IT-Grundschutz für die Container-Virtualisierung mit dem neuen BSI-Baustein SYS. 1.6

Christoph Haar[1], Erik Buchmann[2]

Abstract:

Mit Hilfe der Container-Virtualisierung lassen sich Anwendungen flexibel in die Cloud auslagern, administrieren, zwischen Rechenzentren migrieren, etc. Dafür baut die Containervirtualisierung auf eine komplexe IT-Landschaft auf, in der Hardware, Betriebssystem und Anwendungen von verschiedenen Parteien bereitgestellt und genutzt werden. Der IT-Sicherheit kommt daher eine große Bedeutung zu. Das Bundesamt für Sicherheit in der Informationstechnik (BSI) stellt mit dem IT-Grundschutz eine Methode zur Umsetzung von angemessenen Schutzmaßnahmen im IT-Umfeld zur Verfügung. Es gibt jedoch wenig Erfahrung mit der Absicherung der Container-Virtualisierung gemäß IT-Grundschutz: Das Grundschutz-Kompendium und die Standards zur Risikoanalyse wurden erst im November 2017 in überarbeiteter Form neu eingeführt, und der Baustein SYS. 1.6 zur Container-Virtualisierung wurde erst im Mai 2018 als Community Draft veröffentlicht.

In dieser Arbeit untersuchen wir, wie gut sich der aktuelle IT-Grundschutz auf einen Web-Shop anwenden lässt, der mittels Docker virtualisiert wurde. Wir gehen dabei insbesondere auf die Gefährdungsanalyse, Docker-spezifische Gefährdungen sowie entsprechende Maßnahmen zur Abwendung dieser Gefährdungen ein. Darüber hinaus diskutieren wir, wie sich unsere Erkenntnisse über das Docker-Szenario hinaus auf Container-Technologie verallgemeinern lassen. Wir haben festgestellt, dass der Baustein SYS. 1.6 das Grundschutz-Kompendiums eine umfassende Hilfestellung zur Absicherung von Containern bietet. Wir haben jedoch zwei zusätzliche Gefährdungen identifiziert.

Keywords: IT-Grundschutz; IT-Sicherheit; Container-Virtualisierung; Docker Container.

1 Einleitung

Die Container-Virtualisierung ermöglicht es, innovative und cloudbasierte Anwendungen auf eine agile, kosteneffiziente Weise umzusetzen, auszuliefern und zu warten. Ein prominentes Beispiel ist das Open-Source Projekt Docker [Dob]. Durch die Nutzung von Containern wird es möglich, die eigentliche Anwendung von der IT-Infrastruktur zu trennen. So wird es beispielsweise möglich, aus einer öffentlichen Registry ein Image mit einem Betriebssystem zu holen, dieses zusammen mit einer Anwendung in einen Container zu packen und diesen Container dann in einer Testumgebung zu prüfen. Der selbe Container lässt sich dann in das Produktivsystem übertragen und beispielsweise bei gewachsenem Ressourcenbedarf auf

[1] Hochschule für Telekommunikation Leipzig, haar@hft-leipzig.de
[2] Hochschule für Telekommunikation Leipzig, buchmann@hft-leipzig.de

einen größeren Host-Rechner umziehen. Dafür verwendet die Container-Virtualisierung eine komplexe IT-Landschaft, in der verschiedene Parteien Softwarekomponenten oder Hardwareressourcen zur Verfügung stellen, Container bereitstellen oder die Virtualisierungsumgebung betreiben. Da die Container sensible Firmendaten oder personenbezogene Informationen enthalten können, kommt der IT-Sicherheit dabei eine wesentliche Rolle zu. Unternehmen müssen daher bei der Anpassung ihrer IT-Infrastruktur an die Container-Virtualisierung [Gö17] ihr IT-Sicherheitskonzept überarbeiten.

Wenn das Sicherheitskonzept eines Unternehmens auf dem IT-Grundschutz [Bu11] des Bundesamts für Sicherheit in der Informationstechnik (BSI) beruhen soll oder im Rahmen einer ISO 27001 Zertifizierung auf dem IT-Grundschutz aufgebaut [Bu14] wird, war dies bisher schwierig. Sowohl der Baustein B 3.304 Virtualisierung aus dem alten BSI-Grundschutz-Katalog [Bu16] als auch der korrespondierende Baustein SYS. 1.5 des aktuellen BSI Grundschutz-Kompendiums [Bu17c] zielen eher auf eine Hypervisor-Visualisierungsschicht ab. Im Mai 2018 wurde ein Community Draft für einen neuen Baustein SYS. 1.6 „Container“ [Bu] für die Container-Virtualisierung mittels Docker oder alternativer Technologien veröffentlicht. Dieser Baustein hat jedoch einen vorläufigen Charakter. Es gibt daher keine Erfahrungen, ob die darin beschriebenen Gefährdungen und Maßnahmen in der Praxis ausreichen, um ein gegebenes Anwendungsszenario ausreichend abzusichern. Um diese Frage zu klären, haben wir ein typisches Szenario zugrunde gelegt:

Szenario: *Ein Einzelhändler verwendet einen Web-Shop, um sein Ladengeschäft zu ergänzen. Der Web-Shop verfügt über eine eigene Datenbank mit Produktbeschreibungen und Kundenkonten. Darüber hinaus ist der Web-Shop mit einem Internet-Zahlungssystem ausgestattet, das über einen Dienstleister Bezahlvorgänge über unterschiedliche Kanäle sicher abwickelt. Anwendung, Datenbank und Zahlungssystem sind auf verschiedene Docker-Container aufgeteilt. Die Container werden auf einem eigenen Rechner in einer On-Premise-Umgebung ausgeführt, bei der der Einzelhändler nicht nur für die Container verantwortlich ist, sondern auch die Infrastruktur und die Container-Plattform betreibt.* □

Für dieses Szenario haben wir eine Absicherung nach dem aktuellen IT-Grundschutz durchgeführt. Dabei haben wir uns auf die Container-Virtualisierung konzentriert. Das heißt, wir haben für die bereits sehr gut untersuchte Absicherung [Dä18; Ec13] der Infrastruktur sowie der organisationsübergreifenden Aspekte ein bereits bestehendes Sicherheitskonzept nach IT-Grundschutz als gegeben vorausgesetzt.

Aufbauend auf [Ba17] und [BHB18] haben wir nach BSI-Standard 200-2 [Bu17a]den Informationsverbund für unser Docker-System modelliert, dafür eine Schutzbedarfsfeststellung durchgeführt, und die in den BSI-Bausteinen SYS. 1.5 Virtualisierung und SYS. 1.6 Container beschriebenen Elementargefährdungen analysiert. Da einige Daten den Schutzbedarf „hoch“ erfordern, haben wir in einem zweiten Schritt eine Risikoanalyse nach BSI-Standard 200-3 [Bu17b] zur Identifikation und Behandlung von zusätzlichen Gefährdungen für unser Docker-Szenario durchgeführt. Im Anschluss haben wir analysiert, inwiefern sich die von uns identifizierten zusätzlichen Gefährdungen und Maßnahmen von denen des

IT-Grundschutz-Kompendiums in den Bausteinen SYS. 1.5 und SYS. 1.6 unterscheiden. Die zentrale Erkenntnis unserer Arbeit ist, dass das BSI mit dem neuen Container-Baustein SYS. 1.6 das BSI-Grundschutz-Kompendium um einen wertvollen Baustein zur Absicherung von Containern erweitert. Auf ein konkretes Szenario angewendet, ergeben sich jedoch zusätzliche Gefährdungen, sodass eine Erweiterung des Bausteins zu überlegen ist.

