Benutzerdefinierter Baustein

HISOLUTIONS

SYS: IT-Systeme

SYS.bd.2: Container unter openshift

1 Beschreibung

1.1 Einleitung

Container stellen eine besondere Form der Anwendungsbetriebsbereitstellung dar. Die eigentliche Anwendungssoftware wird mit sämtlichen für deren Ablauf erforderlichen Softwarekomponenten (Bibliotheken, Middleware etc.) in einem Container zusammengefasst.

Auf zentral bereitgestellte Dienste außerhalb des Containers wird über definierte Schnittstellen zugegriffen. Technisch wird der Container über ein Container-Image (Abbild) sowie eine (betriebssystemspezifische) Laufzeitumgebung auf dem Host-System umgesetzt. Eine Container-Build-Umgebung unterstützt darüber hinaus über definierte Schnittstellen die automatisierte Erzeugung von Container-Images aus einer Menge vorgegebener Software-Artefakte. In ähnlicher Form unterstützt eine Container-Laufzeitumgebung wiederum über definierte Schnittstellen die Ablaufkontrolle wie etwa das Starten oder Stoppen eines Containers. Die Container-Laufzeitumgebung stellt über das Host-Betriebssystem Systemaufrufe für die im Container enthaltenen Softwarekomponenten zur Verfügung. Damit teilen sich auf einem Host-System mehrere Container den jeweiligen Betriebssystem-Kernel.

Die Verwendung von Containern bietet viele Vorteile für die Entwicklung und den Betrieb von Anwendungen in einem Informationsverbund. Ähnlich wie bei einer Hardware-Virtualisierung werden Anwendungen in ihren jeweiligen Containern voneinander isoliert betrieben und nutzen hierbei gemeinsam Ressourcen des jeweiligen Host-Systems. Im Gegensatz zur klassischen Hardware-Virtualisierung sind Container je-doch sehr klein gehalten, da sie kein vollständiges Betriebssystem mehr enthalten. Darüber hinaus können Container wesentlich schneller gestartet werden, da in der Regel kein Neustart des Betriebssystems erforderlich ist.

Schließlich beinhalten Container sämtliche benötigten Softwarekomponenten in der jeweils erforderlichen Version. Anwendungen werden also in einer einheitlichen Softwareumgebung entwickelt, getestet und betrieben. Auch erübrigt sich bei Bereitstellung als vorkonfigurierter und ablauffähiger Container die Installation per Installationspaket auf dem Zielsystem.

Die zum Erstellen von Containern benötigten Artefakte werden meist in Repositories oder in Quellcode-Managementsystemen vorgehalten. Die Vorhaltung von fertigen Container-Images erfolgt in Registries. Container-Build-Umgebung, Container-Laufzeitumgebung und Registries bilden zusammen mit der zugrunde liegenden Infrastruktur die openshift-Plattform.

Container werden während ihrer Laufzeit nicht verändert. Bewegungsdaten und Betriebsstatus werden im Arbeitsspeicher gehalten und ggf. auf externen Speichersystemen persistiert. Bei Updates

oder Patches werden jeweils neue Container-Images erstellt und gegen die bereits im Betrieb befindlichen Container ausgetauscht. Hierdurch ergeben sich grundlegende Sicherheitsvorteile gegenüber herkömmlichen Update- und Patch-Prozessen.

1.2 Zielsetzung

Zielsetzung dieses benutzerdefinierten Bausteins ist der Schutz von Informationen, die im Zusammenhang mit Container unter openshift erstellt, gelesen, bearbeitet, gespeichert oder versandt werden.

1.3 Abgrenzung

Der benutzerdefinierte Baustein SYS.bd.2 *Container unter openshift* beinhaltet allgemeine Maßnahmen im Kontext einer Container-Bereitstellung und der physischen Hardwarekomponenten der Server-Systeme. Themen wie Platform as a Service, Software-defined Networking (SDN), Software-defined Storage (SDS) und DevOps werden in diesem Baustein nicht behandelt, sofern sie nicht für die technische Umsetzung relevant sind.

Container spielen eine wichtige Rolle bei der Umsetzung von kontinuierlichen Entwicklungs-, Bereitstellungs-, Test- und Inbetriebnahmeprozessen (Continuous Development/Continuous Delivery). Hierbei sind besondere Sicherheitsüberlegungen notwendig, die nicht im Fokus dieses Bausteins liegen. Die hier betrachteten Container basieren in erster Linie auf den Standards der Open Container Initiative (OCI), die auch den Docker-Standard mit einschließt und Linux-Container ebenso wie Windows-Container umfasst. Gefährdungslage und Anforderungen sind weitgehend unabhängig von der jeweils eingesetzten Container-Technologie, jedoch spezifisch für den hier gebrauchten Containerbegriff.

Container stellen keine Virtualisierungstechnologie im klassischen Sinn dar, sondern isolieren Anwendungen über Kernel-Funktionen. Einige der Gefährdungen und Maßnahmen des Bausteins “Virtualisierung” können jedoch auch analog bei Containern Anwendung finden.

