Version 1.0 (08.05.2025)

Inhaltsverzeichnis

1 Zielstellung Leitfaden Softwarelieferant . . . 1
2 Rolle Softwarelieferant für Entscheider . . . 1
3 Anforderungen an Softwarelieferanten der Deutschen Verwaltungscloud . . . 2
3.1 Phase 1: Planungsphase . . . 4
3.2 Phase 2: Umsetzungsphase . . . 7
3.3 Phase 3: Unterstützungsphase . . . 29
3.4 Phase 4: Wartungsphase . . . 29
4 Rollenübergreifende Erläuterungen zur DVC . . . 30
4.1 Motivation DVC . . . 30
4.2 Rahmenwerk . . . 30
4.3 Organisation der DVC . . . 30
4.4 Motivation Leitfäden . . . 30
4.5 Abgrenzung der Rollen . . . 30
4.6 Nomenklatur . . . 32
Anhang . . . 33
Glossar . . . 33
Referenzdokumente . . . 33

1 Zielstellung Leitfaden Softwarelieferant

Ziel dieses Dokumentes ist es, Softwarelieferanten aufzuzeigen, welche Möglichkeiten existieren, ihre Software über einen Cloud-Service-Anbieter in der DVC als Cloud-Service anbieten zu können und eine "Checkliste von notwendigen Schritten" zur Verfügung zu stellen ("Habe ich alles bearbeitet, um mir die Rolle Softwarelieferant zu erschließen?")

Gemäß DVC-Rollenkonzept (Rahmenwerk, Kap 3.2) ist die Rolle Softwarelieferant wie folgt beschrieben:

- Softwarelieferanten entwickeln Softwareanwendungen und stellen diese dem Softwarebetreiber bereit. Softwarelieferanten übernehmen ggf. die Wartung und Pflege der Software und sind in dieser Rolle Unterauftragnehmer der Softwarebetreiber. Softwarelieferanten können sowohl Akteure der ÖV als auch verwaltungsexterne Anbieter sein.

Für die Rollen Softwarebetreiber und Plattformbetreiber werden gesonderte Leitfäden bereitgestellt.

Der vorliegende Leitfaden beschreibt in Kapitel 2 zunächst eine Management-Sicht auf die Rolle Softwarelieferant. Anschließend folgen die konkreten Anforderungen in Kapitel 3 inkl. einer Checkliste aufgeteilt in die vier für einen Softwarelieferanten relevanten Phasen Planung, Entwicklung, Unterstützung bei Inbetriebnahme und Support nach der Produktivnahme. Abschließend enthält Kapitel 4 noch einige allgemeine rollenübergreifende Erläuterungen zur DVC.

2 Rolle Softwarelieferant für Entscheider

Bevor es in den folgenden Kapiteln schwerpunktmäßig um die technischen Fragestellungen rund um die Rolle Softwarelieferant geht, beschäftigt sich dieses Kapitel zunächst mit der Frage **„Warum sollte ich/mein Unternehmen/meine Community überhaupt in der Rolle Softwarelieferant der DVC aktiv werden?“**.

Mit der Deutschen Verwaltungscloud wird ein **neuer digitaler Marktplatz** für die Behörden von Bund, Ländern und Kommunen gebaut, auf dem sie Cloud-Services von den IT-Dienstleistern der öffentlichen Verwaltung schnell, sicher und souverän beziehen können.

Perspektivisch sollen auf diesem Marktplatz **alle IT-Lösungen** (soweit technisch sinnvoll), die die Verwaltung in Deutschland benötigt, **als Cloud-Services** angeboten und verwaltet werden.

Die Cloud-Services in der Deutschen Verwaltungscloud sollen also zukünftig den Behörden Deutschlands in Bund, Land und Kommunen angeboten werden. Das ist ein enormer Markt und bietet die Möglichkeit die souveräne Digitalisierung Deutschlands entscheidend voranzubringen.

Als Beitrag zur Erreichung der vier eigenen strategischen Ziele der Deutsche Verwaltungscloud-Strategie DVS

- Reduktion von Abhängigkeiten,

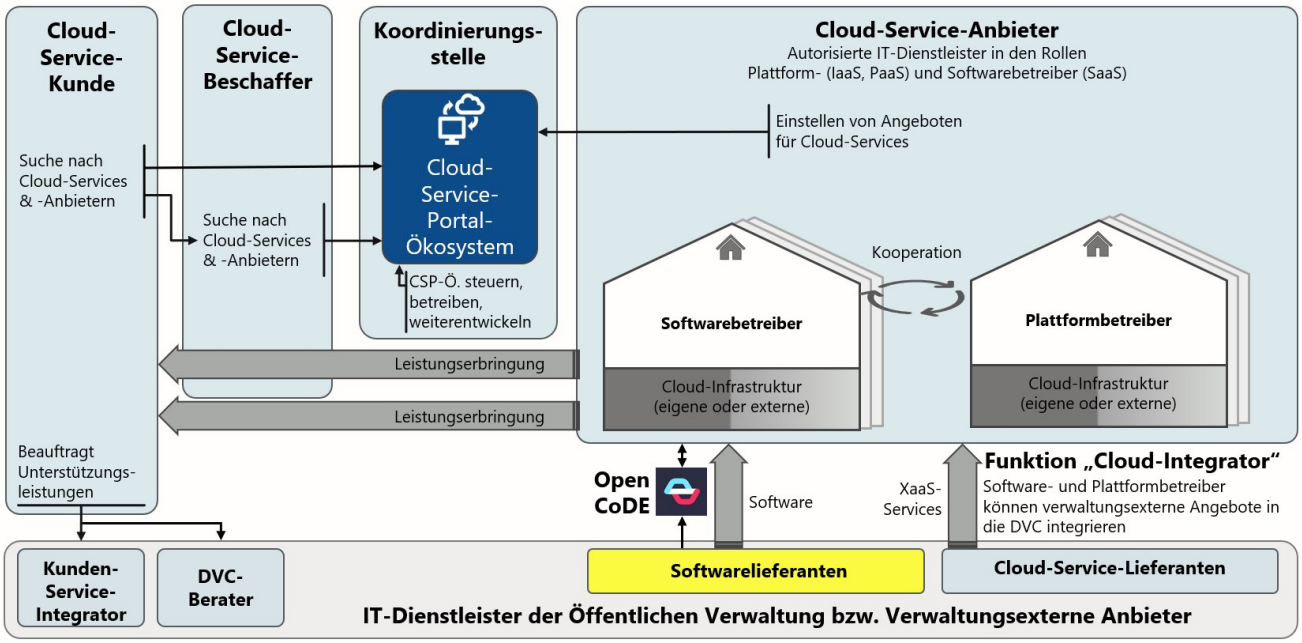

Abbildung 1: Zentrale Rollen und Akteure in der Deutschen Verwaltungscloud (illustrative Darstellung)

- Steigerung der Effizienz und Effektivität in Entwicklung, Inbetriebnahme und Betrieb,
- Sicherstellung und Stärkung von Datenschutz und Informationssicherheit,
- Optimierung von Datenaustausch, -speicherung und -nutzung zwischen Bund, Ländern und Kommunen.

gibt nachfolgender Leitfaden Rahmenbedingungen und Vorgaben für die Erstellung von Software und die Zusammenarbeit mit einem Softwarebetreiber.

Das kann Anpassungen innerhalb des gesamten DevOps-Lifecycles für Ihre Software bedingen, da Ihre Software eine Reihe von Kriterien erfüllen muss, damit sie in der DVC bereitgestellt werden kann. Sie haben dabei die technische Wahlfreiheit, wie Sie diese Ziele erreichen können. Die DVC macht keine konkreten Vorgaben bezüglich Entwicklungsumgebungen, Tools etc. sondern fordert nur, dass das Ergebnis - also Ihre Software - den technischen und formalen Kriterien entspricht.

Die Aufgabenverteilung innerhalb der Rollen in der DVC ermöglicht es Ihnen als Softwarelieferant sich auf die Herstellung DVC-konformer Software und die Vertragsbeziehung zu einem oder mehreren Softwarebetreibern zu konzentrieren, während der jeweilige Softwarebetreiber die DVC-konforme Herstellung und den Betrieb der entsprechenden Cloud-Services und die Vertragsbeziehungen zu den Cloud-Service Kunden in der Deutschen Verwaltungscloud übernimmt.

3 Anforderungen an Softwarelieferanten der Deutschen Verwaltungscloud

Wie sollten Sie als Softwarelieferant konkret den Einstieg in das Projekt “Wir erstellen Software zur Veröffentlichung in der DVC” vornehmen?

Die nachfolgenden Unterkapitel zeigen, in welchen Zyklen Sie dabei gemäß DevOps-Lifecycle vorgehen sollten:

- Phase 1: Planungsphase: Planungsphase (Anforderungen) inkl. Kontaktaufnahme / Verhandlungen mit einem oder mehreren Softwarebetreibern
- Phase 2: Umsetzungsphase: Inhouse-Aufgaben beim Softwarelieferanten: Erstellung der Software unter Beachtung der DVC-Vorgaben
- Phase 3: Unterstützungsphase: Unterstützung des Softwarebetreibers bei der Inbetriebnahme der Software als Service
- Phase 4: Wartungsphase: Nach der Produktivnahme des Services startet die vertraglich geregelte Wartungsphase

Ihren Fortschritt bei der Bearbeitung der erforderlichen Themen können Sie in der folgenden Checkliste protokollieren:

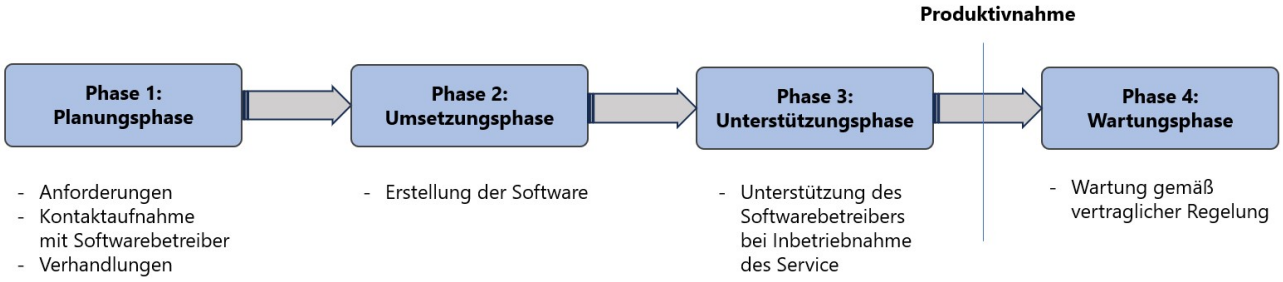

Abbildung 2: DVC Softwarelieferant Phasen

Phase	Anforderung	Erfüllt (ja/nein)	Referenz
Planungsphase	Softwarebetreiber gefunden?		Kapitel 3.1.1
Planungsphase	Vertrag mit Softwarebetreiber geschlossen?		Kapitel 3.1.2
Planungsphase	Formale Anforderungen an die Software erfüllt?		Kapitel 3.1.3
Umsetzungsphase	Vorgaben für "Interaktion zwischen Softwarebetreiber und Softwarelieferant" erfüllt?		Kapitel 3.2.1
Umsetzungsphase	Vorgaben für die "Erstellung von Software" erfüllt?		Kapitel 3.2.2
Umsetzungsphase	Vorgaben für "Dokumentation" erfüllt?		Kapitel 3.2.3
Unterstützungsphase	Sonstige Entwicklungs-Vorgaben erfüllt? ("Lieferung", "Architektur", "Export von Daten", "IAM-Anbindung")		Kapitel 3.2.4 bis 3.2.7
Unterstützungsphase	Einrichtungssupport eingeplant?		Kapitel 3.3.1
Unterstützungsphase	Support im operativen Tagesgeschäft eingeplant?		Kapitel 3.3.2
Wartungsphase	IT Servicemanagement eingeplant?		Kapitel 3.4

Die im Folgenden aufgelisteten für die Rolle Softwarelieferant relevanten Anforderungen sind 1:1 aus anderen Dokumenten (Detailstandards, Rahmenwerk, Reifegradmodell, Whitepaper) übernommen worden. Dabei wurde die folgende Nomenklatur gewählt, die auch in den zugrundeliegenden DVC-Detailstandards verwendet wird:

ID	Rolle	Modalverb	Detailanforderung
s. Tabelle in Kapitel 4.6 Nomenklatur	Softwarelieferant	MUSS | SOLL | KANN	<Anforderungstext >

INFO

Die aus den anderen Dokumenten übernommenen Anforderungen entsprechen dem Stand zum Zeitpunkt der Veröffentlichung dieses Leitfadens. Bei Aufnahme der Tätigkeit als Softwarelieferant sollten diese mit dem dann jeweils aktuellen Stand der referenzierten Dokumente auf der DVC-Webseite abgeglichen werden.