Aufbau der Arbeit: Abschnitt 2 beschreibt die Grundlagen dieser Arbeit. In Abschnitt 3 und 4 wird wir eine Risikoanalyse für Docker nach BSI-Standard durch und vergleichen unsere Erkenntnisse mit denen des BSI. In Abschnitt 5 verallgemeinern wir unsere Erkenntnisse. Die Arbeit schließt mit einer Zusammenfassung in Abschnitt 6.

2 Grundlagen

In diesem Abschnitt stellen wir das Docker-System [Pe15], die BSI-Bausteine SYS. 1.5 und SYS. 1.6 [Bu] sowie die Vorgehensweisen zur Standardabsicherung und Risikoanalyse nach den aktuellen BSI-Standards [Bu17a; Bu17b] vor.

2.1 Docker-Container

Die Container-Virtualisierung hat sich aus der Hypervisor-Virtualisierung [Ch07] entwickelt. Ein Hypervisor zieht eine Abstraktionsschicht zwischen Host-System und den darauf ablaufenden Gast-Systemen ein, eine virtuelle Hardware bereitstellt. Dies hat unter anderem den Nachteil, dass für jeden Gast ein vollständiges Betriebssystem aufzusetzen ist. Im Gegensatz dazu werden bei der leichtgewichtigen Container-Virtualisierung Container zusammengestellt, die nur die Anwendung und eine leichtgewichtige Ablaufumgebung enthalten. Die Container nutzen also den Kernel des Host-Betriebssystems mit. Dies ermöglicht es, Systemressourcen wie Prozessor, Netzwerk oder Speicher effizient zu nutzen, und Applikationen über Systeme hinweg zu verschieben, ohne dabei komplette Betriebssysteme mit zu migrieren. Auf der anderen Seite wird es jedoch schwerer, mehrere Container, die auf dem selben Host-System ablaufen, zuverlässig voneinander zu isolieren.

Eine sehr häufig eingesetzte Lösung für die Container-Virtualisierung ist das auf einem Linux-Betriebssystem aufsetzende Docker. Linux-typisch besteht die Docker Architektur [Dob] aus Docker Client, Docker Daemon, Docker Registry und den Docker Objekten (Images, Docker Files, Container). Der Docker Client und Docker Daemon bilden zusammen die Docker Engine. Ein Container enthält zwei Hauptverzeichnisse: /bin enthält die Binärdateien und /lib die dynamischen Bibliotheken und Kernel-Module, die für die Funktionalität eines Containers benötigt werden. Client und Daemon können auf dem gleichen Host-System laufen oder der Client wird mit einem Remote Daemon verbunden. Die externe Kommunikation findet über eine REST API, ein UNIX Socket oder eine andere Netzwerkschnittstelle statt. Docker ist in seinen Grundeinstellungen so konfiguriert, dass

nach Images aus dem Docker Hub gesucht wird. Es ist auch möglich, eine private Registry für Images anzulegen (Docker Trusted Registry).

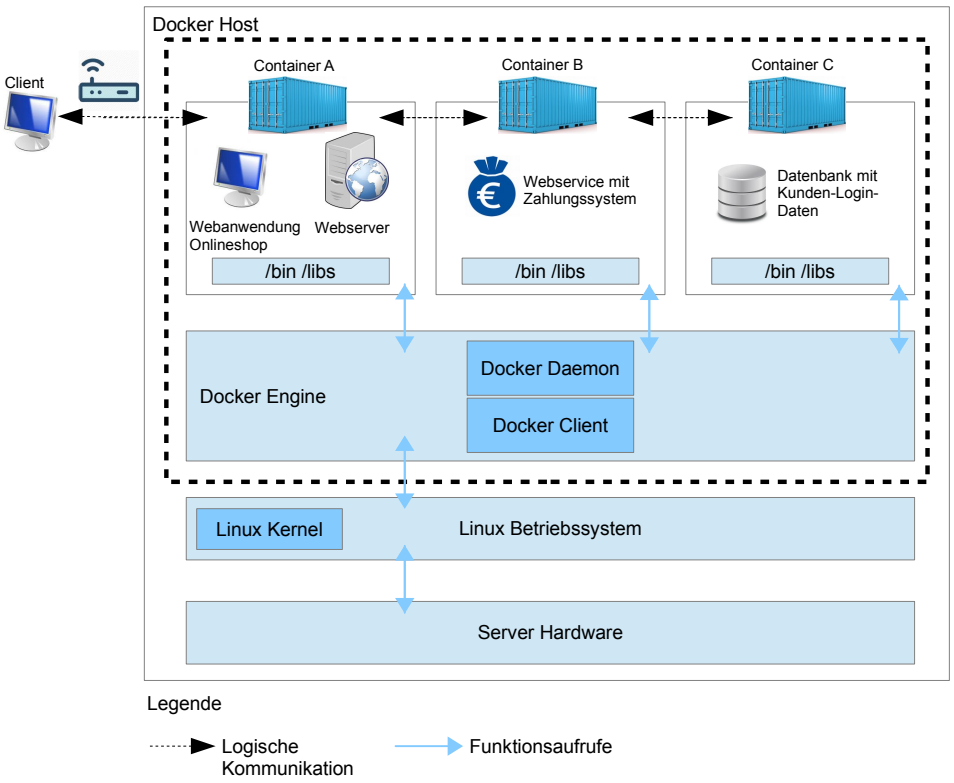

Abb. 1: Beispiel-System mit Dockerarchitektur

Abbildung 1 beschreibt eine typische Docker-Installation, wie wir sie auch für unser Anwendungsszenario zugrunde gelegt haben: Auf dem Linux-Kernel setzt die Docker Engine auf. Für jede sachlogisch voneinander getrennte Aufgabe wird ein eigener Container betrieben. In unserem Fall sind dies drei Container, die voneinander isoliert Datenbank, Zahlungssystem und Web-Anwendung für den Online-Shop bereitstellen. Die Container kommunizieren über Linux-übliche Netzwerkschnittstellen miteinander und mit dem Internet (schwarze Pfeile). Zu diesem Zweck nutzen sie Funktionen der Docker Engine (graue Pfeile). Im Folgenden konzentrieren wir uns auf die Absicherung des in der Abbildung gestrichelt dargestellten Bereichs nach dem Ende 2017 überarbeiteten IT-Grundschutz. Eine Risikoanalyse nach dem alten IT-Grundschutz ist Teil unserer Vorarbeiten [BHB18].

2.2 Standard-Absicherung und Risikoanalyse nach BSI

Der BSI-Standard 200-2 beschreibt, in welchen Schritten eine Standard-Absicherung eines Systems durchzuführen ist [Bu17a].

