7.3 Ports, Dienste und typische Standardports

Ports gehören zu OSI-Schicht 4.

Sie dienen dazu, Daten auf einem Gerät der richtigen Anwendung oder dem richtigen Dienst zuzuordnen.

Eine IP-Adresse zeigt:

```
zu welchem Gerät?
```

Ein Port zeigt:

```
zu welchem Dienst auf diesem Gerät?
```

Beispiel:

```
192.168.10.50:443
```

Bedeutung:

```
Gerät:
192.168.10.50

Dienst:
Port 443
```

Merksatz:

```
IP bringt zum Gerät.
Port bringt zum Dienst.
```

Warum braucht man Ports?

Ein Server kann viele Dienste gleichzeitig anbieten.

Beispiel:

Dienst	Port
SSH	22
DNS	53
HTTP	80
HTTPS	443
RDP	3389

Alle Dienste können auf derselben IP-Adresse laufen.

Die Portnummer entscheidet, welcher Dienst angesprochen wird.

Merksatz:

```
Mehrere Dienste auf einer IP-Adresse werden durch Ports unterschieden.
```

Portnummern

Portnummern sind 16 Bit groß.

Deshalb gibt es Portnummern von:

```
0
```

bis:

```
65535
```

Insgesamt sind das:

```
65536 mögliche Werte
```

Wichtig:

```
TCP und UDP haben jeweils eigene Portbereiche.
```

Das bedeutet:

```
TCP 53
und
UDP 53
```

sind technisch getrennte Kommunikationsmöglichkeiten.

Merksatz:

```
Portnummern gehen von 0 bis 65535.
```

TCP-Port und UDP-Port unterscheiden

Ein Port muss immer zusammen mit dem Transportprotokoll betrachtet werden.

Beispiel:

```
TCP 53
UDP 53
```

Beide können DNS betreffen, aber technisch sind sie nicht dasselbe.

Eine Firewall-Regel für TCP 53 erlaubt nicht automatisch UDP 53.

Eine Portweiterleitung für UDP 53 leitet nicht automatisch TCP 53 weiter.

Merksatz:

```
Portnummer immer mit TCP oder UDP nennen.
```

Portbereiche

Portnummern werden grob in drei Bereiche eingeteilt.

Bereich	Portnummern	Bedeutung
Well-Known Ports	0 – 1023	bekannte Standarddienste
Registered Ports	1024 – 49151	registrierte Anwendungsports
Dynamic / Ephemeral Ports	49152 – 65535	temporäre Client-Ports

Merksatz:

0–1023 bekannt,

1024–49151 registriert,

49152–65535 temporär.

Well-Known Ports

Well-Known Ports sind bekannte Standardports.

Sie werden für häufig genutzte Netzwerkdienste verwendet.

Beispiele:

Dienst	Protokoll	Port
FTP Steuerverbindung	TCP	21
SSH	TCP	22
Telnet	TCP	23
SMTP	TCP	25
DNS	TCP / UDP	53
DHCP Server	UDP	67
DHCP Client	UDP	68
HTTP	TCP	80
POP3	TCP	110
IMAP	TCP	143
NTP	UDP	123
HTTPS	TCP	443

Merksatz:

Well-Known Ports sind prüfungsrelevant.

Registered Ports

Registered Ports liegen im Bereich:

1024 bis 49151

Sie werden häufig von Anwendungen, Herstellern oder Diensten genutzt.

Beispiele:

Dienst / Anwendung	Protokoll	Port
MySQL / MariaDB	TCP	3306
RDP	TCP	3389
PostgreSQL	TCP	5432
VNC	TCP	5900
HTTP-Alternative	TCP	8080
HTTPS-Alternative	TCP	8443

Wichtig:

```
Nicht jeder Port in diesem Bereich ist automatisch sicher oder unsicher.
Entscheidend ist, welcher Dienst dort tatsächlich läuft.
```

Merksatz:

```
Registered Ports werden häufig von Anwendungen genutzt.
```

Dynamic oder Ephemeral Ports

Dynamic Ports werden auch Ephemeral Ports genannt.

Sie liegen häufig im Bereich:

```
49152 bis 65535
```

Clients nutzen diese Ports meistens als temporäre Quell-Ports.

Beispiel:

```
Client:
192.168.10.20:52344

Server:
93.184.216.34:443
```

Der Client nutzt Port 52344 nur vorübergehend für diese Verbindung.