Die Abgrenzung erfolgt an den Schnittstellen der openshift-Plattform. Anforderungen von einzelnen Server-Betriebssystemen (zum Beispiel Windows oder Unix) und direkte Bezüge zu betriebssystemspezifischen Container-Technologien (unter anderem Docker für Windows oder Linux) werden nicht vorgenommen. Die BSI IT-Grundschutz Bausteine APP.3.1 *Webanwendungen*, APP.3.2 *Webserver*, CON.1 *Kryptokonzept*, OPS.1.1.3 *Patch- und Änderungsmanagement*, OPS.1.1.4 *Schutz vor Schadprogrammen*, OPS.1.1.5 *Protokollierung*, OPS.3.2 *Cloud-Anbieter*, ORP.3 *Sensibilisierung und Schulung*, ORP.4 *Identitäts- und Berechtigungsmanagement*, SYS 1.5 *Virtualisierung*, SYS.1.8 *Speicherlösungen/Cloud Storage*, DER.3.3 *Penetrationstests*, DER.4 *Notfallmanagement* und die Bausteine aus der Schicht NET.1: Netze werden im benutzerdefinierten Baustein SYS.bd.2 *Container unter openshift* nicht berücksichtigt.

2 Gefährdungslage

Folgende spezifische Bedrohungen und Schwachstellen sind für den benutzerdefinierten Baustein SYS.bd.2 *Container unter openshift* von besonderer Bedeutung:

2.1 Unangemessene Berechtigungen im Management von Containern

Zur Steuerung von Containern ist die Verwaltung und Vergabe von Berechtigungen für das zentrale Management erforderlich. Werden die Berechtigungen nicht restriktiv vergeben, entstehen Risiken für die Informationssicherheitsschutzziele Vertraulichkeit und Integrität, da Zugriffsrechte der Benutzer nicht auf die Aufgabenerfüllung beschränkt sind und daher unberechtigte Zugriffe erfolgen können.

Gleiches gilt bei einer fehlenden Definition der Zugriffsart für die Zugriffsrechte. Neben möglichen Verfälschungen oder unberechtigten Zugriffen, kann auch die Aufgabenbewältigung eingeschränkt werden, sofern Daten verändert, gelöscht oder unveröffentlicht werden.

2.2 Unzureichender Isolationsgrad bei Containern

Mit Containern wird eine isolierte Umgebung geschaffen. Dabei teilen sich die Container und der Host einen gemeinsamen Kernel. Außerdem besitzen Container eine eigenständige festgelegte Laufzeitumgebung (unter anderem mittels Control Groups und Namespaces) und grenzen sich so voneinander ab.

Je nach Umsetzung der Container-Technologie kann diese unter Umständen nicht für jeden Namespace-Typ eine Isolierung bereitstellen. Wird etwa eine Isolierung für User-Namespaces nicht unterstützt, so eignet sich dieser Container nicht zur Benutzer-ID-Isolation. In diesem Fall sind Benutzerkennungen nicht auf den Container beschränkt. Root-Rechte in einem Container werden demnach auch auf den Host übertragen. Somit können unberechtigte Kernel-Interaktionen nicht ausgeschlossen werden. Dies kann zu weitreichenden Manipulationen am Betriebssystem führen.

2.3 Nicht geregelter Umgang mit Container-Image-Versionen

Die Erstellung und Entfernung von Containern kann prinzipbedingt schnell und flexibel durchgeführt werden. Prozessual können viele verschiedene Container-Images in der Breite angelegt werden. Container-Images können zum Beispiel für den späteren Einsatz vorbereitet oder als Reserven vorgehalten werden. Durch ein ungeregeltes Vorgehen können dabei jedoch schnell Risiken hinsichtlich der Verwaltung von Containern entstehen, sofern durch die Institution kein geregelter Umgang mit Container-Image-Versionen vorgeschrieben wird.

Im Gegensatz zum klassischen Patchen werden bei den Containern sogenannte Image-Templates erstellt und bei einer neuen Version ausgetauscht. Werden diese Templates prozessual nicht regelmäßig (z. B. nach Prüfung neuer verfügbarer Image-Versionen) ausgetauscht, können ungewollte Angriffsflächen entstehen. Eine veraltete oder manipulierte Container-Image-Version kann die Vertraulichkeit, Verfügbarkeit und Integrität der Daten und Informationen beim Umgang mit Containern gefährden.

2.4 Ungesicherte Kommunikation/Unbeschränkter Netz-Traffic

Wird der Netz-Traffic zwischen den Containern auf demselben Host uneingeschränkt gestattet, so wird dadurch das Risiko erhöht, dass unberechtigte Informationen an andere Container weitergeleitet werden. Gerade bei unterschiedlichen Schutzbedarfen von Containern kann eine nicht ausreichend abgesicherte Kommunikation die Integrität und Vertraulichkeit stark beeinträchtigen.

Ohne spezifische Kontrollmechanismen über den Austausch von Netz-Traffic in der Container-Infrastruktur werden keine ausreichenden Beschränkungen zu Containern gewährleistet. Diese ungesicherte Kommunikation ermöglicht unter anderem Cross-Container-Angriffe.

