

8.1 OSI-Schicht 5 - Sitzungsschicht

Die Sitzungsschicht ist Schicht 5 des OSI-Modells.

Sie liegt zwischen:

Schicht 4: Transportschicht
und
Schicht 6: Darstellungsschicht

Die Hauptaufgabe von Schicht 5 ist:

Sitzungen zwischen Kommunikationspartnern aufbauen, verwalten und beenden.

Dabei geht es vor allem um:

- Sitzungen
- Dialogsteuerung
- Verbindungsverwaltung auf Sitzungsebene
- Wiederaufnahme von Sitzungen
- Synchronisationspunkte
- geordnete Kommunikation zwischen Anwendungen

Merksatz:

Schicht 5 = Sitzungen zwischen Anwendungen verwalten.

Grundidee von Schicht 5

Schicht 4 transportiert Daten mit TCP oder UDP.

Schicht 5 organisiert die logische Sitzung zwischen Anwendungen.

Eine Sitzung ist dabei eine zusammenhängende Kommunikationsbeziehung.

Beispiel:

Benutzer meldet sich an einem Dienst an.

Der Dienst merkt sich:

Dieser Benutzer gehört zu dieser Sitzung.

Solange die Sitzung besteht, kann der Benutzer weiterarbeiten.

Merksatz:

Schicht 4 transportiert.

Schicht 5 verwaltet die Sitzung.

Was ist eine Sitzung?

Eine Sitzung ist ein zusammenhängender Kommunikationsabschnitt zwischen zwei Systemen oder Anwendungen.

Beispiele:

- Benutzer ist an einer Webanwendung angemeldet
- Remote-Verbindung ist aktiv
- Dateiübertragung läuft
- Datenbankverbindung besteht
- Videokonferenz ist verbunden
- Anwendung merkt sich den Zustand eines Benutzers

Merksatz:

Sitzung = zusammenhängende Kommunikationsbeziehung.

Warum braucht man Sitzungen?

Viele Anwendungen müssen wissen:

Wer kommuniziert gerade?

Ist der Benutzer angemeldet?

Welche Aktion gehört zu welcher Verbindung?

Welche Daten gehören zusammen?

Wo wurde eine Übertragung unterbrochen?

Kann eine Sitzung wieder aufgenommen werden?

Ohne Sitzungsverwaltung müsste jede Anfrage komplett unabhängig betrachtet werden.

Merksatz:

Sitzungen helfen,
zusammengehörige Kommunikation zu verwalten.

Einordnung im OSI-Modell

OSI-Schicht	Name	Aufgabe
7	Anwendungsschicht	Netzwerkdienste für Anwendungen
6	Darstellungsschicht	Darstellung, Codierung, Verschlüsselung
5	Sitzungsschicht	Sitzungen aufbauen, verwalten und beenden
4	Transportschicht	TCP, UDP, Ports
3	Vermittlungsschicht / Netzwerkschicht	IP, Routing
2	Sicherungsschicht	Frames, MAC, VLAN
1	Bitübertragungsschicht	Bits, Signale

Merksatz:

Schicht 5 liegt über TCP/UDP
und unter Darstellung und Anwendung.

Sitzungsschicht im Prüfungsverständnis

Die Sitzungsschicht ist in modernen Netzwerken nicht immer klar als eigenes Protokoll sichtbar.

Viele Aufgaben der Sitzungsschicht werden heute durch Anwendungen, Bibliotheken oder Protokolle höherer Schichten übernommen.

Trotzdem ist die Grundidee wichtig:

Sitzungen starten,
Sitzungen steuern,

Sitzungen aufrechterhalten,
Sitzungen beenden.

Merksatz:

Schicht 5 ist im Alltag oft weniger sichtbar,
aber für das OSI-Verständnis wichtig.

