17.4 Logs, Monitoring und Dienstprüfung in der Praxis

Logs, Monitoring und Dienstprüfung gehören zu den wichtigsten Werkzeugen in der Systemadministration.

Sie helfen dabei,

- Fehler zu erkennen
- Angriffe zu erkennen
- Ursachen einzugrenzen
- Änderungen nachzuvollziehen
- Dienste zu überwachen
- Ausfälle schneller zu beheben
- Sicherheitsvorfälle zu untersuchen
- Betrieb zu dokumentieren

Merksatz:

Logs zeigen,
was passiert ist.
Monitoring zeigt,
was gerade passiert oder sich anbahnt.

Warum Logs wichtig sind

Logs protokollieren Ereignisse auf Systemen, Diensten, Anwendungen und Netzwerkkomponenten.

Beispiele:

Benutzer meldet sich an.
Dienst startet.
Dienst stoppt.
Fehler tritt auf.
Firewall blockiert Verbindung.
Backup schlägt fehl.
Admin ändert Konfiguration.
SSH-Login wird abgelehnt.

Webserver liefert Fehlercode.
Datenbank meldet Verbindungsfehler.

Merksatz:

Ohne Logs ist Fehlersuche oft nur Raten.

Warum Monitoring wichtig ist

Monitoring überwacht den Zustand von Systemen und Diensten.

Typische Fragen:

Ist der Server erreichbar?
Läuft der Dienst?
Ist die Festplatte voll?
Ist die CPU überlastet?
Ist genug Arbeitsspeicher frei?
Ist das Backup erfolgreich?
Antwortet die Webseite?
Läuft das Zertifikat bald ab?
Gibt es ungewöhnlich viele Loginfehler?

Merksatz:

Monitoring erkennt Probleme früh,
bevor Benutzer sie melden.

Logs und Monitoring unterscheiden

Bereich	Zweck	Beispiel
Logs	Ereignisse nachvollziehen	Benutzerlogin fehlgeschlagen
Monitoring	Zustand überwachen	CPU dauerhaft bei 95 Prozent
Alerting	Verantwortliche alarmieren	Backup fehlgeschlagen
Reporting	Verlauf dokumentieren	Verfügbarkeit im Monat

Merksatz:

```
Logs erklären Ereignisse.
Monitoring beobachtet Zustände.
Alerting meldet kritische Abweichungen.
```

Typische Logquellen

Wichtige Logquellen in der Praxis:

```
Betriebssystemlogs
Anwendungslogs
Webserverlogs
Datenbanklogs
Authentifizierungslogs
SSH-Logs
RDP-Logs
Firewall-Logs
VPN-Logs
DNS-Logs
DHCP-Logs
E-Mail-Logs
Backup-Logs
Cloud-Audit-Logs
EDR-Logs
SIEM-Ereignisse
```

Merksatz:

```
Jede wichtige Komponente sollte nachvollziehbare Logs erzeugen.
```

Betriebssystemlogs

Betriebssystemlogs zeigen Ereignisse des Systems.

Beispiele:

```
Start und Stopp des Systems
Kernelmeldungen
Treiberprobleme
```

```
Benutzeranmeldungen
Dienststarts
Dienstfehler
Speicherprobleme
Hardwarefehler
```

Merksatz:

```
Betriebssystemlogs sind die Grundlage vieler Fehleranalysen.
```

Anwendungslogs

Anwendungslogs stammen von Programmen oder Diensten.

Beispiele:

```
Webanwendung meldet Fehler.
Datenbankverbindung schlägt fehl.
API gibt Fehler zurück.
Anwendung kann Datei nicht schreiben.
Lizenzprüfung schlägt fehl.
Benutzeraktion wird protokolliert.
```

Merksatz:

```
Anwendungslogs zeigen,
was innerhalb der Anwendung passiert.
```

Authentifizierungslogs

Authentifizierungslogs zeigen Anmeldeereignisse.

Beispiele:

```
erfolgreiche Anmeldung
fehlgeschlagene Anmeldung
gesperrtes Konto
falsches Passwort
```

```
unbekannter Benutzer
MFA-Fehler
Anmeldung von ungewöhnlichem Ort
Rechteerhöhung mit sudo
Adminlogin
```

Merksatz:

```
Authentifizierungslogs sind wichtig für Sicherheit und Zugriffskontrolle.
```

Firewall-Logs

Firewall-Logs zeigen, welcher Verkehr erlaubt oder blockiert wurde.

