7.6 Fehler tritt nur manchmal auf

Sporadische Fehler treten unregelmäßig auf und sind bei einer direkten Prüfung häufig nicht mehr vorhanden. Beispiele sind kurze Verbindungsabbrüche, gelegentliche Anmeldefehler, zeitweise langsame Anwendungen oder Dienste, die sich scheinbar ohne Eingriff wieder stabilisieren.

Die besondere Schwierigkeit besteht darin, dass der Systemzustand während der Störung meistens nicht direkt beobachtet wird. Deshalb müssen Diagnoseinformationen bereits vor dem nächsten Auftreten gesammelt werden.

```
Sporadischer Fehler
        ↓
genauen Zeitpunkt erfassen
        ↓
Messwerte und Protokolle sichern
        ↓
wiederkehrendes Muster erkennen
        ↓
gemeinsame Ursache eingrenzen
```

---

1. Typische Symptome

- Netzwerkverbindung fällt für wenige Sekunden aus.
- Anwendung reagiert gelegentlich nicht.
- Benutzer kann sich manchmal nicht anmelden.
- Server ist nur zu bestimmten Zeiten langsam.
- VPN-Verbindung wird unregelmäßig getrennt.
- WLAN-Verbindung verliert kurzzeitig den Access Point.
- DNS-Auflösung schlägt vereinzelt fehl.
- Dateiübertragung bricht gelegentlich ab.
- Dienst startet nach einem Fehler automatisch neu.
- Druckauftrag funktioniert erst beim zweiten Versuch.
- Datenbankverbindung läuft in einen Timeout.
- Fehler verschwindet nach erneutem Laden der Anwendung.
- Problem tritt nur an einzelnen Tagen oder Uhrzeiten auf.
- Überwachung zeigt kurze Spitzen, obwohl der Durchschnitt normal ist.

---

2. Mögliche Auswirkungen

- Arbeitsunterbrechungen,
- verlorene oder doppelt ausgeführte Vorgänge,

- beschädigte Dateiübertragungen,
- getrennte Sitzungen,
- fehlgeschlagene Sicherungen,
- unvollständige Synchronisation,
- verzögerte Geschäftsprozesse,
- schwer nachvollziehbare Benutzerbeschwerden,
- steigender Supportaufwand,
- falsche Ursachenzuordnung,
- unerkannt zunehmende Hardwarefehler,
- Sicherheits- oder Verfügbarkeitsrisiken.

Ein sporadischer Fehler ist nicht automatisch harmlos. Kurze Unterbrechungen können besonders bei Datenbanken, Transaktionen, Authentifizierungen, Telefonie, VPN-Verbindungen und Schreibvorgängen erhebliche Auswirkungen haben.

---

3. Sicherheits- und Betriebshinweis

Dauerüberwachung, Protokollierung und Paketmitschnitte dürfen nur autorisiert durchgeführt werden.

Dabei können verarbeitet werden:

- IP-Adressen,
- Benutzernamen,
- Gerätenamen,
- DNS-Abfragen,
- Kommunikationsbeziehungen,
- Sitzungsinformationen,
- Anmeldedaten,
- Anwendungsinhalte,
- personenbezogene Daten,
- vertrauliche Unternehmensdaten.

Vor einer erweiterten Protokollierung sind deshalb zu klären:

- betriebliche Freigabe,
- erforderlicher Umfang,
- Speicherort,
- Zugriffsschutz,
- Aufbewahrungsdauer,
- Datenschutz,
- anschließende Löschung,
- mögliche Systembelastung.

Protokollierung darf nicht unbegrenzt und ohne festgelegten Zweck aktiviert bleiben.

---

4. Fehler nicht vorschnell als „nicht reproduzierbar“ schließen

Wenn ein Fehler bei der Prüfung nicht mehr vorhanden ist, bedeutet dies nur:

```
Der Fehler ist im aktuellen Moment nicht sichtbar.
```

Es bedeutet nicht:

```
Der Fehler hat nicht stattgefunden.
```

Ein sporadischer Fehler kann zwischen zwei Prüfungen auftreten und wieder verschwinden.
Deshalb müssen folgende Informationen gesammelt werden:

- exakter Zeitpunkt,
- Dauer,
- betroffener Benutzer,
- betroffenes Gerät,
- verwendete Anwendung,
- ausgeführte Aktion,
- Zielsystem,
- angezeigte Fehlermeldung,
- Netzwerkverbindung,
- Standort,
- Häufigkeit,
- mögliche Auslöser,
- Zustand vor und nach dem Fehler.

---

5. Exakten Zeitpunkt erfassen

Die wichtigste Angabe ist ein möglichst genauer Zeitstempel.

Ungeeignet:

```
Der Fehler war heute Nachmittag.
```

Besser:

```
02.08.2026, ungefähr 14:17 Uhr
```

Optimal:

```
02.08.2026, 14:17:32 bis 14:17:47 Uhr
```

Zusätzlich dokumentieren:

```
Benutzer:       Max Mustermann
Client:         NB-023
IP-Adresse:     192.0.2.45
Anwendung:      ERP-Client
Zielserver:     ERP-SRV-01
Aktion:         Auftrag speichern
Fehlermeldung:  Verbindung zum Server unterbrochen
Beginn:         14:17:32 Uhr
Ende:           14:17:47 Uhr
```

Je genauer der Zeitpunkt bekannt ist, desto gezielter können Ereignisprotokolle, Monitoringdaten und Paketmitschnitte ausgewertet werden.

---

6. Zeitsynchronisation prüfen

Protokolle verschiedener Systeme lassen sich nur zuverlässig vergleichen, wenn deren Uhren korrekt synchronisiert sind.

Zu prüfen sind:

- aktuelle Uhrzeit,
- Zeitzone,
- NTP-Quelle,
- Synchronisationsstatus,
- Zeitabweichung,
- Sommer- oder Winterzeit,
- UTC gegenüber lokaler Zeit,
- Zeitstempel von Netzwerkgeräten,
- Zeitstempel von Containern und virtuellen Maschinen.

Unter Windows:

```
w32tm /query /status
```

Konfigurierte Zeitquelle:

```
w32tm /query /source
```

Unter Linux:

```
timedatectl status
```

Bei chrony, sofern eingesetzt:

```
chronyc tracking
```

```
chronyc sources
```

Beispiel

```
Client:     14:17:32 Uhr
Server:     14:15:21 Uhr
Firewall:   12:17:30 UTC
```

Ohne Berücksichtigung von Zeitabweichung und Zeitzone könnten zusammengehörige Ereignisse fälschlich als getrennte Vorfälle bewertet werden.

Die Systemzeit darf nicht unkontrolliert auf Produktivsystemen geändert werden. Zeitänderungen können Protokolle, Zertifikate, Authentifizierung, Datenbanken und verteilte Systeme beeinflussen.

---

7. Häufigkeit und Muster bestimmen

Zu klären ist, ob der Fehler:

- mehrmals täglich,
- einmal täglich,
- an bestimmten Wochentagen,
- nach einer bestimmten Laufzeit,
- nach einer Anmeldung,
- nach dem Standby,
- nach einem Neustart,
- während hoher Last,
- nach einem Netzwerkwechsel,
- bei Ablauf einer Sitzung,
- bei einem Backup,
- während eines Updates,

- bei Temperaturänderungen,
- nur an einem Standort,
- nur im WLAN,
- nur über VPN

auftritt.

