WHITEPAPER

WENN CLOUD – DANN SICHER.

Aspekte sicherer Cloud-Anwendungen

Der Grundstein für eine sichere, stabile und gut betreibbare Cloud-Anwendung wird in den frühen Phasen gelegt:
Beim sorgfältigen Anforderungsmanagement und beim Entwurf der passenden Architektur. Ein tiefes, präzises Verständnis und Erfahrung mit den eingesetzten Technologien sind dabei wie so oft der Schlüssel.

V1 – November 2023

exXcellent solutions

trust in quality.

I DER START IST ENTSCHEIDEND.

Denkt man an die Erstellung neuer Business-Anwendungen, dann sind die modernen Cloud-Technologien mit ihren Vorteilen im Hinblick auf Skalierung und einfachen Betrieb omnipräsent. Die Sicherheit solcher Anwendungen wird dabei als Grundbedingung vorausgesetzt.

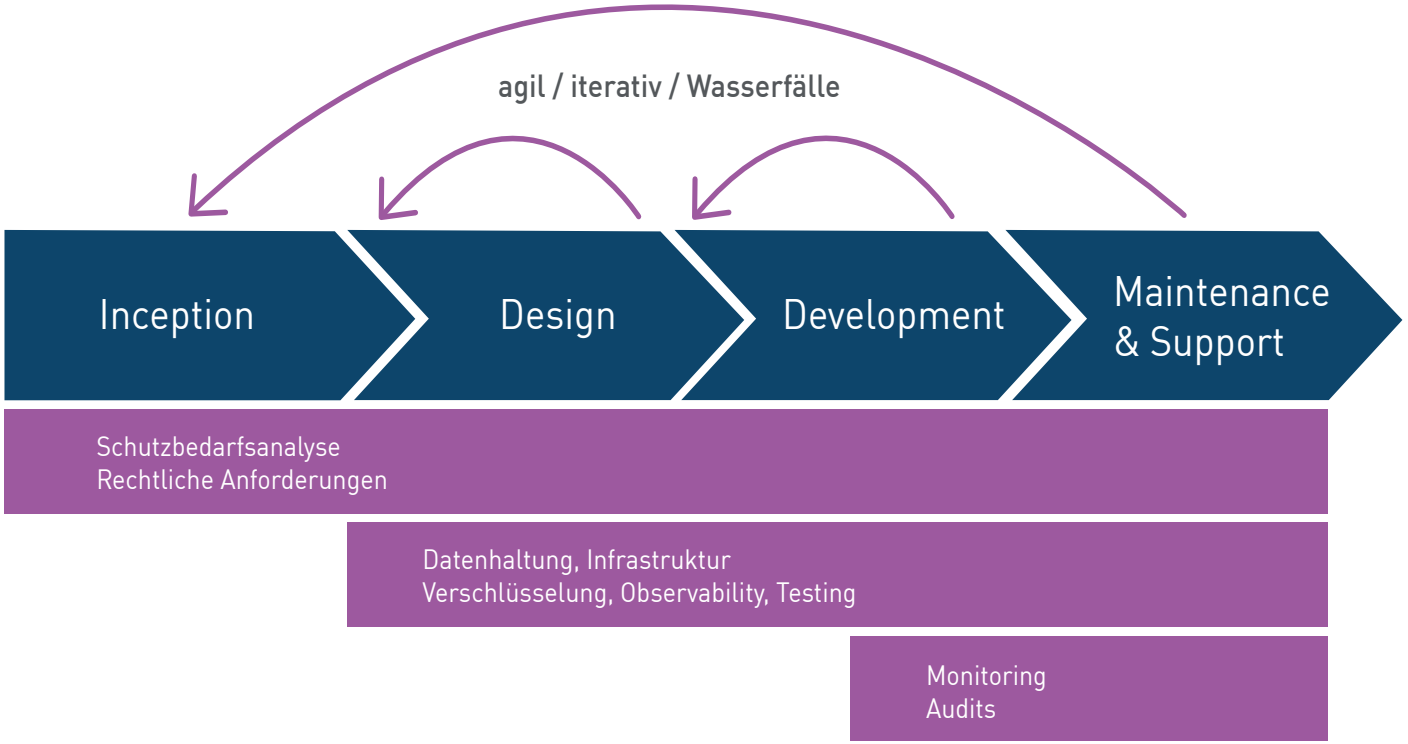
Die attraktiven Möglichkeiten bringen jedoch eine Komplexität mit, die weit höher ist als bei klassisch im eigenen Rechenzentrum betriebenen Anwendungen. Die Krux daran ist, dass diese Komplexität nicht direkt ersichtlich ist, sondern sich in zahlreichen Frameworks, Abstraktionsschichten und hinter Konfigurationswerkzeugen der jeweiligen Cloud-Anbieter verbirgt. Zwei elementare Bereiche, die dabei immer wieder als nachrangig betrachtet werden, sind Sicherheit und Datenschutz.

In diesem Whitepaper zeigen wir, wie Cloud-Anwendungen sicher entwickelt und betrieben werden können und welche Aspekte in Bezug auf Sicherheit und Datenschutz zu beachten sind.

Dabei zeigt sich, dass der Grundstein für eine sichere, stabile und gut betreibbare Cloud-Anwendung in den frühen Phasen gelegt wird: Beim sorgfältigen Anforderungsmanagement und beim Entwurf der passenden Architektur. Ein tiefes, präzises Verständnis und Erfahrung mit den eingesetzten Technologien ist dabei der Schlüssel. Wir teilen unser Know-How mit Ihnen, für einen sicheren Weg in die Cloud.

trust in quality.

II WORAUF ES ANKOMMT – DIE SCHLÜSSELSTELLEN



Entwurf und Entwicklung von Cloud-Anwendungen lassen sich in dieselben Phasen einteilen, wie jedes andere Software-Entwicklungsprojekt. Dabei gibt es in jeder Phase bestimmte Themen, die wir im Folgenden genauer betrachten.

trust in quality.

KAPITEL 1

Anforderungsmanagement: Wissen, was wichtig ist

1.1 Die Architektur nachvollziehbar aus Anforderungen gestalten

Der Begriff der „sicheren Cloud-Anwendung“ suggeriert, dass man lediglich ein paar Punkte richtig machen muss und danach eine komplett sichere Anwendung hat. Absolute Sicherheit ist jedoch nicht erreichbar und ab einem gewissen Sicherheitsniveau treiben weitere Verbesserungen den Aufwand und damit auch die Kosten drastisch in die Höhe.

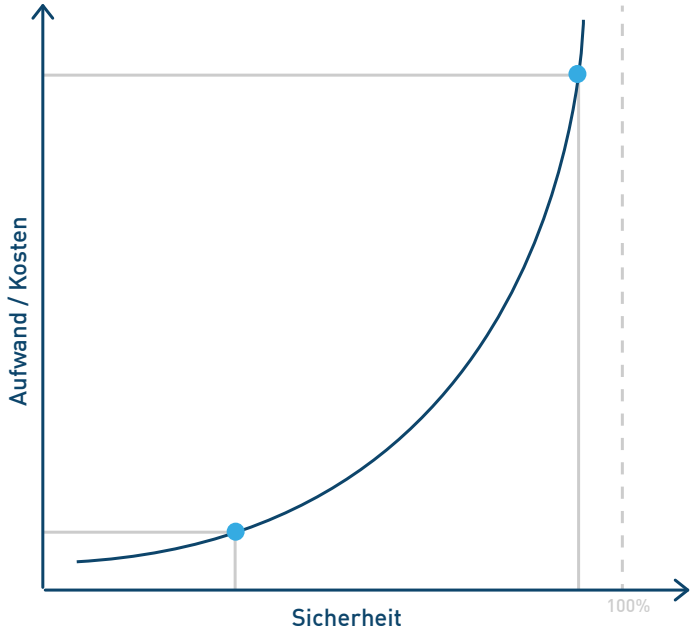
Mit der „sicheren Cloud-Anwendung“ meinen wir im Folgenden deshalb immer eine „Cloud-Anwendung mit angemessener Sicherheit“.

