription Management/Secure Landing Zones

With the on-premises data centres, developers had to request a server to deploy their applications to. Not long ago, these were physical machines, and the ordering process could take months. By keeping servers in stock, the process could be accelerated and by using virtualisation the process could be accelerated even further.

But still it would take several days to configure the (virtual) server before it was ready to use by the developers.

In cloud environments, subscriptions and the associated resource groups can be considered the equivalent of the physical environment. DevOps teams can deploy Azure services into resource groups which are logical containers. Resource groups are part of subscriptions which have limits or quotas on the number of resources you can create and use. Organisations can use subscriptions to manage costs. As part of subscription management, we consider the design and implementation of Azure management groups, subscriptions and resource groups and their (hierarchical) relations. The design is important because the hierarchical relations determine how certain characteristics are inherited. These characteristics include policies and access rights. For example, applying a certain policy at management group level that restricts the configuration of a service to a specific value will result in all underlying subscriptions and resources experiencing that same restriction.

Proper subscription/management group design can facilitate cloud adoption when it meets the organisation's requirements. The structure will depend on the nature of the organisation's activities, the geographical set-up, the types (and variety) of applications/workloads, the number of workloads/applications, etc. Azure management groups are designed to be flexible so they can be used to design a management group structure that reflects the expected organisational needs. The following example in Fig. 5 illustrates how different strategies of organising subscriptions can be combined:

A minimal design would be one management group under the root management group. Under the management group one shared subscription group is made for the developers and one subscription group for the Azure support team. Later a separate management group can be added for Information Security Officers and the Security Operations Centre where the activity log files can be stored and evaluated. The applications within the shared subscription could still be separated by using different resource groups.

When a DevOps team requests their first environment (e.g. development) in Azure, at least the following is required: one resource group, one service principal account, one AAD group (to add the DevOps team members to), a DevOps project, and a pipeline to deploy services/applications in the resource group. The pipeline is connected to the resource group via a service connection. Once teams need to be onboarded to the shared subscription, soon the subscription limit of 980 resource groups will be reached. Considering the environment types per application (development, test, acceptance, and production) only 245 applications can be hosted, which might be enough for small companies but certainly not enough for large companies. From cost management perspective, organisations may also want to implement more than one subscription. As time goes by, the DevOps teams became more experienced, and their demand to be more autonomous will increase. By monitoring adherence to Azure policies, the (central) department responsible for Azure policy management over time will gain more insight into which policies need to be enforced at which level. These developments can trigger organisations towards the decision to adopt secure landing zones: these are environments where

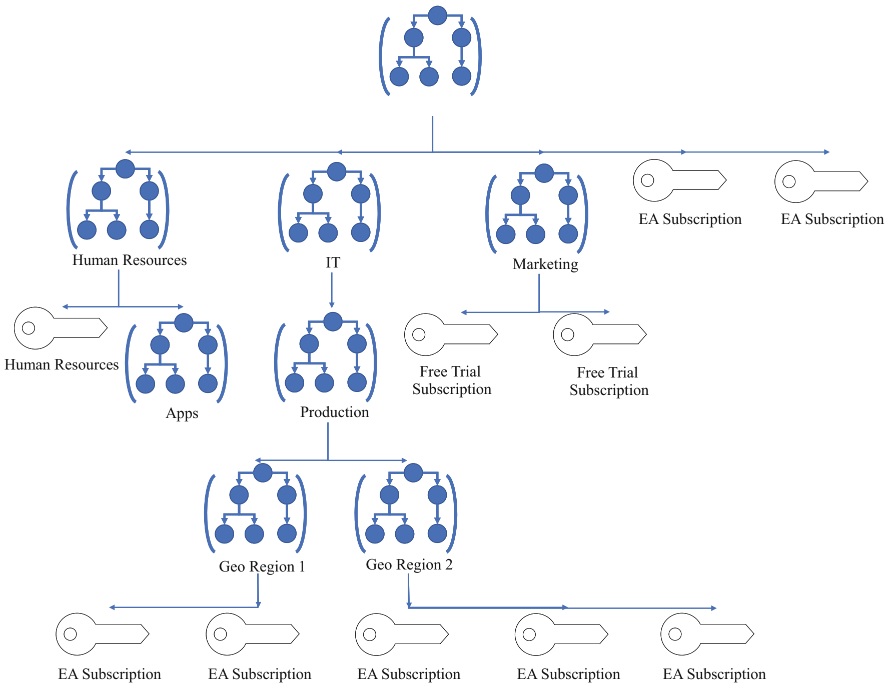


Fig. 5 Mix design strategy: departmental hierarchy, followed by geographic distinction for the Production department within IT, adopted from Microsoft (2022e)

non-customised services can be configured/deployed while the cloud controls/policies are determined at management group level (and then via inheritance will be enforced/in deny mode on lower levels).

To enable enforcement of different policies in different environments, an organisation can choose to implement a separate management group for the development and test environments and another for the acceptance and production environments. When an organisation introduces secure landing zones at a later phase (i.e. not from the start), it should account for future changes regarding management groups, subscriptions, and resource groups. Teams that already have applications in the shared subscription or in their individual subscriptions need to migrate their applications to the secure landing zone environments. Migration can only be done after all applicable policies have been implemented. Therefore, a planning for application migration needs to be made. Migration of applications to secure landing zones is not done overnight and organisation may want to avoid changes in the management group-subscription-resource group structure that require application migrations. To mitigate that risk, an IT auditor could review the design process and assess whether sufficient expertise was involved. Not only to speed up the subscription/resource group deployment process, but also to ensure that every DevOps team starts in the same position (with the same controls) the deployment process should be automated.

The IT auditor should verify that changes to the deployment process are detected and that appropriate follow-up actions are taken.

6.2.5 Network Management

Network management covers design, implementation, and maintenance of logical network configurations to enable secure communication. Connectivity, in general the ability to communicate between two points, is provided by networks. In addition, network components provide many more services like IP address resolving by Domain Name System (DNS), DDoS protection, load balancing, filtering by firewalls, intrusion detection/prevention (not necessarily all provided by Azure). From a different perspective one could say that networks provide isolation. Within Azure isolation is provided on many levels, starting with/at the tenant level. When an Azure subscription has been agreed upon and a customer/billing relationship has been established, then this Azure subscription is associated with one unique Azure Active Directory (Azure AD). By using a dedicated instance of the Azure AD tenant isolation is established; members of other tenants do not have access unless the tenant admin grants it through federation or by provisioning user accounts. Between the DevOps team subscriptions or secure landing zones there is also isolation and each must have their own dedicated range of (internal) IP addresses. Within the subscriptions virtual networks can be defined and finally there is isolation at Azure PaaS services level by firewall rules.

Typically network management responsibilities are divided into two parts. Centrally managed (in the management subscription) is the infrastructure that is shared, like circuits connecting Azure with the ABN AMRO Data Centres, more in specific Azure ExpressRoute. Also Azure Firewall and the DNS are centrally managed. Lastly, the assignment of IP-address ranges (IPAM: IP-Address Management) is centrally done. The DevOps teams usually manage local network configurations in their subscriptions like PaaS firewalls (e.g. in Azure SQL server or Azure Data Factory) and local virtual networks (VNets). Depending on the services that are used within a subscription, a VNet is required. VNets are required for among others virtual machines (VM), VM scale sets, and Kubernetes. A network security framework (NSF) that describes which traffic flows require which network controls can help organisations secure their (Azure) networks. The NSF model could utilise data classification regarding confidentiality and integrity. Roughly speaking, the higher the confidentiality and/or integrity rating, the more network controls need to be applied on specific dataflows.

