Teure Agilität

Sicherheit für Container

Allzu häufig wird die IT-Sicherheit vernachlässigt, das ist bei Containern nicht anders als im Rest der (IT-)Welt. Wie kann man es besser machen und die Sicherheit von Anfang an einbeziehen?

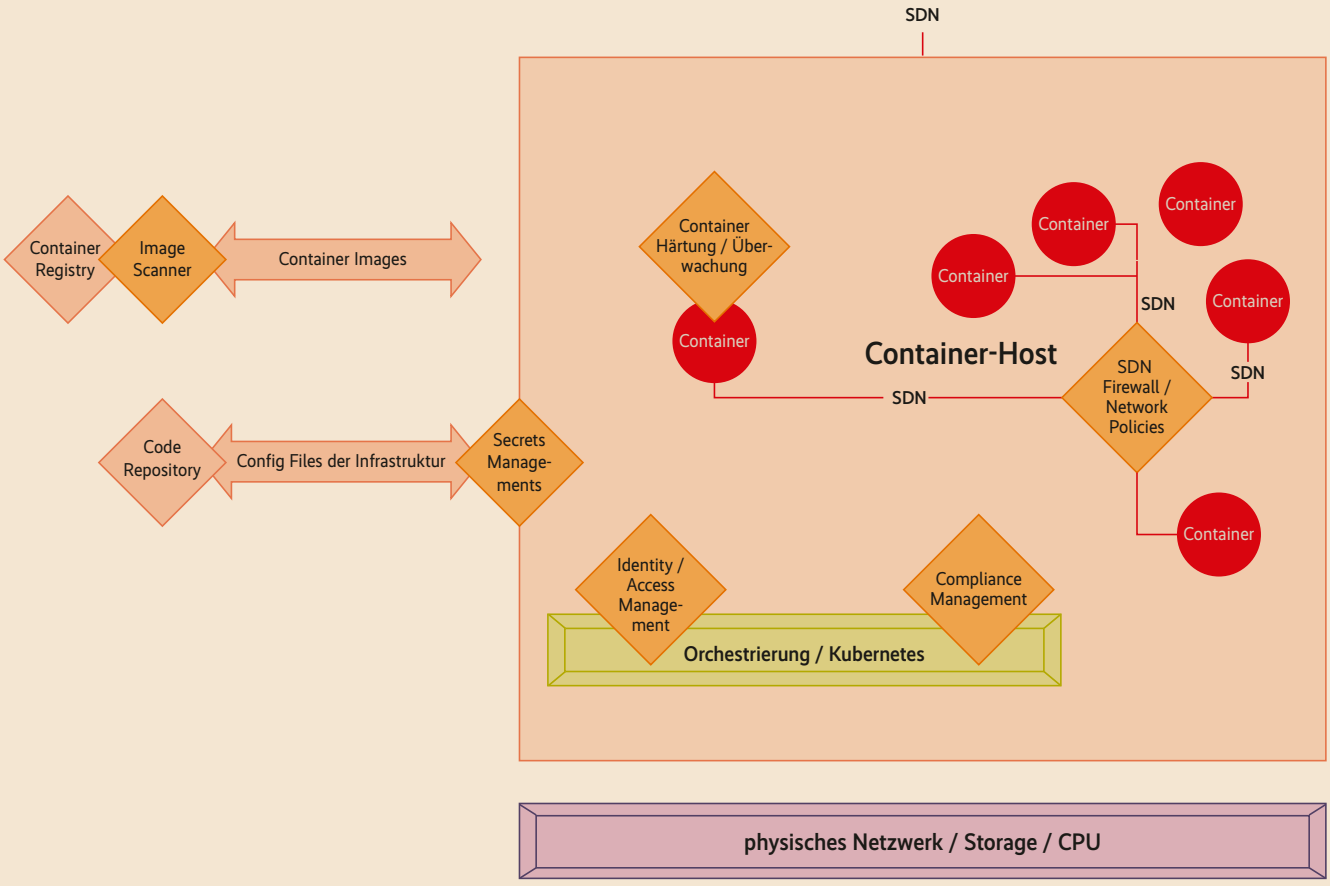
Agile IT und Container bringen immense Vorteile. Leider wird das, was dauert und was kostet, ohne einen direkt ersichtlichen oder messbaren Nutzen zu bringen, häufig nicht beachtet: die Sicherheit der Infrastruktur und auch der Anwendungen in der schönen neuen Container-Welt. Zu viele DevOps-Teams vergessen, die Absicherung in ihre agile Kultur mit einzubauen.