Beispielhafte Fehlerliste

Datum	Uhrzeit	Dauer	Benutzer	System	Beobachtung
30.07.2026	14:17	15 s	Max Mustermann	NB-023	ERP-Verbindung getrennt
31.07.2026	14:16	18 s	Max Mustermann	NB-023	ERP-Verbindung getrennt
01.08.2026	14:18	12 s	mehrere	verschiedene	Dateiserver kurz nicht erreichbar
02.08.2026	14:17	15 s	mehrere	verschiedene	ERP und Dateiserver betroffen

Das wiederkehrende Zeitfenster deutet auf einen geplanten oder regelmäßig ausgelösten Prozess hin.

---

8. Umfang des Fehlers bestimmen

Folgende Fragen helfen bei der Eingrenzung:

1. Ist nur ein Benutzer betroffen?
2. Sind mehrere Benutzer betroffen?
3. Sind alle Benutzer eines Standorts betroffen?
4. Ist nur eine Anwendung betroffen?
5. Sind mehrere Dienste gleichzeitig betroffen?
6. Ist nur ein Client betroffen?
7. Sind alle Geräte an einem Switch betroffen?
8. Sind ausschließlich WLAN-Clients betroffen?
9. Sind nur VPN-Verbindungen betroffen?
10. Ist nur eine Übertragungsrichtung betroffen?
11. Ist nur ein bestimmter Server betroffen?
12. Sind interne und externe Ziele betroffen?

Bewertung

Beobachtung	Möglicher Prüfbereich
nur ein Benutzer	Benutzerprofil, Sitzung oder Berechtigung

Beobachtung	Möglicher Prüfbereich
nur ein Client	Client, Treiber, Kabel, WLAN oder lokale Software
mehrere Clients am gleichen Switch	Switch, Uplink oder Stromversorgung
alle WLAN-Clients eines Access Points	Access Point, Funkkanal oder AP-Uplink
mehrere Dienste auf einem Server	Server, Betriebssystem oder gemeinsame Ressource
mehrere Systeme gleichzeitig	Netzwerk, DNS, Authentifizierung, Storage oder Infrastruktur
nur VPN-Benutzer	Internetzugang, Tunnel, Gateway, MTU oder Sitzungszeit
alle Standorte gleichzeitig	zentrales System, Provider oder übergreifender Dienst

---

9. Fehler automatisiert erfassen

Ein sporadischer Fehler sollte möglichst durch wiederholte Messungen sichtbar gemacht werden.

Geeignete Messwerte sind beispielsweise:

- Erreichbarkeit,
- Antwortzeit,
- Paketverlust,
- DNS-Auflösung,
- TCP-Port-Erreichbarkeit,
- HTTP-Status,
- Anwendungsantwortzeit,
- CPU-Auslastung,
- Arbeitsspeicher,
- Storage-Latenz,
- Interfacezustand,
- Fehlerzähler,
- Dienststatus,
- Anzahl der Sitzungen,
- Temperatur,
- Spannungs- oder Stromereignisse.

Die Messung muss einen Zeitstempel enthalten.

Grundprinzip

```
Zeitstempel + Messwert + Ziel + Ergebnis
```

Beispiel:

```
2026-08-02 14:17:30 | ERP-SRV-01 | Ping 1 ms | TCP 443 erreichbar
2026-08-02 14:17:35 | ERP-SRV-01 | Zeitüberschreitung
2026-08-02 14:17:40 | ERP-SRV-01 | Zeitüberschreitung
2026-08-02 14:17:45 | ERP-SRV-01 | Ping 2 ms | TCP 443 erreichbar
```

---

10. Kontinuierliche Erreichbarkeit unter Windows prüfen

Ein dauerhafter Ping kann kurzfristige Unterbrechungen sichtbar machen:

```
ping <ziel-ip> -t
```

Der Test wird mit `Strg + C` beendet.

Aussagekräftiger ist eine protokollierte PowerShell-Messung mit Zeitstempel:

```
while ($true) {
    $timestamp = Get-Date -Format "yyyy-MM-dd HH:mm:ss"
    $result = Test-Connection -ComputerName "<ziel-ip>" -Count 1 -Quiet
    "$timestamp | <ziel-ip> | Erreichbar: $result" |
        Out-File -FilePath ".\erreichbarkeit.log" -Append
    Start-Sleep -Seconds 5
}
```

TCP-Port wiederholt prüfen:

```
while ($true) {
    $timestamp = Get-Date -Format "yyyy-MM-dd HH:mm:ss"
    $result = Test-NetConnection -ComputerName "<server>" -Port 443
    "$timestamp | TCP 443: $($result.TcpTestSucceeded)" |
        Out-File -FilePath ".\tcp-443.log" -Append
    Start-Sleep -Seconds 5
}
```

Wichtig

- Testintervall an die erwartete Fehlerdauer anpassen.
- Speicherbedarf begrenzen.
- Logdateien regelmäßig kontrollieren.
- Messung nach Abschluss wieder beenden.

- Ein erfolgreicher Ping beweist nicht, dass die Anwendung funktioniert.
- Ein fehlgeschlagener Ping beweist nicht automatisch einen vollständigen Ausfall, da ICMP blockiert oder begrenzt werden kann.

---

11. Kontinuierliche Erreichbarkeit unter Linux oder macOS prüfen

Linux:

```
ping <ziel-ip>
```

macOS:

```
ping <ziel-ip>
```

Mit Zeitstempeln unter Linux, sofern `ping` diese Option unterstützt:

```
ping -D <ziel-ip>
```

Alternativ kann eine kontrollierte Schleife verwendet werden:

```
while true; do
    timestamp=$(date '+%Y-%m-%d %H:%M:%S')
    if ping -c 1 -W 2 <ziel-ip> >/dev/null 2>&1; then
        echo "$timestamp | <ziel-ip> | erreichbar"
    else
        echo "$timestamp | <ziel-ip> | nicht erreichbar"
    fi
    sleep 5
done
```

Da sich Optionen von `ping` zwischen Linux und macOS unterscheiden können, müssen die Parameter des verwendeten Systems geprüft werden:

```
man ping
```

---

12. Mehrere Messpunkte gleichzeitig verwenden

Eine einzelne Zielmessung zeigt nicht, an welcher Stelle der Fehler auftritt. Deshalb sollten mehrere Punkte parallel überwacht werden.

```
Client
  |
Standardgateway
  |
interner Server
  |
externer Dienst
```

Beispiel:

```
14:17:35 | Gateway erreichbar
14:17:35 | interner Server nicht erreichbar
14:17:35 | externer Dienst nicht erreichbar
```

Mögliche Eingrenzung:

```
Clientzugang funktioniert.
Die Störung beginnt hinter dem Gateway oder auf einem gemeinsamen Folgepfad.
```

Anderes Beispiel:

```
14:17:35 | Gateway nicht erreichbar
14:17:35 | interner Server nicht erreichbar
14:17:35 | externer Dienst nicht erreichbar
```

Mögliche Eingrenzung:

```
Client, lokaler Zugang, WLAN, Switchport oder lokales Netzwerk prüfen.
```

