

15.2 IaaS, PaaS und SaaS im Detail

IaaS, PaaS und SaaS sind die wichtigsten Cloud-Service-Modelle.

Sie beschreiben, welche Teile der IT vom Anbieter bereitgestellt werden und welche Teile der Kunde selbst verwaltet.

Die drei Begriffe stehen für:

IaaS:

Infrastructure as a Service

PaaS:

Platform as a Service

SaaS:

Software as a Service

Merksatz:

IaaS,

PaaS

und SaaS unterscheiden sich vor allem durch die Verantwortungsverteilung.

Warum sind Service-Modelle wichtig?

In Prüfungen und in der Praxis muss man erkennen können:

Was betreibt der Anbieter?

Was betreibt der Kunde?

Wer ist für Updates verantwortlich?

Wer kümmert sich um Benutzer?

Wer sichert Daten?

Wer konfiguriert die Anwendung?

Wer schützt Zugänge?

Das ist wichtig, weil Cloud nicht bedeutet, dass der Anbieter automatisch alles übernimmt.

Merksatz:

Cloud-Nutzung heißt nicht:
keine eigene Verantwortung mehr.

Grundidee der drei Modelle

Vereinfacht:

IaaS:

Der Anbieter liefert die technische Infrastruktur.

PaaS:

Der Anbieter liefert zusätzlich die Plattform für Anwendungen.

SaaS:

Der Anbieter liefert die fertige Anwendung.

Je weiter man von IaaS zu SaaS geht, desto mehr übernimmt der Anbieter.

Merksatz:

IaaS = viel Kontrolle.

SaaS = viel fertiger Dienst.

PaaS = dazwischen.

Vergleich in Kurzform

Modell	Kunde bekommt	Kunde verwaltet vor allem
IaaS	virtuelle Infrastruktur	Betriebssystem, Anwendung, Daten
PaaS	Plattform für Anwendungen	Anwendung, Daten, Konfiguration

Modell	Kunde bekommt	Kunde verwaltet vor allem
SaaS	fertige Anwendung	Benutzer, Rechte, Daten, Einstellungen

Merksatz:

Je fertiger der Dienst,
desto weniger technische Basis verwaltet der Kunde selbst.

IaaS im Detail

IaaS bedeutet:

Infrastructure as a Service

Der Anbieter stellt grundlegende Infrastruktur bereit.

Dazu gehören zum Beispiel:

- virtuelle Maschinen
- virtuelle Netzwerke
- virtuelle Festplatten
- Firewalls
- Load Balancer
- öffentliche IP-Adressen
- private IP-Adressen
- Snapshots
- Storage
- Routingtabellen

Der Kunde baut darauf seine Systeme auf.

Merksatz:

IaaS ist wie ein gemieteter virtueller Serverraum.

Typische IaaS-Beispiele

Typische IaaS-Ressourcen:

- virtuelle Server
- virtuelle Router
- virtuelle Firewalls
- virtuelle Netzwerke
- Block Storage
- Object Storage
- Load Balancer
- VPN-Gateway
- Backup-Speicher

Typische Nutzung:

- eigene Anwendungen betreiben
- eigene Server migrieren
- Testumgebungen aufbauen
- Entwicklungsumgebungen bereitstellen
- Disaster Recovery vorbereiten

Merksatz:

IaaS eignet sich,
wenn man viel Kontrolle über Server und Netzwerk braucht.

Was übernimmt der Anbieter bei IaaS?

Der Anbieter übernimmt typischerweise:

- Rechenzentrum
- physische Server
- Stromversorgung
- Kühlung
- physische Netzwerkanbindung
- Hardwarewartung
- Virtualisierungsplattform
- Grundverfügbarkeit der Infrastruktur

Der Kunde muss sich nicht selbst um physische Hardware kümmern.

Merksatz:

Bei IaaS übernimmt der Anbieter vor allem die physische und virtuelle Basis.

Was übernimmt der Kunde bei IaaS?

