# 7.8 Server ist langsam

### Ziel dieser Seite

Diese Seite beschreibt die systematische Fehleranalyse, wenn ein Server oder ein darauf bereitgestellter Dienst ungewöhnlich langsam reagiert.

Dabei muss zwischen verschiedenen Ursachen unterschieden werden:

- CPU-Auslastung,
- Arbeitsspeichermangel,
- Paging oder Swapping,
- langsamer Datenträger,
- hohe Datenträgerlatenz,
- Netzwerkprobleme,
- überlastete Anwendung,
- Datenbanksperren,
- Hintergrundjobs,
- Hypervisor-Engpässe,
- externe Abhängigkeiten.

> „Der Server ist langsam“ ist lediglich eine Symptombeschreibung. Erst durch Messwerte lässt sich bestimmen, welche Ressource oder Abhängigkeit den Engpass verursacht.

---

### 1. Problem exakt beschreiben

Vor der technischen Analyse müssen Soll- und Istzustand dokumentiert werden.

| Angabe | Beispiel |
|---|---|
| betroffener Server | `SRV-APP01` |
| betroffener Dienst | interne Webanwendung |
| Symptom | Seitenaufbau dauert 15 Sekunden |
| Normalzustand | unter 2 Sekunden |
| Beginn | 02.08.2026, etwa 09:15 Uhr |
| Umfang | alle Benutzer |
| Häufigkeit | zeitweise |
| letzter funktionierender Zustand | vor dem nächtlichen Backup |

| Angabe | Beispiel |
| --- | --- |
| letzte Änderung | Anwendungsupdate |
| Vergleichssystem | `SRV-APP02` reagiert normal |

Zu klären ist:

- Ist der gesamte Server langsam oder nur eine Anwendung?
- Sind alle Benutzer betroffen?
- Tritt das Problem lokal und über das Netzwerk auf?
- Sind nur bestimmte Funktionen betroffen?
- Besteht das Problem dauerhaft oder nur zu bestimmten Zeiten?
- Trat das Problem nach einer Änderung auf?
- Sind weitere Server betroffen?
- Gibt es ein funktionierendes Vergleichssystem?
- Wie wird die Verzögerung gemessen?
- Welche Antwortzeit gilt als normal?

Die Aussage „der Server ist langsam“ reicht für eine gezielte Fehleranalyse nicht aus.

---

### 2. Server-, Anwendungs- und Netzwerkproblem unterscheiden

Ein langsamer Dienst beweist nicht automatisch, dass der Server selbst überlastet ist.

```
Client
  ↓
DNS
  ↓
Netzwerk
  ↓
Firewall oder Proxy
  ↓
Webserver
  ↓
Anwendung
  ↓
Datenbank
  ↓
Storage
```

Jede dieser Komponenten kann die beobachtete Verzögerung verursachen.

| Beobachtung | Wahrscheinlicher Prüfbereich |
|---|---|
| lokale Bedienung ebenfalls langsam | Serverressourcen oder Betriebssystem |
| nur entfernte Zugriffe langsam | Netzwerk, VPN, Firewall oder Proxy |
| nur eine Anwendung langsam | Anwendung, Datenbank oder Abhängigkeit |
| alle Dienste langsam | CPU, RAM, Storage, Hypervisor oder Netzwerk |
| nur bestimmte Abfragen langsam | Datenbank, Sperren oder fehlende Indizes |
| nur zu festen Uhrzeiten langsam | Backup, Scan, Wartung oder geplanter Job |
| nur eine VM betroffen | VM-Konfiguration oder Gastbetriebssystem |
| mehrere VMs auf demselben Host betroffen | Hypervisor oder gemeinsames Storage |
| Ping normal, Anwendung langsam | Dienst, TLS, Authentifizierung oder Backend |
| Ping langsam oder instabil | Netzwerkpfad oder Serverüberlastung |
| Zugriff per IP schnell, per Name langsam | DNS-Auflösung |
| Anmeldung langsam, Dienste danach normal | Authentifizierung, GPO, Profil oder DNS |

---

### 3. Auswirkungen und Priorität bestimmen

Vor der tieferen Analyse muss die betriebliche Auswirkung bewertet werden.

Zu klären ist:

- Ist ein geschäftskritischer Dienst betroffen?
- Können Benutzer noch eingeschränkt arbeiten?
- Sind Datenverlust oder Folgefehler möglich?
- Betrifft das Problem einen oder mehrere Standorte?
- Verschlechtert sich der Zustand weiter?
- Gibt es ein funktionierendes Ersatzsystem?
- Besteht eine vereinbarte Reaktions- oder Wiederherstellungszeit?

Beispiel:

| Kriterium | Bewertung |
|---|---|
| Auswirkung | alle Mitarbeiter können nur verzögert arbeiten |
| Dringlichkeit | hoch |
| Priorität | hoch |
| Ersatzlösung | zweiter Anwendungsserver vorhanden |
| Datenverlust | derzeit nicht erkennbar |
| Eskalation | Anwendungs- und Infrastrukturteam informieren |

Eine hohe Auslastung ist nicht automatisch ein kritischer Vorfall. Entscheidend ist, ob der vereinbarte Dienst beeinträchtigt wird.

---

**4. Ausgangszustand sichern**

Vor einem Neustart oder einer Konfigurationsänderung sollten flüchtige Informationen gesichert werden:

- genaue Uhrzeit mit Zeitzone,
- CPU-Auslastung,
- CPU-Auslastung pro Prozess,
- Arbeitsspeicherbelegung,
- Paging- oder Swap-Aktivität,
- Datenträgerauslastung,
- Datenträgerlatenz,
- Warteschlangen,
- Netzwerkfehler,
- aktive Verbindungen,
- Prozess- und Dienststatus,
- Eventlogs oder Journal,
- Anwendungslogs,
- Datenbankstatus,
- Hypervisorwerte,
- laufende Jobs,
- letzte Änderungen.

Ein Neustart kann das Symptom vorübergehend beseitigen, vernichtet aber möglicherweise den für die Ursachenanalyse benötigten Zustand.