Typische Informationen:

```
Zeit
Quell-IP
Ziel-IP
Quellport
Zielport
Protokoll
Aktion
Regelname
Interface
Benutzer je nach System
```

Merksatz:

```
Firewall-Logs helfen bei Verbindungsproblemen und Angriffserkennung.
```

VPN-Logs

VPN-Logs zeigen Zugriffe über VPN.

Typische Informationen:

```
Benutzer
Anmeldezeit
Abmeldezeit
Quell-IP
zugewiesene VPN-IP
Authentifizierungsergebnis
MFA-Status
Verbindungsabbrüche
Fehlercodes
```

Merksatz:

```
VPN-Logs zeigen,
wer wann aus der Ferne verbunden war.
```

DNS-Logs

DNS-Logs zeigen Namensauflösungen.

Beispiele:

```
welcher Client fragt welchen Namen an?
welcher Name wird häufig abgefragt?
gibt es verdächtige Domains?
gibt es Namensauflösungsfehler?
nutzt ein Client ungewöhnliche DNS-Ziele?
```

Merksatz:

```
DNS-Logs helfen bei Fehlersuche und Malware-Erkennung.
```

DHCP-Logs

DHCP-Logs zeigen automatische IP-Vergaben.

Typische Informationen:

MAC-Adresse
vergebene IP-Adresse
Hostname
Lease-Zeit
Zeitpunkt
DHCP-Server
Fehler bei Vergabe

Merksatz:

DHCP-Logs helfen,
IP-Konflikte und Clientzuordnungen nachzuvollziehen.

Webserverlogs

Webserverlogs zeigen Zugriffe auf Webseiten und APIs.

Typische Informationen:

Client-IP
Zeitpunkt
HTTP-Methode
URL
Statuscode
Antwortgröße
User-Agent
Referrer
Antwortzeit

Merksatz:

Webserverlogs zeigen,
welche Anfragen ein Webdienst erhalten hat.

HTTP-Statuscodes in Logs

Wichtige HTTP-Statuscode-Gruppen:

Bereich	Bedeutung
2xx	erfolgreich
3xx	Weiterleitung
4xx	Clientfehler
5xx	Serverfehler

Beispiele:

```
200:
Anfrage erfolgreich

301 oder 302:
Weiterleitung

401:
nicht authentifiziert

403:
verboten

404:
nicht gefunden

500:
interner Serverfehler

502:
fehlerhafte Antwort vom Backend

503:
Dienst nicht verfügbar
```

Merksatz:

```
4xx deutet oft auf Client- oder Berechtigungsproblem.
5xx deutet oft auf Server- oder Backendproblem.
```

Backup-Logs

Backup-Logs zeigen, ob Sicherungen erfolgreich waren.

Zu prüfen:

```
Backup gestartet?
Backup abgeschlossen?
Datenmenge plausibel?
Fehler aufgetreten?
Ziel erreichbar?
Speicherplatz ausreichend?
Verschlüsselung aktiv?
Aufbewahrung korrekt?
Restore-Test durchgeführt?
```

Merksatz:

```
Backup gilt erst als zuverlässig,
wenn Erfolg und Wiederherstellung geprüft wurden.
```

Cloud-Audit-Logs

Cloud-Audit-Logs zeigen sicherheitsrelevante Aktionen in Cloud-Umgebungen.

Beispiele:

```
Benutzer angemeldet
Rolle geändert
Firewall-Regel geöffnet
Speicher öffentlich gemacht
API-Schlüssel erzeugt
VM gestartet
Datenbank gelöscht
Backup geändert
MFA deaktiviert
```

Merksatz:

```
Cloud-Audit-Logs zeigen,
wer was in der Cloud geändert hat.
```

Loglevel

Logs haben oft verschiedene Schweregrade.

Typische Loglevel:

Loglevel	Bedeutung
Debug	sehr detaillierte Informationen für Fehlersuche
Info	normale Betriebsinformation
Warning	Warnung, noch kein vollständiger Fehler
Error	Fehler ist aufgetreten
Critical	schwerer Fehler mit hoher Auswirkung
Fatal	Dienst oder Anwendung kann nicht weiterarbeiten

Merksatz:

```
Nicht jede Warnung ist ein Ausfall,
aber Warnungen können frühe Hinweise sein.
```

Debug-Logs

Debug-Logs sind sehr detailliert.