1. Zunächst ist der **Geltungsbereich** („Informationsverbund“) festzulegen, für den das Sicherheitskonzept realisiert werden soll. Dies kann eine technische Infrastruktur oder eine Organisationseinheit innerhalb eines Unternehmens sein. Der Geltungsbereich für unser Beispiel wird in Abb. 1 dargestellt.
2. Voraussetzung für die Anwendung des IT-Grundschutz-Kompendiums ist die Durchführung einer **Strukturanalyse** für den Informationsverbund. Hierbei werden Prozesse, Anwendungen, IT-Systeme, Infrastrukturen, etc. aufgelistet.
3. Mit Hilfe der **Schutzbedarfsfeststellung** wird ein angemessener Schutz für die Geschäftsprozesse, die darin verarbeiteten Informationen und die verwendete Informationstechnik ermittelt. Hierbei wird untersucht, welche Schäden durch eine Beeinträchtigung der Vertraulichkeit, Integrität und Verfügbarkeit für die Anwendungen und Informationen im Geltungsbereich entstehen können.
4. Bei der **Modellierung** müssen Sicherheitsanforderungen und umzusetzende Maßnahmen für den vorliegenden Informationsverbund identifiziert werden. Dies erfolgt mit den **Bausteinen** des IT-Grundschutz-Kompendiums [Bu17c].
5. Mit dem **IT-Grundschutz-Check** wird geprüft, ob die bereits umgesetzten Maßnahmen zur Absicherung des Informationsverbunds zum BSI-Grundschutz äquivalent sind. Abweichungen sind zu begründen oder nachzubessern. In unserem Anwendungsszenario wurden bisher keine Schutzmaßnahmen umgesetzt, daher übergehen wir diesen Schritt.
6. Zusätzlich ist eine **Risikoanalyse** durchzuführen, wenn
   - für ein Zielobjekt ein hoher oder sehr hoher Schutzbedarf besteht,
   - für ein Zielobjekt kein passender BSI-Baustein existiert, oder
   - das Zielobjekt auf eine Art und Weise betrieben wird, die der existierende Baustein nicht berücksichtigt.

   In diesem Fall müssen eine Gefährdungsübersicht für alle Elementargefährdungen erstellt und Szenario-spezifische Gefährdungen identifiziert werden. Diese Gefährdungen müssen ihrer potentiellen Schadenshöhe und Eintrittswahrscheinlichkeit entsprechend eingestuft und bewertet werden. Nach einer Risikobewertung können dann Behandlungsoptionen zum Umgang mit den Risiken festgelegt werden.

2.3 SYS. 1.5 Virtualisierung und SYS. 1.6 Container

Im BSI-Grundschutz-Kompendium [Bu17c] werden abzusichernde Zielobjekte in Form von Bausteinen beschrieben. Für jedes Zielobjekt wird im Rahmen des Bausteins eine Zielstellung definiert, die das Ergebnis der Absicherung des Zielobjektes beschreibt. Darüber hinaus findet in jedem Baustein eine klare Abgrenzung statt, in der der Rahmen festgelegt

wird, welche Bestandteile zur Absicherung des Zielobjektes zum Baustein gehören und welche nicht. Im Weiteren werden in jedem Baustein spezifische Gefährdungen für das Zielobjekt beschrieben. Zur Abwendung dieser Gefährdungen werden in jedem Baustein Anforderungen definiert. Diese sind unterteilt in (1) Basis-Anforderungen, die zwingend umgesetzt werden müssen, in (2) Standard-Anforderungen, die umgesetzt werden sollten um einen Grundschutz zu erreichen, und in (3) Anforderungen, die bei einem erhöhten Schutzbedarf in Betracht gezogen werden sollten. Zuletzt werden in jedem Baustein zusätzliche Informationen zu Gefährdungen und Sicherheitsmaßnahmen bereitgestellt. In den Anlagen ist zusätzlich eine Übersicht zu elementaren Gefährdungen für das Zielobjekt angehangen.

Zur Absicherung des Docker-Systems benötigen wir zwei Bausteine. Der Virtualisierungs-Baustein SYS. 1.5 des BSI-Kompendiums [Bu17c] behandelt die Gefährdungslage für Virtualisierungs-Systeme. Zwar adressiert der Baustein explizit nicht die Container-Virtualisierung. Da diese jedoch auf eine klassische Virtualisierung aufsetzt, haben wir SYS. 1.5 in unsere Analyse mit einbezogen. Der Baustein identifiziert folgende Gefahren:

- 2.1 Fehlerhafte Planung der Virtualisierung
- 2.2 Fehlerhafte Konfiguration der Virtualisierung
- 2.3 Unzureichende Ressourcen für virtuelle IT-Systeme
- 2.4 Informationsabfluss oder Ressourcen-Engpass durch Snapshots
- 2.5 Ausfall des Verwaltungsservers für Virtualisierungs-Systeme
- 2.6 Missbräuchliche Nutzung von Gastwerkzeugen
- 2.7 Kompromittierung der Virtualisierungssoftware

Der im Mai 2018 als Community Draft veröffentlichte Baustein SYS. 1.6 [Bu] berücksichtigt explizit die Gefährdungslage für die Container-Virtualisierung. In diesem Baustein werden folgende Gefahren beschrieben:

- 2.1 Schwachstellen in Images
- 2.2 Administrative Zugänge ohne Absicherung
- 2.3 Tool-basierte Orchestrierung ohne Absicherung
- 3.4 Datenverluste durch fehlende Persistenz
- 2.5 Vertraulichkeitsverlust von Zugangsdaten

Beide Bausteine sind hersteller- und produktneutral verfasst. Das heißt, es wird beschrieben, wie in einer Virtualisierungsumgebung die verarbeiteten, bereitgestellten und übertragenen Informationen anwendungsunabhängig zu schützen sind, und wie Container abzusichern und zu orchestrieren sind. Darüber hinaus wird beschrieben, wie verwendete Images vom Anwender selbst verwaltet werden können.

Damit bleibt die Frage offen, ob die in den Bausteinen aufgeführten Elementargefährdungen und spezifischen Gefährdungen auch die für Docker spezifischen Bedrohungen für die IT-Sicherheit mit abdecken, und wie gut es möglich ist, diese Bedrohungen mit den Bau-

steinen als Handlungsunterstützung zu identifizieren und ihnen die passenden Maßnahmen gegenüberzustellen.

2.4 Identifikation zusätzlicher Gefährdungen

Wenn der Schutzbedarf für die betrachtete IT-Komponente in einem drei Grundwerte „Vertraulichkeit“, „Integrität“ oder „Verfügbarkeit“ die Schutzbedarfsklasse „normal“ überschreitet, dann fordert der BSI-Standard 200-3 [Bu17b] eine Risikoanalyse. In dieser Risikoanalyse sollen die zusätzlichen Gefährdungen identifiziert, eingeschätzt und um Maßnahmen ergänzt werden, die über die in den Bausteinen aufgeführten Gefährdungen hinausgehen.