To access confidential data (like account balance), besides Multi-Factor Authentication (MFA) measures, also the following measures are required: encryption, specific firewall rules, logging and monitoring, intrusion detection, data leakage prevention, etc. To access public data from the website, only a subset of these measures is required.

From an auditor's perspective, the possibility of DevOps teams to configure public endpoints and to allow access from public internet deserves attention. Besides authentication, access control lists should be implemented to restrict access appropriately. In addition, adherence to the NSF needs to be monitored and non-compliance requires follow-up.

On some Azure services a parameter can be specified that enables bypassing all network separation rules. In practice, setting this parameter allows other tenants of the Azure cloud to access these services regardless of other firewall rules or deny public internet access settings. The parameter is 'allow Azure services' and can be specified on services like Azure SQL Server, Azure Data Factory, and Azure Synapse. The parameter is a feature and at the same time a huge security risk because it allows all users of Azure to attempt to access your service and data within that service. Valid credentials (user id and password) are the only thing that now stand between the other tenant/hacker and your data. When the parameter is misused in the development environment to circumvent connectivity problems, it will do the same in the production environment. Of course, this is highly undesired and should be prevented. Auditors should be aware of these types of parameters and verify whether the organisation has processes in place to identify and mitigate these types of vulnerabilities.

6.2.6 Security Event Monitoring

In a world of cyber threats, it is extremely important to detect/monitor events that could indicate compromise. In addition, in a highly regulated industry it is important to demonstrate compliance with regulatory demands regarding e.g. highly privileged access. In both cases, you need to record/log certain events, process/evaluate them, and take appropriate action. This is what we call security event monitoring. In principle, the DevOps teams are responsible also for arranging security monitoring, but identifying security relevant events and turning them into analytic rules in Sentinel requires a certain risk mindset/type of security awareness and specific skills that may not always be present in DevOps teams. Besides, as many teams use the same Azure services, they are likely to run the same risks. A central application security monitoring team can prove to be beneficial in developing several generic use cases that are likely to be expanded as more workloads are hosted in the cloud environment.

With powerful resources like activity logs from Azure services and tools like LogAnalytics and Azure Sentinel, one would expect the capability to correlate and monitor almost anything. And that may be true, however not out-of-the-box. With specific logging and monitoring requirements, additional measures need to be taken in order to fulfil those requirements. Logging of events may be required to fulfil regulatory requirements; however, the focus should be on events that threaten business processes, operations, and confidentiality and integrity of data. It may also be necessary to distinguish between changes made via pipelines/Everything as Code versus manual changes in troubleshoot situations.

In order to prevent alerting everything and flooding the Security Operations Centre (SOC) with false positives, in advance the organisation needs to determine exactly what are the activities/events that need to be known and that require follow-up actions. These events may be generic (like disabling MFA for a user, elevation of privileges, making policy adjustments) and service specific (e.g. changing the access to AKV or changing TLS minimum level on SQL server). The set of events to monitor may grow over time, based on experience/new insights.

In our experience, identification of events to monitor is the trickiest part because maintenance and support staff are quite hesitant to identify security events in advance. Elevation of privileges by using Azure services like PIM is quite easy to monitor. Regarding subsequent actions, it is much harder to identify which pose a threat. Two arguments are often heard:

1. We do not know in advance which actions will be performed (using high privileges).
2. An action/event can in one situation/subscription be valid and required (e.g. viewing and updating data or configuration settings) and in another situation/subscription unauthorised. How to differentiate between those two?

The difference may be whether an incident was reported via another channel: when the changes made concern the incident, then they are probably fine. When there is no incident, then further investigation may be required.

Focusing on the riskiest events is a sensible approach. A few events may be identified in advance (like the ones mentioned above), while the rest may be based on new insights. This path is however still uncertain because it depends on vigilance of the maintenance staff to detect out of the ordinary actions. And it may not be the best way forward: e.g. for a single event it cannot be decided without additional information whether it was performed with malicious intent. From the on-premises IT landscape we already know that correlation of events is important to recognise patterns and compare them with attack tree scenarios. Within Azure, machine learning may be able to fulfil these requirements, an area definitely worth experimenting.

Once a security relevant event has been identified, it has to be figured out into which activity/log file (or a combination of log files) the event is recorded and whether the recorded data is sufficient to generate a useful/actionable alert. Next, a so-called analytic rule has to be set in Sentinel and the follow-up action needs to be determined/specified. Let us say an alert has to be sent to the SOC. The staff at SOC need to have instructions how to act on different alerts. Probably not only the SOC needs to be alerted, but also the product/application owner needs to be informed. For analysis of trends, the alerts may be aggregated.

From an audit perspective, we would expect that every Sentinel analytic rule has an owner and that a life cycle process applies to them all. As security events should occur exceptionally, their relevance needs to be determined periodically. The rules must have a documented rationale and follow-up actions must have been described. False positives need to be eliminated or at least minimised during development. The use of Sentinel comes with a bonus (in addition to machine learning): the MITRE ATT&CK framework is used within Azure Sentinel to help classify threats to the organisation and to provide quicker understanding of the level where intrusion exists. The MITRE ATT&CK framework is a curated knowledge base and model for cyber adversary behaviour, reflecting the various phases of an adversary's attack lifecycle and the platforms they are known to target. Being able to classify threat events into the framework is a major step in demonstrating coverage and control.

6.2.7 Summary of Key Risks and Controls for Infrastructure Managed Services

In Table 7 below the key risks and controls for Infrastructure Managed Services have been summarised.

6.3 Services and Workloads

Next to the centralised functions, generic services and boundaries described in the previous section (i.e. Landing zone), DevOps teams need to set up and maintain specific Azure services that provide compute, network, and storage functionality to host the actual workloads (i.e. business applications).

Azure provides more than 200 services, which are divided into 21 categories. These categories include computing, networking, storage, IoT, migration, mobile, analytics, containers, artificial intelligence, and other machine learning, integration, management tools, developer tools, security, databases, DevOps, media identity, and web services.[4] Via the Azure portal DevOps teams can use these services to create cloud-based resources, such as virtual machines (VM) and databases for their workloads. Depending on the services being used, controls need to be implemented to adopt and use these services securely. In this section, the key control domains are described that are applicable to all consumable services. Microsoft has extensive online documentation that can be reviewed for the specifics of each service.

In this section, we will refrain from giving guidance on the audit of functional application controls (e.g. input/processing/output controls) as the audit of these functional application controls is only marginally different from the audit of these in an on-premises environment. Existing literature can be reviewed on this subject.

