Cloud Security from a Users-Perspective

Daniel Niecke

Zusammenfassung—In dieser Ausarbeitung soll untersucht werden, was beachtet werden muss, wenn eine Anwendung in der Cloud betrieben werden soll. Der Fokus soll dabei auf Kunden mit geringen Ressourcenanforderungen liegen, die Funktionen von großen Providern nutzen und nicht die Möglichkeit haben direkten Einfluss auf den Umfang der Funktionen zu nehmen. Viele Kunden aus diesem Segment haben nicht die Möglichkeiten komplexe IT-Infrastrukturen selbst zu betreiben.

Index Terms—Cloud Security, Cloud API, Container, Cluster, GCP, AWS, Azure

1 Einleitung

In den letzten Jahren scheint das Thema Cloud immer mehr an Bedeutung zu gewinnen, wenn man die jährlichen Umsätze in der Public Cloud [1] betrachtet. Diese sind von 42,8 Milliarden US-Dollar in 2010 auf 153,5 Milliarden US-Dollar in 2017 angestiegen, was einer Verdreifachung in nur sieben Jahren entspricht. Des Weiteren wurde ein Anstieg auf ca. 300 Milliarden US-Dollar in 2021 prognostiziert.

Unabhängig von diesem Wachstum nehmen die Sicherheitsbedenken der Unternehmen laut Bitkom Cloud Monitor 2018 immer weiter zu, während der allgemeine Widerstand gegen die Cloud langsam abnimmt [2]. Gerade durch die DSGVO sind allerdings viele neue juristische Fragen und Probleme aufgekommen, gerade was die Angebote ausländischer Cloud-Anbieter betrifft.

In der folgenden Arbeit soll daher untersucht werden, was aus der Sicht eines Kunden beachtet werden sollte, wenn eine Anwendung auf einer Cloud-Plattform betrieben werden soll. Im Fokus stehen Kunden, die lediglich Funktionen einer großen Cloud-Plattform nutzen und nicht die Möglichkeit haben direkt Einfluss auf den Funktionsumfang zu nehmen. Dabei werden Beispiele von den gängigen Cloud-Anbietern Amazon, Google und Microsoft verwendet. Natürlich könnte diese Liste noch beliebig erweitert werden, um beispielsweise die Alibaba Cloud, Angebote von SAP und Salesforce, sowie viele andere. Eine Betrachtung aller Cloud-Plattformen und Angebote ist allerdings durch den Umfang nicht möglich, weshalb nur Beispiele der ersten drei verwendet werden.

Um die Betrachtungen der verschiedenen Aspekte zu strukturieren, wird als Grundlage die Definition für Cloud Computing von der NIST verwendet [3]. Diese umfasst fünf essenzielle Charakteristika die ein Cloud-Dienst haben muss, um als solcher verstanden zu werden. Zudem unterteilt die Definition Cloud-Dienste in die drei Service Modelle *Software as a Service (Saas)*, *Platform as a Service (PaaS)* und *Infrastructure as a Service (Iaas)*. Im Bereich der Service Modelle gibt es sehr viele neue Entwicklungen, wie Security as a Service oder Software as a Service, die auch immer häufiger im Cloud Bereich verwendet werden. Neben den Charakteristika und den Service Modellen umfasst der Standard die vier Deployment Modelle *Private Cloud*, *Community Cloud*, *Public Cloud* und *Hybrid Cloud*.

Die Ausarbeitung ist wie folgt aufgebaut: Im nächsten Abschnitt wird zuerst auf verwandte Arbeiten, danach wird näher auf die fünf essenziellen Charakteristika des NIST Standard eingegangen. Im darauffolgenden Abschnitt werden einige Anwendungsszenarien und welche Sicherheitsaspekte bei den verschiedenen Szenarien wichtig sind beschrieben. In Abschnitt 5 wird auf einige weitere Problematiken eingegangen, die zu beachten sind, wenn beliebige Anwendungen auf einer Cloud-Plattform betrieben werden. Im Anschluss daran werden Lösungsansätze präsentiert, um den Problemen entgegenzuwirken.

2 Related Work

Chen et al. haben bereits im Jahr 2010 in einer Arbeit die Frage behandelt, was wirklich neu im Bereich Cloud Security im Vergleich zu traditionellen Systemen ist [4]. Die Autoren sind damals zu dem Ergebnis gekommen, dass die meisten Aspekte der Cloud Security bereits bei traditionellen Systemen relevant waren und sind. Als wirklich neu und grundlegend im Bereich Cloud Security bezeichnen die Autoren lediglich „the complexities of multi-party trust considerations, and the ensuing need for mutual auditability“. Die meisten damals diskutierten Aspekte konnten jedoch auch bei Systemen nachgewiesen werden, die nicht der Definition von Cloud-Diensten genügen. Da beide Problematiken von Kunden einer Cloud-Plattform meistens nur in geringem Maße beeinflussbar sind, werden diese Aspekte in der folgenden Arbeit jedoch nicht in den Fokus genommen.

Im Bereich Cloud wurden bereits diverse Bereiche diskutiert, so haben zum Beispiel der Springer Verlag und das IEEE jeweils Journals zum Thema Cloud gestartet. Security wurde im Zusammenhang mit der Cloud bereits aus vielen Perspektiven diskutiert und wird häufig als eines der relevantesten Themen im Bereich Cloud beschrieben [5], [6]. Appelrath et al. haben 2014 eine Ausarbeitung für die Deutsche Akademie der Technikwissenschaften verfasst, die sich mit Deutschland als Cloud Computing Standort auseinandersetzt. Die Autoren kamen zu dem Schluss, dass viele deutsche Unternehmen Cloud-Anwendungen skeptisch gegenüberstehen und das die meiste Skepsis durch Sicherheitsbedenken begründet wurde [7].

Wenn es um Security im Cloud-Umfeld geht, werden auch häufig Service Level Agreement (SLA) [8] diskutiert.

Dabei halten der Dienstleister/Anbieter und der Kunde verschiedene Rahmenbedingungen über die Qualität eines Produktes oder einer Dienstleistung fest. Bei Cloud-Diensten sollten die folgenden Rahmenbedingungen immer in einer SLA festgehalten werden [9]:

- Verfügbarkeit
- Performance
- Reaktionszeit
- Wiederherstellungszeit
- Wartungsfenster und Downtimes

3 Essentielle Charakteristika

Im folgenden Abschnitt werden die fünf essenziellen Charakteristika von Cloud-Diensten laut NIST betrachtet. Dabei wird auf abstrakte Problematiken eingegangen, die sich durch diese Charakteristika ergeben. Da es sich lediglich um Konzepte handelt und nicht um konkrete Implementierungen ist es leider nicht möglich konkrete Umsetzung zu entwickeln, um die Problematiken zu vermeiden.

3.1 On-Demand Self-Service

Viele Cloud-Anbieter werben damit, dass der Kunde nur für die Leistungen bezahlt, die er auch wirklich verwendet. Dies bedeutet allerdings auch, dass entweder der Kunde wissen muss, welche Leistungen er benötigt oder er muss eine Anwendung auf der Cloud-Plattform betreiben, die selbst die nötigen Ressourcen buchen kann, die sie benötigt. Bei den meisten Anwendungen ist diese Form der Skalierung nicht automatisch möglich. Daher muss der Kunde gegebenenfalls für Zusatzfunktionen bezahlen oder regelmäßig aktiv werden.

Ein weiteres Problem entsteht, wenn der Kunde sich nicht weit genug mit der Anwendung, die betrieben werden soll, auskennt. Eine Einschätzung der benötigten Ressourcen für zum Beispiel eine Virtuelle Maschine durch den Kunden ist in dem Fall praktisch nicht möglich.