Schicht 4 und Schicht 5 unterscheiden

Thema	Schicht
TCP-Verbindung	4
UDP-Datagramm	4
Portnummer	4
Verbindungsaufbau mit SYN, SYN-ACK, ACK	4
Benutzer-Sitzung	5 / 7-Bezug
Session-ID	5 / 7-Bezug
Anmeldung bleibt erhalten	5 / 7-Bezug
Sitzungswiederaufnahme	5 / 6 / 7-Bezug

Merksatz:

TCP-Verbindung ist nicht automatisch dasselbe wie Benutzersitzung.

TCP-Verbindung ist nicht gleich Sitzung

Eine TCP-Verbindung ist eine Transportverbindung auf Schicht 4.

Eine Sitzung ist eine logische Kommunikationsbeziehung auf Schicht 5 oder höher.

Beispiel Webanwendung:

TCP-Verbindung wird aufgebaut.
HTTPS-Anfrage wird gesendet.
Benutzer bleibt durch Session-Cookie angemeldet.

Die Benutzersitzung kann länger bestehen als eine einzelne TCP-Verbindung.

Merksatz:

TCP verbindet technisch.
Sitzung hält den logischen Zusammenhang.

Beispiel: Webanmeldung

Ein Benutzer meldet sich an einer Webseite an.

Ablauf vereinfacht:

1. Benutzer öffnet Webseite.
2. Verbindung wird aufgebaut.
3. Benutzer gibt Zugangsdaten ein.
4. Server prüft Anmeldung.
5. Server erstellt eine Sitzung.
6. Browser erhält eine Session-ID oder ein Token.
7. Weitere Anfragen werden der Sitzung zugeordnet.

Dadurch bleibt der Benutzer angemeldet.

Merksatz:

Websitzungen sorgen dafür,
dass ein Benutzer wiedererkannt wird.

Session-ID

Eine Session-ID ist eine Kennung für eine Sitzung.

Sie hilft dem Server, Anfragen einer bestimmten Sitzung zuzuordnen.

Beispiel:

Anfrage 1:
Benutzer meldet sich an.

Anfrage 2:
Benutzer öffnet eine geschützte Seite.

Der Server erkennt anhand der Session-ID:

Diese Anfrage gehört zur angemeldeten Sitzung.

Merksatz:

Session-ID = Kennung einer Sitzung.

Session-Cookie

Bei Webanwendungen wird eine Session-ID häufig in einem Cookie gespeichert.

Der Browser sendet dieses Cookie bei weiteren Anfragen mit.

Dadurch erkennt der Server die Sitzung wieder.

Wichtig:

Cookies gehören fachlich eher zur Anwendungsschicht,
erfüllen aber eine typische Sitzungsfunktion.

Merksatz:

Session-Cookie hilft,
eine Websitzung wiederzuerkennen.

Token

Ein Token kann ebenfalls zur Sitzungs- oder Zugriffserkennung genutzt werden.

Beispiele:

- Session-Token
- Access Token
- Refresh Token
- API-Token

Ein Token kann dem Server zeigen:

Wer fragt an?

Welche Rechte hat diese Anfrage?

Ist die Sitzung noch gültig?

Merksatz:

Token können Sitzungs- und Zugriffsinformationen tragen.

Sitzung aufbauen

Eine Sitzung wird aufgebaut, wenn zwei Kommunikationspartner eine zusammenhängende Kommunikation beginnen.

Beispiele:

- Benutzer meldet sich an
- Remote-Sitzung wird gestartet
- Datenbankclient verbindet sich
- Dateiübertragung beginnt
- Videokonferenz wird betreten

Merksatz:

Sitzungsaufbau = Beginn einer logischen Kommunikation.

Sitzung verwalten

Während einer Sitzung müssen Zustände verwaltet werden.

Beispiele:

- Benutzer ist angemeldet
- Warenkorb enthält Artikel
- Dateiübertragung ist bei 60 Prozent
- Remote-Desktop-Sitzung ist aktiv
- Datenbanktransaktion läuft
- Videokonferenzteilnehmer sind verbunden

Merksatz:

Sitzungsverwaltung merkt sich den Zustand einer Kommunikation.

