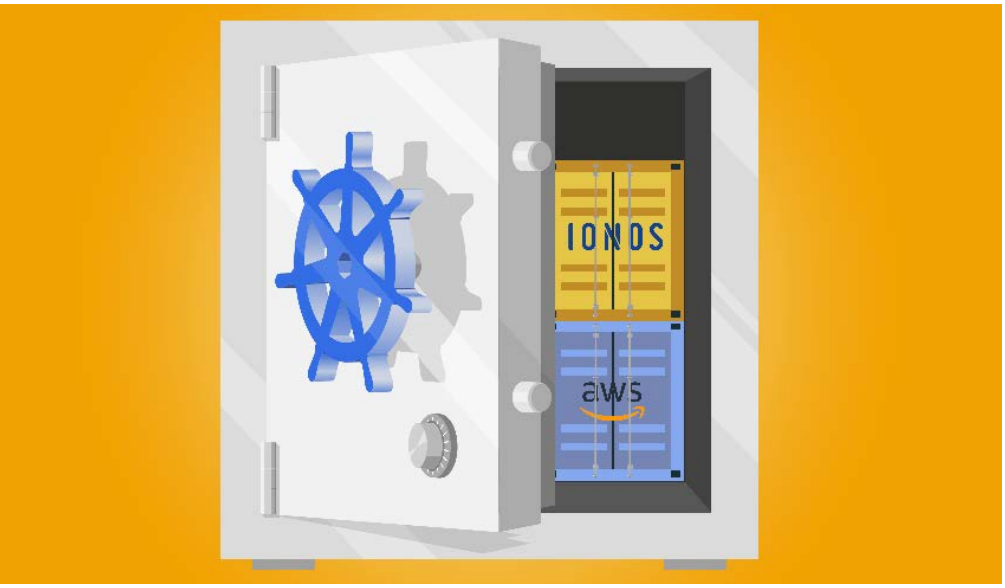


Kubernetes von AWS und IONOS härten

Bei den großen Anbietern ist Kubernetes stärker in die Managed-Angebote integriert und somit variiert das Vorgehen zur Absicherung auch stärker. Nach Google und Azure sind diesmal AWS und IONOS an der Reihe.

Von Christoph Puppe

Für diese Artikelserie wurden neben den bekannten Angeboten auch einheimische Cloud-Service-Provider (CSP) ausgewählt. Digitale Souveränität, einfachere Datenschutz-Folgenabschätzungen und weniger Risiko von Industriespionage durch die Geheimdienste der Heimatländer der CSP sind für viele Organisationen gewichtige Argumente. Auch der Preis der Dienste spricht für die kleineren CSP. Teilweise sind es Größenordnungen, um die sich die Anbieter hier voneinander unterscheiden.

Jeder Anbieter nimmt mit einem Demo-Cluster an der Auswertung teil. Diese sind soweit möglich alle gleich aufgebaut. Die Installation folgt dabei den Empfehlungen des Anbieters, die in den Anleitungen zu finden sind. Die Cluster sind für den Test aus dem Internet erreichbar und auch der Master ist öffentlich zugänglich, auch wenn das in einer Produktionsumgebung natürlich tunlichst zu vermeiden ist. Wo es möglch ist, sind die Master-Server als sogenannte Hidden Master konfiguriert. Dabei ist der Server zwar sichtbar, aber der CSP verwaltet diesen vollständig und der Kunde hat keinen Zugriff auf den Server, auf dem der K8s-API-Server läuft. Viele Sicherheitstests nach dem CIS-Benchmark sind damit nicht möglich.

Dieser Teil der Serie betrachtet erneut die Sicherheit der CSP selbst und listet auf, was Kunden bei diesen Anbietern selbst tun müssen, um eine ausreichende Sicherheit für die Container zu erreichen. Auch wenn die im Artikel „Kubernetes mit Capabilities, SECCOMP und SELinux“ [1] aufgeführten allgemeinen Schritte weiterhin gleich sind, unterscheidet sich die Umsetzung bei den Plattformen erheblich. Die kleineren Anbieter wie IONOS bieten im Vergleich zu den großen Hyperscalern wie AWS ein unverfälschtes Kubernetes. Es ist weniger verändert, die Images kommen eher von den Herstellern der Werkzeuge direkt statt aus der Registry des CSP. Besonders merkbar ist der Unterschied bei der Zahl der zusätzlich verfügbaren Dienste, was sich auch auf die Absicherung auswirkt. Wo Kunden bei AWS, Azure und GCP etwa die Protokollierung als SaaS direkt in der Cloud bekommen, müssen sie sie bei den kleineren CSP mit einigem Aufwand selbst implementieren.

