9.2 TLS, Zertifikate und Verschlüsselung

TLS gehört zu den wichtigsten Sicherheitsgrundlagen moderner Netzwerke.

TLS steht für:

```
Transport Layer Security
```

TLS wird genutzt, um Daten bei der Übertragung zu schützen.

Das bekannteste Beispiel ist:

```
HTTPS
```

HTTPS bedeutet:

```
HTTP über TLS
```

Merksatz:

```
TLS schützt Daten zwischen Client und Server.
```

Warum braucht man TLS?

Ohne Verschlüsselung können Daten auf dem Übertragungsweg mitgelesen oder manipuliert werden.

Beispiele für sensible Daten:

```
- Passwörter
- Session-Cookies
- Kundendaten
- Zahlungsdaten
- persönliche Daten
- API-Tokens
- interne Verwaltungsdaten
```

TLS schützt diese Daten während der Übertragung.

Merksatz:

```
TLS schützt Daten unterwegs.
```

Welche Schutzziele hat TLS?

TLS schützt vor allem drei Dinge:

Schutzziel	Bedeutung
Vertraulichkeit	Daten können nicht einfach mitgelesen werden
Integrität	Daten können nicht unbemerkt verändert werden
Authentizität	Gegenstelle kann überprüft werden

Merksatz:

```
TLS schützt Vertraulichkeit,
Integrität
und Authentizität.
```

Vertraulichkeit

Vertraulichkeit bedeutet:

```
Nur berechtigte Kommunikationspartner können die Daten lesen.
```

Beispiel:

```
Ein Benutzer meldet sich auf einer Webseite an.
```

Ohne TLS könnten Zugangsdaten im Netzwerk mitgelesen werden.

Mit TLS werden die Daten verschlüsselt übertragen.

Merksatz:

```
Vertraulichkeit = Schutz vor Mitlesen.
```

Integrität

Integrität bedeutet:

```
Daten werden nicht unbemerkt verändert.
```

Beispiel:

```
Ein Angreifer verändert unterwegs eine Antwort des Servers.
```

TLS soll solche Manipulationen erkennbar machen.

Wenn die Daten verändert wurden, wird die Verbindung als ungültig erkannt.

Merksatz:

```
Integrität = Schutz vor unbemerkter Veränderung.
```

Authentizität

Authentizität bedeutet:

```
Der Kommunikationspartner ist wirklich der,
für den er sich ausgibt.
```

Beispiel:

```
Browser verbindet sich mit:
https://bank.example
```

Der Browser prüft über das Zertifikat:

```
Passt das Zertifikat zum Namen?
Ist es gültig?
Ist es vertrauenswürdig?
```

Merksatz:

```
Authentizität = Echtheit der Gegenstelle.
```

TLS und HTTPS

HTTP ist unverschlüsselt.

HTTPS ist HTTP über TLS.

Vergleich:

Protokoll	Verschlüsselung	typischer Port
HTTP	nein	TCP 80
HTTPS	ja, TLS	TCP 443

Bei HTTPS werden HTTP-Daten innerhalb einer TLS-geschützten Verbindung übertragen.

Merksatz:

```
HTTPS = HTTP + TLS.
```

TLS im OSI-Modell

TLS wird oft der Darstellungsschicht zugeordnet.

Warum?

```
TLS verschlüsselt,
entschlüsselt
und schützt die Darstellung der übertragenen Daten.
```

In der Praxis liegt TLS zwischen Anwendung und Transport.

Vereinfacht:

```
HTTP
TLS
TCP
IP
Ethernet
```

Merksatz:

```
TLS liegt praktisch zwischen Anwendung und TCP.
```

TLS ist nicht TCP

TLS und TCP dürfen nicht verwechselt werden.

Thema	Einordnung
TCP	Transportprotokoll auf Schicht 4
TLS	Verschlüsselungs- und Schutzschicht
HTTPS	HTTP über TLS
Port 443	typischer HTTPS-Port

TCP baut die Transportverbindung auf.

TLS schützt die darüber übertragenen Daten.

Merksatz:

```
TCP transportiert.
TLS schützt.
```

TLS ist nicht HTTPS

TLS ist die Schutzschicht.

