GANZ EINFACH

# Private Cloud

## Sicher, lokal und absolut unabhängig



Danilo Sieren

Danilo Sieren

# Private Cloud

*Eigene Dienste sicher aufbauen*
*Dateien Fotos und Anwendungen*
*betreuen*

Überarbeitete und erweiterte
Neuauflage 2026

Verstehen · Anwenden · Überprüfen

# Vorwort

Eine Private Cloud ist ein selbst betriebener Dienst für deine Dateien, Fotos oder andere persönliche Anwendungen. Du entscheidest über Speicherorte und Zugänge. Gleichzeitig übernimmst du Aufgaben, die bei einem gebuchten Dienst der Anbieter erledigt: Aktualisierung, Datensicherung, Wiederherstellung und Hilfe für die Menschen, die deine Cloud verwenden.

Dieses Buch begleitet dich vom ersten Plan bis zum dauerhaft betreuten Betrieb. Du lernst, welche Hardware zu deinen Aufgaben passt, wie Container ihre Daten speichern und wie du Dienste im Heimnetz und von unterwegs erreichst. Nextcloud, Immich, Vaultwarden, Pi-hole, Jellyfin, Plex, Paperless-ngx und Home Assistant werden nach ihrer tatsächlichen Aufgabe eingeordnet. Du musst sie nicht alle installieren.

Der durchgängige Weg beginnt mit einem ungefährlichen lokalen Versuch. Danach folgen ein erster vollständiger Dienst, dessen Sicherung und eine erprobte Rückkehr. Erst wenn dieser kleine Aufbau funktioniert, nimmst du weitere Anwendungen und Menschen hinzu. Ein selbst betriebener Server soll deinen Alltag unterstützen und auch nach einem Fehler verständlich bleiben.

Absolute Unabhängigkeit entsteht dabei nicht. Strom, Internetanschluss, Hardware, Softwareprojekte und gegebenenfalls externe Namens- oder Vermittlungsdienste bleiben beteiligt. Du kannst diese Abhängigkeiten bewusst wählen und für wichtige Ausfälle vorsorgen. Eine gute Private Cloud erkennst du daran, dass du ihre Daten findest, ihre Zugänge erklären und einen brauchbaren Stand zurückholen kannst.

Danilo Sieren

# So arbeitest du mit diesem Buch

## Der Lernweg vom Versuch zum Betrieb

Lies zunächst die Planungs- und Grundlagenkapitel. Sie verhindern, dass du eine fertige Anwendung mit einer vollständig betreuten Cloud verwechselst. Die späteren Dienstkapitel lassen sich nach Bedarf lesen. Für jeden dauerhaft übernommenen Dienst bearbeitest du zusätzlich die Kapitel zu Sicherung, Updates und Wiederherstellung.

Mira und Paul begleiten dich als erfundene Beispielpersonen. Mira möchte Familienfotos und Dokumente ordnen. Paul betreut den Server und lernt, daraus einen nachvollziehbaren Betrieb zu machen. Ihre Beispiele beschreiben mögliche Abläufe und keine gemessenen Produktleistungen. Alle Übungsdateien enthalten erfundene Angaben.

## Beispiele und Befehle richtig verwenden

Befehle stehen in eigenen Druckblöcken. Die Zeichen am Anfang einer Zeile gehören zum Beispiel; zusätzliche Terminalaufforderungen wie ein Dollarzeichen werden nicht mitgedruckt. Mehrzeilige Shellbefehle verwenden gegebenenfalls einen Rückstrich als Fortsetzung. YAML benötigt Einrückungen mit Leerzeichen. Erstelle solche Dateien mit einem Texteditor als normalen Text.

Arbeite im Lernbereich mit neuen Namen und entbehrlichen Daten. Ein vorhandener wichtiger Ordner darf nicht nur deshalb überschrieben werden, weil sein Name einem Beispiel ähnelt. Beispieladressen unter example.org oder example.invalid sind Platzhalter. Setze echte Zugangsdaten nur dort ein, wo die

konkrete Anwendung sie verlangt, und übernimm sie nicht in öffentliche Hilfefragen.