Für Kunden ist der Vergleich der Anbieter – und in späteren Teilen der Artikelserie der Distributionen für den Eigenbetrieb – wichtig. Der erste Schritt zur Absicherung ist dabei die Entscheidung zwischen einem Managed Service oder einer Distribution. Sobald sie getroffen ist, müssen die Kunden ihren Teil der Verantwortung übernehmen und für eine sichere Konfiguration sorgen. In zwei Tabellen sind die Punkte aufgeführt, die vom Anbieter bereits erfüllt sind, die Kunden mit Bordmitteln umsetzen können und für die zusätzliche Software erforderlich ist.

Dieser Artikel konzentriert sich auf die To-do-Liste für AWS und IONOS. Vergleichbare Informationen zu Google und Azure sowie weitere technische Hintergründe finden sich im dreiteiligen Tutorial zum Absichern von Kubernetes [2–4]. Mit den Härtungsaspekten bei OpenShift, Rancher, plusserver und der Open Telekom Cloud werden sich weitere iX- Artikel befassen.

iX-TRACT

- IONOS bietet für wenig Geld ein Managed Kubernetes mit Hidden Master an.
- AWS hat ein sehr umfangreiches Angebot und Kubernetes stark in die eigenen Dienste integriert, besonders IAM.
- Nutzer müssen bei AWS viele Sicherheitsfunktionen aktivieren, die bei anderen Anbietern Standard sind.

Digital, sicher, souverän

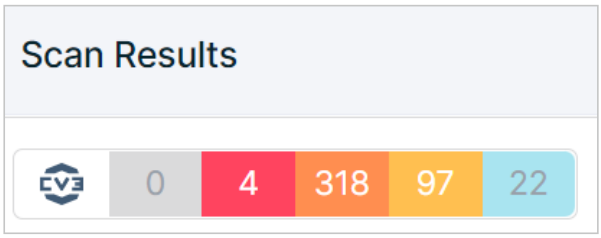
Souveräner IT-Betrieb bezeichnet den Einsatz von eigenen Systemen, die Anmietung von Servern und die Nutzung von SaaS aus heimischen Landen, also Angeboten, bei denen die Hardware in Deutschland steht, aber auch der Betrieb durch eine Firma in der Bundesrepublik erfolgt. In den letzten Jahren ist dieses Thema als Strategie für die öffentliche Hand immer wichtiger geworden und seit 2021 im Beschluss [09/21] auch die offizielle Linie (siehe ix.de/znx2). Neben dem Datenschutz, der eine der wichtigen Triebfedern dieses Trends ist, ist während der Präsidentschaft von Trump auch das Thema Unabhängigkeit von anderen Staaten mehr in den Fokus gerückt. Deutschland soll seine Verwal-

tungs-IT so weit selbst im Griff und unter Kontrolle haben, dass wir das Staatswesen auch bei überraschenden Ereignissen weiterführen können. Für die Privatwirtschaft ist das Thema auch relevant, sofern die USA Sanktionen verhängen, die die Bundesrepublik nicht mitträgt, und deutsche Firmen deswegen den Zugang zu ihren Cloud-Anbietern verlieren könnten.

Für die öffentliche und auch die private Hand ist der Datenschutz von außerordentlicher Bedeutung, und da es bis heute keine Rechtssicherheit beim Einsatz von Cloud-Diensten aus dem EU-Ausland gibt, sind die Anbieter mit Sitz der Firma und der Rechenzentren in der EU hier im Vorteil. Eine Übertragung der Daten ins Ausland ist bei den heimischen Anbietern ausgeschlossen – natürlich sofern sie nicht gehackt werden.