Der Standard schreibt vor, dass nach Gefährdungen zu suchen ist, die zu einem beträchtlichen Schaden führen können und die im vorliegenden Anwendungsfall und Einsatzumfeld realistisch sind. Zu diesem Zweck gibt der Standard folgende Fragen zur Ermittlung zusätzlicher Gefährdungen vor:

- Welche Gefahren aus dem Bereich „höhere Gewalt“ sind für unseren Informationsverbund besonders relevant?
- Wie kann die Informationssicherheit durch die Vermeidung organisatorischen Mängel vermieden werden?
- Durch welche menschlichen Fehlhandlungen kann die Sicherheit der Informationen besonders beeinträchtigt werden?
- Welche speziellen Sicherheitsprobleme kann technisches Versagen im betrachteten Informationsverbund hervorrufen?
- Welche Gefährdungen können durch externe Angriffe entstehen?
- Wie ist es internen Mitarbeitern möglich, den sicheren Betrieb des jeweiligen Zielobjekts mutwillig zu beeinträchtigen?
- Können besondere Gefahren durch Objekte entstehen, die nicht im betrachteten Informationsverbund berücksichtigt wurden?
- Welche Hinweise geben die Herstellerdokumentation sowie Warn- und Informationsdienste von Dritten?

Diese Fragen sollen von Experten, Mitarbeitern, Administratoren und Benutzern gemeinsam beantwortet werden.

3 Schutzbedarfsfeststellung und Elementargefährdungen für Docker

In diesem Abschnitt entwickeln wir nach BSI-Standard 200-2 Abschnitt 8 [Bu17a] eine Standard-Absicherung für unser Anwendungsszenario. Da wir uns auf das Docker-System konzentrieren, beginnen wir mit der Modellierung des Informationsverbunds und einer Schutzbedarfsfeststellung. In einem nächsten Schritt untersuchen die in den Bausteinen SYS. 1.5 und SYS. 1.6 vorgegebenen Elementargefährdungen auf ihre Anwendbarkeit für Docker.

3.1 Das Docker-System

Unser in Abbidlung 1 dargestelltes Docker-Szenario besteht aus einem Online-Shop, dessen Komponenten auf drei Container aufgeteilt ist. In Container A läuft eine Webanwendung auf einem Webserver, die ein Shop-System incl. Einkaufskorb, Kundenrezensionen etc. umsetzt. In Container B wird ein Webservice betrieben, der die Zahlungsabwicklung für unseren Onlineshop realisiert. Container C enthält eine Datenbank, die Produkt-, Kunden- und Bestelldaten beinhaltet. Diese drei Container werden isoliert (jeder Container besitzt sein eigenen Ressourcen) voneinander betrieben und bilden zusammen mit der Docker Engine das Host-System. Der Informationsverbund ist in den nachfolgenden Tabellen zusammengefasst:

Nr.	Datenobjekt	Beschreibung
D1	Personendaten	Einzelangaben zu einer natürlichen Person
D2	Nutzdaten	Generische Fachdaten der Anwendungen und Services
D3	Accountdaten	Anmelde- und Berechtigungsdaten der Anwender
D4	Konfigurationsdaten	Daten zur Änderung, Einstellung und Anpassung an IT-Systeme
D5	Protokolldaten	Statusinformationen und Funktionen von IT-Systemen

Nr.	Beschreibung	verarbeitete Daten	Software
A1	Webanwendung	D1, D2, D3, D4, D5	Allgemeine Anwendung z.B. PHP
A2	Webserver	D1, D2, D4, D5	Apache Webserver
A3	Webservice	D2, D3, D4, D5	REST-basierter Dienst
A4	Datenbank	D1, D2, D3, D4, D5	Allgemeine Datenbank z.B. MySQL

Nr.	Beschreibung	verarbeitete Daten	IT-System
SSW1	Docker Software	D4, D5	S1 und S1

Nr.	Beschreibung	verarbeitete Daten	Plattform	Ort
S1	Host-System	D1, D2, D3, D4, D5	x86 Linux-Server	RZ 1

Dieser Informationsverbund ist typisch für viele Docker-Installationen. Da wir uns auf die Absicherung von Docker konzentrieren wollen, haben wir unter S1 „Host-System“ die gesamte Host-Umgebung zusammengefasst, d.h., das Rechenzentrum mit dem Host-Rechner und dem darauf installierten Host-Betriebssystem.

3.2 Schutzbedarfsfeststellung

Um herauszufinden, welche Maßnahmen für den Schutz der Objekte in unserm Informationsverbund angemessen sind, haben wir eine Schutzbedarfsfeststellung durchgeführt. Wir verwenden die im Standard 200-2 [Bu17a] definierten Schutzbedarfskategorien „normal“, „hoch“ und „sehr hoch“.

Bei der Schutzbedarfsfeststellung vererben sich die Schutzbedarfe einzelner Datenobjekte auf die Anwendungen, die diese Daten verarbeiten, und von dort auf die Systeme, auf denen diese Anwendungen ablaufen. Speichert ein System Daten mit unterschiedlichen Schutzbedarfen, so wird dem System der höchste dieser Schutzbedarfe zugewiesen.

Daraus ergibt sich eine Besonderheit für die Container-Virtualisierung: Sämtliche Container laufen auf der gleichen physischen Maschine. Funktionen des Betriebssystem-Kernels des Hosts werden von allen Containern gleichermaßen verwendet. Zudem funktioniert das Gesamtsystem – in unserem Falle der Online-Shop – nur, wenn sämtliche Container betriebsbereit sind. Deswegen vererbt sich der höchste Schutzbedarf jedes einzelnen Containers automatisch auf den gesamten Informationsverbund. Für die Schutzbedarfsfeststellung genügt es deswegen, über alle Container hinweg nach den Daten oder Diensten mit dem höchsten Schutzbedarf bezüglich Vertraulichkeit, Integrität und Verfügbarkeit zu suchen und diesen dann für das Gesamtsystem zu übernehmen. Für unser Anwendungsszenario bedeutet dies:

- **Vertraulichkeit:** In Container C wird eine Datenbank betrieben, die Kundendaten mit Personenbezug speichert. Daher besteht für den Informationsverbund ein hoher Schutzbedarf für den Grundwert Vertraulichkeit.
- **Integrität:** In Container B werden die Zahlungsvorgänge der Kunden abgewickelt. Der Schutzbedarf des Informationsverbunds bezüglich der Integrität ist deshalb hoch.
- **Verfügbarkeit:** Der Web-Shop ist geschäftskritisch, funktioniert aber nur, wenn alle drei Container sowie das Betriebssystem und die Hardware verfügbar sind. Deswegen besteht für den gesamten Informationsverbund ein hoher Schutzbedarf für die Verfügbarkeit.

Unsere zentrale Erkenntnis aus der Schutzbedarfsfeststellung ist, dass die Schutzbedarfe für Vertraulichkeit, Integrität und Verfügbarkeit für sehr viele Einsatzszenarien mindestens „hoch“ sind. Dies gilt beispielsweise für alle von Docker aufgeführten Kundenprojekte [Doa]. Bezogen auf den BSI-Grundschutz bedeutet dies, dass in jedem Fall nach der Standard-Absicherung eine Risikoanalyse durchzuführen ist (s. Abschnitt 4).