## Mit unterschiedlichen Plattformen umgehen

Der Mac bleibt als Einstieg und Verwaltungsrechner Bestandteil des Buches. Ein Linux-Server oder eine Linux-VM kann für bestimmte Dienste besser zum offiziell vorgesehenen Betrieb passen. Docker Desktop auf macOS führt Linux-Container in einer virtuellen Maschine aus; Netzwerk, Dateizugriff und Hardwarebeschleunigung unterscheiden sich deshalb von einem nativen Linux-Host. [D01, D02]

Menünamen hängen von Sprache und Version ab. Englische Bezeichnungen werden dort genannt, wo sie den dokumentierten Eintrag eindeutig machen. Nutze für die tatsächlich installierte Ausgabe zusätzlich deren eigene Anleitung. Die Quellenkennungen führen zur Originaldokumentation am Ende des Buches.

# Inhalt

Vorwort ..... 2
So arbeitest du mit diesem Buch ..... 3
1 Deine Private Cloud sinnvoll begrenzen ..... 7
2 Hardware nach deinen Arbeitslasten auswählen ..... 11
3 Speicher und Stromversorgung vorbereiten ..... 15
4 Dein Netz verstehen und dokumentieren ..... 19
5 Den Host sicher einrichten ..... 23
6 Container und ihre Grenzen verstehen ..... 27
7 Ein vollständiges Compose Lernprojekt aufbauen ..... 31
8 Dauerhafte Daten und Geheimnisse verwalten ..... 35
9 Zugänge mit VPN und HTTPS planen ..... 39
10 Reverse Proxy und Zertifikate betreuen ..... 43
11 Nextcloud als Dateidienst aufbauen ..... 47
12 Dateien Kalender und Kontakte übernehmen ..... 51
13 Fotos mit Immich geordnet übernehmen ..... 55
14 Fotoarchive pflegen und wiederherstellen ..... 59
15 Passwörter mit einem klaren Notfallweg verwalten ..... 63
16 DNS mit Pi hole gezielt filtern ..... 67
17 Rekursive Auflösung und Netzausfälle beherrschen ..... 71
18 Medien mit Jellyfin und Plex bereitstellen ..... 75
19 Paperless ngx als Dokumentenarchiv einführen ..... 79
20 Home Assistant als eigenen Dienst betreuen ..... 83
21 Sicherungen nach ihrem Schutzziel planen ..... 87
22 Mit restic einen Rückweg erproben ..... 91
23 Datenbanken konsistent sichern ..... 95
24 Sicherungsabläufe zuverlässig automatisieren ..... 99

25 Updates mit einem Rückweg durchführen .....103
26 Auslastung und Funktionsfähigkeit überwachen .....107
27 Nach einem Ausfall geordnet zurückkehren .....111
28 Die Familiencloud langfristig betreuen .....115
29 Einen zweiten Sicherungsstandort betreiben .....119
30 Kapazität und Wiederanlauf nachvollziehbar messen .....123
Anhang A Vorlagen für deinen Cloudbetrieb .....127
Anhang B Ein Cloudprojekt vollständig durchspielen.....135
Anhang C Dateiintegrität mit einem kleinen Prüflabor verstehen..142
Anhang D Fehlerfälle aus dem Cloudalltag.....151
Anhang E Containerdaten und Diagnose praktisch vertiefen .....162
Anhang F Begriffe und Entscheidungen nachschlagen .....170
Quellen und offizielle Dokumentation.....175
Ausführliches Inhaltsverzeichnis .....179
Stichwortverzeichnis.....188
Impressum.....192

# 1 Deine Private Cloud sinnvoll begrenzen

## 1.1 Den Nutzen an einer Aufgabe festmachen

Beginne mit einem konkreten Bedarf. Möchtest du Fotos vom Telefon übernehmen, Unterlagen zwischen zwei Rechnern bearbeiten oder einen gemeinsam genutzten Kalender führen? Diese Aufgaben benötigen verschiedene Programme und unterschiedliche Prüfungen. Eine lange Liste verfügbarer Anwendungen ersetzt keine Entscheidung über die erste Aufgabe.