Merksatz:

```
Ephemeral Port = temporärer Client-Port.
```

Quell-Port und Ziel-Port

Ein TCP- oder UDP-Header enthält:

```
Quell-Port
Ziel-Port
```

Beispiel HTTPS:

```
Client:
192.168.10.20:52344

Server:
93.184.216.34:443
```

Dabei ist:

```
52344 = Quell-Port des Clients
443 = Ziel-Port des Servers
```

Merksatz:

```
Ziel-Port zeigt den gewünschten Dienst.
Quell-Port hilft bei der Rückzuordnung.
```

Warum ist der Quell-Port wichtig?

Ein Client kann mehrere Verbindungen gleichzeitig öffnen.

Beispiel:

```
Browser-Tab 1 zu HTTPS
Browser-Tab 2 zu HTTPS
Update-Dienst zu HTTPS
Cloud-Sync zu HTTPS
```

Alle können zu Port 443 gehen.

Der Client nutzt unterschiedliche Quell-Ports, damit Antworten korrekt zugeordnet werden.

Merksatz:

```
Quell-Ports unterscheiden gleichzeitige Verbindungen eines Clients.
```

Socket

Ein Socket besteht aus:

```
IP-Adresse
und
Port
```

Beispiel:

```
192.168.10.20:52344
```

Bei TCP-Verbindungen ist die Verbindung eindeutig über:

```
Quell-IP
Quell-Port
Ziel-IP
Ziel-Port
Protokoll
```

Beispiel:

```
192.168.10.20:52344
→ 93.184.216.34:443
```

```
TCP
```

Merksatz:

```
Socket = IP-Adresse plus Port.
```

Dienst hört auf einem Port

Ein Dienst muss auf einem Port lauschen, damit Verbindungen oder Anfragen angenommen werden können.

Beispiel:

```
Webserver hört auf TCP 443.
```

Das bedeutet:

```
Der Dienst wartet auf HTTPS-Verbindungen.
```

Wenn kein Dienst auf einem Port lauscht, ist der Port aus Sicht des Clients geschlossen.

Merksatz:

```
Dienst muss auf dem Port lauschen.
```

LISTEN und ESTABLISHED

Bei TCP sind zwei Zustände besonders wichtig.

Zustand	Bedeutung
LISTEN	Dienst wartet auf eingehende Verbindungen
ESTABLISHED	Verbindung besteht

Beispiel:

```
Webserver im Zustand LISTEN auf TCP 443
```

bedeutet:

```
Der Webserver wartet auf HTTPS-Verbindungen.
```

Merksatz:

```
LISTEN = Dienst wartet.
ESTABLISHED = Verbindung steht.
```

Offener Port

Ein Port ist offen, wenn ein Dienst dort erreichbar ist.

Beispiel:

```
TCP 443 offen
```

Bedeutung:

```
Ein Dienst nimmt TCP-Verbindungen auf Port 443 an.
```

Aber:

```
Ein offener Port beweist nicht automatisch,
dass die Anwendung korrekt funktioniert.
```

Merksatz:

```
Offener Port = Dienst grundsätzlich erreichbar.
```

Geschlossener Port

Ein Port ist geschlossen, wenn kein Dienst auf diesem Port lauscht.

Beispiel:

```
Server ist per Ping erreichbar.
TCP 22 ist geschlossen.
```

Bedeutung:

```
Der Host ist erreichbar,
aber SSH läuft dort nicht oder nicht auf Port 22.
```

Merksatz:

```
Geschlossener Port heißt nicht:
Gerät ist offline.
```

Gefilterter Port

Ein Port wirkt gefiltert, wenn eine Firewall Pakete blockiert oder verwirft.

Typisch:

```
keine klare Antwort
Timeout
Verbindung bleibt hängen
```

Mögliche Ursachen:

```
- Firewall blockiert
- Paketfilter verwirft still
- NAT-Regel fehlt
- Rückweg fehlt
- Security-Gruppe blockiert
```

Merksatz:

```
Gefiltert bedeutet häufig:
Firewall oder Paketfilter im Weg.
```

Portstatus im Vergleich

Status	Bedeutung
offen	Dienst antwortet oder nimmt Verbindung an

Status	Bedeutung
geschlossen	Host erreichbar, aber kein Dienst auf diesem Port
gefiltert	keine klare Antwort, oft Firewall
offen, aber Anwendung fehlerhaft	Port erreichbar, Dienstproblem auf höherer Schicht

Merksatz:

> Portstatus hilft,
> Schicht 4 von höheren Schichten zu trennen.