6.3.1 Network Configuration & Management

Network configuration is the process of setting policies, flows, and controls for an organisation's network infrastructure. It's a critical step to ensure that the application network works properly and stays secure. Within Azure it's important to create and maintain network segmentation, control inbound/outbound communication, control

[4] https://azure.microsoft.com/en-us/services/

Table 7 Risks and controls for Infrastructure Managed Services

Risks	Controls
6.2.1 Identity Management	
Unauthorised changes to AAD, product catalogue (may be ABN AMRO specific), policies, central network configuration	The Azure Ownership role should only be assigned to non-personal accounts (like SPN) and temporarily to emergency/troubleshoot groups
	Azure roles are not granted (directly) to personal user accounts
Abuse of privileged access	Access rights can only be elevated using PIM, requiring approval by peers
	Logging and monitoring of changes and timely follow-up on incidents
Access cannot be denied timely	Only groups that correspond to groups in central IAM solution may be used
Creation of local accounts (not known to AAD) with weak authentication measures	Detection of these accounts can be implemented by a combination of policies and logging and monitoring measures
6.2.2 Policy Management	
Azure built-in policies are changed by Microsoft	Monitor changes made by the CSP, assess the impact of the change, and take corrective action when required
Enforced Azure policies do not meet company rules (or are inconsistent)	Life cycle management process is implemented, rationale needs to be plausible and supported by experts, policies are approved by policy board, policies are tested before enforcing them, the complete set of policies is evaluated periodically
Enforced Azure policies do not cover all threats	Coverage is monitored by policy board or equivalent
Deployed services do not fully comply to Azure policies	Central oversight function to monitor non-compliance and follow-up
6.2.3 Product Development	
Product catalogue contains old versions (that have vulnerabilities)	Life cycle management of products to ensure that only most recent versions are available/deployable
Product catalogue does not include all services required by DevOps teams	Keep a backlog and prioritisation mechanism
Baseline not defined	Central oversight function to monitor baseline adoption and adherence
Hardening reduction	Central oversight function to monitor baseline adoption and adherence
Lack of vulnerability management	Central oversight function to monitor vulnerability scan execution and vulnerability remediation
Lack of patch management	Central oversight function to monitor unpatched services/software and severity levels; escalate when remediation measures are delayed

(continued)

Table 7 (continued)

Risks	Controls
Product descriptions obscure, incomplete, and outdated	Periodically verify with consumers/DevOps teams whether the descriptions are comprehensible and adequate
Lack of scalability/products not timely available or incompletely customised	Extend the product development teams or refrain from customisation and compensate by means of policies (which also require maintenance)
6.2.4 Subscriptions/secure landing zones	
Structure of management groups, subscriptions, and resource groups does not fit business requirements (causing costly migrations when the structure changes)	Design review by experts, test the design in separate environment using a pre-defined/agreed upon requirement list
Deployment of subscriptions is not timely and repeatable (leading to different starting positions)	Automate and monitor subscription deployment
Unauthorised changes to deployment process (leading to subscriptions with e.g. less or no controls)	Monitor changes to automated deployment process
6.2.5 Network configuration	
Usage of public endpoints and allowing access from public internet/networks (applies to several Azure services)	Central oversight function to monitor and verify whether compensating controls have been implemented
Allow Azure services = yes (applies to several Azure services)	Implement custom policy
Measures do not correspond to data classification	Monitor compliance to network security framework and take appropriate action regarding non-compliance
6.2.6 Security event monitoring	
Not all security events are (timely) identified/lack of insight in coverage of use cases	Central monitoring of security event identification, supported by the MITRE ATT&CK framework mapping in Azure sentinel to periodically assess coverage
Not all services/components are monitored with developed/applicable use cases	Central monitoring of appropriate activity logs being loaded/processed
Inadequate follow-up actions defined	Life cycle management of rules/use cases—periodically review follow-up of alerts
Rules are outdated or will never trigger an alert	Life cycle management of rules/use cases—periodical review to verify effectiveness

communications between Azure resources, route and filter network traffic. Moreover, not only should the (virtual) network be well-architected, it should also adhere to well-established principles such as layering and tiering. Each Azure service has networking configuration items (e.g. VNETs, subnets, Firewalls, IP addresses) that should be taken into account as part of securing the network.

It is important to note that there is a difference between the overall network perimeter (i.e. Landing zone, discussed in Sect. 6.2.4), and the specific network configuration of a certain application. Auditors need to take into account both configurations, specifically for the application configuration careful attention needs to be given in the network rules and settings: do these adhere to standards, are these sufficiently hardened, and periodically reviewed. Moreover, it needs to be checked where sufficient isolation and tiering is in place between applications (i.e. sound architecture): point-to-point connections should receive extra attention on this matter in terms of potential security vulnerabilities. An example to consider is Network Security Groups which in essence are a basic, stateful, packet filtering firewall, that controls access based on the configuration of source IP address/port number, destination IP address/port number, and the protocol in use. Just as important is the implementation and configuration of Azure Firewall which is a fully stateful firewall service with built-in high availability and unrestricted cloud scalability. There are a lot more network measures that can be implemented depending on the requirements of the environment. It is important that the auditor first understands the network design (e.g. via documentation and flow-diagrams) and implementation and whether this is fit-for-purpose. The next step would be to check and verify each measure and solution.

6.3.2 Identity & Access Management

Identity & Access Management (IAM) ensures that the right users have the appropriate access to Azure services and resources. Azure has many capabilities that can help secure IAM, such as: Single sign-on, Multi-Factor Authentication (MFA), Azure role-based access control (Azure RBAC), Security monitoring, alerts, and machine learning-based reports, Privileged identity management, Identity protection, etc. (Microsoft, 2022d). Every service on Azure makes use of an identity alongside certain privileges that needs to be controlled.

The auditor should keep in mind that next to the centrally managed identities and accounts (i.e. Landing zone), certain Azure services and applications have their built-in accounts and identities. Similar to traditional audits, (non-personal)/(privileged) accounts should be reviewed and checked by the auditor against the principle of least privilege, adherence to periodic access reviews, and the implementation of strong authentication (e.g. enablement of MFA).

There are four fundamental built-in roles within Azure (Azure RBAC): Owner (full access to all resources), Contributor (create and change resources but can’t grant access to others), Reader (read/view only), User Access Administrator (manages user access to Azure resources). The auditor needs to understand the use of each of these roles for the specific application and determine whether its use is controlled and appropriate. Another point of attention for the auditor could be the reports about administrator access history and changes in administrator assignments. The auditor can make use of a variety of reports within Azure to gain insight into the controls around IAM and how the organisation is operating: e.g. via sign-in anomaly reports, user-specific reports which display device sign-in activity data for a specific user, activity logs containing audited events within certain timeframes (24 h, 7 or 30 days, etc.).

6.3.3 Resource Security

Azure services need to be secured just like any other resource. Depending on the type of service being consumed (e.g. IAAS/PAAS/SAAS), patching needs to be performed and endpoint protection (e.g. virus/malware protection) should be in place. Additional security measures include disk encryption, secure data transfer between resources, and adequate key management.

For the auditor it is important to note that the burden of maintaining resource security by the IT organisation is the most for IAAS (e.g. managing *all of the resources* within the Virtual Machine). For PAAS certain resources are taken care for by the cloud service provider and for SAAS this part is less applicable as the CSP is typically fully responsible for resource security. In the case of IAAS, the auditor should consider auditing the whole VM and all of its contents (as this is not managed by cloud service provider), this means general IT controls testing on the Operating System, Middleware, and database as all of these components are managed by the IT organisation. Key controls include: change management, lifecycle management, patch management, vulnerability management, system hardening management, etc.

For PAAS, the auditor needs to understand the PAAS-components that are managed by the IT organisation, typically this translates to configuration settings on networking (e.g. which components are allowed to communicate with each other?), admin access (e.g. who, what, when, and which conditions apply?), and hardening (e.g. legacy/weak protocols allowed?).

Depending on the Azure configuration policies set throughout the organisation, the auditor needs to perform more or in-depth testing of controls. This means that in the case that Azure policies are not globally applicable and enforced with no override possibility, the auditor needs to consider testing each Azure resource (e.g. product/service) relevant to a certain application as this could potentially deviate from security best practices. As mentioned in the previous section, DevOps teams may enjoy a certain degree of autonomy and freedom within their specific block and subscription which allows them to have less than optimal implementations.

6.3.4 Logging and Monitoring

Logs are event records where events related to the state of a specific Azure service are collected. There are a multitude of logs (e.g. performance, integrity, availability) for different Azure services. Selecting useful information to store and archive is key here: selecting metrics, rules, classification of alerts *for each service*. It's also important to ensure the security and confidentiality of stored logs, and control the quality of log data by analysing and adding missing information to logs.