3.2 Broad Network Access

Ein Vorteil der meisten Cloud-Plattformen ist die enorm leistungsstarke Netzanbindung, da die meisten Cloud-Plattformen in hochverfügbaren Rechenzentren betrieben werden, welche eine dementsprechende Netzanbindung besitzen. Hier ist der Vorteil gegenüber Server-Maschinen, die vom Kunden selbst betrieben werden, eindeutig. Die meisten Kunden haben nicht die Möglichkeit hochverfügbare Netzanbindungen aufzubauen. Somit ist es deutlich einfacher einen Web-Shop beispielsweise vielen Kunden auf der gesamten Welt bereitzustellen, wenn dieser in der Cloud betrieben wird.

Jedoch birgt diese Netzanbindung auch Risiken, wenn zum Beispiel Schadsoftware in die eigene Anwendung gebracht wird. In diesem Fall profitiert die Schadsoftware ebenfalls von der Netzanbindung und kann zum Beispiel für ein leistungsstarkes Botnetz verwendet werden. Zudem gibt es Anwendungen, die nicht zwingend über ein Netzwerk erreichbar sein müssen und bei denen eine Netzanbindung mehr Risiken birgt als sie Vorteile bringt. Hierunter fallen Anwendungen, die benötigt werden, um vor Ort installierte Hardware zu betreiben, wie Anlagensteuerungen oder Anwendungen, die für den Betrieb aus anderen Gründen unerlässlich sind und keinen hohen Ressourcenbedarf haben. Denn sobald die Verbindung des Kunden der Cloud-Plattform zum Internet ausfällt, kann dieser auf sämtliche in der Cloud betriebene Anwendungen nicht mehr zugreifen. Dies ist bei einem Web-Shop womöglich ein vertretbares Risiko, da die Kunden des Web-Shops weiter auf ihn zugreifen können und somit nicht vom Ausfall betroffen sind. Bei einer Produktionsanlage, die beim Netzwerkausfall ebenfalls ausfällt, sollte dieses Risiko allerdings nicht unterschätzt werden.

3.3 Resource Pooling

Grundlage aller Cloud-Plattformen ist die gemeinsame Nutzung von Rechenressourcen, unabhängig, ob in einem Unternehmen (Privat Cloud) oder bei einem externen Anbieter (Public Cloud). Dadurch, dass Cloud-Plattformen ihre Dienste vielen Kunden bereitstellen, sind die Infrastrukturen dementsprechend größer. Die hierbei entstehenden Skaleneffekte führen dazu, dass die Preise für Arbeitsspeicher und CPUs deutlich günstiger sind, als bei zum Beispiel kleinen Inhouse-Lösungen. Geht man davon aus, dass der Anbieter die perfekte Kapselung der einzelnen Kunden erreichen kann, was wahrscheinlich nicht erreichbar ist, so teilen sich die Kunden immer noch gemeinsame Server-Maschinen. Das kann im Extremfall bedeuten, dass Hardware von Ermittlungsbehörden beschlagnahmt wird, weil der Verdacht besteht, dass einer der Kunden eben diese Hardware für illegale Handlungen verwendet hat [10]. Dies bedeutet, dass im schlimmsten Fall alle Kunden betroffen sein können, wenn bereits einer eine Straftat begeht.

3.4 Rapid Elasticity

Bei Cloud Anwendungen wird erweiternd zu dem On-Demand Self-Service ebenfalls immer erwartet, dass die Ressourcen, die der Kunde ordern kann, praktisch beliebig skalierbar sind. Diese Skalierbarkeit hat natürlich ebenfalls gewisse Grenzen, welche allerdings im Normalfall hoch genug angesetzt werden, sodass ein durchschnittlicher Kunde sie nicht erreichen kann.

Viel eher ist es hierbei im Interesse des Kunden selbst, dass die Skalierbarkeit beschränkt ist, damit eine falsch konfigurierte Anwendung nicht zum explosionsartigen Kostenanstieg führt. Die Google Cloud Plattform bietet hier Möglichkeiten den Zugriff auf Cloud-Ressourcen durch verschiedene Projekte abzusichern und für die jeweiligen Projekte sogenannte Quotas einzustellen [11]. VMs in der AWS werden zum Beispiel dadurch beschränkt, dass beim Konfigurieren der VM eine Klasse gewählt wird, welche feste Ressourcen Maximalwerte hat. Dadurch ist sichergestellt, dass zumindest diese eine VM nicht beliebig skaliert werden kann. Mit diesen Beschränkungen wird vermieden, dass eine Anwendung, durch fehlerhafte Konfiguration oder eingeschleuste Schadsoftware, zu viel Ressourcen verbraucht. Allerdings können diese Kontingente vom Kunden selbst wieder verändert werden. Eine wirksame Absicherung gegen Innentäter wird somit nicht erreicht.

3.5 Measured Service

Für die Abrechnung einer Cloud-Ressource wird im Normalfall gemessen, wie stark die Ressource beansprucht wurde. Aus diesen Werten wird am Ende die Abrechnung erstellt, damit der Kunde nur das bezahlt, was er wirklich verwendet hat. In der Theorie ist dieses Konzept gerade für Kunden mit ungewissem Ressourcenbedarf und starken Schwankungen sehr interessant. Anstatt viele Server-Maschinen zu betreiben oder anzumieten, die sich einen Großteil der Zeit im Leerlauf befinden, laufen praktisch alle Anwendungen immer mit maximaler Ressourcenauslastung, da sich das Maximum frei bewegen kann. Zum einen lassen sich aus den Ressourcennutzungen bereits Muster ableiten, die allerdings nur dem Cloud-Anbieter zur Verfügung stehen und für andere Kunden des Anbieters praktisch nicht einsehbar sein sollten. Die Abrechnungen der Ressourcen bei vielen Providern sind häufig jedoch nicht in dem Maße transparent, wie teilweise propagiert wird. Mit dieser Abrechnungsproblematik haben sich bereits Arbeiten befasst, allerdings schlagen diese meistens Lösungen vor, die aufseiten des Cloud-Providers implementiert werden können und nicht aufseiten des Kunden [12], [13]. Ein gewisser Spielraum sollte daher immer für das Budget von Cloud-Anwendungen eingeräumt werden.

Microsoft bietet den Kunden in Kooperation mit Cloudyn zusätzlich einen erweiterten Monitoring Service. Diesen sogenannten Cost Monitor können Kunden verwenden, um ihre Ausgaben für Cloud-Ressourcen immer im Überblick zu haben [14]. Das Interessante an dieser Umsetzung ist, dass sie ebenfalls für die AWS und die Google Cloud Platform verwendet werden kann.

4 Anwendungsszenarien

Auf Grundlage der Charakteristika, die im vorherigen Abschnitt beschrieben wurden, sind viele neue Anwendungen entstanden und einige erleben eine Renaissance. Anders als häufig dargestellt wird, sind diverse Cloud-Anwendungen vom Konzept her keine neuen Erfindungen, sondern haben in ähnlicher Form bereits existiert. Allerdings haben sich bei vielen Anwendungen relevante Veränderungen durch zum Beispiel die Erreichbarkeit über das Internet ergeben. Die Nutzer sind nicht mehr an ein Mainframe gebunden, da die Cloud im übertragenen Sinne das neue „Mainframe“ ist und diese über das Internet frei zugänglich ist. Eine Vereinheitlichung, wie man sie vom Stromnetz oder dem Telekommunikationsnetz kennt, ist allerdings nach aktuellem Stand nicht vorhanden.

Im Folgenden werden vier Anwendungsszenarien beschrieben, wie sie auf Cloud-Plattformen häufig zu finden sind und welche teilweise ganz eigene Sicherheitsaspekte haben.