Sitzung beenden

Eine Sitzung kann geordnet beendet werden.

Beispiele:

- Benutzer meldet sich ab
- Remote-Verbindung wird geschlossen
- Dateiübertragung ist abgeschlossen
- Anwendung beendet Verbindung
- Sitzung läuft wegen Inaktivität ab

Merksatz:

Sitzungsende = Kommunikation wird logisch abgeschlossen.

Timeout

Ein Timeout beendet eine Sitzung nach einer bestimmten Zeit ohne Aktivität.

Beispiel:

Benutzer ist 30 Minuten inaktiv.
Webanwendung meldet den Benutzer automatisch ab.

Warum ist das sinnvoll?

- Sicherheit
- Ressourcenschonung
- Schutz vor vergessenen Anmeldungen

Merksatz:

Timeout beendet inaktive Sitzungen automatisch.

Sitzungswiederaufnahme

Manche Systeme können eine Sitzung nach einer Unterbrechung wieder aufnehmen.

Beispiele:

- Remote-Desktop-Sitzung reconnect
- VPN-Verbindung stellt Sitzung wieder her
- Dateiübertragung wird fortgesetzt
- TLS-Sitzung wird wiederverwendet
- Webanwendung nutzt gespeichertes Token

Merksatz:

Sitzungswiederaufnahme spart Zeit und erhält Zustand.

Synchronisationspunkte

Synchronisationspunkte helfen, eine Übertragung an bestimmten Stellen abzusichern.

Wenn eine Unterbrechung passiert, muss nicht immer von vorne begonnen werden.

Beispiel:

Eine große Datei wird übertragen.
Nach einer Unterbrechung wird ab einem bestimmten Punkt fortgesetzt.

Im OSI-Modell wird diese Idee der Sitzungsschicht zugeordnet.

Merksatz:

Synchronisationspunkte helfen bei Wiederaufnahme nach Unterbrechung.

Dialogsteuerung

Dialogsteuerung bedeutet:

Wer darf wann senden?

Wie wird die Kommunikation koordiniert?

Gibt es einen geordneten Ablauf?

Beispiele:

- Anfrage und Antwort
- abwechselnde Kommunikation
- gleichzeitiger Austausch
- Steuerung einer Remote-Sitzung

Merksatz:

Dialogsteuerung organisiert den Ablauf einer Kommunikation.

Halbduplex und Vollduplex als Denkmodell

Bei der Dialogsteuerung kann man sich vereinfacht vorstellen:

Halbduplex:

Kommunikation abwechselnd

Vollduplex:

Kommunikation gleichzeitig in beide Richtungen

Diese Begriffe wurden schon bei unteren Schichten verwendet, können aber auch als Denkmodell für Dialogabläufe helfen.

Merksatz:

Dialogsteuerung regelt,
wie Kommunikationspartner miteinander sprechen.

Sitzungen bei Remote Desktop

Bei Remote Desktop gibt es eine Benutzersitzung.

Beispiel:

Benutzer verbindet sich per RDP.
Desktop-Sitzung wird gestartet.
Programme laufen innerhalb dieser Sitzung.
Verbindung kann getrennt werden.
Sitzung kann später wieder verbunden werden.

Wichtig:

Die Benutzersitzung kann weiter bestehen,
obwohl die Netzwerkverbindung kurz getrennt wurde.

Merksatz:

Remote-Sitzung kann länger leben als die einzelne Verbindung.

Sitzungen bei SSH

Bei SSH gibt es ebenfalls eine Sitzung.

Beispiele:

- Shell-Sitzung
- Kommandoausführung
- SFTP-Sitzung
- Tunnel

Wenn die SSH-Verbindung beendet wird, endet normalerweise auch die Sitzung.