Schreibe einen Satz, der das erwartete Ergebnis beschreibt: Mira soll ein neues Foto auf dem Telefon aufnehmen und später am Rechner in voller Auflösung öffnen können. Dazu kommen ein klarer Speicherort, ein erlaubter Zugang und eine unabhängige Sicherung. Erst diese Ergänzungen machen aus einem gelungenen Upload einen betreubaren Ablauf.

Lass zunächst Dienste weg, deren Nutzen du nicht erklären kannst. Ein zusätzlicher Container braucht Updates, Speicher und einen Wiederherstellungsweg. Gerade bei einem kleinen Haushalt ist ein verständlicher Aufbau oft wertvoller als eine große Sammlung selten genutzter Oberflächen.

## 1.2 Kontrolle und Verantwortung zusammen betrachten

Eigene Hardware gibt dir unmittelbaren Einfluss auf Installation und Speicher. Sie schützt aber nicht automatisch vor Schadsoftware, Fehlbedienung oder fremdem Zugriff. Ein öffentlich erreichbarer kleiner Server kann ebenso automatisiert angegriffen werden wie ein großer Dienst. Seine geringe Bekanntheit ist kein Schutzkonzept.

Auch bei lokal gespeicherten Inhalten können Daten das Haus verlassen. Dazu gehören freigegebene Downloads, Vorschaudienste, Kartenanbieter, E-Mail-Benachrichtigungen und Verbindungen einer Erweiterung. Prüfe Datenwege je Funktion. Der Ausdruck selbst gehostet beschreibt zunächst den Betrieb der Anwendung, nicht sämtliche beteiligten Dienste.

Du kannst Aufgaben bewusst auslagern. Eine verschlüsselte Sicherung an einem anderen Standort oder ein betreuter Passwortdienst kann Teil eines durchdachten Plans sein. Entscheidend ist, welche Abhängigkeit du damit eingehst und wie du bei deren Ausfall weiterkommst.

## 1.3 Kosten vollständig rechnen

Erfasse Anschaffung, Ersatzteile, Strom, Sicherungsspeicher, Domain und gegebenenfalls Softwarelizenzen. Auch ein vorhandener Rechner hat einen Verbrauch und kann während des Dauerbetriebs früher ersetzt werden müssen. Deine Zeit lässt sich separat als Aufwand dokumentieren, selbst wenn du ihr keinen Geldbetrag zuordnen möchtest.

Die Stromrechnung folgt einer einfachen Beziehung: mittlere Leistung in Watt geteilt durch tausend, multipliziert mit Betriebsstunden und deinem Arbeitspreis pro Kilowattstunde. Bei erfundenen 18 Watt Dauerleistung sind es jährlich 157,68 Kilowattstunden. Mit einem ausdrücklich beispielhaften Preis von 0,35 Euro ergibt das 55,19 Euro. Netzteile, Platten und zusätzliche Netzwerktechnik gehören zur Messung.

Vergleiche denselben Leistungsumfang. Ein großer Anbieter kann Ausfallschutz und Support enthalten, während dein Heimserver zunächst nur eine Platte und eine verantwortliche Person besitzt. Eine sofortige Amortisation lässt sich daraus nicht allgemein ableiten. Kündige bestehende Dienste erst nach einer erfolgreichen Übernahme.

## 1.4 Die wichtigsten Datenklassen unterscheiden

Ordne deine Inhalte nach Bedeutung und Änderungsverhalten. Ein ersetzbarer Film, ein selbst aufgenommenes Video und eine Passwortdatenbank verlangen verschiedene Prioritäten. Eine aktive Datenbank benötigt zudem einen anderen Sicherungsablauf als eine geschlossene Textdatei.