HTTPS ist ein Anwendungsprotokoll, das TLS nutzt.

Beispiel:

```
HTTPS = HTTP über TLS
```

TLS kann aber auch bei anderen Protokollen verwendet werden.

Beispiele:

```
- SMTPS
- IMAPS
```

```
- POP3S
- LDAPS
- FTPS
```

Merksatz:

```
HTTPS nutzt TLS,
aber TLS ist nicht nur HTTPS.
```

Zertifikat

Ein Zertifikat ist ein digitaler Nachweis.

Es hilft dem Client zu prüfen:

```
Spreche ich wirklich mit dem richtigen Server?
```

Ein TLS-Zertifikat enthält unter anderem:

```
- Domainname
- öffentlicher Schlüssel
- Aussteller
- Gültigkeitszeitraum
- Signatur
- alternative Namen
```

Merksatz:

```
Zertifikat = digitaler Vertrauensnachweis.
```

Domainname im Zertifikat

Ein Zertifikat muss zum aufgerufenen Namen passen.

Beispiel:

```
Aufgerufen:
wiki.firma.de
```

Das Zertifikat muss für diesen Namen gültig sein.

Wenn das Zertifikat nur für einen anderen Namen gilt, warnt der Browser.

Beispiel Fehler:

```
Zertifikat gilt für:
server.local

Aufgerufen wird:
wiki.firma.de
```

Merksatz:

```
Der Name im Browser muss zum Zertifikat passen.
```

Subject Alternative Name

Moderne Zertifikate nutzen häufig sogenannte alternative Namen.

Diese heißen:

```
Subject Alternative Name
```

Kurz:

```
SAN
```

Dort stehen die Namen, für die das Zertifikat gültig ist.

Beispiele:

```
wiki.firma.de
cloud.firma.de
www.firma.de
```

Merksatz:

```
SAN enthält gültige Namen eines Zertifikats.
```

Wildcard-Zertifikat

Ein Wildcard-Zertifikat gilt für mehrere Subdomains einer Domain.

Beispiel:

```
*.firma.de
```

Das kann gelten für:

```
wiki.firma.de
cloud.firma.de
mail.firma.de
```

Aber normalerweise nicht automatisch für:

```
firma.de
intern.wiki.firma.de
```

Merksatz:

```
Wildcard-Zertifikat gilt für viele Subdomains einer Ebene.
```

Zertifizierungsstelle

Eine Zertifizierungsstelle wird auch genannt:

```
Certificate Authority
```

Kurz:

```
CA
```

Die CA stellt Zertifikate aus und signiert sie.

Der Client vertraut dem Zertifikat, wenn er der ausstellenden CA vertraut.

Merksatz:

```
CA = vertrauenswürdige Stelle für Zertifikate.
```

Zertifikatskette

Ein Serverzertifikat steht meistens nicht allein.

Es gehört zu einer Zertifikatskette.

Typische Bestandteile:

```
- Serverzertifikat
- Zwischenzertifikat
- Root-Zertifikat
```

Der Client prüft die Kette bis zu einer vertrauenswürdigen Root-CA.

Merksatz:

```
Zertifikatskette verbindet Serverzertifikat mit vertrauenswürdiger CA.
```

Root-Zertifikat

Ein Root-Zertifikat ist ein besonders vertrauenswürdiges Zertifikat einer CA.

Betriebssysteme und Browser bringen viele Root-Zertifikate bereits mit.

Wenn ein Root-Zertifikat vertraut wird, können daraus abgeleitete Zertifikatsketten ebenfalls vertraut werden.

Merksatz:

```
Root-Zertifikat = Vertrauensanker.
```

Zwischenzertifikat

Zwischenzertifikate liegen zwischen Root-CA und Serverzertifikat.

Sie werden auch genannt:

```
Intermediate Certificates
```

Wenn ein Zwischenzertifikat fehlt, kann die Vertrauenskette unvollständig sein.

Folge:

```
Browser oder Client zeigt Zertifikatswarnung.
```

Merksatz:

```
Fehlendes Zwischenzertifikat kann TLS-Warnungen verursachen.
```

Selbstsigniertes Zertifikat

Ein selbstsigniertes Zertifikat wurde nicht von einer allgemein vertrauenswürdigen CA ausgestellt.

