

Werkzeuge

Jedes Werkzeug kurz erklärt: wofür, wie, und wie man das Ergebnis liest.

- [DNS-Abfrage](#)
- [Mail-Check](#)
- [SSL/TLS-Check](#)
- [Blacklist-Check](#)
- [IP-Info und HTTP-Header](#)
- [Port-Check](#)
- [Speedtest](#)
- [Webhook-Tester](#)
- [Werkzeuge im Browser](#)

DNS-Abfrage

tools.ngosu-corp.de/dns fragt einen DNS-Eintrag gleichzeitig bei mehreren Stellen ab:

- dem **System-Resolver** des Servers
- den öffentlichen Resolvern **Cloudflare, Google und Quad9**
- den **zuständigen Nameservern** der Zone (markiert mit *auth*)

Typen: A, AAAA, CNAME, MX, NS, TXT und PTR.

Beispiele

Frage	Eingabe	Typ
Wohin zeigt die Domain?	example.de	A / AAAA
Wer nimmt Mails an?	example.de	MX
Wie lautet der SPF-Eintrag?	example.de	TXT
DMARC-Eintrag	_dmarc.example.de	TXT
Name zu einer IP (Reverse DNS)	1.1.1.1	PTR

So liest du das Ergebnis

- Die **Resolver** zeigen, was die meisten Nutzer gerade sehen, inklusive Zwischenspeicher.
- Die **zuständigen Nameserver** zeigen den aktuellen Stand ohne Zwischenspeicher.
- **Gelb markierte Zeilen** weichen voneinander ab. Direkt nach einer Änderung ist das normal, bis die TTL abgelaufen ist. Bleibt der Unterschied länger, ist meist ein Nameserver nicht synchron.

Typische Fragen

Ich habe den Eintrag geändert, sehe aber noch den alten. Schau auf die zuständigen Nameserver: Zeigen die schon den neuen Wert, wartest du nur noch auf den Ablauf der TTL bei den Resolvern.

Kein Ergebnis bei *auth*. Dann ist die Zone falsch delegiert oder die NS-Einträge beim Registrar stimmen nicht.

Im Terminal

```
curl tools.ngosu-corp.de/dns/example.de/MX  
curl "tools.ngosu-corp.de/dns/example.de/TXT?json"
```

Mail-Check

tools.ngosu-corp.de/mail prüft, ob eine Domain sauber für E-Mail eingerichtet ist. Du kannst eine Domain oder eine ganze Mail-Adresse eingeben.

Was geprüft wird

Prüfung	Worum es geht
MX	Welche Server Mails annehmen, mit Reverse DNS ihrer IPs
SPF	Welche Server im Namen der Domain senden dürfen. Gezählt werden auch die DNS-Lookups, mehr als 10 machen den Eintrag ungültig
DKIM	Signaturschlüssel für ausgehende Mails
DMARC	Was Empfänger mit Mails tun sollen, die SPF/DKIM nicht bestehen, und wohin Berichte gehen
MTA-STS	Erzwingt verschlüsselte Zustellung an deine Mailserver
TLS-RPT	Adresse für Berichte über Verschlüsselungsprobleme

Zu jedem Fund gibt es einen Hinweis, was fehlt und wie man es behebt.

DKIM-Selektor

DKIM-Schlüssel liegen unter `<selektor>._domainkey.<domain>`. Gängige Selektoren (Google, Microsoft 365, Mailjet ...) probiert der Check automatisch. Deinen eigenen findest du im Header einer gesendeten Mail im Feld `DKIM-Signature` hinter `s=`. Trag ihn dann im Feld *DKIM-Selektor* ein.

Häufige Befunde

- **„SPF: zu viele Lookups“**: `include:` verschachtelt zu tief. Entferne Dienste, die du nicht mehr nutzt, oder fasse IPs direkt zusammen.
- **„DMARC fehlt“**: Fang mit `v=DMARC1; p=none; rua=mailto:dmarc@example.de` an, wertet die Berichte aus und stell später auf `quarantine` oder `reject`.