Vor einer Maßnahme sollte daher mindestens dokumentiert werden:

```
Zeitpunkt:
betroffener Dienst:
gemessene Antwortzeit:
CPU:
RAM:
Datenträger:
Netzwerk:
auffälliger Prozess:
auffälliges Ereignis:
laufender Hintergrundjob:
letzte Änderung:
```

---

### 5. Sichere Schnellprüfung

Die erste Prüfung sollte möglichst nur lesend erfolgen.

1. Zeitpunkt und Umfang der Störung bestimmen.
2. CPU, RAM, Datenträger und Netzwerk gleichzeitig betrachten.
3. Prozesse mit ungewöhnlicher Ressourcennutzung ermitteln.
4. Backups, Virenscans, Snapshots und Wartungsjobs prüfen.
5. Betriebssystem-, Anwendungs- und Hypervisorlogs zeitlich vergleichen.
6. Antwortzeit vom Client und direkt auf dem Server messen.
7. Mit einem funktionierenden Vergleichssystem vergleichen.
8. Erst danach eine kontrollierte Maßnahme auswählen.

Windows:

```
Get-Date

Get-Process |
    Sort-Object CPU -Descending |
    Select-Object -First 10 Name, Id, CPU, WorkingSet64
```

Linux:

```
date
uptime
top
```

Diese Momentaufnahmen reichen noch nicht für eine vollständige Bewertung. Kurze Spitzen oder zeitweise auftretende Fehler können dabei unentdeckt bleiben.

---

### 6. CPU-Auslastung untersuchen

Eine hohe CPU-Auslastung kann durch einen einzelnen Prozess, mehrere konkurrierende Prozesse oder eine zu geringe bereitgestellte Rechenleistung verursacht werden.

Zu prüfen sind:

- gesamte CPU-Auslastung,
- Auslastung einzelner Kerne,
- Auslastung pro Prozess und Thread,
- Dauer der Auslastung,
- System- und Benutzerzeit,
- Interrupts,

- virtuelle CPU-Bereitstellung,
- CPU-Wartezeiten des Hypervisors,
- CPU-Limits oder Drosselung.

Windows:

```
Get-Counter '\Processor(_Total)\% Processor Time'
```

Prozesse nach gesamter CPU-Zeit sortieren:

```
Get-Process |
    Sort-Object CPU -Descending |
    Select-Object -First 15 Name, Id, CPU, Threads
```

Aktuelle Prozessorauslastung über CIM:

```
Get-CimInstance Win32_Processor |
    Select-Object Name, LoadPercentage
```

Linux:

```
top
```

Mit installiertem Paket `sysstat`:

```
pidstat 1
```

Auslastung pro logischer CPU:

```
mpstat -P ALL 1
```

| Befund | Mögliche Bedeutung |
|---|---|
| ein Prozess belegt dauerhaft einen Kern | Single-Thread-Engpass oder Endlosschleife |
| alle Kerne dauerhaft stark ausgelastet | CPU-Sättigung |
| hohe Systemzeit | Kernel, Treiber, Netzwerk oder I/O prüfen |
| viele Interrupts | Netzwerk- oder Hardwareproblem möglich |
| CPU im Gast niedrig, Anwendung trotzdem langsam | Storage, Netzwerk, Sperren oder Hypervisor prüfen |

| Befund | Mögliche Bedeutung |
|---|---|
| mehrere VMs gleichzeitig langsam | Hostüberlastung oder CPU-Scheduling prüfen |
| kurze regelmäßige Spitzen | geplanter Job, Scan oder Monitoring |
| Prozess-CPU steigt ständig | fehlerhafte Anwendung oder unpassende Last |

Eine Gesamtauslastung von beispielsweise 25 Prozent schließt einen CPU-Engpass nicht aus. Auf einem System mit vier logischen Prozessoren kann ein einzelner vollständig ausgelasteter Thread ungefähr 25 Prozent Gesamtauslastung verursachen.

---

### 7. Arbeitsspeicher untersuchen

Ein hoher belegter Arbeitsspeicher ist nicht automatisch ein Fehler. Betriebssysteme verwenden verfügbaren Speicher beispielsweise für Caches.

Entscheidend sind:

- verfügbarer Arbeitsspeicher,
- Auslagerungsaktivität,
- Speicherdruck,
- Arbeitsspeicher pro Prozess,
- Page Faults,
- Kernel- oder Pool-Speicher,
- zeitliche Entwicklung,
- Speicherlimits einer VM oder eines Containers.

Windows:

```
Get-CimInstance Win32_OperatingSystem |
    Select-Object TotalVisibleMemorySize,
                  FreePhysicalMemory
```

Prozesse nach Arbeitsspeicher sortieren:

```
Get-Process |
    Sort-Object WorkingSet64 -Descending |
    Select-Object -First 15 Name,
                            Id,
                            WorkingSet64,
                            PagedMemorySize64
```

Leistungsindikatoren abfragen:

```
Get-Counter `
  '\Memory\Available MBytes',
  '\Memory\Pages/sec'
```

Linux:

```
free -h
```

Ausführlichere Anzeige:

```
vmstat 1
```

Prozesse nach Speichernutzung sortieren:

```
ps aux --sort=-%mem | head
```

| Befund | Mögliche Bedeutung |
|---|---|
| wenig freier Speicher, aber keine Auslagerung | nicht zwingend problematisch |
| dauerhaft starke Paging- oder Swap-Aktivität | Arbeitsspeicherengpass |
| ein Prozess wächst kontinuierlich | mögliches Speicherleck |
| Anwendung erreicht festgelegtes Limit | Container-, VM- oder Prozesslimit prüfen |
| Server wird nach längerer Laufzeit langsamer | Speicherleck oder zunehmender Cache möglich |
| OOM-Ereignis unter Linux | Speicher war erschöpft |
| hoher Commit-Wert unter Windows | zugesicherter virtueller Speicher prüfen |

Nur den belegten Arbeitsspeicher zu betrachten, reicht nicht aus. Besonders Paging, Swap und die zeitliche Entwicklung sind entscheidend.

---

### 8. Datenträger und Storage untersuchen

Ein langsamer Datenträger kann den gesamten Server ausbremsen, obwohl CPU und Arbeitsspeicher unauffällig erscheinen.