Wichtige Standardports

Dienst	Protokoll	Port	Zweck
FTP	TCP	21	Dateiübertragung Steuerverbindung
SSH	TCP	22	sicherer Fernzugriff
Telnet	TCP	23	unsicherer Fernzugriff
SMTP	TCP	25	E-Mail-Versand zwischen Servern
DNS	TCP / UDP	53	Namensauflösung
DHCP Server	UDP	67	IP-Konfiguration anbieten
DHCP Client	UDP	68	IP-Konfiguration empfangen
HTTP	TCP	80	unverschlüsselte Webübertragung
POP3	TCP	110	E-Mail-Abruf
NTP	UDP	123	Zeitsynchronisation
IMAP	TCP	143	E-Mail-Abruf und Verwaltung
HTTPS	TCP	443	verschlüsselte Webübertragung
RDP	TCP	3389	Windows-Remote-Desktop

Merksatz:

> HTTP 80,
> HTTPS 443,
> SSH 22,

```
DNS 53,
DHCP 67/68.
```

FTP

FTP steht für:

```
File Transfer Protocol
```

Typischer Port:

```
TCP 21
```

FTP dient der Dateiübertragung.

Wichtig:

```
Klassisches FTP ist unverschlüsselt.
Benutzername, Passwort und Daten können ohne Schutz übertragen werden.
```

Sichere Alternativen sind zum Beispiel:

```
SFTP
FTPS
```

Merksatz:

```
FTP = TCP 21,
klassisch unverschlüsselt.
```

SSH

SSH steht für:

```
Secure Shell
```

Typischer Port:

```
TCP 22
```

SSH wird genutzt für:

```
- sicheren Fernzugriff
- Administration von Servern
- sichere Kommandozeile
- Tunnel
- Dateiübertragung über SFTP
```

Merksatz:

```
SSH = TCP 22,
verschlüsselter Fernzugriff.
```

Telnet

Telnet nutzt typischerweise:

```
TCP 23
```

Telnet ist ein älteres Fernzugriffsprotokoll.

Problem:

```
Telnet ist unverschlüsselt.
```

Deshalb sollte Telnet in modernen Netzwerken nicht für administrative Zugriffe verwendet werden.

Merksatz:

```
Telnet = TCP 23,
unsicher und unverschlüsselt.
```

SMTP

SMTP steht für:

```
Simple Mail Transfer Protocol
```

Typischer Port:

```
TCP 25
```

SMTP wird für den E-Mail-Transport verwendet, besonders zwischen Mailservern.

Weitere Mailports können sein:

```
587 für Mail Submission
465 für SMTPS
```

Für AP1/AP2 ist besonders wichtig:

```
SMTP = Mailversand
```

Merksatz:

```
SMTP = E-Mail-Versand.
```

POP3

POP3 steht für:

```
Post Office Protocol Version 3
```

Typischer Port:

```
TCP 110
```

POP3 dient dem Abruf von E-Mails.

Häufig werden Mails dabei vom Server heruntergeladen.

Verschlüsselte Variante:

```
POP3S über TCP 995
```

Merksatz:

```
POP3 = E-Mail-Abruf.
```

IMAP

IMAP steht für:

```
Internet Message Access Protocol
```

Typischer Port:

```
TCP 143
```

IMAP dient ebenfalls dem Abruf und der Verwaltung von E-Mails.

Im Unterschied zu POP3 bleiben Mails häufig auf dem Server und werden zwischen Geräten synchronisiert.

Verschlüsselte Variante:

```
IMAPS über TCP 993
```

Merksatz:

```
IMAP = E-Mail-Verwaltung auf dem Server.
```

SMTP, POP3 und IMAP unterscheiden

Protokoll	Richtung / Zweck	typischer Port
SMTP	E-Mail senden / transportieren	TCP 25
POP3	E-Mail abrufen	TCP 110
IMAP	E-Mail abrufen und verwalten	TCP 143

Merksatz:

```
SMTP sendet.
POP3 und IMAP holen ab.
```

DNS

DNS steht für:

```
Domain Name System
```

DNS ordnet Namen IP-Adressen zu.