Vergleichstabelle

Gateway	Interner Server	Externes Ziel	Eingrenzung
nicht erreichbar	nicht erreichbar	nicht erreichbar	Client oder lokaler Zugang
erreichbar	nicht erreichbar	erreichbar	interner Pfad oder Server

Gateway	Interner Server	Externes Ziel	Eingrenzung
erreichbar	erreichbar	nicht erreichbar	Internet-, Provider- oder externer Pfad
überall erreichbar	Anwendung fehlerhaft	Anwendung, Dienst oder Sitzung	
mehrere Ziele gleichzeitig langsam	gemeinsamer Netzwerkpfad oder Infrastruktur		

---

13. Windows-Ereignisprotokolle zeitlich filtern

Ereignisse eines bestimmten Zeitraums anzeigen:

```
Get-WinEvent -FilterHashtable @{
    LogName   = "System"
    StartTime = [datetime]"2026-08-02 14:15:00"
    EndTime   = [datetime]"2026-08-02 14:20:00"
} |
    Select-Object TimeCreated, ProviderName, Id, LevelDisplayName, Message
```

Anwendungsprotokoll:

```
Get-WinEvent -FilterHashtable @{
    LogName   = "Application"
    StartTime = [datetime]"2026-08-02 14:15:00"
    EndTime   = [datetime]"2026-08-02 14:20:00"
} |
    Select-Object TimeCreated, ProviderName, Id, LevelDisplayName, Message
```

Fehler und Warnungen aus dem Systemprotokoll:

```
Get-WinEvent -FilterHashtable @{
    LogName   = "System"
    Level     = 2, 3
    StartTime = (Get-Date).AddHours(-4)
} |
    Select-Object TimeCreated, ProviderName, Id, LevelDisplayName, Message
```

Zu prüfen sind unter anderem:

- Treiberfehler,
- Netzwerkunterbrechungen,
- Dienstabbrüche,
- unerwartete Neustarts,
- Datenträgerfehler,
- DNS-Fehler,
- Authentifizierungsfehler,
- Zeitabweichungen,
- Ressourcenknappheit,
- Anwendungsabstürze,
- automatische Wiederherstellungen.

Ein zeitgleiches Ereignis ist zunächst eine Korrelation. Erst weitere Messungen zeigen, ob es tatsächlich die Ursache des Fehlers ist.

---

14. Linux-Protokolle zeitlich filtern

Systemprotokoll eines genauen Zeitfensters:

```
journalctl \
    --since "2026-08-02 14:15:00" \
    --until "2026-08-02 14:20:00"
```

Nur Meldungen eines bestimmten Dienstes:

```
journalctl \
    -u <dienstname> \
    --since "2026-08-02 14:15:00" \
    --until "2026-08-02 14:20:00"
```

Kernelmeldungen:

```
journalctl \
    -k \
    --since "2026-08-02 14:15:00" \
    --until "2026-08-02 14:20:00"
```

Nur Warnungen und schwerwiegendere Meldungen:

```
journalctl -p warning \
    --since "2026-08-02 14:15:00" \
    --until "2026-08-02 14:20:00"
```

Live verfolgen:

```
journalctl -f
```

Zu prüfen sind:

- Dienstneustarts,
- Kernelmeldungen,
- Link-Up- und Link-Down-Ereignisse,
- DHCP-Ereignisse,
- DNS-Fehler,
- Speicherfehler,
- Out-of-Memory-Ereignisse,
- Datenträger- und Dateisystemfehler,
- Authentifizierungsfehler,
- Containerneustarts,
- Prozessabbrüche.

Ob ältere Meldungen verfügbar sind, hängt von der Konfiguration und Aufbewahrung des Journals ab.

---

15. Protokolle zentral zusammenführen

Wenn mehrere Systeme beteiligt sind, reichen lokale Protokolle häufig nicht aus.

Mögliche Quellen:

- Client,
- Server,
- Switch,
- Router,
- Firewall,
- Access Point,
- VPN-Gateway,
- DNS-Server,
- DHCP-Server,
- Verzeichnisdienst,
- Hypervisor,
- Containerplattform,

- Anwendung,
- Datenbank,
- Storage,
- Monitoring,
- Cloud- oder Providerdienst.

Vorteile einer zentralen Protokollierung:

- gemeinsame Suche,
- einheitliche Aufbewahrung,
- Vergleich mehrerer Systeme,
- Alarmierung,
- Erkennung wiederkehrender Muster,
- Ereigniszeitleisten,
- langfristige Auswertung.

Dabei müssen Zeitsynchronisation, Zugriffsschutz, Speicherbedarf und Datenschutz berücksichtigt werden.

---

16. Auflösung des Monitorings erhöhen

Ein Fünf-Minuten-Mittelwert kann einen Fehler von zehn Sekunden vollständig verdecken.

Beispiel

```
Messintervall: 5 Minuten
Störung:       10 Sekunden
Linklast:      während der Störung 100 %
Mittelwert:    nur 8 %
```

Der Mittelwert wirkt unauffällig, obwohl kurzzeitig eine vollständige Überlastung bestand.

Zu prüfen sind:

- Messintervall,
- Aggregation,
- Durchschnitt gegenüber Maximum,
- kurzfristige Spitzen,
- Anzahl der Messwerte,
- Aufbewahrungsstufen,
- Alarmverzögerung,
- Alarmrücksetzung,
- fehlende Messwerte.

Mögliche Anpassung:

```
Vorher: Messung alle 5 Minuten
Temporär: Messung alle 10 Sekunden
```

Eine höhere Auflösung erzeugt mehr Last und benötigt mehr Speicher. Sie sollte kontrolliert, zeitlich begrenzt und auf die erforderlichen Messwerte beschränkt werden.

---

17. Ringpuffer-Paketmitschnitt verwenden

Wenn ein Netzwerkfehler nur selten auftritt, kann ein begrenzter Ringpuffer-Mitschnitt die Pakete vor und während des Fehlers erhalten.

Verfügbare Schnittstellen anzeigen:

```
dumpcap -D
```

Beispiel für einen Ringpuffer:

```
dumpcap \
    -i 1 \
    -b duration:300 \
    -b files:12 \
    -w sporadischer-fehler.pcapng
```

Bedeutung:

```
-i 1              Schnittstelle 1
-b duration:300   alle 300 Sekunden neue Datei
-b files:12       höchstens 12 Ringpufferdateien
-w                Ausgabedatei
```

Damit werden ungefähr die letzten 60 Minuten vorgehalten:

```
12 Dateien × 5 Minuten = 60 Minuten
```

Mitschnitt auf einen bestimmten Host begrenzen:

```
dumpcap \
  -i 1 \
  -f "host 192.0.2.45" \
  -b duration:300 \
  -b files:12 \
  -w sporadischer-fehler.pcapng
```

Wichtig

- richtige Netzwerkschnittstelle auswählen,
- Sichtbarkeit des relevanten Verkehrs prüfen,
- Speicherbedarf begrenzen,
- Capture-Filter möglichst eng setzen,
- Systembelastung beobachten,
- Zeitpunkt des Fehlers sofort dokumentieren,
- betroffene Dateien vor dem Überschreiben sichern,
- Mitschnitt nach der Diagnose beenden,
- Dateien geschützt speichern und anschließend geregelt löschen.

Ein Paketmitschnitt auf dem Client sieht nicht automatisch den gesamten Netzwerkverkehr. Für andere Segmente kann ein autorisierter Mirror- beziehungsweise SPAN-Port erforderlich sein.