Vorteil:

```
helfen bei genauer Fehlersuche
```

Nachteil:

```
erzeugen viele Daten
können sensible Informationen enthalten
können Performance belasten
```

Merksatz:

```
Debug-Logging gezielt aktivieren
und danach wieder reduzieren.
```

Logrotation

Logrotation bedeutet:

```
Logdateien werden regelmäßig umbenannt,
archiviert
komprimiert
oder gelöscht.
```

Ziel:

```
Festplatte läuft nicht voll.
Logs bleiben übersichtlich.
Aufbewahrung wird kontrolliert.
```

Merksatz:

```
Ohne Logrotation können Logs Speicherplatzprobleme verursachen.
```

Logaufbewahrung

Logaufbewahrung legt fest, wie lange Logs gespeichert werden.

Zu beachten:

```
gesetzliche Vorgaben
Datenschutz
Sicherheitsanforderungen
Speicherplatz
Nachvollziehbarkeit
forensische Analyse
interne Richtlinien
```

Merksatz:

```
Logs so lange wie nötig,
aber nicht unbegrenzt ohne Grund speichern.
```

Datenschutz bei Logs

Logs können personenbezogene Daten enthalten.

Beispiele:

```
Benutzername
IP-Adresse
E-Mail-Adresse
Zeitpunkte
Geräteinformationen
aufgerufene URLs
Standortinformationen
```

Deshalb wichtig:

```
Zugriff begrenzen
Aufbewahrung regeln
Zweck festlegen
sensible Inhalte vermeiden
Protokollierung dokumentieren
```

Merksatz:

```
Logs sind sicherheitsrelevant,
aber auch datenschutzrelevant.
```

Zentrale Logsammlung

Zentrale Logsammlung bedeutet:

```
Logs mehrerer Systeme werden an einem zentralen Ort gesammelt.
```

Vorteile:

```
bessere Suche
bessere Korrelation
Schutz vor lokaler Manipulation
```

einfachere Auswertung
bessere Nachvollziehbarkeit
Grundlage für SIEM

Merksatz:

Zentrale Logs helfen,
Ereignisse systemübergreifend zu verstehen.

Syslog

Syslog ist ein verbreiteter Standard zur Übertragung von Logmeldungen.

Häufig genutzt bei:

Linux-Systemen
Netzwerkgeräten
Firewalls
Switches
Routern
Appliances

Merksatz:

Syslog transportiert Logmeldungen zu einem Logserver.

Windows-Ereignisanzeige

Windows-Systeme verwenden Ereignisprotokolle.

Typische Bereiche:

Anwendung
Sicherheit
System
Setup
weitergeleitete Ereignisse

Wichtige Ereignisse:

```
Anmeldung
Fehlanmeldung
Dienstfehler
Richtlinienänderung
Kontoänderung
Systemfehler
```

Merksatz:

```
Die Windows-Ereignisanzeige ist zentrale Logquelle auf Windows-Systemen.
```

---

journalctl

Auf vielen Linux-Systemen werden Systemlogs über systemd-journald verwaltet.

Das Werkzeug zur Anzeige heißt:

```
journalctl
```

Typische Nutzung:

```
Systemmeldungen anzeigen
Logs eines Dienstes prüfen
Logs seit bestimmtem Zeitpunkt prüfen
Bootvorgänge untersuchen
```

Merksatz:

```
journalctl zeigt systemd-Journal-Logs auf Linux-Systemen.
```

Dienstprüfung

Dienstprüfung bedeutet:

```
Es wird geprüft,
ob ein Dienst läuft,
erreichbar ist
und korrekt arbeitet.
```

Zu prüfen:

```
läuft der Dienst?
lauscht der Port?
ist die Konfiguration korrekt?
gibt es Fehlerlogs?
antwortet der Dienst?
erreicht er Backend-Systeme?
gibt es Berechtigungsprobleme?
wurde etwas geändert?
```

Merksatz:

```
Dienst läuft nicht automatisch gleich Dienst funktioniert vollständig.
```

Dienststatus

Der Dienststatus zeigt, ob ein Dienst gestartet, gestoppt oder fehlerhaft ist.

Beispiele:

```
running
stopped
failed
active
inactive
disabled
```

Merksatz:

```
Dienststatus ist der erste Blick,
aber nicht die vollständige Prüfung.
```

Portprüfung

Ein Dienst kann nur erreicht werden, wenn er auf einem Port lauscht und der Netzwerkweg erlaubt ist.