Es kann technisch verschlüsseln, aber der Client vertraut ihm normalerweise nicht automatisch.

Typische Nutzung:

```
- Testumgebung
- internes Labor
- Homelab
- Entwicklung
```

Für produktive öffentliche Dienste nutzt man normalerweise ein Zertifikat einer vertrauenswürdigen CA.

Merksatz:

```
Selbstsigniert kann verschlüsseln,
ist aber nicht automatisch vertrauenswürdig.
```

Gültigkeitszeitraum

Zertifikate sind zeitlich begrenzt gültig.

Ein Zertifikat hat:

```
gültig ab
gültig bis
```

Wenn ein Zertifikat abgelaufen ist, zeigen Clients Warnungen.

Auch eine falsche Systemzeit kann Zertifikatsfehler auslösen.

Merksatz:

```
Zertifikat und Systemzeit müssen gültig sein.
```

Öffentlicher und privater Schlüssel

TLS verwendet asymmetrische Kryptografie.

Dabei gibt es:

```
öffentlichen Schlüssel
privaten Schlüssel
```

Der öffentliche Schlüssel steckt im Zertifikat.

Der private Schlüssel bleibt geheim auf dem Server.

Wichtig:

```
Der private Schlüssel darf nicht weitergegeben werden.
```

Merksatz:

```
Öffentlicher Schlüssel darf bekannt sein.
Privater Schlüssel muss geheim bleiben.
```

Warum darf der private Schlüssel nicht verloren gehen?

Wenn ein Angreifer den privaten Schlüssel besitzt, kann er sich unter Umständen als der Server ausgeben oder geschützte Kommunikation gefährden.

Deshalb muss der private Schlüssel geschützt werden.

Schutzmaßnahmen:

- Dateirechte
- sichere Speicherung
- Zugriffsbeschränkung
- keine Weitergabe
- regelmäßige Erneuerung bei Verdacht

Merksatz:

Privater Schlüssel ist kritisch.

TLS-Handshake

Beim TLS-Handshake einigen sich Client und Server auf die geschützte Kommunikation.

Vereinfacht passiert:

1. Client kontaktiert Server.
2. Server sendet Zertifikat.
3. Client prüft Zertifikat.
4. Beide einigen sich auf Verschlüsselungsverfahren.
5. Sitzungsschlüssel werden gebildet.
6. Danach werden Daten verschlüsselt übertragen.

Merksatz:

TLS-Handshake bereitet die sichere Verbindung vor.

TCP-Handshake und TLS-Handshake unterscheiden

Bei HTTPS passieren mehrere Schritte.

Vereinfacht:

1. TCP-Verbindung wird aufgebaut.
2. TLS-Handshake wird durchgeführt.
3. HTTP-Daten werden verschlüsselt übertragen.

Unterschied:

Vorgang	Zweck
TCP-Handshake	Transportverbindung aufbauen
TLS-Handshake	sichere Verschlüsselung aushandeln
HTTP-Anfrage	Anwendungskommunikation

Merksatz:

Erst TCP,
dann TLS,
dann HTTP.

Symmetrische und asymmetrische Verschlüsselung

TLS nutzt Konzepte aus beiden Bereichen.

Asymmetrisch:

öffentlicher und privater Schlüssel

Symmetrisch:

gemeinsamer Sitzungsschlüssel

Warum?

Asymmetrische Verfahren helfen beim sicheren Aushandeln.
Symmetrische Verfahren sind schneller für die Datenübertragung.

Merksatz:

```
Asymmetrisch für Aushandlung,
symmetrisch für schnelle Datenübertragung.
```

Sitzungsschlüssel

Ein Sitzungsschlüssel ist ein temporärer Schlüssel für eine konkrete Verbindung.

Er wird während des TLS-Handshakes ausgehandelt.

Danach werden die Nutzdaten mit diesem Sitzungsschlüssel verschlüsselt.

Merksatz:

```
Sitzungsschlüssel schützt die konkrete TLS-Verbindung.
```

TLS-Versionen

TLS hat verschiedene Versionen.