Mit zusätzlichen Werkzeugen können Sitzungen erhalten bleiben.

Beispiele:

- tmux
- screen

Merksatz:

SSH-Verbindung und Arbeitssitzung hängen oft eng zusammen.

Sitzungen bei Datenbanken

Datenbankverbindungen nutzen Sitzungszustände.

Beispiele:

- angemeldeter Benutzer
- aktive Transaktion
- temporäre Tabellen
- gesetzte Rollen
- aktuelle Datenbank
- Session-Variablen

Wenn die Sitzung endet, gehen manche sitzungsbezogenen Zustände verloren.

Merksatz:

Datenbanksitzungen speichern Kontext für die Verbindung.

Sitzungen bei Webanwendungen

Webanwendungen nutzen Sitzungen, obwohl HTTP ursprünglich zustandslos ist.

Zustandslos bedeutet:

Jede Anfrage ist zunächst unabhängig.

Damit ein Benutzer angemeldet bleiben kann, nutzt man zum Beispiel:

- Cookies
- Session-IDs
- Tokens
- serverseitige Sitzungsdaten

Merksatz:

Websitzungen machen zustandslose Anfragen zusammenhängend.

HTTP ist zustandslos

HTTP behandelt einzelne Anfragen grundsätzlich unabhängig.

Beispiel:

Anfrage 1:
Login-Seite öffnen

Anfrage 2:
Passwort senden

Anfrage 3:
Dashboard öffnen

Ohne Sitzungsmechanismus wüsste der Server bei Anfrage 3 nicht automatisch, dass der Benutzer bereits angemeldet ist.

Merksatz:

HTTP ist zustandslos,
Sitzungen schaffen Zusammenhang.

Session-State

Session-State bedeutet:

Zustand einer Sitzung

Beispiele:

- Benutzer-ID
- Rollen und Rechte
- Warenkorb
- Spracheinstellung
- Login-Status
- letzter Zugriff
- CSRF-Token

Session-State kann serverseitig oder clientseitig unterstützt werden.

Merksatz:

Session-State = gespeicherter Zustand einer Sitzung.

Sitzung und Authentifizierung

Sitzungen hängen oft mit Authentifizierung zusammen.

Authentifizierung bedeutet:

Wer bist du?

Nach erfolgreicher Anmeldung wird häufig eine Sitzung erstellt.

Danach muss der Benutzer nicht bei jeder Anfrage Benutzername und Passwort neu senden.

Merksatz:

Anmeldung startet oft eine Sitzung.

Sitzung und Autorisierung

Autorisierung bedeutet:

Was darfst du?

Eine Sitzung kann Informationen enthalten oder referenzieren wie:

- Benutzerrolle
- Berechtigungen
- Gruppenmitgliedschaft
- Zugriffsebene

Beispiel:

Benutzer ist angemeldet,
darf aber nicht in den Adminbereich.

Merksatz:

Authentifizierung prüft Identität.
Autorisierung prüft Rechte.

Sitzung und Sicherheit

Sitzungen müssen geschützt werden.

Wichtige Risiken:

- Session-Hijacking
- Session-Fixation
- gestohlene Cookies
- zu lange Sitzungsdauer
- fehlender Logout
- unsichere Tokens
- fehlende HTTPS-Verschlüsselung

Merksatz:

Wer eine Sitzung übernimmt,
kann oft wie der Benutzer handeln.

Session-Hijacking

Session-Hijacking bedeutet:

Ein Angreifer übernimmt eine gültige Sitzung.

Mögliche Ursachen:

- gestohlenes Session-Cookie
- unsichere Verbindung

- Schadsoftware
- XSS-Angriff
- unsichere Token-Speicherung

Schutzmaßnahmen:

- HTTPS verwenden
- sichere Cookie-Attribute
- kurze Lebensdauer sensibler Sitzungen
- erneute Anmeldung bei kritischen Aktionen
- Logout ermöglichen
- Schutz vor XSS

Merksatz:

Session-Hijacking = Übernahme einer gültigen Sitzung.