Der Kunde übernimmt typischerweise:

Betriebssysteminstallation
Betriebssystemupdates
Benutzerkonten
Firewall-Konfiguration
Anwendungsinstallation
Anwendungskonfiguration
Daten
Backups
Monitoring
Sicherheitskonfiguration
Patches innerhalb der VM

Merksatz:

Eine Cloud-VM muss der Kunde meistens selbst administrieren.

IaaS-Vorteile

Vorteile:

hohe Kontrolle
flexible Servergrößen
schnelle Bereitstellung
keine eigene Hardwarebeschaffung
gut für Migration vorhandener Server
gut für individuelle Anwendungen
skalierbare Infrastruktur
netzwerktechnisch flexibel

Merksatz:

IaaS bietet viel Flexibilität und Kontrolle.

IaaS-Nachteile

Nachteile:

Kunde muss viel selbst verwalten
Betriebssystempflege bleibt notwendig
Sicherheitskonfiguration bleibt Kundenthema
Fehlkonfigurationen möglich
Kosten können steigen
Monitoring und Backup müssen geplant werden
Know-how für Betrieb nötig

Merksatz:

IaaS nimmt Hardware ab,
aber nicht den Serverbetrieb.

IaaS-Prüfungsbeispiel

Aufgabe:

Ein Unternehmen betreibt eine virtuelle Linux-VM in der Cloud.
Darauf läuft eine eigene Webanwendung.

Frage:

Wer ist für Betriebssystemupdates verantwortlich?

Antwort:

In der Regel der Kunde.

Warum?

Der Anbieter stellt die Infrastruktur.

Der Kunde betreibt Betriebssystem und Anwendung.

Merksatz:

Cloud-VM heißt nicht:

Anbieter patcht automatisch das Betriebssystem.

PaaS im Detail

PaaS bedeutet:

Platform as a Service

Der Anbieter stellt eine Plattform bereit, auf der Anwendungen betrieben werden können.

Der Kunde muss sich nicht direkt um die darunterliegende Infrastruktur kümmern.

Typische Bestandteile:

Laufzeitumgebung

Web-App-Plattform

verwaltete Datenbank

Middleware

Build- und Deployment-Funktionen

Skalierungsfunktionen

Monitoring-Grundfunktionen

Merksatz:

PaaS ist eine fertige Plattform für eigene Anwendungen.

Typische PaaS-Beispiele

Typische PaaS-Angebote:

verwaltete Datenbank
Web-App-Service
App-Plattform
Containerplattform als Dienst
Entwicklungsplattform
API-Plattform
Message Queue als Dienst
Cache-Dienst
Laufzeitumgebung für Code

Typische Nutzung:

eigene Anwendung bereitstellen,
ohne Serverbetrieb im Detail zu übernehmen.

Merksatz:

PaaS ist gut,
wenn man Anwendung betreiben will,
aber weniger Serververwaltung möchte.

Was übernimmt der Anbieter bei PaaS?

Der Anbieter übernimmt typischerweise:

physische Infrastruktur
Virtualisierung
Betriebssystembasis
Laufzeitumgebung
Plattformupdates
Skalierungsmechanismen
Teile des Monitorings
Hochverfügbarkeitsfunktionen je nach Dienst

Merksatz:

Bei PaaS übernimmt der Anbieter mehr als bei IaaS.

Was übernimmt der Kunde bei PaaS?

Der Kunde übernimmt typischerweise:

- Anwendungscode
- Anwendungskonfiguration
- Daten
- Benutzer und Rollen
- Zugriffsschlüssel
- API-Konfiguration
- sichere Einstellungen
- fachliche Backups je nach Dienst
- Berechtigungen
- Datenschutz und Compliance

Merksatz:

Bei PaaS bleibt die Anwendung Kundenthema.