- **„MX ohne Reverse DNS“:** Der PTR-Eintrag der Mailserver-IP fehlt oder passt nicht. Den setzt dein Hoster.

Eine Schritt-für-Schritt-Anleitung kommt ins Regal „Wissen für IT-Admins“.

Im Terminal

```
curl tools.ngosu-corp.de/mail/example.de  
curl "tools.ngosu-corp.de/mail/example.de?selector=s1&json"
```

SSL/TLS-Check

tools.ngosu-corp.de/ssl prüft das Zertifikat und die Verschlüsselung eines Servers. Das geht für Webserver und für Mailserver per STARTTLS.

Eingabe

- `example.de` prüft Port 443.
- `mail.example.de:25` prüft SMTP mit STARTTLS.
- `mail.example.de:993` prüft IMAPS (TLS direkt).

Unter **Verbindung** steht standardmäßig *automatisch*: Bei den Ports 25, 587, 143 und 110 wird STARTTLS benutzt, sonst TLS direkt. Du kannst die Verbindungsart auch fest wählen.

Was geprüft wird

- **Zertifikat**: für wen ausgestellt (inklusive aller Namen/SANs), Aussteller, Ablaufdatum, Schlüssel, Fingerabdruck
- **Kette**: ob alle Zwischenzertifikate mitgeschickt werden und die Kette vertrauenswürdig ist
- **Protokolle**: TLS 1.0 bis 1.3 einzeln. Gut ist: 1.2 und 1.3 an, 1.0 und 1.1 aus
- **Schwache Cipher, HSTS** und **OCSP-Stapling**

Häufige Befunde

- **„Kette unvollständig“**: Der Server schickt nur sein Zertifikat ohne Zwischenzertifikat. Browser verzeihen das manchmal, Apps und Mailserver oft nicht. In nginx `fullchain.pem` statt `cert.pem` angeben.
- **„Name passt nicht“**: Der geprüfte Name steht nicht im Zertifikat. Zertifikat mit allen Namen neu ausstellen.
- **„Läuft bald ab“**: Automatische Erneuerung prüfen, z. B. `certbot renew --dry-run`. Mit Premium kann das [Monitoring](#) rechtzeitig warnen.

Im Terminal

```
curl tools.ngosu-corp.de/ssl/example.de  
curl tools.ngosu-corp.de/ssl/mail.example.de:25  
curl "tools.ngosu-corp.de/ssl/example.de?json"
```

Blacklist-Check

tools.ngosu-corp.de/blacklist prüft, ob eine IP-Adresse oder Domain auf Spam-Sperrlisten steht. Das ist die häufigste Ursache, wenn Mails plötzlich im Spam landen oder abgewiesen werden.

- **IP-Adresse:** Abfrage bei Spamhaus, SpamCop, Barracuda, Mailspike und weiteren Listen
- **Domain:** zusätzlich Spamhaus DBL und SURBL (Domains in Links) sowie die IPs der Mailserver der Domain

Die Abfrage dauert ein paar Sekunden, weil jede Liste einzeln gefragt wird.

Gelistet, was nun?

1. **Ursache finden:** Meist sind es ein gehacktes Postfach, ein offenes Kontaktformular oder ein infizierter Rechner im Netz. Prüfe Mail-Logs und Versandmengen.
2. **Ursache abstellen,** sonst wirst du sofort wieder gelistet.
3. **Austragen:** Unter *Details und Austragen* steht zu jeder Liste der Link zum Austragen. Manche Listen tragen automatisch nach einiger Zeit aus.
4. **Sicher gehen:** Danach SPF, DKIM und DMARC im [Mail-Check](#) prüfen.

Steht deine **private IP vom Internetanschluss** auf einer Liste (z. B. Spamhaus PBL)? Das ist normal: Von solchen Anschlüssen sollen Mails nur über einen Mailserver rausgehen, nicht direkt.