3.3 Analyse der Elementargefährdungen

Nach der Schutzbedarfsfeststellung sieht das BSI die Modellierung eines Grundschutzkonzepts auf der Basis der im Grundschutz-Kompendium definierten Bausteine vor. Wir haben bereits festgestellt, dass für das Docker-System die Bausteine SYS. 1.5 und SYS. 1.6 relevant sind. Wir haben nun geprüft, welche der in den beiden Bausteinen genannten Elementargefährdungen für unser Docker-System zutreffen. Elementargefährdungen sind grundsätzlicher Natur, d.h., sie treffen für verschiedene Objekte im Informationsverbund zu. SYS. 1.5 und SYS. 1.6 listen 25 Gefährdungen auf:

- G 0.14 Ausspähen von Informationen (Spionage)
- 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- oder Software
- G 0.22 Manipulation von Informationen
- G 0.23 Unbefugtes Eindringen in IT-Systeme
- G 0.24 Zerstörung von Geräten oder Datenträgern
- 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 0.37 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

Das BSI bietet zu jeder dieser Elementargefährdungen eine kurze Erläuterung an. Anhand der Erläuterungen lässt sich nachvollziehen, dass tatsächlich jede dieser allgemeinen Gefährdungen auch auf unser Docker-System zutrifft. Mit einer Kreuzreferenztabelle werden in den Bausteinen den einzelnen Elementargefährdungen spezifische Sicherheitsanforderungen zugeordnet, deren Umsetzung das Risiko für einen Schaden im Sinne der Gefährdungen auf ein für die Standard-Absicherung ausreichendes Maß senkt. Beispielsweise sind der Elementargefährdung G 0.20 (Informationen oder Produkte aus unzuverlässiger Quelle) die folgenden Anforderungen zugeordnet:

SYS. 1.5. A10 Verwaltungsprozesse virtueller Systeme

SYS. 1.5. A27 zertifizierte Virtualisierungssoftware
SYS. 1.6. A1 Planung des Container-Einsatzes
SYS. 1.6. A7 Verwendung sicherer Images

Zu jeder Anforderung gehört ebenfalls eine präzise Erläuterung. Da der Umgang mit einmal identifizierten Elementargefährdungen bereits etablierte Praxis ist, möchten wir darauf nicht näher eingehen. Im nächsten Abschnitt legen wir unseren Fokus auf den Umgang mit Docker-spezifischen Bedrohungen, die über Elementargefährdungen hinausgehen.

4 Docker-spezifische Gefährdungen

Der BSI-Grundschutz kennt zwei Arten von spezifischen Gefährdungen: (a) Die in den Bausteinen genannten, über die Elementargefährdungen hinausgehenden Bedrohungen und (b) die im Rahmen einer Risikoanalyse identifizierten zusätzlichen individuellen Gefährdungen für die im Informationsverbund modellierten Objekte. Da in unserem Informationsverbund mehrere Objekte einen über „normal“ hinausgehenden Schutzbedarf ausweisen, haben wir im Dialog mit Experten der Open Telekom Cloud eine Risikoanalyse durchgeführt (vgl. Abs. 2.2). Im Folgenden werden die dabei identifizierten Gefährdungen aufgelistet, eine Risikobewertung vorgenommen und ergänzende Maßnahmen zur Risikobehandlung aufgeführt. Dabei haben wir untersucht, ob sich die uns identifizierten Gefährdungen von denen der BSI-Bausteine unterscheiden.

4.1 Identifikation zusätzlicher Gefährdungen

Im Rahmen der Risikoanalyse haben wir 14 Gefährdungen identifiziert (s. Abbildung 2). 12 dieser Gefährdungen sind auch in den Bausteinen SYS. 1.5 und SYS. 1.6 als spezifische Gefährdungen enthalten. Darüber hinaus konnten wir zwei zusätzliche Gefährdungen identifizieren. Dieser Sachverhalt ist nachfolgend in Abb. 2 dargestellt.

Details zu allen von uns identifizierten Gefährdungen finden sich in [Ba17]. Die beiden nicht in den Bausteinen enthaltenen Gefährdungen sind der Container Breakout und die unautorisierte Änderung von Konfigurationsdateien:

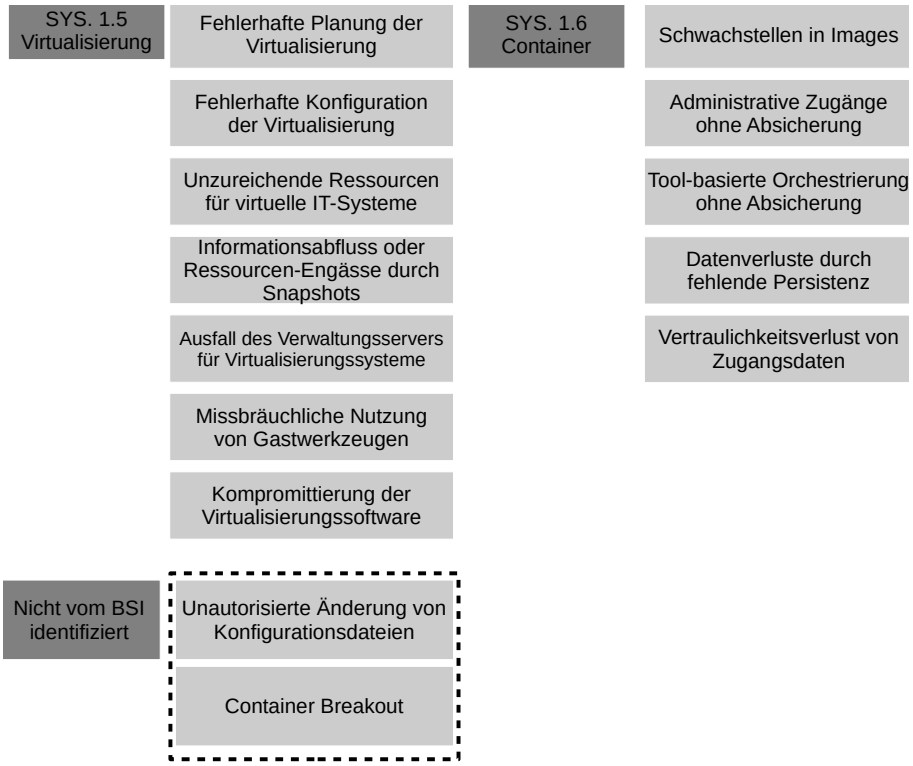

Abb. 2: Spezifische Gefährdungen für das Docker-System

Docker-System Vertraulichkeit: hoch Integrität: hoch Verfügbarkeit: hoch
G z. 0.1 Container Breakout
Beschreibung G z. 0.1: Der Container Breakout erlaubt einem Angreifer Zugriff auf das Host-System oder auf weitere Container im gleichen System, und zwar mit den Privilegien des Containers, aus dem der Ausbruch erfolgte.
G z. 0.2 Unautorisierte Änderung von Konfigurationsdateien
Veränderungen an der Konfiguration, insbesondere der des mit root-Privilegien arbeitenden Docker Daemons, können sich auf alle Container im System auswirken.

Ein **Container Breakout** [Ro] wird möglich, wenn die Container durch Schwachstellen in der Implementierung nicht lückenlos voneinander isoliert sind. Bei einem erfolgreichen

Breakout kann der Angreifer mit den Privilegien des Containers, aus dem er ausgebrochen ist, auf Daten oder Dienste des Host-Systems oder anderer Container zugreifen. Der Container Breakout ist daher sehr ähnlich zu den in der Vergangenheit bereits häufig vorgekommenen Ausbrüchen aus einer Java-Sandbox [Co15]. Ein Container Breakout beeinträchtigt nicht nur die Vertraulichkeit, sondern auch die Verfügbarkeit und die Integrität von anderen Objekten im Informationsverbund.