---

18. Interface-Flapping untersuchen

Interface-Flapping bedeutet, dass eine Netzwerkschnittstelle wiederholt zwischen aktiv und inaktiv wechselt.

Typisches Muster:

```
14:17:31 Interface down
14:17:34 Interface up
14:18:02 Interface down
14:18:05 Interface up
```

Unter Windows:

```
Get-NetAdapter |
  Format-Table Name, Status, LinkSpeed, InterfaceDescription
```

Statistiken:

```
Get-NetAdapterStatistics
```

Unter Linux:

```
ip link show
```

```
ip -s link show
```

Kernelmeldungen zum Netzwerk:

```
journalctl -k |
    grep -Ei "link.*(up|down)|carrier|network"
```

Auf Cisco-IOS-/IOS-XE-Switches beispielsweise:

```
show logging
```

```
show interfaces status
```

```
show interfaces <interface>
```

Zu prüfen sind:

- Link-Up- und Link-Down-Zeitpunkte,
- Anzahl der Statuswechsel,
- CRC- oder FCS-Fehler,
- Interface Resets,
- Kabel,
- Stecker,
- Switchport,
- Transceiver,
- Dockingstation,
- Netzwerkkartentreiber,
- Energiesparzustand,
- Power over Ethernet,
- Port-Security,
- Spanning Tree,
- Aushandlung von Geschwindigkeit und Duplex.

Ein Link-Flap kann sehr kurz sein und zwischen zwei manuellen Prüfungen vollständig verschwinden. Protokolle und historische Zähler sind deshalb besonders wichtig.

---

19. Fehlerzähler als Verlauf beobachten

Ein einzelner Zählerstand zeigt nicht, wann ein Fehler entstanden ist. Deshalb müssen Ausgangswert und späterer Wert verglichen werden.

Beispiel ohne aktuelle Zunahme

```
14:00 Uhr: 25 CRC-Fehler
15:00 Uhr: 25 CRC-Fehler
```

Der Zähler enthält historische Fehler, steigt im Beobachtungszeitraum jedoch nicht.

Beispiel mit sporadischer Zunahme

```
14:00 Uhr: 25 CRC-Fehler
14:16 Uhr: 25 CRC-Fehler
14:18 Uhr: 4.260 CRC-Fehler
14:20 Uhr: 4.260 CRC-Fehler
```

Die Fehler entstehen in einem kurzen Zeitfenster und korrelieren möglicherweise mit der gemeldeten Störung.

Zu beobachten sind:

- CRC-/FCS-Fehler,
- Input Errors,
- Output Errors,
- Discards,
- Drops,
- Interface Resets,
- Kollisionen,
- Retransmissions,
- WLAN-Retries,
- Queue Drops,
- verlorene Pakete,
- Linkstatuswechsel.

Fehlerzähler sollten erst nach der Dokumentation und nur nach betrieblicher Freigabe zurückgesetzt werden.

---

20. DHCP-Lease und Adresskonflikte prüfen

Ein Fehler kann beim Erneuern einer DHCP-Lease oder durch eine doppelt verwendete IP-Adresse auftreten.

Unter Windows:

```
ipconfig /all
```

Zu prüfen sind:

- DHCP aktiviert,
- Lease erhalten,
- Lease läuft ab,
- DHCP-Server,
- aktuelle IPv4- und IPv6-Adresse,
- Standardgateway,
- DNS-Server,
- unerwartete Adressänderung,
- APIPA-Adresse,
- doppelte IP-Adresse.

Unter Linux, abhängig vom verwendeten Netzwerkdienst:

```
journalctl |
    grep -Ei "dhcp|lease"
```

Mögliche Hinweise:

```
Fehler tritt regelmäßig bei Lease-Erneuerung auf.
Client erhält kurzzeitig keine gültige Adresse.
Zwei Geräte verwenden dieselbe statische IP-Adresse.
DHCP-Server sind nicht einheitlich konfiguriert.
```

Eine zeitliche Übereinstimmung zwischen Lease-Erneuerung und Fehler ist ein Hinweis, aber noch kein Beweis für die Ursache.

---

21. DNS-TTL und DNS-Auflösung prüfen

Ein sporadischer Fehler kann auftreten, wenn zwischengespeicherte DNS-Einträge ablaufen und anschließend eine fehlerhafte oder langsame Auflösung erfolgt.

Unter Windows:

```
Resolve-DnsName <hostname>
```

DNS-Cache anzeigen:

```
ipconfig /displaydns
```

Unter Linux oder macOS, sofern `dig` vorhanden ist:

```
dig <hostname>
```

Wiederholte Abfrage:

```
dig <hostname> +noall +answer
```

Zu prüfen sind:

- Antwortzeit,
- zurückgegebene IP-Adressen,
- TTL,
- mehrere DNS-Server,
- unterschiedliche Antworten,
- fehlgeschlagene Abfragen,
- kurze TTL,
- veraltete Einträge,
- fehlerhafte Weiterleitung,
- DNSSEC-Fehler,
- Suchdomänen,
- Split-DNS,
- Wechsel zwischen funktionierenden und fehlerhaften Zieladressen.

Beispiel

```
DNS-Antwort 1: 192.0.2.80 – Dienst funktioniert
DNS-Antwort 2: 192.0.2.81 – Dienst nicht erreichbar
```

In diesem Fall kann der Fehler nur bei Auswahl einer bestimmten Zieladresse auftreten.

DNS-Caches dürfen nicht pauschal gelöscht werden, bevor der aktuelle Zustand dokumentiert wurde. Dadurch könnten wichtige Diagnoseinformationen verloren gehen.

22. Sitzungen, Token und Timeouts berücksichtigen

Viele sporadische Fehler treten nach einer bestimmten Zeitspanne auf.

Mögliche Zeitgrenzen:

- Sitzungs-Timeout,
- Idle-Timeout,
- Access-Token-Ablauf,
- Refresh-Token-Ablauf,
- Kerberos-Ticketlaufzeit,
- VPN-Sitzungslaufzeit,
- Firewall-Session-Timeout,
- Proxy-Timeout,
- Load-Balancer-Timeout,
- Datenbankverbindungs-Timeout,
- DHCP-Lease,
- DNS-TTL,
- Zertifikatsablauf,
- API-Ratenlimit,
- Sperrzeit einer Anwendung.

Beispiel

```
Anmeldung:            08:00 Uhr
Fehler:               16:00 Uhr
Zeit bis zum Fehler:  8 Stunden
```

Tritt der Fehler regelmäßig nach fast derselben Sitzungsdauer auf, sind zeitbasierte Gültigkeiten und Timeouts zu prüfen.

Zu dokumentieren sind:

- Zeitpunkt der Anmeldung,
- Zeitpunkt der letzten Aktivität,
- Zeitpunkt der Tokenausstellung,
- Ablaufzeit,
- Zeitpunkt des Fehlers,
- erfolgreiche oder fehlgeschlagene Erneuerung,
- Verhalten nach erneuter Anmeldung.

Ein erneutes Anmelden kann den Fehler vorübergehend beseitigen, beweist aber noch nicht, welche Komponente die Sitzung beendet hat.