Wichtige Versionen:

```
TLS 1.2
TLS 1.3
```

Ältere Versionen wie SSL oder TLS 1.0 und TLS 1.1 gelten heute als veraltet.

Für moderne Systeme gilt:

```
aktuelle TLS-Versionen verwenden
alte unsichere Verfahren deaktivieren
```

Merksatz:

```
SSL ist veraltet.
TLS 1.2 und TLS 1.3 sind heute wichtig.
```

SSL und TLS

SSL steht für:

```
Secure Sockets Layer
```

SSL ist der ältere Vorgänger von TLS.

Im Alltag sagen manche noch „SSL-Zertifikat“.

Technisch korrekt ist heute meistens:

```
TLS-Zertifikat
```

oder einfach:

```
Zertifikat für HTTPS
```

Merksatz:

```
SSL ist alt,
TLS ist der moderne Begriff.
```

Cipher Suite

Eine Cipher Suite beschreibt, welche kryptografischen Verfahren für eine TLS-Verbindung genutzt werden.

Dazu gehören zum Beispiel:

```
- Schlüsselaustausch
- Verschlüsselung
- Integritätsschutz
```

Moderne Systeme sollten sichere Cipher Suites verwenden.

Merksatz:

```
Cipher Suite = Kombination kryptografischer Verfahren.
```

Perfect Forward Secrecy

Perfect Forward Secrecy bedeutet:

> Selbst wenn später ein langfristiger Schlüssel kompromittiert wird,
> sollen alte aufgezeichnete Verbindungen nicht nachträglich entschlüsselt werden können.

Dafür werden kurzlebige Sitzungsschlüssel verwendet.

Für AP1/AP2 reicht meistens:

> Moderne TLS-Konfigurationen sollen alte Verbindungen besser schützen.

Merksatz:

> Forward Secrecy schützt alte Sitzungen besser.

Zertifikatsprüfung durch den Client

Ein Client prüft bei TLS unter anderem:

> - Ist das Zertifikat noch gültig?
> - Passt der Name zur aufgerufenen Adresse?
> - Ist die Zertifikatskette vollständig?
> - Ist die ausstellende CA vertrauenswürdig?
> - Wurde das Zertifikat widerrufen?
> - Ist die Signatur korrekt?

Wenn eine Prüfung fehlschlägt, erscheint eine Warnung oder die Verbindung wird blockiert.

Merksatz:

> TLS-Vertrauen entsteht durch mehrere Prüfungen.

Zertifikatswiderruf

Ein Zertifikat kann vor Ablauf ungültig werden.

Gründe:

- privater Schlüssel kompromittiert
- Zertifikat falsch ausgestellt
- Domain nicht mehr berechtigt
- Sicherheitsvorfall

Möglichkeiten zur Prüfung:

- CRL
- OCSP

Merksatz:

Zertifikate können vor Ablauf widerrufen werden.

Häufige Zertifikatsfehler

Typische Zertifikatsfehler sind:

- Zertifikat abgelaufen
- Zertifikat noch nicht gültig
- falscher Domainname
- Zertifikat nicht vertrauenswürdig
- Zwischenzertifikat fehlt
- selbstsigniertes Zertifikat
- Zertifikat wurde widerrufen
- Systemzeit falsch
- falsches Zertifikat auf dem Server

Merksatz:

Zertifikatsfehler sind häufig Namens-, Zeit- oder Vertrauensprobleme.

Fehlerbild: Zertifikat abgelaufen

Symptom:

Browser meldet abgelaufenes Zertifikat.

Mögliche Ursachen:

- Zertifikat nicht erneuert
- automatische Erneuerung fehlgeschlagen
- falsches Zertifikat aktiv
- Server lädt alte Zertifikatsdatei
- Systemzeit falsch

Lösungsidee:

Zertifikat erneuern,
Dienst neu laden,
Zertifikatskette prüfen.

Merksatz:

Abgelaufenes Zertifikat muss erneuert werden.

Fehlerbild: Name passt nicht

Symptom:

Browser meldet,
dass das Zertifikat nicht zum Namen passt.