2.5 Nicht Vertrauenswürdige Quellen

Sofern die Institution nicht sicherstellt, dass die Images der Container aus vertrauenswürdigen Quellen stammen (zum Beispiel aus der trusted Community oder vom Hersteller), können gravierende Sicherheitslücken entstehen. Durch unzuverlässige Quellen, deren Herkunft und Korrektheit nicht ausreichend geprüft wurde, können vor allem Integritätsschäden entstehen. Neben Malware und Sicherheitslücken in der Image-Architektur könnten auch unbeabsichtigte Zugriffspfade durch Backdoors entstehen.

2.6 Unzureichende Prüfung auf Schwachstellen

Ohne eine durchgehende Prüfung auf Schwachstellen werden notwendige Aktualisierungen (Updates und Patches) im Rahmen des Patch- und Änderungsmanagements gegebenenfalls nicht rechtzeitig durchgeführt. In Folge dessen können beispielsweise Verzögerungen in der Bereitstellung von benötigten Softwarepaketen für den sicheren Betrieb der Container resultieren. Ein unzureichendes Schwachstellen-Management kann demnach ungewollte Angriffsflächen für unberechtigte Dritte bieten. Insbesondere wenn in den Containern vertrauliche Informationen und Daten verwaltet werden, können weitreichende Schäden für die Institution entstehen.

2.7 Denial-of-Service-Angriffe innerhalb der openshift-Plattform

Denial-of-Service-Angriffe können zur Überlastung der CPU-, Speicher- oder Netz-Komponenten führen. Die Hardware-Ressourcen können so stark beansprucht werden, dass dies schwerwiegende Auswirkungen auf das Gast-/Host-System hat. Zum Beispiel können PRNG-Devices beeinflusst werden, wodurch kryptografische Funktionen geschwächt beziehungsweise aufgebrochen werden können.

2.8 New-Code-Problematik

Container vereinfachen schnelle Release-Folgen und somit häufige Änderungen von Code. Jeder neue Code kann Fehler oder neue Schwachstellen beinhalten. Dadurch können neues Codes Sicherheitslücken eröffnen und zu weitreichenden Bedrohungen für den Betrieb von Containern führen.

2.9 Ausnutzung von CPU-, Kernel- oder Registry-Schwachstellen

Schnittstellen zur CPU, zum Kernel und der Registry sind für den Betrieb von Containern notwendig, daher wirken sich Sicherheitslücken im Kernel und der Registry direkt auf den Betrieb von Containern aus. Ohne wirksame Mechanismen zum Patch- und Änderungsmanagement sowie einer geeigneten Konfiguration und einer Härtung für die IT-Systeme/-Komponenten können diese vorhandenen Schnittstellen ausgenutzt werden. Dadurch werden unberechtigte Zugriffe auf die Informationen und Daten in den Containern ermöglicht. Die Sicherheitslücken wirken sich dabei auf die Verfügbarkeit, Vertraulichkeit und Integrität der Container-Umgebung aus.

3 Anforderungen

Im Folgenden sind die spezifischen Anforderungen des benutzerdefinierten Bausteins SYS.bd.2 *Container unter openshift* aufgeführt. Grundsätzlich ist der IT-Betrieb für die Erfüllung der Anforderungen zuständig. Der Informationssicherheitsbeauftragte (ISB) ist bei strategischen Entscheidungen stets einzubeziehen. Außerdem ist der ISB dafür verantwortlich, dass alle Anforderungen gemäß dem festgelegten Sicherheitskonzept erfüllt und überprüft werden. Zusätzlich kann es noch andere Rollen geben, die weitere Verantwortlichkeiten bei der Umsetzung von Anforderungen haben. Diese sind dann jeweils explizit in eckigen Klammern in der Überschrift der jeweiligen Anforderungen aufgeführt.

Bausteinverantwortlicher	IT-Betrieb
Weitere Verantwortliche	Leiter IT, Fachverantwortliche, Informationssicherheitsbeauftragter

3.1 Basis-Anforderungen

Die folgenden Anforderungen MÜSSEN für den benutzerdefinierten Baustein *SYS.bd.2 Container unter openshift* vorrangig umgesetzt werden:

A 3.1.1 Planung der openshift-Plattform

Der Einsatz beziehungsweise Einsatzzweck von Containern unter dem spezifischen Betriebssystem beziehungsweise in einer Netzumgebung MUSS sorgfältig geplant werden. Die betrieblichen, sicherheitstechnischen, datenschutzrechtlichen und IT-Notfall-Anforderungen an die Verwendung von Containern und die Anforderungen an die geeignete Anwendungsbereitstellung von Software in Containern MÜSSEN vor der Beschaffung geprüft und benannt werden. Die Schnittstellen und Abhängigkeiten zur ganzheitlichen Architektur der openshift-Plattform (Laufzeitumgebung) und der zugrunde liegenden Infrastruktur-Umgebung (Hardware-Infrastruktur) MÜSSEN bei der Planung berücksichtigt werden. Für den geeigneten Betrieb von Containern MÜSSEN sich die verantwortlichen Administratoren und Benutzer mit den Verantwortlichen für die openshift-Plattform abstimmen.

A 3.1.2 Planung der Interaktion von openshift-Plattform und IaaS

Für die erforderliche organisatorische Interaktion von openshift-Plattform und zugrunde liegender

Infrastruktur MÜSSEN erforderliche technischen Bestandteile für die Bereitstellung der Anwendungen erhoben werden. Für die Auswahl MUSS daher zunächst ein Anforderungskatalog erstellt und mit dem Lizenzmanagement der Institution abgestimmt werden. Der Anforderungskatalog MUSS sich an den Empfehlungen des Bausteins CON.4 *Auswahl und Einsatz von Standardsoftware* orientieren.