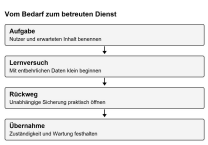
Notiere zu jeder Klasse die maßgebliche Fassung. Beim Telefonfoto kann das zunächst das Original auf dem Gerät sein. Bei einer gemeinschaftlich bearbeiteten Datei ist es später die bestätigte Serverfassung. Zwei unbemerkt parallel bearbeitete Bestände sind keine zuverlässige Redundanz.

Ergänze den zulässigen Verlustzeitraum. Wenn ein Fotoarchiv monatlich wächst, ist eine andere Sicherungshäufigkeit möglich als bei täglich neu eingescannten Rechnungen. Entscheide diesen Zeitraum bewusst, statt ihn zufällig aus einem vorinstallierten Zeitplan zu übernehmen.

## 1.5 Einen kleinen ersten Betrieb festlegen

Für den Einstieg genügt ein Dienst mit wenigen Testdaten und einem Benutzer. Definiere davor, wer Änderungen vornimmt, wo die Beschreibung liegt und wann der Versuch endet. Ein Experiment kann scheitern, ohne dass dadurch private Originale betroffen sind.

Paul beginnt mit einem lokalen Webdienst ohne persönliche Daten. Danach nimmt er eine Anwendung für Miras Dokumente hinzu. Er ergänzt noch keine Fotosammlung und keine Haussteuerung. So kann er Fehler einem begrenzten Aufbau zuordnen und den Sicherungsweg verstehen, bevor mehrere Anwendungen gleichzeitig wachsen.



## 1.6 Werkstatt Deinen Auftrag auf eine Seite schreiben

Beschreibe erste Aufgabe, beteiligte Geräte, verantwortliche Person und erwartetes Ergebnis. Nenne außerdem die Daten, die im Versuch ausdrücklich entbehrlich sind, sowie den getrennten Sicherungsort für spätere Originale.

Füge ein Ende hinzu: Welche Prüfung erlaubt die Übernahme in den Alltag, und bei welchem offenen Problem bleibst du beim bisherigen Dienst? Lies die Beschreibung am nächsten Tag erneut. Wenn du ohne zusätzliche Erklärung beginnen kannst, ist der Auftrag ausreichend konkret.

# 2 Hardware nach deinen Arbeitslasten auswählen

## 2.1 Vorhandene Geräte prüfen

Inventarisiere Rechner, Prozessorarchitektur, Arbeitsspeicher, Speichermöglichkeiten und unterstütztes Betriebssystem. Ein funktionsfähiger älterer Rechner kann einen Versuch gut tragen. Für den Dauerbetrieb zählen zusätzlich verfügbare Sicherheitsupdates, Lüftung, Netzteilzustand und der Wiederanlauf nach einer Unterbrechung.

Ein Notebook ist nicht automatisch der geeignete unbeaufsichtigte Server. Reisen, zugeklappter Betrieb, Akkuzustand und wechselnde Netze können seine Erreichbarkeit verändern. Nutzt du es trotzdem, behandle dies als bewusste Betriebsentscheidung und teste genau diese Zustände. Ein täglicher Arbeitsrechner und eine Familiencloud teilen sich außerdem Ausfälle und Wartung.

Bewahre die bisherigen Daten vor einem Betriebssystemwechsel unabhängig auf. Die Tatsache, dass ein Rechner künftig Server wird, ist keine Erlaubnis, seine noch benötigten Arbeitsdateien zu löschen. Ein neuer Datenträger oder ein klar getrennter Versuch kann die Umstellung vereinfachen.