4.1 Storage

Eine der breiten Masse an Internetnutzern bekannte Anwendung ist der Online-Speicher [15]. Also ein Speicher, der sich nicht mehr beim Nutzer vor Ort befindet, sondern in einem Rechenzentrum an einer zentralen Stelle. Diese Form des Datenspeichers ist sowohl im Geschäftsbereich, als auch im privaten Sektor sehr beliebt. Wobei der größte Anstieg in der jüngeren Vergangenheit im privaten Sektor wahrscheinlich durch Unternehmen wie DropBox entstanden ist [16], welche es gerade Laien enorm leicht gemacht haben, Daten online zu speichern. Am Anfang schienen die Sicherheitsbedenken hier noch relativ gering und nur technisch visierte Menschen haben sich damit überhaupt befasst. Dies scheint sich gerade in Deutschland und auch in Europa nicht zuletzt durch die DSGVO stark zu ändern.

Als Kunde eines Datenspeichers einer Cloud-Plattform gibt es häufig nicht viele Möglichkeiten die Sicherheit zu beeinflussen. Die Verfügbarkeit und Geschwindigkeit, sowie der Preis und die Kapazität sind meistens maßgebend für die Wahl des Anbieters. Für größere Mengen sollte hierbei immer noch geprüft werden, wie einfach sich die Daten exportieren lassen und in welche Formate. Handelt es sich um einen einfachen Festplattenspeicher ist dies meistens sehr einfach möglich, werden die Daten allerdings in speziellen Datenbanken abgelegt kann dies zum Problem führen, sollte einmal die Cloud-Plattform gewechselt werden. Viele Cloud-Plattformen bieten an, die Daten auf den Festplatten der Cloud-Plattform zu verschlüsseln, sodass die Daten nicht einfach von den Festplatten ausgelesen werden können. Ob dieser Schutz allerdings wirksam ist, wird häufig angezweifelt [17]. Einen echten Schutz erreicht man so gesehen nur, wenn man die Daten verschlüsselt und erst dann hochlädt. Dadurch lässt sich der Speicher der Cloud-Plattform allerdings nur noch als Archiv nutzen und nicht um auf den Daten zu rechnen ohne sie herunterladen zu müssen oder auf der Cloud-Plattform zu entschlüsseln.

4.2 Cloud APIs

Die großen Cloud-Plattformen wie die GCP, AWS und Microsofts Azure bieten alle ein breites Spektrum an APIs an. Damit sind Funktionen wie Text-To-Speech, Geocoding, Bilderkennung und einige weitere direkt per API abrufbar. Die Verwendung solcher APIs ist meistens deutlich günstiger, als die Funktionen selbst in zum Beispiel Apps für Mobilgeräte oder IoT Devices zu integrieren. Der größte Rechenaufwand wird damit zusätzlich auf die Cloud-Plattform ausgelagert. Bei der Verwendung solcher APIs tritt meistens allerdings sehr schnell ein Vendor Lock-In auf, welcher in Abschnitt 5.2 weiter beschrieben wird. Des Weiteren ist ein Internet-Zugang erforderlich, um die Anwendung in vollem Umfang nutzen zu können. Im schlimmsten Fall kann durch das Fehlen eines stabilen Netzwerkzugangs die komplette Anwendung zeitweise unbrauchbar werden.

4.3 Container und Virtuelle Maschinen

Diverse Container-Formate und auch Virtuelle Maschinen (VM) lassen sich standardmäßig auf Cloud-Plattformen betreiben. Der Betrieb von Containern erfüllt dabei meistens mehr Eigenschaften der NIST Definition als eine Virtuelle Maschine. Bei Containern gibt es keinen fest zugewiesenen Arbeitsspeicher und auch keine feste Anzahl von vCPUs, daher lassen sich diese Anwendungen flexibler skalieren und es ist möglich zu messen, welche Ressourcen wirklich durch den Container verwendet wurden. Bei klassischen VMs ist dies nicht möglich, da ein vollständiges Betriebssystem installiert wird, welches einen festen Arbeitsspeicher benötigt und eine feste Anzahl von vCPUs. Die AWS zum

Beispiel versucht auch hier nur abzurechnen, was die VM wirklich verbraucht hat. Damit bezahlt der Kunde weniger, als seine VM im Monat maximal kosten würde. Erreicht die VM allerdings das Limit der gewählten VM Klasse, muss die gesamte VM auf eine neue VM migriert werden, um weitere Ressourcen verwenden zu können. Da Container als Prozesse im Host-Betriebssystem laufen unterliegen sie nicht dieser Restriktion. Eine allgemeine Einstufung, ob VMs oder Container sicherer sind, ist praktisch nicht möglich, da die Sicherheit stark von der Art der Verwendung der beiden Ansätze abhängt. Der klare Vorteil einer VM liegt darin, dass sie einen eigenen Kernel verwendet, dies scheint in den vergangenen Jahren allerdings nicht mehr zwingend ein Sicherheitsvorteil zu sein. Hiermit hat der Kunde lediglich die Möglichkeit einen Kernel zu wählen, der besser geeignet ist für seine Anwendung. Dies ist wichtig, wenn eine Anwendung betrieben werden soll, die nur mit einem sehr speziellen Betriebssystem kompatibel ist. Betrachtet man die Ressourcennutzung, Neustartzeiten und den Migrationsaufwand stehen Container jedoch besser da als VMs.

4.4 Application Hosting

Das letzte hier betrachtete Anwendungsszenario ist das Application Hosting. Hierbei wird eine komplette Anwendung auf der Cloud-Plattform betrieben. Der Kunde muss sich also ähnlich, wie bei den Cloud APIs, keine Gedanken darüber machen auf welchem Betriebssystem oder auf welcher Hardware die Applikation läuft und welche Bibliotheken benötigt werden. Anders als bei den APIs, die meist nur eine Schnittstelle für andere Programme bieten, handelt es sich bei den gehosteten Applikationen häufig um Abwandlungen bereits existierender eigenständiger Anwendungen. Diese werden meistens über einen Webbrowser direkt vom Kunden aufgerufen und nicht von einer anderen Anwendung. Prominente Beispiele in diesem Bereich sind Office 365[1] und die Lösung von Salesforce[2].

Bei solchen Anwendungen hat ein Kunde leider meistens praktisch keinen Einfluss auf die Sicherheit der Anwendung, daher ist es noch wichtiger, dass der Betreiber der Anwendung vorher genau geprüft wird. Hierbei ist es sinnvoll zu prüfen, ob der Anbieter die DSGVO einhält, welche weiteren Zertifizierungen der Anbieter besitzt und welche Reaktionszeiten und Verfügbarkeiten in den SLAs definiert sind. Ob Backups von den Daten angelegt werden oder vom Kunden angelegt werden können, ist zudem sehr interessant.

5 Allgemeine Sicherheitsprobleme bei Cloud-Anwendungen

Im folgenden Abschnitt wird auf einige allgemeine Sicherheitsprobleme eingegangen, die beachtet werden sollten, bevor eine Anwendung für eine Cloud-Plattform entwickelt wird oder auf einer solchen Plattform betrieben werden soll. Da konkrete für die Implementierungen direkt relevante Aspekte plattformabhängig sind und eine vollständige Liste nicht erstellt werden kann, wird an dieser Stelle lediglich auf abstraktere Aspekte eingegangen. Diese dürften bei Verwendung der meisten Cloud-Plattformen relevant sein, wobei es seien kann, dass einige Plattformen bereits zusätzliche proprietäre Lösungen anbieten.