Der Baustein SYS. 1.6 adressiert diesen Punkt nur indirekt durch die Basis-Anforderung SYS. 1.6. A2 „Planung der Separierung“. Diese Anforderung zielt auf die Separierung von Containern mit jeweils unterschiedlichen Schutzbedarfen im Rahmen eines Netzzonenkonzepts ab.

Durch **unautorisierte Änderungen an Konfigurationsdateien** der virtuellen Infrastruktur können erhebliche und tiefgreifende Schäden entstehen, ebenso wie durch vorsätzliche oder versehentliche Fehlkonfigurationen der Netzzuordnung. Hier stellt insbesondere der Docker Daemon eine Angriffsoberfläche dar, da dieser root-Privilegien besitzt und die Funktionsfähigkeit aller Container beeinflussen kann. Für Vertraulichkeit, Integrität oder Verfügbarkeit der Objekte im Informationsverbund ist die Integrität von Konfigurationsdaten daher ausschlaggebend.

Auch diese Gefährdung wird nur mittelbar in den Bausteinen des Grundschutz-Kompendiums adressiert. Die Standard-Anforderung A17 „Überwachung des Betriebszustands und der Konfiguration der virtuellen Infrastruktur“ im Baustein SYS. 1.5 adressiert die Netzzuordnungen im Virtualisierungslayer. Der Baustein SYS. 1.6 betrachtet Konfigurationsdateien nur im Bezug auf die Freigabe von Images (Standard-Anforderung A12), jedoch nicht im Hinblick auf die komplexe Orchestrierung von Containern zur Laufzeit.

4.2 Risikoeinstufung

Nachdem sämtliche spezifische Gefährdungen für das Docker-System identifiziert und die nicht von den BSI-Bausteinen adressierten zusätzlichen Gefährdungen erläutert wurden, muss im nächsten Schritt das Risiko ermittelt werden, welches von der jeweiligen Gefährdung ausgeht. Dazu wird nach BSI-Standard 200-3 eine qualitative Risikobewertung herangezogen. Abbildung 3 zeigt den Zusammenhang zwischen Eintrittshäufigkeiten und Schadenshöhen.

Unsere zentrale Erkenntnis ist hier, dass sich alle spezifischen Gefährdungen für das Docker-System auf technische Schwachstellen beziehen, für die sich Angriffe automatisieren lassen. Wird beispielsweise ein Exploit bekannt, durch den sich ein Container Breakout durchführen lässt, so kann dieser Exploit auch automatisiert auf eine große Zahl von anfälligen Containern angewendet werden. Wir gehen darum davon aus, dass die Eintrittshäufigkeit für jede Gefährdung für Docker sehr häufig ist (gestrichelter Bereich in Abb. 3. Für die Risikobewertung genügt es also, die Schadenshöhe der jeweiligen Gefährdungen zu ermitteln.

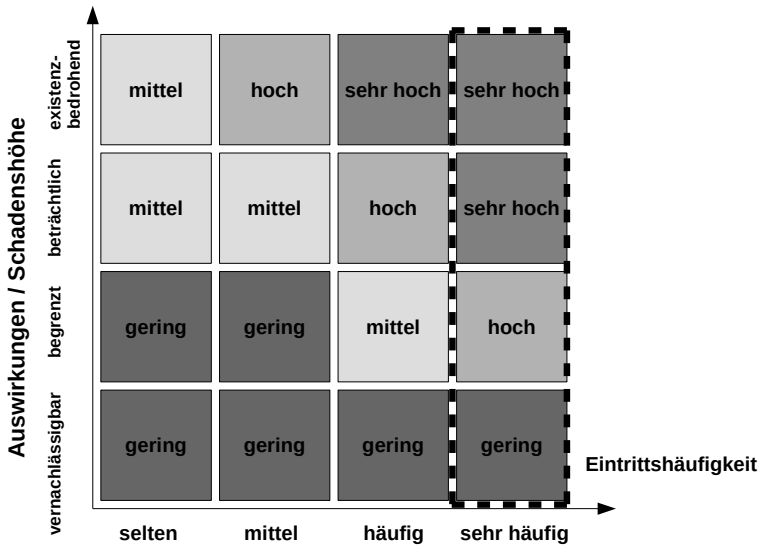

Abb. 3: Risikoeinstufung nach BSI [Bu17b]

4.3 Risikobewertung

Die Risikobewertung für die nicht in den BSI-Bausteinen berücksichtigten, zusätzlichen Gefährdungen ist wie folgt:

Docker-System	Vertraulichkeit: hoch Integrität: hoch Verfügbarkeit: hoch	
Gefährdung G z 0.1: Container Breakout	Beeinträchtigte Grundwerte: Vertraulichkeit	
Eintrittshäufigkeit ohne zus. Maßnahme: sehr häufig	**Auswirkungen** ohne zusätzliche Maßnahmen: beträchtlich	**Risiko** ohne zusätzliche Maßnahme: hoch
Beschreibung: Ein Gefährdungsszenario ist der Container Breakout, der einem Angreifer Zugriff auf das Host-System oder auf weitere Container im gleichen System erlaubt, und zwar mit den Privilegien des Containers, aus dem der Ausbruch erfolgte.		
Bewertung: Ein Container Breakout würde zum Verlust der Vertraulichkeit von z.B. Kundendaten führen. Diese gelten als hoch schutzbedürftig, sodass das Schadensausmaß bei einem Container Breakout beträchtlich und das Risiko als hoch einzustufen ist.		

Docker-System	Vertraulichkeit: hoch Integrität: hoch Verfügbarkeit: hoch	
Gefährdung G z 0.2: Unautorisierte Änderung von Konfigurationsdateien	Beeinträchtigte Grundwerte: Vertraulichkeit, Integrität, Verfügbarkeit	
Eintrittshäufigkeit ohne zus. Maßnahme: sehr häufig	**Auswirkungen** ohne zusätzliche Maßnahmen: beträchtlich	**Risiko** ohne zusätzliche Maßnahme: hoch
Beschreibung: Durch unautorisierte Änderungen an Konfigurationsdateien der virtuellen Infrastruktur können erhebliche und tiefgreifende Schäden entstehen, ebenso wie durch vorsätzliche oder versehentliche Fehlkonfigurationen der Netzzuordnung. Hier stellt insbesondere der Docker Daemon eine Angriffsoberfläche dar, da dieser root-Rechte besitzt und die Funktionsfähigkeit aller Container beeinflussen kann.		
Bewertung: Für Vertraulichkeit, Integrität oder Verfügbarkeit der Daten ist die Integrität von Konfigurationsdaten ausschlaggebend. Unautorisiertes Ändern solcher Daten kann zu erheblichen Schäden für das Unternehmen führen, weshalb das Risiko als hoch einzustufen ist.		

4.4 Risikobehandlung

Das BSI definiert verschiedene Risikobehandlungsoptionen (Risikoreduktion durch zusätzliche Sicherheitsmaßnahmen oder Umstrukturierung der Prozesse, Risikoakzeptanz oder Risikotransfer). Da die ermittelten zusätzlichen Gefährdungen ausschließlich „hoch“ eingestuft wurden, scheidet „Risikoakzeptanz“ als Option aus. Für die Gefährdungen, die auch das BSI identifiziert hat, verweisen wir auf die Bausteine SYS.1.5 und SYS.1.6.