Session-Fixation

Session-Fixation bedeutet:

Ein Angreifer bringt einen Benutzer dazu,
eine vorher bekannte Session-ID zu verwenden.

Wenn sich der Benutzer dann anmeldet, kennt der Angreifer die Sitzung bereits.

Schutzmaßnahme:

Session-ID nach erfolgreichem Login erneuern.

Merksatz:

Nach Login sollte eine neue Session-ID vergeben werden.

Sichere Cookies

Bei Websitzungen sind sichere Cookie-Einstellungen wichtig.

Beispiele:

Secure:

Cookie nur über HTTPS senden.

HttpOnly:

JavaScript soll Cookie nicht auslesen können.

SameSite:

Schutz gegen bestimmte Cross-Site-Angriffe.

Merksatz:

Sichere Cookies schützen Websitzungen.

Logout

Ein Logout beendet eine Sitzung bewusst.

Dabei sollte die Sitzung auf dem Server ungültig gemacht werden.

Nur das Browserfenster zu schließen reicht nicht immer.

Wichtig:

Nach Logout sollte die Session-ID nicht mehr gültig sein.

Merksatz:

Logout sollte die Sitzung serverseitig beenden.

Session-Timeout und Sicherheit

Ein Session-Timeout reduziert Risiken durch vergessene Sitzungen.

Beispiel:

Benutzer bleibt an einem öffentlichen PC angemeldet.

Nach Inaktivität wird die Sitzung automatisch beendet.

Für kritische Systeme sind kürzere Timeouts sinnvoll.

Merksatz:

Timeout schützt vor dauerhaft offenen Sitzungen.

Keepalive

Keepalive-Mechanismen können Verbindungen oder Sitzungen aktiv halten.

Beispiele:

- Anwendung sendet regelmäßig kleine Nachrichten
- Server prüft, ob Client noch verbunden ist
- Verbindung wird nicht wegen Inaktivität geschlossen

Wichtig:

Keepalive kann sinnvoll sein,
aber auch Ressourcen verbrauchen.

Merksatz:

Keepalive hält Kommunikation aktiv.

Sitzung und TLS

TLS gehört eher zu Schicht 6 beziehungsweise zwischen Transport und Anwendung.

Trotzdem gibt es bei TLS Sitzungsmechanismen.

Beispiele:

- TLS Session Resumption
- Wiederverwendung von Schlüsselinformationen
- schnellerer erneuter Verbindungsaufbau

Merksatz:

TLS kann Sitzungen wieder aufnehmen,
gehört aber nicht rein zu Schicht 5.

Warum Schicht 5 oft schwer abzugrenzen ist

In der Praxis werden die OSI-Schichten 5, 6 und 7 oft gemeinsam betrachtet.

Viele moderne Protokolle und Anwendungen übernehmen mehrere Aufgaben gleichzeitig.

Beispiel HTTPS:

TCP:
Transport auf Schicht 4

TLS:
Verschlüsselung und Darstellungsschutz

HTTP:
Anwendung

Cookies / Tokens:
Sitzungsverwaltung

Merksatz:

Schicht 5 ist in modernen Protokollen oft in Anwendungen integriert.

Typische Protokolle mit Sitzungsbezug

Beispiel	Sitzungsbezug
NetBIOS Session Service	klassische Sitzungskommunikation
RPC	entfernte Funktionsaufrufe mit Kommunikationskontext
SMB	Datei- und Freigabesitzungen
RDP	Remote-Desktop-Sitzung
SSH	Shell- oder SFTP-Sitzung
Datenbankverbindung	angemeldeter Datenbankkontext
Webanwendung	Session-ID, Cookie oder Token

Merksatz:

Sitzungsfunktionen können in vielen Protokollen stecken.