Zu prüfen sind:

- Datenträgerlatenz,
- Lese- und Schreibraten,
- Warteschlangenlänge,

- IOPS,
- freier Speicherplatz,
- Dateisystemfehler,
- RAID-Zustand,
- SAN- oder NAS-Verbindung,
- Snapshot-Aktivität,
- Backup-Aktivität,
- Storage-Überbelegung,
- Zustand physischer Datenträger.

Windows-Leistungsindikatoren:

```
Get-Counter `
  '\PhysicalDisk(_Total)\Avg. Disk sec/Read',
  '\PhysicalDisk(_Total)\Avg. Disk sec/Write',
  '\PhysicalDisk(_Total)\Current Disk Queue Length',
  '\PhysicalDisk(_Total)\Disk Reads/sec',
  '\PhysicalDisk(_Total)\Disk Writes/sec'
```

Freien Speicherplatz prüfen:

```
Get-Volume |
  Select-Object DriveLetter,
                FileSystemLabel,
                FileSystem,
                Size,
                SizeRemaining
```

Linux:

```
df -h
```

Blockgeräte anzeigen:

```
lsblk
```

I/O-Auslastung mit installiertem `sysstat`:

```
iostat -xz 1
```

Prozesse mit I/O-Aktivität:

```
pidstat -d 1
```

| Befund | Mögliche Bedeutung |
|---|---|
| hohe Latenz bei geringer Datenrate | langsames oder blockiertes Storage |
| Warteschlange wächst dauerhaft | Datenträger kann Anfragen nicht schnell genug verarbeiten |
| freier Speicherplatz nahezu erschöpft | Dateisystem, Datenbank oder Anwendung beeinträchtigt |
| mehrere VMs gleichzeitig betroffen | gemeinsames Storage prüfen |
| Problem während eines Backups | konkurrierende I/O-Last |
| hohe I/O-Wartezeit unter Linux | Prozesse warten auf Storage |
| RAID degradiert | Leistung und Ausfallsicherheit reduziert |
| Snapshot wächst stark | zusätzliche Storage-Belastung möglich |

Allgemeine Grenzwerte dürfen nur als Orientierung verwendet werden. Ob eine Latenz problematisch ist, hängt vom Speichermedium, der Anwendung und den vereinbarten Leistungswerten ab.

---

### 9. Netzwerk untersuchen

Ein Netzwerkproblem kann aus Benutzersicht wie ein langsamer Server wirken.

Zu prüfen sind:

- Paketverlust,
- Latenz,
- Schwankungen der Latenz,
- DNS-Auflösung,
- Duplex- oder Geschwindigkeitsfehler,
- Interfacefehler,
- Überlastung,
- VPN,
- Firewall,
- Proxy,
- Load Balancer,
- Routing,
- MTU-Probleme,
- viele oder erschöpfte Verbindungen.

Windows:

```
Test-Connection SRV-APP01 -Count 10
```

TCP-Port prüfen:

```
Test-NetConnection SRV-APP01 -Port 443
```

Netzwerkadapter anzeigen:

```
Get-NetAdapter |
    Select-Object Name,
                  Status,
                  LinkSpeed,
                  MacAddress
```

Adapterstatistiken:

```
Get-NetAdapterStatistics
```

Aktive TCP-Verbindungen:

```
Get-NetTCPConnection |
    Group-Object State |
    Sort-Object Count -Descending
```

Linux:

```
ping -c 10 srv-app01
```

Route untersuchen:

```
traceroute srv-app01
```

Socketübersicht:

```
ss -s
```

Interfaceinformationen:

```
ip -s link
```

DNS-Auflösung prüfen:

```
dig srv-app01.example.local
```

| Befund | Mögliche Bedeutung |
|---|---|
| Paketverlust | Überlastung, fehlerhafter Link oder Routingproblem |
| schwankende Latenz | Überlastung oder instabiler Netzwerkpfad |
| Zugriff per IP schnell, per Name langsam | DNS prüfen |
| nur VPN-Benutzer betroffen | VPN-Gateway, Tunnel oder Internetverbindung |
| Fehlerzähler steigen | Kabel, Port, Treiber oder Netzwerkkarte |
| Anwendung lokal schnell, entfernt langsam | Netzwerkpfad, Proxy oder Firewall |
| nur große Übertragungen problematisch | MTU, Bandbreite oder Paketverlust prüfen |
| viele Verbindungen im Wartezustand | Anwendung, Netzwerk oder Porterschöpfung |

Ein erfolgreicher Ping beweist nur eingeschränkt die Funktionsfähigkeit. Er prüft nicht automatisch den eigentlichen Anwendungsport oder die Antwortzeit des Dienstes.

---

### 10. DNS untersuchen

Langsame oder fehlerhafte Namensauflösung kann viele Serverdienste beeinträchtigen.

Typische Auswirkungen:

- langsame Anmeldung,
- verzögerte Verbindungen,
- Authentifizierungsprobleme,
- lange Wartezeiten beim Zugriff auf andere Server,
- langsame Datenbankverbindungen,
- Verzögerungen bei Reverse-Lookups.

Windows:

```
Resolve-DnsName SRV-DB01
```

Konfigurierte DNS-Server:

```
Get-DnsClientServerAddress
```

DNS-Zwischenspeicher:

```
Get-DnsClientCache
```

Linux:

```
resolvectl status
```

Namensauflösung messen:

```
time getent hosts srv-db01.example.local
```

Mit `dig`:

```
dig srv-db01.example.local
```

Zu prüfen sind:

- korrekte DNS-Server,
- Antwortzeit,
- Forward- und Reverse-Auflösung,
- fehlerhafte oder veraltete Einträge,
- Suchdomänen,
- Erreichbarkeit der DNS-Server,
- unnötige Abfragen an externe DNS-Server.

---

### 11. Prozesse und Dienste untersuchen

Nicht jede hohe Ressourcennutzung ist die Ursache. Der auffällige Prozess kann selbst nur auf eine andere Komponente warten.