Beispiel:

Aufruf:
https://wiki.firma.de

Zertifikat gültig für:

cloud.firma.de

Mögliche Ursachen:

- falsches Zertifikat installiert
- falscher virtueller Host
- DNS zeigt auf falschen Server
- Reverse Proxy liefert falsches Zertifikat
- SAN-Eintrag fehlt

Merksatz:

Aufgerufener Name muss im Zertifikat enthalten sein.

Fehlerbild: Zertifikat nicht vertrauenswürdig

Symptom:

Client vertraut dem Zertifikat nicht.

Mögliche Ursachen:

- selbstsigniertes Zertifikat
- interne CA nicht installiert
- Zertifikatskette unvollständig
- unbekannte CA
- manipuliertes Zertifikat
- Proxy mit eigener CA

Merksatz:

Client muss der ausstellenden CA vertrauen.

Fehlerbild: Zwischenzertifikat fehlt

Symptom:

Manche Clients akzeptieren Zertifikat,
andere zeigen Warnungen.

Mögliche Ursache:

Server liefert die Zertifikatskette nicht vollständig aus.

Besonders ältere oder strengere Clients können dann nicht bis zur Root-CA prüfen.

Merksatz:

Server muss vollständige Zertifikatskette liefern.

Fehlerbild: Falsche Systemzeit

Symptom:

Viele Zertifikate wirken plötzlich ungültig.

Mögliche Ursache:

Datum oder Uhrzeit des Clients ist falsch.

Warum?

Zertifikate haben einen Gültigkeitszeitraum.

Wenn die Systemzeit falsch ist, kann ein gültiges Zertifikat als ungültig bewertet werden.

Merksatz:

Falsche Uhrzeit kann TLS-Probleme verursachen.

Fehlerbild: TLS-Version zu alt

Symptom:

Verbindung zu einem Dienst schlägt fehl,
obwohl IP und Port erreichbar sind.

Mögliche Ursache:

Client oder Server unterstützt nur veraltete TLS-Versionen.

Beispiele:

TLS 1.0
TLS 1.1
alte SSL-Versionen

Moderne Systeme blockieren solche Verbindungen oft.

Merksatz:

Alte TLS-Versionen können Verbindungen verhindern.

Fehlerbild: Cipher Suite passt nicht

Symptom:

TLS-Verbindung kommt nicht zustande.

Mögliche Ursache:

Client und Server finden kein gemeinsames sicheres Verfahren.

Das kann passieren bei:

- alten Clients
- alten Servern
- zu strenger TLS-Konfiguration
- veralteten Verschlüsselungsverfahren
- fehlenden Updates

Merksatz:

Client und Server brauchen gemeinsame TLS-Verfahren.

Fehlerbild: HTTPS geht intern, extern nicht

Mögliche Ursachen:

- DNS intern und extern unterschiedlich
- Reverse Proxy liefert extern anderes Zertifikat
- Portweiterleitung falsch
- Firewall blockiert TCP 443
- Zertifikat gilt nicht für externen Namen
- CGNAT oder Providerproblem
- interne CA wird extern nicht vertraut

Merksatz:

HTTPS-Probleme können DNS, TLS, Firewall und NAT betreffen.

Fehlerbild: HTTPS geht, aber Browser warnt

Wenn eine Webseite erreichbar ist, aber der Browser warnt, ist meist der Transportweg bis zum Server möglich.

Das Problem liegt dann häufig bei:

- Zertifikat
- Name
- Vertrauenskette
- Gültigkeit
- TLS-Konfiguration

Merksatz:

Erreichbar mit Warnung bedeutet oft:
TLS-Vertrauen stimmt nicht.

Verschlüsselung ist nicht gleich Authentifizierung

Verschlüsselung bedeutet:

```
Daten sind nicht lesbar für Unbefugte.
```

Authentifizierung bedeutet:

```
Identität wird geprüft.
```

TLS kann beides unterstützen:

```
Verschlüsselung der Daten
Prüfung des Serverzertifikats
```

Aber:

```
Ein verschlüsselter Kanal allein sagt nicht,
dass der Benutzer berechtigt ist.
```

Merksatz:

```
Verschlüsselung schützt Daten.
Authentifizierung prüft Identität.
```

Serverauthentifizierung

Bei HTTPS authentifiziert sich normalerweise der Server gegenüber dem Client.