A 3.1.3 Geeignete Auswahl einer Software für die openshift-Plattform

Die Auswahl einer Software für die openshift-Plattform inkl. Funktionsumfang MUSS unter Berücksichtigung der ermittelten Schutzbedarfe, des geplanten Einsatzzwecks der Container und der Umsetzbarkeit der erforderlichen Absicherungsmaßnahmen erfolgen. Im Rahmen der Softwareauswahl MÜSSEN die IT-Infrastrukturen der Institution und die Softwareentwicklungsprozesse berücksichtigt werden.

Die Auswahlergebnisse (zum Beispiel gewählte Lizenzmodelle oder Versionen) MÜSSEN als Erweiterungen in den etablierten Beschaffungsprozess übertragen werden.

A 3.1.4 Container-Image-Sicherheit

Die Sicherheit der Images für die Container MUSS durch geeignete Sicherheitsmechanismen der Institution zum allgemeinen Umgang mit Images gewährleistet werden. Zusätzlich MÜSSEN Vorgaben und Regelungen zur Verwaltung der Container-Image-Templates etabliert werden. Der notwendige Austausch der Image-Templates im Rahmen neuer Container-Image-Versionen MUSS im Patch- und Änderungsmanagement der Institution berücksichtigt sein.

Die potenziellen Image-Risiken MÜSSEN durch ein angemessenes Monitoring mit zentraler Berichterstattung und unter Einsatz von Schwachstellen-Scannern minimiert werden. Zu diesem Zweck MÜSSEN die eingesetzten Container-Technologien in das Schwachstellen-Management der Institution eingebunden werden. Die Schwachstellen-Management-Prozesse, die für den Betrieb von Containern erforderlich sind, MÜSSEN durch ein angemessenes Tool unterstützt werden. Die Images MÜSSEN nach Möglichkeit gescannt werden, ohne das Image dabei auszuführen.

A 3.1.5 Sichere Installation der openshift-Plattform und zugehörigen Infrastruktur

Bei der Installation der openshift-Plattform-Software MUSS der allgemeine Softwarebereitstellungs- und Installationsprozess der Institution eingehalten werden. Für die Installation der openshift-Plattform MÜSSEN die spezifischen Container-Basistechnologien und das Container-Management für den Anwendungsfall von abgestimmt sein und in einer Installationsdokumentation beschrieben werden.

Anhand der Installationsdokumentation MUSS die openshift-Plattform durch die zuständigen Administratoren eingerichtet werden.

A 3.1.6 Sichere Administration der openshift-Plattform

Für die geeignete Administration der Container MÜSSEN die administrativen Aufgaben zur sicheren Einrichtung der openshift-Plattform, zur Einstellung der Kernel Security und zu Softwaretests vor produktiver Inbetriebnahme der Softwarelösung (bei erstmaliger Verwendung) erfüllt werden.

Berücksichtigt werden MÜSSEN die Administrationsaspekte der administrativen Trennung in den Containern, Root-Zugriffe, Dateirechte von laufenden Containern, Festlegungen der Container hinsichtlich Ressourcenlimits des Hosts, Content-Trust-Funktionen und Überprüfungen der Konfigurationen. Die Administration der openshift-Plattform DARF NICHT über deren Managementschnittstellen erfolgen, sondern MUSS über eine dedizierte Managementschnittstelle realisiert werden.

Für notwendige lokale Administrationskonten MÜSSEN einzigartige, sichere Passwörter verwendet werden. Die für die openshift-Plattform zuständigen Administratoren MÜSSEN regelmäßig in den sicherheitsrelevanten Aspekten der openshift-Plattform geschult werden. Die Verwendung von administrativen Rechten durch die verantwortlichen Administratoren MUSS geregelt werden.

A 3.1.7 Sichere Inbetriebnahme der openshift-Plattform

Vor Inbetriebnahme MUSS eine grundlegende Systemhärtung der Komponenten der openshift-Plattform

umgesetzt werden. Für die Systemhärtung MÜSSEN funktionsspezifische Sicherheitsvorlagen erstellt, gepflegt und angewendet werden. Müssen auf den Komponenten der produktiven openshift-Plattform im Rahmen des Patch- und Änderungsmanagements Servicepacks oder Enhancement-Packs installiert werden, MUSS zuvor geprüft werden, ob die bisherigen Härtungsmaßnahmen weiterhin ausreichen oder gegebenenfalls eine Anpassung notwendig ist.

Darüber hinaus MÜSSEN für die Konfiguration der openshift-Plattform die Anforderungen der Maßnahmen Sichere Konfiguration virtueller IT-Systeme und Sichere Konfiguration eines Netzes für virtuelle Infrastrukturen aus dem BSI IT-Grundschutz-Baustein SYS.1.5 *Virtualisierung* berücksichtigt werden.