23. Zertifikats- und Gültigkeitszeiträume prüfen

Zertifikatsprobleme können nur einzelne Systeme, Dienste oder Verbindungswege betreffen.

Mit OpenSSL, sofern verfügbar:

```
openssl s_client \
    -connect <server>:443 \
    -servername <hostname>
```

Nur Zertifikatsdaten ausgeben:

```
openssl s_client \
    -connect <server>:443 \
    -servername <hostname> \
    </dev/null 2>/dev/null |
    openssl x509 -noout -subject -issuer -dates
```

Zu prüfen sind:

- Beginn der Gültigkeit,
- Ablaufdatum,
- vollständige Zertifikatskette,
- Hostname,
- Zwischenzertifikate,
- Systemzeit,
- unterschiedliche Zertifikate an mehreren Servern,
- Load-Balancer-Knoten,
- Proxy- oder TLS-Inspection,
- automatische Zertifikatserneuerung.

Beispiel

```
Load-Balancer-Knoten 1: gültiges Zertifikat
Load-Balancer-Knoten 2: abgelaufenes Zertifikat
```

Der Fehler erscheint dadurch möglicherweise nur bei Verbindungen, die auf den zweiten Knoten verteilt werden.

---

24. Geplante Aufgaben und Timer prüfen

Regelmäßig auftretende Fehler können mit zeitgesteuerten Prozessen zusammenhängen.

Unter Windows:

```
Get-ScheduledTask |
    Select-Object TaskName, TaskPath, State
```

Weitere Informationen:

```
Get-ScheduledTask |
    Get-ScheduledTaskInfo |
    Select-Object TaskName, LastRunTime, LastTaskResult, NextRunTime
```

Unter Linux mit systemd:

```
systemctl list-timers --all
```

Cron-Einträge müssen entsprechend der verwendeten Distribution und betrieblichen Berechtigung geprüft werden.

Mögliche zeitgesteuerte Auslöser:

- Backups,
- Datenbankwartung,
- Logrotation,
- Virenscans,
- Softwareverteilung,
- Patchmanagement,
- Replikation,
- Cloud-Synchronisation,
- Storage-Snapshots,
- Zertifikatserneuerung,
- Berichterstellung,
- Dateiimporte,
- Exporte,
- Neustarts,
- Container-Updates,
- automatische Skalierung,
- Indexierung.

---

25. Backups, Scans und Updates zeitlich vergleichen

Beispiel

```
02:00:00 Backup startet
02:00:12 Storage-Latenz steigt
02:00:18 Datenbankantwortzeit steigt
02:00:25 Anwendung meldet Timeout
02:20:00 Backup endet
02:20:15 Storage-Latenz normalisiert sich
```

Diese Ereignisse bilden eine plausible Kette:

```
Backup
  ↓
hohe Storage-Last
  ↓
langsame Datenbank
  ↓
Anwendungs-Timeout
```

Die zeitliche Korrelation ist stark, muss aber durch Messwerte bestätigt werden.

Zu vergleichen sind:

- Start- und Endzeit,
- CPU-Auslastung,
- Arbeitsspeicher,
- Storage-Latenz,
- Netzwerkauslastung,
- Warteschlangen,
- Anwendungsantwortzeit,
- Anzahl der Fehler,
- Verhalten ohne den geplanten Prozess.

Produktive Sicherungen oder Sicherheitsprüfungen dürfen nicht ohne Freigabe deaktiviert werden. Eine kontrollierte Zeitplanänderung ist häufig sicherer als das vollständige Abschalten.

---

26. Temperatur und Stromversorgung berücksichtigen

Sporadische Hardwarefehler können durch Umwelt- oder Strombedingungen ausgelöst werden.

Zu prüfen sind:

- CPU- und Gerätetemperatur,
- Lüfterdrehzahl,
- Überhitzungswarnungen,
- Netzteilstatus,
- redundante Netzteile,
- USV-Ereignisse,
- Spannungsschwankungen,
- PoE-Leistungsbudget,
- Stromausfälle,
- fehlerhafte Steckverbindungen,
- Temperatur im Serverschrank,
- Tageszeit und Umgebungstemperatur.

Typisches Muster:

```
Temperatur steigt
      ↓
Gerät reduziert Leistung oder schaltet Komponente ab
      ↓
Dienst oder Verbindung fällt kurzzeitig aus
      ↓
Temperatur sinkt
      ↓
System funktioniert wieder
```

Hardwaregrenzwerte und Diagnosebefehle sind hersteller- und modellspezifisch und müssen anhand der offiziellen Gerätedokumentation geprüft werden.

---

27. WLAN-Störungen untersuchen

Kurzzeitige WLAN-Probleme können durch wechselnde Funkbedingungen entstehen.

Zu prüfen sind:

- Signalstärke,
- Signal-Rausch-Abstand,
- Kanalauslastung,
- Retry-Rate,
- Roaming,
- Kanalwechsel,
- Radarerkennung bei DFS-Kanälen,
- Access-Point-Neustarts,
- Anzahl aktiver Clients,

- Airtime,
- fremde WLAN-Netze,
- Bluetooth- oder andere Funkquellen,
- Mikrowellengeräte,
- bauliche Hindernisse,
- bewegliche Störquellen,
- fehlerhafter AP-Uplink,
- PoE-Unterbrechungen.

Unter Windows:

```
netsh wlan show interfaces
```

Bericht zur WLAN-Nutzung:

```
netsh wlan show wlanreport
```

Der erzeugte Bericht enthält Informationen zu WLAN-Sitzungen und Verbindungsereignissen. Speicherort und Zugriff müssen auf dem jeweiligen System geprüft werden.

Vergleich

Ethernet	WLAN	Eingrenzung
stabil	sporadische Ausfälle	Funkstrecke, Roaming oder Access Point
beide gleichzeitig gestört	gemeinsamer Pfad oder Zielsystem	
nur ein WLAN-Client betroffen	Client, Treiber, Standort oder Frequenzband	
alle Clients eines AP betroffen	Access Point, Funkkanal, PoE oder Uplink	

---

28. Provider- und WAN-Störungen berücksichtigen

Wenn nur externe oder standortübergreifende Verbindungen betroffen sind, müssen auch WAN- und Providerpfade untersucht werden.

Zu prüfen sind:

- Verlust zum Provider-Gateway,
- Latenzsprünge,
- Leitungsstatus,
- Modem- oder Routerprotokolle,

- WAN-Interface-Fehler,
- Neuverbindungen,
- öffentliche IP-Adresswechsel,
- PPPoE-Sitzungen,
- Mobilfunkqualität,
- Providerwartungen,
- BGP- oder Routingänderungen,
- VPN-Neuaufbau,
- Auslastung der Standortanbindung,
- beide Übertragungsrichtungen.

Ein einzelner öffentlicher Speedtest reicht nicht aus. Benötigt werden wiederholte und möglichst vergleichbare Messungen zu internen und externen Referenzzielen.

---

29. Mehrere Systeme auf einer gemeinsamen Zeitleiste korrelieren

Die wichtigste Methode bei sporadischen Fehlern ist eine gemeinsame Ereigniszeitleiste.