Windows-Dienste:

```
Get-Service |
    Where-Object Status -eq 'Running'
```

Automatisch startende, aber nicht laufende Dienste:

```
Get-CimInstance Win32_Service |
    Where-Object {
        $_.StartMode -eq 'Auto' -and
        $_.State -ne 'Running'
    } |
    Select-Object Name, DisplayName, State, StartMode
```

Linux-Dienste:

```
systemctl --failed
```

Status eines Dienstes:

```
systemctl status nginx
```

Zu prüfen sind:

- ungewöhnlich hohe CPU- oder Speichernutzung,
- häufige Dienstneustarts,
- Prozesse im Warte- oder Blockierungszustand,
- sehr viele Threads,
- sehr viele offene Dateien,
- abgestürzte Unterprozesse,
- lange Warteschlangen,
- Ressourcenlimits,
- fehlerhafte Abhängigkeiten.

Ein Dienstneustart sollte erst erfolgen, nachdem relevante Messwerte und Protokolle gesichert wurden.

---

**12. Anwendung untersuchen**

Wenn nur eine Anwendung langsam ist, muss deren interner Ablauf betrachtet werden.

Mögliche Ursachen:

- fehlerhafte Konfiguration,
- zu wenige Worker oder Threads,
- zu kleine Verbindungspools,
- blockierte Warteschlangen,
- langsame Datenbankabfragen,
- nicht erreichbare externe Dienste,

- Authentifizierungsprobleme,
- Cachefehler,
- unzureichende Ressourcenlimits,
- fehlerhaftes Update,
- wachsendes Log- oder temporäres Verzeichnis.

Zu prüfen sind:

- Antwortzeit einzelner Funktionen,
- Anwendungslogs,
- Fehler- und Zeitüberschreitungen,
- Anzahl aktiver Sitzungen,
- Thread- oder Worker-Auslastung,
- Warteschlangenlänge,
- Cachetreffer,
- Verbindungspools,
- interne Gesundheitsprüfungen,
- Abhängigkeiten.

Beispielhafte Abgrenzung:

```
Startseite schnell
Suchfunktion langsam
        ↓
Suchfunktion getrennt untersuchen
        ↓
Anwendungslog und Datenbankabfrage vergleichen
```

Nicht sofort den gesamten Server neu starten, wenn nur ein einzelner Anwendungsbereich betroffen ist.

---

**13. Datenbank untersuchen**

Datenbanken können trotz geringer CPU-Auslastung langsam reagieren.

Mögliche Ursachen:

- blockierende Transaktionen,
- Deadlocks,
- fehlende oder ungeeignete Indizes,
- langsame Abfragen,
- überlastetes Storage,
- zu wenig Arbeitsspeicher,
- zu viele Verbindungen,

- fehlerhafte Ausführungspläne,
- veraltete Statistiken,
- lange offene Transaktionen,
- Replikationsprobleme.

Zu prüfen sind:

- Abfragelaufzeiten,
- aktive Verbindungen,
- blockierte Sitzungen,
- Locks und Deadlocks,
- langsame Abfrageprotokolle,
- Cache- und Speichernutzung,
- Storage-Latenz,
- Replikationsverzögerung,
- Anzahl wartender Transaktionen.

Beispielhafte Ursache-Wirkungs-Kette:

```
lange Datenbankabfrage
        ↓
Anwendungs-Worker wartet
        ↓
Verbindungspool wird belegt
        ↓
weitere Benutzer müssen warten
        ↓
Anwendung erscheint vollständig langsam
```

Änderungen an Indizes, Abfragen oder Datenbankparametern sollten nur nach Sicherung und Abstimmung mit der verantwortlichen Stelle erfolgen.

---

**14. Hintergrundjobs prüfen**

Regelmäßig auftretende Leistungseinbrüche sprechen häufig für geplante Prozesse.

Typische Verursacher:

- Datensicherung,
- Virenscan,
- Patchinstallation,
- Datenbankwartung,
- Protokollrotation,

- Indexerstellung,
- Synchronisation,
- Replikation,
- Berichtserstellung,
- Snapshot,
- Datenimport oder Datenexport.

Windows – geplante Aufgaben:

```
Get-ScheduledTask |
    Where-Object State -eq 'Running'
```

Linux – systemd-Timer:

```
systemctl list-timers
```

Cron-Konfiguration prüfen:

```
crontab -l
```

Zu vergleichen sind:

- Startzeit des Jobs,
- Beginn der Verlangsamung,
- CPU-Auslastung,
- I/O-Auslastung,
- Netzwerkverkehr,
- Ende des Jobs,
- Rückkehr zum Normalzustand.

Eine zeitliche Übereinstimmung ist ein Hinweis, aber noch kein vollständiger Ursachennachweis.

---

**15. Protokolle zeitlich vergleichen**

Protokolle müssen mit dem dokumentierten Störungszeitpunkt abgeglichen werden.

Windows-Systemereignisse:

```
Get-WinEvent -FilterHashtable @{
    LogName   = 'System'
    StartTime = (Get-Date).AddHours(-2)
```

```
} |
    Select-Object TimeCreated,
                  Id,
                  LevelDisplayName,
                  ProviderName,
                  Message
```

Windows-Anwendungsereignisse:

```
Get-WinEvent -FilterHashtable @{
    LogName   = 'Application'
    StartTime = (Get-Date).AddHours(-2)
} |
    Select-Object TimeCreated,
                  Id,
                  LevelDisplayName,
                  ProviderName,
                  Message
```

Linux:

```
journalctl --since "2 hours ago"
```

Fehler mit hoher Priorität:

```
journalctl -p err --since "2 hours ago"
```

Kernelmeldungen:

```
journalctl -k --since "2 hours ago"
```

Zu suchen sind:

- Datenträgerfehler,
- Netzwerkfehler,
- Treiberprobleme,
- Dienstabbrüche,
- Zeitüberschreitungen,
- Speicherfehler,

- OOM-Ereignisse,
- Authentifizierungsfehler,
- Datenbankfehler,
- wiederholte Neustarts,
- Hardwarewarnungen.

Ein einzelner protokollierter Fehler beweist nicht automatisch die Ursache. Zeitliche Übereinstimmung, technische Wirkung und Reproduzierbarkeit müssen zusammenpassen.