A 3.1.8 Geregelte Löschung der openshift-Plattform und Container

Vor der Löschung von Containern MÜSSEN die Löschparameter festgelegt werden. Sofern die Daten in den Containern weiterhin benötigt werden, MÜSSEN vor der Löschung Datensicherungen der relevanten Inhalte durchgeführt werden. Es MÜSSEN durch die Institution Sicherheitsvorgaben erstellt werden, das eine Löschung nur bei gestoppten Containern erfolgt. Für die Löschung MÜSSEN verschiedene Container-Löschmöglichkeiten zur Verfügung stehen.

Die Löschung der Software für die openshift-Plattform durch die Administratoren MUSS gemäß den allgemeinen Regelungen zum Umgang mit eingesetzter Software erfolgen.

A 3.1.9 Regelmäßige Datensicherung bei der openshift-Plattform

Alle aus Betriebssicht kritischen Komponenten der openshift-Plattform MÜSSEN in das etablierte Datensicherungskonzept aufgenommen werden.

3.2 Standard-Anforderungen

Gemeinsam mit den Basis-Anforderungen entsprechen die folgenden Anforderungen dem Stand der Technik für den benutzerdefinierten Baustein *SYS.bd.2 Container unter openshift*. Sie SOLLTEN grundsätzlich umgesetzt werden.

A 3.2.1 Bereitstellung von gehärteten Basis-Images

Die Layering-Struktur von Images SOLLTE dazu genutzt werden, um ein gehärtetes Basis-Image bereitzustellen, in dass die weiteren Anwendungskomponenten ergänzend als weitere Layer hinzugefügt werden.

Die Integrität von Images und insbesondere des Basis-Images SOLLTE durch die Verwendung von Signaturen sichergestellt und vor Inbetriebnahme von Containern geprüft werden.

A 3.2.2 Geeignete Auswahl eines externen Hosting-Anbieters

Für den Betrieb der openshift-Plattform SOLLTEN die Container und die Programmierschnittstelle (API) intern oder extern gehostet werden. Beim externen Hosting SOLLTEN intern technische, organisatorische, regulatorische und dokumentarische Anforderungen zum externen Hosting definiert werden. Die Auswahl eines externen Hosting-Anbieters SOLLTE anhand eines darauf basierenden Anforderungskatalogs erfolgen. Der Anforderungskatalog für die openshift-Plattform SOLLTE maßgeblich für die Hosting-Entscheidung sein.

In Fällen, in denen keine technische Verifizierung durch Trusted Platform Module vorgesehen ist, SOLLTE die Institution die Hardware-Vertrauensvoraussetzungen als Teil ihrer Service Vereinbarungen mit dem Dienstleister festlegen.

A 3.2.3 Absicherungsmaßnahmen des Host-Betriebssystems

Bei Verwendung einer openshift-Plattform SOLLTEN möglichst containerspezialisierte minimalistische Betriebssysteme eingesetzt werden. Die dafür vorzusehenden Härtungsparameter SOLLTEN festgelegt und umgesetzt werden. Es SOLLTEN ausschließlich schreibgeschützte Dateisysteme verwendet werden. Container- und Nicht-Container-Workloads SOLLTEN NICHT auf derselben Host-Instanz betrieben werden. Das Dateisystem SOLLTE nur-lesend (read-only) eingehängt sein.

Das produktive Host-Betriebssystem und die zugehörigen Komponenten SOLLTEN gemäß des Patch- und Änderungsmanagements der Institution aktualisiert werden. Es SOLLTE eine Validierung der Versionierung von Basis-Betriebssystem-Komponenten realisiert werden.

Alle Anwendungskomponenten inklusive deren Schnittstellen SOLLTEN in Containern gebündelt bereitgestellt werden.

Alle Dateiänderungen, die Container auf der Festplatte persistieren müssen, SOLLTEN innerhalb von speziell für diesen Zweck zugewiesenen Speichermedien vorgenommen werden. Die Speichermedien SOLLTEN geeignet isoliert werden. Auf keinen Fall SOLLTEN Container in der Lage sein, empfindliche Verzeichnisse auf dem Dateisystem eines Hosts zu mounten.

A 3.2.4 Aufteilung der Administrationstätigkeiten der openshift-Plattform

Die Administrationstätigkeiten der openshift-Plattform SOLLTEN, sofern mehrere Administratoren an der Plattform und den Containern arbeiten, aufgeteilt werden.

Grade in großen Institutionen SOLLTE die Administration der openshift-Plattform auf mehrere Personen verteilt werden, um so eine Funktionstrennung herbeizuführen. Die Aufteilung der Administrationstätigkeiten SOLLTE die etablierten Stellvertreterregelungen berücksichtigen.

A 3.2.5 Berechtigungsmanagement für Container

Das Berechtigungsmanagement für die Container SOLLTE auf dem allgemeinen Identitäts- und Berechtigungsmanagement der Institution aufbauen und die spezifischen Mandaten-Anforderungen der openshift-Plattform berücksichtigen.

Die Anwendungsbereitstellungsprozesse finden mit der openshift-Plattform häufig verteilt statt (auf verschiedenen IT-Systemen und an unterschiedlichen Standorten), daher SOLLTE sichergestellt werden, wo und wie die Container bearbeitet werden dürfen. Die Rechtevergabe für die Verwendung und Verwaltung von Containern SOLLTE restriktiv gehandhabt werden. Die Mandanten-Zugriffe auf Container SOLLTEN regelmäßig kontrolliert werden.