3.1 Phase 1: Planungsphase

3.1.1 Finden eines Softwarebetreibers

Sie wollen Software in die DVC einbringen? Dann benötigen Sie zunächst einen Softwarebetreiber, der Ihre Software im Rahmen eines neu zu schaffenden Cloud-Services in der Deutschen Verwaltungscloud anbieten möchte. Beim Softwarebetreiber handelt es sich um eine eigene Rolle gemäß Rollenkonzept DVC (Rahmenwerk, Kap 3.2), die wie folgt definiert ist: "Softwarebetreiber sind Cloud-Service-Anbieter, die Cloud-Services im SaaS-Modell bereitstellen".

Als verwaltungsinterner Anbieter können Sie zusätzlich zur Rolle Softwarelieferant auch die Rolle eines Softwarebetreibers selbst einnehmen.

Als verwaltungsexterner Softwarelieferant müssen Sie sich an dieser Stelle aber einen oder mehrere Softwarebetreiber aus der ÖV suchen. Im FAQ-Bereich der Webseite der Deutschen Verwaltungscloud (Abschnitt: "Wie kann ich meine Software in der Deutschen Verwaltungscloud anbieten?") stellt die Koordinierungsstelle allgemeine Informationen hinsichtlich Kontaktdaten für die Suche nach Softwarebetreibern bereit.

3.1.2 Vertragsschluss mit einem Softwarebetreiber

Ergebnis der Phase 1 ist eine vertragliche Vereinbarung zwischen Softwarebetreiber und Softwarelieferant, auf dessen Basis dann die Entwicklung der Software (Phase 2) und die Inbetriebnahme (Phase 3) erfolgen kann.

Sofern die Software zum Zeitpunkt der ersten Kontaktaufnahme zwischen Softwarelieferant und DVC schon existiert, so ist dies kein Hinderungsgrund für eine DVC-Aufnahme.

Der Softwarebetreiber wird prüfen, ob das Entwickeln eines Cloud-Services auf Basis der von Ihnen angebotenen Software für ihn möglich ist. Dabei kann z.B. betrachtet werden, ob der neu zu entwickelnde Cloud-Service einen Mehrwert für die Kunden der Deutschen Verwaltungscloud bietet, wie wirtschaftlich erfolgreich der Service bewertet wird, ob Entwicklungskapazitäten vorhanden sind und ob eine Finanzierung der Serviceentwicklung sichergestellt werden kann.

Der Softwarebetreiber wird dann in der Regel eine Markterkundung und eine Ausschreibung durchführen, um eine Software und ggf. den Support für die Software vergabekonform zu beziehen.

Sollte die von Ihnen angebotene Software und/oder der von Ihnen angebotene Support für die Software die Ausschreibung gewinnen, so kann es zum Vertragsschluss zwischen Ihnen (als Softwarelieferant) und dem Softwarebetreiber kommen.

3.1.3 Formale Anforderungen an die Software

Die gesamte Architektur der DVC ist Cloud-basiert. Somit gibt es auch die Empfehlung an die Softwarelieferanten, ihre für die DVC bestimmte Software ebenfalls als cloudbasierte Software zu entwickeln. Dies ist aber keine Muss-Bedingung: auch proprietär entwickelte Software kann durch einen Softwarelieferanten in der DVC bereitgestellt werden. In dieser Konstellation kommt dann der Abstimmung zwischen Softwarebetreiber und Softwarelieferant bei der Erstellung eines Services für die proprietäre Software eine noch größere Bedeutung zu.

3.1.3.1 Beschreibung der Service Level Agreements

Die Service Level Agreements (SLA) werden für die Services erstellt. Damit liegt deren Beschreibung im Verantwortungsbereich des Softwarebetreibers. Dieser wird sich aber bezüglich dieser Beschreibung mit dem Softwarelieferanten abstimmen.

3.1.3.2 Hoheit über Krypto-Module und -Schlüssel

Die folgenden Vorgaben für Softwarelieferanten bezüglich Krypto-Modulen und Schlüsseln sind dem Detailstandard 40 entnommen.

ID	Rolle	Modal-verb	Detailanforderung
DS_40_A002	Software-lieferant	MUSS	als Vorgabe für den Einsatz von Krypto-Technologien und Tools TR02102 beachten.
DS_40_A003	Software-lieferant	MUSS	sicherstellen, dass der Softwarebetreiber alle mit Krypto-Modulen verbundenen Aufgaben im operativen Tagesgeschäft durchführen kann. Die Hoheit über alle benötigten Schlüssel liegt beim Softwarebetreiber.
DS_40_A004	Software-lieferant	MUSS	sicherstellen, dass alle benötigten Informationen bezüglich Zertifikaten, Schlüsseln etc. in der Dokumentation des Softwarelieferanten enthalten sind.

3.1.3.3 Nutzungsrechte und Hoheit über Software

Diese Themen sind Bestandteil der Verhandlungen und vertraglichen Vereinbarungen zwischen Softwarelieferant und Softwarebetreiber.

3.1.3.4 Tools und Ressourcen

Die DVC macht keine verpflichtenden Vorgaben oder Verbote zu einzelnen für die Softwareentwicklung notwendigen Werkzeugen.

3.1.4 Zertifizierung von Softwarelieferanten

Ein Softwarelieferant muss keine Zertifizierung vorlegen, um in der DVC mitwirken zu können. Es kann aber sein, dass er vom Softwarebetreiber zum Einhalten von Kriterien des DVC Reifegradmodells sowie individuellen Zusicherungen/Vereinbarungen/SLAs verpflichtet wird. Der vorliegende Leitfaden enthält Vorgaben, die wiederum Bedingungen des BSI-Grundschutzes referenzieren. Softwarelieferanten wird empfohlen, bereits bei der Erstellung der Software auf die Einhaltung dieser Bedingungen zu achten und damit die Abstimmung und Vertragsverhandlungen mit dem Softwarebetreiber zu vereinfachen.

Aus den Mindestkriterien des Reifegradmodells der DVC leiten sich jedoch für den Softwarelieferanten folgende Anforderungen ab. Diese sind zu berücksichtigen, damit der zukünftige Softwarebetreiber die Mindestkriterien für den zukünftigen Services erfüllen kann.

ID	Rolle	Modal-verb	Detailanforderung
RG_A6	Software-lieferant	SOLL	aus "Bereich Funktionelles: Mandantentrennung": - Der Service unterstützt optimalerweise eine nach aktuellem Stand der Technik sichere Art der Mandantentrennung
RG_A12	Software-lieferant	SOLL	aus "Bereich Service&Support: Benutzerdokumentation": - Eine Benutzer-Dokumentation für alle wesentlichen Funktionen ist vorhanden und für Kunden des CSP online verfügbar
RG_A16	Software-lieferant	MUSS	aus "Bereich Informationssicherheit & Datenschutz: Transport-Verschlüsselung": - Alle Verbindungen zum Service sind gemäß BSI TR-02102 transportverschlüsselt.
RG_A18	Software-lieferant	SOLL	aus "Bereich Informationssicherheit & Datenschutz: Authentisierung": - Der Service besitzt ein Authentisierungsverfahren.

Fortsetzung auf nächster Seite

Fortsetzung von vorheriger Seite

ID	Rolle	Modal-verb	Detailanforderung
RG_A19	Software-lieferant	KANN	aus "Bereich Informationssicherheit & Datenschutz: Autorisierung": - Der Service besitzt ein Autorisierungsverfahren

3.2 Phase 2: Umsetzungsphase

Die folgenden Punkte dienen als organisatorischer Rahmen für die Entwicklung und Auslieferung der Software durch den Softwarelieferanten. Die DVC macht keine Vorschriften dazu, welche Entwicklungsumgebungen, Tools etc. der Softwarelieferant verwenden muss, sondern nur welche Kriterien das Ergebnis der Software-Entwicklung erfüllen muss.

3.2.1 Vorgaben für Interaktion zwischen Softwarebetreiber und Softwarelieferant

Für viele Aufgaben, die in den Zuständigkeitsbereich des Softwarebetreibers gehören, ist der Softwarebetreiber auf Vorleistungen von sowie Absprachen mit dem Softwarelieferanten angewiesen. Die nachfolgend beschriebenen Punkte muss der Softwarelieferant bereits bei der Erstellung der Software beachten und damit die Voraussetzungen für die Interaktion mit dem Softwarebetreiber schaffen.

3.2.1.1 Detailstandards der CI/CD-Pipeline

Die folgenden Vorgaben für Softwarelieferanten bezüglich der CI/CD-Pipeline sind dem Detailstandard 10 entnommen.

3.2.1.1.1 Verwaltung der Codebase

Zur Sicherstellung der Nachvollziehbarkeit von Codeänderungen werden folgende Vorgaben definiert:

ID	Rolle	Modal-verb	Detailanforderung
WP_SD_5.1_1	Software-lieferant	SOLL	die Source-Files und Configuration-Files in einem Git-Repository speichern.
WP_SD_5.1_2	Software-lieferant	SOLL	seine Codebase in einem Versionsmanagementsystem verwalten.
WP_SD_5.1_3	Software-lieferant	SOLL	Es soll immer eine eindeutige Korrelation zwischen einer Codebase und einer App bestehen. Jede Änderung ist zu versionieren.
WP_SD_5.1_4	Software-lieferant	MUSS	Das Staging muss in den Repositories berücksichtigt werden. Es muss für jede Stage die zugehörige Codebase bekannt sein.
WP_SD_5.1_5	Software-lieferant	MUSS	Für die Verwaltung des Codes ist ein geeigneter Workflow zu wählen und die Dokumentation für die Nutzenden zugänglich zu machen.
WP_SD_5.1_6	Software-lieferant	MUSS	Es müssen Git-Regeln wie z.B - Code und Konfigurationen trennen - Entwicklungs- und Produktion in unterschiedliche Directories trennen - keine Branches für unterschiedliche Environments verwenden festgelegt und durchgesetzt werden.
DS_10_A001	Software-lieferant	MUSS	für die Unveränderlichkeit seiner Releases Sorge tragen. Hierfür muss ein Softwarelieferant ein Code-Repository einsetzen und Releases hierin in einer unveränderlichen Art und Weise ablegen. Für das Code-Repository muss nach BSI-Vorgaben (OPS.1.1.2.A26) eine Backupstrategie eingerichtet werden.

3.2.1.1.2 Qualitätssicherung der Sourcen

Im Rahmen eines automatischen Deployments ist die Qualitätssicherung integraler Bestandteil des Deploymentprozesses. Aus diesem Grund werden folgende Vorgaben empfohlen:

ID	Rolle	Modal-verb	Detailanforderung
WP_SD_5.2_1	Software-lieferant	MUSS	Für die (automatisierte) Qualitätssicherung müssen Festlegungen im Projekt getroffen werden.
WP_SD_5.2_2	Software-lieferant	SOLL	Die Prüfung der Sourcen (Sicherheit und Qualität) sollte automatisch erfolgen.
WP_SD_5.2_3	Software-lieferant	SOLL	Das Einchecken und Mergen in Hauptbranches sollte nur bei erfolgreicher Prüfung nach der Ausführung von Unittests abgeschlossen werden können.
WP_SD_5.2_4	Software-lieferant	MUSS	Quelltext und Konfigurationsdaten müssen immer getrennt werden. Eine Anpassung der Konfiguration soll ausschließlich über Konfigurationsobjekte/Umgebungsvariablen usw. erfolgen.
WP_SD_5.2_5	Software-lieferant	MUSS	Es dürfen keine Credentials, private Schlüssel, Secrets usw. unverschlüsselt im Repository abgelegt werden.
WP_SD_5.2_6	Software-lieferant	MUSS	Das Deployment muss so gestaltet werden, dass die Nutzung der abhängigen Dienste individuell konfiguriert werden kann
DS_10_A002	Software-lieferant	MUSS	dafür Sorge tragen, dass im Rahmen einer automatischen Bereitstellung die Qualitätssicherung integraler Bestandteil des Bereitstellungsprozesses ist. Innerhalb der Qualitätssicherung ist der Code via Tests und Statischer-Code-Analyse zu prüfen 0und ein SBOM zu generieren. Das generierte SBOM ist bei der Bereitstellung des Artefakts mit anzuhängen.

Folgende Punkte sollten explizit betrachtet werden:

ID	Rolle	Modal-verb	Detailanforderung
WP_SD_5.2_7	Software-lieferant	SOLL	Alle Abhängigkeiten sollen explizit deklariert und isoliert werden.
WP_SD_5.2_8	Software-lieferant	SOLL	Die Prüfung auf Schwachstellen sollte bereits zu einem frühen Zeitpunkt erfolgen.
WP_SD_5.2_9	Software-lieferant	SOLL	Es soll sichergestellt werden, dass SBOMs (Open Source-Tool: Syft = CLI bzw. Go Library) erzeugt und verwendet werden.
WP_SD_5.2_10	Software-lieferant	MUSS	Im Source-Code dürfen keine ungenutzten Bibliotheken referenziert werden.