SMB und Sitzungen

SMB wird für Datei- und Druckerfreigaben verwendet.

Dabei gibt es sitzungsähnliche Zustände.

Beispiele:

- Benutzer authentifiziert sich
- Zugriff auf Freigabe wird aufgebaut
- Dateien werden geöffnet
- Berechtigungen werden geprüft
- Sitzung wird beendet

Merksatz:

Dateifreigaben brauchen Sitzungs- und Zugriffskontext.

RPC und Sitzungen

RPC steht für:

Remote Procedure Call

Damit kann ein Programm Funktionen auf einem entfernten System aufrufen.

Dabei kann ein Kommunikationskontext entstehen.

Beispiele:

- Windows-Dienste
- Verwaltungsfunktionen
- verteilte Anwendungen

Merksatz:

RPC ermöglicht entfernte Funktionsaufrufe mit Kommunikationskontext.

Schicht 5 und Fehlersuche

Typische Fragen bei Sitzungsproblemen:

Wird die Sitzung aufgebaut?
Bleibt die Sitzung bestehen?
Wird der Benutzer abgemeldet?
Läuft die Sitzung wegen Timeout ab?
Wird die Session-ID korrekt übertragen?
Werden Cookies blockiert?
Wird ein Token ungültig?
Geht die Sitzung nach Verbindungswechsel verloren?
Gibt es Load-Balancer-Probleme?
Wird die Sitzung auf dem richtigen Server gespeichert?

Merksatz:

Sitzungsprobleme zeigen sich oft als unerwartete Abmeldung oder verlorener Zustand.

Fehlerbild: Benutzer wird ständig abgemeldet

Mögliche Ursachen:

- Session-Cookie wird nicht gespeichert
- Cookie wird blockiert
- Session-Timeout zu kurz
- Server verliert Session-State
- Load Balancer verteilt ohne Session-Stickiness
- falsche Domain oder falscher Pfad im Cookie
- Zeitproblem zwischen Systemen
- Anwendung invalidiert Sitzung

Merksatz:

Ständige Abmeldung deutet oft auf Sitzungs- oder Cookie-Probleme hin.

Fehlerbild: Warenkorb ist plötzlich leer

Bei Webshops kann der Warenkorb Teil der Sitzung sein.

Mögliche Ursachen:

- Session-ID verloren
- Cookie gelöscht
- serverseitige Sitzung abgelaufen
- Benutzer wechselt Gerät oder Browser
- Load-Balancer-Problem
- Anwendung speichert Zustand falsch

Merksatz:

Verlorener Warenkorb kann verlorene Sitzung bedeuten.

Fehlerbild: Nach Login wieder auf Login-Seite

Mögliche Ursachen:

- Session-Cookie wird nicht gesetzt
- Cookie wird nicht zurückgesendet
- HTTPS/HTTP-Wechsel stört Cookie
- falsche Domain im Cookie
- SameSite-Problem
- Server speichert Sitzung nicht
- Uhrzeit oder Token-Gültigkeit falsch

Merksatz:

Login-Schleife ist oft ein Sitzungsproblem.

Fehlerbild: Remote-Sitzung bricht ab

Mögliche Ursachen:

- Netzwerkunterbrechung
- Timeout
- Server beendet Sitzung
- Lizenz- oder Ressourcenproblem
- Firewall trennt inaktive Verbindung
- VPN verliert Verbindung
- Keepalive fehlt

Merksatz:

Remote-Sitzungsabbrüche können Netzwerk- oder Sitzungsursachen haben.

Fehlerbild: Load Balancer und Sitzungen

Bei mehreren Webservern kann ein Load Balancer Anfragen verteilen.

Problem:

Anfrage 1 geht an Server A.
Anfrage 2 geht an Server B.

Wenn die Sitzung nur auf Server A gespeichert ist, kennt Server B den Benutzer nicht.