---

**16. Virtuelle Maschinen untersuchen**

Bei einer virtuellen Maschine müssen Gast und Hypervisor getrennt betrachtet werden.

Im Gast zu prüfen:

- vCPU-Auslastung,
- Arbeitsspeicher,
- Paging oder Swapping,
- virtuelle Datenträger,
- Netzwerkschnittstelle,
- Betriebssystemlogs,
- installierte Integrationstreiber.

Auf dem Hypervisor zu prüfen:

- CPU-Auslastung des Hosts,
- CPU-Wartezeit,
- Speicherüberbelegung,
- Ballooning oder Swapping,
- Storage-Latenz,
- Netzwerküberlastung,
- Limits und Reservierungen,
- Snapshots,
- Zustand anderer VMs.

Typische Konstellation:

```
VM zeigt geringe CPU-Auslastung
        ↓
Anwendung ist trotzdem langsam
        ↓
Hypervisor prüfen
        ↓
VM erhält CPU-Zeit verzögert
```

Mehr virtuelle CPUs verbessern die Leistung nicht automatisch. Zu viele vCPUs können das Scheduling erschweren und die Wartezeit erhöhen.

---

**17. Container untersuchen**

Bei Containern müssen zusätzlich Limits und der Hostzustand berücksichtigt werden.

Zu prüfen sind:

- CPU-Limit,
- Arbeitsspeicherlimit,
- Neustartzähler,
- Containerlogs,
- Dateisystem,
- Storage-Treiber,
- Netzwerk,
- Zustand des Containerhosts,
- gemeinsam genutzte Ressourcen.

Docker:

```
docker stats
```

Laufende Container:

```
docker ps
```

Containerzustand:

```
docker inspect CONTAINERNAME
```

Protokollausgabe:

```
docker logs --since 2h CONTAINERNAME
```

Ressourcenlimits anzeigen:

```
docker inspect CONTAINERNAME \
    --format 'Memory={{.HostConfig.Memory}} NanoCPUs={{.HostConfig.NanoCpus}}'
```

Ein Container kann langsam sein, obwohl der Host noch freie Ressourcen besitzt, wenn für den Container selbst ein niedriges Limit festgelegt wurde.

---

**18. Externe Abhängigkeiten prüfen**

Eine Anwendung kann auf andere Systeme warten, obwohl der eigene Server unauffällig ist.

Mögliche Abhängigkeiten:

- Datenbankserver,
- DNS-Server,
- Active Directory,
- Dateiserver,
- API,
- Cloud-Dienst,
- Authentifizierungsdienst,
- Mailserver,
- Proxy,
- Lizenzserver,
- Netzwerkspeicher.

Zu prüfen sind:

- Erreichbarkeit,
- Namensauflösung,
- Portverbindung,
- Antwortzeit,
- Zertifikate,
- Authentifizierung,
- Zeitüberschreitungen,
- Rate Limits,
- Statusmeldungen des Dienstanbieters.

Beispiel:

```
Webserver reagiert lokal schnell
        ↓
Anmeldung dauert 20 Sekunden
        ↓
Anwendung wartet auf Verzeichnisdienst
        ↓
DNS- oder LDAP-Verbindung untersuchen
```

---

### 19. Hardwarezustand prüfen

Bei physischen Servern müssen auch Hardwareprobleme berücksichtigt werden.

Mögliche Ursachen:

- überhitzte CPU,
- thermische Drosselung,
- fehlerhafter Arbeitsspeicher,
- degradierter RAID-Verbund,
- defekter Datenträger,
- fehlerhafte Netzwerkkarte,
- instabiles Netzteil,
- Firmware- oder Treiberproblem.

Zu prüfen sind:

- Hardwaremanagement wie iLO, iDRAC oder vergleichbare Systeme,
- Temperaturwerte,
- Lüfterstatus,
- RAID-Controller,
- SMART-Werte,
- Hardwareereignisse,
- Firmwarestand,
- Treiberstand,
- Herstellerdiagnose.

Linux – vorhandene Hardwarewarnungen:

```
dmesg --level=err,warn
```

SMART-Werte dürfen nur mit passenden Werkzeugen und unter Berücksichtigung des verwendeten Controllers geprüft werden.

---

### 20. Letzte Änderungen untersuchen

Leistungsprobleme treten häufig nach einer Veränderung auf.

Mögliche Änderungen:

- Betriebssystemupdate,
- Anwendungsupdate,
- Treiberupdate,
- neue Sicherheitssoftware,
- geänderte Richtlinie,

- zusätzliche Benutzer,
- Datenwachstum,
- neue VM,
- geändertes Backupfenster,
- neue Datenbankabfrage,
- geänderte Ressourcenlimits,
- Netzwerkanpassung.

Zu dokumentieren sind:

| Änderung | Zeitpunkt | Verantwortlich | Mögliche Auswirkung |
|---|---|---|---|
| Anwendungsupdate | 08:30 Uhr | Anwendungsteam | neue Abfrage oder Fehler |
| Backupzeit geändert | 09:00 Uhr | Betriebsteam | höhere Storage-Last |
| neue VM gestartet | 09:10 Uhr | Virtualisierungsteam | Hostressourcen |
| GPO geändert | Vortag | Administration | Dienst- oder Sicherheitseinstellung |

Die zeitliche Nähe einer Änderung zum Fehler ist ein wichtiges Indiz, aber kein alleiniger Beweis.

---

### 21. Baseline und Vergleichswerte verwenden

Ohne Vergleichswerte lässt sich schwer beurteilen, ob ein Messwert ungewöhnlich ist.

Eine Baseline kann enthalten:

- typische CPU-Auslastung,
- typischer Arbeitsspeicherbedarf,
- normale Datenträgerlatenz,
- normale Netzwerklatenz,
- typische Benutzeranzahl,
- typische Antwortzeit,
- übliche Anzahl von Verbindungen,
- normale Auslastung zu bestimmten Tageszeiten.