Die Dateisystemberechtigungen bei Containern SOLLTEN auf ein Minimum beschränkt werden.

Alle Authentifizierungen für das Host-Betriebssystem der openshift-Plattform SOLLTEN innerhalb der Protokollierung überwacht werden.

A 3.2.6 Entwicklung eines Testplans für die openshift-Plattform

Für die eingesetzte Lösung zur Umsetzung der openshift-Plattform SOLLTE ein Testplan für das Testen sowohl der Software der openshift-Plattform als auch der zu betreibenden Container auf Basis des Anforderungskataloges erstellt werden. Die Testumgebung SOLLTE gemäß Vorgaben für das Testen von Standardsoftware logisch oder physisch von der Produktivumgebung getrennt sein. Die durchgeführten Tests SOLLTEN nachvollziehbar, reproduzierbar und vollständig dokumentiert werden. Die Testdokumentation SOLLTE dabei aus Testplänen, -zielen, -verfahren und -ergebnissen bestehen und die Übereinstimmung zwischen den Tests und den spezifizierten Anforderungen aufzeigen.

A 3.2.7 Umsetzung des Testplans für die openshift-Plattform

In der Testvorbereitung für die openshift-Plattform SOLLTEN die Testmethoden für die Einzeltests mit Testarten, -verfahren und -werkzeugen festgelegt werden. Die Durchführung der Tests SOLLTE anhand der definierten Testpläne gemäß A 3.2.6 *Entwicklung eines Testplans für die openshift-Plattform* erfolgen. Im Testumfang SOLLTEN Standard-, Fehler- und Ausnahmefälle berücksichtigt werden. Werden in den Tests Echtdaten verwendet beziehungsweise verarbeitet, SOLLTEN diese für die Tests anonymisiert werden. Die Installation und Konfiguration der Testumgebung SOLLTE dokumentiert werden. Die Ergebnisse der durchgeführten Tests SOLLTEN dokumentiert und anhand definierter Entscheidungskriterien bewertet werden.

A 3.2.9 Sicherheit der Registry in openshift-Plattform

Jeder Zugriff auf eine Registry, die proprietäre und vertrauliche Images enthält, SOLLTE einem geeigneten

Authentifizierungsschema unterliegen. Die Vergabe der Leserechte, Schreibrechte und weiterführenden Rechte SOLLTE den allgemeinen Anforderungen des Identitäts- und Berechtigungsmanagements der Institution unterliegen. Account-Sicherheitsmechanismen durch Kopplungen mit Verzeichnisdiensten SOLLTEN aktiviert und genutzt werden (z. B. Sicherheitskontrollen).

Die Kanäle beziehungsweise Verbindungen von Entwicklungswerkzeugen, Orchestrators und Containern zur Registry SOLLTEN verschlüsselt werden.

Es SOLLTE darauf geachtet werden, dass alte Images in der Registry eindeutig gekennzeichnet werden. Eine Aktualisierung der in der Registry hinterlegten Images SOLLTE regelmäßig erfolgen.

A 3.2.10 Orchestrator-Absicherung

Für die Orchestratoren SOLLTE ein privilegiertes Zugangsrechte-Modell eingerichtet werden, um unbegrenzte administrative Zugänge zu vermeiden. Diese SOLLTEN auf den erforderlichen Funktionsumfang begrenzt werden, der auf erforderlichen Aktionen für Hosts, Containers und Images beruht. Der Zugriff auf clusterweite administrative Accounts SOLLTE überwacht werden.

Die Segmentierung nach Zweck, Vertraulichkeit und Bedrohungslage von Containern, die durch Orchestratoren bereitgestellt werden, SOLLTE in geeigneten Sets erfolgen.

Orchestrator-Plattformen SOLLTEN so konfiguriert werden, dass alle erforderlichen Anwendungen in einer sicheren Umgebung betrieben werden. Nodes SOLLTEN gemäß den vorliegenden Schutzbedarfen in die vorhanden Cluster integriert werden. Die Nodes sind zu inventarisieren und der Status der Nodes-Konnektivität SOLLTE überwacht werden. Der Umgang mit kompromittierten Nodes ist zu regeln. Dabei ist vor allem auf die Isolation von Nodes einzugehen. Die Institution SOLLTE möglichst Orchester auswählen, die gegenseitig authentifizierte Netzwerkverbindungen zwischen Clustermitgliedern ermöglichen und Ende-zu-Ende-Verschlüsselung von Intra-Cluster-Verkehr bieten.

A 3.2.11 Sicherheit von Containern

Die Institution SOLLTE geeignete technische Vorgaben zur Container-Laufzeit (runtime) dokumentieren.

Bei einem erhöhten Schutzbedarf SOLLTEN zusätzliche Tools mit spezifischem Lernverhalten und Sicherheitsprofilen für automatisch veröffentlichte Container-Anwendungen genutzt werden. Die Sicherheitsprofile SOLLTEN für die Detektion von Laufzeit-Problemen geeignet sein.