Typischer Port:

```
UDP 53
```

DNS kann auch nutzen:

```
TCP 53
```

Beispiele für TCP bei DNS:

```
- große Antworten
- Zonentransfers
- bestimmte DNSSEC-Fälle
```

Merksatz:

```
DNS = UDP 53 häufig,
TCP 53 ebenfalls möglich.
```

DHCP

DHCP steht für:

```
Dynamic Host Configuration Protocol
```

DHCP vergibt automatisch IP-Konfigurationen.

Wichtige Ports:

Rolle	Protokoll	Port
DHCP-Server	UDP	67
DHCP-Client	UDP	68

DHCP vergibt zum Beispiel:

```
- IP-Adresse
- Subnetzmaske
- Gateway
- DNS-Server
```

Merksatz:

```
DHCP = UDP 67 und 68.
```

HTTP

HTTP steht für:

```
Hypertext Transfer Protocol
```

Typischer Port:

```
TCP 80
```

HTTP wird für Webseiten und Webdienste verwendet.

Wichtig:

```
HTTP ist unverschlüsselt.
```

Heute wird für Webseiten meist HTTPS bevorzugt.

Merksatz:

```
HTTP = TCP 80,
unverschlüsselt.
```

HTTPS

HTTPS steht für:

```
Hypertext Transfer Protocol Secure
```

Typischer Port:

```
TCP 443
```

HTTPS nutzt Verschlüsselung über TLS.

HTTPS schützt unter anderem:

```
- Vertraulichkeit
- Integrität
- Serverauthentizität über Zertifikate
```

Merksatz:

```
HTTPS = TCP 443,
HTTP mit TLS-Verschlüsselung.
```

NTP

NTP steht für:

```
Network Time Protocol
```

Typischer Port:

```
UDP 123
```

NTP dient zur Zeitsynchronisation.

Warum wichtig?

Viele IT-Funktionen benötigen korrekte Zeit.

Beispiele:

- Zertifikate
- Kerberos
- Logs
- Monitoring
- Dateisynchronisation

Merksatz:

NTP = UDP 123,
Zeitsynchronisation.

RDP

RDP steht für:

Remote Desktop Protocol

Typischer Port:

TCP 3389

RDP wird häufig für grafischen Fernzugriff auf Windows-Systeme genutzt.

Wichtig:

RDP sollte nicht ungeschützt direkt ins Internet geöffnet werden.

Besser:

VPN
Zugriffsbeschränkung

Multi-Faktor-Authentifizierung
sichere Gateway-Lösungen

Merksatz:

RDP = TCP 3389,
Fernzugriff sorgfältig absichern.

Sichere und unsichere Protokolle

Einige ältere Protokolle übertragen Daten unverschlüsselt.

eher unsicher / unverschlüsselt	sicherere Alternative
Telnet	SSH
FTP	SFTP oder FTPS
HTTP	HTTPS
POP3	POP3S
IMAP	IMAPS
SMTP ohne TLS	SMTP mit STARTTLS oder SMTPS

Merksatz:

Für Administration und sensible Daten verschlüsselte Protokolle nutzen.

Ports und Firewall-Regeln

Firewalls arbeiten häufig mit IP-Adressen und Ports.

Beispiel-Regel:

Erlaube TCP 443
von Client-Netz
zum Internet

Bedeutung:

HTTPS-Verbindungen nach außen sind erlaubt.

Andere Ports können weiterhin blockiert sein.

Merksatz:

```
Firewall-Regel = Quelle, Ziel, Protokoll, Port, Aktion.
```

Firewall-Regel vollständig denken

Eine sinnvolle Firewall-Regel enthält:

```
- Quelle
- Ziel
- Protokoll
- Port
- Richtung
- Aktion
- Kommentar oder Zweck
```

Beispiel:

```
Quelle:
Mitarbeiter-VLAN

Ziel:
Internet

Protokoll:
TCP

Port:
443

Aktion:
erlauben
```

Merksatz:

```
Nicht nur Port nennen,
sondern Quelle, Ziel und Protokoll.
```

TCP und UDP bei Firewall-Regeln

Ports müssen bei Firewalls mit TCP oder UDP angegeben werden.

Beispiel DNS:

```
UDP 53 erlauben
TCP 53 erlauben, wenn benötigt
```

Beispiel HTTPS:

```
TCP 443 erlauben
```

Beispiel HTTP/3:

```
UDP 443 kann zusätzlich relevant sein
```

Prüfungsfalle:

```
TCP 53 erlaubt nicht automatisch UDP 53.
```

Merksatz:

```
Firewall-Regeln brauchen Protokoll und Port.
```

Ports und Portweiterleitung

Bei Portweiterleitungen muss ebenfalls das richtige Protokoll gewählt werden.