Verstehen Sie die Anforderungen an Ihre Anwendung, um eine optimale Architektur zu erstellen.

Die Kunst besteht darin, die Domäne, die Anforderungen und die Kritikalität der Anwendung sehr gut zu verstehen und auf dieser Basis intelligent die notwendigen und sinnvollen Maßnahmen zu ergreifen.

Werden Anforderungen zu spät oder nicht erkannt, entstehen oft große Aufwände für Architekturänderungen. Irrelevante Anforderungen treiben die Aufwände und Komplexität in die Höhe.

In der Phase der Inception wird auf Basis der Rahmenbedingungen die Architektur der künftigen Anwendung entworfen. Dabei helfen sogenannte ADRs (Architectural Decision Records). In ihnen wird dokumentiert, aufgrund welcher Abwägungen Architekturentscheidungen getroffen wurden. Dadurch ist auch die Angemessenheit der ergriffenen Maßnahmen später nachvollziehbar und mögliche Fehlentscheidungen werden frühzeitig erkannt.



1.2 Übergreifende und rechtliche Anforderungen kennen

Um mit der DSGVO nur eine zu nennen, spielen je nach Domäne und Art der verarbeiteten Daten, häufig regulatorische Aspekte eine Rolle. Versäumt man deren Berücksichtigung bei der Architektur der Anwendung, werden Änderungen mit fortschreitendem Projekt teuer und zeitaufwändig.

Bei personenbezogenen Daten ergeben sich dadurch u.a. folgende rechtliche Konsequenzen:

- Die gesetzlichen Anforderungen der DSGVO müssen umgesetzt sein
- Ein Auftragsverarbeitungsvertrag (AVV) muss bei Unternehmen zwischen nutzendem Unternehmen und Anbieter abgeschlossen werden
- Die Einhaltung der DSGVO liegt eigentlich in der Verantwortung des Auftraggebers, nicht beim Cloud-Betreiber. Der AVV fordert jedoch die DSGVO-Konformität, sodass schlussendlich doch der Cloud-Betreiber in der Pflicht ist. Daraus entstehen dann wiederum eine regelmäßige Kontrollpflicht und die Notwendigkeit von Zertifizierungen.

Weitere Beispiele für speziell regulierte Domänen sind:

- Die kritischen Infrastrukturen (KRITIS)
- Im öffentlichen Sektor die Anforderungen an Barrierefreiheit (BITV 2.0) sowie Standards des BSI
- Der von der BaFin (Bundesanstalt für Finanzdienstleistungsaufsicht) regulierte Bereich
- Der PCI DSS (Payment Card Industry Data Security Standard), für Anwendungen, welche Zahlungsverkehr über Kreditkarten abwickeln. Die Anforderungen daraus „vererben“ sich über die Schnittstellen.
- Branchenspezifische Verordnungen, wie z.B. im Gesundheitswesen oder die Patentanwaltsordnung im Patentwesen.

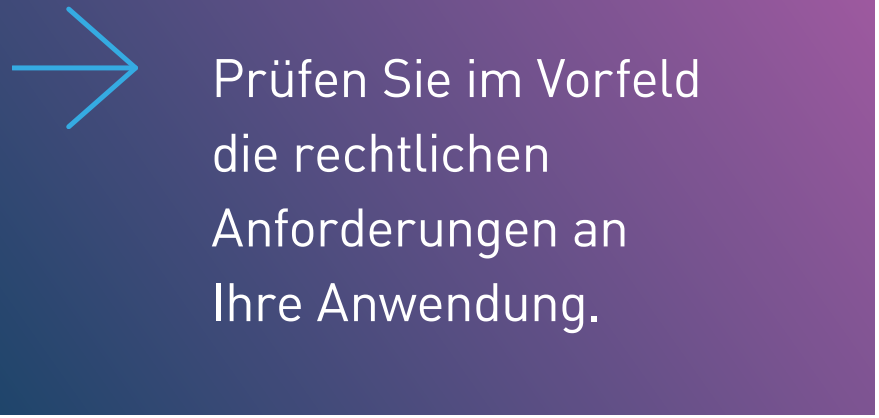
1.2 Übergreifende und rechtliche Anforderungen kennen

Ein Großteil der Regulierung ist dabei national, und muss daher für jeden Zielmarkt untersucht werden. Grenzüberschreitende Transaktionen bergen selbst innerhalb der EU, je nach Fall, auch so manche Überraschung.

Neben behördlicher Regulierung gibt es auch in den Unternehmen unterschiedliche Vertraulichkeitsstufen von bestimmten Daten, aus welchen sich ebenfalls zusätzliche Anforderungen ergeben können. Essenziell ist, diese Randbedingungen direkt in der Konzeptionsphase zu entdecken. Hierzu empfiehlt es sich, frühzeitig dedizierte Branchenexperten einzubeziehen. Sind die rechtlichen und übergreifenden Randbedingungen klar, kann die geplante Anwendung dagegen geprüft werden.

Der amerikanische „Cloud Act" birgt hier Konfliktpotenzial mit der europäischen DSGVO. Dieser regelt, dass US-amerikanische Cloud-Betreiber ggf. Daten herausgeben müssen, welche nicht in US-Rechenzentren gespeichert sind. Es empfiehlt sich, den konkreten Einzelfall mit entsprechenden Experten zu betrachten.

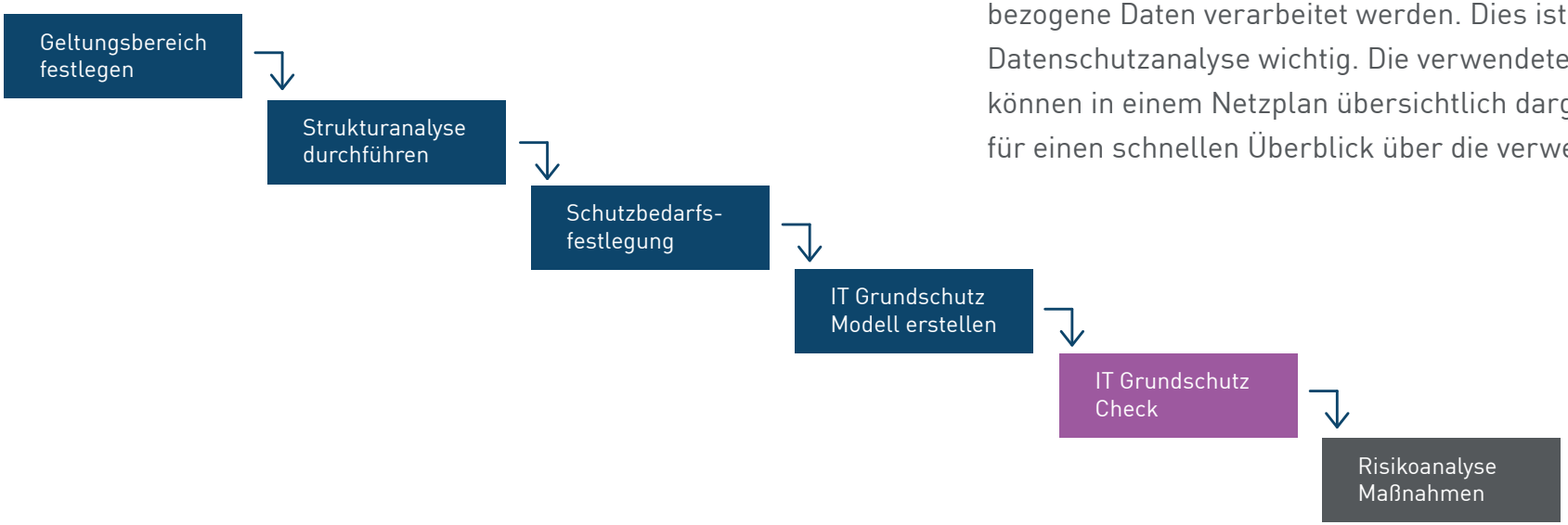
Ist schließlich die Cloud-Anwendung umgesetzt, müssen diese Rahmenbedingungen kontinuierlich überwacht werden. Ein Beispiel gibt hier wieder die DSGVO, die fordert, dass die Sicherheit der Verarbeitung dem Stand der Technik entsprechen muss. Waren dies vor einigen Jahren bspw. bei der Authentifizierung noch Username und Passwort, so ist es heute eine 2-Faktor-Authentifizierung.