Zu prüfen:

```
lauscht der Dienst auf richtiger IP?
lauscht der Dienst auf richtigem Port?
blockiert lokale Firewall?
blockiert Netzwerkfirewall?
stimmt DNS?
stimmt Routing?
ist der Dienst nur lokal gebunden?
```

Merksatz:

```
Dienst erreichbar bedeutet:
Prozess,
Port,
Firewall,
DNS
und Routing müssen passen.
```

Dienst lauscht nur lokal

Ein Dienst kann nur auf localhost lauschen.

Beispiel:

```
127.0.0.1
```

Dann ist er nur vom eigenen System erreichbar.

Wenn andere Systeme zugreifen sollen, muss geprüft werden, ob der Dienst auf der passenden Netzwerkschnittstelle lauscht.

Merksatz:

```
127.0.0.1 bedeutet nur lokal auf demselben System.
```

Typische Dienstfehler

Häufige Ursachen:

```
Dienst nicht gestartet
falsche Konfiguration
Port bereits belegt
fehlende Berechtigung
Zertifikat abgelaufen
Datenbank nicht erreichbar
DNS-Fehler
Firewall blockiert
Speicher voll
falsche Zugangsdaten
Update hat Abhängigkeit geändert
```

Merksatz:

```
Dienstfehler entstehen oft durch Konfiguration,
Abhängigkeiten
Berechtigungen
oder Ressourcen.
```

Abhängigkeiten von Diensten

Viele Dienste hängen von anderen Diensten ab.

Beispiele:

```
Webanwendung braucht Datenbank.

Anwendung braucht Redis oder Cache.

Login braucht Verzeichnisdienst.

Backup braucht Speicherziel.

Monitoring braucht Netzwerkzugriff.
```

E-Mail-Dienst braucht DNS.

Merksatz:

Bei Fehlern immer auch abhängige Dienste prüfen.

Metriken im Monitoring

Metriken sind messbare Werte.

Typische Metriken:

CPU-Auslastung
RAM-Nutzung
Festplattenbelegung
I/O-Last
Netzwerklast
Paketverlust
Latenz
Antwortzeit
Fehlerquote
Anzahl Loginfehler
Queue-Länge
Verfügbarkeit

Merksatz:

Metriken zeigen messbare Zustände von Systemen und Diensten.

CPU-Auslastung

Hohe CPU-Auslastung kann bedeuten:

Dienst ist überlastet
Prozess hängt
Malware oder Cryptominer aktiv
zu viele Anfragen

schlecht optimierte Anwendung
Backup oder Scan läuft
Hardware zu schwach

Merksatz:

Hohe CPU ist ein Symptom,
nicht automatisch die Ursache.

RAM-Nutzung

Hohe RAM-Nutzung kann bedeuten:

Anwendung braucht viel Speicher
Speicherleck
zu viele Prozesse
Caching
falsche Dimensionierung
Containerlimit zu klein
Datenbank nutzt Speicher intensiv

Merksatz:

RAM-Auslastung immer im Zusammenhang mit Anwendung und System bewerten.

Festplattenbelegung

Volle Festplatten verursachen viele Probleme.

Mögliche Folgen:

Dienst stoppt
Datenbank kann nicht schreiben
Logs können nicht gespeichert werden
Updates schlagen fehl
Backup schlägt fehl
System wird instabil

Merksatz:

```
Volle Festplatte ist eine häufige Ursache für Dienstfehler.
```

I/O-Last

I/O beschreibt Ein- und Ausgabe auf Speichergeräten.

Hohe I/O-Last kann auftreten durch:

```
Datenbank
Backup
Virenscan
Logflut
große Dateioperationen
langsamen Speicher
zu viele gleichzeitige Zugriffe
```

Merksatz:

```
Hohe I/O-Last kann Systeme langsam machen,
auch wenn CPU frei ist.
```

Netzwerklast

Netzwerklast zeigt, wie stark eine Verbindung genutzt wird.

Hohe Netzwerklast kann entstehen durch:

```
Backup
Dateiübertragung
Streaming
DDoS
Malware
Synchronisation
falsche Schleife
große Updates
```

Merksatz:

Netzwerklast immer mit Quelle,
Ziel
und Anwendung betrachten.

Latenz und Paketverlust

Latenz:

Verzögerung bei der Übertragung

Paketverlust:

Pakete gehen verloren

Folgen:

langsame Verbindungen
Verbindungsabbrüche
schlechte Sprachqualität
langsame Remote-Sitzungen
Timeouts
schlechte Anwendungserfahrung

Merksatz:

Latenz und Paketverlust beeinflussen spürbar die Qualität von Netzwerkdiensten.