Genutzte Demo-Cluster

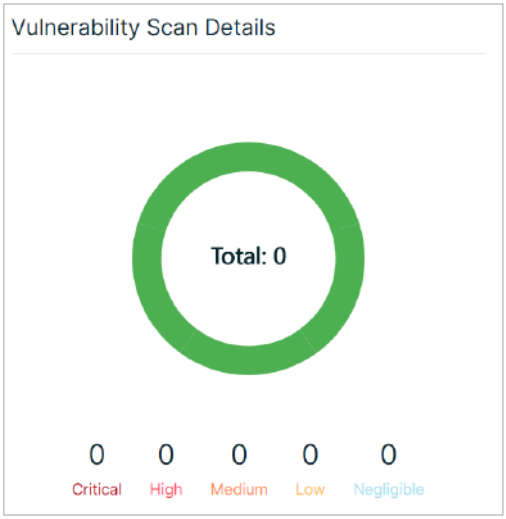
IONOS: Während andere CSPs auf CLI und Terraform setzen, ist IONOS klar auf eine andere Zielgruppe ausgerichtet. Es begrüßt mit einem grafischen Datacenter-Designer, der in etwa dem entspricht, was bei anderen Anbietern das Projekt oder die Subscription ist. Zuerst muss man ein leeres Datacenter anlegen und danach über den Menüpunkt „Container/Kubernetes“ ein Managed Kubernetes und einen Node Pool erstellen. Die Bedienungsführung ist im Vergleich zu anderen CSPs etwas gewöhnungsbedürftig, aber der erste Cluster ist schnell aufgebaut. Die Optionen sind übersichtlich, das Wichtigste ist abgedeckt.



Der Scan nach Verwundbarkeiten auf den Nodes bei IONOS findet wenige/keine schwere Sicherheitslücken. Negativ ist nur, dass der Scan überhaupt möglich war, da der AquaSec Enforcer dafür erhebliche Rechte auf dem Node benötigt (Abb. 1).

Insgesamt sind wenige Dienste vorinstalliert. Das erklärt auch die geringe Zahl der Verwundbarkeiten, die Trivy auf den Nodes fand (siehe Abbildung 1). Die Images kommen bei IONOS direkt vom Hersteller oder aus dem Docker-Hub, eine eigene Registry ist nicht im Einsatz.

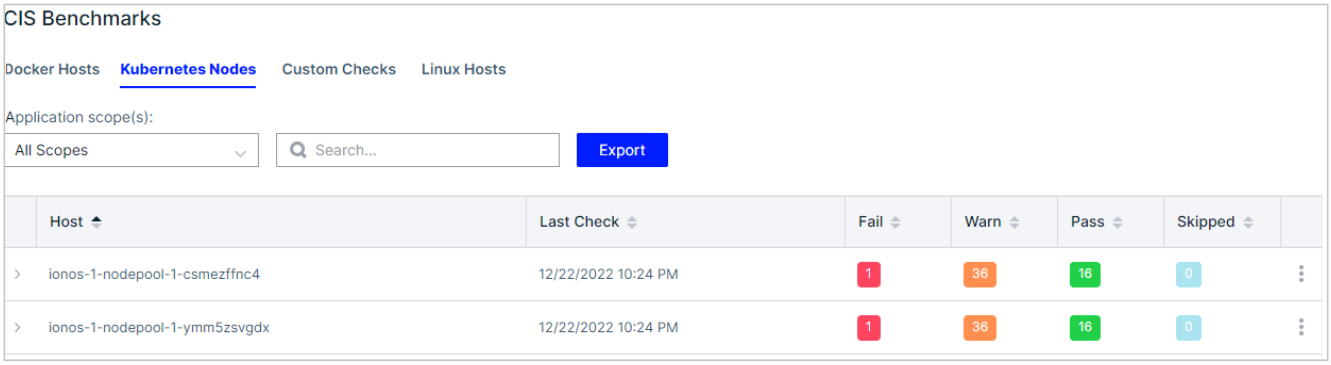
AWS: Ein Amazon Elastic Kubernetes Service (EKS) ist schnell gebaut. Es sind allerdings einige Vorarbeiten wie das Anlegen von Netzwerken, Sicherheitsgruppen und Rollen erforderlich. Für diesen Artikel entstand ein Standard-EKS-Cluster, der möglichst nahe am von AWS in den Anleitungen beschriebenen und empfohlenen Aufbau ist. Auf diese Weise sind öffentliche und private Netze in der Virtual Private Cloud (VPC) und auch der Hidden Master des Managed Kubernetes ist aus dem Internet erreichbar. Nach dem Aufbau des Clusters benötigt dieser noch Node Pools, die man in einem zweiten Schritt hinzufügt. Auf den Nodes lagen direkt nach der Installation mit den drei empfohlenen Plug-ins und allen aktiven Protokollierungsoptionen rekordverdächtige 302 Images – allein von `amazonaws.com/amazon-k8s-cni` gab es acht Versionen. Ein Angreifer mit Zugang zu einem Node kann jedes dieser



Der Scan der Nodes des EKS-Clusters ist vorbildlich: Es wurden keine Schwachstellen oder Malware gefunden. Nur die auf den Nodes liegenden Images bergen Sicherheitslücken (Abb. 2).