Monitoring is also important to detect any lack of service performance and to detect attacks in real time. In order to detect these anomalies, Azure provides centralised supervision tooling to aggregate the different logs and to enable real-time monitoring (e.g. Microsoft Sentinel, Defender for Cloud, Azure Monitor, etc.). Of course, each service needs to be connected and configured to use the centralised tooling, and the tooling itself needs configuration and maintenance as well.

Although certain monitoring can be arranged centrally (refer to previous section), for each application and set of resources managed by DevOps teams, certain events can be logged and monitored. It is important that the auditor keeps in mind that both dimensions should be taken into account. For example, flow logs can provide insight in network traffic patterns. There are roughly three categories of logs within Azure: Control/management logs (e.g. create/update/delete of azure resources), Data plane logs (e.g. events raised as part of Azure resource usage for example via Windows event system, security, and application logs in a virtual machine), and Processed event logs (e.g. provide information about analysed events/alerts, examples of this type are Microsoft Defender for Cloud alerts). Finally, the auditor should also take into account that all of these logs need to be monitored in some shape or form via the monitoring solution. Next to performance, attention areas for the monitoring solution include the security posture of virtual machines, networks, storage and data services, and applications to discover and prioritise potential security issues. Azure provides extensive logging and monitoring capabilities for DevOps teams that can be utilised for each application and the resources involved.

Microsoft's Defender for Cloud continually assesses Azure resources for security issues and presents the results on a dashboard in the Azure portal. The recommendations vary from low to high severity and could be grouped into categories like: System updates should be installed, Log Analytics agent should be installed on virtual machine scale sets, Vulnerability assessment should be enabled on SQL servers, Authorised IP ranges should be defined on Kubernetes Services, etc. When security issues have been identified, Defender for Cloud gives recommendations how to improve and remediate these issues. IT auditors should be careful in interpreting these issues, especially concerning their validity: the tool makes no difference whether resources in a development or production environment are assessed, but for the risk profile this makes a difference. Furthermore, the tool may not 'see' compensating controls that are not based on Azure services/features, e.g. using Splunk instead of LogAnalytics or using DDoS protection from third parties. No doubt Defender for Cloud provides added value by identifying weaknesses in configurations, but the recommendations should be regarded with due caution. In addition to security issues, Defender for Cloud also provides statistics on regulatory compliance like ISO27001 and PCI/DSS. From the auditor's perspective it is worthwhile to retrieve this information to determine how DevOps teams are managing the environment.

Azure Monitor provides a comprehensive solution for collecting, analysing, and acting on telemetry from cloud environments which gives several 'insights' or views on the resources (metrics) on the one hand and operational alerts and access to LogAnalytics on the other hand. These insights regard applications, but also VMs, containers, network, storage accounts, and a few others and can be tailored to specific needs. The DevOps teams need to determine which events require operational monitoring and how to respond to alerts and incidents. Just like vulnerability management operational monitoring may not be obvious to all DevOps teams. Especially when availability requirements are 7 × 24. A word of warning seems applicable when using Azure Monitor. Performance problems caused by badly written queries, or not timely reorganised (SQL) database indexes may be obscured or compensated by scalability measurements. Due to lack of production workload limitations, performance problems may not always directly surface. Also, performance problems can originate from Microsoft incidents as well. In February 2022, performance problems were encountered in Europe with the Azure DevOps service: Boards, Repos, Pipelines, and Test Plans were all affected.

From an audit perspective, availability of applications/business functionality is one of the key aspects. Typically, Azure Monitor is restricted to the Azure cloud environment and is therefore not implemented as an end-to-end monitoring solution. Therefore, additional measures should also be taken into account by the auditor. Depending on the application functionality and the Azure components used, a sensible selection of parameters to monitor have to be made. DevOps teams must be able to demonstrate their monitoring controls and explain their selection parameters.

6.3.5 Security Incident Response

Security Incident Response is about developing and implementing an incident response infrastructure (e.g. plans, defined roles, training, communications, management oversight) for quickly discovering an attack and then effectively containing the damage, eradicating the attacker's presence, and restoring the integrity of the network and systems.

It is important again to distinguish centralised operations (landing zone-level) as well as decentralised operations (application-level). The auditor needs to be aware that at both levels, runbooks need to be developed and periodically tested. Moreover, it is key to verify whether the involved teams have the right capabilities to handle incidents and how well the communication takes place between and across teams. Existing literature on this topic should provide sufficient guidance on how to assess this process.

6.3.6 Data Encryption

The main areas of encryption include encryption of data-at-rest, data-in-transit, and key management with Azure Key Vault. Data encryption at rest is available for most of the Azure services including file, disk, blob, and table storage. Microsoft also provides encryption to protect Azure SQL Database, Azure Cosmos DB, and Azure

Data Lake. The auditor should be aware of this option and depending of the services being used verify if this encryption is actually enabled.

Another point of attention is that Azure supports various encryption models, including server-side encryption that uses service-managed keys, customer-managed keys in Key Vault, or customer-managed keys on customer-controlled hardware. With client-side encryption, the customer manages and store keys on-premises or in another secure location (encryption performed outside of Azure). The three server-side encryption models offer different key management characteristics that the auditor should be aware of in order to assess the appropriateness of the implementation being used:

1. Service-managed keys (combination of control and convenience with low overhead).
2. Customer-managed keys (gives customer control over the keys, incl. Bring Your Own Keys (BYOK) support, or allows you to generate new ones).
3. Service-managed keys in customer-controlled hardware (customer manages keys in their repository, outside of Microsoft control, configuration is complex and most Azure services don't support this model).

The auditor should pay close attention to the Key Management process and Key Storage solution (e.g. Hardware Security Module). There are different options available for Key Storage and each solution has its certain pros and cons depending on the requirements of the organisation. Key requirements to check are tenancy (multi or single), integration possibilities (SAAS/PAAS/IAAS), supported key operations (public/private; key-lengths; ciphers), scalability/availability, FIPS-140 level support and certification, level of control over keys (full/partial/none) and compliance with regulations, and operational responsibilities (backup/restore, patching, upgrades, etc.).

Data-in-transit can be secured via various ways, some examples that the auditor could verify are whether the site-to-site VPNs are properly set up, SSH and RDP sessions are set up to use protocol encryption, REST API calls make use of HTTPS, and whether the TLS protocol is used to protect data between services.

6.3.7 Business Continuity and Disaster Recovery (BCDR)

Two factors are especially important for the resilience of an application: its availability (the proportion of time the application is functional) and recoverability (the ability to recover from failures). Although availability of Azure services is guaranteed for up to 99.95%, things can and will go wrong. The high availability of Azure services does not dismiss organisations of the responsibility to take measures to guarantee that applications (which most likely are supported by a combination of services) and data are safeguarded from outages. The measures consist either of implementing redundancy or the ability to quickly recover.