5.1 Netzwerkausfälle

Vor der Entwicklung oder dem Einkauf einer neuen Anwendung, die auf einer Cloud-Plattform betrieben werden soll oder zumindest Schnittstellen einer Cloud-Plattform im Betrieb benötigt, sollte immer geklärt werden, welche Probleme bei einem Netzwerkausfall entstehen. Im Normalfall dürfte es für einen Kunden sehr aufwendig sein ein System mit ähnlicher Verfügbarkeit aufzubauen, wie es von großen Cloud-Anbietern geboten wird. Hiervon sind solche Unternehmen ausgenommen, die eigene Rechenzentren betreiben. Daher dürfte es für viele Kunden eher von Relevanz sein, ob Externe als wiederum Kunden des Cloud-Kunden auf die Anwendung zugreifen oder Mitarbeiter des Cloud-Kunden selbst. Wird auf der Cloud-Plattform zum Beispiel ein Web-Shop betrieben, dann greifen zwar Externe und Interne auf das System zu, aber alleine im Interesse der Verfügbarkeit für die Kunden des Web-Shops ist die Cloud bereits eine interessante Lösung. Anders sieht es bei lediglich intern genutzten Anwendungen aus, wie zum Beispiel Steuerungen für Maschinen. Zwar existieren bereits Ansätze diese ebenfalls auf Cloud-Plattformen zu migrieren, allerdings bleibt hier immer die Frage, ob bei diesen Anwendungen so viele Berechnungen anfallen, dass sich der Aufwand einer Migration auf die Cloud-Plattform überhaupt lohnt.

5.2 Vendor Lock-In

Gerne wird das Nutzen von Cloud-Ressourcen mit dem von Elektrizität aus der Steckdose verglichen. Beim Entwickeln von Anwendungen zeigt sich allerdings oftmals, dass dieser Vergleich leider an vielen Stellen hinkt. Die Verwendung von Cloud-Ressourcen ist nach wie vor deutlich komplexer. Es bietet zwar fast jeder Provider eine SDK an, die Entwickler dabei unterstützen soll Schnittstellen der Cloud zu nutzen, aber es gibt keinen echten Standard für diese Schnittstellen oder die darauf aufbauenden SDKs. Sollen zum Beispiel Schnittstellen der GCP verwendet werden, muss die Google Cloud SDK[3] in die Anwendung integriert werden. Wenn nun irgendwann statt der GCP die AWS verwendet werden soll, muss die Google Cloud SDK entfernt werden und durch die AWS SDK[4] ersetzt werden. Da die Schnittstellen allerdings nicht einheitlich sind, ist das Risiko, dass erhebliche Änderungen am Code vorgenommen werden müssen, immer gegeben. Hierbei muss es zusätzlich noch eine SDK der jeweiligen Cloud-Plattform für die verwendete Programmiersprache geben.

Selbst ohne Wechsel der Cloud-Plattform ist man allerdings nicht komplett sicher vor Veränderungen. Wird zum Beispiel die Schnittstelle verändert bzw. durch eine neue Version seitens des Providers ersetzt, kann dies zu erheblichen Anpassungen führen. Im schlimmsten Fall müssen

1. Office 365 für Unternehmen: https://products.office.com/de-de/business/explore-office-365-for-business?rtc=1
2. Salesforce Produktportfolio: https://www.salesforce.com/de/products/
3. GCP SDK https://cloud.google.com/sdk/
4. AWS SDK https://aws.amazon.com/de/tools/

Abbildung 1: Soft- und Hardwarestack einer konventionellen Anwendung.

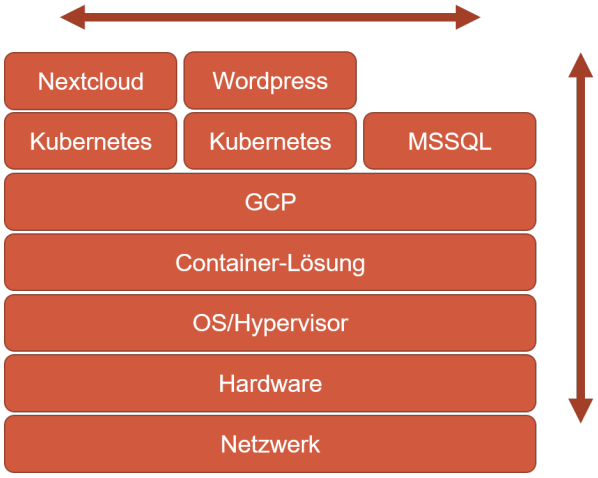

Abbildung 2: Soft- und Hardwarestack einer Cloud-Anwendung.

Teile vollständig ersetzt werden. In diesen Fällen spricht man vom Vendor Lock-In [18]. Dies ist zwar im engeren Sinne kein Sicherheitsproblem, kann allerdings bereits die Verfügbarkeit einer Anwendung stark beeinflussen und hat somit Auswirkungen auf die Sicherheit im weiteren Sinne. Zudem wird relativ viel Code in eine Anwendung integriert, der für die Kommunikation über das Internet gedacht ist. Auch hierdurch können bereits Sicherheitslücken in die Anwendung gelangen.

5.3 Soft- und Hardware Stack

Der Aufbau einer Cloud-Plattform kann relativ Komplex werden, wobei der Kunde lediglich auf ein paar wenige Ebenen Einfluss nehmen kann. In Abbildung 1 wir an einem Beispiel gezeigt, welche Schichten zu verwalten sind, beim Betrieb einer Nextcloud[5] auf einer eigenen Server-Maschine. Nextcloud ist eine Anwendung zum hosten einer privaten Storage Cloud, die um viele Funktionen erweitert werden kann und auch bei öffentlichen Einrichtungen in Deutschland immer beliebter wird [19]. Hierbei handelt es sich um die einfachste Art des Deployments von Nextcloud, da es nur eine Instanz auf einem Server gibt. Eventuelle Datenbanken und Caching-Systeme wurden hierbei außer Acht gelassen. Allerdings musste bei diesem Deployment bereits entschieden werden, welches Betriebssystem (OS) verwendet werden soll und auf welcher Hardware das Betriebssystem laufen soll. Zudem wurde eine Netzwerkschicht benötigt.

Vergleicht man diese Struktur jetzt mit der einer Nextcloud, welche in der GCP gehostet wird, wie sie in Abbildung 2 dargestellt ist, fallen zusätzliche Schichten auf. Für das einfache Deployment der Nextcloud in der GCP wurde Kubernetes verwendet, welches einem ermöglicht mit relativ geringem Aufwand den von der Nextcloud GmbH selbst bereitgestellten Docker Container direkt in der GCP zu betreiben. Auf das ganze System wird über die GCP zugegriffen, welche Virtuelle Maschinen bereitstellt, auf denen der Kubernetes Cluster läuft. Doch diese Ebenen liegen größtenteils außerhalb des Einflussbereichs des Kunden. Er kann lediglich entscheiden, wie viel Ressourcen den VMs zugewiesen wird und noch welche Prozessorarchitektur verwendet werden soll. Unterhalb der GCP hat er praktisch kaum noch Möglichkeiten der Anpassung. Die hier parallel zur Nextcloud Instanz dargestellten Applikationen (Wordpress und Microsoft SQL Server) sollen verdeutlichen, dass parallel zur eigenen Instanz weitere Anwendungen betrieben werden, welche auf der gleichen Server-Maschine liegen können. Das heißt, die Komplexität beim Betrieb steigt in zwei Dimensionen. Zum einen, weil das entstehende System mehr Schichten besitzt und zum anderen, weil es mehr parallele Nutzer in dem System gibt. In einer kurzen Ausführung „Complexity Is the Enemy“ erklärte D. E. Geer Jr. warum Komplexität immer ein Sicherheitsrisiko in sich birgt [20]. Bei immer komplexer werdenden Strukturen sind unerwartete Ereignisse immer wahrscheinlicher und widersprechen an dieser Stelle der immer wieder propagierten Simplizität einer Cloud-Lösung. Wird dies nicht berücksichtigt, läuft man Gefahr Fehler gar nicht erst zu erwarten.