Im Terminal

```
curl tools.ngosu-corp.de/blacklist/1.2.3.4  
curl "tools.ngosu-corp.de/blacklist/example.de?json"
```

IP-Info und HTTP-Header

IP-Info

tools.ngosu-corp.de/ip zeigt deine öffentliche IP-Adresse, wie sie im Internet ankommt:

- IPv4 und IPv6
- Reverse DNS (Name zur IP)
- Netz, ASN und Provider

Gezielt nur eine Protokollversion: [ip4.ngosu-corp.de](https://tools.ngosu-corp.de/ip4) und [ip6.ngosu-corp.de](https://tools.ngosu-corp.de/ip6). Zeigt *ip6* nichts an, hast du keine funktionierende IPv6-Verbindung.

```
curl ip.ngosu-corp.de          # nur die IP
curl ip.ngosu-corp.de/full    # alles als Text
curl ip.ngosu-corp.de/json    # als JSON
curl ip4.ngosu-corp.de        # erzwingt IPv4
```

Praktisch in Skripten, z. B. für dynamisches DNS: `IP=$(curl -s ip4.ngosu-corp.de)`.

HTTP-Header

tools.ngosu-corp.de/headers zeigt, was beim Server von deiner Anfrage ankommt:

- alle Request-Header, Protokoll (HTTP/1.1, 2 oder 3) und TLS-Version
- **Hinweise auf Proxys:** Header wie `X-Forwarded-For` oder `Via` zeigen, dass ein Proxy, eine Firewall oder ein Virens Scanner dazwischen sitzt

Nützlich, wenn eine Seite sich hinter der Firmen-Firewall anders verhält als zu Hause.

```
curl tools.ngosu-corp.de/headers
curl tools.ngosu-corp.de/headers/json
```

Port-Check

tools.ngosu-corp.de/ports prüft von außen, welche Ports **deiner eigenen Verbindung** erreichbar sind. Damit siehst du, ob eine Portfreigabe im Router klappt oder ob versehentlich etwas offen ist.

Geprüft wird immer nur die IP, von der die Anfrage kommt. Fremde Adressen lassen sich hier nicht prüfen, dafür gibt es den Port-Scan für freigegebene Ziele (Premium).

Gängige oder gezielte Ports

- Ohne Angabe prüft der Check gängige Ports: FTP, SSH, SMTP, DNS, HTTP, POP3, IMAP, HTTPS, SMB, RDP, VNC und weitere.
- Gezielt: einzelne Ports mit Komma, Bereiche mit Bindestrich, höchstens 20 auf einmal.
Zum Beispiel `22,443,8000-8010`.

Ergebnis lesen

Status	Bedeutung
offen	Ein Dienst antwortet. Soll der wirklich aus dem Internet erreichbar sein?
geschlossen	Dein Router lehnt ab, kein Dienst dahinter
gefiltert	Keine Antwort, eine Firewall verwirft die Pakete. Für nicht genutzte Ports ist das ideal

Achtung bei offenen Ports für RDP (3389), SMB (445), VNC (5900) oder Datenbanken: Die sollten nie direkt im Internet stehen. Besser per VPN erreichbar machen.

IPv4 und IPv6 getrennt

Router behandeln IPv4 (Portweiterleitung) und IPv6 (Firewall-Freigabe) unterschiedlich. Über ip4. bzw. ip6. prüfst du jeweils eins davon.

```
curl tools.ngosu-corp.de/ports
curl tools.ngosu-corp.de/ports/22,443,3389
```

```
curl ip6.ngosu-corp.de/ports/443
```

Speedtest

speed.ngosu-corp.de misst Ping, Jitter, Download und Upload zwischen dir und dem Server von Ngosu.

Gemessen wird mit mehreren parallelen Verbindungen. Die erste Anlaufphase zählt nicht mit, damit das Ergebnis die Leitung zeigt und nicht den Verbindungsaufbau. Die Testdaten sind zufällig, so misst niemand versehentlich die Kompression.