Beispiel

Zeit	Client	Switch	Firewall	Server	Anwendung
14:17:30	normal	normal	normal	normal	normal
14:17:32	Verbindung hängt	Uplink-Drops steigen	Sitzungen normal	Anfragen warten	Timeout beginnt
14:17:35	Ziel nicht erreichbar	Uplink 100 %	Paketverluste	keine neuen Anfragen	Fehlermeldung
14:17:42	Verbindung wieder da	Drops sinken	normal	Anfragen treffen ein	Wiederherstellung
14:18:00	normal	normal	normal	normal	normal

Die gemeinsame Zeitleiste zeigt, welche Veränderung zuerst auftrat.

```
Uplink-Auslastung und Drops
        ↓
Paketverlust
        ↓
Server nicht erreichbar
        ↓
Anwendungs-Timeout
```

Ein Anwendungsfehler am Ende der Kette ist in diesem Beispiel nicht automatisch die ursprüngliche Ursache.

30. Reihenfolge von Ursache und Folge beachten

Gleichzeitig protokollierte Fehler können unterschiedliche Rollen besitzen.

Beispiel

```
14:17:32 Switchport verliert Link
14:17:33 Client verliert Netzwerk
14:17:34 DNS-Abfrage schlägt fehl
14:17:35 Anwendung meldet Serverfehler
```

Mögliche Reihenfolge:

```
Switchportproblem
    ↓
Netzwerkverlust
    ↓
DNS-Fehler
    ↓
Anwendungsfehler
```

Der DNS-Fehler und der Anwendungsfehler sind hier wahrscheinlich Folgen des vorherigen Linkverlusts.

Deshalb ist zu fragen:

- Welches Ereignis trat zuerst auf?
- Welche Systeme sind voneinander abhängig?
- Welches Ereignis erklärt die nachfolgenden Fehler?
- Welcher Messwert änderte sich bereits vor der Benutzermeldung?
- Welche Meldung beschreibt nur eine Folge?

---

31. Kontrollierte Reproduktion versuchen

Wenn ein Muster erkannt wurde, kann eine kontrollierte Reproduktion möglich sein.

Beispiele:

- gleiche Aktion wiederholen,
- gleiche Sitzungsdauer abwarten,
- denselben Netzwerkpfad verwenden,
- Last unter kontrollierten Bedingungen erzeugen,

- Test vor und während eines geplanten Prozesses durchführen,
- betroffenes und funktionierendes Gerät vergleichen,
- unterschiedliche Serverknoten gezielt prüfen,
- WLAN und Ethernet vergleichen.

Dabei gelten folgende Regeln:

1. nur autorisierte Tests durchführen,
2. Ausgangszustand dokumentieren,
3. erwartete Beobachtung festlegen,
4. nur einen Einflussfaktor verändern,
5. Messwerte während des Tests erfassen,
6. Test bei unerwarteten Auswirkungen abbrechen,
7. ursprünglichen Zustand wiederherstellen,
8. Ergebnis dokumentieren.

Ein produktiver Fehler darf nicht absichtlich durch riskante Last, Stromunterbrechung, Sicherheitsdeaktivierung oder unkontrollierte Netzwerkänderung provoziert werden.

---

32. Hypothese messbar formulieren

Ungeeignet:

```
Vielleicht ist das Netzwerk schuld.
```

Besser:

```
Wenn der Fehler auftritt, verliert der Client vermutlich kurzzeitig
die Verbindung zum Standardgateway.
```

Prüfung:

```
Client → Gateway kontinuierlich messen
Client → Server kontinuierlich messen
Client- und Switchportprotokolle vergleichen
```

Noch genauer:

```
Wenn NB-023 den Fehler erneut zeigt, muss innerhalb desselben
Zeitfensters entweder ein Link-Down-Ereignis am Client oder am
Switchport Gi1/0/12 sichtbar sein.
```

Eine gute Hypothese enthält:

- erwartetes Ereignis,
- betroffene Komponente,
- Messwert,
- Zeitfenster,
- mögliche Widerlegung.

---

33. Genau eine kontrollierte Änderung durchführen

Nach ausreichender Eingrenzung wird nur eine Änderung vorgenommen.

Mögliche kontrollierte Änderungen:

- defektes Kabel ersetzen,
- betroffenen Switchport wechseln,
- vorgesehenen Treiber aktualisieren,
- fehlerhaften Serverknoten aus dem Load Balancer nehmen,
- geplanten Prozess nach Freigabe verschieben,
- Sitzungs-Timeout abgestimmt korrigieren,
- fehlerhafte DNS-Antwort korrigieren,
- defekten Access Point ersetzen,
- vorgesehene Bandbreitensteuerung anpassen,
- fehlerhafte Stromversorgung austauschen.

Nicht mehrere Änderungen gleichzeitig durchführen.

Ungeeignet:

```
Kabel, Switchport, Treiber und DNS gleichzeitig ändern
```

Danach wäre nicht mehr feststellbar, welche Änderung wirksam war.

---

34. Rollback festlegen

Vor der Änderung dokumentieren:

- ursprüngliche Konfiguration,
- ursprüngliche Version,
- ursprünglicher Zeitplan,
- ursprünglicher Port,
- ursprüngliche Verkabelung,
- Sicherung vorhandener Einstellungen,
- verantwortliche Person,

- erlaubtes Änderungsfenster,
- Abbruchkriterium,
- Rückweg.

Beispiel

```
Änderung:
Backupstart von 14:00 Uhr auf 22:00 Uhr verschieben.

Rollback:
Ursprünglichen Startzeitpunkt 14:00 Uhr wiederherstellen.

Abbruchkriterium:
Sicherung startet nicht oder überschneidet sich mit einem anderen Auftrag.
```

---

35. Verifikation über ausreichend lange Zeit durchführen

Bei einem sporadischen Fehler reicht ein einzelner erfolgreicher Test nicht aus.

Beispiel

```
Fehler trat bisher ungefähr einmal täglich auf.
Nach der Änderung war das System 15 Minuten stabil.
```

Dieses Ergebnis ist noch kein ausreichender Nachweis.

Besser:

```
Beobachtungszeitraum vor Änderung: 7 Tage
Fehlerhäufigkeit vorher:          1 bis 3 Ereignisse pro Tag
Beobachtungszeitraum nach Änderung: 7 Tage
Fehlerhäufigkeit nachher:         0 Ereignisse
```

Zusätzlich prüfen:

- ursprüngliche Symptome,
- technische Messwerte,
- Protokolle,
- Fehlerzähler,
- Nebenwirkungen,
- Sicherheitsfunktionen,

- abhängige Systeme,
- Benutzerbestätigung.

---

36. Beispiel für eine systematische Diagnose

Symptom

Mehrere Benutzer melden an Werktagen gegen 14:17 Uhr kurze Verbindungsabbrüche zu verschiedenen internen Diensten.

Erste Beobachtung

```
Dauer:              10 bis 20 Sekunden
Betroffene Clients: mehrere
Betroffene Dienste: ERP, Dateiserver und Intranet
Gateway:            während der Störung erreichbar
Interne Server:     während der Störung teilweise nicht erreichbar
```

Kontinuierliche Messung

```
14:17:30 Gateway:       1 ms
14:17:30 Dateiserver:   2 ms
14:17:35 Gateway:       1 ms
14:17:35 Dateiserver:   Zeitüberschreitung
14:17:40 Gateway:       1 ms
14:17:40 Dateiserver:   Zeitüberschreitung
14:17:45 Gateway:       1 ms
14:17:45 Dateiserver:   2 ms
```

Switch-Monitoring

```
14:17:33 Server-Uplink erreicht 100 % Auslastung
14:17:34 Output Drops steigen
14:17:44 Auslastung sinkt
14:17:45 Output Drops steigen nicht weiter
```

Zeitgesteuerte Aufgaben

```
14:17:30 Replikationsauftrag startet
14:17:45 Replikationsauftrag beendet
```

Hypothese

Der Replikationsauftrag überlastet kurzzeitig den gemeinsamen Server-Uplink. Die entstehenden Warteschlangen und Drops verursachen Paketverlust und Anwendungsabbrüche.