Azure services run on servers in datacentres across the globe. These datacentres are grouped into availability zones, and availability zones are grouped into regions

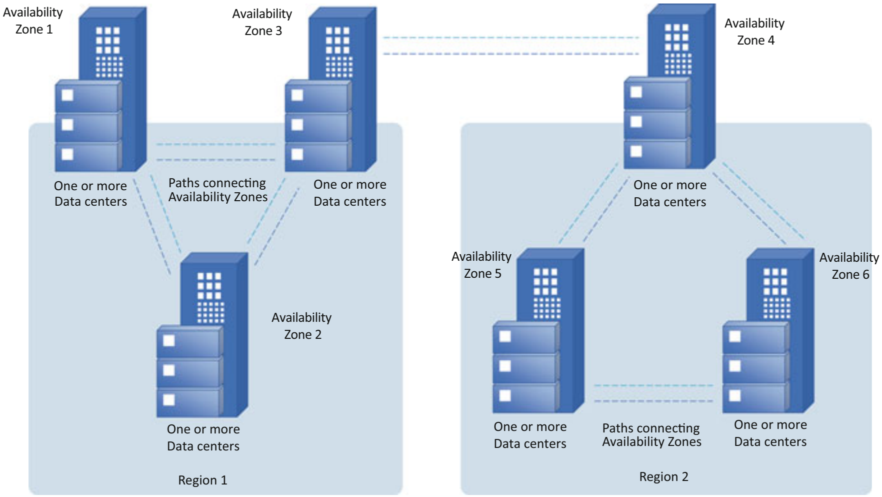


Fig. 6 Azure data centres, availability zones, and regions according to Microsoft (2022b)

such as North and West Europe. The datacentres are connected through a dedicated regional low-latency network. Azure Availability Zones are physically separate locations within each Azure Region that are tolerant to local failures. Failures can range from software and hardware failures to events such as earthquakes, floods, and fires. Tolerance to failures is achieved because of redundancy and logical isolation of Azure services. To ensure resiliency, a minimum of three separate availability zones are present in all availability zone-enabled regions. This design per region is outlined in Fig. 6.

Resilient solutions can be designed by using Azure services that use availability zones. The services can be divided into zonal, zone-redundant, and always-available services. The zonal services can be deployed to a specific, self-selected availability zone to achieve more stringent latency or performance requirements. Examples are Azure Backup, Azure Site Recovery, and Azure Virtual Machines. Resiliency is self-architected by replicating applications and data to one or more availability zones within the region.

With zone-redundant services, resources are replicated or distributed across zones automatically. For example, zone-redundant services replicate the data across three zones so that a failure in one zone doesn't affect the high availability of the data. Examples of services are Azure SQL, Azure Storage Account, Azure KeyVault, and Azure Data Factory.

Always-available services are always available across all Azure geographies and are resilient to zone-wide outages and region-wide outages. Examples are Azure Active Directory, Azure Policy, and Azure Portal. Always-available should be taken with a grain of salt, because in 2020 and 2021 Microsoft experienced several Azure AD outages.

Table 8 Risks and controls services and workloads[a]

Risks	Controls
Inappropriate access to data	– Identity & Access Management (e.g. multi-factor authentication, access reviews, segregation of duties, etc.)
	– Network configuration and management (e.g. VPNs, network segmentation, firewalls, etc.)
Exposure of confidential data in-transit and at-rest	– Secure key storage, adequate key management processes
Inability to mitigate and/or recover from data loss/exposure/manipulation	– Logging and monitoring (e.g. key events, alerts, follow-up procedures)
	– Security incident response (e.g. training, testing, documentation)
Compromised integrity of resource	– Resource security (e.g. patching, endpoint protections)
Unavailability of data and systems	– Appropriate zoning of data for compute and storage activities
Incomplete/inaccurate/invalid records	– Application controls e.g. (input/output/processing controls)

[a] This table is a high-level summary, refer to the above paragraphs for the key differentiating aspects related to the controls

It is evident that when an application consists of several Azure services, it is not so easy to achieve RTO and RPO values on the application level. When availability and performance are not critical, these measures probably are sufficient. However, for applications that are critical, e.g. financial transaction processing, performance objectives, and RPO = 0 will be difficult to be met and need careful consideration.

6.3.8 Summary Key Risks and Controls for Services and Workloads

In Table 8 below, the key risks and controls for services and workloads have been summarised.

6.4 Processes

Most processes that will be in scope of audits that relate to traditional (non-cloud) IT environments are also relevant in a public cloud context. Change Management, Problem Management, Deployment Management, and Capacity Management are just a few examples. As there are many sets of best practices, guidelines, and audit programs available for IT auditors to audit IT-related processes, we will refrain from covering this here extensively. However, there are a few subjects that are worth mentioning as they require specific attention from IT auditors.

Configuration management and the configuration management database (CMDB) that are at the core of most service management processes, must also cover the cloud environment. Usually, the public cloud in question will be able to deliver details on the configuration items, tags and dependencies by using the standard configuration management facilities offered by the CSP (e.g. Resource Graph, Azure Service Map, and Azure Application Map). Configuration management data can then be extracted and synchronised with the CMDB. In many cases, it will be possible to use native integration facilities, but. . .the configuration management data needs to fit the CMDB data model and that will usually require a lot of effort to make all the necessary adjustments. One of the recommendations therefore is to use the CMDB as the main tool for cross platform insights, but to use the CSP CMDB as centre of truth for deployed resources instead of fully syncing with the CMDB.

It is not the intention to cover the DevOps way of working as it represents a way of working that is closely related to public cloud, but it is a completely different subject nonetheless. However, there are two DevOps subjects that every IT auditor must be aware of when auditing a public cloud implementation in a DevOps context, as they will significantly impact the level of control over public cloud implementations.

The first subject pertains to the high level of autonomy and freedom of the DevOps teams that ideally work in a self-service model in the public cloud. This means that every process might be executed differently by different DevOps teams. This does raise the question: How does the organisation ensure that the DevOps teams work within the boundaries as set out in the corporate policies and standards. Based on our experiences, this can only be achieved if three interrelated conditions are met: First, there needs to be a set of goals for the DevOps teams that strikes the right balance between run and change responsibilities.[5] Second, (senior) management must direct and redirect the DevOps teams based on the actual performance on these balanced goals. This will support the culture that is needed to stimulate the right behaviour, i.e. that DevOps team members do not favour Development over Ops activities or vice versa. This also implies that senior management needs to be committed to these goals themselves. Although this sounds obvious, given the fact that many DevOps teams and their senior managers will have their roots in the Software Development area and that they are under pressure to deliver functionality for their (business) product owners, there is a risk that development activities get priority over operations and support activities. And third, this system can only work if the right management information regarding the goals and boundaries is available. This will require using reporting tools to frequently give insight in process performance for all processes and for all DevOps teams.

[5] According to the DevOps Research Assessment report ‘State of DevOps 2021’ by Google Cloud, there are four metrics of software delivery performance that can be considered in terms of throughput and stability. These metrics are the lead time of code changes (that is, time from code commit to release in production), the deployment frequency, the time to restore a service after an incident, and the change failure rate. According to that report, high performers score consistently higher on all four metrics.

The second item pertains to the fact that relying on the formal handover procedure between change and run teams is no longer possible when working in a public cloud/ DevOps context. The so-called segregation of duties mechanism that relies on conflicting interests between run and change teams no longer exists if run and change activities are carried out in the same team. And it turns out to be difficult to ensure that mitigating controls operate effectively, as the privileges of DevOps team members enable them to bypass many of the theoretical controls. For example, the correct use of automated CI/CD pipelines could enable automated security tests, unit tests but also deployments under dual control. This would mitigate the lack of segregation of duties. However, it is inherent to the DevOps way of working that team members can adjust pipeline code/building blocks. In other words: without additional measures, DevOps teams could turn off dual control as part of the deployment process, security testing as part of the development process, etc. This is something IT auditors should be aware of and must consider when auditing the chain of change-related processes.

6.5 Policies and Standards

Policies hold sets of formalised rules, principles, and minimum control requirements that must be in place to direct behaviour, actions, and decisions in an organisation. Policies are generally based on laws and regulations or added requirements the organisation may be subject to or may subject itself to. They will generally be set in line with the organisation's risk appetite.