Lösungen:

- Session-Stickiness
- zentrale Session-Datenbank
- verteiltes Cache-System
- stateless Token

Merksatz:

Bei mehreren Servern muss der Session-State sauber geplant sein.

Session-Stickiness

Session-Stickiness bedeutet:

Ein Benutzer wird während einer Sitzung möglichst immer zum gleichen Server geleitet.

Das hilft, wenn Sitzungsdaten lokal auf einem Server gespeichert sind.

Nachteil:

Lastverteilung wird weniger flexibel.
Serverausfall kann Sitzungen betreffen.

Merksatz:

Session-Stickiness bindet eine Sitzung an einen Server.

Stateful und Stateless

Stateful bedeutet:

Ein System merkt sich Zustand.

Stateless bedeutet:

Jede Anfrage enthält alle nötigen Informationen
oder wird unabhängig verarbeitet.

Beispiel stateful:

Server speichert Session-Daten.

Beispiel stateless:

Token enthält relevante Informationen,
Server muss keine klassische Sitzung speichern.

Merksatz:

Stateful speichert Zustand.

Stateless vermeidet serverseitigen Sitzungszustand.

Sitzungsschicht und Datenschutz

Sitzungen können sensible Informationen enthalten.

Beispiele:

- Benutzerkennung
- Rollen
- Anmeldestatus
- Zugriffstoken
- Kundendaten
- Warenkorb
- Arbeitsstand

Deshalb müssen Sitzungsdaten geschützt werden.

Merksatz:

Sitzungsdaten können sicherheits- und datenschutzrelevant sein.

Was Schicht 5 nicht macht

Schicht 5 macht nicht:

- Bits über Kabel übertragen
- MAC-Adressen im LAN verwenden
- IP-Pakete routen
- TCP-Ports öffnen
- Daten verschlüsseln
- Zeichencodierung umwandeln
- konkrete Anwendungslogik vollständig bereitstellen

Diese Aufgaben liegen auf anderen Schichten.

Merksatz:

Schicht 5 verwaltet Sitzungen,
aber transportiert nicht selbst wie TCP.

Einordnung in das OSI-Modell

Thema	Schicht
Kabel, Signal	1
MAC-Adresse, Frame	2
IP-Adresse, Routing	3
TCP, UDP, Port	4
Sitzung, Session-ID, Dialogsteuerung	5
Codierung, Verschlüsselung, Kompression	6
HTTP, DNS, SMTP, Anwendung	7

Merksatz:

Sitzung = Schicht 5.
Port = Schicht 4.
Anwendung = Schicht 7.

Schicht 5, 6 und 7 gemeinsam verstehen

Die oberen OSI-Schichten sind in der Praxis oft eng miteinander verbunden.

Schicht	Grundidee
5	Sitzung verwalten
6	Daten darstellen, codieren, verschlüsseln
7	Anwendung und Netzwerkdienst

Beispiel Web:

Sitzung:
Login-Status und Session-Cookie

Darstellung:
TLS, Zeichencodierung, Datenformat

Anwendung:

HTTP, Webseite, API

Merksatz:

Schicht 5 bis 7 arbeiten in modernen Anwendungen oft zusammen.

Typische IHK-Fragen

In AP1 und AP2 kann zum Beispiel gefragt werden:

- Welche Aufgabe hat die Sitzungsschicht?
- Was ist eine Sitzung?
- Warum sind Sitzungen für Anwendungen wichtig?
- Was ist eine Session-ID?
- Was ist ein Session-Cookie?
- Was bedeutet Timeout?
- Was ist Sitzungswiederaufnahme?
- Was bedeutet Dialogsteuerung?
- Warum ist HTTP ohne Zusatzmechanismen zustandslos?
- Warum kann ein Benutzer ständig abgemeldet werden?
- Warum ist Sitzungsverwaltung sicherheitsrelevant?
- Was ist der Unterschied zwischen TCP-Verbindung und Anwendungssitzung?