PaaS-Vorteile

Vorteile:

- weniger Serveradministration
- schnellere Entwicklung
- einfachere Skalierung
- integrierte Plattformfunktionen
- weniger Betriebssystempflege
- oft automatische Updates der Plattform
- gut für moderne Anwendungen
- häufig gute Integration in DevOps-Prozesse

Merksatz:

PaaS entlastet vom Betrieb der technischen Basis.

PaaS-Nachteile

Nachteile:

- weniger Kontrolle über darunterliegende Systeme
- Abhängigkeit von Plattformfunktionen
- mögliche Anbieterbindung
- Einschränkungen bei Spezialkonfigurationen
- Kostenmodell muss verstanden werden
- Fehlkonfigurationen weiterhin möglich

Merksatz:

PaaS nimmt Arbeit ab,
kann aber stärker an den Anbieter binden.

PaaS-Prüfungsbeispiel

Aufgabe:

Ein Unternehmen nutzt eine verwaltete Datenbank in der Cloud.

Frage:

Muss das Unternehmen den Datenbankserver selbst installieren?

Antwort:

Nein.

Aber das Unternehmen bleibt verantwortlich für:

- Daten
- Benutzerrechte
- Zugriffskonfiguration
- Backup-Einstellungen je nach Dienst
- sichere Konfiguration

Merksatz:

Verwaltete Datenbank bedeutet nicht:
keine Datenverantwortung.

SaaS im Detail

SaaS bedeutet:

Software as a Service

Der Kunde nutzt eine fertige Anwendung, die vom Anbieter betrieben wird.

Typische Beispiele:

E-Mail
Online-Office
CRM
Ticketsystem
Videokonferenzdienst
Projektmanagement
Cloud-Speicher
Buchhaltungssystem
Lernplattform

Merksatz:

SaaS ist fertige Software aus der Cloud.

Was übernimmt der Anbieter bei SaaS?

Der Anbieter übernimmt typischerweise:

Anwendung
Plattform
Infrastruktur
Updates
Wartung
Grundverfügbarkeit

- technische Skalierung
- technische Sicherheitsmaßnahmen
- Betrieb der Datenbank
- Betrieb der Server

Merksatz:

Bei SaaS übernimmt der Anbieter den größten Teil des technischen Betriebs.

Was übernimmt der Kunde bei SaaS?

Der Kunde übernimmt typischerweise:

- Benutzerkonten
- Rollen
- Berechtigungen
- MFA-Einstellungen
- Datenklassifizierung
- Freigaben
- Aufbewahrungsrichtlinien
- Löschkonzepte
- Schulung der Benutzer
- sichere Konfiguration
- Vertrags- und Datenschutzprüfung

Merksatz:

SaaS nimmt Betrieb ab,
aber nicht Benutzer-,
Rechte-
und Datenverantwortung.

SaaS-Vorteile

Vorteile:

- schnell nutzbar
- keine eigene Serverinstallation
- automatische Updates durch Anbieter
- ortsunabhängiger Zugriff
- einfache Skalierung nach Benutzern
- oft geringe Einstiegshürde
- gute Zusammenarbeit
- weniger technischer Betrieb

Merksatz:

SaaS ist schnell einsatzbereit und reduziert Betriebsaufwand.

SaaS-Nachteile

Nachteile:

- weniger technische Kontrolle
- Abhängigkeit vom Anbieter
- Datenstandort prüfen
- Datenschutz prüfen
- Lizenzkosten beachten
- Internetabhängigkeit
- eingeschränkte Anpassbarkeit
- falsche Freigaben können Daten offenlegen

Merksatz:

SaaS ist bequem,
braucht aber sorgfältige Rechte- und Datenschutzkontrolle.

SaaS-Prüfungsbeispiel

Aufgabe:

Ein Unternehmen nutzt einen Cloud-E-Mail-Dienst.

Frage:

Wer ist für Benutzerrechte,
MFA
und Postfachfreigaben verantwortlich?

Antwort:

Der Kunde.

Der Anbieter betreibt den Dienst, aber der Kunde verwaltet Zugriffe und Einstellungen.