Für ein aussagekräftiges Ergebnis

- Wenn möglich per **Kabel** statt WLAN messen.
- **VPN** aus, andere Downloads und Streams pausieren.
- Mehrmals zu verschiedenen Tageszeiten messen. Abends ist das Netz oft voller.

Werte verstehen

- **Ping:** Zeit für eine Anfrage hin und zurück. Unter 20 ms ist sehr gut, für Videokonferenzen reichen auch 50 ms.
- **Jitter:** wie stark der Ping schwankt. Hoch heißt: Aussetzer bei Telefonie und Spielen.
- **Download / Upload:** Datenrate in Mbit/s.

Das Ergebnis zeigt die Strecke zu diesem einen Server. Liegt es weit unter deinem Tarif, teste zusätzlich beim Messdienst deines Providers bzw. der Bundesnetzagentur (breitbandmessung.de).

Im Terminal

```
curl -o /dev/null https://speed.ngosu-corp.de/download?size=100M
```

`curl` zeigt am Ende die durchschnittliche Geschwindigkeit in Bytes pro Sekunde.

Webhook-Tester

Mit dem [Webhook-Tester](#) bekommst du eine eigene Adresse und siehst live, was dort ankommt. Methode und Pfad sind egal. Praktisch zum Testen von GitHub, Stripe, Home Assistant, Monitoring-Systemen oder eigenen Skripten.

So geht's

1. **Neue Webhook-Adresse** erzeugen.
2. Die Adresse (`https://tools.ngosu-corp.de/h/...`) beim Absender eintragen.
3. Eingehende Anfragen erscheinen sofort, mit Methode, Pfad, Headern, Query und Body. JSON wird lesbar formatiert.

Zum Ausprobieren:

```
curl -d 'hallo=welt' https://tools.ngosu-corp.de/h/<deine-id>
curl -H 'Content-Type: application/json' -d '{"test": true}' https://tools.ngosu-corp.de/h/<deine-id>
```

Ohne und mit Konto

	ohne Konto	mit Konto (Premium)
Aufbewahrung	im Arbeitsspeicher, verfällt 24 h nach dem letzten Zugriff	dauerhaft, verfällt 30 Tage nach dem letzten Zugriff
Anfragen	die letzten 50, Body bis 64 KiB	die letzten 200, Body bis 256 KiB
eigene Antwort	-	Statuscode, Content-Type, Body, Verzögerung
Weiterleitung	-	an ein eigenes Ziel, wahlweise als Proxy

Ohne Konto sieht jeder die Anfragen, der den Link kennt. Teile ihn also nicht öffentlich. Konto-Adressen sieht nur ihr Besitzer.

Im Terminal

```
curl -X POST tools.ngosu-corp.de/hooks/new      # neue Adresse  
curl tools.ngosu-corp.de/hooks/<id>             # Liste und neueste Anfrage  
curl tools.ngosu-corp.de/hooks/<id>/1          # Anfrage Nummer 1
```

Werkzeuge im Browser

Unter tools.ngosu-corp.de/utils gibt es kleine Helfer. Alles läuft **komplett in deinem Browser**, nichts davon wird an den Server geschickt. Du kannst also auch echte Tokens und Passwörter hineinkopieren.

Werkzeug	Was es kann
Subnetzrechner	IPv4 und IPv6: Netz, Broadcast, erste/letzte Adresse, Anzahl, Aufteilen in Teilnetze. Eingabe z. B. <code>192.168.10.77/22</code> oder <code>2001:db8::/48</code>
Passwort-Generator	zufällig, als Passphrase oder gut aussprechbar, mit Stärke-Anzeige und Buchstabiertafel zum Vorlesen am Telefon
JWT	JSON Web Tokens dekodieren: Header, Inhalt, Ablaufzeit. Die Signatur wird nicht geprüft
Base64	kodieren und dekodieren, auch URL-sichere Variante
Zeitstempel	Unix-Zeit in Datum umrechnen und zurück
Hash und UUID	SHA-256 und Co. berechnen, UUID v4 und v7 erzeugen

Tipp: Mit  kommst du auf der Startseite direkt in die Werkzeugsuche.