Die Risikobehandlungsoptionen für die beiden zusätzlichen Gefährdungen werden in der folgenden Tabelle aufgelistet:

Docker-System	Vertraulichkeit: hoch Integrität: hoch Verfügbarkeit: hoch	
Gefährdung	Risiko-kategorie	Risikobehandlungsoption
G z 0.1 Container Breakout	hoch	**Definition der Systembenutzer:** Docker-Container sind grundsätzlich nicht als privilegierte Container zu betreiben, damit Angreifer im Erfolgsfall nur unprivilegierten Zugriff auf andere Ressourcen erhalten. **Rechtemanagement:** Es sind die Berechtigungen für die definierten Benutzergruppen auf Minimalität zu prüfen. **Rollenaufteilung:** Es ist auch für virtuelle IT-Systeme eine Aufteilung in verschiedene Rollen notwendig. Weitere Linux Funktionen: Schutzmaßnahmen wie apparmor, selinux, seccomp, Filter und namespaces, die auf dem Host-System installiert werden, können das Risiko eines Ausbruchs aus einem Gastcontainer reduzieren.
G z 0.2 Unautorisierte Änderung von Konfigurationsdateien	hoch	**Prüfsummen:** Die Prüfung auf unautorisierte Änderungen der Konfigurationsdateien kann beispielsweise mittels Werkzeugen wie OS-SEC erfolgen [OS]. **Docker Bench for Security:** Docker selbst bietet das Docker Bench for Security Script [Ce] an, welches die eigene Docker Konfiguration prüft. Voraussetzung ist eine Dockerversion 1.10.0 oder aktueller. **Konfiguration der Netzfunktionen:** Da Docker Container auf einem gängigen Linux-System betrieben werden, kann man auf bekannte Werkzeuge wie beispielsweise Puppet [AJ17] zurückgreifen, um die Netzkomponenten zentral zu überwachen. **Benennung virtueller Netze:** Wenn Netzverbindungen auf verschiedenen Host-Systemen gleich benannt sind, kann ein Container versehentlich mit dem falschen Netzwerk verbunden werden. Eine eindeutige und aussagekräftige Benennung der Netze sollte anhand der Funktion des Netzwerkes vorgenommen werden [AJ17]. **Storage Zentralisierung:** Im Sicherheitskonzept muss festgelegt werden, ob Daten nach Beenden des Containers gelöscht werden oder ob ein Dateiverzeichnis des Containers auf ein Dateiverzeichnis des Host- Systems verknüpft wird. Das Host-System muss dann die Isolation von Betriebssystem, Systembibliotheken [Va17] und gemeinsamen Anwendungen sicherstellen.

		Monitoring: Das Monitoring lässt sich durch den Einsatz eines Linux-Servers mit den systemeigenen Monitoring Systemen wie Nagios bewerkstelligen [AJ17]. **Kommunikation zwischen Containern:** Wird das Container Linking aktiviert [Ja], so müssen Container, die nicht miteinander kommunizieren dürfen, über einen anderen Host betrieben werden.

4.5 Anforderungen für ergänzende Maßnahmen

Innerhalb eines jeden Bausteins werden im BSI-Grundschutz-Kompendium Anforderungen definiert, die zur Absicherung des betreffenden Zielobjektes eingehalten werden müssen. Damit die von uns vorgeschlagenen Maßnahmen zur Abwehr der noch nicht vom BSI identifizierten Gefährdungen auch tatsächlich im neuen Baustein SYS.1.6 umgesetzt werden können, ist es daher im letzten Schritt erforderlich, für die von uns vorgeschlagenen Maßnahmen entsprechende Anforderungen für den neuen Baustein zu definieren. Einige unserer Maßnahmen werden hierbei schon durch entsprechende Anforderungen vom BSI abgedeckt. Nachfolgend werden die entsprechenden Maßnahmen und die zugehörigen Anforderungen aufgelistet:

- **Maßnahme:** Definition der Systembenutzer
  Baustein/Anforderung: SYS. 1.6 / A17
- **Maßnahme:** Rechtemanagement
  Baustein/Anforderung:SYS. 1.5 / A25
- **Maßnahme:** Rollenaufteilungen
  Baustein/Anforderung: SYS. 1.6 / A16, A17, A23
- **Maßnahme:** Prüfsummen
  Baustein/Anforderung: SYS.1.6 / A12
- **Maßnahme:** Konfiguration der Netzfunktionen
  Baustein/Anforderung: SYS. 1.5 / A4

Für alle anderen Maßnahmen müssen nun Anforderungen definiert werden.

- **Anforderung für Docker Bench Security:** Für jeden Container ist die Docker Bench Security zur Prüfung der Konfigurationen eines Containers anzuwenden.
- **Anforderung für die Benennung virtueller Netze:** Für jedes virtuelle Netz ist ein eigener Name zu verwenden.
- **Anforderung für die Storage Zentralisierung:** Für jeden Container ist festzulegen, ob dessen Daten nach Beendigung gespeichert bleiben oder gelöscht werden.

- **Anforderung für das Monitoring:** Überwachung der Konfigurationsdaten zur Verhinderung unautorisierter Änderung ist für jeden Container durchzuführen.
- **Anforderung für die Kommunikation zwischen Containern:** Alle Container, die nicht miteinander kommunizieren, müssen voneinander isoliert werden.

Da wir in unserem Beispiel einen hohen Schutzbedarf für das gesamte System ermittelt haben, müssen unsere definierten Anforderungen im Baustein SYS.1.6 entsprechend dem Bereich der Anforderungen für erhöhten Schutzbedarf zugeordnet werden.

5 Diskussion

In diesem Abschnitt wird kurz diskutiert, inwiefern sich die in einem typischen Anwendungsfall für das Docker-System gewonnenen Erkenntnisse (a) auf die Container-Virtualisierung und (b) auf allgemeine Anwendungsszenarien anwenden lassen.

(a) Container-Virtualisierung Der BSI-Baustein SYS. 1.6 ist bereits von der eingesetzten Technologie unabhängig definiert. Die von uns identifizierten zusätzlichen Gefährdungen sind jedoch ebenfalls nicht Docker-spezifisch. Im Gegensatz zur traditionellen Hypervisor-Virtualisierung [Ch07] nutzen die leichtgewichtigen Container Funktionen aus dem Kernel Host-Betriebssystems [Dob], beispielsweise um zu kommunizieren oder um Ressourcen zu allokieren. Diese Funktionen öffnen potentielle Zugriffspfade für einen Angreifer, um aus der isolierten Container-Umgebung auszubrechen. Auch unautorisierte Änderungen der Konfigurationsdateien stellen für Container eine Gefährdung dar. Jeder Container wird entsprechend seiner benötigten Berechtigungen konfiguriert. Unautorisierte Änderungen an den Konfigurationsdateien können daher einen erheblichen Einfluss auf die Integrität, Verfügbarkeit und Vertraulichkeit sowohl der Container als auch des Host-Systems haben. Es würde sich daher anbieten, diese beiden Gefährdungen explizit in den neuen Container-Bausteins SYS. 1.6 aufzunehmen.