Merksatz:

SaaS-Anbieter betreibt die Software,
der Kunde verwaltet Nutzung und Zugriff.

Verantwortungsschichten

Man kann IT grob in Schichten betrachten:

Daten
Anwendung
Laufzeitumgebung
Middleware
Betriebssystem
Virtualisierung
Server
Speicher
Netzwerk
Rechenzentrum

Je nach Service-Modell verschiebt sich Verantwortung.

Merksatz:

Service-Modelle verschieben Verantwortung zwischen Anbieter und Kunde.

Verantwortung grob vergleichen

Schicht	IaaS	PaaS	SaaS
Daten	Kunde	Kunde	Kunde
Benutzer und Rechte	Kunde	Kunde	Kunde
Anwendung	Kunde	Kunde	Anbieter
Laufzeitumgebung	Kunde	Anbieter	Anbieter
Betriebssystem	Kunde	Anbieter	Anbieter
Virtualisierung	Anbieter	Anbieter	Anbieter
Physische Server	Anbieter	Anbieter	Anbieter
Rechenzentrum	Anbieter	Anbieter	Anbieter

Merksatz:

Daten,
Benutzer
und Rechte bleiben fast immer Kundenthema.

Shared Responsibility Model

Das Shared Responsibility Model beschreibt, wie sich Anbieter und Kunde Verantwortung teilen.

Wichtig:

Es gibt keine pauschale Antwort für alle Dienste.

Die genaue Verantwortung hängt ab von:

Service-Modell
Anbieter
Vertrag
Dienstkonfiguration
Sicherheitsoptionen
Unternehmensrichtlinien

Merksatz:

Shared Responsibility immer konkret für den genutzten Dienst prüfen.

Typischer Denkfehler

Falsch:

Das ist Cloud,
also kümmert sich der Anbieter um Sicherheit.

Richtig:

Der Anbieter kümmert sich je nach Modell um Teile der Sicherheit.
Der Kunde bleibt für sichere Nutzung,
Konfiguration,
Daten,
Identitäten
und Berechtigungen verantwortlich.

Merksatz:

Cloud-Sicherheit ist gemeinsame Verantwortung.

Beispiel: Unsichere IaaS-Konfiguration

Ein Unternehmen betreibt eine VM in der Cloud.

Fehler:

SSH offen für das gesamte Internet
schwaches Passwort
keine Updates
keine Host-Firewall
keine Backups
keine Logs

Obwohl die Cloud-Infrastruktur sicher betrieben wird, ist die VM unsicher konfiguriert.

Merksatz:

IaaS-Fehlkonfigurationen liegen oft beim Kunden.

Beispiel: Unsichere PaaS-Konfiguration

Ein Unternehmen nutzt eine verwaltete Datenbank.

Fehler:

Datenbank öffentlich erreichbar
zu breite IP-Freigabe
schwache Zugangsdaten
keine Verschlüsselung erzwungen
keine Backups aktiviert
Adminzugang für zu viele Benutzer

Merksatz:

PaaS muss sicher konfiguriert werden.

Beispiel: Unsichere SaaS-Konfiguration

Ein Unternehmen nutzt Cloud-Speicher.

Fehler:

Ordner öffentlich freigegeben
Gastzugriff zu breit erlaubt
MFA nicht aktiviert
alte Benutzerkonten aktiv
externe Freigaben nicht geprüft
keine Aufbewahrungsrichtlinien

Merksatz:

SaaS-Fehler entstehen oft durch falsche Freigaben und Rechte.

Lift and Shift

Lift and Shift bedeutet:

bestehende Server oder Anwendungen werden möglichst unverändert in die Cloud verschoben.

Beispiel:

lokaler Windows-Server
wird als Cloud-VM betrieben.

Vorteil:

schnelle Migration

Nachteil:

Cloud-Vorteile wie PaaS,
Skalierung
und Automatisierung werden oft nur teilweise genutzt.