3.2.1.1.3 Build-Erstellung

Die Build-Erstellung ist die Grundlage für jede Softwarelösung. Daher werden folgende Vorgaben definiert:

ID	Rolle	Modal-verb	Detailanforderung
WP_SD_5.3_1	Software-lieferant	MUSS	Es muss eine Trennung in Build-, Release- und Run-Phase erfolgen (siehe auch The Twelve-Factor App).
WP_SD_5.3_2	Software-lieferant	SOLL	Es soll eine Phase Test zwischen Build und Release verwendet werden.
WP_SD_5.3_3	Software-lieferant	MUSS	Jeder Build muss eine eindeutige ID haben.
WP_SD_5.3_4	Software-lieferant	MUSS	Eine Veränderung des Containers zur Laufzeit ist nicht zulässig. Alle Änderungen müssen über die Versionsverwaltung dokumentiert werden.
WP_SD_5.3_5	Software-lieferant	SOLL	Jede Änderung erzeugt eine neue Release. Jedes Release sollte eine eindeutige Release-ID und eine Versionsnummer haben.
WP_SD_5.3_6	Software-lieferant	MUSS	Jedes Release muss eindeutig auf die benutzten Komponentenversionen zurückgeführt werden können.
WP_SD_5.3_7	Software-lieferant	SOLL	Für die Versionsnummer soll Semantic Versioning 2.0 genutzt werden
WP_SD_5.3_8	Software-lieferant	SOLL	Es sollen Build-Container („Builder“) für die jeweilige Zielumgebung (Production, Development) verwendet werden. „Builder“ sind temporäre Container, die als Grundlage für den Ziel-Container verwendet werden.
WP_SD_5.3_9	Software-lieferant	SOLL	Es soll ein Set an erlaubten Tools für Produktions-Container definiert werden.
WP_SD_5.3_10	Software-lieferant	MUSS	Ein Build muss vollautomatisch erstellt werden, so dass keine Eingriffe in den Build-Prozess erfolgen.
WP_SD_5.3_11	Software-lieferant	SOLL	Es sollen immer die aktuellsten Versionen von Base Images verwendet werden (das sollte durch automatisches Anstoßen der CI/CD-Pipeline erfolgen).
WP_SD_5.3_12	Software-lieferant	MUSS	Ein vorgegebener Satz an Meta-Daten (Label von Images) je Build-Vorgang muss mindestens definiert sein (Versionierung, Ersteller, Repository)
WP_SD_5.3_13	Software-lieferant	SOLL	Für die eindeutige Build-Identifikation soll die Verwendung von Sub-Modulen, Branching oder Tagging erfolgen.
WP_SD_5.3_14	Software-lieferant	SOLL	Es sollen Regelsätze für das Tagging verwendet werden (Informationen zu enthaltenen Komponenten sind in den Metadaten zu speichern)
WP_SD_5.3_15	Software-lieferant	SOLL	Alle (externen) Abhängigkeiten sollten explizit deklariert und isoliert sein (z.B. DB-Image).
WP_SD_5.3_16	Software-lieferant	SOLL	Alle unterstützenden Dienste sollen als Ressource lose gekoppelt werden (z.B. Message Broker).
WP_SD_5.3_17	Software-lieferant	SOLL	Die Build-Erstellung erfolgt vollautomatisch.
DS_10_A003	Software-lieferant	MUSS	Builds werden in separate Phasen unterteilt, mit einer Testphase vor dem Release.
DS_10_A004	Software-lieferant	MUSS	Jeder Build erhält eine eindeutige Kennung und Änderungen werden dokumentiert.
DS_10_A005	Software-lieferant	MUSS	Bei jedem Release wird eine unveränderbare Version erstellt.

Fortsetzung auf nächster Seite

Fortsetzung von vorheriger Seite

ID	Rolle	Modal-verb	Detailanforderung
DS_10_A006	Software-lieferant	MUSS	Die Build-Erstellung erfolgt vollautomatisch.

3.2.1.1.4 Release und Deployment

Für die Veröffentlichung von Releases und deren Installation werden folgende Maßnahmen für Softwarelieferanten definiert:

ID	Rolle	Modal-verb	Detailanforderung
WP_SD_5.4_1	Software-lieferant	SOLL	Es sollen signierte Container (trusted CA) verwendet werden (Sicherstellung, dass das Image von einer vertrauenswürdigen Quelle bezogen wurde).
WP_SD_5.4_2	Software-lieferant	SOLL	Es sollen Build-Regelsätze für das jeweilige Deployment angewendet werden (Best Practices für Build).
WP_SD_5.4_3	Software-lieferant	MUSS	Alle Konfigurationsinformationen für Applikationen müssen außerhalb des Containers gespeichert werden
WP_SD_5.4_4	Software-lieferant	SOLL	Es sollen Kubernetes Config-Maps und Kubernetes Secrets (in verschlüsselter Form) verwendet werden.
WP_SD_5.4_5	Software-lieferant	MUSS	Alle Pod-Definitionen müssen Memory- und CPU-Anforderungen enthalten.
WP_SD_5.4_6	Software-lieferant	MUSS	Readiness- und Liveness-Probes müssen in den Definitionen (z.B. bei Deployments oder StatefulSets) enthalten sein.
WP_SD_5.4_7	Software-lieferant	SOLL	Es sollten PodDisruptionBudgets konfiguriert sein, wenn es zwingend erforderlich ist. Single Pod Disruption Budgets sind zu vermeiden.
WP_SD_5.4_8	Software-lieferant	MUSS	Node Affinity Rules sind auf das zwingend erforderliche Maß zu beschränken (z.B. bei Hardware-Abhängigkeiten).
WP_SD_5.4_9	Software-lieferant	SOLL	Alle Pods sollten terminierbar sein, ohne die Stabilität der Applikation zu gefährden. Dabei ist auch auf Session Handling zu achten.
WP_SD_5.4_10	Software-lieferant	SOLL	Die Auto-Scaling Mechanismen sollen - wenn auf der Plattform verfügbar - genutzt werden können.
WP_SD_5.4_11	Software-lieferant	SOLL	Die Deployments sollen auf Basis des Standard-Update (einzeln je Pod), Blue/Green-oder Canary-Modells möglich sein.
WP_SD_5.4_12	Software-lieferant	SOLL	Fehlgeschlagene Deployments sollten über ein Rollback wieder ersetzt werden können.
WP_SD_5.4_13	Software-lieferant	SOLL	Es sollen in einem Container immer nur 1 Dienst und nicht mehrere Dienste (wie z.B. DB und AppServer) laufen.
WP_SD_5.4_14	Software-lieferant	SOLL	Die Application-Logs sollen auf stdout/stderr geschrieben werden, so dass die Zielumgebung für Logs frei gewählt werden kann.
WP_SD_5.4_15	Software-lieferant	SOLL	Es sollen Rate-Limits, Connection-Timeouts und Retries verwendet werden.
WP_SD_5.4_16	Software-lieferant	SOLL	Es soll immer zwischen Build- und Runtime-Images unterschieden werden.
WP_SD_5.4_17	Software-lieferant	MUSS	Pods müssen in einem eingeschränkten Security Context laufen (non-root), Least Privilege Prinzip.

Fortsetzung auf nächster Seite

Fortsetzung von vorheriger Seite

ID	Rolle	Modal-verb	Detailanforderung
WP_SD_5.4_18	Software-lieferant	SOLL	Für die Kommunikation zwischen Pods soll eine Verschlüsselung gemäß BSI Anforderungen (TR-02102) verwendet werden.
WP_SD_5.4_19	Software-lieferant	SOLL	Es soll ein Zero Trust-Modell verwendet werden und die Kommunikation ausgehend von „deny all“ durch Network-Policies explizit definiert werden.
WP_SD_5.4_20	Software-lieferant	SOLL	Es müssen Rolling-Updates möglich sein.
WP_SD_5.4_21	Software-lieferant	MUSS	Die Updates an Datenstrukturen müssen automatisiert möglich sein. Eine Abwärtskompatibilität der Datenstrukturen (Additive Updates) muss über die laufenden Pods während der Updates sichergestellt sein.
WP_SD_5.4_22	Software-lieferant	SOLL	Updates sollen ohne Betriebsunterbrechung möglich sein. Es bieten sich Microservice-Technologien an.
WP_SD_5.4_23	Software-lieferant	MUSS	Die maximale Anzahl nicht verfügbarer Pods muss definiert werden und darf nicht unterschritten werden.
WP_SD_5.4_24	Software-lieferant	MUSS	Die maximal zulässige Anzahl von Pods und Ressourcen muss so ausgelegt werden, dass Rolling-Updates ausgeführt werden können.

3.2.1.1.5 Technische Vorgaben

Zur Vereinheitlichung der Umgebungen werden folgende technische Vorgaben gemacht:

ID	Rolle	Modal-verb	Detailanforderung
WP_SD_5.5_1	Software-lieferant	MUSS	Alle Dienste müssen zustandslos implementiert werden.
WP_SD_5.5_2	Software-lieferant	MUSS	Alle Daten müssen in unterstützenden Diensten gespeichert werden, normalerweise einer Datenbank oder persistent Storage.
WP_SD_5.5_3	Software-lieferant	SOLL	Es sollen keine lokalen Speicher der Workernodes benutzt werden, ausgenommen Ephemeral Storage via EmptyDir Mounts.
WP_SD_5.5_4	Software-lieferant	SOLL	Dateien sollen so gespeichert werden, dass diese ggf. durch andere Container nachgenutzt werden können, wenn der speichernde Container neu gestartet wird.
WP_SD_5.5_5	Software-lieferant	MUSS	Sticky Sessions sind nicht zulässig. Dass Session-Handling muss so implementiert werden, dass die Weitergabe der Sessions zwischen den Pods, z. B. über persistente Speicher, realisiert wird.
WP_SD_5.5_6	Software-lieferant	MUSS	Web- und Applikationsdienste bringen einen eigenen Web- oder Applikationsserver im Container mit.
WP_SD_5.5_7	Software-lieferant	MUSS	Die Kommunikation zwischen den Diensten erfolgt über dokumentierte Ports und werden über Services (Cluster / Namespace intern) oder Ingress Routen (Cluster extern) implementiert.
WP_SD_5.5_8	Software-lieferant	MUSS	Die eingehende Kommunikation im Cluster beginnt am Ingress-Controller.
WP_SD_5.5_9	Software-lieferant	SOLL	Pro Container soll nur ein Dienst laufen.

Fortsetzung auf nächster Seite

Fortsetzung von vorheriger Seite

ID	Rolle	Modal-verb	Detailanforderung
WP_SD_5.5_10	Software-lieferant	SOLL	Dienste gleicher Art sollen gruppiert werden, damit eine horizontale Skalierung unterstützt wird. Das Skalieren soll auf Basis von Prozessen statt auf Basis von Threads realisiert werden.
WP_SD_5.5_11	Software-lieferant	SOLL	Ein automatisches horizontales Skalieren der Anwendung soll unterstützt werden.
WP_SD_5.5_12	Software-lieferant	SOLL	Health-Checks und Monitoring-Punkte in Containern sollen so implementiert werden, dass die automatischen Steuerungsmechanismen der Containerplattform genutzt werden können.
WP_SD_5.5_13	Software-lieferant	MUSS	Die Grenzen für die Instanziierung der Container müssen implementiert werden, z. B. limits und quotas, cpu, memory, anzahl.
WP_SD_5.5_14	Software-lieferant	SOLL	Die Prozesse sollen mittels asynchroner Eventmodelle kommunizieren, um Deadlocks zu verhindern und die Reaktionen auf Benutzereingaben sicherzustellen.
WP_SD_5.5_15	Software-lieferant	SOLL	Das Verhalten der Skalierung sollte immer überwacht werden, um Anomalien zu ermitteln.
WP_SD_5.5_16	Software-lieferant	MUSS	Zum Schutz der Plattform und anderer Cluster-Nutzer müssen klare Vorgaben für die Ressourcennutzung definiert sein und ein Monitoring- und Alertsystem etabliert werden.
WP_SD_5.5_17	Software-lieferant	MUSS	Die Protokollinformationen werden über stdout und stderr ausgegeben (siehe dazu: Logging Architecture).
WP_SD_5.5_18	Software-lieferant	MUSS	Eine benötigte Revisionssicherheit der Protokollierung wird außerhalb der Containerumgebung sichergestellt.
WP_SD_5.5_19	Software-lieferant	SOLL	Softwarebetreiber und Softwareentwickler sollen auf die Protokollierungslösung im Cluster zugreifen können.
WP_SD_5.5_20	Software-lieferant	MUSS	Softwarebetreiber und Softwareentwickler dürfen im regulären Betrieb keine Benutzerinteraktionen über stdout und stderr ausgegeben. Die Protokollierung zur Nachvollziehbarkeit in der Anwendung soll in der Anwendung erfolgen, z. B. in der Datenbank.
WP_SD_5.5_21	Software-lieferant	SOLL	Für die Protokollierung soll der syslog-Standard beziehungsweise standardisierte JSON Formate der Logging Systeme (Splunk, Elastic, Loki) verwendet werden, damit eine einheitliche Auswertbarkeit der Daten erreicht werden kann.
WP_SD_5.5_22	Software-lieferant	SOLL	Debuginformationen sollen im produktiven Betrieb ausschließlich im Rahmen des Problemmanagements zur Fehlersuche zeitlich begrenzt ausgegeben werden. (siehe dazu Logging Architecture)

3.2.1.2 Schwachstellenscan von Softwarelösungen

Die folgenden Vorgaben für Softwarelieferanten bezüglich Schwachstellenscans von Softwarelösungen sind dem Detail-standard 17 entnommen.