1.3 Risiken und Schutzbedarf ermitteln

Um die Sicherheitsanforderungen an die Cloud bzw. den Cloud-Anbieter zu definieren, muss zuerst der Schutzbedarf der Anwendung ermittelt werden. Mithilfe von Analysen, lassen sich geeignete Maßnahmen für einen sicheren Betrieb ableiten.

Eine bewährte Methode zur Feststellung des Schutzbedarfes, ist die BSI-Grundschutzanalyse. IT-Systeme, Geschäftsprozesse, Daten und viele weitere Aspekte werden dabei begutachtet und auf Risiken überprüft. Anhand der Ergebnisse lassen sich die Sicherheitsanforderungen an die Applikation, die Cloud und den Betreiber ablesen.



Geltungsbereich bzw. Informationsverbund

Der Informationsverbund beinhaltet u.a. die notwendigen Geschäftsprozesse, Organisationseinheiten und die Infrastruktur. Alle Bestandteile müssen mit den passenden Grundschutz-Bausteinen [1] abgesichert werden.

Strukturanalyse

In der Strukturanalyse wird das Zusammenspiel der einzelnen Komponenten des Informationsverbunds analysiert und dokumentiert, welche Geschäftsprozesse mit welchen Informationen verarbeitet werden. Dabei wird geprüft, ob auch personenbezogene Daten verarbeitet werden. Dies ist später für die Datenschutzanalyse wichtig. Die verwendeten IT-Systeme können in einem Netzplan übersichtlich dargestellt werden, für einen schnellen Überblick über die verwendeten Systeme.

1.3 Risiken und Schutzbedarf ermitteln

Schutzbedarfsfeststellung

In der Schutzbedarfsfeststellung wird ermittelt, welcher Schutz für die Daten, Geschäftsprozesse und IT-Systeme notwendig ist. Dazu wird geprüft, welche Schäden entstehen können, wenn die Vertraulichkeit, Verfügbarkeit oder die Integrität beeinträchtigt sind. Für eine realistische Einschätzung sollten vorher definierte Schutzbedarfskategorien[2] eingesetzt werden.

Modellierung

Mit den Informationen aus der Strukturanalyse und Schutzbedarfsfeststellung wird nun mit Hilfe der BSI-Bausteine aus dem IT-Grundschutz-Kompendium ein IT-Grundschutz-Modell für den Informationsverbund aufgebaut. Es gibt prozessorientierte und systemorientierte Bausteine, die passend zur geplanten Anwendung ausgewählt werden können. Für den Cloud-Betrieb gibt es spezielle Bausteine (z.B. „Kubernetes“, „Relationale Datenbanken“, „Webserver“ etc.), die verwendet werden sollten.

IT-Grundschutz-Check

Auf Basis der Modellierung wird nun der Grundschutz-Check für die einzelnen Bausteine durchgeführt und ihre Anforderungen überprüft. Hieraus lassen sich die Anforderungen an die Anwendung, die Cloud und den Cloud-Anbieter ableiten.

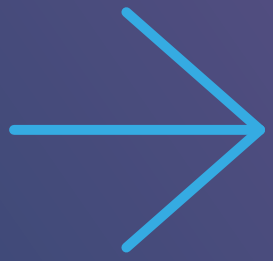
Risikoanalyse

Sollte sich herausstellen, dass nicht alle notwendigen Anforderungen aus den Bausteinen umgesetzt werden können oder teilweise ein erhöhter Sicherheitsbedarf besteht, sollte für diese Punkte eine Risikoanalyse durchgeführt werden. Falls notwendig, können daraus entsprechende Maßnahmen definiert werden, die bei der Umsetzung und dem Betrieb der Applikation beachtet werden müssen.

Zusätzlich zum Grundschutz wird ein Datenschutzkonzept erarbeitet, um die Verarbeitung personenbezogener Daten zu überprüfen. Vorsicht ist geboten, bei besonders schützenswerten Daten. Hier gelten strengere Vorschriften.

Ermitteln Sie den Schutzbedarf mithilfe einer Grundschutzanalyse oder nutzen Sie ein bestehendes ISMS.

Eine Grundschutzanalyse ist nur eine Möglichkeit, die Sicherheitsanforderungen zu bestimmen. Existiert im Unternehmen ein **Informationssicherheitsmanagementsystem (ISMS)** können auch die dort enthaltenen Prozesse und Werkzeuge genutzt werden, um eine entsprechende Analyse durchzuführen.



KAPITEL 2

Infrastruktur: Sichere Umgebung wählen

2.1 Sicherheitsbewertung des Cloudanbieters

Aus den Risiken und dem identifizierten Schutzbedarf sowie den relevanten rechtlichen Rahmenbedingungen, sollte vor Entwicklungsbeginn der angedachte Cloud-Anbieter geprüft werden.

Meist kann dies anhand entsprechender Zertifizierungen leicht nachgewiesen werden:

- Als Mindeststandard empfehlen wir ISO27001 compliance, was bedeutet, dass der Anbieter ein professionelles Informationssicherheitsmanagementsystem etabliert hat.
- Im Optimalfall ist auch ein ISO27018 Zertifikat und ein BSI Cloud Computing C5 Prüfungsnachweis vorhanden.
- Darüber hinaus sollten Sie auf notwendige branchenspezifische Zertifizierungen achten.

Durch entsprechende Zertifizierungen wird es deutlich einfacher, die Anforderungen für die eigenen Zertifizierungen zu erfüllen und den Nachweis für die Erfüllung der gesetzlichen Anforderungen zu erbringen.

2.2 Shared Responsibility: Zuständigkeiten

Ein Schlüsselkonzept bei den gängigen Cloudanbietern, ist die „shared responsibility". Shared responsibility bedeutet, dass es eine klare Abgrenzung gibt, bis zu welcher der Cloudanbieter für die Sicherheit und Updates etc. verantwortlich ist. Für alles darüber hinaus liegt die Verantwortlichkeit beim Kunden.

Klären Sie die Zuständigkeiten des Anbieters und Ihre eigenen Verantwortlichkeiten.

Wo die Grenze der Verantwortlichkeit liegt, hängt von den genutzten Services ab. Wichtig ist, die Abgrenzung genau zu verstehen, um blinde Flecken zu vermeiden.

Ein Beispiel hierzu sind virtuelle Maschinen mit Betriebssystem. Hier wählt der Kunde bei der Konfiguration die Betriebssystemversion, RAM-Größe usw. aus. Anschließend liegt die virtuelle Maschine in seiner Verantwortung, d.h. Sicherheits-Updates vom Betriebssystem sind vom Kunden selbst auszuführen.

Ein Gegenbeispiel ist eine **managed PostgreSQL database**. Hier sind in der Regel die Sicherheitsupdates der PostgreSQL Datenbank durch den Cloud-Anbieter abgedeckt (im Einzelnen muss geprüft werden, was ein Anbieter unter „managed" versteht).