Die Container-Laufzeiten SOLLTEN im gesamten IT-Betrieb überwacht werden. Risiken durch eine anfällige Laufzeit der Container und Hosts SOLLTEN durch Laufzeitüberwachungen minimiert werden. Der Egress-Netzwerkverkehr SOLLTE mindestens an den Netzgrenzen kontrolliert werden, um einen unbegrenzten Netzzugang aus Containern zu unterbinden. Der Traffic zwischen Netzen unterschiedlicher Vertraulichkeitsstufen SOLLTE vermieden werden. Eingesetzte App-aware Tools SOLLTEN sowohl den Inter-Container-Traffic einsehen können als auch dynamische Filterregeln für Container-Applikationen verwenden.

A 3.2.12 Sicherer Betrieb von Containern

Beim Betrieb einer Containerlösung SOLLTEN die Verantwortlichen der Institution sicherstellen, dass der innerhalb des Containers verwendete Code aus einer bekannten und vertrauenswürdigen Quelle stammt.

Um potenzielle Schwachstellen beim Betrieb von Container-Lösungen erkennen zu können, SOLLTEN in regelmäßigen Abständen Tests mithilfe von Schwachstellen-Analyse-Tools stattfinden. Nach einem Versionswechsel oder Versionsupdate SOLLTE, unabhängig vom Testturnus, eine Schwachstellen-Analyse erfolgen. Die Ergebnisse der Schwachstellen-Analyse SOLLTEN dokumentiert und archiviert werden.

A 3.2.13 Trusted Computing

Alle Firmware-, Software- und Konfigurationsdaten SOLLTEN hinsichtlich ihrer Vertraulichkeit beurteilt werden. Die Ausführung dieser Daten SOLLTE mit einem Root of Trust for Measurement (RTM) erfolgen.

Alle Beurteilungen SOLLTEN in einer Hardware-Root des Vertrauens gespeichert werden [z. B. Trusted Platform Module (TPM)]. Das Trusted-Computing-Modell SOLLTE angemessene und sichere Bootvorgänge

gewährleisten und eine Bereitstellung verifizierter Systemplattformen ermöglichen.

A 3.2.14 Überwachung der Funktionen und Konfigurationen der openshift-Plattform

Die Funktionen und Konfigurationen der openshift-Plattform SOLLTEN geeignet überwacht werden. Die Überwachung SOLLTE durch ein zentrales Monitoring der Institution erfolgen. Die Benutzer der openshift-Plattform SOLLTEN zusätzlich alle ausgemachten Sicherheitslücken oder aufgetretene Sicherheitsvorfälle an die zuständige Stelle in der Intuition senden. Die interne Revision SOLLTE regelmäßige Überprüfungen der openshift-Plattform durchführen.

A 3.2.15 Sicherer Umgang mit Container-Snapshots

Für den sicheren Umgang mit Container-Snapshots SOLLTEN Vorgaben durch die verantwortlichen Administratoren in Abstimmung mit dem Informationssicherheitsbeauftragten festgelegt werden. Bei der Verwendung von Container-Snapshots SOLLTE darauf geachtet werden, dass die Snapshots keine vollständige Datensicherungslösung darstellen und die Snapshot-Dateien nur eine Aufzeichnung von allen Änderungen im Vergleich zum originalen Container abbilden. Die Container-Snapshot-Größe SOLLTE durch die Verwendungsdauer von Snapshots begrenzt werden. Dabei SOLLTEN Aspekte der Löschung und Konsolidierung berücksichtigt werden. Nach der Erzeugung eines neuen Snapshots SOLLTE dessen Funktionsfähigkeit überprüft werden. Ist der Snapshot funktionsfähig, SOLLTEN ältere Snapshots gelöscht werden.

Sind für die Snapshots Drittanbieter-Produkte im Einsatz, SOLLTEN die Systeme, die für die Datensicherung konfiguriert wurden, regelmäßig hinsichtlich der Aufbewahrungszeiten überprüft werden. Zusätzlich SOLLTE dann (sofern möglich) regelmäßig über die Command-Line geprüft werden, dass keine Snapshots nach dem Löschen mehr vorhanden sind.

Für die Snapshot-Überwachungen können in den meisten Snapshot-Anwendungen Alarmierungsfunktionen zur Information des zuständigen Administrators bei automatischer Durchführung von Snapshots eingestellt werden.

A 3.2.16 Geeignete Anwendungsbereitstellung als Container (Deploy)

Für die Anwendungsbereitstellung SOLLTEN geeignete Mechanismen durch die zuständigen Administratoren in Abstimmung mit dem Informationssicherheitsbeauftragten der Institution festgelegt werden.

Deploy-Mechanismen und Skripte SOLLTEN nicht während des Prozesses zur Anwendungsbereitstellung gewechselt werden.

A 3.2.17 Eingebettete Geheimnisse (Embedded secrets)

Alle Geheimnisse SOLLTEN außerhalb vom Image gespeichert werden und dynamisch innerhalb der Laufzeit zur Verfügung stehen. Die Institution SOLLTE gewährleisten, dass Geheimnisse nur den relevanten Containern zur Verfügung stehen.