Aus den BSI Grundschutz Bausteinen OPS.1.1.1.A10 Führen eines Schwachstelleninventars und SYS.1.6.A6 Verwendung sicherer Images ergibt sich, dass Softwarelieferanten und Softwarebetreiber Schwachstellenscans durchführen müssen bzw. sollen. Die Ergebnisse müssen zentral dokumentiert werden. Der Detailstandard 17 beschreibt ein am IT-Grundschutz ausgerichtetes, einheitliches Modell zum Umgang mit veröffentlichten Schwachstellen in allen Cloud Standorten.

Aus den Vorgaben ergeben sich folgende Detailanforderungen:

ID	Rolle	Modal-verb	Detailanforderung
DS_17_A001	Software-lieferant	SOLL	seine gelieferte Software regelmäßig auf Schwachstellen prüfen und die Ergebnisse dokumentieren.
DS_17_A002	Software-lieferant	SOLL	die Schwachstellenscans in seine Continuous Integration Pipeline einbauen
DS_17_A003	Software-lieferant	MUSS	bei gefundenen und nicht bereits abgestimmten Schwachstellen eine Abstimmung mit ihm bekannten Softwarebetreibern initiieren mit Fokus auf einem CVSS von 7.0 oder höher vor dem Hintergrund von via BSI als "schwerwiegend" eingestuften Scorewerten und damit etwaiger Auslösung von Security Incidents

3.2.1.3 Vorgaben für die Lieferung oder Bereitstellung von containerisierten Softwareprodukten

Die folgenden Vorgaben für Softwarelieferanten bezüglich der Lieferung und Bereitstellung von containerisierten Softwareprodukten sind den Detailstandards 41 und 51 entnommen.

ID	Rolle	Modal-verb	Detailanforderung
WP_EC_6_1	Software-lieferant	MUSS	eine vollständige Software Bill of Materials (SBOM) in einem abgestimmten Format abgeben. (SYS.1.6.A13 S) Umsetzungshinweise: Dokumentation der genutzten Komponenten, Ermöglichung der Prüfung von Lizenzen.
WP_EC_6_2	Software-lieferant	MUSS	einen definierten Prozess für den Bezug und die Weitergabe von Images etablieren und nachvollziehbar dokumentieren. (SYS.1.6.A4 B) Umsetzungshinweise: Der Softwarebetreiber muss seinen Prozess entsprechend anpassen können. Die Nachvollziehbarkeit ist Grundlage für die Prüfung der gelieferten Software.
WP_EC_6_3	Software-lieferant	MUSS	einen standardkonformen Übergabepunkt zum Softwarebetreiber verwenden. (SYS.1.6.A4 B) Umsetzungshinweise: Bereitstellung in einer Registry nach OCI-Registry-Format.
WP_EC_6_4	Software-lieferant	MUSS	die unterschiedlichen Schutzbedarfe bei der Bereitstellung der Lösung unterstützen (APP.4.4.A2 B). Umsetzungshinweise: Gerade beim Test muss die strikte Trennung (Schutzbedarf und Mandant) umgesetzt werden, da hier Code ausgeführt wird.
WP_EC_6_5	Software-lieferant	SOLL	Deployment Manifeste zur Installation bereitstellen (SYS.1.6.A9 S).
WP_EC_6_6	Software-lieferant	SOLL	sicherstellen, dass zusammenhängende Dienste durch das Deployment bzw. die Orchestrierung als Verwaltungseinheiten bereitgestellt werden können und z.B. geeignete Deployment-Artefakte (Manifeste) liefern. (SYS.1.6.A11 S)
WP_EC_6_7	Software-lieferant	SOLL	für die durch ihn bereitgestellten Software-Artefakte automatisiert zu verarbeitende Deployment-Anweisungen bereitstellen (z.B. YAML-basierte Deployments). (SYS.1.6.A11 S)

Fortsetzung auf nächster Seite

Fortsetzung von vorheriger Seite

ID	Rolle	Modal-verb	Detailanforderung
WP_EC_6_8	Software-lieferant	SOLL	Build-Spezifikationen (bspw. Dockerfile) für Images liefern. (SYS.1.6.A13 S) Umsetzungshinweise: Dient der Ermöglichung der statischen Analyse auf Schwachstellen und Schadsoftware.
WP_EC_6_9	Software-lieferant	SOLL	neben einem Freigabeprozess für die erstellte Software (siehe dazu auch Baustein OPS.1.1.6 Software-Test und Freigaben) auch einen Freigabeprozess für Images und Deployment Deskriptoren umsetzen. (SYS.1.6.A13 S)
WP_EC_6_10	Software-lieferant	SOLL	die Freigabebedingungen in einem definierten Freigabeprozess im Lieferantenkontext prüfen und alle Informationen für den weiteren Freigabeprozess an den Softwarebetreiber übergeben. Zu den übergebenen Informationen gehören auch die Ergebnisse des eigenen Freigabeprozesses. (SYS.1.6.A13 S)
WP_EC_6_11	Software-lieferant	KANN	die zu liefernden Images bereits vor der finalen Lieferung mit dem Prüfstandard des Softwarebetreibers überprüfen. (SYS.1.6.A13 S) Umsetzungshinweise: Das ermöglicht dem Lieferanten, seine eigene Software qualitätszusichern und nicht vom Prüfergebnis des Betreibers "überrascht" zu werden. Das kann z.B. durch eine standardisierte Entwicklungsumgebung realisiert werden, die dem Lieferanten bereitgestellt wird.

3.2.1.3.1 Art und Weise der Lieferung/Bereitstellung

ID	Rolle	Modal-verb	Detailanforderung
DS_41_A001	Software-lieferant	MUSS	- mit dem Softwarebetreiber einen Prozess für die Lieferung von Container Images etablieren. - sich hierbei an der Schnittstelle authentifizieren - Die Übertragung MUSS verschlüsselt sein. - hierbei die Möglichkeit haben, die Container sowie die geforderten Dokumente über diese Schnittstelle zu liefern.
DS_41_A003	Software-lieferant	MUSS	bei jeder Bereitstellung zusätzlich zu den Containern folgende Dokumente liefern: - Software Bill of Material (SBOM) - Dokumentation der Container und deren mögliche unterstützte Konfigurationsoptionen - Dokumentation der im Container enthaltenen Software - Dokumentation eines "Geeigneten Testmanagements" bzw. "QA-Prozesses" für die bereitgestellte Software, um sicherzustellen, dass keine Komponenten ungetestet in die DVC eingeführt werden über die gesamte Lieferkette hinweg. Dies KANN über einen Nachweis entsprechender Zertifikate wie ISO:27001 durch den Softwarelieferanten erfolgen oder wird sonst im Einzelfall zwischen Softwarelieferant und Softwarebetreiber vereinbart. - Changelog - Optional: Deployment Manifeste, Dockerfile

Fortsetzung auf nächster Seite

Fortsetzung von vorheriger Seite

ID	Rolle	Modal-verb	Detailanforderung
DS_51_A004	Software-lieferant	MUSS	sich bezüglich DS_51_A003 an der Schnittstelle authentifizieren und die Übertragung MUSS geeignet verschlüsselt sein (erzwungen durch OCI-konforme Registry)
DS_51_A005	Software-lieferant	MUSS	bezüglich DS_51_A003 die Möglichkeit haben die Kubernetes Artefakte sowie die geforderten Dokumente über diese Schnittstelle zu liefern.
DS_51_A006	Software-lieferant	MUSS	das Artefakt versioniert übergeben.
DS_51_A007	Software-lieferant	MUSS	alle im Kubernetes Artefakt genutzten Images wie in DVC Detailstandards - (41) Vorgaben für die Lieferung oder Bereitstellung von containerisierten Softwareprodukten beschrieben bereitstellen
DS_51_A008	Software-lieferant	MUSS	gewährleisten, dass keine weiteren externen Artefakte während des Betriebs des Kubernetes Artefaktes angezogen werden. Der Softwarelieferant muss alle benötigten Artefakte innerhalb der gelieferten Images bereitstellen.
DS_51_A010	Software-lieferant	MUSS	bei jeder Bereitstellung zusätzlich zu den Kubernetes Artefakten folgende Dokumente liefern: Handbuch in deutscher Sprache § 23 VwVfG - Einzelnorm: - Dokumentation der Konfigurationsoptionen - Herstellername - Supportinformation - Infrastruktur Diagramm, um einen schnellen Überblick über die Ressourcen zu geben. Dies hilft insbesondere bei komplexen Architekturen. Changelog

3.2.1.3.2 Prüfung der gelieferten Artefakte

ID	Rolle	Modal-verb	Detailanforderung
DS_41_A007	Software-lieferant	MUSS	aussagefähig sein, wie die Software kontinuierlich im Entwicklungsprozess und bezogen auf übergebene Artefakte einer Quellcode-Kontrolle unterworfen ist

3.2.1.3.3 Kubernetes Ressourcen

ID	Rolle	Modal-verb	Detailanforderung
DS_51_A002	Software-lieferant	MUSS	für den Softwarebetreiber in seinen gelieferten Kubernetes Ressourcen die Möglichkeit einbauen genutzte Images sowie Annotationen zu überschreiben, z.B. Helm: Pods and PodTemplates

3.2.1.4 Deployment von Softwarelösungen

Der Detailstandard 16 enthält Vorgaben bezüglich Deployment von Softwarelösungen. Sie betreffen zu großen Teilen Kernaufgaben des Softwarebetreibers, zu denen ein Softwarelieferant Zuarbeiten (z.B. Dokumentation) im Rahmen der Interaktion zwischen Softwarebetreiber und Softwarelieferant zu leisten hat. Die folgenden Vorgaben für die Rolle Softwarebetreiber sind daher hier im Leitfaden für Softwarelieferanten ausschließlich zu Informationszwecken aufgenommen worden.

Vorgaben für das Szenario "Der Softwarebetreiber erlangt Kenntnisse über eine neue Softwareversion":

ID	Rolle	Modal-verb	Detailanforderung
DS_16_A003	Software-betreiber	MUSS	den Schutzbedarf der Daten für die eingesetzte Software kennen.
DS_16_A004	Software-betreiber	MUSS	die Infrastrukturanforderungen kennen.
DS_16_A005	Software-betreiber	MUSS	die Ressourcenanforderungen kennen.
DS_16_A006	Software-betreiber	SOLL	sicherstellen, dass Detailstandard 40: "Vorgaben für die Erstellung von containerisierten Softwareprodukten" und Detailstandard 41 "Vorgaben für die Lieferung oder Bereitstellung von containerisierten Softwareprodukten" eingehalten werden.

Vorgaben für das Szenario "Der Softwarebetreiber plant die Inbetriebnahme der Software":

ID	Rolle	Modal-verb	Detailanforderung
DS_16_A007	Software-betreiber	SOLL	einen Bauplan (IaC) für das Rollout der Software erstellen. Hierbei MÜSSEN die Vorgaben des Plattformbetreibers und des BSI Grundschutzes berücksichtigt werden.
DS_16_A008	Software-betreiber	MUSS	den Plattformbetreibern bei Bedarf eine Architekturskizze im Sinne von a) einer vollständige Konfigurations-Anforderung an die genutzten Infrastrukturkomponenten sowie b) einem vollständigen Netzkommunikationsplan (intern und extern) des Software-Service (Protokolle, Ports, Sourcen, Targets) bereitstellen.
DS_16_A009	Software-betreiber	SOLL	den Bauplan (IaC) dem Plattformbetreiber zur Prüfung - ggf. durch Automatisierung - zur Verfügung stellen.