Physische Sicherheit

Die Cloud-Anbieter informieren umfassend über ihre Sicherheitsmaßnahmen. Als Faustregel lässt sich sagen, dass das von den Cloud-Anbietern bereitgestellte Niveau meist deutlich besser ist als das eines firmeninternen Rechenzentrums.

Netzwerksicherheit

Hier sorgen die Cloud-Anbieter dafür, dass die Mechanismen (Subnetz, DMZ, VPN, Firewalls) sicher und stabil funktionieren. Die Konfiguration der Komponenten (Firewall-Regeln, Routing-Regeln) obliegt komplett dem Kunden. Beispielsweise ist der Kunde selbst verantwortlich, dass die Datenbank, und sonstige genutzte Services, nicht von außen zugreifbar sind.

Beim Aufbau einer Cloud-Infrastruktur ist es deshalb erforderlich, alle Einstellungen – auch die Voreinstellungen des Cloud-Anbieters – im Detail zu verstehen und kritisch zu prüfen.

Hier gilt es dann auch bei den anwendungsspezifischen Anpassungen nach dem Least Privilege-Prinzip (also dem Prinzip der am geringsten notwendigen Privilegien) vorzugehen.

2.2 Shared Responsibility: Zuständigkeiten

Datenhaltung

Das Stichwort ist hierbei „Encryption at Rest", d.h. die Verschlüsselung der Datenablage. Hierzu haben fast alle Cloudanbieter Angebote, teilweise ist auch die Verschlüsselung unter Verwendung eigener Schlüssel/Zertifikate möglich.

Intrusion Detection

Besteht eine automatisierte Angriffserkennung/Prävention als Anforderung für die eigene Cloud-Anwendung, so ist zunächst ein detailliertes Verständnis der aufgebauten Umgebung, der Sicherheitsanforderungen und für das Zusammenspiel des IDS (Intrusion Detection System) mit den Komponenten erforderlich. Insbesondere die großen Cloud-Anbieter haben Zusatzangebote, auf die zurückgegriffen werden kann. Die genaue Funktionsweise und ob diese die Sicherheitsanforderungen abdeckt, muss jedoch im Einzelfall betrachtet werden.

Container

Container (Docker mit ggfs. Kubernetes Managementumgebung) sind ein sehr komfortables Mittel, um Infrastruktur bereitzustellen. Beim Einsatz muss jedoch darauf geachtet werden, dass alles innerhalb der Container in Verantwortung des Kunden liegt. Dies beinhaltet zwei Aspekte:

- Auswahl des sogenannten Base-Image, also der Vorlage für den Container. Hierbei muss auf die Aktualität aller Bestandteile und Vertrauenswürdigkeit des Maintainers geachtet werden.
- Bei eigenen Anpassungen empfiehlt es sich, die Container möglichst schlank zu halten – alles, was im Container nicht vorhanden ist, birgt auch keine Angriffsmöglichkeiten.

Einfache, klare Strukturen

Trotz der vielen Möglichkeiten der Cloud-Anbieter ist es essenziell, die Architektur der Infrastruktur im Detail mit allen ihren Aspekten zu verstehen und die Design-Entscheidungen zu dokumentieren.

2.3 Verschlüsselung von Kommunikation

Die Verschlüsselung von Datentransfers im Internet entspricht dem aktuellen Stand der Technik, um die Vertraulichkeit von übermittelten Daten zu gewährleisten. Aber auch innerhalb einer privaten Cloud-Umgebung sollten Datenverbindungen bekannt und durch Verschlüsselung geschützt sein.

Bereits beim Entwurf des Systems muss darauf geachtet werden, welche Arten von Daten von welchen Komponenten verarbeitet werden. Ebenso gilt es herauszufinden, welche Systeme auf welchen Wegen miteinander kommunizieren müssen.

Ein besonderes Augenmerk verdienen **Managed Services**, die von Cloudanbietern entwickelt, betrieben sowie verwaltet werden und innerhalb der eigenen Anwendung zum Einsatz kommen. Beispiele sind **Managed Databases, Managed Kubernetes-Clusters** oder auch andere externe Services. Da diese Dienste außerhalb der eigenen Netzwerkumgebung betrieben werden, findet die Kommunikation zwangsweise über ein fremdes Netzwerk statt. Hier muss daher eine SSL/TLS gesicherte Verbindung oder ein VPN genutzt werden.

Ein weiterer Punkt ist die Kommunikation zwischen allen Komponenten innerhalb der eigenen Cloud-Umgebung (Services, Clusters, externe Dienste). In der Vergangenheit wurde oftmals die riskante Annahme getroffen, dass in bereits geschützten Zonen keine Verschlüsselung notwendig ist. Wenn jedoch ein Angreifer in diese Zone vorgedrungen ist, stehen ihm alle Türen offen. Deshalb werden moderne Cloud-Umgebungen nach dem Prinzip „Zero Trust“ aufgebaut und auf allen Ebenen abgesichert.

Eine mögliche Umsetzung ist der Einsatz von sogenannten **Service Meshes** oder **mtls („mutual TLS“)**, über die eine verschlüsselte und nachvollziehbare Kommunikation zwischen Cluster-Knoten gesteuert und abgesichert werden kann. Die Komplexität einer solchen Lösung darf jedoch nicht unterschätzt werden.

Schützen Sie Managed Services außerhalb des eigenen Netzwerks durch eine gesicherte Verbindung.

2.3 Backups

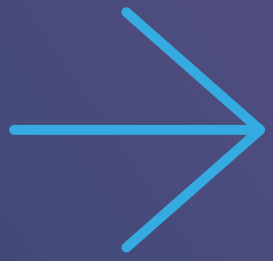
Auch wenn die Cloud-Anbieter sehr hohe Werte für die Verfügbarkeit und Ausfallsicherheit der Komponenten zusichern, ist der Nutzer der Cloud-Umgebung weiterhin für die Datensicherheit im Sinne von Backups nach seinen Anforderungen verantwortlich.

Durch fehlerhafte Algorithmen, Systemausfälle, Angriffe etc. können jederzeit Daten verändert, überschrieben oder gelöscht werden. Auch Konfigurationsfehler der Cloud-Anbieter können dazu führen, dass Daten, die redundant an unterschiedlichen Standorten gespeichert werden, unwiderruflich verloren gehen.

Schützen Sie Daten nicht nur vor Angriffen, sondern auch vor Verlust.