5.4 Nutzer- und Rechteverwaltung

Bei den großen Cloud-Plattformen wird grundsätzlich ein Rechte und Nutzermanagement vorgesehen. Dieses soll helfen die Rechte, die eine Applikation, die in der Cloud betrieben wird oder die einer Applikation, die zum Beispiel Schnittstellen einer Cloud nutzt, zu kontrollieren. Grundsätzlich wird hierbei versucht die Rechte möglichst gering zu halten und den Applikationen eigene Zugänge mit eigenen Authentifizierungsinformationen zuzuordnen. Allerdings ist es sehr verlockend gerade bei Applikationen, die neu entwickelt werden, weitgehende Rechte einzurichten, damit man bei der Entwicklung seltener Anpassungen vornehmen muss. Diesem Ansatz muss daher von Kundenseite entgegengewirkt werden, indem zum Beispiel Entwickler keine Rechte vergeben können.

Für diese Verwaltungsaufgaben stellt Microsofts Azure zum Beispiel Active Directorys zur Verfügung, während Amazon und Google eigene Lösungen direkt für ihre jeweiligen Plattformen entwickelt haben. Allerdings ist die

5. Nextcloud https://nextcloud.com/

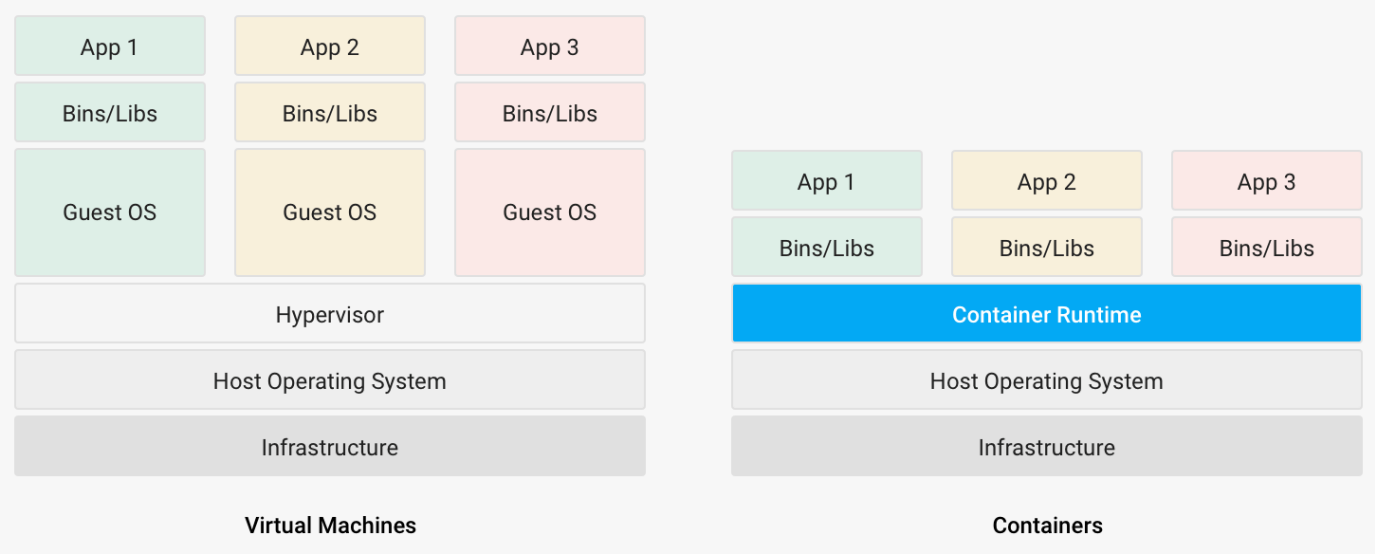

Abbildung 3: Virtuelle Maschinen im Vergleich zu Containern[6]

korrekte Konfiguration von Rechten und der damit verbundenen Limits oder Kontingente keine simple Aufgabe und benötigen relativ viel Zeit um wirksam vor Bedrohungen zu schützen.

5.5 Hardware-Ausfälle

Anders als von vielen Kunden angenommen handelt es sich bei Cloud-Lösungen nicht zwingend um redundante Systeme. Bei praktisch allen Anbietern sind zwar die Stromversorgung und das Netzwerk redundant, allerdings ist bereits das Netzwerk meistens nicht vollständig redundant, sondern nur bis zum Server-Rack. Fällt also der Switch des Server-Racks aus, in dem die Applikation eines Kunden betrieben wird, ist diese nicht mehr erreichbar. Gleiches gilt für die Server-Maschinen auf denen die Cloud-Plattform gehostet wird. Fällt eine dieser Maschinen aus, ob durch einen Defekt oder weil sie durch eine Ermittlungsbehörde beschlagnahmt wird, sind die Applikationen, die auf genau dieser Maschine ausgeführt wurden, nicht mehr verfügbar. Dies zeigt sich auch zum Beispiel darin, dass die GCP einen vor der höheren Ausfallwahrscheinlichkeit einer SSD warnt, wenn man diese als Speicher verwenden möchte.

Auch größere Hardwareausfälle sind bei Cloud-Plattformen nicht völlig ausgeschlossen. Vor allem wetterbedingte Ausfälle treten selbst bei der GCP oder bei Microsofts Azure durch zum Beispiel Hurrikans oder Gewitter auf [21], [22]. Daher werden Kunden der GCP zum Beispiel über die GCP Console vor Hurrikans gewarnt, wenn sie Anwendungen in womöglich betroffenen Rechenzentren betreiben.

6 Ansätze zur Erhöhung der Sicherheit

Im Allgemeinen gibt es viele Ansätze Sicherheit in EDV-Anwendungen zu erreichen und praktisch alle lassen sich auch auf Cloud-Plattformen übertragen. Die Verwendung sicherer Passwörter und das regelmäßige Einspielen von Software-Updates sind wahrscheinlich die Punkte, denen jeder zustimmen wird. Bei Antiviren-Software herrscht bereits weniger Konsens [23]. Bei den ersten beiden Punkten werden Kunden in der Cloud bereits aktiv unterstützt, da die Applikationen einer Cloud-Plattform meistens vom Anbieter aktualisiert werden und zu schwache Passwörter gar nicht erst verwendet werden können. Zudem setzen alle großen Cloud-Anbieter SSL/TLS-Verschlüsselungen für Datenübertragungen und Private/Public-Key Authentifizierungen voraus. Allerdings gilt dies nur für den Zugang des Kunden zur Cloud-Plattform. Wenn der Kunde einen Web-Shop oder eine selbst entwickelte Anwendung auf der Cloud-Plattform betreibt, wird er dabei häufig nicht mehr aktiv vom Anbieter unterstützt. Das Geschäftsmodell der großen Anbieter lässt eine direkte Unterstützung jedes Kunden nicht zu. Also gilt es hier zumindest die gleichen Regeln zu beachten, wie bei einer hausinternen EDV.

Im folgenden Abschnitt wird auf einige Punkte eingegangen, die zu beachten sind, wenn eine Cloud-Lösung verwendet wird.