Images starten, ohne dass eine weitere Prüfung erfolgt.

Dieser Cluster dient als Grundlage für die Beschreibung der notwendigen Schritte zur Absicherung. Wo der Serverless-Container-Dienst Fargate die Absi-



Die CIS-Benchmarks für die Nodes waren bei IONOS möglich, bei anderen CSP nicht. Die IONOS-Nodes kommen gut durch den Benchmark (Abb. 3).

cherung einfacher macht, ist dies in der Tabelle erwähnt.

Hintergrundinformationen zur Härtungsliste

Die beiden Tabellen „Empfehlungen zum BSI-IT-Grundschutz-Baustein SYS.1.6 Containerisierung“ und „Empfehlungen zum BSI-IT-Grundschutz-Baustein APP.4.4 Kubernetes“ enthalten nur die Anforderungen, die technisch auf dem Cluster umzusetzen sind. Die organisatorischen Anforderungen wie Regelungen für den Betrieb von Containern, Freigaben oder Aktualisierungen liegen in der Verantwortung der Kunden. Auch bleiben hier die Anforderungen unberücksichtigt, die den eigentlichen Betrieb des Servers betreffen. Da bei den Managed-Kubernetes-Angeboten der Cloud-Service-Provider für den Betrieb in der Pflicht ist, ist von den Anforderungen der Bausteine „SYS.1.1 Allgemeiner Server“ und „SYS.1.3 Server unter Linux“ nur „SYS.1.1.A16 Sichere Grundkonfiguration von Servern (S)“ relevant. Mit aufgeführt sind drei übergeordnete Anforderungen, die das BSI im Zuge der Modularisierung des IT-Grundschutz-Kompendiums aus den SYS-Bausteinen in die oberen Schichten verlagert hat. Die Kommentare zur Umsetzung sind farblich hervorgehoben. Die Anforderungen werden wie folgt nach der Art der Umsetzung kategorisiert:

Empfehlungen zum BSI-IT-Grundschutz-Baustein APP.4.4 Kubernetes

Anforderung[1]	IONOS[2]	AWS[2]
APP.4.4.A1 Planung der Separierung der Anwendungen (B)	Private Netze müssen Nutzer selbst anlegen.	Die geplanten Netzzonen und die Separierung in Namespaces und Cluster ist umzusetzen.
APP.4.4.A3 Identitäts- und Berechtigungsmanagement bei Kubernetes (B)	Nutzer können den Zugriff auf den Cluster im IAM einrichten, dies gilt aber nur für die Verwaltung und nicht für die K8s-API. Dies ist von Nutzern selbst zu implementieren.	Zugangsrechte sind über IAM geregelt und Nutzer müssen sie bei der Installation einrichten.
APP.4.4.A4 Separierung von Pods (B)	Die Pods sind mit den Standards auf einem Linux-Kernel umgesetzt. Weiter gehende Separierung scheint nicht möglich.	Die Pods sind mit den Standards auf einem Linux-Kernel umgesetzt. Seit Ende November 2022 sind auch Confidential-Computing-Enklaven (Nitro Enclaves) für EKS möglich.
APP.4.4.A5 Datensicherung im Cluster (B)	Für die Umgebung muss der Nutzer eine Backup Unit anlegen und dort die selbst zu erstellenden Backups speichern.	Cluster Volumes sind aus dem EBS und werden dort gesichert. Dies ist vom Nutzer einzustellen. AWS verweist auf Velero, dessen Nutzung dokumentiert ist.
APP.4.4.A7 Separierung der Netze bei Kubernetes (S)	siehe SYS.1.6.A5	siehe SYS.1.6.A5
APP.4.4.A9 Nutzung von Kubernetes Service-Accounts (S)	Nutzer müssen dies für ihre Cluster mit den K8s-Optionen konfigurieren.	Per Plug-in können die IAM-User aus AWS für Pods eingesetzt werden. Diese können dann auch Rollen im AWS nutzen.
APP.4.4.A12 Absicherung der Infrastruktur-Anwendungen (S)	Es gibt bis auf die sehr rudimentäre Registry keine weiteren Infrastrukturanwendungen. Nutzer müssen hier selbst Hand anlegen und eine Software installieren.	Abhängig von den verwendeten Anwendungen. CodeBuild, CodePipeline und anderen AWS-Tools ermöglichen die Umsetzung, sofern keine 3rd-Party-Werkzeuge im Einsatz sind.
APP.4.4.A13 Automatisierte Auditierung der Konfiguration (H)	Nutzer müssen hier selbst Hand anlegen und eine Software installieren.	Ist in AWS Config und dem Security Hub möglich, verursacht allerdings zusätzliche Kosten und ist vergleichsweise umständlich. Einen CIS-Benchmark für EKS bietet AWS über Kube-Bench von AquaSec.
APP.4.4.A14 Verwendung dedizierter Nodes (H)	Management Nodes sind separat und in der Verwaltung durch den Anbieter, daher gibt es für Nutzer keine Shell-Zugriffe. Nodes für Speicher sind nicht notwendig. Die Aufgabe der Bastion Nodes übernehmen die Loadbalancer als Sicherheitsgateway, sofern der Nutzer dies einrichtet.	Management Nodes sind separat und in der Verwaltung durch den Anbieter, daher gibt es für Nutzer keine Shell-Zugriffe. Nodes für Speicher sind nicht notwendig. Die Aufgabe der Bastion Nodes übernehmen die Loadbalancer als Sicherheitsgateway.
APP.4.4.A16 Verwendung von Operatoren (H)	Nutzer müssen hier selbst Hand anlegen und eine Software installieren.	Nutzer müssen hier selbst Hand anlegen und eine Software installieren.
APP.4.4.A17 Attestierung von Nodes (H)	nicht möglich	Attestierung ist für die Nitro Enklaven möglich. Diese sind seit Kurzem auch für EKS verfübar.
APP.4.4.A18 Verwendung von Mikro-Segmentierung (H)	Nutzer müssen hier selbst Hand anlegen und eine Software installieren.	Nutzer müssen hier selbst Hand anlegen und eine Software installieren.
APP.4.4.A19 Hochverfügbarkeit von Kubernetes (H)	siehe SYS.1.6.A25	siehe SYS.1.6.A25
APP.4.4.A20 Verschlüsselte Datenhaltung bei Pods (H)	Nutzer müssen hier selbst Hand anlegen und eine Software installieren.	Die Datenhaltung ist per Default verschlüsselt. Optional können Nutzer eigene Schlüssel einsetzen.

[1](B) Basisanforderung, (H) Anforderung bei erhöhtem Schutzbedarf, (S) Standardanforderung; [2]Erläuterung der Farben siehe Text

Von AquaSec Enforcer gefundene Verwundbarkeiten

	Critical	High	Medium	Low	Summe
AWS	2070	3901	2490	270	8731
IONOS	25	124	55	6	210
Aqua	0	2	2	0	4
Azure[1]	240	1111	630	85	2066
GCP[1]	2	44	14	1	61

[1]zum Vergleich die bisherigen Ergebnisse anderer Plattformen [3]

- Grün: Die Standardinstallation über die Weboberfläche reicht aus. Es ist keine weitere Konfiguration erforderlich. Wer eine andere Methode zum Erstellen des Clusters nutzt, muss auch die grün markierten Anforderungen explizit aktivieren.
- Rot: Nutzer können und sollten die Anforderungen mit den Bordmitteln umsetzen, die notwendige Konfiguration ist jedoch bei der Standardinstallation über die Weboberfläche nicht voreingestellt.
- Gelb: Es ist zusätzliche Software erforderlich.
- Blau: Die Umsetzung ist nicht möglich. Informationen, wo die Umsetzung empfohlen ist, finden sich im zugehörigen Kommentar.

Verwundbarkeiten im Cluster

Wie im letzten Artikel der Serie [3] beschrieben, wird jeder der CSP auch auf die Sicherheit der Cluster untersucht. Da die CIS-Benchmarks mangels Rechten auf den Master-Servern ausfallen, beschränkt sich die Untersuchung bei AWS auf den Stand der Images. Trivy, AquaSecs Image-Scanner für Verwundbarkeiten und fehlende Patches, hat auf dem Demo-Cluster die Images auf den Nodes untersucht. Ein AquaSec Enforcer dient dazu, die Images auf den Nodes zu erreichen.