Beispiel Webserver:

```
TCP 80
TCP 443
```

Beispiel DNS-Server:

```
UDP 53
TCP 53
```

Beispiel Spielserver:

```
je nach Anwendung TCP oder UDP
```

Merksatz:

```
Portweiterleitung muss TCP oder UDP korrekt treffen.
```

Portweiterleitung und Dienst

Eine Portweiterleitung funktioniert nur, wenn mehrere Dinge passen.

Prüfpunkte:

```
- öffentliche Erreichbarkeit vorhanden?
- kein CGNAT?
- externer Port korrekt?
- internes Ziel korrekt?
- internes Ziel hört auf dem Port?
- Firewall erlaubt?
- Dienst läuft?
- DNS zeigt auf richtige IP?
```

Merksatz:

```
Portweiterleitung reicht nicht,
wenn der Dienst nicht läuft oder blockiert wird.
```

Ports und NAT/PAT

PAT nutzt Port-Zuordnungen.

Beispiel:

Intern	Extern
192.168.10.20:52344	93.184.100.10:40001
192.168.10.30:52344	93.184.100.10:40002

Dadurch können mehrere interne Geräte dieselbe öffentliche IP-Adresse nutzen.

Merksatz:

```
PAT unterscheidet Verbindungen über Ports.
```

Ports und Dienste prüfen

Typische Fragen bei der Prüfung eines Dienstes:

```
Ist der Host per IP erreichbar?
Ist der richtige Port offen?
Nutzt der Dienst TCP oder UDP?
Läuft der Dienst wirklich?
Hört der Dienst auf der richtigen Schnittstelle?
Blockiert eine Firewall?
Gibt es NAT oder Portweiterleitung?
Ist die Anwendung korrekt konfiguriert?
```

Merksatz:

```
Dienstproblem = IP, Port, Protokoll, Firewall und Anwendung prüfen.
```

Ping und Porttest unterscheiden

Ping nutzt ICMP.

Ping prüft keine TCP- oder UDP-Ports.

Beispiel:

```
ping server funktioniert
```

Das bedeutet nicht automatisch:

```
TCP 443 ist offen.
```

Um Ports zu prüfen, braucht man andere Tests.

Beispiele sinngemäß:

```
Verbindung zu TCP-Port testen
Dienstabfrage durchführen
Portscan kontrolliert einsetzen
Logs prüfen
```

Merksatz:

```
Ping prüft IP.
Porttest prüft Schicht 4.
```

Portscan

Ein Portscan prüft, welche Ports auf einem Ziel erreichbar sind.

Er kann nützlich sein für:

```
- Administration
- Fehlersuche
- Sicherheitsprüfung
- Inventarisierung
```

Aber:

```
Portscans dürfen nur in erlaubten Netzen durchgeführt werden.
```

Unberechtigte Scans können als Angriff gewertet werden.

Merksatz:

```
Portscan nur mit Berechtigung durchführen.
```

Port offen heißt nicht automatisch sicher

Ein offener Port bedeutet:

```
Dort ist ein Dienst erreichbar.
```

Das sagt noch nicht:

```
ob der Dienst sicher konfiguriert ist
ob er aktuell ist
ob die Authentifizierung stark ist
ob Schwachstellen vorhanden sind
```

Beispiel:

```
SSH offen kann normal sein.
SSH mit schwachem Passwort ist gefährlich.
```

Merksatz:

```
Sicherheit hängt nicht nur vom Port,
sondern vom Dienst ab.
```

Port geschlossen heißt nicht automatisch geschützt

Ein geschlossener Port bedeutet:

```
Auf diesem Port lauscht aktuell kein Dienst.
```

Aber:

```
andere Ports können offen sein.
Dienste können später gestartet werden.
Firewall-Regeln können sich ändern.
interne Angriffe können andere Wege nutzen.
```

Merksatz:

Geschlossene Ports sind gut,
ersetzen aber kein Sicherheitskonzept.

Prinzip: Nur notwendige Ports öffnen

Ein wichtiges Sicherheitsprinzip lautet:

Nur notwendige Dienste und Ports freigeben.

Das reduziert die Angriffsfläche.