Kontrollierte Maßnahme

Nach betrieblicher Freigabe wird der Replikationsauftrag in ein geeignetes Zeitfenster verschoben und mit der vorgesehenen Bandbreitensteuerung versehen.

Rollback

```
Ursprünglichen Zeitplan und ursprüngliche Bandbreitenrichtlinie wiederherstellen.
```

Verifikation

```
Beobachtungszeitraum: 7 Tage
Unterbrechungen:      keine
Output Drops:         keine neue Zunahme
Anwendungsfehler:     keine
Replikation:          weiterhin erfolgreich
```

Festgestellte Ursache

Ein regelmäßig gestarteter Replikationsauftrag erzeugte eine kurzzeitige Überlastung des gemeinsamen Uplinks.

---

37. Beispiel „Fehler nach acht Stunden“

Symptom

Ein Benutzer wird fast täglich am Nachmittag aus einer Anwendung abgemeldet.

Zeitvergleich

```
Anmeldung:       08:05 Uhr
Fehler:          16:05 Uhr
Sitzungsdauer:   8 Stunden
```

Weitere Tage:

```
Anmeldung 07:58 Uhr → Fehler 15:58 Uhr
Anmeldung 08:12 Uhr → Fehler 16:12 Uhr
Anmeldung 08:03 Uhr → Fehler 16:03 Uhr
```

Das Problem tritt nicht zu einer festen Uhrzeit, sondern nach einer festen Sitzungsdauer auf.

Prüfbereich

- Access-Token,
- Sitzungs-Timeout,
- Refresh-Token,
- Load Balancer,
- Proxy,
- Anwendungssitzung,
- Authentifizierungsdienst.

Festgestellte Ursache

Die Anwendung erneuerte ein nach acht Stunden ablaufendes Token nicht korrekt.

Lehre

```
Feste Uhrzeit und feste Laufzeit müssen unterschieden werden.
```

---

38. Beispiel „sporadischer Fehler durch einen Serverknoten“

Symptom

Eine Webanwendung funktioniert meistens, zeigt aber gelegentlich einen TLS- oder Serverfehler.

Aufbau

```
Client
  |
Load Balancer
  |
  +-- Webserver 1
  +-- Webserver 2
  +-- Webserver 3
```

Beobachtung

Zielknoten	Ergebnis
Webserver 1	funktioniert
Webserver 2	funktioniert
Webserver 3	Fehler
Zugriff über Load Balancer	nur gelegentlich Fehler

Festgestellte Ursache

Nur Webserver 3 besitzt eine fehlerhafte Konfiguration. Der Fehler tritt deshalb nur auf, wenn der Load Balancer eine Anfrage diesem Knoten zuweist.

Lehre

Bei verteilten Systemen müssen alle Knoten einzeln verglichen werden.

---

39. Ungeeignete Sofortmaßnahmen

Nicht als erste Maßnahme verwenden:

- System ohne Protokollsicherung neu starten,
- Fehler als „nicht reproduzierbar“ schließen,
- alle Protokolle pauschal löschen,
- Fehlerzähler vor der Dokumentation zurücksetzen,
- DNS-Cache vor der Bestandsaufnahme leeren,
- mehrere Komponenten gleichzeitig ersetzen,
- Monitoring unbegrenzt hochauflösend aktivieren,
- unbegrenzten Paketmitschnitt starten,
- Sicherheitsfunktionen deaktivieren,
- VPN oder Authentifizierung umgehen,
- Backups ohne Freigabe abschalten,
- Sitzungszeiten pauschal erhöhen,
- automatische Updates vollständig deaktivieren,
- WLAN-Kanäle wahllos verändern,
- Produktivsysteme unkontrolliert belasten,
- aus einem einzelnen Ereignis eine Ursache ableiten,
- nur auf die Aussage „jetzt funktioniert es“ vertrauen.

Solche Maßnahmen können Beweise vernichten, neue Störungen verursachen oder Sicherheits- und Betriebsvorgaben verletzen.

---

40. Vollständige Prüfreihenfolge

1. genaue Fehlermeldung und Benutzeraktion erfassen.
2. exakten Zeitpunkt und Dauer dokumentieren.
3. betroffenen Benutzer, Client und Standort bestimmen.
4. betroffene Anwendung und Zielsysteme erfassen.
5. prüfen, ob ein oder mehrere Benutzer betroffen sind.
6. feste Uhrzeit von fester Laufzeit unterscheiden.
7. Häufigkeit und wiederkehrendes Muster bestimmen.
8. Zeitsynchronisation aller beteiligten Systeme prüfen.
9. Monitoring- und Protokollaufbewahrung prüfen.
10. kontinuierliche Messung mit Zeitstempeln einrichten.
11. Gateway, internes Ziel und externes Ziel vergleichen.
12. TCP-Port und Anwendungsantwort getrennt prüfen.
13. Windows- oder Linux-Protokolle zeitlich filtern.
14. Client-, Netzwerk- und Serverprotokolle vergleichen.
15. Monitoringauflösung kontrolliert erhöhen.
16. Schnittstellenstatus und Link-Flaps beobachten.
17. Fehler- und Verwerfungszähler als Verlauf erfassen.
18. WLAN-Ereignisse, Roaming und Retry-Rate prüfen.
19. DHCP-Lease und mögliche Adresskonflikte untersuchen.
20. DNS-Auflösung, TTL und unterschiedliche Antworten prüfen.
21. Sitzungslaufzeiten, Token und Timeouts vergleichen.
22. Zertifikatslaufzeiten und Serverknoten prüfen.
23. geplante Aufgaben, Timer und Cronjobs untersuchen.
24. Backups, Scans, Updates und Replikationen vergleichen.
25. CPU, Arbeitsspeicher, Storage und Netzwerk überwachen.
26. Temperatur, Stromversorgung und PoE berücksichtigen.
27. WAN-, Provider- und VPN-Ereignisse prüfen.
28. bei Bedarf autorisierten Ringpuffer-Mitschnitt einrichten.
29. alle Ereignisse auf einer gemeinsamen Zeitleiste darstellen.
30. zuerst auftretendes Ereignis von Folgefehlern unterscheiden.
31. genau eine messbare Hypothese formulieren.
32. kontrollierte Reproduktion planen.
33. Ausgangszustand und Rollback dokumentieren.
34. genau eine freigegebene Änderung durchführen.
35. ursprüngliche Messungen wiederholen.
36. ausreichend langen Beobachtungszeitraum einhalten.
37. Nebenwirkungen und Sicherheitsfunktionen kontrollieren.
38. temporäre Diagnosemaßnahmen entfernen.
39. Ursache, Maßnahme, Rollback und Nachweis dokumentieren.