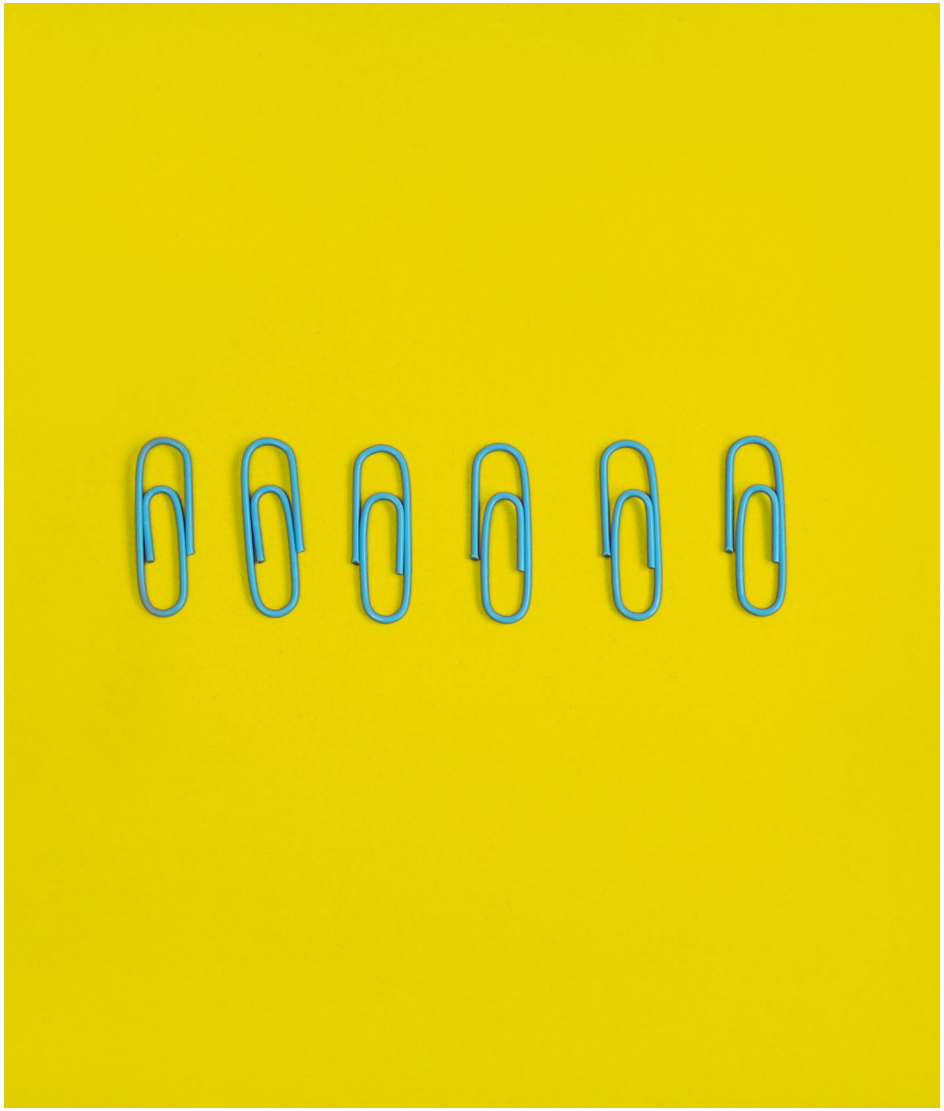
Es ist daher zwingend erforderlich, sich bei der Konzeption einer Cloud-Anwendung mit den Anforderungen an Datensicherheit und entsprechenden Sicherungs- und Recoverystrategien zu befassen.

Bei allen gängigen Cloud-Anbietern gibt es eingebaute Möglichkeiten zur automatischen Datensicherung, die für eine passgenaue Sicherungs- und Recoverystrategie mit eingesetzt werden können.



KAPITEL 3
Anwendung: Sichere Strukturen schaffen

3.1 Datenkategorien und besondere Anforderungen



Anhand der im Anforderungsmanagement (vgl. Kapitel 1) gesammelten Aspekte, muss die Anwendung, die betroffenen Daten passend zu den jeweiligen Anforderungen verarbeiten.

Hierzu bietet es sich an, die Daten im Domänenmodell entsprechend zu kategorisieren und für jede der Kategorien über ADRs zu dokumentieren, wie die jeweiligen Anforderungen in der Architektur berücksichtigt werden.

> Steuern Sie die Verarbeitung der Daten mithilfe von Kategorien.

Für die einzelnen Datenkategorien gibt es Best Practices, an denen man sich orientieren kann.

Bei sehr hohen Datenschutz-Anforderungen besteht die Option, die Daten bereits client-seitig zu verschlüsseln, sodass die Backendanwendung bzw. die Cloud lediglich verschlüsselte Daten erhält.

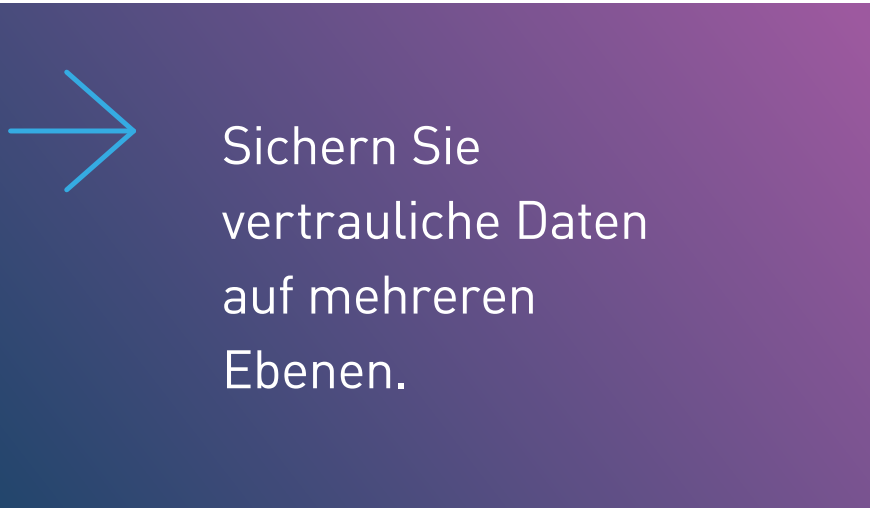
3.2 Trennung von Daten bei mandantenfähigen Systemen

Viele Cloud-Anwendungen sind mandantenfähig aufgebaut. Eine Anwendung wird hier von verschiedenen Nutzern verwendet, die unterschiedlichen Nutzer können jedoch ihre Daten nicht gegenseitig einsehen. Würden die Nutzer aufgrund einer Schwachstelle oder eines Fehlers plötzlich Daten anderer sehen, könnte dies Geschäftsgeheimnisse oder hochvertrauliche Informationen offenlegen. Die Trennung der Daten der einzelnen Mandanten ist somit eine sehr kritische Anforderung, welche sehr stark abgesichert werden muss.

Hierzu gibt es mehrere Lösungsansätze. Welcher Ansatz sich am besten eignet, ist abhängig von den Rahmenbedingungen der Anwendung:

- Welche Daten werden verarbeitet?
- Gibt es mandantenübergreifende Daten, welche nur einmal gepflegt werden sollen?
- Wie viele Mandanten nutzen das System und wie sehen die Wachstumsprognosen aus?

Jeder Ansatz hat eigene Vorteile aber auch Anforderungen an Betrieb und Wartung der Anwendung.



3.2 Trennung von Daten bei mandantenfähigen Systemen

Eigene Instanz der Anwendung, eigene Datenbank:

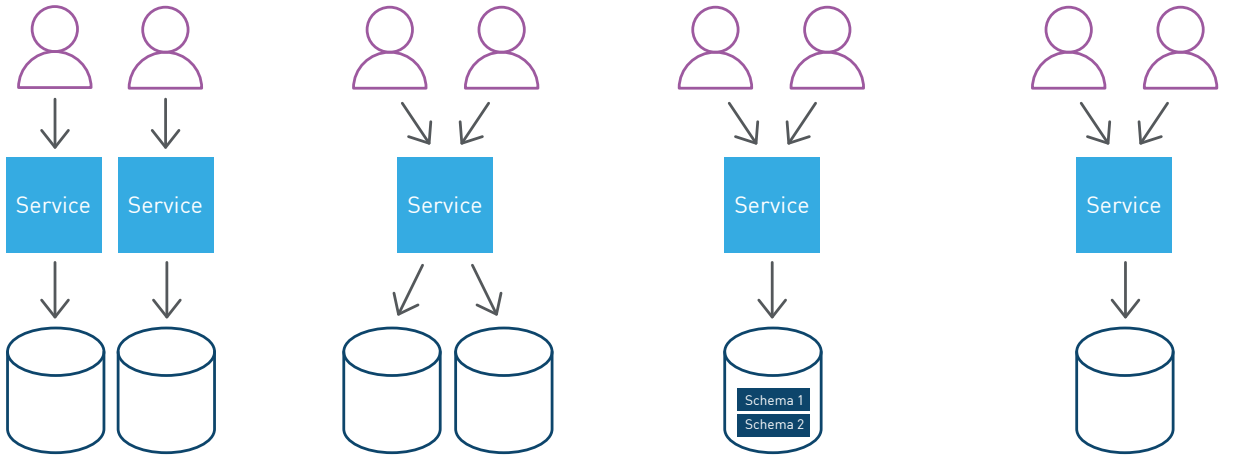
Hier sind die Daten strikt getrennt und eine Vermischung sehr unwahrscheinlich. Das Risiko von Fehlzugriffen ist am geringsten. Gleichzeitig bedeutet es aber durch die Vervielfachung von Instanzen auch den höchsten Aufwand. Ein Nebenaspekt ist, dass durch die exklusive Nutzung von Instanzen für jeweils nur einen Mandanten auch explizite Performancezusagen für die Mandanten möglich sind.