Vorgaben für das Szenario "Der Softwarebetreiber ist für das Deployment der übergebenen Softwareversion verantwortlich":

ID	Rolle	Modal-verb	Detailanforderung
DS_16_A010	Software-betreiber	MUSS	die Verantwortung für das Deployment der übergebenen Softwareversion übernehmen. Dabei muss der BSI Grundschutz zugrundeliegen und die Vorgaben des Plattformbetreibers eingehalten werden. Die Gestaltung des Rollouts obliegt dem Softwarebetreiber.

3.2.2 Vorgaben für die Erstellung von Software

Dieser Abschnitt beschreibt die Anforderungen im Bereich "Erstellung von containerisierten Softwareprodukten". Sie sind den Detailstandards 40 und 44 entnommen.

3.2.2.1 Implementierung/Containererstellung

3.2.2.1.1 Datenspeicherung

ID	Rolle	Modal-verb	Detailanforderung
WP_EC_4.1.1_1	Software-lieferant	MUSS	die Nutzung von temporärem Speicher so vorsehen, dass ein Restart des Containers und ein Wechsel des Hosts zur Laufzeit möglich ist. (SYS.1.6.A9 S)
WP_EC_4.1.1_2	Software-lieferant	MUSS	notwendige persistente Volumen und deren Nutzung (z. B. RWM, RWO, RW/RO) im Deployment Descriptor beschreiben. (SYS.1.6.A19 S) Umsetzungshinweise: z.B. Zielpfad im Containerbetrieb, Größe, Berechtigungen, Dateisystem, Performance. Die Beschreibung sollte in einer formalen Sprache erfolgen. Das Physical Volume Claim soll dynamisch erfolgen. Bitte mögliche Einschränkungen im Rechenzentrum beachten.
WP_EC_4.1.1_3	Software-lieferant	MUSS	sicherstellen, dass keine Zugangsdaten (z.B. Passworte, geheime/private Schlüssel, API-Keys, Schlüssel für symmetrische Verschlüsselungen) in Container-Images bzw. in Konfigurationsdateien gespeichert werden. (SYS.1.6.A8 B) Umsetzungshinweise: Anstatt statische Secrets auszuliefern können Secrets beim Deployment generiert werden.
WP_EC_4.1.1_4	Software-lieferant	MUSS	die Software so gestalten, dass diese keine Geheimnisse und Kennwörter als Plain-Text nutzt. (APP.4.4.A2 B)
WP_EC_4.1.1_5	Software-lieferant	MUSS	für seine Software sicherstellen, dass diese mit minimalen Berechtigungen lauffähig ist.(SYS.1.6.A21 H) Umsetzungshinweise: Beispielsweise sollen für Anwendungen, die nur Lesezugriff benötigen, nur Nutzer verwendet werden, die keinen Schreibzugriff haben.
WP_EC_4.1.1_6	Software-lieferant	MUSS	die Container mit Ressourcenbeschränkungen ausliefern (z.B. Datenmenge und Speicherzugriffe). (SYS.1.6A23 H)
WP_EC_4.1.1_7	Software-lieferant	MUSS	Daten in eigene Mount-Verzeichnisse schreiben. (SYS.1.6A23 H)
WP_EC_4.1.1_8	Software-lieferant	MUSS	sicherstellen, dass Daten, die im Container ausschließlich gelesen werden, als Read-Only-Volume in den Container eingebunden werden. (SYS.1.6A23 H) Umsetzungshinweise: Ein technischer Ansatz kann die Pod-Security-Policy "ReadOnlyRootFileSystem" sein
WP_EC_4.1.1_9	Software-lieferant	SOLL	keine lokalen Speicher der Workernodes benutzen. (SYS.1.6.A19 S) Umsetzungshinweise: Es sollte nur persistent Storage genutzt werden. "Sollte" bietet Ausnahmen für flüchtige Anwendungsfälle (virenscan)

Fortsetzung auf nächster Seite

Fortsetzung von vorheriger Seite

ID	Rolle	Modal-verb	Detailanforderung
WP_EC_ 4.1.1_10	Software-lieferant	SOLL	für seine Software sicherstellen, dass diese nicht in das Root-File-System des Containers schreiben darf. (SYS.1.6A23 H)
WP_EC_ 4.1.1_11	Software-lieferant	SOLL	das temporäre Speichern von Daten in dediziert dafür bereitgestellten Volumes (gängigerweise Ephemeral Storage) vornehmen. (SYS.1.6.A23 H) Umsetzungshinweise: bspw. Daten in temp-Verzeichnissen, wie File-Uploads, etc.

3.2.2.1.2 Dienste und Service-Accounts

ID	Rolle	Modal-verb	Detailanforderung
WP_EC_4.1.2_1	Software-lieferant	MUSS	bei Neuentwicklungen die Software so gestalten, dass jeder Container nur einen Dienst bereitstellt. (SYS.1.6.A11 S)
WP_EC_4.1.2_2	Software-lieferant	MUSS	für Dienste, die unterschiedlich skalieren oder auf verschiedenen Worker-Nodes betrieben werden sollen, entsprechende Parameter der Deployment Descriptoren vorsehen. (SYS.1.6.A11 S) Umsetzungshinweise: Für Helm Charts ist die Parametrisierung über die values.yaml vorzusehen.
WP_EC_4.1.2_3	Software-lieferant	MUSS	pro Anwendung (mindestens) einen eigenen Service-Account bereitstellen, mit dem die Anwendung vollständig betrieben werden kann. (APP.4.4.A9 S)
WP_EC_4.1.2_4	Software-lieferant	MUSS	die Software so entwickeln, dass die Service-Accounts mit den geringst möglichen Berechtigungen (Least-Privilege) betrieben werden können. (APP.4.4.A9 S)
WP_EC_4.1.2_5	Software-lieferant	SOLL	bei der Nutzung von zeitlich wiederkehrenden Aktivitäten auf Systemebene (z.B. CRON-Jobs) die Mechanismen der Container-Plattform zur Ausführung nutzen. (SYS.1.6.A9 S)
WP_EC_4.1.2_6	Software-lieferant	SOLL	die Container so auslegen, dass eine horizontale Skalierung möglich ist. (SYS.1.6.A9 S)
WP_EC_4.1.2_7	Software-lieferant	SOLL	die Software so gestalten, dass jeder Container nur einen Dienst bereitstellt. (SYS.1.6.A11 S) Umsetzungshinweise: Die Einstufung mit “soll” ist nur aufgrund der Übergangsphase aus der “klassischen” in die Containerwelt vorgenommen worden. Es wird ein “muss” angestrebt.
WP_EC_4.1.2_8	Software-lieferant	SOLL	die Software so gestalten, dass Pods nicht den default Service-Account benötigen. (APP.4.4.A9 S) Umsetzungshinweise: Bei OpenShift sind die Rechte des “default” Service-Account bereits maximal eingeschränkt.

3.2.2.1.3 Monitoring

ID	Rolle	Modal-verb	Detailanforderung
WP_EC_4.1.3_1	Software-lieferant	SOLL	Liveness- und Readiness-Checks gemäß der Kubernetes-Dokumentation bereitstellen. Die Checks, insbesondere die Readiness-Checks, sind anwendungsspezifisch zu betrachten (siehe einschlägige Best-Practices, bspw. Google Cloud Blog). (APP.4.4.A11 S)
WP_EC_4.1.3_2	Software-lieferant	SOLL	die forensische Analyse unterstützen, indem u.a. die eingehenden und ausgehenden Requests, sowie Konfigurationsänderungen geloggt werden. (SYS.1.6.A22 H)
WP_EC_4.1.3_3	Software-lieferant	SOLL	ein für seine Software erwartbares Verhalten (Systemaufrufe, Kommunikationsbeziehungen usw.) definieren, beschreiben und dem Softwarebetreiber vorlegen. (SYS.1.6.A24 H) Umsetzungshinweise: Wenn der Softwarelieferant das normale Verhalten - wie oben angesprochen - nicht angeben kann, dann muss der Softwarebetreiber das normale Verhalten durch Erprobung feststellen
WP_EC_4.1.3_4	Software-lieferant	KANN	Start-up-Checks für den Start der Container implementieren. (APP.4.4.A11 S) Umsetzungshinweise: siehe z.B. Configure Liveness, Readiness and Startup Probes

3.2.2.1.4 Images

ID	Rolle	Modal-verb	Detailanforderung
WP_EC_4.1.4_1	Software-lieferant	MUSS	sicherstellen, dass gelieferte Artefakte, insb. Images, dem IT-Grundschutz entsprechen. (SYS.1.6.A10 S)
WP_EC_4.1.4_2	Software-lieferant	MUSS	das Image mit Metainformation versehen und signieren. (SYS.1.6.A12 S) Umsetzungshinweise: Es muss eine Liste von Metadaten die mindestens vorhanden sein müssen existieren (Versionierung und Tagging von Images).
WP_EC_4.1.4_3	Software-lieferant	MUSS	bewerten und sicherstellen, dass sämtliche Bestandteile seiner Anwendungsimages aus vertrauenswürdigen Quellen stammen. Dabei kann auf vom Plattformbetreiber freigegebene Base-Images aufgesetzt werden. (SYS.1.6.A6 S)
WP_EC_4.1.4_4	Software-lieferant	MUSS	dem Softwarebetreiber Container Images mit allen zur Runtime benötigten Modulen bereitstellen. Ein Nachladen von Modulen DARF NICHT erfolgen. (SYS.1.6.A6 S)
WP_EC_4.1.4_5	Software-lieferant	MUSS	regelmäßig Images auf Schwachstellen und Schadcode prüfen und bei Bedarf geeignete Updates zur Verfügung stellen. (SYS.1.6.A6 S) Umsetzungshinweise: Gängig sind in diesem Zusammenhang Security Scanner der Image Registry. Welche Reaktionszeiten angemessen sind, ist vertraglich zu definieren.

3.2.2.1.5 Kommunikationsverbindungen

ID	Rolle	Modal-verb	Detailanforderung
WP_EC_4.1.5_1	Software-lieferant	MUSS	die Software so bereitstellen, dass keine Fernwartung möglich ist. (SYS.1.6.A16 S) Umsetzungshinweise: Die Container Images DÜRFEN KEINE Komponenten zur Fernwartung (zum Beispiel ssh, telnet) enthalten.
WP_EC_4.1.5_2	Software-lieferant	MUSS	Zugriffe und Berechtigungen auf Ressourcen durch seine Software auf die technisch notwendigen Zugriffe beschränken. (APP.4.4.A21 H und SYS.1.6.A21 H)
WP_EC_4.1.5_3	Software-lieferant	MUSS	die Vorgaben und Rahmenbedingungen des Software- und des Plattformbetreibers für die Kommunikationsbeziehungen berücksichtigen und seine Implementierung entsprechend auslegen. (SYS.1.6.A5B) Umsetzungshinweise: Bei Anbietern für Standardsoftware sollte der Plattformbetreiber prüfen, ob seine Anforderungen dem Marktangebot gerecht werden.
WP_EC_4.1.5_4	Software-lieferant	MUSS	die eingehende und ausgehende Kommunikation der Anwendung über Kubernetes-Serviceressourcen (Ingress und Egress) umsetzen.
WP_EC_4.1.5_5	Software-lieferant	MUSS	die notwendigen Kommunikationsbeziehungen (auch innerhalb von Namespaces) zwischen den Komponenten der Anwendung (z.B. Containern und Pods), sowie mit Komponenten außerhalb der Container-Plattform in geeigneter Weise dokumentieren. (APP.4.4.A18)
WP_EC_4.1.5_6	Software-lieferant	SOLL	die zertifikatsbasierte Authentifizierung unterstützen und Kommunikationsverbindungen nur für die in den Zertifikaten hinterlegten Identitäten erlauben. (APP.4.4.A18) Umsetzungshinweise: z.B. mittels mTLS. Es sollte mindestens eine Serverauthentifizierung umgesetzt werden. Eine Client-Authentifizierung sollte möglich werden. Die Umsetzung kann auch über einen Service-Mesh (Bereitstellung durch Plattformbetreiber notwendig) umgesetzt werden.