Beispiele:

SSH nur für Admin-Netz erlauben.
Datenbank nicht direkt ins Internet öffnen.
RDP nur über VPN bereitstellen.
Webserver nur mit benötigten Ports freigeben.
Gastnetz von internen Diensten trennen.

Merksatz:

Weniger offene Ports = kleinere Angriffsfläche.

Typische Port-Fehler

Typische Fehler sind:

- falscher Port
- falsches Protokoll TCP statt UDP
- falsches Protokoll UDP statt TCP
- Dienst läuft nicht
- Dienst hört nur auf localhost
- Firewall blockiert
- NAT-Regel fehlt
- Portweiterleitung falsch
- DNS zeigt auf falschen Server
- Reverse Proxy nutzt falschen Zielport

```
- Client nutzt alten Port
```

Merksatz:

```
Portprobleme sind oft Konfigurationsfehler zwischen Dienst, Firewall und NAT.
```

Beispiel: HTTPS funktioniert nicht

Fehlerbild:

```
Server ist per Ping erreichbar.
Webseite öffnet nicht über HTTPS.
```

Mögliche Ursachen:

```
- TCP 443 geschlossen
- Firewall blockiert TCP 443
- Webserver läuft nicht
- Webserver hört auf anderem Port
- TLS-Zertifikat fehlerhaft
- Reverse Proxy falsch
- DNS zeigt auf falsche IP
- NAT-Portweiterleitung falsch
```

Merksatz:

```
Ping geht,
HTTPS nicht:
TCP 443 und Anwendung prüfen.
```

Beispiel: DNS funktioniert nicht

Fehlerbild:

```
Internet-IP ist erreichbar.
Namen werden nicht aufgelöst.
```

Mögliche Ursachen:

```
- DNS-Server falsch
- UDP 53 blockiert
- TCP 53 blockiert bei großen Antworten
- DNS-Dienst läuft nicht
- Firewall blockiert
- falsche DNS-Zone
- DNS-Cache fehlerhaft
```

Merksatz:

```
DNS braucht Namensdienst und passende Ports.
```

Beispiel: DHCP funktioniert nicht

Fehlerbild:

```
Client bekommt keine IP-Adresse.
```

Mögliche Ursachen:

```
- DHCP-Server nicht erreichbar
- UDP 67/68 blockiert
- falsches VLAN
- DHCP-Relay fehlt
- DHCP-Bereich erschöpft
- DHCP-Dienst läuft nicht
- Trunk transportiert VLAN nicht
```

Merksatz:

```
DHCP-Probleme können Schicht 2, 3, 4 und 7 berühren.
```

Beispiel: SSH funktioniert nicht

Fehlerbild:

Server ist erreichbar,
aber SSH-Verbindung klappt nicht.

Mögliche Ursachen:

- TCP 22 geschlossen
- SSH-Dienst läuft nicht
- Firewall blockiert
- SSH hört auf anderem Port
- Zugriff nur aus bestimmten Netzen erlaubt
- Benutzer oder Schlüssel falsch
- Fail2ban oder Schutzsystem blockiert

Merksatz:

SSH braucht TCP-Port,
laufenden Dienst
und gültige Anmeldung.

Einordnung in das OSI-Modell

Thema	Schicht
IP-Adresse	3
Routing	3
TCP	4
UDP	4
Port	4
Socket	4 mit IP-Bezug
DNS-Dienst	7, nutzt TCP/UDP 53
DHCP-Dienst	7, nutzt UDP 67/68
HTTP	7, nutzt TCP 80
HTTPS	7, nutzt TCP 443
SSH	7, nutzt TCP 22
Firewall nach Port	3 / 4
NAT/PAT	3 / 4

Merksatz:

```
Port = Schicht 4.
Dienst = höhere Schicht.
```

Was Ports nicht leisten

Ports lösen nicht automatisch:

```
- Namensauflösung
- Authentifizierung
- Verschlüsselung
- Benutzerrechte
- Dienstkonfiguration
- Zertifikatsprüfung
- Anwendungslogik
- Routing
- VLAN-Zuordnung
```

Ports sagen nur:

```
welcher Dienst angesprochen werden soll.
```

Merksatz:

```
Port erreichbar heißt nicht:
Anwendung korrekt.
```

Typische IHK-Fragen

In AP1 und AP2 kann zum Beispiel gefragt werden:

```
- Was ist ein Port?
- Warum braucht man Ports?
- Was ist der Unterschied zwischen IP-Adresse und Port?
- Was ist der Unterschied zwischen Switch-Port und TCP-Port?
- In welchem Bereich liegen Portnummern?
```

- Was sind Well-Known Ports?
- Was sind Ephemeral Ports?
- Welche Ports nutzen HTTP, HTTPS, SSH, DNS und DHCP?
- Warum muss man bei Portfreigaben TCP und UDP unterscheiden?
- Was bedeutet offener, geschlossener oder gefilterter Port?
- Warum kann Ping funktionieren, aber ein Dienst nicht?
- Warum sollte man nur notwendige Ports öffnen?