Merksatz:

Lift and Shift verschiebt Systeme,
modernisiert sie aber nicht automatisch.

Cloud-native

Cloud-native bedeutet:

Anwendungen werden speziell für Cloud-Umgebungen entwickelt oder angepasst.

Typische Merkmale:

Skalierbarkeit
Automatisierung
Container
Microservices
API-orientierte Architektur

Monitoring
schnelle Bereitstellung
Resilienz

Merksatz:

Cloud-native nutzt Cloud-Prinzipien bewusst aus.

Managed Service

Ein Managed Service ist ein Dienst, bei dem der Anbieter Betrieb und Wartung teilweise übernimmt.

Beispiele:

verwaltete Datenbank
verwalteter Kubernetes-Cluster
verwalteter Cache
verwaltete Firewall
verwaltetes Backup

Merksatz:

Managed Service reduziert Betriebsaufwand,
aber nicht automatisch Kundenverantwortung.

Self-Managed

Self-Managed bedeutet:

Der Kunde betreibt und verwaltet einen Dienst selbst.

Beispiel:

Datenbank selbst auf einer Cloud-VM installieren.

Der Kunde ist dann verantwortlich für:

Installation
Updates
Backup
Sicherheit
Monitoring
Skalierung
Wiederherstellung

Merksatz:

Self-Managed gibt Kontrolle,
aber auch Betriebsverantwortung.

Managed und Self-Managed vergleichen

Merkmal	Managed Service	Self-Managed
Betrieb	Anbieter übernimmt mehr	Kunde übernimmt mehr
Kontrolle	etwas geringer	höher
Wartung	geringer für Kunden	höher
Flexibilität	abhängig vom Dienst	oft höher
Verantwortung	geteilt	stärker beim Kunden
Beispiel	Cloud-Datenbank	Datenbank auf eigener VM

Merksatz:

Managed spart Betrieb.
Self-Managed gibt Kontrolle.

As-a-Service-Prinzip

As a Service bedeutet:

Eine IT-Funktion wird als Dienst bereitgestellt.

Neben IaaS, PaaS und SaaS gibt es weitere Begriffe.

Beispiele:

Backup as a Service
Database as a Service
Desktop as a Service
Security as a Service
Function as a Service
Network as a Service

Merksatz:

As a Service bedeutet:
Nutzung als Dienst statt vollständig eigener Betrieb.

DaaS

DaaS kann je nach Kontext bedeuten:

Desktop as a Service

oder:

Database as a Service

In Prüfungen muss man auf den Kontext achten.

Desktop as a Service:

virtuelle Desktops aus der Cloud

Database as a Service:

verwaltete Datenbank als Cloud-Dienst

Merksatz:

DaaS immer im Kontext lesen.

FaaS

FaaS steht für:

Function as a Service

Dabei werden einzelne Funktionen ausgeführt, wenn ein Ereignis eintritt.

Beispiel:

Datei wird hochgeladen.
Funktion startet automatisch.
Datei wird verarbeitet.

FaaS gehört oft zum Serverless-Bereich.

Merksatz:

FaaS führt Code ereignisgesteuert aus.

BaaS

BaaS kann bedeuten:

Backup as a Service

oder:

Backend as a Service

Backup as a Service:

Backups werden als Cloud-Dienst bereitgestellt.

Backend as a Service:

Backend-Funktionen für Anwendungen werden bereitgestellt,
zum Beispiel Authentifizierung,

Datenbank
oder Push-Benachrichtigungen.

Merksatz:

BaaS ebenfalls immer im Kontext verstehen.

XaaS

XaaS steht für:

Anything as a Service

oder:

Everything as a Service

Es ist ein Sammelbegriff für viele Dienste, die als Cloud-Service bereitgestellt werden.

Merksatz:

XaaS ist ein Oberbegriff für Dienste aus der Cloud.