Beispiel:

| Messwert | Normalzustand | Störungszeitpunkt |
|---|---:|---:|
| Antwortzeit | 1,5 s | 15 s |
| CPU | 35 % | 42 % |
| verfügbarer RAM | 8 GB | 7,5 GB |
| Storage-Latenz | 4 ms | 85 ms |
| aktive Benutzer | 120 | 125 |

In diesem Beispiel spricht der Vergleich stärker für ein Storage-Problem als für CPU-, RAM- oder Benutzerlast.

---

### 22. Korrelation und Ursache unterscheiden

Zwei gleichzeitig auftretende Ereignisse müssen nicht ursächlich zusammenhängen.

Beispiel:

```
CPU-Auslastung steigt
Anwendung wird langsam
```

Mögliche Interpretationen:

1. Die hohe CPU-Auslastung verursacht die Verzögerung.
2. Die langsame externe Abhängigkeit führt zu zusätzlichen Wiederholungen und erhöht dadurch die CPU-Auslastung.
3. Ein dritter Prozess verursacht gleichzeitig beide Effekte.
4. Die CPU-Auslastung ist normal und nicht relevant.

Ein Ursachennachweis wird stärker, wenn:

- der Zeitpunkt übereinstimmt,
- die technische Wirkung plausibel ist,
- das Verhalten reproduzierbar ist,
- eine gezielte Maßnahme den Messwert und das Symptom verändert,
- alternative Ursachen ausgeschlossen wurden.

---

### 23. Kontrollierte Maßnahmen durchführen

Nach der Analyse sollte genau eine passende Maßnahme durchgeführt werden.

Mögliche Maßnahmen:

- störenden Hintergrundjob pausieren,
- fehlerhaften Prozess kontrolliert neu starten,
- Anwendung auf einen zweiten Knoten umleiten,
- Ressourcenlimit korrigieren,
- fehlerhafte Änderung zurücknehmen,
- freien Speicherplatz schaffen,
- Storage- oder Netzwerkproblem eskalieren,
- Datenbankblockierung durch die zuständige Stelle auflösen.

Vor jeder Maßnahme:

- Ausgangszustand sichern,
- Auswirkung bestimmen,
- Freigabe prüfen,
- Rückweg festlegen,
- betroffene Benutzer informieren,
- Messkriterien bestimmen.

Danach:

- dieselben Messwerte erneut erfassen,
- Antwortzeit erneut messen,
- Protokolle prüfen,
- Nebenwirkungen ausschließen,
- Ergebnis dokumentieren.

---

**24. Warum ein Neustart nicht die erste Maßnahme sein sollte**

Ein Neustart kann:

- blockierte Prozesse beenden,
- Arbeitsspeicher freigeben,
- Warteschlangen zurücksetzen,
- Verbindungen neu aufbauen,
- ein Symptom vorübergehend beseitigen.

Er kann aber auch:

- wichtige Fehlerzustände vernichten,
- Protokollzusammenhänge erschweren,
- eine eigentliche Ursache verbergen,
- ungeplante Ausfallzeit verursachen,
- Daten oder Transaktionen gefährden,
- das Problem später erneut auftreten lassen.

Ein Neustart ist daher keine vollständige Ursachenanalyse.

Falls ein Neustart erforderlich ist, sollten vorher mindestens folgende Informationen gesichert werden:

- Prozessliste,
- Ressourcenauslastung,
- aktive Verbindungen,
- relevante Protokolle,
- laufende Aufgaben,
- Zeitpunkt und Symptom,
- Wiederanlaufplan,

- Rückfalloption.

---

**25. Nachkontrolle**

Nach einer Maßnahme muss nicht nur die Ressourcenauslastung, sondern auch die eigentliche Funktion geprüft werden.

Zu kontrollieren sind:

- Antwortzeit des betroffenen Dienstes,
- CPU-Auslastung,
- Arbeitsspeicher,
- Paging oder Swap,
- Datenträgerlatenz,
- Netzwerkfehler,
- Dienststatus,
- Anwendungslogs,
- Datenbankstatus,
- Benutzerfunktion,
- Zustand abhängiger Systeme.

Beispiel:

```
Vorher:
Antwortzeit 15 Sekunden
Storage-Latenz 85 ms

Maßnahme:
konkurrierenden Backupjob beendet

Nachher:
Antwortzeit 1,7 Sekunden
Storage-Latenz 5 ms
```

Erst der Vergleich vor und nach der Maßnahme zeigt, ob die vermutete Ursache tatsächlich zum Problem beigetragen hat.

---

**26. Dokumentation**

Die Abschlussdokumentation sollte enthalten:

- betroffener Server,
- betroffener Dienst,

- Beginn und Ende,
- Sollzustand,
- Istzustand,
- betroffene Benutzer,
- gemessene Werte,
- relevante Protokolle,
- identifizierte Ursache,
- durchgeführte Maßnahme,
- Rückweg,
- Ergebnis der Nachkontrolle,
- Nebenwirkungen,
- vorbeugende Maßnahme,
- verantwortliche Stelle.

Beispiel:

```
Symptom:
Webanwendung benötigte statt 1–2 Sekunden etwa 15 Sekunden.

Ursache:
Der gleichzeitig laufende Backupjob verursachte eine hohe
Latenz auf dem gemeinsam genutzten Storage.

Maßnahme:
Backupjob kontrolliert beendet und Zeitplan angepasst.

Ergebnis:
Antwortzeit wieder unter 2 Sekunden.
Storage-Latenz wieder im üblichen Bereich.

Vorbeugung:
Monitoring für Storage-Latenz eingerichtet und Backupfenster geändert.
```

---

### 27. Entscheidungsbaum

```
Server oder Dienst langsam
        ↓
Ist der gesamte Server betroffen?
        ├── Ja
        │   ↓
```

```
    │  CPU, RAM, Storage, Netzwerk und Hypervisor prüfen
    │
    └── Nein
        ↓
    Nur eine Anwendung betroffen?
        ├── Ja
        │   ↓
        │   Anwendung, Datenbank und Abhängigkeiten prüfen
        │
        └── Nein
            ↓
        Gemeinsame Komponente der betroffenen Dienste suchen
```