Antwortzeit

Antwortzeit beschreibt, wie lange ein Dienst für eine Antwort braucht.

Beispiele:

Webseite lädt langsam.
API antwortet verzögert.
Datenbankabfrage dauert lange.
Login dauert ungewöhnlich lange.

Merksatz:

```
Dienst ist erreichbar,
kann aber trotzdem zu langsam sein.
```

Verfügbarkeit

Verfügbarkeit beschreibt, ob ein Dienst nutzbar ist.

Monitoring kann prüfen:

```
Ping
TCP-Port
HTTP-Antwort
API-Antwort
Login-Test
Datenbankverbindung
Zertifikat
Backup-Erfolg
```

Merksatz:

```
Verfügbarkeit sollte aus Sicht der Nutzung geprüft werden.
```

Health Check

Ein Health Check prüft, ob ein Dienst gesund ist.

Beispiele:

```
Prozess läuft
Port offen
HTTP-Status 200
Datenbank erreichbar
Abhängigkeiten funktionieren
Antwortzeit im Grenzwert
```

Merksatz:

```
Ein guter Health Check prüft mehr als nur,
ob ein Prozess läuft.
```

Alerting

Alerting bedeutet:

```
Bei kritischen Zuständen wird automatisch alarmiert.
```

Beispiele:

```
Festplatte über 90 Prozent
Dienst nicht erreichbar
Backup fehlgeschlagen
Zertifikat läuft bald ab
viele Loginfehler
CPU dauerhaft sehr hoch
ungewöhnlicher Datenexport
Firewall blockiert ungewöhnlich viel
```

Merksatz:

```
Monitoring ohne sinnvolles Alerting wird leicht übersehen.
```

Alarmmüdigkeit

Alarmmüdigkeit entsteht, wenn zu viele unwichtige Alarme erzeugt werden.

Folge:

```
wichtige Alarme werden ignoriert.
```

Gegenmaßnahmen:

```
sinnvolle Schwellwerte
Prioritäten
Eskalationsregeln
```

Alarmzusammenfassung

regelmäßige Anpassung

klare Zuständigkeit

Merksatz:

Zu viele schlechte Alarme sind fast so problematisch wie keine Alarme.

Schwellwerte

Schwellwerte legen fest, wann ein Alarm ausgelöst wird.

Beispiele:

Festplatte über 90 Prozent

CPU länger als 10 Minuten über 95 Prozent

Zertifikat läuft in 14 Tagen ab

Backup zweimal hintereinander fehlgeschlagen

Merksatz:

Schwellwerte sollten zur Kritikalität des Dienstes passen.

SLA, SLO und SLI in der Praxis

SLI:

Messwert

SLO:

Zielwert

SLA:

```
vertragliche Zusage
```

Beispiel:

```
SLI:
gemessene Verfügbarkeit

SLO:
99,9 Prozent Verfügbarkeit als internes Ziel

SLA:
99,5 Prozent vertraglich zugesichert
```

Merksatz:

```
SLI misst.
SLO setzt Ziel.
SLA ist Zusage.
```

Uptime und Verfügbarkeit

Uptime beschreibt, wie lange ein System ohne Unterbrechung läuft.

Verfügbarkeit beschreibt, ob ein Dienst für Benutzer nutzbar ist.

Wichtig:

```
Ein Server kann laufen,
aber die Anwendung kann trotzdem nicht funktionieren.
```

Merksatz:

```
Uptime ist nicht automatisch Dienstverfügbarkeit.
```

Zertifikatsüberwachung

TLS-Zertifikate laufen ab.

Wenn ein Zertifikat abläuft, können Benutzer Warnungen erhalten oder Dienste funktionieren nicht mehr.

Monitoring sollte prüfen:

```
Ablaufdatum
Hostname passt
Zertifikatskette gültig
richtige Zertifizierungsstelle
automatische Erneuerung funktioniert
```

Merksatz:

```
Abgelaufene Zertifikate sind vermeidbare Ausfälle.
```

Backup-Monitoring

Backup-Monitoring prüft, ob Sicherungen erfolgreich laufen.

Zu überwachen:

```
letzter erfolgreicher Lauf
Fehlermeldungen
Datenmenge
Laufzeit
Speicherziel
verfügbare Kapazität
Verschlüsselung
Aufbewahrung
Restore-Test
```

Merksatz:

```
Backup-Erfolg muss aktiv überwacht werden.
```

Logauswertung bei Sicherheitsvorfällen

Bei Verdacht auf Sicherheitsvorfall helfen Logs.