6.1 Übersetzung der eigenen Rechte und Rollen in die Cloud-Umgebung

Striktes Rechte- und Rollenmanagement wird häufig von Nutzern eher als störend und hinderlich gesehen, da eventuell Freigaben für spezielle Aufgaben benötigt werden oder Prozesse unnötig komplex erscheinen. Für Cloud-Anwendungen sind strikte Rechte und Rollen jedoch noch wichtiger, da viele Anwendungen in einem größeren Umfang selbst regieren. Da Cloud-Plattformen einen Zugang anbieten, mit dem der Kunde auf jegliche Ressourcen zugreifen kann, ist es sinnvoll eben diesen Zugang auch vernünftig abzusichern. Ein unsicheres Passwort kann bei schlecht konfigurierten Rechten und Rollen zur kompletten Übernahme des Kundenkontos führen. Dies hätte verheerende Folgen für den Kunden. Doch selbst, wenn der Zugang zur Cloud-Plattform vom Kunden mit nötig komplexen Passwörtern gesichert wurde, schützt dies nicht vor Innentätern. Um ebenfalls einen akzeptablen Schutz gegen

6. https://cloud.google.com/containers/

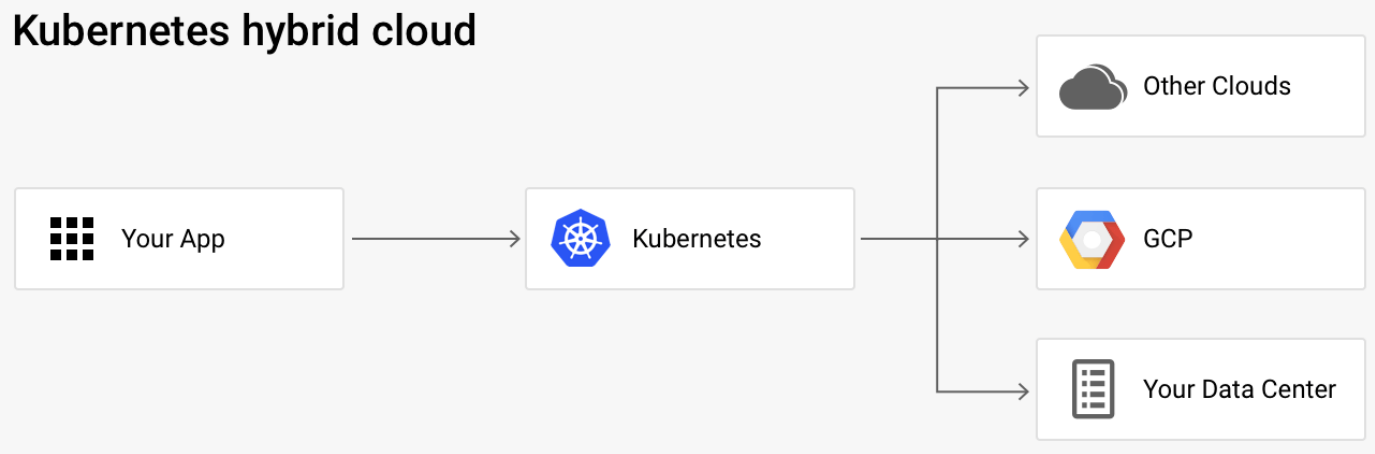

Abbildung 4: Anwendungen in Private und Public Cloud mit Kubernetes[7]

Innentäter zu erreichen, wird ein restriktives Rollenmodell benötigt.

6.2 Strikte Kontingente festlegen

Wenn es um die Skalierbarkeit von Anwendungen geht, ist es dringend nötig dieser Skalierbarkeit feste Grenzen zu setzen. Selbst wenn dies schon durch den Cloud-Anbieter geschieht, ist hier meistens noch Handlungsbedarf. Zum Beispiel bietet die AWS bei einem Standard Nutzerkonto die Möglichkeit fünf Instanzen diverser EC2 Konfigurationen zu starten. Da es sich bei EC2 Instanzen um VMs handelt können diese teilweise beachtliche Arbeitsspeicher und CPU Ressourcen verwenden. Für einige Projekte sind fünf Instanzen deutlich zu wenig und dieser Wert muss nach oben korrigiert werden, allerdings können fünf Instanzen für eine kleine Anwendung bereits deutlich zu viel sein. Dann wäre es sinnvoll diesen Wert weiter zu reduzieren, damit gar nicht erst die Möglichkeit besteht, mehr Ressourcen zu buchen als nötig sind.

Um eine vernünftige Grundlage für Anwendungen auf einer Cloud-Plattform zu schaffen, sollte sich der Kunde immer mit dem Einrichten von Kontingenten und der Konfiguration von Zugängen vertraut machen. Dies ist womöglich für kleine Unternehmen noch wichtiger als bei großen, da eine falsch konfigurierte Anwendung durch automatische Skalierung enorme Verluste bedeuten kann.

6.3 Container und Cluster

Bereits 2014 erklärte Joe Beda in einer Präsentation, dass die Google Cloud Plattform praktisch vollständig auf Container setzt [24]. Da diese leichtgewichtiger als herkömmliche VMs sind und weniger Ressourcen verbrauchen, lassen sie sich deutlich leichter in großem Maßstab verwenden. Doch auch für kleinere Anwendungen sind Container eine interessante Lösung. Zum einen werden viele Open-Source-Projekte mindestens in einem Container-Format ausgeliefert und sind als solche meistens deutlich einfacher einzurichten als die herkömmlichen Formate. Zum anderen bietet jede große Cloud-Plattform native Container-Unterstützung für die meisten Formate. Der interessante Unterschied zwischen VMs und Containern liegt darin, dass kein Gast-Betriebssystem existiert. Container laufen als einfache Prozesse des Host-Betriebssystems wie in Abbildung 3 dargestellt. Durch zum Beispiel die Verwendung von Namensräumen auf Kernelebene werden Container strikt voneinander getrennt. Ein Container kann somit grundsätzlich andere Container nicht sehen und auch nicht auf andere zugreifen. Damit ist die Möglichkeit geboten, jede Anwendung in einen separaten Container zu schieben, ohne deutlich mehr Ressourcen zu verbrauchen. Sollte in diesem Fall eine Anwendung angegriffen oder mit Schadsoftware infiziert werden, kann dies nicht ohne Weiteres auf eine Anwendung in einem anderen Container überspringen. Somit ist es möglich einen Web-Shop, der als Container ausgeliefert wird, direkt auf einer Cloud-Plattform zu betreiben und zum Beispiel die Datenbank des Shops in einem Weiteren, ohne vorher VMs mit eigenen Betriebssystemen aufzusetzen und die nötigen Bibliotheken und Zusatzprogramme zu installieren. All dies steckt bereits im Container oder wird von der Cloud-Plattform bereitgestellt.

Im nächsten Schritt lassen sich Container zudem in einem Cluster betreiben, der zum Beispiel über Kubernetes administriert wird. Auch dies wird von vielen Cloud-Plattformen nativ unterstützt. Da über Kubernetes mehrere Instanzen der gleichen Anwendung betrieben werden können und Kubernetes selbst die Verteilung der Container, auf den darunter liegenden Maschinen vornimmt, erreicht man hiermit bereits eine bessere Ausfallabsicherung und kann bereits mit wenig Aufwand die Lasten besser verteilen.