Hier helfen DevSecOps, also die Integration von Sicherheit in den Entwicklungsprozess von Anfang an, und das von Google entwickelte Site Reliability Engineering (SRE) mit dem Ziel der Sicherstellung des täglichen Betriebs und als Regelwerk für auftretende Störungen. Die Grundpfeiler für eine agile und sichere Entwicklung und den Betrieb sind:

- Standardisierung und Automatisierung: Manuelle Veränderungen sollten nicht vorkommen, so reduziert sich die Wahrscheinlichkeit von unabsichtlichen Fehlern.
- Scans des Codes und der Images: In jeder Phase ist das Einhalten von Sicherheitsvorgaben automatisiert zu kontrollieren und dies fest in die Pipelines einzubauen.
- IAM (Identity and Access Management) und RBAC (Role-based Access Control): Identitätsmanagement und rollenbasierte Rechte nicht nur für Personen, sondern auch für alle Prozesse.
- Separierung und Abgrenzung: Zwischen Prozessen, Services und mindestens der Verwaltung der Umgebungen ist eine ausreichende Sicherheitsgrenze zu ziehen;
- Schnelle Patches: Jedes Image und jeder Server muss ständig und umgehend neue Versionen erhalten können.

Vertrauen ist gut, Monitoring besser

Grundlegend geändert hat sich hier das Monitoring auf Verfügbarkeit. Früher wurde gemessen, ob ein Prozess läuft oder ein Port offen ist, und wer gut war, hat die Anwendung einen Statuscode ausgeben lassen. Heute erwarten alle, dass Anwendungen immer laufen und dass sie immer antworten. Die Herausforderung jetzt ist die Geschwindigkeit der Antwort und ob diese auch den Erwartungen entspricht. Der Betrieb hat sich weiterentwickelt zu einer umfassenden Überwachung aller Anfragen auf Korrektheit und Latenz. Google hat mit SRE hier neben dem be-



Ein Orchestrierungstool wie Kubernetes verwaltet und steuert den Einsatz zahlreicher Sicherheitswerkzeuge, die an unterschiedlichen Ebenen ansetzen.

kannten Service Level Agreement (SLA) noch die Service Level Objective (SLO) und den Service Level Indicator (SLI) festgelegt. Beide definieren die Messung der Verfügbarkeit und sind nicht auf „an“ oder „aus“ beschränkt.

Der zweite Aspekt des Monitorings ist die Überwachung, ob Sicherheitsvorgaben eingehalten werden. Moderne Betriebsumgebungen sind aus vielen Schichten aufgebaut, und auf jeder Schicht gibt es Hunderte von Einstellungen, die die Sicherheit beeinflussen.

Hier wie überall sonst ist Automatisierung notwendig, um diese Einstellungen zu überwachen, Fehler zu beheben und sie über alle Betriebsstätten hinweg im Blick zu haben.

Wer seine Anwendungen nicht nur im eigenen Rechenzentrum auf vielen Clustern und Umgebungen hat, sondern auch in der Virtual Private und der Public Cloud betreibt, hat schnell Dutzende unterschiedlicher Benutzeroberflächen zu bedienen, um die aktuelle Sicherheit zu auditieren.

Werkzeuge zum Compliance-Management helfen hier sehr gut weiter und sind in größeren Umgebungen die einzige Chance, den Überblick zu behalten. Auch wenn diese Themen in der „alten“ Welt ohne Container schon relevant waren – mit Containern steigt die Flexibilität, die Betriebsumgebungen sind um Größenordnungen komplexer, und somit ist die Notwendigkeit der Automatisierung solcher Überwachungsvorgänge noch dringlicher.