(b) allgemeine Anwendungsszenarien Unsere Risikoanalyse hat ergeben, dass sich die von uns identifizierten zusätzlichen Gefährdungen für automatisierbare Angriffe eignen, sobald eine entsprechende Schwachstelle für eine Container-Technologie entdeckt wird. Daher sind unsere Erkenntnisse über unser Anwendungsszenario und dessen konkrete Schutzbedarfe hinaus wichtig.

Wir haben unsere Risikoanalyse auf der Basis einer Schutzbedarfsfeststellung durchgeführt, bei der die Bedarfe für Vertraulichkeit, Integrität und Verfügbarkeit für das Gesamtsystem mit „hoch“ festgesetzt wurde. Wir haben festgestellt, dass dies aufgrund des Maximumprinzips typisch ist für viele kommerzielle Anwendungen der Container-Virtualisierung. Für Anwendungsfälle, bei denen die Schadensauswirkungen ein „existenziell bedrohliches,

katastrophales Ausmaß erreichen“ [Bu17a] können, ist jedoch eine umfassendere Risikoanalyse erforderlich. Ein Beispiel für so ein Anwendungsszenario könnte ein Krankenhaus sein, das medizinische Geräte über eine Container-Lösung steuert.

6 Zusammenfassung

Das Ziel dieser Arbeit bestand darin, zu untersuchen, wie gut die aktuellen BSI-Standards und der neue Baustein SYS. 1.6 auf einen typischen Anwendungsfall der Container-Virtualisierung angewendet werden kann. Dazu haben wir eine Standard-Absicherung und eine Risikoanalyse nach BSI IT-Grundschutz für Docker Container in einer On-Premise-Umgebung durchgeführt. Wir haben festgestellt, dass der neue Baustein SYS. 1.6 in Verbindung mit dem Virtualisierungs-Baustein SYS. 1.5 ein wertvolles Werkzeug bei der Erstellung eines Sicherheitskonzepts für Docker darstellt. In unserem konkreten Anwendungsfall hat sich jedoch gezeigt, dass zwei zusätzliche Gefährdungen für Docker existieren, die im Rahmen des neuen Bausteins SYS. 1.6 noch nicht berücksichtigt wurden. Wir haben gezeigt, dass sich unsere gewonnenen Erkenntnisse nicht nur auf Docker-Szenarien beschränken sondern im Allgemeinen für Container-Technologien gelten. Daher ist eine Ergänzung des Baustein SYS. 1.6 um unsere zusätzlichen Gefährdungen sowie der dazugehörigen Maßnahmen und Anforderungen zu überlegen.

Literatur

[AJ17] Atug, M.; Jedecke, D.: iX Kompakt - Container und Virtualisierung. Heise Medien, 2017.

[Ba17] Bauer, S.: Erarbeitung eines Informationssicherheitskonzepts nach IT-Grundschutz für Docker Container. Bachelor-Arbeit, Hochschule für Telekommunikation Leipzig, Kopie s. http://www.webcitation.org/6xAkE4g1l/, 2017.

[BHB18] Buchmann, E.; Hartmann, A.; Bauer, S.: Informationssicherheitskonzept nach IT-Grundschutz für Containervirtualisierung in der Cloud. SICHERHEIT 2018/, 2018.

[Bu] Bundesamt für Sicherheit in der Informationstechnik: SYS.1.6 Container, https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Grundschutz/IT-Grundschutz-Modernisierung/BS_Container.html, abgerufen Sept. 2018.

[Bu11] Bundesamt für Sicherheit in der Informationstechnik: Webkurs IT-Grundschutz, IT -Grundschutz im Selbststudium. https://www.bsi.bund.de/, 2011.

[Bu14] Bundesamt für Sicherheit in der Informationstechnik: Zertifizierung nach ISO 27001 auf der Basis von IT-Grundschutz. https://www.bsi.bund.de/, 2014.

[Bu16] Bundesamt für Sicherheit in der Informationstechnik: IT-Grundschutz-Kataloge, 15.Ergänzungslieferung. https://www.bsi.bund.de/, 2016.

[Bu17a] Bundesamt für Sicherheit in der Informationstechnik: BSI-Standard 200-2, IT-Grundschutz-Methodik. https://www.bsi.bund.de/, 2017.

[Bu17b] Bundesamt für Sicherheit in der Informationstechnik: BSI-Standard 200-3, Risikomanagement. https://www.bsi.bund.de/, 2017.

[Bu17c] Bundesamt für Sicherheit in der Informationstechnik: IT-Grundschutz-Kompendium 2018, 1. Edition. https://www.bsi.bund.de/, 2017.

[Ce] Center for Internet Security: Docker Community Edition Benchmark, https://www.cisecurity.org, abgerufen Sept. 2018.

[Ch07] Chisnall, D.: The Definitive Guide to the Xen Hypervisor. Prentice Hall, 2007.

[Co15] Coker, Z.; Maass, M.; Ding, T.; Le Goues, C.; Sunshine, J.: Evaluating the flexibility of the Java sandbox. In: Proceedings of the 31st Annual Computer Security Applications Conference. ACM, S. 1–10, 2015.

[Dä18] Dännart, S.; Diefenbach, T.; Hofmeier, M.; Rieb, A.; Lechner, U.: IT-Sicherheit in Kritischen Infrastrukturen–eine Fallstudien-basierte Analyse von Praxisbeispielen. Multi-Konferenz Wirtschaftsinformatik (MKWI'18)/, 2018.

[Doa] Docker Inc.: Docker Customers, https://www.docker.com/customers, abgerufen Sept. 2018.

[Dob] Docker Inc.: Docker Overview, https://docs.docker.com/engine/docker-overview, abgerufen Sept. 2018.

[Ec13] Eckert, C.: IT-Sicherheit: Konzepte-Verfahren-Protokolle. Walter de Gruyter, 2013.

[Gö17] Göbel, L.: Container-as-a-Service - Die Zukunft der Virtualisierung, https://www.cloudcomputing-insider.de/container-as-a-service-die-zukunft-der-virtualisierung-a-576244, abgerufen Sept. 2018, 2017.

[Ja] Jacqueline von Ogden: The Top 5 Security Risks in Docker Container Deployment, https://www.cimcor.com/blog/the-top-5-security-risks-in-docker-container-deployment, abgerufen Sept. 2018.

[OS] OSSEC Project Team: OSSEC's Documentation, https://ossec-docs.readthedocs.io/en/latest, abgerufen Sept. 2018.

[Pe15] Pethuru Raj Jeeva S. Chelladhurai, V. S.: Learning Docker. Packt Publishing, 2015.

[Ro] Rob Shapland: Eine Schwachstelle in Container-Techniken erlaubt Angriffe auf den Host, https://www.searchsecurity.de/tipp/Eine-Schwachstelle-in-Container-Techniken-erlaubt-Angriffe-auf-den-Host, abgerufen Sept. 2018.

[Va17] Vasily Tarasov, L. R.: In Search of the Ideal Storage Configuration for Docker Containers. IEEE 2nd International Workshops on Foundations and Applications of Self* Systems (FAS*W)/, 2017.