Standards are an extension to one or more specific policies and must always be consistent with these policies. Standards are used to describe detailed mandatory requirements, criteria, calculations, or methodologies associated with the implementation, enforcement, and support of the policies.

When auditing public cloud implementations, it is therefore necessary to assess the coverage and quality of the policies and standards that pertain to public cloud. Depending on the organisation's preferences, there might be cloud-specific policies and standards. Or the organisation might have decided to keep the policy framework more abstract and only outline high-level requirements that are applicable irrespective of the platforms in question. Nevertheless, the policies and standards should provide the IT organisation involved with clear direction and boundaries. They should make it clear what is acceptable use of public cloud technology and which controls must be implemented, depending on the specific situation. For example: A cloud policy might have rules that point out whether the use of public cloud is allowed for critical or regulated workloads—and if so—under which conditions. Cloud standards will give more detailed rules as to how the implementations must take place and which specific controls must be implemented.

6.6 Governance

There are several definitions for the term ‘Governance’. In the ABN AMRO organisation, it is primarily defined as the activities that are aimed at providing direction (mission, vision, strategy, and goals), putting the organisation in place that will work to efficiently achieve the strategic goals, and that ensures that the organisation and its staff are held to account.

For audits on public cloud implementations, this implies that the different elements of governance should be assessed. The organisation should have a sharp vision of the role of public cloud. This will link to the boundaries set in the companies’ policies. In some cases, the vision of the role of public cloud will be reflected in a specific document that outlines the platform strategy. In practice, auditors should verify that the vision sufficiently supports decision-making. For example, is it clear which types of workloads are allowed to land on the public cloud. And if there is more than one public cloud that is being used: Which types of workloads must land on which public cloud?

Furthermore, the goals of the implementation of public cloud should be clear and the management control system should be aligned with these and support accountability. For example, if the implementation is primarily aimed at cost reductions, does the management control system ensure that cost levels are measured and reported on and that it is clear who has been accountable and responsible for these cost levels?

One specific element of governance relates to the requirement (European Banking Authority, 2019) to have appropriately documented plans for the exit from arrangements with Cloud Service Providers that will enable the organisation to exit the arrangement without undue disruption to the business activities. A distinction can be made between the exit strategy and the more concrete exit plans. The European Banking Federation/Cloud Banking Forum has issued a technical paper (European Banking Federation, 2022) that gives guidance to create a common understanding as to the requirements for the exit strategy and exit plans. In the exit strategy, the organisation should include the identification of an alternative solution/provider, and on a strategic level, which threat scenario could ultimately lead to an exit being triggered. It should furthermore contain an overview of the roles and responsibilities, the human and financial resources that are required to execute the exit and the high-level timelines.

With regard to the concrete exit plan, in our opinion this should not just be a more detailed version of the exit strategy. There should be an exit plan for every workload that has been implemented on the public cloud (component 3 ‘Services and Workloads’ in our framework) and one for the Infrastructure Managed Services (component 2 in our framework) separately. Main reason for this is that the exit requirements can vary per criticality of the service or application in question. These plans should take into account the limitations of the alternative solutions (e.g. the services used might not have a good alternative) and they should describe the steps required to take the data from the service provider and transfer them to alternative providers or back to the organisation.

7 Discussion

As will be clear from the description in Sect. 6, auditing public cloud implementations has many similarities with traditional IT auditing. The subject matter requires specific knowledge on cloud technology in general and the architecture and services of the CSP that this concerns specifically, but the control objectives will be identical and so will most of the control domains. However, there are also some noteworthy differences that require special attention and a different approach that could also have an impact on the required audit resources, both qualitatively and quantitatively. These are elaborated on in the following paragraphs.

7.1 Manual Versus Automated Controls and the Impact on Audit Procedures and Costs

While management of traditional non-cloud environments rely on a combination of manual and automated controls, for public cloud environments, due to the high level of automation and the use of standardised services, they mostly rely on (semi-) automated controls. Typically, these services include out-of-the-box dashboards, metrics, and security baselines. This allows for a shift from distributed/siloed systems to centralised administration (policies/configurations), oversight and control. Consequently, audit procedures will contain more data analyses, which can even be scripted/automated. As more control testing is automated and less manual controls need to be tested, less auditors are required to perform these audit procedures while coverage will usually increase.

However, this is offset by the fact that many companies use more than one public cloud or a combination of private cloud or on-premises systems and public cloud computing. In this situation, even more audit resources are required as more audit terrain is to be covered. Furthermore, one should realise that using public cloud involves outsourcing activities to cloud service providers, and that requires auditors to sufficiently cover these outsourcing arrangements and associated governance and procedure in the audit plan.

7.2 Control over Public Cloud Environments Versus On-Premises IT

The on-premises IT landscape is typically managed by dedicated groups of engineers, and responsibilities between application development and support is usually separated from platform maintenance and support. The high degree of specialisation and sense of responsibility for each on his own area/terrain makes it possible to establish secure and highly available environments. One would expect that cloud environments that are used to develop and host applications are more secure and better controlled because of the potentially strong central control possibilities over the entire environment, such as policy management, continuous monitoring of the implementation/configuration of services, and security event monitoring. Compared to a pluriform on-premises IT landscape, where for each platform a separate set of controls needs to be implemented and central oversight is hard to gain because organisations need to gather and harmonise the data themselves (or connect to central systems like IAM systems, CMDBs, and Splunk), a cloud environment such as Azure at least holds the promise of better control.

But especially while transitioning to the public cloud, there are a few important risks that must be considered. First, the DevOps teams that originated from the former application development and support teams now also need to assume platform maintenance responsibilities, something that they are not accustomed to. Consequently, there is a good chance that these new responsibilities get overlooked. Second, they need to get acquainted with the new (cloud) platform with different services, a new (DevOps) way of working and associated organisational changes and pressure to migrate/transform/rebuild applications during the transition to public cloud. This could be too much to absorb for the teams in question to also keep their environment/application secure. Third, especially in the beginning, finding the right balance between software development and application/platform maintenance and operations tasks is a challenge. Chances are that some of these activities will not get the priority they need. This could manifest itself by configuration/baseline deviations, policy non-compliances, inadequate resource life cycle management and lack of vulnerability management and patching. Azure services like Defender for Cloud and Policy Manager enable organisations to identify many of these shortcomings but these are always easily remediated. Fourth, although Azure supplies powerful services to manage the environment and resources, not all enterprise requirements can be easily met. For example, several access control requirements (e.g. access on least privilege basis) cannot be enforced by one single policy. Another example on access controls pertains to the mapping of authorisations to functions by using authorisation matrices. Unfortunately, there is no automated way to verify these ‘soll’ requirements against the actual implementations (‘ist’). And on data leakage prevention: This requires data labelling, which may be aided by Azure Machine learning, but otherwise is a manual activity. Once labelled (assuming that it will not change), monitoring measures must be developed and implemented. These examples show that considerable effort has to made to enable the control requirements to be enforced, which is quite similar to the work required to control a traditional on-premises IT landscape.

7.3 Public Cloud and DevOps

The implementation of public cloud technology and the introduction of DevOps often go together (Google Cloud, 2021). Although these two implementations should reinforce each other, there are also disadvantages to it. DevOps teams are relatively autonomous, and the general expectation is that these teams will take full responsibility for their workloads (and—depending on the situation—also the underlying platform). In practice however, the maturity level differs between the teams and—consequently—not all teams are able to keep their environments 'clean', i.e. are able to configure all components or services correctly and keep them up-to-date and patched. This might also be caused by the lack of targets that strike the right balance between run and change tasks and that drive priorities of the teams. The way the teams are then consistently (re)directed by line management will affect their behaviour and performance.