## 2.2 Mac und Linux sachlich vergleichen

Ein Mac kann als Verwaltungsrechner, Testhost und für passende Dienste sinnvoll sein. Seine Eignung folgt jedoch nicht allein aus dem Chipnamen. Docker Desktop benötigt ein unterstütztes macOS und eine passende Ausgabe für Apple Silicon oder Intel. Die Herstelleranforderung von mindestens vier Gigabyte Arbeitsspeicher ist keine Kapazitätszusage für sämtliche Cloud-Anwendungen. [D01]

Für Immich empfiehlt das Projekt aktuell ein Linux- beziehungsweise Unix-artiges 64-Bit-System und rät von nicht auf Linux basierenden Docker-Umgebungen deutlich ab. Die Anforderungen nennen mindestens sechs, empfohlen acht Gigabyte RAM sowie mindestens zwei, empfohlen vier CPU-Kerne. Das ist eine wichtige Grundlage für die Auswahl und widerspricht einer pauschalen Vier-Gigabyte-Konfiguration für den ganzen Server. [D20]

Ein kleiner Linux-Rechner kann deshalb der eigentliche Diensthost sein, während du ihn bequem vom Mac aus verwaltest. Ein NAS ist eine weitere Möglichkeit, sofern Herstellerplattform, Containerunterstützung und die jeweilige Projektanleitung zusammenpassen. Prüfe solche Voraussetzungen am genauen Modell.

## 2.3 Prozessorarchitektur und Beschleunigung trennen

Die Bezeichnungen amd64 und arm64 beschreiben unterschiedliche Prozessorplattformen. Ein Image muss die verwendete Plattform unterstützen. Emulation kann zusätzliche Möglichkeiten schaffen, verändert aber Leistung und Fehlersuche. Wähle nach Möglichkeit native, offiziell angebotene Images.

Ein Mehrkernprozessor ersetzt keine fehlende Hardwareweitergabe. Grafikbeschleunigung unter macOS steht einem Linux-Container nicht automatisch in derselben Form zur Verfügung. Ob Video oder Bilderkennung die vorhandene Hardware nutzen, prüfst du je Anwendung und Betriebsart.

Bei älteren x86-Rechnern reicht die Angabe 64 Bit nicht immer aus. Immich nennt für den Machine-Learning-Container der Version-3-Reihe auf amd64 mindestens die Mikroarchitekturstufe x86-64-v2. Eine virtuelle Maschine muss passende CPU-Merkmale ebenfalls bereitstellen. Lies solche Bedingungen vor einem Hardwarekauf. [D20]

## 2.4 Arbeitsspeicher mit Reserve planen

Erfasse Dienste einschließlich Datenbank, Vorschaubilderzeugung und Hintergrundaufgaben. Ein weitgehend ruhendes Dateisystem kann sparsam arbeiten, während der erste Fotoimport kurzzeitig erheblich mehr Ressourcen braucht. Miss mindestens Alltag, Import und Sicherung als getrennte Zustände.

Teile einer virtuellen Maschine nicht gedankenlos den gesamten physischen Speicher zu. Host und weitere Anwendungen benötigen ebenfalls Platz. Häufige Auslagerung kann einen Engpass anzeigen; sie ist aber nicht automatisch der Beweis, dass eine einzelne Anwendung fehlerhaft ist.

Plane Wachstum in nachvollziehbaren Schritten. Wenn Bilderkennung und Videoverarbeitung gleichzeitig laufen, kann ihre zeitliche Trennung ausreichen. Mehr RAM hilft nicht gegen eine begrenzte Uploadleitung oder eine fehlerhafte Platte. Notiere den beobachteten Engpass vor jeder Aufrüstung.

## 2.5 Anschluss und Gehäuse prüfen

Achte bei externem Speicher auf Stromversorgung, Kühlung, Kabel und das Verhalten nach einem Neustart. Ein Stecker mit USB-C-Form sagt allein noch nichts über die tatsächlich verwendete Übertragungsart. Prüfe die gesamte Verbindung vom Host bis zum Laufwerk.