Typische Einordnung von Beispielen

Beispiel	Service-Modell
virtuelle Maschine	IaaS
virtueller Datenträger	IaaS
virtuelles Netzwerk	IaaS
verwaltete Datenbank	PaaS
Web-App-Plattform	PaaS
Containerplattform als Dienst	PaaS
fertiges E-Mail-System	SaaS
Online-Office	SaaS
CRM aus der Cloud	SaaS
Ticketsystem aus der Cloud	SaaS

Beispiel	Service-Modell
einzelne Funktion bei Ereignis	FaaS / Serverless

Merksatz:

Frage immer:
 Nutze ich Infrastruktur,
 Plattform
 oder fertige Software?

Entscheidungshilfe

Ziel	passendes Modell
maximale Kontrolle über Server	IaaS
eigene Anwendung ohne Serverbetrieb	PaaS
fertige Anwendung schnell nutzen	SaaS
Ereignisgesteuerte kleine Funktionen	FaaS
klassische Servermigration	IaaS / Lift and Shift
moderne skalierbare Anwendung	PaaS / Container / Serverless

Merksatz:

Das Modell richtet sich nach Kontrolle,
 Aufwand
 und Anwendungszweck.

Kostenunterschiede

IaaS, PaaS und SaaS haben unterschiedliche Kostenmodelle.

IaaS:

VM-Laufzeit
 Speicher
 Netzwerkverkehr
 Backups

PaaS:

Dienstgröße
Nutzung
Datenbankleistung
Anfragen
Speicher

SaaS:

Benutzerlizenzen
Funktionspakete
Speicher
Zusatzmodule

Merksatz:

Cloud-Kosten entstehen je nach Service-Modell unterschiedlich.

Sicherheitsunterschiede

IaaS:

Kunde muss Server stark selbst absichern.

PaaS:

Kunde muss Anwendung,
Daten
und Zugriff absichern.

SaaS:

Kunde muss Benutzer,
Rollen,
Freigaben
und Daten absichern.

Merksatz:

Sicherheit verlagert sich,
verschwindet aber nicht.

Betriebsunterschiede

IaaS:

viel Betrieb beim Kunden

PaaS:

mittlerer Betrieb beim Kunden

SaaS:

wenig technischer Betrieb beim Kunden

Aber:

Fachliche Administration bleibt bei allen Modellen wichtig.

Merksatz:

Weniger technischer Betrieb heißt nicht keine Administration.

Typische IHK-Fragen

In AP1 und AP2 kann zum Beispiel gefragt werden:

- Was bedeutet IaaS?
- Was bedeutet PaaS?
- Was bedeutet SaaS?
- Worin unterscheiden sich IaaS, PaaS und SaaS?
- Wer ist bei IaaS für Betriebssystemupdates verantwortlich?

- Wer verwaltet bei SaaS Benutzer und Rechte?
- Was bedeutet Shared Responsibility?
- Warum ist eine Cloud-VM nicht automatisch sicher?
- Was ist ein Managed Service?
- Was ist Self-Managed?
- Was bedeutet Lift and Shift?
- Was bedeutet Cloud-native?
- Was ist FaaS?
- Warum ist Hashing oder Verschlüsselung allein kein Ersatz für Rechteverwaltung?
- Warum bleibt der Kunde auch bei SaaS für Daten verantwortlich?

Typische Prüfungsfallen

IaaS ist nicht fertige Software.

SaaS ist nicht eigener Server.

PaaS liegt zwischen IaaS und SaaS.

Cloud-VMs muss der Kunde oft selbst patchen.

Verwaltete Datenbank heißt nicht:

Daten sind automatisch korrekt geschützt.

SaaS heißt nicht:

alle Benutzerrechte sind automatisch sicher.

Shared Responsibility gilt immer konkret je Dienst.

Daten bleiben Kundenthema.

Benutzer und Rechte bleiben Kundenthema.

Backup muss geprüft werden.

Cloud ist nicht automatisch sicher.

Cloud ist nicht automatisch günstiger.