Geteilte Instanz der Anwendung mit eigener Datenbank oder eigenen Schemata:

Diese Zwischenvarianten bieten auf Datenbankebene eine saubere Trennung, bergen aber weiterhin das Risiko, dass Daten in der Anwendung selbst gemischt werden. Auch aufwands- und kostenseitig bewegt sich das Modell zwischen den beiden anderen.

Geteilte Instanz der Anwendung, eine Datenbank, gemeinsames Schema:

Diese Variante bedeutet, dass die Trennung der Mandanten rein über die Art der Zugriffe durch die Anwendung gewährleistet wird. Dies ist die effizienteste Variante, jedoch auch diejenige mit dem höchsten Risiko.



Eigene Instanz der Anwendung, Eigene Datenbank

Geteilte Instanz der Anwendung, Eigene Datenbank je Mandant

Geteilte Instanz der Anwendung, Eine Datenbank getrennte Schemata

Geteilte Instanz der Anwendung, Geteilte Datenbank

3.3 Authentifizierung und Autorisierung

Authentifizierung beantwortet die Frage: „Wer?“. Sie stellt sicher, dass ein Aufruf bzw. eine Anfrage von dem Benutzer oder dem Service kommt, der behauptet dieser zu sein. In der Regel muss dazu die Identität nachgewiesen werden, etwa durch ein Passwort, welches nur dem tatsächlichen Benutzer bekannt sein darf. **Autorisierung** dagegen beantwortet die Frage: „Darf der Benutzer das?”. Sie bezieht sich immer auf eine bestimmte Art von Zugriff und stellt sicher, dass für diese Art von Zugriff auch die Berechtigung vorliegt.

Der wichtigste Grundsatz beim Thema Authentifizierung und Autorisierung ist: **„Keine Implementierungen von Identitäts- und Zugriffsverwaltung selbst entwickeln!“**

Dies umzusetzen ist leicht, denn die Auswahl an Industrie standards und passenden hochwertigen, gut gepflegten Implementierungen ist groß. Sowohl Open Source-Lösungen, kommerzielle Software zur Integration ins eigene System als auch gehostete Produkte stehen zur Verfügung. Diese bieten ausgereifte, erprobte Konzepte und Lösungen zu Themen wie Credential Management, Mehrfaktorauthentifizierung, User Self Service usw.

Das Cloud-Umfeld bietet eine optimale Ausgangsbasis: Viele Cloudanbieter stellen eigene, gut integrierte, einfach nutzbare Services zur Verfügung, wie z.B. AWS Cognito oder Azure AD, welche neben den meisten Standard-Aufgaben noch weitaus mehr abdecken. Aber auch im Anbieter-unabhängigen Bereich gibt es passende und bewährte Standardimplementierungen, wie z.B. Keycloak, auf die man zurückgreifen kann.

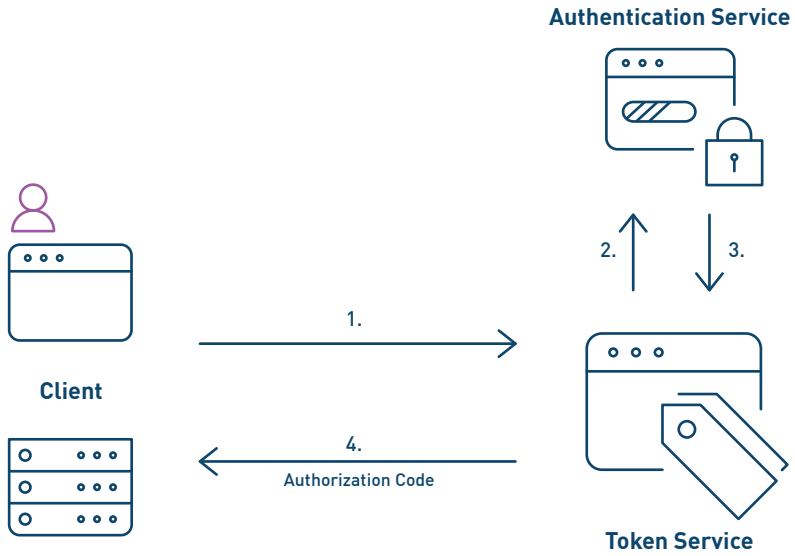
Im Folgenden gehen wir auf die beiden Aspekte der Authentifizierung und Autorisierung tiefer ein.

Entwickeln Sie keine Identitäts- und Zugriffsverwaltungen selbst!

3.3.1 Authentifizierung

Authentifizierungsverfahren ändern sich, und sind nicht Teil der fachlichen Software-Lösung. Der Klassiker “Benutzername und Passwort” war bspw. lange Zeit Standard, weist aber grundsätzliche Schwächen auf und wird deshalb sukzessive abgelöst.

Die zukunftssichere Lösung ist eine Authentifizierung außerhalb der Anwendung, über einen Dienst, welcher das notwendige Spektrum an Authentifizierungsmethoden bietet, z.B. **Zwei- oder Mehr-Faktor-Authentifizierung** mit Hardwaretokens, Einmalpasswörtern, Apps etc.



> Nutzen Sie externe Authentifizierungsdienste, die sich mit dem Stand der Technik weiterentwickeln, für dauerhaft zeitgemäße Sicherheit.

Oft ist ein solcher zentraler Authentifizierungsservice in der Organisation bereits vorhanden (Google-Accounts, Active Directory, etc.) und in übergreifende Prozesse eingebunden, z.B. beim Onboarding neuer Mitarbeiter. Neben dem Vorteil, dass den Anwendungen damit die Komplexität der Authentifizierung (Login, Password Reset, Mehrfaktorauthentifizierung) erspart bleibt, ist damit gleichzeitig der Grundstein für ein **Single-Sign-On** innerhalb der Organisation gelegt.

Sollte ein solcher Dienst noch nicht vorhanden sein, oder aus bestimmten Gründen nicht genutzt werden, kann ein Dienst des Cloudanbieters oder auch eine anwendungsspezifische Lösung wie Keycloak genutzt werden.

3.3.2 Autorisierung

Ist ein Benutzer erfolgreich authentifiziert worden, so stellt der Autorisierungsservice ein sogenanntes Autorisierungstoken (eine Art Eintrittskarte) aus. Dieses Token gibt die Anwendung an aufgerufene Services weiter. Die aufgerufenen Services prüfen und entscheiden, ob das jeweilige Token für den angefragten Service-Aufruf „ausreichend“ ist.

Diese Entscheidung kann fallabhängig entweder direkt über im Token enthaltene Informationen (Scopes) getroffen werden oder über einen weiteren Aufruf eines Token-Validierungsservices. Dieser prüft z.B. im Azure AD, ob das Token zu einem Benutzer einer gewissen Benutzergruppe gehört.

Um eine Autorisierung ohne Programmierung querschnittlich sicherzustellen, kommen häufig API-Gateways zum Einsatz. Diese schotten die Services ab und lehnen nicht-autorisierte Service-Aufrufe entsprechend ab. Neben dieser Grundaufgabe bieten API-Gateways viele weitere nützliche Funktionen zum Monitoring, zur Dokumentation etc. Beispiele für **API-Gateways** sind Kong, Cloudflare, Apigee, Amazon API Gateway und Azure API Management.