Azure supplies monitoring capabilities but these cannot remediate these issues. To address them partially, one could consider—at least temporarily (in the first period after transitioning to the cloud and into a DevOps way of working)—centralising platform maintenance/platform operations tasks. This would relieve some of the burden of the DevOps teams and give them the opportunity to grow into their role.

7.4 Relevance of the Distinction Between IaaS and PaaS

The shared responsibility model distinguishes between IaaS and PaaS services to make clear where maintenance responsibilities lie. But to what extent is this distinction relevant for IT auditors?

We can imagine that a company would only use IaaS services and build all additional functionality themselves or implement third party software on their virtual machines. In that case many cloud/Azure control measures will not apply (e.g. Defender for Cloud most probably does not know these third party products and Azure policies will not apply. In addition, security event monitoring must be configured largely separately). As the IaaS deployment model comes closest to an on-premise environment, many of the benefits of public cloud will not be enjoyed. For example, the benefits from service features like scalability, elasticity, and site recovery will not be available for organisations that just use IaaS services. However, it will give the highest level of control over what is implemented when and where exactly and it gives the organisation the highest independence of the CSP (which could be beneficial if an exit from the CSP needs to take place). It also requires the most additional measures to make the environment secure. Besides general IT controls (e.g. logical access, hardening, vulnerability management, and patch management) you may expect to test cloud-specific controls that pertain to the IaaS services used.

Let us elaborate on this a bit further. Typically, IaaS services at Azure are categorised as compute, network, and storage. When we look at compute (hosting services responsible for hosting and running the application workloads), the following services are available: Azure Virtual Machines (VMs), Azure Container Service, Azure App Services, Azure Batch, and Azure ServiceFabric. There are VMs for Windows and Linux. Now suppose that you run the application on Linux, then the aforementioned general IT controls also apply to that Linux VM. Nothing new because you probably already knew Linux from your on-premise IT landscape.

When the Azure service is not familiar to the organisation because it does not have a counterpart on on-premise implementations, such as probably Azure Batch, then you would probably also not use it in the cloud. But if you do, then from an audit perspective you would probably look at the same aspects that are covered by the general IT controls. Because after all it is just software that provides functionality. No matter how magical the cloud services sometimes may seem because they are unparalleled in the on-premise domain, it is software that was coded (with potential flaws that need to be patched) and can be configured (which may affect hardening). In other words, the Azure services may look different from what you are used to, but in essence the same general IT controls apply. That does not mean that nothing changes in the audit practice. When you as an IT auditor have access to the Azure portal with read access on most resources, you will have to get used to the interface, get acquainted with most used services, learn to use services like the Policy Manager or Defender for Cloud, get a feeling of where critical settings are to be found, etc. We can tell from experience that it is another world in appearance, and it takes time to get used to.

If it were possible to use only PaaS services, then your audit activities would change compared to only IaaS, because the number of components that you cannot 'see' increases. For example, Azure SQL Database is a fully managed platform as a service (PaaS) database engine that handles most of the database management functions such as upgrading, patching, backups, and monitoring without user/customer involvement. Azure SQL Database is always running on the latest stable version of the SQL Server database engine and patched OS with 99.99% availability (Microsoft, 2022a). All the PaaS services have in common that regarding the general IT controls you can no longer assess hardening measures, vulnerability management, and patch management, because they are not under the customer's control. The CSP takes care of these and if you want assurance on how they do it, you have to rely on external certifications or carry out an audit on the relevant CSP activities.

So, which audit activities remain? In this case and most probably in general they would pertain to logical access controls and data controls. Regarding access, you would like to verify whether the required access levels, described in e.g. an authorisation matrix, are actually enforced and cannot be circumvented. It should cover all types of access by users, administrators, and applications/NPAs, including emergency/troubleshoot access. Regarding confidential data you would expect encryption of data in-transit and at-rest. The encryption measures should comply to your (and regulatory, e.g. GDPR) requirements, which could mean that you would have to assess cryptographic key management measures performed by your company as well. When availability requirements are high, you would have to verify whether the appropriate measures have been taken. In this case, e.g. turning on Azure SQL geo-replication and making sure that your Allowed Location policies apply. Maybe regulatory retention periods apply to your data. Then you would have to implement additional measures to meet those. The IT auditor must assess whether the design is adequate and whether the measures are operationally effective.

Most probably, application developers/DevOps teams will use a combination of IaaS and PaaS. In that case of course the beforementioned considerations regarding IaaS and PaaS apply. But you would need to make the distinction in order to be able to decide which controls you need to test. The first control would be to verify whether only the services described in a solution design are deployed in a specific resource group. Chances are that in time more services are used/added then originally foreseen and documented.

In principle all the services in that resource group need to be assessed. From each you have to determine whether it is IaaS or PaaS before you can start assessing the applicable controls per service, considering company and regulatory requirements. Regarding data flows you may need to verify whether they are as designed/required and whether network security measures comply to your requirements (like our internal network security framework).

When availability requirements are high, it is important to establish per service used which measures apply (because they can differ per service) and whether they have been implemented adequately. Based upon a view on the individual services, you can assess impact on application availability.

When performing an audit, the distinction between IaaS and PaaS is very relevant. In view of the way things organised at your company, e.g. with centrally managed infrastructure services and (decentral) DevOps teams responsible for application development, it makes sense to divide audit activities as well. The audit department is traditionally mirrored to the organisation and therefore facilitates the division. Application auditors focus on the deployments/workloads/applications and the auditors assigned to IT infrastructure focus on the infrastructure managed services. Typically, with every application audit, an auditor from the infrastructure team participates. This is beneficial because it stimulates knowledge cross-pollination and allows better understanding of the relevant aspects, which will ultimately result in better risk assessments and audit engagements.

7.5 Managing Costs

One of the strengths of CSPs is that they provide services on a pay-per-use basis. To support this feature, usage of every service needs to be metered. The customer has access to these statistics and the costs. Usage/costs can be viewed from different angles and on different levels.

From an audit perspective you would not only be interested in whether a targeted cost reduction on company level was met, but also whether productivity increased and time-to-market was reduced. Additionally you could assess whether budgets for application development or development of new features have decreased, compared to the actual costs and when there are differences, whether the appropriate measures have been taken.

In our experience the expectations may be too high. When your organisation, or at least the application development part, has a high level of maturity, the DevOps teams are used to the new way of working, they have perfect understanding of the cloud services and are stable in composition (i.e. no or low attrition), then your chances of realising your targets are the best. But from the list of requirements you can already deduce that for companies embarking on a cloud journey most probably these requirements will not all be met, certainly not from the beginning. Should you give DevOps teams carte blanche at the start? That is completely at the other end of the spectrum and probably nobody would agree. For sure DevOps teams need to experiment and learn how to use the services and what their features are. This will take time and resources and it would not be fair to expect the same productivity from a starting team as from an experienced team.

Of course, CSPs can only measure consumption of their services. That however, are not the only costs of application development. Companies hire staff or outsource functions, have management costs, provide their staff with working places, laptops and mobile phones, etc. So it would be an oversimplification to say that migration to a cloud environment would give you more control over your IT costs. That only holds true for the cloud services consumption part.

In addition, productivity of DevOps teams is only very indirectly related to cloud resource consumption. You can measure usage of services but you must not confuse that with the development efforts to provide functionality. Suppose a new feature needs to be added to an already developed application that consists of a number of (IaaS and PaaS) services and additional application code. The design, development, testing, and deployment efforts will consume cloud services but are no indication that functional requirements have been met. At best, higher resource consumption during development may indicate complexity.