Erweiterte Eingrenzung:

```
Problem nur aus der Ferne?
    ├── Ja → Netzwerk, DNS, VPN, Firewall oder Proxy
    └── Nein
        ↓
Problem nur zu festen Zeiten?
    ├── Ja → Backup, Scan, Wartung oder geplanter Job
    └── Nein
        ↓
Mehrere VMs auf einem Host betroffen?
    ├── Ja → Hypervisor und gemeinsames Storage
    └── Nein → Gastbetriebssystem und Anwendung
```

---

### 28. Häufige Fehlerbilder

| Fehlerbild | Wahrscheinlicher Prüfbereich |
|---|---|
| Server reagiert nach Neustart wieder schnell | Speicherleck, blockierter Prozess oder wartende Ressource |
| Problem kehrt regelmäßig zurück | Ursache wurde nicht behoben oder geplanter Job |
| CPU dauerhaft hoch | Prozess, Thread, Malware, Scan oder zu geringe Leistung |
| CPU niedrig, Server trotzdem langsam | Storage, Netzwerk, Locks oder externe Abhängigkeit |
| RAM nahezu vollständig belegt | Paging und Speicherdruck prüfen |
| Swap-Nutzung steigt dauerhaft | Arbeitsspeicherengpass |
| Datenträger dauerhaft ausgelastet | Backup, Datenbank, Scan oder langsames Storage |

| Fehlerbild | Wahrscheinlicher Prüfbereich |
|---|---|
| nur Datenbankfunktionen langsam | Abfragen, Locks, Indizes oder Storage |
| nur Anmeldung langsam | DNS, Verzeichnisdienst, GPO oder Benutzerprofil |
| nur Webzugriff langsam | Webserver, TLS, Proxy, Anwendung oder Datenbank |
| nur bestimmte Uhrzeit betroffen | geplanter Job oder Lastspitze |
| mehrere VMs betroffen | Host, Netzwerk oder gemeinsames Storage |
| nur eine VM betroffen | Gastbetriebssystem, Limits oder VM-Konfiguration |
| Zugriff per IP schnell | DNS-Auflösung prüfen |
| lokal schnell, entfernt langsam | Netzwerkpfad prüfen |
| Anwendung wartet auf Timeout | externe Abhängigkeit untersuchen |
| Leistung nimmt über Tage ab | Speicherleck, Datenwachstum oder Warteschlange |

---

## 29. Typische Prüfungsfragen

### Warum ist die Aussage „der Server ist langsam“ für die Fehleranalyse nicht ausreichend?

**Antwort anzeigen**

Die Aussage beschreibt nur ein allgemeines Symptom. Es fehlen unter anderem der betroffene Dienst, die gemessene Antwortzeit, der Normalzustand, der Zeitpunkt, der Umfang und mögliche Änderungen.

### Warum beweist eine hohe CPU-Auslastung noch nicht die Fehlerursache?

**Antwort anzeigen**

Die Auslastung kann für die vorhandene Arbeitslast normal sein oder erst als Folge eines anderen Problems entstehen. Für einen Ursachennachweis müssen zeitlicher Zusammenhang, technische Wirkung und das Ergebnis einer kontrollierten Maßnahme geprüft werden.

### Warum kann ein Server trotz niedriger CPU-Auslastung langsam sein?

**Antwort anzeigen**

Prozesse können auf Storage, Netzwerk, Datenbanksperren, externe Dienste oder andere Ressourcen warten. Während dieser Wartezeit wird nur wenig CPU-Leistung benötigt.

**Warum ist vollständig belegter Arbeitsspeicher nicht automatisch ein Fehler?**

**Antwort anzeigen**

Betriebssysteme verwenden freien Arbeitsspeicher unter anderem für Caches. Entscheidend sind Speicherdruck, Paging beziehungsweise Swapping und die zeitliche Entwicklung.

**Warum sollten vor einem Neustart Messwerte gesichert werden?**

**Antwort anzeigen**

Ein Neustart beendet Prozesse, leert Warteschlangen und verändert den Systemzustand. Dadurch können wichtige Hinweise auf die eigentliche Ursache verloren gehen.

**Was bedeutet eine hohe Datenträgerlatenz?**

**Antwort anzeigen**

Lese- oder Schreibanforderungen benötigen ungewöhnlich lange. Anwendungen können dadurch warten, obwohl CPU und Arbeitsspeicher unauffällig sind.

**Warum reicht ein erfolgreicher Ping nicht als Nachweis für einen funktionierenden Dienst?**

**Antwort anzeigen**

Ping prüft ICMP-Erreichbarkeit. Der eigentliche Anwendungsport, die Authentifizierung, die Verarbeitung im Dienst und dessen Backend werden dadurch nicht geprüft.

**Warum müssen bei einer VM auch Hypervisorwerte geprüft werden?**

**Antwort anzeigen**

Der Gast kann nur die ihm bereitgestellten Ressourcen sehen. Wartezeiten, Überbelegung oder Storage-Probleme auf dem Host sind im Gast möglicherweise nicht eindeutig erkennbar.

**Warum sollte nach einer Maßnahme erneut gemessen werden?**

**Antwort anzeigen**

Nur der Vergleich von Sollwert, Ausgangswert und neuem Messwert zeigt, ob die Maßnahme das eigentliche Problem behoben hat.

---

### 30. Prüfungsfallen

- „Server langsam“ nicht genauer eingrenzen.
- einen einzelnen hohen Messwert sofort als Ursache bewerten.
- nur CPU und Arbeitsspeicher prüfen.
- Datenträgerlatenz nicht berücksichtigen.
- belegten Arbeitsspeicher automatisch mit Speichermangel gleichsetzen.
- Ping als vollständigen Diensttest ansehen.
- DNS nicht prüfen.
- Anwendung und Server nicht voneinander trennen.
- externe Abhängigkeiten übersehen.
- nur den Gast und nicht den Hypervisor untersuchen.
- Containerlimits nicht berücksichtigen.
- Hintergrundjobs nicht mit dem Störungszeitpunkt vergleichen.
- Protokolle ohne genaue Uhrzeit untersuchen.
- mehrere Maßnahmen gleichzeitig durchführen.
- sofort neu starten.
- Zustand vor der Maßnahme nicht sichern.
- nach der Maßnahme nicht erneut messen.
- technische Messwerte prüfen, aber die Benutzerfunktion nicht testen.
- zeitliche Korrelation automatisch als Ursache bewerten.
- das funktionierende Vergleichssystem nicht nutzen.
- keine vorbeugende Maßnahme dokumentieren.