3.2.2.1.6 Systemarchitektur

ID	Rolle	Modal-verb	Detailanforderung
WP_EC_4.1.6_1	Software-lieferant	MUSS	Software-Produkte so entwickeln, dass ein Re-Deployment auf einem "Restore-Cluster" jederzeit möglich ist (APP.4.4.A5 B)
WP_EC_4.1.6_2	Software-lieferant	SOLL	die Anwendung so bereitstellen, dass die lt. Schutzbedarf erforderliche Anwendung von technischen Schutzmaßnahmen wie z.B. die Verwendung von Betriebssystem-Mechanismen, die Isolation und Kapselung der Anwendung und auch die Netzwerktrennung durch die Software-Architektur unterstützt und nicht verhindert wird (durch Plattform vorgegebene Isolationsmechanismen wie z.B. SE Linux und CGroups; plattformspezifische Isolationsmechanismen wie z.B. Namespaces, Pod Security Admission und Network Policies) (SYS.1.6.A3 B, APP.4.4.A4 B)

Fortsetzung auf nächster Seite

Fortsetzung von vorheriger Seite

ID	Rolle	Modal-verb	Detailanforderung
WP_EC_4.1.6_3	Software-lieferant	SOLL	Software-Produkte so entwickeln, dass zu jedem beliebigen Zeitpunkt eine konsistente Datensicherung seiner Anwendung durch den Software- oder Plattformbetreiber durchgeführt werden kann und eine Point in Time Wiederherstellung möglich ist. (APP.4.4.A5 B) Umsetzungshinweise: Bei einem Recovery ist der exakt gleiche Zustand des Clusters (gleiche Anzahl an Pods) nicht erreichbar. Folgende Mechanismen sollten beachtet werden: - Festspeicher (Persistent Volumes) - Konfigurationsdateien von Kubernetes und den weiteren Programmen der Control Plane - plattformspezifische Erweiterungen wie z.B. Monitoring, Logging, Ingress etc. - Datenbanken der Konfiguration, namentlich hier etcd - alle Infrastrukturanwendungen die zum Betrieb des Clusters und der darin befindlichen Dienste notwendig sind - die Datenhaltung der Code und Image Registries
DS_40_A001	Software-lieferant	SOLL	bei der Erstellung von Containern maximal einen Prozess im Container erzeugen (Single-Process Only).

3.2.2.2 Konfiguration/Administration

ID	Rolle	Modal-verb	Detailanforderung
WP_EC_4.2_1	Software-lieferant	MUSS	Konfigurationsanpassungen so ausführen, dass keine manuellen Anpassungen in laufenden Containern erfolgen. (SYS.1.6.A9 S)
WP_EC_4.2_2	Software-lieferant	MUSS	die Software so gestalten, dass die Container-Runtime und alle instanziierten Container nur von einem nicht-privilegierten System-Account ausgeführt werden, der keine erweiterten Rechte für den Container-Dienst bzw. das Betriebssystem des Host-Systems benötigt oder diese Rechte erlangen kann. Ausnahmen hiervon sind Container, welche Aufgaben des Host-Systems übernehmen.(kein "privileged"-Mode für die Container-Runtime und den laufenden Containern. Beispiel für Ausnahmen sind: ingress, egress und Infrastrukturcontainer) (SYS.1.6.A17 S)
WP_EC_4.2_3	Software-lieferant	MUSS	die Privilegien für Container auf das erforderliche Minimum begrenzen. (SYS.1.6.A17 S)
WP_EC_4.2_4	Software-lieferant	MUSS	die Informationen zur Einschränkung der Nutzung von erweiterten Privilegien dem Softwarelieferanten zur Verfügung stellen. (SYS.1.6.A17 S)
WP_EC_4.2_5	Software-lieferant	MUSS	die Software so gestalten, dass die genutzten User- und Group-ID's keine Berechtigungen auf die System und Datenbereiche des Hosts erfordern. (SYS.1.6.A18 S)
WP_EC_4.2_6	Software-lieferant	MUSS	die erforderlichen User- und Group-ID's und deren Berechtigungen benennen (z.B. in den Konfigurationsdateien). (SYS.1.6.A18 S)

Fortsetzung auf nächster Seite

Fortsetzung von vorheriger Seite

ID	Rolle	Modal-verb	Detailanforderung
WP_EC_4.2_7	Software-lieferant	MUSS	die mitgelieferte Konfiguration der Container versioniert bereitstellen und den Bezug zu Imageversionen sicherstellen. (SYS.1.6.A20 S) Umsetzungshinweise: Entweder z.B. durch Label in der YAML oder als git-repo.
WP_EC_4.2_8	Software-lieferant	MUSS	das vom Softwarebetreiber benötigte Verteilungsschema für die Pods unterstützen. (APP.4.4.A14 H) Umsetzungshinweise: z.B. Aufteilung der Pods entsprechend der Aufgaben. Der Softwarebetreiber muss das Verteilungsschema für die Pods definieren. Die Nodes müssen mindestens nach folgenden Aufgaben unterschieden und getrennt werden: - Bastion-Nodes zur Realisierung von ingress und egress - Anwendungs-Nodes zum Betrieb der Pods für die Anwendungen - Speicher-Nodes zur Bereitstellung der Speicherlösungen - Management-Nodes für den Cluster
WP_EC_4.2_9	Software-lieferant	MUSS	notwendige Berechtigungen im Deployment beschreiben. Diese müssen minimal gewählt werden. (APP.1.6.A19 S)
WP_EC_4.2_10	Software-lieferant	MUSS	den Abschluss eines Init-Container sicherstellen. (APP.4.4.A6 S)
WP_EC_4.2_11	Software-lieferant	MUSS	den Init-Container eindeutig von der Applikation logisch trennen. (APP.4.4.A6 S)
WP_EC_4.2_12	Software-lieferant	SOLL	das automatische Management seiner Software durch die Bereitstellung von geeigneten Administrationswerkzeugen unterstützen. (SYS.1.6.A2 B) Umsetzungshinweise: Werkzeuge können Deployment-Scripte, Pipeline-Scripte für CI/CD, Kubernetes Operatoren sein.
WP_EC_4.2_13	Software-lieferant	SOLL	Init-Container für Konfigurationsanpassungen nutzen. (SYS.1.6.A9 S)
WP_EC_4.2_14	Software-lieferant	SOLL	die Deployments entsprechend der Richtlinien des Softwarebetreibers ausrichten. (APP.4.4.A13 H). Umsetzungshinweise: Der Softwarebetreiber muss Richtlinien für die Einstellungen von Deployments definieren und dem Softwarelieferanten bekannt geben.
WP_EC_4.2_15	Software-lieferant	SOLL	bei besonders kritischen Anwendungen Operatoren zur Automatisierung von Betriebsaufgaben liefern. (APP.4.4.A16 H)

3.2.2.3 Protokollierung

ID	Rolle	Modal-verb	Detailanforderung
WP_EC_4.3_1	Software-lieferant	MUSS	alle Protokolldaten der Anwendung im Container über die Standardausgabe ausgeben. Log-Daten MÜSSEN über STDOUT / STDERR ausgegeben werden (SYS.1.6.A7 B) (Vorgabe aus dem Whitepaper für die Erstellung von Containern Kapitel 4.3, siehe Referenzdokumente).

Fortsetzung auf nächster Seite

Fortsetzung von vorheriger Seite

ID	Rolle	Modal-verb	Detailanforderung
DS_44_A001	Software-lieferant	MUSS	die Vorgaben des BSI OPS.1.1.5 einhalten (siehe Referenzdokumente).
DS_44_A002	Software-lieferant	MUSS	die folgende Vorgabe für das Logging einhalten: - Datum- und Zeitangaben MÜSSEN im ISO 8601 Format in der Zeitzone GMT+01:00 geloggt werden.
DS_44_A003	Software-lieferant	SOLL	die folgende Vorgabe für das Logging einhalten: - Es SOLL ein JSON-Format für das Logging verwendet werden, welches für die Einheitlichkeit mindestens folgende Felder beinhaltet: timestamp, message, hostname, level, app-version
DS_44_A004	Software-lieferant	KANN	die folgende Vorgabe für das Logging einhalten: - Es KANN folgendes Feld benutzt werden, sofern es die Anwendung unterstützt: trace-id
DS_44_A005	Software-lieferant	KANN	die folgende Vorgabe für das Logging einhalten: - Es KÖNNEN durch die Anwendung seitens des Softwarelieferanten weitere Felder hinzugefügt werden, diese müssen allerdings mit dem prefix "app-" starten (z.B.: app-prozessschritt)
DS_44_A007	Software-lieferant	MUSS	für das Setzen des Log-Levels eine Konfiguration vorsehen, welche zur Laufzeit des Containers gesetzt werden kann.
DS_44_A008	Software-lieferant	SOLL	16KB pro Logeintrag nicht überschreiten.

3.2.2.4 Fehlertoleranz

ID	Rolle	Modal-verb	Detailanforderung
WP_EC_4.4_1	Software-lieferant	MUSS	eine Fehlertoleranz für die Anwendung für den Wiederanlauf nach einem ungeordneten Abbruch der Datenverarbeitung sicherstellen. (APP.4.4.A19 H) Umsetzungshinweise: Mögliche Maßnahmen: - atomare Gestaltung - keine persistente Daten im Container - Synchrone Transaktionen falls mehrere Datenbanken, Webservices usw. genutzt werden - Mechanismus zur Prüfung der Konsistenz der Daten - ggf. Mittel zur Wiederherstellung der Konsistenz.
WP_EC_4.4_2	Software-lieferant	SOLL	die Anwendung stateless gestalten, um die Mittel der Plattform bei einer Umsetzung von Hochverfügbarkeit nutzen zu können. (APP.4.4.A19 H) Umsetzungshinweise: Die Optionen hinsichtlich Zustandslosigkeit sind für Bestandsanwendungen beschränkt und lassen sich ggf. nicht umsetzen.

Fortsetzung auf nächster Seite

Fortsetzung von vorheriger Seite

ID	Rolle	Modal-verb	Detailanforderung
WP_EC_4.4_3	Software-lieferant	SOLL	Container stateless gestalten und eine transaktionsorientierte Funktionsweise der Anwendung sicherstellen. (SYS.1.6.A9 S) Umsetzungshinweise: Die transaktionsorientierte Funktionsweise soll sicherstellen, dass alle Verarbeitungen korrekt abgeschlossen werden können und keine inkonsistenten Zustände entstehen, selbst wenn Container neu deployed werden.
WP_EC_4.4_4	Software-lieferant	KANN	Start-up-Checks für den Start der Container implementieren. (APP.4.4.A11 S) Umsetzungshinweise: siehe Configure Liveness, Readiness and Startup Probes

3.2.2.5 Update- und Patchmanagement

ID	Rolle	Modal-verb	Detailanforderung
WP_EC_4.5_1	Software-lieferant	MUSS	ein Patchmanagement gewährleisten. (SYS.1.6.A10 S)
WP_EC_4.5_2	Software-lieferant	MUSS	sicherstellen, dass Updates von Software nicht im laufenden Container installiert werden. Bei persistenten Containern SOLLTE geprüft werden, ob in (absolut seltenen) Ausnahmefällen ein Update des jeweiligen Containers geeigneter ist, als den Container vollständig neu zu provisionieren. (SYS.1.6.A14 S) Umsetzungshinweise: Es dürfen keine Tools enthalten sein, welche nicht für den Betrieb der Applikation notwendig sind (Paketmanager, YUM, APT, APK usw.).
WP_EC_4.5_3	Software-lieferant	MUSS	bei sicherheitsrelevanten und featurebasierten Updates der Anwendungen oder der Standardimages neue Images erstellen und eindeutig versioniert innerhalb der vereinbarten Zeiträume / SLAs zur Verfügung stellen. (SYS.1.6.A14 S)
WP_EC_4.5_4	Software-lieferant	MUSS	das Rechte- und Rollenkonzept für die Verwaltung des Sourcecodes umsetzen. (APP.4.4.A2 B) (s, auch CON.8)
WP_EC_4.5_5	Software-lieferant	SOLL	sicherstellen, dass die Software in der Lage ist, den Service auch während Updates entsprechend der definierten Anforderungen aufrechtzuerhalten. (SYS.1.6.A14 S) Umsetzungshinweise: d.h. Clusterfähigkeit der Software

3.2.3 Dokumentation

Dieser Abschnitt beschreibt die Anforderungen im Bereich "Dokumentation". Sie sind dem Detailstandard 40 entnommen.