6.4 Private Cloud oder Private Knoten

Cloud-Plattformen wie die GCP, Microsofts Azure oder die AWS bieten grundlegend immer Ressourcen in einer Public Cloud an, also Ressourcen auf Server-Maschinen, die sich ein Kunde mit womöglich vielen anderen teilt. Da Public Clouds häufig gerade wegen diesem Teilen von Server-Maschinen in der Kritik sind, gibt es meistens auch die Möglichkeit separate Maschinen anzumieten. Hierbei handelt es sich allerdings nur bedingt um eine Private Cloud Lösung wie das NIST sie definiert, da sich die Server-Maschine immer noch in der Public Cloud befindet,

7. https://cloud.google.com/containers/

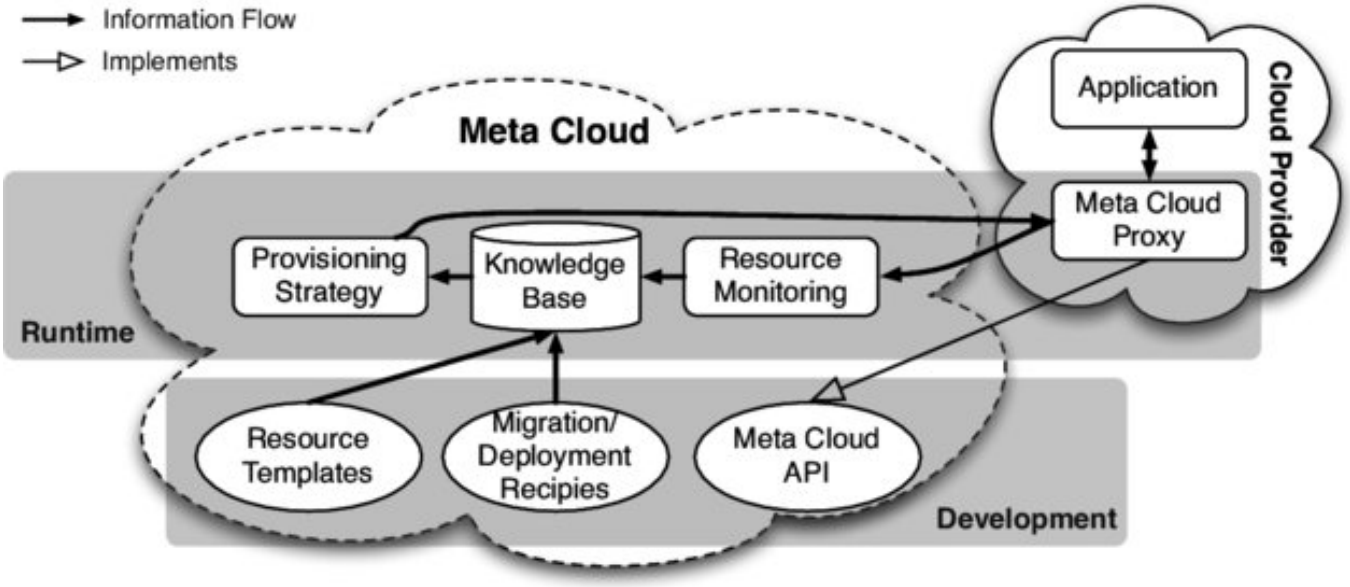

Abbildung 5: Abstrakte Darstellung einer Meta Cloud nach Satzger et al. [25]

allerdings durch die Cloud-Plattform sichergestellt wird, dass von keinem anderen Kunden Anwendungen auf dieser Maschine ausgeführt werden. Ob hierdurch wirklich eine höhere Sicherheit oder Verfügbarkeit erreicht wird, lässt sich durch äußere Betrachtung leider schwer sagen.

Einen interessanteren Ansatz bieten Orchestrierungs-Tools wie Kubernetes. Damit ist es dem Kunden nicht nur möglich wie im vorherigen Abschnitt beschrieben, eine Anwendung zu Skalieren, sondern auch auf verschiedenen Cloud-Plattformen zu betreiben. Abbildung 4 stellt schematisch das Konzept von Kubernetes dar. Da Kubernetes nur Container verwalten kann, wird hierfür natürlich vorausgesetzt, dass die Anwendung des Kunden (Your App) als Container ausgeliefert wird. Diese kann aus vielen verschiedenen oder einem großen Dienst bestehen, welcher durch Kubernetes automatisch zum Beispiel in der Azure und parallel in der GCP betrieben wird. Wenn der Kunde zusätzlich über eigene Server verfügt, auf denen er Kubernetes Nodes einrichtet, können ebenfalls Teile der Anwendung auf seinen eigenen Servern betrieben werden. Der Betrieb eines Kubernetes Clusters ist zusätzlich komplett ohne Public Cloud Ressourcen möglich. Hiermit erreicht man zumindest für Anwendungen, die als Container ausgeliefert werden eine sehr gute Verfügbarkeit. Durch die Verwendung eines standardisierten Formates zur Auslieferung der Anwendung, nämlich dem Container, kann man zusätzlich zumindest in gewissem Maße ein Vendor Lock-In vermeiden.

6.5 Meta Cloud

Einen Ansatz zur Meta Cloud beschreiben Satzger et al. in *„Winds of Change: From Vendor Lock-In to the Meta Cloud“* [25]. Bei der Meta Cloud geht es darum, Middleware zwischen die eigentliche Anwendung des Kunden und die Cloud-Plattform zu legen. Diese Middleware müsste jedoch von einer großen Community in einer transparenten Art und Weise entwickelt werden, um ein Meta Cloud Lock-In zu vermeiden. Die Autoren stellen in ihrer Arbeit selbst fest, dass zwar alle Komponenten existieren, um eine Meta Cloud zu entwickeln, es jedoch an der Integration mangelt. Daher ist gerade für kleine Unternehmen, eine Meta Cloud aktuell praktisch nicht umsetzbar. In Abbildung 5 ist die Meta Cloud nach Satzger et al. dargestellt. Wichtig ist hierbei die Unterscheidung zwischen Runtime und Development. Während der Entwicklung einer Anwendung für die Meta Cloud würde, anstatt der API einer konkreten Cloud-Plattform, die Meta Cloud API verwendet werden. Die für den Betrieb verwendete Cloud-Plattform muss noch zusätzlich die API der Meta Cloud unterstützen, damit ein reibungsloser Betrieb möglich ist.

Jeon et al. haben einen Ansatz zu einer Meta Cloud präsentiert, welcher lediglich die Vereinheitlichung des Datenspeichers vorsieht. Hierbei gehen sie weniger auf den Vendor Lock-In, sondern auf eine erhöhte Verfügbarkeit ein. Diese soll durch redundante Sicherung auf verschiedenen Cloud-Plattformen erreicht werden. Bei vielen Daten dürfte es allerdings relativ schnell nicht mehr wirtschaftlich sein, die Redundanz weiter zu erhöhen für einen marginalen Zugewinn an Verfügbarkeit.

6.6 Security Outsourcing

Wenn Anwendungen auf einer Cloud-Plattform betrieben werden sollen und damit die physikalische Infrastruktur durch Outsourcing in den Verantwortungsbereich eines größeren Providers geht, lohnt es sich womöglich die Sicherheit durch einen externen Dienstleister überwachen zu lassen. Hierbei gibt es viele Möglichkeiten, von Managed Firewalls, Intrusion Detection, DDoS-Schutz und Penetrationstests. In der Vergangenheit ist ein großer Markt im Bereich solcher Managed Security oder Security as a Service Lösungen entstanden, wodurch praktisch alle Bereiche abgedeckt werden [26]. Da hier aufseiten des Anbie-

ters die gleichen Skaleneffekte entstehen, wie bei Cloud-Plattformen, können die Anbieter ihre Dienste meistens deutlich günstiger anbieten, als dies bei kleinen Inhouse-Lösungen möglich wäre. Damit kann sich der Betreiber eines Web-Shops auf einer Cloud-Plattform zum Beispiel noch mehr auf sein Kerngeschäft konzentrieren. Ob die Dienste allerdings auch Sicherheitsaspekte bei der eigentlichen Entwicklung von Anwendungen für die Cloud unterstützen ist im Detail zu klären.

7 Fazit

Der Betrieb von Anwendungen jeglicher Art auf einer Cloud-Plattform ist bei weitem nicht so einfach, wie die Cloud-Anbieter es darstellen. Deshalb hinkt der Vergleich mit dem Stromnetz oder dem Telekommunikationsnetz nach wie vor. Dies bedeutet ebenfalls, dass das Thema Sicherheit in der Cloud nicht unterschätzt werden darf. Da das Erstellen oder Betreiben einer Anwendung auf einer Cloud-Plattform allerdings meistens deutlich schneller geht als bei Inhouse-Lösungen, ist die Gefahr groß, dass Sicherheitsprobleme übersehen werden oder ihnen nicht die nötige Relevanz beigemessen wird.