Teste einen längeren Lese- und Schreibvorgang mit entbehrlichen Daten. Ein kurzer erfolgreicher Kopiervorgang deckt weder thermische Probleme noch Abbrüche unter gleichzeitiger Belastung zuverlässig auf. Kontrolliere anschließend Dateiinhalte und Systemmeldungen.

Kaufe Gehäuse nach Wartbarkeit und passenden Eigenschaften. Ein Markenname ist keine allgemeine Zuverlässigkeitsgarantie. Falls ein Gehäuse eigene Verschlüsselung oder RAID-Metadaten

verwendet, gehört dessen Wiederherstellungsweg in die Planung. Die Platte muss außerhalb des ursprünglichen Gehäuses nicht automatisch lesbar sein.

## 2.6 Werkstatt Eine begründete Gerätewahl treffen

Erstelle zwei mögliche Aufbauten mit denselben Aufgaben. Erfasse vorhandene Teile, notwendige Ergänzungen und bekannte Einschränkungen. Rechne den Stromverbrauch mit einer gemessenen oder ausdrücklich geschätzten Leistung und kennzeichne den Unterschied.

Wähle den kleineren Aufbau, der deine erste Aufgabe nachvollziehbar erfüllt und einen Sicherungsweg zulässt. Notiere, welcher konkrete neue Bedarf später eine Erweiterung auslösen würde. Dadurch wird aus einer allgemeinen Kaufempfehlung ein zu deinem Haushalt passender Plan.

# 3 Speicher und Stromversorgung vorbereiten

## 3.1 Daten nach ihrer Rolle ablegen

Trenne Anwendungskonfiguration, aktive Datenbank, persönliche Inhalte und Sicherungsarchive in deiner Beschreibung. Sie müssen nicht zwingend auf vier verschiedenen Platten liegen. Die Trennung hilft zunächst zu verstehen, was bei einem Verlust tatsächlich fehlt und welche Bestandteile gemeinsam wiederhergestellt werden müssen.

Eine SSD kann für aktive Datenbanken und viele kleine Zugriffe sinnvoll sein. Große weitgehend ruhende Inhalte können auf einer geeigneten HDD liegen. Daraus entsteht keine allgemeine Regel, dass jede HDD langsam oder jede SSD zuverlässig genug wäre. Prüfe Latenz, Durchsatz, Reserven und die vorgesehenen Dateisystemeigenschaften.

Lege wichtige Daten nicht nur in einen flüchtigen Containerbereich. Dauerhafte Speicherorte werden ausdrücklich eingebunden oder als verwaltete Volumes eingerichtet. Ihre tatsächliche Lage hängt vom Host ab. Unter Docker Desktop liegen verwaltete Volumes gewöhnlich in der Linux-VM und nicht als normaler Ordner neben deiner Compose-Datei. [D03]

## 3.2 Dateisystem und Berechtigungen zusammen prüfen

Wähle für aktive Serverdaten ein zur Plattform und Anwendung passendes Dateisystem. Austauschformate für wechselnde Rechner sind nicht automatisch für Datenbanken und Linux-Rechte geeignet. Auch Netzwerkfreigaben können eigene Grenzen bei Sperren, Dateinamen und Metadaten besitzen.

Lass Datenbanken nur auf Speicher liegen, den die Projektanleitung unterstützt. Immich schließt Netzwerkfreigaben

für seinen Datenbankpfad ausdrücklich aus. Eine große NAS-Freigabe als Ablage für Originalfotos beantwortet daher nicht die Frage, wo PostgreSQL seine aktiven Daten halten darf. [D21]

Prüfe den Benutzerkontext. Eine Anwendung kann einen im Finder sichtbaren Ordner wegen fehlender Freigabe an die VM oder wegen anderer Rechte trotzdem nicht verwenden. Erkläre die Ursache, bevor du weitreichende Rechte öffnest. Ein rekursiver Vollzugriff für alle ist kein geeigneter Standard.