---

41. Checkliste „Fehler tritt nur manchmal auf“

- [ ] genaue Fehlermeldung wurde dokumentiert.
- [ ] Benutzeraktion wurde beschrieben.

- [ ] exakter Zeitpunkt wurde erfasst.
- [ ] Dauer des Fehlers wurde erfasst.
- [ ] betroffener Benutzer wurde bestimmt.
- [ ] betroffener Client wurde bestimmt.
- [ ] Standort und Netzwerkverbindung wurden erfasst.
- [ ] Anwendung und Zielsystem wurden bestimmt.
- [ ] Anzahl der betroffenen Benutzer wurde geprüft.
- [ ] feste Uhrzeit und feste Laufzeit wurden unterschieden.
- [ ] Häufigkeit wurde dokumentiert.
- [ ] wiederkehrendes Muster wurde gesucht.
- [ ] Zeitzonen wurden berücksichtigt.
- [ ] Zeitsynchronisation wurde geprüft.
- [ ] kontinuierliche Messung wurde eingerichtet.
- [ ] Messungen enthalten Zeitstempel.
- [ ] Gateway wurde überwacht.
- [ ] internes Ziel wurde überwacht.
- [ ] externes Ziel wurde berücksichtigt.
- [ ] TCP-Port wurde getrennt geprüft.
- [ ] Anwendungsantwort wurde getrennt geprüft.
- [ ] Windows-Ereignisprotokolle wurden geprüft.
- [ ] Linux-Systemprotokolle wurden geprüft.
- [ ] Anwendungsprotokolle wurden geprüft.
- [ ] Netzwerkgeräteprotokolle wurden geprüft.
- [ ] Protokollaufbewahrung wurde geprüft.
- [ ] Monitoringauflösung wurde geprüft.
- [ ] kurzfristige Spitzen wurden berücksichtigt.
- [ ] Interface-Flapping wurde geprüft.
- [ ] Fehlerzähler wurden als Verlauf dokumentiert.
- [ ] CRC-/FCS-Fehler wurden berücksichtigt.
- [ ] Drops und Discards wurden berücksichtigt.
- [ ] DHCP-Lease wurde geprüft.
- [ ] IP-Adresskonflikte wurden berücksichtigt.
- [ ] DNS-Auflösung wurde wiederholt geprüft.
- [ ] DNS-TTL wurde berücksichtigt.

- [ ] unterschiedliche DNS-Antworten wurden verglichen.
- [ ] Sitzungs- und Idle-Timeouts wurden geprüft.
- [ ] Tokenlaufzeiten wurden geprüft.
- [ ] Zertifikatslaufzeiten wurden geprüft.
- [ ] alle Serverknoten wurden verglichen.
- [ ] geplante Aufgaben wurden geprüft.
- [ ] Backups und Replikationen wurden berücksichtigt.
- [ ] Virenscans und Updates wurden berücksichtigt.
- [ ] CPU- und Speicherauslastung wurden beobachtet.
- [ ] Storage-Latenz wurde beobachtet.
- [ ] WLAN-Roaming und Funkstörungen wurden berücksichtigt.
- [ ] Provider-, WAN- und VPN-Ereignisse wurden geprüft.
- [ ] Temperatur und Stromversorgung wurden berücksichtigt.
- [ ] erforderlicher Paketmitschnitt wurde autorisiert.
- [ ] Ringpuffer und Speichergrenze wurden festgelegt.
- [ ] gemeinsame Ereigniszeitleiste wurde erstellt.
- [ ] Ursache und Folge wurden unterschieden.
- [ ] genau eine Hypothese wurde formuliert.
- [ ] Ausgangszustand wurde dokumentiert.
- [ ] Rollback wurde festgelegt.
- [ ] nur eine kontrollierte Änderung wurde durchgeführt.
- [ ] ausreichend lange Verifikation wurde durchgeführt.
- [ ] Nebenwirkungen wurden geprüft.
- [ ] temporäre Diagnosemaßnahmen wurden entfernt.
- [ ] Ursache, Maßnahme und Nachweis wurden dokumentiert.

---

42. Schnellreferenz

Beobachtung	Nächster Prüfbereich
Fehler immer zur gleichen Uhrzeit	geplante Aufgaben, Backups, Scans oder Updates
Fehler immer nach gleicher Laufzeit	Sitzung, Token, Lease, Timeout oder Zertifikat
nur ein Client betroffen	Client, Treiber, Kabel, WLAN oder lokale Software
mehrere Clients gleichzeitig betroffen	gemeinsame Infrastruktur oder zentraler Dienst
Gateway ebenfalls nicht erreichbar	Clientzugang, WLAN, Switchport oder lokales Netz

Beobachtung	Nächster Prüfbereich
Gateway erreichbar, Server nicht	interner Folgepfad, Servernetz oder Server
nur Anwendung fehlerhaft	Dienst, Sitzung, Datenbank oder Anwendung
nur ein Serverknoten fehlerhaft	Konfiguration dieses Knotens
Link-Up-/Link-Down-Ereignisse	Kabel, Port, Transceiver, Treiber oder Stromversorgung
CRC-Fehler steigen kurzzeitig	physischer Übertragungsweg
Output Drops steigen kurzzeitig	Auslastung oder nachgelagerter Engpass
Fehler während Backup	Netzwerk-, CPU- oder Storage-Auslastung vergleichen
Fehler beim DNS-TTL-Ablauf	DNS-Server, Cache und Zieladressen vergleichen
Fehler bei DHCP-Erneuerung	DHCP-Server, Lease und Adresskonflikte
Fehler nur im WLAN	Funkstrecke, Roaming, Kanal, Airtime oder Access Point
Fehler nur über VPN	Tunnel, Gateway, Timeout, MTU oder Internetpfad
Fehler bei hoher Temperatur	Kühlung, Lüfter oder Hardware
Monitoring zeigt nichts	Messintervall und Aggregation prüfen
Neustart beseitigt Fehler vorübergehend	Zustand vor dem Neustart vollständig erfassen
Ereignisse verschiedener Systeme passen nicht	Uhrzeit, Zeitzone und NTP prüfen
Fehler nicht direkt reproduzierbar	kontinuierliche Messung und Ringpuffer einsetzen

Merksatz

> Bei einem sporadischen Fehler ist der genaue Zeitpunkt der wichtigste Ausgangspunkt. Messungen und Protokolle müssen bereits vor dem nächsten Auftreten bereitstehen. Erst durch Zeitstempel, wiederkehrende Muster und eine gemeinsame Ereigniszeitleiste lassen sich Ursache und Folge zuverlässig voneinander unterscheiden.

Quellen und weiterführende Dokumentation

- Microsoft Learn – Get-WinEvent
- Microsoft Learn – Test-Connection
- Microsoft Learn – Test-NetConnection
- Microsoft Learn – Get-NetAdapterStatistics
- Microsoft Learn – Resolve-DnsName
- Microsoft Learn – w32tm

- Microsoft Learn – netsh wlan
- Microsoft Learn – Get-ScheduledTask
- systemd – journalctl
- systemd – systemctl
- Wireshark – dumpcap Manual Page
- Wireshark – Capture Setup
- Wireshark – User’s Guide
- Cisco – Troubleshoot Port Flaps on Catalyst 9000 Series Switches
- Cisco – Troubleshoot Link Flap Issues on Nexus 9000