Mächtige Werkzeugkästen im Einsatz

Das Compliance-Management ist teilweise auch in den verfügbaren Werkzeugkästen für die Absicherung der Container und ihrer Betriebsumgebungen enthalten. Oft sind spezialisierte Werkzeuge da noch etwas umfassender, aber zumindest ein Abgleich aller Einstellungen mit den Vorgaben der Sicherheits-Benchmarks des Center for Internet Security (CIS) ist eigentlich in allen enthalten. Das CIS erstellt sehr umfangreiche Listen mit technischen Vorgaben, die zumeist die Konfiguration der Dienste betreffen. Diese sind meist mehrere Hundert Seiten lang und ein manueller Abgleich ist sehr mühselig. Um eine ständige Überwachung der Sicherheit zu ermöglichen, sollte diese Aufgabe für alle Betriebsumgebungen automatisiert sein.

Wer die Werkzeugkästen und die spezialisierten Tools der Containerabsicherung verstehen will, muss wissen, dass Container und das damit heute fast immer mit im Einsatz befindliche Orchestrierungstool Kubernetes (K8S) nicht einfach eine andere Art sind, einen Prozess zu starten, sondern dass K8S eigentlich den gesamten RZ-Betrieb übernimmt. Es steuert die softwaredefinierten Netzwerke, verwaltet die Firewallregeln in diesen Netzwerken, überwacht die Anwendungen, startet und stoppt die Prozesse, verlagert Prozesse von ausgefallenen Servern auf andere, und wenn es zu wenige Server sind, bestellt es gleich noch welche dazu.

Auch steuert Kubernetes die Speicher-Backends und weist den Speicher den Anwendungen zu. Allerdings ist K8S eher der Vermittler zwischen einer Vielzahl an Werkzeugen, die jeweils einen Job erfüllen.

Unter der Haube von K8S arbeiten Dutzende Tools zusammen, die speichern, wie die Umgebung aussehen soll, verifizieren, dass der Nutzer ausreichende Rechte hat, den aktuellen Stand überwachen und den gewünschten Zustand wiederherstellen.

Und genau da setzen diese Multi-Tools an. Zumeist übernehmen sie drei wichtige Grundfunktionen:

- das Überwachen von Netzwerkregeln;
- das Überwachen von Verhaltensregeln wie SECCOMP und APPArmor oder SELinux;
- das Erzwingen von Compliance-Monitoring der Server und der Container sowie von K8S selbst.

Teilweise ist noch die Überwachung und die Reaktion auf Sicherheitsverstöße mit enthalten, wobei hier der Fokus der einzelnen Produkte etwas unterschiedlich ist. Manche sind eher aus dem Prozessmonitoring und mit Host-based Intrusion Detection vergleichbar, während andere mehr die Netzwerkthemen inklusive Network-based Intrusion Detection bedienen.

Allen gemeinsam ist jedoch, dass sie über eine rein passive Überwachung hinausgehen und bei Vorfällen auch eingreifen, indem sie beispielsweise verdächtige Prozesse beenden, Netzwerkregeln anpassen und Container neu starten, um den Angreifer nicht weiter vordringen zu lassen. Sie gehen sogar bis hin zur Automatisierung der Forensik, indem sie bei Vorfällen alle Daten sammeln, die für die Aufklärung durch einen Menschen hilfreich sein können.

Rechtliche Vorgaben

Für viele Organisationen stellen die Richtlinien der Kreditkartenindustrie (PCI), die KRITIS-Verordnung oder das Grundschutz-Kompendium des BSI rechtlich verbindliche Vorgaben oder zumindest starke Empfehlungen dar, mit denen sie die Absicherung der Daten auch in Sachen Datenschutz nachweisen können. Hier hat das BSI immerhin schon vor drei Jahren den ersten Entwurf eines Bausteins für den Betrieb von Containern zur Kommentierung gestellt und Anfang dieses Jahres stark überarbeitet (siehe ix.de/zfaq). (ur@ix.de)

Quellen

Der Community Draft des BSI zur Absicherung von Containern ist unter ix.de/zfaq zu finden.

Christoph Puppe
ist Security Consultant bei SVA, Auditteamleiter für ISO 27001 nach Grundschutz, Mitautor des Grundschutz-Kompendiums und ehemaliger Penetrationstester.