AWS lagert auf den Nodes viele Versionen von Images. Das kann aber nicht der einzige Grund für die tausendmal höhere Zahl der Verwundbarkeiten im Vergleich zu GCP sein. Vor allem sticht der überproportional hohe Anteil der kritischen Verwundbarkeiten ins Auge. Azure ist hier im Mittelfeld mit immerhin rund hundertmal so vielen kritischen Schwachstellen. Auch wenn alte Images nur problematisch sind, wenn sie auch gestartet werden, ist dies das Bild, das sich Nutzern, die eine CWPP (Cloud Workload Protection Platform) einsetzen, nach der Installation bietet.

Wie man bei GCP sehen kann, ist es möglich, auch Images fast ohne Verwundbarkeiten in Betrieb zu haben. Interessant ist auch, dass einige Sicherheitsprogramme selbst viele Schwachstellen enthalten. Der Autor verwendet daher einen

Empfehlungen zum BSI-IT-Grundschutz-Baustein SYS.1.6 Containerisierung

Anforderung[1]	IONOS[2]	AWS[2]
SYS.1.1.A16 Sichere Grundkonfiguration von Servern (S)	Es wird ein Ubuntu 20.04.5 LTS als Image verwendet. Die Härtung des Images wurde nicht untersucht. Scan der Verwundbarkeiten siehe Abb. 1 und 2.	Es gibt neben dem Amazon Linux 2 auch Bottlerocket und Windows als Images. Die Härtung des Images wurde nicht untersucht. Ergebnisse des Scans nach Verwundbarkeiten siehe Abb. 1 und 2.
OPS.1.1.2.A18 Durchgängige Protokollierung administrativer Tätigkeiten (H)	Dies scheint zu fehlen, zumindest hat der Autor diese Funktion nicht gefunden.	Nutzer müssen CloudTrail für alle Regionen aktivieren, in denen Ressourcen sind. Die Logs über Veränderungen und Ereignisse im Cluster selbst (Scheduler, Authenticator und API, Controller, Manager, Audit) sollten Nutzer an CloudWatch leiten lassen.
OPS.1.1.3.A1 Konzept für das Patch- und Änderungsmanagement [Fachverantwortliche] (B)	Auto-Update ist eingebaut. Es wird ein Wartungsfenster in der Konfiguration eingestellt.	Auto-Update lässt sich beim Erstellen einstellen.
SYS.1.6.A5 Separierung der Administrations- und Zugangsnetze bei Containern (B)	Nutzer müssen private Netze selbst anlegen. Die Konfiguration, den API-Server nur aus internen Netzen zu erreichen, war laut FAQ möglich, aber in der Oberfläche nicht zu finden. Die API enthält die Einstellung.	Ein privates VPC muss vom Nutzer angelegt werden. Sofern so geschehen, ist die K8s-API nur aus dem internen Netz erreichbar. EKS hat interne Netze, die nicht aus den anderen VPC erreichbar sind.
SYS.1.6.A6 Verwendung sicherer Images (B)	Die optional einzustellende Registry ist sehr einfach. Funktionen wie Scans der Images nach Verwundbarkeiten oder Regeln sind nur mit zusätzlicher Software möglich.	Die von AWS bereitgestellte Registry bietet Scans nach Verwundbarkeiten. Verwundbarkeiten können Events auslösen, die Reaktion kann dann beispielsweise einen neuen Build auslösen. Im Security Hub sind die Ergebnisse zu sehen. Der Nutzer kann sich zwischen Basic und Enhanced Scan entscheiden.
SYS.1.6.A7 Persistenz von Protokollierungsdaten der Container (B)	Nutzer müssen hier selbst Hand anlegen und die Persistierung der Logs implementieren.	Nutzer müssen manuell eine Log-Group erstellen und dort die Logs für Nodes und Container hineinleiten.
SYS.1.6.A8 Sichere Speicherung von Zugangsdaten bei Containern (B)	Nutzer müssen hier selbst Hand anlegen und eine Software installieren.	Nutzer müssen die Verschlüsselung von Secrets einstellen. AWS speichert Parameter, Passwörter und Zertifikate entweder in SSM oder KSM. Beide können mit einem eigenen Schlüssel verschlüsselt werden.
SYS.1.6.A10 Richtlinie für Images und Container-Betrieb (S)	Nutzer müssen hier selbst Hand anlegen und eine Software installieren.	Nutzer müssen hier selbst Hand anlegen und eine Software installieren.
SYS.1.6.A18 Accounts der Anwendungsdienste (S)	siehe APP.4.4.A3	siehe APP.4.4.A3
SYS.1.6.A21 Erweiterte Sicherheitsrichtlinien (H)	Nutzer müssen hier selbst Hand anlegen und eine Software installieren.	Nutzer müssen hier selbst Hand anlegen und eine Software installieren.
SYS.1.6.A22 Vorsorge für Untersuchungen (H)	Nutzer müssen hier selbst Hand anlegen und eine Software installieren.	Nutzer müssen hier selbst Hand anlegen und eine Software installieren.
SYS.1.6.A24 Hostbasierte Angriffserkennung (H)	Nutzer müssen hier selbst Hand anlegen und beispielsweise eine Überwachung der Logs einrichten (siehe auch OPS.1.1.2.A18) oder entsprechende Werkzeuge zur Erkennung von Angriffen installieren.	Nutzer müssen hier selbst Hand anlegen und beispielsweise eine Überwachung der Logs einrichten (siehe auch OPS.1.1.2.A18) oder entsprechende Werkzeuge zur Erkennung von Angriffen installieren.
SYS.1.6.A25 Hochverfügbarkeit von containerisierten Anwendungen (H)	Nodes sind über maximal zwei Zonen in einer Region verteilt.	Default sind drei Zonen in einer Region.