Zu prüfen:

```
ungewöhnliche Loginzeiten
viele Fehlversuche
neue Adminrechte
neue Benutzer
neue Weiterleitungsregeln
Datenexporte
verdächtige DNS-Anfragen
ungewöhnliche Prozesse
Firewall-Drops
VPN-Zugriffe
EDR-Meldungen
```

Merksatz:

```
Sicherheitsvorfälle lassen sich nur gut untersuchen,
wenn Logs vorhanden und geschützt sind.
```

Zeitliche Korrelation

Zeitliche Korrelation bedeutet:

```
Ereignisse aus verschiedenen Logs werden zeitlich zusammengeführt.
```

Beispiel:

```
10:01 VPN-Login

10:03 Adminrechte vergeben

10:05 Zugriff auf Dateiserver

10:10 Datenexport
```

Merksatz:

```
Korrelation zeigt Zusammenhänge,
die einzelne Logs allein nicht zeigen.
```

NTP und Zeitbasis

NTP steht für:

```
Network Time Protocol
```

NTP synchronisiert Systemzeiten.

Warum wichtig?

```
Logs verschiedener Systeme müssen zeitlich vergleichbar sein.
```

Ohne korrekte Zeit:

```
Ereignisse sind schwer zuzuordnen.
Sicherheitsanalyse wird ungenau.
Nachweise werden schwieriger.
```

Merksatz:

```
Gemeinsame Zeitbasis ist wichtig für Logs und Forensik.
```

Manipulationsschutz für Logs

Logs können für Angreifer gefährlich sein, weil sie Spuren enthalten.

Deshalb versuchen Angreifer oft, Logs zu löschen oder zu verändern.

Schutz:

```
zentrale Logsammlung
eingeschränkte Rechte
unveränderliche Speicherung
regelmäßige Sicherung
```

getrennte Systeme
Alarm bei Logausfall
Auditierung

Merksatz:

Logs müssen vor Manipulation geschützt werden.

Dashboard

Ein Dashboard zeigt wichtige Zustände übersichtlich an.

Beispiele:

Systemstatus
Dienstverfügbarkeit
CPU und RAM
Netzwerkstatus
Backupstatus
Sicherheitsalarme
Zertifikatsstatus
offene Incidents

Merksatz:

Dashboard zeigt schnell,
wo Aufmerksamkeit nötig ist.

Runbook

Ein Runbook ist eine Schritt-für-Schritt-Anleitung für wiederkehrende Situationen.

Beispiele:

Dienst ausgefallen
Backup fehlgeschlagen
Zertifikat abgelaufen
Speicherplatz voll

Benutzerkonto gesperrt
VPN funktioniert nicht
Malware-Verdacht

Merksatz:

Runbooks helfen,
Fehler schneller und einheitlicher zu beheben.

Dokumentation bei Störungen

Bei Störungen sollte dokumentiert werden:

Zeitpunkt
betroffene Systeme
Symptome
Fehlermeldungen
Logs
getroffene Maßnahmen
Verantwortliche
Ursache
Lösung
Restprobleme
Lessons Learned

Merksatz:

Gute Dokumentation verhindert,
dass derselbe Fehler immer wieder neu gesucht wird.

Typische Praxisreihenfolge bei Dienstproblemen

Sinnvolle Reihenfolge:

1. Ist das Problem reproduzierbar?
2. Wer ist betroffen?
3. Seit wann besteht das Problem?

4. Gab es Änderungen?
5. Läuft der Dienst?
6. Lauscht der Port?
7. Ist DNS korrekt?
8. Ist Netzwerkverbindung möglich?
9. Blockiert eine Firewall?
10. Gibt es Fehlerlogs?
11. Sind Abhängigkeiten erreichbar?
12. Sind Ressourcen ausreichend?
13. Funktioniert Authentifizierung?
14. Ist ein Zertifikat abgelaufen?
15. Wurde die Lösung dokumentiert?

Merksatz:

Fehlersuche wird schneller,
wenn sie systematisch erfolgt.