Managed Service reduziert Betrieb,
aber nicht Verantwortung.

Lift and Shift modernisiert nicht automatisch.

Serverless heißt nicht,
dass keine Server existieren.

FaaS führt Funktionen ereignisgesteuert aus.

DaaS und BaaS im Kontext lesen.

Wichtige Begriffe kurz erklärt

Begriff	Kurze Erklärung
IaaS	Infrastructure as a Service
PaaS	Platform as a Service
SaaS	Software as a Service
Shared Responsibility	geteilte Verantwortung zwischen Anbieter und Kunde
Cloud-VM	virtuelle Maschine in der Cloud
Managed Service	Anbieter übernimmt Teile des Betriebs
Self-Managed	Kunde betreibt Dienst selbst
Lift and Shift	bestehendes System in Cloud verschieben
Cloud-native	speziell für Cloud-Prinzipien entwickelt
FaaS	Function as a Service
DaaS	je nach Kontext Desktop oder Database as a Service
BaaS	je nach Kontext Backup oder Backend as a Service
XaaS	Sammelbegriff für Dienste als Service
Plattform	Umgebung zum Ausführen von Anwendungen
Runtime	Laufzeitumgebung für Anwendungscode
Middleware	Software zwischen Betriebssystem und Anwendung
Service-Modell	beschreibt Verantwortungsverteilung
Anbieter	Cloud-Provider oder Dienstleister
Kunde	nutzende Organisation
Rechteverwaltung	Steuerung von Zugriffen

Begriff	Kurze Erklärung
Konfiguration	technische und fachliche Einstellung eines Dienstes

IHK-sichere Kurzformulierung

IaaS, PaaS und SaaS sind Cloud-Service-Modelle, die beschreiben, welche IT-Schichten vom Anbieter bereitgestellt werden und welche der Kunde selbst verwaltet. Bei IaaS stellt der Anbieter virtuelle Infrastruktur wie Server, Speicher und Netzwerke bereit, während der Kunde Betriebssystem, Anwendungen, Daten und Sicherheit innerhalb der Systeme verwaltet. Bei PaaS stellt der Anbieter zusätzlich Plattform und Laufzeitumgebung bereit, sodass der Kunde sich stärker auf Anwendung und Daten konzentriert. Bei SaaS nutzt der Kunde eine fertige Anwendung, bleibt aber für Benutzer, Rollen, Berechtigungen, Daten, Freigaben und sichere Konfiguration verantwortlich. Das Shared Responsibility Model beschreibt diese geteilte Verantwortung.

Merksätze

IaaS = Infrastructure as a Service.

PaaS = Platform as a Service.

SaaS = Software as a Service.

IaaS liefert virtuelle Infrastruktur.

PaaS liefert Plattform für Anwendungen.

SaaS liefert fertige Software.

IaaS gibt viel Kontrolle.

SaaS nimmt viel Betrieb ab.

PaaS liegt dazwischen.

Cloud-VM muss Kunde meist selbst patchen.

PaaS-Anwendung bleibt Kundenthema.

SaaS-Rechte bleiben Kundenthema.

Daten bleiben fast immer Kundenthema.

Benutzer und Rollen bleiben fast immer Kundenthema.

Shared Responsibility konkret prüfen.

Cloud-Sicherheit ist gemeinsame Verantwortung.

Managed Service reduziert Betriebsaufwand.

Self-Managed erhöht Verantwortung.

Lift and Shift verschiebt,
modernisiert aber nicht automatisch.

Cloud-native nutzt Cloud-Prinzipien.

Serverless heißt keine Serververwaltung durch Kunden.

FaaS führt Funktionen bei Ereignissen aus.

XaaS ist Sammelbegriff.

Frage immer:
Infrastruktur,
Plattform
oder fertige Anwendung?

Weniger technischer Betrieb heißt nicht:
keine Administration.

Cloud ist nicht automatisch sicher.

Cloud ist nicht automatisch günstiger.