Probably, the challenge to predict development time and effort to realise application functionality (and thereby DevOps team productivity) in cloud environments does not differ much from traditional environments, but this would be an interesting topic to explore.

8 Conclusions

We briefly described the rise of public cloud computing and the initial hesitance to adopt public cloud technology by the financial services industry (Sect. 1). Next, we elaborated on cloud deployment models (private, public, and hybrid) and service models (IaaS, PaaS, SaaS) to generally set the scene for audit activities (Sect. 2). In Sect. 3, publicly available audit frameworks and work programs were evaluated in terms of suitability for usage for audits. Section 4 presented the case study of the IT/Cloud transformation of our organisation and in Sect. 5, the audits activities that we performed were presented, which formed the basis for our conceptual framework (Sect. 6). In Sects. 6.2 and 6.3 we have provided examples of concrete/detailed controls regarding commonly used cloud services configuration that can help as a starting point for audits.

Although the look and feel of cloud environments differs hugely from traditional IT landscapes, we came to the conclusion that the audit attention points are largely similar. Therefore the execution of an audit will differ in components and configurations to cover, but risks remain largely the same. The implementation of controls will differ, because cloud environments offer other/new tools and services.

Compared to traditional on-premise IT landscapes, the level of control for a number of areas can be higher in cloud environments. That can be largely attributed to the environment having a uniform basis and being able to have general oversight via maintenance/management tooling. Regarding preventative controls, the same policies can be enforced on all subscriptions and resource groups, which is a very strong control. However, designing and implementing the appropriate policy set can be challenging. Likewise, regarding detective controls, the range of vulnerability scanning and security event monitoring can be across all your subscriptions and resource groups. But also these have their pitfalls: you have to evaluate reported vulnerabilities for applicability and you have to identify security events. Although, the latter may be compensated by machine learning in the near future.

While every customer environment and DevOps team can be different, from enterprise control perspective it can be rewarding to centralise the following functions:

1. Identity Management
2. Policy Management
3. Subscription Management/Secure landing zones
4. Network Management
5. Support for implementing Security Event Monitoring

The organisation size and auditor experience/education are key factors to consider before engaging in cloud audits. Knowledge of technologies, products, and services is essential and larger audit teams are better equipped to facilitate cross-learning between auditors.

References

Akdeniz, D., Bani Hashemi, S. J., Putters, J., & Yavuz, A. (2020). *Pooled audits on cloud service providers—Part 1*. Retrieved from https://www.deitauditor.nl/business-en-it/pooled-audits-on-cloud-service-providers/

Amazon. (2021). *Shared responsibility model*. Amazon Web Services, Inc. Retrieved March 23, 2022, van https://aws.amazon.com/compliance/shared-responsibility-model/

Amazon. (n.d.). *What is cloud computing*. Amazon Web Services, Inc. Retrieved March 23, 2022, from https://aws.amazon.com/what-is-cloud-computing/

Association for Financial Markets in Europe. (2019, November). *The adoption of public Cloud Computing in capital markets*. Retrieved from https://www.afme.eu/Publications/Reports/Details/The-Adoption-of-Public-Cloud-Computing-in-Capital-Markets

Axelos. (2020). *ITIL4*. Retrieved from https://www.axelos.com/certifications/itil-service-management

Bani Hashemi, S. J., Putters, J., & Yavuz, A. (2020). *Pooled audits on cloud service providers—Part 2*. Retrieved from https://www.deitauditor.nl/business-en-it/pooled-audits-on-cloud-service-providers-2/

CSA. (2021). *Cloud controls matrix*. Retrieved from https://cloudsecurityalliance.org/research/cloud-controls-matrix/

European Banking Authority. (2019, February). *EBA guidelines on outsourcing arrangements*. Retrieved from https://www.eba.europa.eu/eba-publishes-revised-guidelines-on-outsourcing-arrangements

European Banking Federation. (2022). *Cloud exit strategy—Testing of exit plans*. Retrieved from https://www.ebf.eu/wp-content/uploads/2020/06/EBF-Cloud-Banking-Forum_Cloud-exit-strategy-testing-of-exit-plans.pdf

European Securities and Markets Authority. (2020, December). *Guidelines on outsourcing to cloud service providers*. Retrieved from https://www.esma.europa.eu/press-news/esma-news/esma-publishes-cloud-outsourcing-guidelines

Gartner. (2021, August 2). *Gartner says four trends are shaping the future of public cloud* [Press release]. Retrieved from https://www.gartner.com/en/newsroom/press-releases/2021-08-02-gartner-says-four-trends-are-shaping-the-future-of-public-cloud

Google Cloud. (2021). *Accelerate State of devops 2021*. Google Inc. Retrieved from https://services.google.com/fh/files/misc/state-of-devops-2021.pdf

Haes, S. D., Grembergen, W. V., Anant, J., & Huygh, T. (2015). COBIT as a framework for enterprise governance of IT. In *Enterprise Governance of Information Technology: Achieving Alignment and Value, Featuring COBIT 5* (2nd ed., pp. 103–128). Springer. https://doi.org/10.1007/978-3-030-25918-1

Institute of Internal Auditors. (2018). *Auditing Third-party Risk Management-supplemental guidance—Practice guide*. Institute of Internal Auditors.

ISACA. (2020–2021). *Cloud Computing Management Audit Program*. Retrieved from https://store.isaca.org/s/store#/store/browse/detail/a2S4w000004KoH1EAK

ISACA. (2020–2022). *Azure Audit Program*. Retrieved from https://store.isaca.org/s/store#/store/browse/detail/a2S4w000004KoGTEA0

Jones, E. (2021). *Types of Cloud Computing—An extensive guide on cloud solutions and technologies in 2021*. Retrieved from https://kinsta.com/blog/types-of-cloud-computing/

Mell, P., & Grance, T. (2011). *The NIST definition of Cloud Computing*. National Institute of Standards and Technology. Retrieved from https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-145.pdf

Microsoft. (2022a, February 5). *What is the Azure SQL Database service?* Microsoft Docs. Retrieved March 23, 2022, from https://docs.microsoft.com/en-us/azure/azure-sql/database/sql-database-paas-overview

Microsoft. (2022b, March 1). *Azure regions and availability zones*. Microsoft Docs. Retrieved March 23, 2022, from https://docs.microsoft.com/en-us/azure/availability-zones/az-overview

Microsoft. (2022c, March 1). *Shared responsibility in the cloud.* Microsoft Docs. Retrieved March 23, 2022, from https://docs.microsoft.com/en-us/azure/security/fundamentals/shared-responsibility

Microsoft. (2022d, March 18). *Azure AD built-in roles.* Microsoft Docs. Retrieved March 18, 2022, from https://docs.microsoft.com/en-us/azure/active-directory/roles/permissions-reference

Microsoft. (2022e, March 21). *Organize your resources with management groups*. Microsoft Docs. Accessed March 18, 2022, from https://docs.microsoft.com/en-us/azure/governance/management-groups/overview

Microsoft. (n.d.). *Tour of Azure services.* Microsoft Docs. Retrieved March 23, 2022, from https://docs.microsoft.com/en-us/learn/modules/intro-to-azure-fundamentals/tour-of-azure-services

Monterie, A. (2020). *ABN Amro maakt geslaagde migratie van AWS naar Azure*. Retrieved from https://www.computable.nl/artikel/achtergrond/cloud-computing/7039237/1444691/abn-amro-maakt-geslaagde-migratie-van-aws-naar-azure.html

Rosa, J., & Dee, M. (2020). *Transformation at ABN AMRO Bank*. Retrieved from https://www.agilealliance.org/resources/experience-reports/devops-transformation-at-abn-amro-bank/