Typische Fehler bei Logs und Monitoring

Häufige Fehler:

Logs nicht aktiviert
Logs nur lokal gespeichert
Logs werden zu früh gelöscht
Logs enthalten zu viele sensible Daten
keine zentrale Auswertung
keine Zeitsynchronisation
keine Alarmierung
zu viele unwichtige Alarme
Backupfehler werden nicht bemerkt
Zertifikatsablauf wird nicht überwacht
Dienst läuft,
aber Anwendung wird nicht geprüft
niemand ist für Alarme zuständig

Merksatz:

```
Monitoring muss betrieben und gepflegt werden,
sonst verliert es seinen Nutzen.
```

Checkliste: Logs und Monitoring sinnvoll einrichten

```
Wichtige Systeme erfassen.
Wichtige Dienste erfassen.
Relevante Logs aktivieren.
Logaufbewahrung festlegen.
Datenschutz beachten.
Zentrale Logsammlung nutzen.
Zeitsynchronisation per NTP sicherstellen.
Kritische Metriken überwachen.
Schwellwerte definieren.
Alarmwege festlegen.
Zuständigkeiten klären.
Backup-Erfolg überwachen.
Zertifikate überwachen.
Health Checks einrichten.
Runbooks erstellen.
Alarme regelmäßig prüfen.
Dokumentation aktuell halten.
```

Merksatz:

```
Logs und Monitoring brauchen Planung,
Zuständigkeit
und regelmäßige Pflege.
```

Typische IHK-Fragen

In AP1 und AP2 kann zum Beispiel gefragt werden:

```
- Warum sind Logs wichtig?
- Was ist der Unterschied zwischen Logging und Monitoring?
- Welche Logquellen gibt es?
- Was sind Authentifizierungslogs?
```

- Wofür nutzt man Firewall-Logs?
- Wofür nutzt man VPN-Logs?
- Was zeigen Webserverlogs?
- Was bedeuten HTTP-Statuscodes 4xx und 5xx?
- Was ist Logrotation?
- Warum ist Logaufbewahrung wichtig?
- Warum sind Logs datenschutzrelevant?
- Was ist zentrale Logsammlung?
- Was ist Syslog?
- Was ist journalctl?
- Was ist ein Health Check?
- Was ist Alerting?
- Was ist Alarmmüdigkeit?
- Warum ist NTP für Logs wichtig?
- Was ist ein Dashboard?
- Was ist ein Runbook?
- Warum sollte Backup-Monitoring eingerichtet werden?

Typische Prüfungsfallen

Logs und Monitoring nicht verwechseln.

Logs zeigen Ereignisse.

Monitoring zeigt Zustände.

Alerting meldet kritische Zustände.

Ohne Logs ist Sicherheitsanalyse schwierig.

Lokale Logs können manipuliert werden.

Zentrale Logsammlung verbessert Nachvollziehbarkeit.

Logs können personenbezogene Daten enthalten.

Logrotation verhindert volle Festplatten.

Debug-Logs nicht dauerhaft unnötig aktiv lassen.

Dienst läuft nicht automatisch gleich Anwendung funktioniert.

4xx bedeutet meist Client- oder Berechtigungsproblem.

5xx bedeutet meist Server- oder Backendproblem.

Uptime ist nicht automatisch Verfügbarkeit.

Health Check sollte mehr prüfen als Prozessstatus.

Backup-Erfolg aktiv überwachen.

Zertifikatsablauf überwachen.

NTP ist wichtig für zeitliche Korrelation.

Zu viele Alarme führen zu Alarmmüdigkeit.

Runbooks helfen bei wiederkehrenden Störungen.

Wichtige Begriffe kurz erklärt

Begriff	Kurze Erklärung
Log	protokolliertes Ereignis
Logging	Aufzeichnung von Ereignissen
Monitoring	Überwachung von Zuständen
Alerting	automatische Alarmierung
Reporting	Auswertung über Zeitraum
Betriebssystemlog	Log des Betriebssystems
Anwendungslog	Log einer Anwendung
Authentifizierungslog	Log zu Anmeldungen
Firewall-Log	Log zu erlaubtem und blockiertem Verkehr
VPN-Log	Log zu VPN-Verbindungen
DNS-Log	Log zu Namensauflösungen
DHCP-Log	Log zu IP-Vergaben