3.3.3 Authentifizierung bei infrastrukturbezogenen Tätigkeiten

Neben der Authentifizierung und Autorisierung der Endnutzer, spielen in der Cloud auch Sicherheitsaspekte bei administrativen Tätigkeiten an der Infrastruktur eine herausgehobene Rolle. Hierbei gelten dieselben Grundsätze wie bei lokal betriebenen bzw. klassisch gehosteten Anwendungen. Allerdings machen es die Rechtesysteme der Cloudprovider einfacher, die administrativen Zugriffsrechte geeignet zu strukturieren. Auch bieten sie direkt eine Möglichkeit an, vergebene Rechte zu dokumentieren und übersichtlich zu visualisieren, wofür sonst eine gesonderte „Buchhaltung” nötig wäre.

KAPITEL 4

Devops:
Sicherheit in den Prozessen verankern

4.1 Secrets Management

In jedem System gibt es Einstellungen, die einem besonderen Schutz unterliegen müssen. Passwörter für Datenbanksysteme, Zertifikate für eine sichere Kommunikation oder Variablen, die nur von bestimmten Teammitgliedern gesehen werden dürfen. Diese Geheimnisse („Secrets“) müssen für die Anwendung zugreifbar sein, weshalb sie häufig schlecht geschützt in Konfigurationsdateien oder gar im Code verpackt sind.

Secret Management Tools kümmern sich um genau dieses Problem. Jedoch gehört zu einer sicheren Anwendung nicht nur das Benutzen dieser Tools, sondern auch geeignete Prozesse, damit Secrets sicher bleiben:

- Secrets sollten regelmäßig neu erstellt werden.
- Sie sollten nicht von jedem lesbar sein und an so wenig Stellen wie möglich gespeichert werden.
- Es sollte nicht notwendig sein, Secrets manuell zu verschicken, andernfalls muss eine sichere Weitergabe ermöglicht werden.

Diese Herausforderungen können mit verschiedenen Mitteln gelöst werden. Ein grundlegendes und sehr gutes Werkzeug für das persönliche Secret Management einer Person sind Passwort Manager, wie z.B. KeePass, LastPass oder Bitwarden.

Für Cloud-Anwendungen reichen persönliche Secret Management Tools nicht mehr aus. Hierbei werden Secrets teilweise vom Entwicklungsteam und weiteren Beteiligten verwaltet, was zu deutlich mehr Angriffspunkten führt. Hier kommen Secret Engines wie „**Hashicorp Vault**“ oder Secret encryption Tools wie „**Sealed Secrets**“ oder „**Ansible Vault**“ ins Spiel.

Secret Engines ermöglichen eine zentrale Verwaltung von Secrets und decken die oben beschriebenen Prozesse ab.

Secret encryption Tools hingegen ermöglichen lediglich eine sichere Ablage von Secrets. Für die oben beschriebenen Aufgaben müssen dann jedoch manuelle Prozesse implementiert werden.

Implementieren Sie sichere Prozesse mittels Secret Engines.

Da die Art und Weise, wie mit den Secrets umgegangen wird, sich auf die Prozesse in Entwicklung, Integration (Build, CI/CD) und auch auf den Betrieb auswirkt, sollte dieser Aspekt gleich zu Beginn eines Projektes betrachtet und architektonisch festgelegt werden.

4.2 Sicherheit im Entwicklungsprozess / Software-Lifecycle

Cloud-Projekte arbeiten in der Regel mit aktuellen agilen Methodiken und Prozessen. Ein Schlagwort ist dabei **„Continuous Delivery“**, welches kleinere, schnelle Releases propagiert. Als Konsequenz bedarf es einer stringenten, hoch automatisierten Absicherung.

Folgende Punkte helfen, in einer solchen Continuous-Delivery-Umgebung, eine pragmatische Risikoabsicherung vorzunehmen:

- Hohes Basiswissen und Sensibilität der Entwickler für Sicherheitsthemen, z.B. durch geeignete Schulungen, Zertifizierungen etc.
- Wissen über das Risikoprofil der erstellten Anwendung und der Aspekte aus der in Kapitel 2.1.2 beschriebenen Grundschutzanalyse.
- Einsatz von Scannern für das Finden von kritischen Mustern und bekannten Schwachstellen (z.B. CVE-Scanner), die auf öffentliche Listen von Schwachstellen zurückgreifen, sich laufend aktualisieren und oft bereits hinsichtlich Kritikalität klassifiziert sind.
- Hohe automatisierte Testabdeckung bzgl. Penetrationtests (siehe 4.3).
- Prozesse, wie das Entwicklungsteam mit Erkenntnissen hinsichtlich Security umgeht – wann kann das Security Finding verschoben werden, wann ist ein Hotfix/Patch notwendig?
- Dabei unterstützen können auch formale Modelle wie z.B. SAMM 3 (Software Assurance Maturity Model) von OWASP.

Auch wenn Cloud-Anwendungen aus der Perspektive der Sicherheit zunächst sehr angreifbar wirken können, bieten sie auch große Vorteile, die sich durch das Schlagwort **„Infrastructure as Code“** (IaC) ergeben:

- Nachdem sich ganze Umgebungen über IaC textuell oder programmatisch definieren lassen, lassen sich auch die Mechanismen des Scannings und der klassischen statischen Code-Analyse auf ganze Infrastrukturen anwenden.
- Cloud-Umgebungen müssen dadurch nicht von einem Betriebsteam in mühsamer Kleinarbeit wieder aufgesetzt werden, sondern lassen sich per Knopfdruck automatisiert erneut reproduzieren. Dies erleichtert das Aufsetzen weiterer Umgebungen (Test, Preproduction etc.) und ist eine wertvolle Hilfe beim Thema Desaster Recovery.

4.3 Testing

Moderne Cloud-Architekturen und Infrastrukturen sind weitaus komplexer als herkömmliche Anwendungen. Sie bestehen meist aus einer Vielzahl von Microservices, die auf verschiedenen Umgebungen ausgeführt werden können.

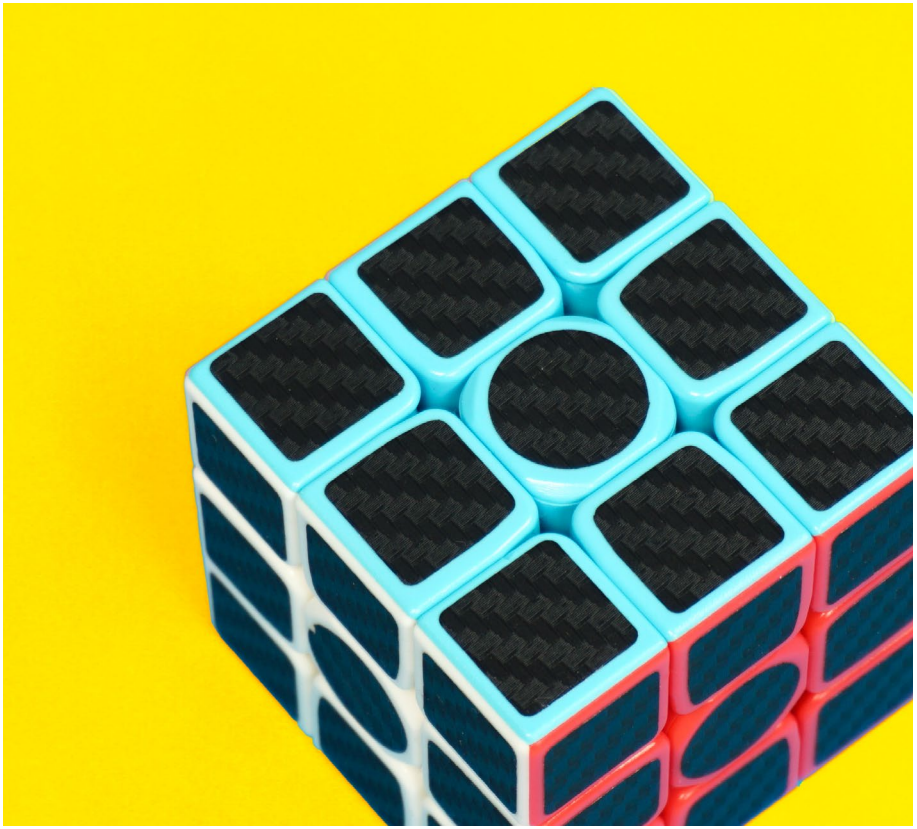
Trotz der Vorteile dieser Architekturen bringt die Komplexität auch zahlreiche Angriffspunkte mit sich. Verstärkt wird das durch die agilen Entwicklungsmodelle, welche zum Ziel haben, viele kleine Features möglichst schnell in den Betrieb zu bringen. Somit muss häufig erneut getestet werden.