Typische Prüfungsfallen

Schicht 5 heißt Sitzungsschicht.

Schicht 5 verwaltet Sitzungen.

Eine TCP-Verbindung ist Schicht 4.

Eine Benutzersitzung ist nicht automatisch dasselbe wie eine TCP-Verbindung.

HTTP ist grundsätzlich zustandslos.

Websitzungen nutzen oft Cookies oder Tokens.

Session-ID muss geschützt werden.

Session-Hijacking bedeutet Übernahme einer Sitzung.

Session-Fixation wird durch neue Session-ID nach Login verhindert.

Timeout kann Sitzungen automatisch beenden.

Load Balancer können Sitzungsprobleme verursachen.

Schicht 5 ist in modernen Protokollen oft schwer getrennt sichtbar.

Wichtige Begriffe kurz erklärt

Begriff	Kurze Erklärung
Sitzungsschicht	OSI-Schicht 5
Sitzung	zusammenhängende Kommunikationsbeziehung
Session	englischer Begriff für Sitzung
Session-ID	Kennung einer Sitzung
Session-Cookie	Cookie zur Wiedererkennung einer Websitzung
Token	Kennung oder Nachweis für Zugriff oder Sitzung
Dialogsteuerung	Steuerung des Kommunikationsablaufs
Synchronisationspunkt	Wiederaufnahmepunkt in einer Kommunikation
Timeout	automatische Beendigung nach Inaktivität
Keepalive	Mechanismus zum Aktivhalten
Session-State	gespeicherter Zustand einer Sitzung
Session-Hijacking	Übernahme einer gültigen Sitzung
Session-Fixation	Ausnutzung einer vorher bekannten Session-ID
Session-Stickiness	Bindung einer Sitzung an einen Server
Stateful	System speichert Zustand
Stateless	Anfragen werden ohne gespeicherten Zustand verarbeitet

IHK-sichere Kurzformulierung

Die Sitzungsschicht ist Schicht 5 des OSI-Modells. Sie ist für den Aufbau, die Verwaltung und das Beenden von Sitzungen zwischen Kommunikationspartnern zuständig. Eine Sitzung ist eine zusammenhängende Kommunikationsbeziehung, zum Beispiel eine angemeldete Websitzung, eine Remote-Desktop-Sitzung oder eine Datenbanksitzung. Die Sitzungsschicht kann Dialogsteuerung, Sitzungswiederaufnahme und Synchronisationspunkte unterstützen. In modernen Netzwerken sind Aufgaben der Sitzungsschicht oft in Anwendungen oder Protokolle höherer Schichten integriert, zum Beispiel durch Session-IDs, Cookies oder Tokens.

Merksätze

Schicht 5 = Sitzungsschicht.

Sitzung = zusammenhängende Kommunikationsbeziehung.

Schicht 4 transportiert.

Schicht 5 verwaltet Sitzungen.

TCP-Verbindung ist nicht automatisch Benutzersitzung.

Session-ID = Kennung einer Sitzung.

Session-Cookie erkennt Websitzung wieder.

HTTP ist grundsätzlich zustandslos.

Sitzungen schaffen Zusammenhang zwischen Anfragen.

Timeout beendet inaktive Sitzungen.

Keepalive hält Kommunikation aktiv.

Synchronisationspunkte helfen bei Wiederaufnahme.

Dialogsteuerung organisiert Kommunikation.

Session-Hijacking = Sitzung übernommen.

Session-Fixation = bekannte Session-ID wird missbraucht.

Nach Login Session-ID erneuern.

Logout sollte Sitzung serverseitig beenden.

Load Balancer können Sitzungsprobleme verursachen.

Stateful = Zustand wird gespeichert.

Stateless = kein serverseitiger Sitzungszustand.

Schicht 5 ist in modernen Protokollen oft in Anwendungen integriert.