---

### 31. Checkliste „Server ist langsam“

- [ ] betroffener Server wurde eindeutig bestimmt.
- [ ] betroffener Dienst wurde bestimmt.
- [ ] Sollzustand wurde dokumentiert.
- [ ] Istzustand wurde gemessen.
- [ ] Beginn und Häufigkeit wurden dokumentiert.
- [ ] betroffene Benutzer und Standorte wurden bestimmt.
- [ ] lokaler und entfernter Zugriff wurden verglichen.
- [ ] funktionierendes Vergleichssystem wurde geprüft.
- [ ] letzte Änderungen wurden ermittelt.

- [ ] Ausgangszustand wurde gesichert.
- [ ] CPU-Gesamtauslastung wurde geprüft.
- [ ] einzelne CPU-Kerne wurden berücksichtigt.
- [ ] CPU-Auslastung pro Prozess wurde geprüft.
- [ ] Arbeitsspeicher wurde geprüft.
- [ ] Paging oder Swapping wurde geprüft.
- [ ] Prozesse mit hoher Speichernutzung wurden ermittelt.
- [ ] freier Speicherplatz wurde geprüft.
- [ ] Datenträgerlatenz wurde geprüft.
- [ ] Datenträgerwarteschlange wurde geprüft.
- [ ] RAID- oder Storage-Zustand wurde berücksichtigt.
- [ ] Netzwerkverbindung wurde geprüft.
- [ ] Paketverlust und Latenz wurden geprüft.
- [ ] Netzwerkfehlerzähler wurden geprüft.
- [ ] DNS-Auflösung wurde geprüft.
- [ ] Anwendungsport wurde geprüft.
- [ ] aktive Verbindungen wurden geprüft.
- [ ] Dienste und Prozesse wurden geprüft.
- [ ] Anwendungslogs wurden geprüft.
- [ ] Betriebssystemlogs wurden geprüft.
- [ ] Datenbankzustand wurde geprüft.
- [ ] Datenbanksperren wurden berücksichtigt.
- [ ] externe Abhängigkeiten wurden geprüft.
- [ ] Backups und Wartungsjobs wurden geprüft.
- [ ] Virenscans wurden berücksichtigt.
- [ ] Hypervisorwerte wurden geprüft.
- [ ] VM- oder Containerlimits wurden geprüft.
- [ ] Hardwarezustand wurde berücksichtigt.
- [ ] Messwerte wurden zeitlich korreliert.
- [ ] genau eine kontrollierte Maßnahme wurde durchgeführt.
- [ ] Rückweg wurde festgelegt.
- [ ] dieselben Messwerte wurden anschließend erneut erfasst.
- [ ] fachliche Funktion wurde praktisch getestet.
- [ ] Nebenwirkungen wurden ausgeschlossen.

- [ ] Ursache, Maßnahme und Ergebnis wurden dokumentiert.
- [ ] vorbeugende Maßnahme wurde festgelegt.

---

## 32. Schnellreferenz

| Beobachtung | Nächster Prüfbereich |
|---|---|
| gesamte Maschine langsam | CPU, RAM, Storage, Hypervisor |
| nur eine Anwendung langsam | Anwendung, Datenbank, Abhängigkeiten |
| CPU dauerhaft hoch | Prozess- und Threadauslastung |
| nur ein Kern ausgelastet | Single-Thread-Engpass |
| CPU niedrig, Antwortzeit hoch | Storage, Netzwerk, Locks |
| starke Paging-Aktivität | Arbeitsspeicher |
| hohe I/O-Wartezeit | Datenträger oder Storage |
| wenig freier Speicherplatz | Dateisystem und Datenwachstum |
| lokale Nutzung schnell | Netzwerk, DNS, Proxy oder Firewall |
| Zugriff per IP schnell | DNS |
| Paketverlust vorhanden | Netzwerkpfad |
| nur bestimmte Uhrzeit betroffen | geplante Aufgaben |
| Problem während Backup | Storage- und Netzwerkbelastung |
| mehrere VMs betroffen | Hypervisor und gemeinsames Storage |
| nur eine VM betroffen | Gast, Limits und virtuelle Hardware |
| Container wird gedrosselt | CPU- und Speicherlimit |
| bestimmte Datenbankabfrage langsam | Locks, Indizes und Ausführungsplan |
| Anmeldung langsam | DNS, AD, GPO oder Profil |
| Anwendung wartet auf Timeout | externe Abhängigkeit |
| nach Neustart vorübergehend schnell | Speicherleck, Warteschlange oder blockierter Prozess |
| Hardwarewarnung vorhanden | Herstellerdiagnose und Austauschplanung |

---

## Merksatz

> Ein langsamer Server wird nicht durch Vermutungen, sondern durch Eingrenzung und Messwerte untersucht. Entscheidend ist, ob CPU, Arbeitsspeicher, Storage, Netzwerk, Anwendung, Datenbank, Hypervisor oder

> eine externe Abhängigkeit den tatsächlichen Engpass verursacht. Vor einer Änderung wird der Ausgangszustand gesichert, anschließend wird genau eine kontrollierte Maßnahme durchgeführt und ihre Wirkung erneut gemessen.

---

**Quellen und weiterführende Dokumentation**


- Microsoft Learn – Windows Performance Monitor
- Microsoft Learn – Get-Counter
- Microsoft Learn – Get-Process
- Microsoft Learn – Get-WinEvent
- Microsoft Learn – Test-NetConnection
- Microsoft Learn – Resolve-DnsName
- Microsoft Learn – Get-NetAdapterStatistics
- Red Hat – Viewing system processes
- Red Hat – Monitoring performance with Performance Co-Pilot
- Docker Docs – Runtime metrics
- Docker Docs – Resource constraints


---