Durch automatisierte Tests und Scanner, lässt sich der Arbeitsaufwand jedoch erheblich reduzieren, sowohl zur Absicherung der Funktionalität als auch für typische Angriffsmuster. Für die getesteten Angriffsmuster bietet es sich an, sich an gängigen Listen wie z.B. OWASP Top 10 zu orientieren, und diese mit dem Schutzbedarf (siehe 1.3) abzugleichen.

Ergänzend erkennen Scanner wie Sonarqube auch Muster im Code, welche Schwachstellen einführen.

Werden diese Tests und Scan-Vorgänge vor jedem Deployment automatisiert ausgeführt, kann die Entwicklung noch vor der Auslieferung reagieren und die Cloud-Anwendung hat anschließend einen definierten Grad der Absicherung.

Insgesamt ist der Aufbau von wirksamen und wartbaren Tests gerade im Bereich Security, anspruchsvoll und bedarf einer passenden Methodik. Ohne eine solche Methodik werden die Tests schnell unüberschaubar und/oder müssen oft überarbeitet werden, was den Nutzen schmälert.



4.4 Observability

In verteilten Systemen gibt es für jede Komponente im System meistens mehrere Instanzen. Zu jeder Zeit können weitere Instanzen erstellt oder laufende beendet werden. Continuous Delivery und Multi-Cloud Systeme erschweren es zusätzlich, den Überblick über das System zu behalten.

Um in diesen komplexen Umgebungen weiterhin einzelne Aufrufe nachvollziehen zu können und auch Aussagen über das Gesamtsystem tätigen zu können, ist eine übergreifende Beobachtbarkeit notwendig.

Die wichtigsten Aspekte von Observability im Cloud-Umfeld sind:

- **Metriken:** Metriken sind Datenpunkte, die den Zustand des Systems zu einem bestimmten Zeitpunkt widerspiegeln. Sie helfen dabei, Trends zu identifizieren und Probleme frühzeitig zu erkennen. In einer Cloud-Umgebung werden Metriken von den verschiedenen Komponenten (Service-Container, Datenbanken, etc.) generiert und in einer zentralen Datenbank gesammelt.
- **Logging:** Logging ist die Aufzeichnung von Ereignissen und Aktivitäten in einem System. Logs helfen bei der Fehlersuche und -behebung, da sie den Entwicklern Informationen über den Zustand und die Aktivitäten der Anwendung liefern.
- **Tracing:** Tracing ist die Fähigkeit, den Weg einer Anfrage durch die verschiedenen Komponenten der Anwendung zu verfolgen. Es hilft bei der Identifizierung von Engpässen und Fehlern in der Anwendung.
- **Alerting:** Alerting benachrichtigt Betreiber und Entwickler über Probleme in einer Anwendung oder einem System. Die Alerts basieren auf den verschiedenen Metriken und Logs der Anwendung.
- **Visualisierung:** Visualisierung hilft dabei, die gesammelten Daten und Metriken durch verständliche Diagramme und Dashboards besser greifbar zu machen. Die Ausgestaltung orientiert sich an den Anforderungen des Projekts und der Art der Anwendung, um einen umfassenden Überblick über den Zustand der Anwendung zu erhalten.

Insgesamt ist dies ist ein wichtiger Aspekt des Betriebs von Cloud-Anwendungen und ermöglicht es schnell auf Probleme zu reagieren, die Leistung zu optimieren und die Zuverlässigkeit von Anwendungen sicherzustellen.

Geeignete Werkzeuge sind Frameworks wie OpenTelemetry in Verbindung mit Auswertungs- und Visualisierungswerkzeugen wie Grafana/Prometheus und viele mehr.

III FAZIT

Gut aufgesetzte Cloud-Anwendungen glänzen mit Eigenschaften, die sowohl aus Business- als auch Betriebssicht eine neue Liga eröffnen: Flexibilität, dynamische Skalierbarkeit, Performance, lückenloses Monitoring, Innovationsgeschwindigkeit, Continuous Everything, um nur paar Vorteile zu nennen.

Technisch gesehen sind Cloud-Anwendungen normale Systeme, die jedoch in einer sehr anspruchsvollen Umgebung eingebettet sind und betrieben werden. Dies bringt auch für die Cloud-Anwendung, insbesondere für die Sicherheit, Komplexitäten mit sich. Für eine sichere Entwicklung sind folgende Aspekte zentral:

- Wissen über die Domäne („know your business"), Zielsetzungen der Anwendung mit den Zusammenhängen und allen relevanten rechtlichen Rahmenbedingungen
- fundiertes Wissen und Erfahrung mit den eingesetzten Technologien
- tiefes Verständnis der genutzten Services / virtuellen Infrastruktur des Cloud-Anbieters
- fundiertes Security-Wissen

All dies beeinflusst maßgeblich die Grundarchitektur des Gesamtsystems. Um Projektrisiken zu vermeiden, gilt es bereits zu Beginn, diese Aspekte zu betrachten und zu dokumentieren und damit den Grundstein für eine sichere Cloud-Anwendung zu legen.

Wissen ist dabei der Schlüssel. Neben den Fachbereichen und den technischen Architekten ist es daher ratsam, gezielt auf externe Experten (technisch, rechtlich, Security etc.) zurückzugreifen. Denn nur wer die enorme Komplexität von Cloud-Anwendungen durchschaut, kann diese systematisch und auf allen Ebenen absichern.

Wer in klare Anforderungen und eine dazu passende Ziel-Architektur investiert, erkennt Risiken und Probleme frühzeitig und kann die Potentiale der Cloud voll ausschöpfen – mit Sicherheit!

QUELLEN

[1] Übersicht aller vorhandenen BSI-Bausteine (zuletzt abgerufen am 25.10.2023)

[2] Definition von Schutzbedarfskategorien (zuletzt abgerufen am 25.10.2023)

trust in quality.

Einzigartige Kunden verdienen maßgeschneiderte Lösungen. Seit über 20 Jahren bauen wir Softwarelösungen, die zukunftsfähig sind und die individuellen Prozesse & Bedürfnisse unserer Kunden von der Beratung bis zu Operations abdecken. Wir begleiten vom ersten Gedanken, bis weit in den Betrieb der neuen Lösung. Wir übernehmen Verantwortung und stehen mit unserem Namen für Qualität, Vertrauen und langjährige Beziehungen.
Wir sind eXXcellent solutions.

Impressum:

eXXcellent solutions gmbh
Ulm: Beim Alten Fritz 2 89075 Ulm +49 731 550 26-0
Darmstadt: Europaplatz 4 64293 Darmstadt +49 731 550 26-0
Geschäftsführer: Dr. Martina Burgetsmeier, Gerhard Gruber, Wilhelm Zorn
Sitz und Registergericht: Ulm, HRB-Nr. 4309