Begriff	Kurze Erklärung
Webserverlog	Log zu HTTP-Anfragen
HTTP-Statuscode	Ergebnis einer Webanfrage
Backup-Log	Log zu Sicherungen
Audit-Log	Log sicherheitsrelevanter Aktionen
Loglevel	Schweregrad einer Logmeldung
Debug	detaillierte Fehlersuche
Info	normale Information
Warning	Warnung
Error	Fehler
Critical	schwerer Fehler
Logrotation	Verwaltung alter Logdateien
Logaufbewahrung	Dauer der Speicherung
zentrale Logsammlung	Sammlung von Logs an zentralem Ort
Syslog	Standard für Logmeldungen
Ereignisanzeige	Windows-Loganzeige
journalctl	Werkzeug für systemd-Journal
Dienstprüfung	Prüfung eines Dienstes
Portprüfung	Prüfung erreichbarer Ports
Metrik	messbarer Wert
CPU-Auslastung	Prozessorlast
RAM-Nutzung	Arbeitsspeichernutzung
I/O-Last	Last durch Ein- und Ausgabe
Latenz	Verzögerung
Paketverlust	verlorene Netzwerkpakete
Antwortzeit	Dauer bis zur Antwort
Verfügbarkeit	Nutzbarkeit eines Dienstes
Health Check	Gesundheitsprüfung eines Dienstes
Schwellwert	Grenze für Alarm
SLI	Service-Messwert
SLO	Service-Zielwert
SLA	vertragliche Servicezusage
NTP	Zeitsynchronisation

Begriff	Kurze Erklärung
Dashboard	Übersicht wichtiger Zustände
Runbook	Schritt-für-Schritt-Anleitung

IHK-sichere Kurzformulierung

Logs protokollieren Ereignisse auf Systemen, Anwendungen, Netzwerkkomponenten und Sicherheitsdiensten. Monitoring überwacht aktuelle Zustände und Metriken wie Verfügbarkeit, CPU-Auslastung, Speicherplatz, Antwortzeiten, Backup-Erfolg und Zertifikatslaufzeiten. Alerting informiert Verantwortliche automatisch bei kritischen Abweichungen. Für eine sinnvolle Analyse sollten wichtige Logs zentral gesammelt, geschützt, mit korrekter Zeitbasis versehen und datenschutzgerecht aufbewahrt werden. Dienstprüfung umfasst die Kontrolle von Dienststatus, Port, Firewall, DNS, Routing, Logs, Ressourcen und Abhängigkeiten. Logs, Monitoring, Health Checks, Dashboards und Runbooks helfen, Fehler schneller zu erkennen, Sicherheitsvorfälle zu untersuchen und den Betrieb nachvollziehbar zu dokumentieren.

Merksätze

Logs zeigen Ereignisse.

Monitoring zeigt Zustände.

Alerting meldet kritische Abweichungen.

Ohne Logs ist Fehlersuche schwierig.

Ohne Monitoring werden Probleme oft zu spät erkannt.

Authentifizierungslogs zeigen Anmeldungen.

Firewall-Logs zeigen erlaubten und blockierten Verkehr.

VPN-Logs zeigen Fernzugriffe.

DNS-Logs helfen bei Namensauflösung und Sicherheit.

DHCP-Logs helfen bei IP-Zuordnung.

Webserverlogs zeigen HTTP-Anfragen.

2xx bedeutet erfolgreich.

3xx bedeutet Weiterleitung.

4xx bedeutet Client- oder Berechtigungsproblem.

5xx bedeutet Server- oder Backendproblem.

Backup-Logs regelmäßig prüfen.

Cloud-Audit-Logs zeigen Änderungen in der Cloud.

Debug-Logs nicht dauerhaft unnötig aktiv lassen.

Logrotation schützt vor voller Festplatte.

Logs können personenbezogene Daten enthalten.

Zentrale Logsammlung verbessert Nachvollziehbarkeit.

Syslog transportiert Logmeldungen.

Windows nutzt Ereignisprotokolle.

journalctl zeigt systemd-Journal-Logs.

Dienststatus ist nur der erste Schritt.

Port,
Firewall,
DNS
und Routing mitprüfen.

127.0.0.1 bedeutet nur lokal erreichbar.

Abhängige Dienste prüfen.

Metriken zeigen messbare Zustände.

Hohe CPU ist Symptom,

nicht automatisch Ursache.

Volle Festplatte verursacht viele Dienstfehler.

Uptime ist nicht automatisch Verfügbarkeit.

Health Check soll echte Nutzbarkeit prüfen.

Zu viele Alarme führen zu Alarmmüdigkeit.

NTP ist wichtig für Logauswertung.

Logs vor Manipulation schützen.

Dashboard schafft Überblick.

Runbooks beschleunigen wiederkehrende Fehlerbehebung.

Störungen dokumentieren.

Logs und Monitoring brauchen Pflege.