3.3 Anforderungen bei erhöhtem Schutzbedarf

Im Folgenden sind für den benutzerdefinierten Baustein *SYS.bd.2 Container unter openshift* exemplarische Vorschläge für Anforderungen aufgeführt, die über das dem Stand der Technik entsprechende Schutzniveau hinausgehen und BEI ERHÖHTEM SCHUTZBEDARF in Betracht gezogen werden SOLLTEN. Die konkrete Festlegung erfolgt im Rahmen einer Risikoanalyse. Die jeweils in Klammern angegebenen Buchstaben zeigen an, welche Grundwerte durch die Anforderung vorrangig geschützt werden (C = Vertraulichkeit, I = Integrität, A = Verfügbarkeit).

A 3.3.1 Redundanz und Hochverfügbarkeit der openshift-Plattform (A)

Alle aus Betriebssicht kritischen Komponenten der openshift-Plattform SOLLTEN in das etablierte Datensicherungskonzept aufgenommen werden.

4 Weiterführende Informationen

Weiterführende Informationen zu Gefährdungen und Sicherheitsmaßnahmen für den benutzerdefinierten Baustein *SYS.bd.2 Container unter openshift* finden sich unter anderem in folgenden Veröffentlichungen:

NIST800190	Application Container Security Guide, Special Publication 800-190, NIST, 09.2017 https://doi.org/10.6028/NIST.SP.800-190
OCP37	Red Hat OpenShift Container Platform Documentation, Red Hat Inc., 01.2018 https://access.redhat.com/documentation/en-us/openshift_container_platform/3.7/?version=3.7
CISRHEL220	Red Hat Enterprise Linux 7 Benchmark version 2.2.0, CIS https://www.cisecurity.org/benchmark/red_hat_linux/
CISK8S120	CIS Kubernetes Benchmark Version 1.2.0, CIS https://www.cisecurity.org/benchmark/red_hat_linux/
CISDOCKER110	CIS Docker Community Edition Benchmark version 1.1.0, CIS https://www.cisecurity.org/benchmark/red_hat_linux/

5 Anlage: Kreuzreferenztabelle zu elementaren Gefährdungen

Die folgenden elementaren Gefährdungen sind für den benutzerdefinierten Baustein S*SYS.bd.2 Container unter openshift* von Bedeutung:

- G 0.15 Abhören
- G 0.18 Fehlplanung oder fehlende Anpassung
- G 0.19 Offenlegung schützenswerter Informationen
- G 0.20 Informationen oder Produkte aus unzuverlässiger Quelle
- G 0.21 Manipulation von Hard- und Software
- G 0.22 Manipulation von Informationen
- G 0.23 Unbefugtes Eindringen in IT-Systeme
- G 0.25 Ausfall von Geräten oder Systemen
- G 0.26 Fehlfunktion von Geräten oder Systemen
- G 0.27 Ressourcenmangel
- G 0.28 Software-Schwachstellen oder –Fehler
- G 0.29 Verstoß gegen Gesetze oder Regelungen
- G 0.30 Unberechtigte Nutzung oder Administration von Geräten und Systemen
- G 0.31 Fehlerhafte Nutzung oder Administration von Geräten und Systemen
- G 0.32 Missbrauch von Berechtigungen
- G 0.36 Identitätsdiebstahl
- G 0.37 Abstreiten von Handlungen
- G 0.39 Schadprogramme
- G 0.40 Verhinderung von Diensten (Denial of Service)
- G 0.43 Einspielen von Nachrichten
- G 0.45 Datenverlust
- G 0.46 Integritätsverlust schützenswerter Informationen
- G 0.47 Schädliche Seiteneffekte IT-gestützter Angriffe

Elementare Gefährdungen / Anforderungen	G 0.15	G 0.18	G 0.19	G 0.20	G 0.21	G 0.22	G 0.23	G 0.25	G 0.26	G 0.27	G 0.28	G 0.29	G 0.30	G 0.31	G 0.32	G 0.36	G 0.37	G 0.39	G 0.40	G 0.43	G 0.45	G 0.46	G 0.47
A.3.1.1		X					X										X						
A.3.1.2		X		X			X				X	X		X				X					X
A.3.1.3				X			X				X	X		X				X					X
A.3.1.4		X			X	X	X	X	X										X				
A.3.1.5		X												X									
A.3.1.6					X	X	X						X	X	X		X			X		X	
A.3.1.7	X		X		X	X	X						X		X			X		X		X	
A.3.1.8		X	X		X	X	X								X								
A.3.1.9		X						X	X	X											X		
A.3.2.1			X		X	X	X						X		X			X				X	
A.3.2.2			X				X						X	X	X								
A.3.2.3			X		X	X	X						X		X			X		X		X	
A.3.2.4					X	X	X		X				X		X		X			X		X	
A.3.2.5					X	X	X		X						X		X			X		X	
A.3.2.6		X		X				X	X		X												
A.3.2.7		X		X				X	X		X												
A.3.2.8					X	X	X						X		X	X	X			X			
A.3.2.9	X				X	X	X						X	X	X	X	X					X	
A.3.2.10		X			X	X	X	X	X	X					X				X				
A.3.2.11				X	X	X	X				X				X			X		X			
A.3.2.12					X	X	X					X			X			X				X	X
A.3.2.13					X	X	X	X	X	X					X				X	X			
A.3.2.14					X	X	X	X	X	X		X									X		
A.3.2.15		X								X							X					X	
A.3.2.16			X		X	X						X				X						X	
A.3.2.17			X			X						X										X	
A.3.3.1								X	X	X											X		