Vivek Kundra, ehemaliger CIO der Vereinigten Staaten, nannte eine interessante Sichtweise auf die Problematik: *„Cloud computing is often far more secure than traditional computing, because companies like Google and Amazon can attract and retain cyber-security personnel of a higher quality than many governmental agencies. “* Dies dürfte auch bei vielen kleinen Unternehmen der Fall sein, die nicht die Möglichkeit haben eine eigene Infrastruktur zu betreiben. Allerdings geht Kundra davon aus, dass die Cloud-Anbieter ein natürliches Interesse daran haben möglichst sichere Server und Anwendung anzubieten. Dies mag häufig zutreffen, sollte allerdings keinesfalls als selbstverständlich angenommen werden.

Allgemein lässt sich sagen, dass mit der Komplexität einer Anwendung immer auch das Risiko von Sicherheitslücken steigt und Cloud-Anwendungen einen Großteil dieser Komplexität verschleiern und somit das wahrgenommene Risiko reduzieren, nicht aber das wirkliche Risiko.

Literatur

[1] *Umsatz mit Cloud Computing weltweit von 2009 bis 2017 und Prognose bis 2021 (in Milliarden US-Dollar)*. Adresse: https://de.statista.com/statistik/daten/studie/195760/umfrage/umsatz-mit-cloud-computing-weltweit/ (besucht am 20. 08. 2018).

[2] *Cloud Monitor 2018*, Bitkom, 2018. Adresse: https://home.kpmg.com/de/de/home/themen/2016/03/cloud-computing.html (besucht am 16. 09. 2018).

[3] P. Mell und T. Grance, “The NIST Definition of Cloud Computing”, 2011.

[4] Y. Chen, V. Paxson und R. H. Katz, “What’s New About Cloud Computing Security”, *University of California, Berkeley Report No. UCB/EECS-2010-5 January*, Jg. 20, Nr. 2010, S. 2010–5, 2010.

[5] F. Sabahi, “Cloud Computing Security Threats and Responses”, in *Communication Software and Networks (ICCSN), 2011 IEEE 3rd International Conference on*, IEEE, 2011, S. 245–249.

[6] C. Rong, S. T. Nguyen und M. G. Jaatun, “Beyond lightning: A survey on security challenges in cloud computing”, *Computers & Electrical Engineering*, Jg. 39, Nr. 1, S. 47–54, 2013, Special issue on Recent Advanced Technologies and Theories for Grid and Cloud Computing and Bio-engineering. Adresse: http://www.sciencedirect.com/science/article/pii/S0045790612000870.

[7] H.-J. Appelrath, H. Kagermann und H. Krcmar, “Future Business Clouds - Cloud Computing am Standort Deutschland zwischen Anforderungen, nationalen Aktivitäten und internationalem Wettbewerb”, 2014.

[8] *Service Level Agreement nach Gabler Wirtschaftslexikon*, Springer Verlag. Adresse: https://wirtschaftslexikon.gabler.de/definition/service-level-agreement-53580/version-276658 (besucht am 20. 08. 2018).

[9] D. K. Manhart, *Service aus der Cloud - Das müssen Sie bei Cloud-SLAs beachten*. Adresse: https://www.tecchannel.de/a/das-muessen-sie-bei-cloud-slas-beachten,3280170 (besucht am 20. 08. 2018).

[10] N. Marangos, P. Rizomiliotis und L. Mitrou, “Digital forensics in the Cloud Computing Era”, in *2012 IEEE Globecom Workshops*, Dez. 2012, S. 775–780.

[11] *Google Cloud Platform - Creating and Managing Projects*. Adresse: https://cloud.google.com/resource-manager/docs/creating-managing-projects (besucht am 16. 09. 2018).

[12] F. A. P. da Silva, P. A. da Mota Silveira Neto, V. C. Garcia, R. E. Assad und F. A. M. Trinta, “Accounting models for cloud computing: A systematic mapping study”, in *Proceedings of the International Conference on Grid Computing and Applications (GCA)*, The Steering Committee of The World Congress in Computer Science, Computer Engineering und Applied Computing (WorldComp), 2012, S. 1.

[13] C. Chen, P. Maniatis, A. Perrig, A. Vasudevan und V. Sekar, “Towards Verifiable Resource Accounting for Outsourced Computation”, *SIGPLAN Not.*, Jg. 48, Nr. 7, S. 167–178, 2013, ISSN: 0362-1340. Adresse: http://doi.acm.org/10.1145/2517326.2451546.

[14] *What is Azure Cost Management?*, Microsoft, 2018. Adresse: https://docs.microsoft.com/en-us/azure/cost-management/overview (besucht am 16. 09. 2018).

[15] K. Narita, *Business in the Cloud*, Ooma, Inc., 2017. Adresse: https://www.ooma.com/blog/business-in-the-cloud/ (besucht am 16. 09. 2018).

[16] A. Wilkens, *Dropbox zählt 500 Millionen Nutzer*, Heise Online, 2017. Adresse: https://www.heise.de/newsticker/meldung/Dropbox-zaehlt-500-Millionen-Nutzer-3130211.html (besucht am 16. 09. 2018).

[17] N. Santos, K. P. Gummadi und R. Rodrigues, “Towards Trusted Cloud Computing”, *HotCloud*, Jg. 9, Nr. 9, S. 3, 2009.

[18] N. Leavitt, “Is Cloud Computing Really Ready for Prime Time?”, *Computer*, Jg. 42, S. 15–20, Jan. 2009, ISSN: 0018-9162. Adresse: doi.ieeecomputersociety.org/10.1109/MC.2009.20.

[19] *German Federal Administration relies on Nextcloud as a secure file exchange solution*, Nextcloud GmbH, 2018. Adresse: https://nextcloud.com/blog/german-

federal - administration - relies - on - nextcloud - as - a - secure - file - exchange - solution/ (besucht am 16. 09. 2018).

[20] D. E. Geer Jr., "Complexity Is the Enemy", *IEEE Security Privacy*, Jg. 6, Nr. 6, S. 88–88, Nov. 2008, ISSN: 1540-7993.

[21] J. Panettieri, *Hurricane Florence: Amazon, Google, Microsoft Cloud Data Centers In Storm's Path?*, CHANNELe2e, 2018. Adresse: https://www.channele2e.com/channel-partners/csps/hurricane-florence-vs-amazon-google-microsoft-cloud-data-centers/ (besucht am 16. 09. 2018).

[22] O. Nickel, *Blitzschlag trifft Kühlung, trifft Azure, trifft Nutzer*, Golem.de, 2018. Adresse: https://www.golem.de/news/microsoft-blitzschlag-trifft-kuehlung-trifft-azure-trifft-nutzer-1809-136413.html (besucht am 16. 09. 2018).

[23] A. Swinnen und A. Mesbahi, "One packer to rule them all: Empirical identification, comparison and circumvention of current Antivirus detection techniques", *BlackHat USA*, 2014.

[24] J. Beda, *Containers at Scale*, Google, 2014. Adresse: https://speakerdeck.com/jbeda/containers-at-scale (besucht am 16. 09. 2018).

[25] B. Satzger, W. Hummer, C. Inzinger, P. Leitner und S. Dustdar, "Winds of Change: From Vendor Lock-In to the Meta Cloud", *IEEE Internet Computing*, Jg. 17, Nr. 1, S. 69–73, Jan. 2013, ISSN: 1089-7801.

[26] *Managed Security Services Market by Type, Security Type, Organization Size, Deployment Mode, Vertical, and Region - Global Forecast to 2023*, Markets und Markets, 2018. Adresse: https://www.marketsandmarkets.com/Market-Reports/managed-security-services-market-5918403.html (besucht am 16. 09. 2018).