Typische Prüfungsfallen

Port gehört zu Schicht 4.

Dienst gehört zu höheren Schichten.

IP-Adresse zeigt zum Gerät.

Port zeigt zum Dienst.

Switch-Port ist nicht TCP-Port.

TCP-Port und UDP-Port sind getrennt.

Portnummern gehen von 0 bis 65535.

Well-Known Ports = 0 bis 1023.

Ephemeral Ports = temporäre Client-Ports.

HTTP = TCP 80.

HTTPS = TCP 443.

SSH = TCP 22.

DNS = UDP 53 und TCP 53.

DHCP = UDP 67 und 68.

NTP = UDP 123.

RDP = TCP 3389.

Ping nutzt ICMP und prüft keine Ports.

Firewall-Regeln brauchen Protokoll und Port.

Portweiterleitung braucht richtiges Protokoll.

Offener Port bedeutet nicht automatisch sichere Anwendung.

Wichtige Begriffe kurz erklärt

Begriff	Kurze Erklärung
Port	logische Dienstnummer auf Schicht 4
TCP-Port	Port für TCP-Kommunikation
UDP-Port	Port für UDP-Kommunikation
Switch-Port	physischer Anschluss am Switch
Quell-Port	Port des Absenders
Ziel-Port	Port des gewünschten Dienstes
Socket	IP-Adresse plus Port
Well-Known Port	Port 0 bis 1023
Registered Port	Port 1024 bis 49151
Ephemeral Port	temporärer Client-Port
LISTEN	Dienst wartet auf Verbindung
ESTABLISHED	Verbindung besteht
offener Port	Dienst erreichbar
geschlossener Port	kein Dienst lauscht
gefilterter Port	Firewall oder Filter blockiert
Portweiterleitung	externer Port wird intern weitergeleitet
Portscan	Prüfung erreichbarer Ports

IHK-sichere Kurzformulierung

Ports sind logische Dienstnummern auf OSI-Schicht 4. Sie werden zusammen mit TCP oder UDP verwendet, um Daten auf einem Gerät dem richtigen Dienst zuzuordnen. Eine IP-Adresse

identifiziert das Gerät, der Port identifiziert den Dienst auf diesem Gerät. Portnummern reichen von 0 bis 65535 und werden in Well-Known Ports, Registered Ports und Dynamic beziehungsweise Ephemeral Ports eingeteilt. TCP- und UDP-Ports sind getrennt zu betrachten. Firewall-Regeln und Portweiterleitungen müssen deshalb immer das richtige Protokoll und den richtigen Port berücksichtigen.

Merksätze

Port = logische Dienstnummer.

Port gehört zu Schicht 4.

IP bringt zum Gerät.

Port bringt zum Dienst.

Switch-Port ist physisch.

TCP-/UDP-Port ist logisch.

TCP-Port und UDP-Port sind nicht dasselbe.

Portnummern gehen von 0 bis 65535.

Well-Known Ports = 0 bis 1023.

Registered Ports = 1024 bis 49151.

Ephemeral Ports = 49152 bis 65535.

Ziel-Port zeigt den Dienst.

Quell-Port hilft bei der Rückzuordnung.

Socket = IP-Adresse plus Port.

LISTEN = Dienst wartet.

ESTABLISHED = Verbindung steht.

HTTP = TCP 80.

HTTPS = TCP 443.

SSH = TCP 22.

DNS = UDP 53 und TCP 53.

DHCP = UDP 67 und 68.

NTP = UDP 123.

RDP = TCP 3389.

Ping prüft keine Ports.

Firewall-Regel braucht Quelle, Ziel, Protokoll und Port.

Portweiterleitung braucht richtiges Protokoll.

Nur notwendige Ports öffnen.

Offener Port heißt nicht automatisch sicher.

Port erreichbar heißt nicht automatisch:
Anwendung funktioniert.