3.2.3.1 Dokumentation des Containers

ID	Rolle	Modal-verb	Detailanforderung
WP_EC_5.1_1	Software-lieferant	MUSS	die Richtlinien und Vorgaben zum sicheren Betrieb dem Softwarelieferanten bereitstellen. (SYS.1.6.A10 S)
WP_EC_5.1_2	Software-lieferant	MUSS	dokumentieren, welche Quellen für Images verwendet wurden. (SYS.1.6.A12 S)
WP_EC_5.1_3	Software-lieferant	MUSS	beschreiben, welche Quellen für Bestandteile des Image verwendet wurden, inklusive eines ggf. genutzten Basis-Images (SYS.1.6.A12 S) Umsetzungshinweise: Dies dient der Verdeutlichung des konzeptionellen Aufbaus der Software (z.B. zur Beurteilung hinsichtlich der verwendeten Lizenzen)
WP_EC_5.1_4	Software-lieferant	MUSS	Dimensionierungen (Requests) und Begrenzungen (Limits) für CPU, RAM sowie temporären Speicher für jeden Container auf den von ihm angenommenen Nutzerprofil empfehlen. (SYS.1.6.A15 S) Umsetzungshinweise: Der Softwarelieferant muss Defaultwerte für die Limits liefern, die dann ggf. vom Softwarebetreiber angepasst werden können.
WP_EC_5.1_5	Software-lieferant	MUSS	die Möglichkeiten und Funktionsweisen zur Skalierung der Dienste beschreiben. (SYS.1.6.A15 S) z.B. kubernetes pod hpa Umsetzungshinweise: Anmerkung: Limitierung muss auch berücksichtigt werden, wenn der Dienst skaliert werden soll
WP_EC_5.1_6	Software-lieferant	MUSS	Konfigurationsoptionen für die Protokollierung beschreiben. (SYS.1.6.A7 B) Umsetzungshinweise: Der Softwarebetreiber soll in die Lage versetzt werden, die Protokollierung, bspw. Log-Level, -Format oder -Kategorien, einzurichten.
WP_EC_5.1_7	Software-lieferant	MUSS	den Einsatz eines Init-Containers dokumentieren. (APP.4.4.A6 S) Umsetzungshinweise: Es müssen die Berechtigungen dargestellt werden. Er muss die notwendigen Ressourcen dokumentieren und eine Empfehlung bereitstellen.
WP_EC_5.1_8	Software-lieferant	MUSS	die Ressourcenanforderungen an die Init-Container beschreiben und dem Softwarebetreiber mitteilen. (APP.4.4.A6 S)
WP_EC_5.1_9	Software-lieferant	SOLL	die Umsetzung der Richtlinien und Vorgaben mit dem Softwarelieferanten vertraglich festhalten. (SYS.1.6.A10 S) Umsetzungshinweise: Insbesondere bei der Lieferung von Images, da hier die Fähigkeit der Anpassung der Images beschränkt wird.
WP_EC_5.1_10	Software-lieferant	SOLL	Handlungsempfehlungen bereitstellen, für den Fall, dass ein Container an seine Ressourcengrenzen kommt. (SYS.1.6.A15 S)
WP_EC_5.1_11	Software-lieferant	SOLL	die Möglichkeiten der Hochverfügbarkeit darstellen. (SYS.1.6.A25 H)

Fortsetzung auf nächster Seite

Fortsetzung von vorheriger Seite

ID	Rolle	Modal-verb	Detailanforderung
WP_EC_5.1_12	Software-lieferant	KANN	dem Softwarebetreiber ein Klassifikationssystem für die Erstellung einer Whitelist für vertrauenswürdige Quellen vorschlagen. (SYS.1.6.A12 S)

3.2.3.2 Dokumentation von Anforderungen an persistenten Speicher

ID	Rolle	Modal-verb	Detailanforderung
WP_EC_5.2_1	Software-lieferant	MUSS	die Dimensionierungen und die Größe des benötigten persistenten Speichers empfehlen. Die Informationen müssen dem Softwarebetreiber mitgeteilt werden. (SYS.1.6.A15 S) Umsetzungshinweise: Der Softwarelieferant muss Defaultwerte für die Limits liefern, die dann ggf. vom Softwarebetreiber angepasst werden können.

3.2.3.3 Kommunikationsbeziehungen dokumentieren

ID	Rolle	Modal-verb	Detailanforderung
WP_EC_5.3_1	Software-lieferant	MUSS	alle erforderlichen Kommunikationsverbindungen inklusive Ziele, Kommunikationsprotokolle und Ports beschreiben, die für Inbetriebnahme und Betrieb erforderlich sind.(APP.4.4.A7 S, siehe auch SYS.1.6.A5 B, siehe auch APP.4.4.A14 H). Umsetzungshinweise: Feste Zuordnung von Containern zu Container-Hosts, Ausführung der einzelnen Container und/oder des Container-Hosts mit Hypervisoren und feste Zuordnung eines einzelnen Containers zu einem einzelnen Container-Host.
WP_EC_5.3_2	Software-lieferant	MUSS	benötigte Zugriffe und Berechtigungen auf Ressourcen dokumentieren. (APP.4.4.A21 H)
WP_EC_5.3_3	Software-lieferant	MUSS	dem Softwarebetreiber die von ihm definierten erweiterten Maßnahmen mitteilen, sofern vorhanden. (SYS.1.6.A26 H) Umsetzungshinweise: Bei klar definierten Zielen sollten FQDN oder IP-Adressen zur Beschreibung genutzt werden. Auch die Kommunikation innerhalb der Anwendung muss bekannt und beschrieben sein.
WP_EC_5.3_4	Software-lieferant	SOLL	Dimensionierungen (Requests) und Begrenzungen (Limits) auf den von ihm angenommenen Nutzerprofil für das Netzwerk definieren und dem Softwarebetreiber mitteilen. (SYS.1.6.A15 S)
WP_EC_5.3_5	Software-lieferant	KANN	in Abstimmung mit dem Softwarebetreiber die Kommunikationsbeziehungen in einer technisch formalen Form darstellen. (APP.4.4.A7 S, siehe auch SYS.1.6.A5 B) Umsetzungshinweise: siehe Network Policies

3.2.3.4 Berechtigungen

ID	Rolle	Modal-verb	Detailanforderung
WP_EC_5.4_1	Software-lieferant	MUSS	darlegen, für welche Zwecke technische User (z.B.: Kubernetes Service-Accounts) Berechtigungen erfordern. (APP.4.4.A3 B) Umsetzungshinweise: Der Softwarebetreiber muss die benötigten Accounts und Berechtigungen beim Plattformbetreiber anfordern und sie müssen den Privilegienrichtlinien entsprechen. Der Softwarebetreiber muss anonyme Zugriffe für administrative Handlungen verhindern.

3.2.3.5 Lizenzen

ID	Rolle	Modal-verb	Detailanforderung
WP_EC_5.5_1	Software-lieferant	MUSS	dem Softwarebetreiber bekannt geben, welche Lizenzen auf der Container-Plattform für den Betrieb der Software erforderlich sind und diese je nach vertraglicher Vereinbarung auch bereitstellen. (SYS.1.6.A9 S)

3.2.4 Lieferung

Die Vorgaben WP_EC_6_1 bis WP_EC_6_11 für Softwarelieferanten bezüglich der Lieferung von containerisierten Softwareprodukten sind bereits oben in Kap.3.2.1.3 (Vorgaben für die Lieferung oder Bereitstellung von containerisierten Softwareprodukten) aufgeführt.

3.2.5 Architektur

Die folgende Architektur-Vorgabe für Softwarelieferanten ist dem Rahmenwerk Kap.4.4 Prinzip DA-03 entnommen:

ID	Rolle	Modal-verb	Detailanforderung
RW_P_DA-03	Software-lieferant	SOLL	Die Datenübertragung wird präferiert im REST-Architekturstil realisiert (Zustandslosigkeit, HTTP-Caching, Einheitlichkeit der Schnittstellen, Adressierbarkeit von Ressourcen, Repräsentation zur Veränderung von Ressourcen, selbstbeschreibende Nachrichten).

3.2.6 Export von Daten

Das Reifegradmodell enthält im Bereich Digitale Souveränität die Kriterien

- Export von Kundendaten
- Export von Konfiguration

In Abhängigkeit des geplanten Reifegrades des Services sind dort die zugehörigen Anforderungen zu finden. Im Gegensatz zu den oben in Kapitel 3.1.4 (Zertifizierung von Softwarelieferanten) aufgeführten Kriterien handelt es sich hierbei aber nicht um Minimalkriterien. Sie sind daher hier im Leitfaden nicht gesondert aufgelistet sondern unterliegen in Abhängigkeit des geplanten Reifegrades des Services der Absprache mit dem Softwarebetreiber.

3.2.7 Integration der IAM und SSO Anbindung

Die Serviceanbindung an das IAM der DVC kann in drei Stufen erfolgen:

- Stufe 1: Keine Integration mit dem IAM der DVC
- Stufe 2: Authentifizierungen über das IAM der DVC
- Stufe 3: Autorisierung über das IAM der DVC

Zum Zeitpunkt der Erstellung dieses Leitfadens werden nur Services der Stufe 1 unterstützt. Eine Dokumentation für die Anbindung in Stufen 2 und 3 wird entsprechend des DVC-Produktfortschrittes auf den Seiten der Deutschen Verwaltungscloud veröffentlicht werden.

3.2.8 Barrierefreiheit

Das Reifegradmodell enthält im Bereich Funktionelles die Kriterien zur Barrierefreiheit. In Abhängigkeit des geplanten Reifegrades des Services sind dort die zugehörigen Anforderungen zu finden.

3.2.9 Künstliche Intelligenz

Die folgende Vorgabe bezüglich der Verwendung von KI-Inhalten (Kennzeichnungspflicht gemäß EU AI Act) ist dem Rahmenwerk Kap. 4.3 Prinzip SI-03 entnommen:

ID	Rolle	Modal-verb	Detailanforderung
RW_P_SI-03	Software-lieferant	MUSS	Für den Cloud-Service-Kunden ist jederzeit eindeutig ersichtlich, ob ein beauftragter Cloud-Service KI-Komponenten im Sinne der KI-Verordnung der EU enthält bzw. ob und unter welchen Bedingungen die Daten eines beauftragten Cloud-Services als Trainingsdaten für KI-Modelle verwendet werden könnten. Es ist sichergestellt, dass die Anforderungen der KI-Verordnung der EU erfüllt werden.

3.3 Phase 3: Unterstützungsphase

3.3.1 Einrichtung und Integration der Software

Die Software ist erstellt und die Auswahl des Softwarebetreibers ist erfolgreich abgeschlossen. Jetzt muss die Software "nur" noch als Service eingerichtet und in Betrieb genommen werden. Dies ist Kernaufgabe des Softwarebetreibers, der diese Schritte auf der Basis der vom Softwarelieferanten bereitgestellten Dokumentation durchführen wird. Als Softwarelieferant sollten Sie an dieser Stelle Einrichtungssupport anbieten können, um diese letzte Hürde vor der Inbetriebnahme gemeinsam zu bewältigen.

3.3.2 Support

Support-Aufgaben im operativen Tagesgeschäft gehören zum Verantwortungsbereich des Softwarebetreibers. Dazu wird der Softwarebetreiber mit dem Softwarelieferanten einen Supportvertrag aushandeln, in dem vereinbart wird, bei welchen Aufgaben der Softwarelieferant den Softwarebetreiber in welchem Umfang bei Supportaufgaben unterstützen wird. Beispiele für auszuhandelnde Support-Aufgaben sind die Annahme von Incidents und Changes.

3.4 Phase 4: Wartungsphase

Der auf der Basis Ihrer Software erstellte Cloud-Service ist nun in der DVC verfügbar und kann von Cloud-Service Kunden bestellt und dann produktiv genutzt werden.

Für Sie als Softwarelieferant bedeutet dies, dass nun im Rahmen des IT Servicemanagements die in Phasen 1 und 3 vereinbarten Servicevereinbarungen zum Tragen kommen. Diese werden von der Kritikalität der Software / des Service abhängen, sich aber über individuelle Vereinbarungen hinaus mindestens in die folgenden Anforderungen einteilen lassen:

ID	Rolle	Modal-verb	Detailanforderung
LF_SL_A001	Software-lieferant	MUSS	Ein Softwarelieferant muss Bugmeldungen (z.B. im Rahmen der Annahme von Incidents, s. oben in Kapitel 3.3.2) entgegennehmen und sie qualifizieren.
LF_SL_A002	Software-lieferant	MUSS	Ein Softwarelieferant muss Bugfixes liefern.
LF_SL_A003	Software-lieferant	KANN	Ein Softwarelieferant bietet betriebsunterstützende Leistungen an.

4 Rollenübergreifende Erläuterungen zur DVC

Grundsätzlich verfolgt die DVC den Ansatz, containerisierte Software als Service zur Verfügung zu stellen. Daher beschreibt der vorliegende Leitfaden auch die Anforderungen an Softwarelieferanten, die solche containerisierte Software entwickeln. Proprietär entwickelte "Nicht Container"-Software kann momentan ebenfalls in die DVC eingebracht werden. Dazu ist es aber erforderlich, dies detailliert mit einem Softwarebetreiber abzustimmen. Die Ausführungen in diesem Dokument betreffen ausschließlich containerisierte Software.

Der vorliegende Leitfaden beschäftigt sich mit technischen und organisatorischen Fragestellungen. Nicht Gegenstand dieses Dokumentes sind die juristischen Aspekte der DVC, die Einfluss auf die Rolle Softwarelieferant haben.