## 3.3 RAID und Sicherung unterscheiden

Ein gespiegelter Plattenverbund kann den Weiterbetrieb nach bestimmten Laufwerksausfällen erleichtern. Er verteilt aber auch beabsichtigte und versehentliche Schreibvorgänge. Eine irrtümliche Löschung bleibt eine Löschung, selbst wenn sie zuverlässig auf zwei Laufwerken ankommt.

Definiere, gegen welches Ereignis eine zweite Kopie schützt. Eine zusätzliche Partition auf derselben Platte teilt deren Ausfall. Zwei Laufwerke am selben Netzteil können weitere gemeinsame Risiken haben. Eine zweite Kopie im selben Synchronisationskreis kann Änderungen und Schadsoftware übernehmen.

Für einen kleinen Start kann ein einzelnes Datenlaufwerk mit unabhängigen, erprobten Sicherungen vertretbar sein. Du akzeptierst dann eine Unterbrechung beim Laufwerksausfall. Wenn sofortige Verfügbarkeit nötig ist, kommt Redundanz zusätzlich zur Sicherung hinzu. Vermische diese Ziele nicht in einer einzigen Zahl von Platten.

## 3.4 Verschlüsselung und Wiederanlauf planen

Verschlüsselung schützt bestimmte Daten im gesperrten Zustand. Sie ersetzt weder Rechte im laufenden System noch die Sicherung. FileVault betrifft das geschützte Mac-Startvolume; externe Laufwerke und Sicherungsziele müssen gesondert

betrachtet werden. Bewahre Wiederherstellungsinformationen unabhängig vom verschlüsselten Rechner auf. [D04]

Ein unbeaufsichtigter Neustart kann vor der Entsperrung stehen bleiben. Teste den echten Kaltstart mit der vorgesehenen Verschlüsselung, den externen Laufwerken und deiner Containerlaufzeit. Ein einmalig erfolgreich geplanter Neustart beweist noch keinen Wiederanlauf nach jedem Stromverlust.

Verzichte auf einen universellen Entsperrbefehl aus einer fremden Anleitung. Unterstützung und Voraussetzungen hängen von Hardware und Betriebssystem ab. Plane im Zweifel einen berechtigten lokalen Zugriff. Sicherheit und unbeaufsichtigte Verfügbarkeit müssen gemeinsam entschieden werden.

## 3.5 Stromausfälle und Wärme berücksichtigen

Eine USV kann eine kurze Unterbrechung überbrücken und bei passender Unterstützung ein geordnetes Herunterfahren ermöglichen. Ihre Eignung hängt von Last, Laufzeit, Schnittstelle, Akku und Abschaltverfahren ab. Eine feste Preis- oder Laufzeitangabe passt nicht zu jedem Aufbau.

Teste die Signalkette, ohne wichtige Schreibvorgänge zu unterbrechen. Server und Laufwerke benötigen ausreichend Zeit. Router und Switch können ebenfalls beteiligt sein. Ein noch laufender Host hilft wenig, wenn die Netzwerkverbindung bereits ausgefallen ist und die Steuerung darüber erfolgen sollte.

Sorge nach Herstellerangaben für freie Lüftungswege und einen geeigneten Aufstellort. Ein geschlossener Schrank kann problematisch sein. Reinigungs- und Aufstellungstipps müssen zum genauen Gerät passen; ein pauschales Hochkantstellen oder Ausblasen ist keine Voraussetzung für einen guten Server.

## 3.6 Werkstatt Einen Neustart vollständig abnehmen

Fahre deinen Testaufbau geordnet herunter und starte ihn anschließend wieder. Prüfe Entsperrung, Einbindung der Laufwerke, verfügbare Pfade, Containerlaufzeit und die tatsächliche Öffnung einer Testdatei.

Notiere auch die notwendige menschliche Handlung. Falls eine Person einen Datenträger entsperren muss, gehört das in den Wiederanlaufplan. Der Versuch ist erst abgeschlossen, wenn die Anwendung ihre Daten wiederfindet und der Sicherungsweg anschließend erneut funktioniert.