[1](B) Basisanforderung, (H) Anforderung bei erhöhtem Schutzbedarf, (S) Standardanforderung; [2]Erläuterung der Farben siehe Text

Eine Auswertung der am häufigsten betroffenen Pakete mit Verwundbarkeiten der auf den Nodes liegenden Images der AWS Worker ergab diese Übersicht (Abb. 4).

WERT	HÄUFIGKEIT
curl	2.640
glibc	1.920
expat	960
ncurses	720
systemd	540

Scanner, der selbst keine Verwundbarkeiten enthält.

Die Tabelle mit den von AquaSec Enforcer gefundenen Verwundbarkeiten führt alle Ergebnisse der Scans aus allen Artikeln dieser Reihe zusammen und enthält auch die Daten für Plattformen, die dieser Teil nicht bespricht. Somit ist eine Vergleichbarkeit zwischen allen Anbietern in dieser Artikelreihe gegeben.

Fazit

IONOS bietet mit seinem Managed K8s einen soliden Service, der immerhin Hidden Master beherrscht, ansonsten aber wenig Integration in andere Dienste bietet. Das ist ein für kleinere CSP typisches Angebot, und in den nächsten Ausgaben vergleichen wir noch weitere deutsche Anbieter mit einem sehr ähnlichen Funktionsspektrum. Die Angebote sind hier eher mit den K8s-Distributionen zu vergleichen, da IONOS ein größtenteils unverändertes K8s bereitstellt.

Anders sieht es bei AWS aus. Wie auch bei den anderen großen CSP sind hier die weiteren Dienste des CSP stärker in das Managed K8s integriert. Insgesamt allerdings nicht so intensiv wie bei den Mitbewerbern. Etwas irritierend ist die Zahl der Verwundbarkeiten auf den von AWS auf den Nodes abgelegten Images – immerhin tausendmal so viele kritische Verwundbarkeiten wie bei GCP. Drei Größenordnungen Unterschied sind nicht nur mit den alten Versionen der Images auf den Nodes zu erklären. (avr@ix.de)

Quellen

[1] Christoph Puppe, Jan Walther; Kubernetes mit Capabilities, SECCOMP und SELinux; iX 12/2022, S. 60

[2] Christoph Puppe; Tutorial: Kubernetes absichern, Episode 1; iX 12/2022, S. 54

[3] Christoph Puppe; Kubernetes-Tutorial, Teil 2: IAM erklärt, AKS und GKE härten; iX 1/2023, S. 126

[4] Christoph Puppe; Tutorial: Admission Controller überwachen K8s-API, Episode 3; iX 2/2023, S. 132

CHRISTOPH PUPPE

ist Managing IT Security Architect und Auditteamleiter für ISO 27001 nach Grundschutz bei SVA System Vertrieb Alexander GmbH, Mitautor des Grundschutz-Kompendiums und ehemaliger Penetrationstester.