4.1 Motivation DVC

Die Deutsche Verwaltungscloud (DVC) verfolgt das Ziel, die Digitalisierung der öffentlichen Verwaltung voranzutreiben und dabei die Digitale Souveränität Deutschlands zu stärken. Die Rahmenbedingungen der DVC sind im Rahmenwerk Zielarchitektur niedergelegt (siehe DVC Rahmenwerk, Kapitel 1).

Formaler Initiator für die DVC ist der IT-Planungsrat, der im Rahmen der Strategie zur Stärkung der Digitalen Souveränität zunächst die Deutsche Verwaltungscloud-Strategie (DVS) verabschiedet und damit die konzeptionelle Grundlage für die Deutsche Verwaltungscloud (DVC) geschaffen hat. Weitere Details zur Motivation der DVC finden sich im DVC Rahmenwerk speziell in den Kapiteln 1 und 2.

4.2 Rahmenwerk

Übergeordnetes Dokument für alle strategischen DVC-Themen ist das DVC Rahmenwerk. Es kann daher als Einstiegslektüre für Interessenten dienen, die sich zunächst rollenübergreifend über das Thema DVC informieren wollen. Das Rahmenwerk fokussiert auf die strategischen Architekturentscheidungen („was").

Die technischen Anforderungen werden im Reifegradmodell, diversen Detailstandards sowie Whitepapern der IGBvC beschrieben.

4.3 Organisation der DVC

Das DVC Rahmenwerk enthält in Kapitel 3 "Geschäftsarchitektur DVC" einen Überblick über die Organisation der DVC.

Relevant für die im vorliegenden Leitfaden beschriebene Rolle Softwarelieferant sind dabei vor allem die Koordinierungsstelle (DVC Rahmenwerk, Kap.3.1) sowie die verschiedenen Rollen (DVC Rahmenwerk, Kap 3.2).

4.4 Motivation Leitfäden

Die zu allen DVC-Rollen verfügbaren Leitfäden werden erstellt, um Interessenten ein Set von "Checklisten" zur Verfügung zu stellen, mit denen sie feststellen können, ob sie alles bearbeitet haben, was für die im Leitfaden beschriebene Rolle erschlossen werden sollte.

Die in den Leitfäden beschriebenen Anforderungen wurden weitestgehend aus anderen Dokumenten gemäß der folgenden Abbildung aus Kapitel 4 des DVC Rahmenwerk übernommen:

4.5 Abgrenzung der Rollen

Im vorliegenden Dokument geht es ausschließlich um die Rolle Softwarelieferant. Es besteht folgende **Abgrenzung zu den Rollen Softwarebetreiber und Plattformbetreiber**:

Der Betrieb der Anwendungen ist Aufgabe der Rollen Softwarebetreiber und Plattformbetreiber. Der Softwarebetreiber übernimmt zusätzlich die Aufgabe, auf Basis von Software der Softwarelieferanten einen "BSI Grundschutz-konform betriebenen" Service bereitzustellen. Als Resultat dieser Aufgabenverteilung ist daher in der DVC keine Software (eines Softwarelieferanten) buchbar, sondern ein (DVC) Cloud-Service eines Softwarebetreibers in der Infrastruktur eines Plattformbetreibers.

Die allgemeinen Geschäftsbedingungen zur Nutzung des Cloud-Service Portals legen fest, dass Cloud-Service Anbieter folgende Anforderungen erfüllen müssen, um Cloud-Services über das Cloud-Service Portal der DVC bereitstellen zu können:

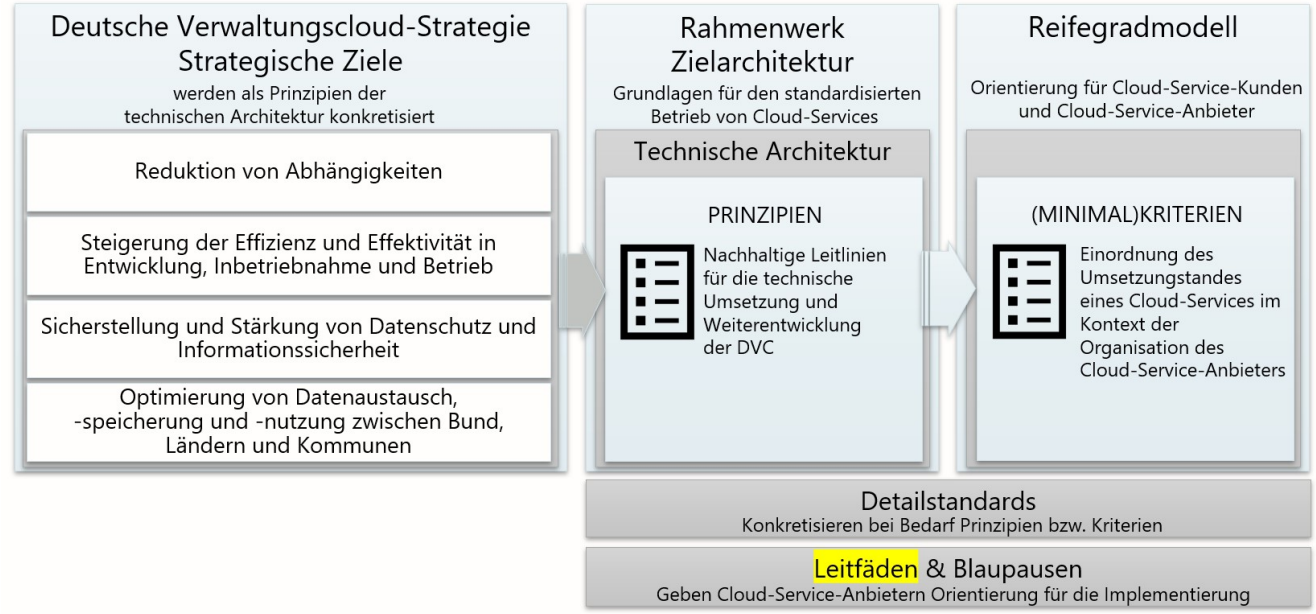

Abbildung 3: DVC Rahmenwerk Dokumentation

- Cloud-Service-Anbieter (darunter fallen die Rollen Softwarebetreiber und Plattformbetreiber) können ausschließlich sein: "Öffentliche Stellen i.S.v. § 1 Abs. 1 OZG, Rechtsträger von Behörden i.S.v. § 2 Abs. 9 OZG, öffentliche Auftraggeber im Sinne des § 99 Nr. 1 bis 3 GWB und juristische Personen des öffentlichen oder privaten Rechts, hinsichtlich derer die Voraussetzungen des § 108 Abs. 1 oder Abs. 4 GWB vorliegen. Cloud-Service-Anbieter kann nicht sein, wer 20 % oder mehr seiner Tätigkeiten am offenen Markt erbringt (vgl. § 108 Abs. 1 Nr. 2, Abs. 4 Nr. 2 bzw. Abs. 6 Nr. 3 GWB)". Anbieter, die diese Definition erfüllen, werden in diesem Dokument "verwaltungsinterne Anbieter" genannt.

Als Cloud-Service-Anbieter gemäß obiger Definition haben Sie die Wahl, entweder beide Rollen (Softwarelieferant und Softwarebetreiber) einzunehmen oder nur als Softwarelieferant mitzumachen, z.B. wenn dies eher ihrem Unternehmens-Scope entspricht. Die Rolle Softwarebetreiber kann ausschließlich durch Anbieter gemäß obiger Definition eingenommen werden. Für verwaltungsexterne Anbieter ist die Rolle Softwarelieferant daher zum Zeitpunkt der Veröffentlichung dieses Leitfadens die einzige Möglichkeit, Software in der DVC bereitzustellen. Sie erhalten dadurch Zugang zu einem großen Markt potentieller Kunden in der ÖV (Bund, 16 Bundesländer, ca. 11000 Kommunen) und somit die Chance, ihre eigene - schon existierende oder noch zu entwickelnde - Software auf diesem Markt zu platzieren.

4.6 Nomenklatur

4.6.1 ID-Vergabe

Die ID-Vergabe in der ersten Spalte der Anforderungen wurde nach dem folgenden Muster so gewählt, dass die Quelle der Anforderung zu erkennen ist:

ID	Quelle der Anforderung	Beispiel
DS_<DS_NR>_A<lfd_nr>	**D**etail**s**tandard	DS_**10**_A**003** (**3**.Anforderung aus Detailstandard **10**)
LF_SL_A<lfd_nr>	**L**eit**f**aden **S**oftware**l**ieferant	LF_SL_A**001** (**1**.Anforderung des vorliegenden Leitfadens, die nicht aus einem referenzierten Dokument stammt)
RG_A<lfd_nr>	DVC-**R**eife**g**radmodell	RG_A**6** (dabei gibt die <lfd_nr=**6**> an, um welche der Anforderungen aus dem Reifegradmodell es sich handelt)
RW_P_<xy>-<lfd_nr>	DVC-**R**ahmen**w**erk	RW_P_**DA-03** (Prinzip **DA-03**)
WP_EC_<Kapitel_nr>_<lfd_nr>	(IGBvC-)**W**hite**p**aper **E**rstellung von **C**ontainern	WP_EC_**6_1** (**1**.Anforderung aus Kapitel **6** des Whitepaper Erstellung von Containern)
WP_SD_<Kapitel_nr>_<lfd_nr>	(IGBvC-)**W**hite**p**aper **S**tandardisierung **D**eployment	WP_SD_**5.3_1** (**1**.Anforderung aus Kapitel **5.3** des Whitepaper Standardisierung Deployment)

4.6.2 Modalverben

Die Verwendung und Bedeutung der modalen Hilfsverben (alternativ Modalverben) ist in der DIN-Norm 820-2 oder RFC2119 geregelt. Ergänzend zu diesen Regeln werden die Modalverben bzw. Schlüsselwörter „MUSS", „SOLL" und „KANN" in den Anforderungstabellen dieses Dokumentes wie folgt verwendet:

- Modalverb "**MUSS**" weist auf eine absolut zu erfüllende Anforderung (uneingeschränkte Anforderung) bzw. bei Negierung ein Verbot (uneingeschränktes Verbot) hin.
- Modalverb "**SOLL**" bedeutet, dass eine Anforderung normalerweise erfüllt werden muss, es aber Gründe geben kann, dies doch nicht zu tun. Dies muss aber sorgfältig abgewogen und stichhaltig begründet werden. Analog bei Negierung: etwas sollte normalerweise nicht getan werden, es gibt aber Gründe, dies doch zu tun. Dies muss aber sorgfältig abgewogen und stichhaltig begründet werden.
- Modalverb "**KANN**" zeigt eine Option an.

Anhang

Glossar

Ein DVC-weit gültiges Glossar ist zu finden im DVC Rahmenwerk Kap. 6.1.

Referenzdokumente

Dokument	Link	PDF
DVC Detailstandards - (10) Detailstandards der CI/CD-Pipeline	Detailstandard 10	Detailstandard 10 PDF
DVC Detailstandards - (16) Deployment von Softwarelösungen	Detailstandard 16	Detailstandard 16 PDF
DVC Detailstandards - (17) Schwachstellenscan von Softwarelösungen	Detailstandard 17	Detailstandard 17 PDF
DVC Detailstandards - (40) Vorgaben für die Erstellung von containerisierten Softwareprodukten	Detailstandard 40	Detailstandard 40 PDF
DVC Detailstandards - (41) Vorgaben für die Lieferung oder Bereitstellung von containerisierten Softwareprodukten	Detailstandard 41	Detailstandard 41 PDF
DVC Detailstandards - (44) Logging von Containern	Detailstandard 44	Detailstandard 44 PDF
DVC Detailstandards - (51) Vorgaben für die Lieferung oder Bereitstellung von Kubernetes Artefakte	Detailstandard 51	Detailstandard 51 PDF
DVS - Deutsche Verwaltungscloud-Strategie: Föderaler Ansatz, in: IT-Planungsrat, 17.11.2020, Kap. 2.1	nur PDF	Deutsche Verwaltungscloud-Strategie
Open CoDE ist die gemeinsame Plattform der Öffentlichen Verwaltung für den Austausch von Open Source Software	openCode	openCode
DVC Rahmenwerk der Zielarchitektur	nur PDF	Rahmenwerk PDF
DVC Reifegradmodell	nur PDF	Reifegradmodell PDF
Technische Richtlinie TR-02102-2	nur PDF	Technische Richtlinie TR-02102-2 Kryptographische Verfahren: Empfehlungen und Schlüssellängen
Whitepaper für die Erstellung von Containern	nur PDF	Whitepaper für die Erstellung von Containern
Whitepaper Standardisierung des Deployments	nur PDF	Whitepaper Standardisierung des Deployments