# 4 Dein Netz verstehen und dokumentieren

## 4.1 Adresse Name und Dienst auseinanderhalten

Ein Rechner benötigt eine erreichbare Adresse, eine passende Route und einen tatsächlich laufenden Dienst. Ein Name erleichtert die Nutzung, ist aber eine zusätzliche Auflösungsschicht. Wenn cloud.home.arpa nicht funktioniert, kann der Fehler daher beim DNS liegen, obwohl der Server unter seiner Adresse erreichbar bleibt.

Zeichne zunächst Router, Server, Verwaltungsrechner und Sicherungsziel. Ergänze getrennte Netze, Gast-WLAN und gegebenenfalls einen VPN-Zugang. Eine Verbindung in der Zeichnung beschreibt, wer welchen Dienst erreichen soll. Sie darf nicht automatisch bedeuten, dass alle Geräte einander vollständig vertrauen.

Für einen stationären Server ist eine nachvollziehbare Adressvergabe hilfreich. Eine DHCP-Reservierung im Router kann dazu dienen, ohne jedes Gerät separat zu konfigurieren. Prüfe, welche Schnittstelle und Kennung der Router verwendet. Nach einem Adapterwechsel kann eine alte Reservierung nicht mehr zum tatsächlichen Gerät passen.

## 4.2 Lokale Namen bewusst wählen

Für private Heimnetznamen ist home.arpa als spezieller Namensraum festgelegt. Ein Beispiel ist cloud.home.arpa. Der Name muss trotzdem von deinem vorgesehenen lokalen DNS aufgelöst werden. Er stellt weder von selbst einen Dienst bereit noch erzeugt er ein öffentlich vertrauenswürdiges Zertifikat. [D05]

Die Endung .local wird häufig für Multicast-Namensauflösung verwendet. Nutze sie nicht gedankenlos als beliebige eigene DNS-Zone. Ein Name, der im eigenen WLAN funktioniert, kann im Gastnetz oder über einen Tunnel anders behandelt werden.

Wenn du eine eigene öffentliche Domain verwendest, dokumentiere ihre Zuständigkeit und Verlängerung. Interne und externe Antworten können verschieden sein. Diese bewusste Aufteilung wird oft Split DNS genannt. Teste beide Wege mit dem tatsächlichen Client, statt nur den Eintrag im Verwaltungsfenster zu betrachten.

## 4.3 Bandbreite richtig einordnen

Die beworbene Geschwindigkeit eines Anschlusses ist nicht die Übertragungsrate einer bestimmten Datei. Netzwerkverkehr benötigt zusätzliche Protokolldaten; WLAN, Datenträger, Verschlüsselung und Gegenstelle können begrenzen. Für den Zugriff von außen ist außerdem der Upload deines Hauses wichtig.

Rechne mit den richtigen Einheiten. Acht Bit ergeben ein Byte. Eine Leitung mit theoretisch 40 Megabit pro Sekunde entspricht damit höchstens fünf Megabyte pro Sekunde vor weiteren Verlusten. Ein erfundener Upload von 100 Gigabyte würde bei konstant fünf Megabyte pro Sekunde etwa 5,56 Stunden benötigen. Reale Übertragungen können deutlich länger dauern.

Miss am vorgesehenen Aufstellort und mit einem typischen Client. Ein kabelgebundener Server vermeidet manche Funkprobleme, beseitigt aber keinen Engpass auf dem Telefon. Wenn ein lokaler Test schnell und ein entfernter langsam ist, untersuchst du die unterschiedlichen Abschnitte.

## 4.4 IPv4 IPv6 und Anschlussgrenzen beachten

Ein Heimanschluss kann eine öffentlich nutzbare IPv4-Adresse besitzen, hinter einer zusätzlichen Anbieterschicht liegen oder