Der Browser prüft das Serverzertifikat.

Ziel:

```
Der Client soll wissen,
dass er mit dem richtigen Server spricht.
```

Merksatz:

```
Serverzertifikat bestätigt die Serveridentität.
```

Clientzertifikat

TLS kann auch Clientzertifikate verwenden.

Dann authentifiziert sich nicht nur der Server, sondern auch der Client mit einem Zertifikat.

Das nennt man häufig:

```
mTLS
```

mTLS steht für:

```
mutual TLS
```

Typische Nutzung:

```
- interne APIs
- Zero-Trust-Umgebungen
- besonders geschützte Systeme
- Maschinen-zu-Maschinen-Kommunikation
```

Merksatz:

```
mTLS = beide Seiten weisen sich mit Zertifikat aus.
```

TLS und E-Mail-Protokolle

Auch E-Mail-Protokolle können TLS nutzen.

Beispiele:

Protokoll	unverschlüsselt	verschlüsselte Variante
SMTP	TCP 25	STARTTLS oder SMTPS
IMAP	TCP 143	IMAPS TCP 993
POP3	TCP 110	POP3S TCP 995

Merksatz:

```
Viele klassische Protokolle haben TLS-geschützte Varianten.
```

STARTTLS

STARTTLS bedeutet:

```
Eine Verbindung beginnt zunächst unverschlüsselt
und wird dann auf TLS umgestellt.
```

Typisch bei:

```
SMTP
IMAP
POP3
```

Wichtig:

```
STARTTLS ist nicht dasselbe wie von Anfang an verschlüsselte Verbindung auf eigenem TLS-Port.
```

Merksatz:

```
STARTTLS schaltet innerhalb einer bestehenden Verbindung auf TLS um.
```

Ende-zu-Ende-Verschlüsselung und Transportverschlüsselung

TLS ist normalerweise Transportverschlüsselung.

Das bedeutet:

```
Die Verbindung zwischen zwei Kommunikationspunkten ist geschützt.
```

Ende-zu-Ende-Verschlüsselung bedeutet:

```
Daten bleiben vom Absender bis zum endgültigen Empfänger geschützt,
auch wenn Zwischenserver beteiligt sind.
```

Beispiel:

```
TLS schützt Verbindung zum Mailserver.
E-Mail-Ende-zu-Ende-Verschlüsselung schützt den Inhalt bis zum Empfänger.
```

Merksatz:

```
TLS ist meist Transportverschlüsselung,
nicht automatisch Ende-zu-Ende-Verschlüsselung.
```

TLS und Reverse Proxy

Ein Reverse Proxy kann TLS beenden.

Das nennt man:

```
TLS Termination
```

Ablauf:

```
Client → HTTPS → Reverse Proxy
Reverse Proxy → HTTP oder HTTPS → interner Server
```

Vorteile:

```
- zentrale Zertifikatsverwaltung
- einheitliche TLS-Konfiguration
- Lastverteilung
- Schutz interner Dienste
```

Risiko:

```
Interne Strecke muss ebenfalls passend geschützt werden,
wenn dort sensible Daten laufen.
```

Merksatz:

```
TLS Termination beendet TLS am Reverse Proxy.
```

TLS Passthrough

TLS Passthrough bedeutet:

```
Der Reverse Proxy leitet die TLS-Verbindung weiter,
ohne sie selbst zu entschlüsseln.
```

Der interne Server verarbeitet dann TLS selbst.

Vorteil:

```
TLS bleibt bis zum Zielserver bestehen.
```

Nachteil:

```
Proxy kann Inhalte nicht so einfach auswerten oder steuern.
```

Merksatz:

```
TLS Passthrough leitet verschlüsselt weiter.
```

HSTS

HSTS steht für:

```
HTTP Strict Transport Security
```

HSTS teilt dem Browser mit:

```
Diese Webseite soll nur noch über HTTPS aufgerufen werden.
```

Dadurch wird verhindert, dass der Browser versehentlich unverschlüsselt über HTTP verbindet.

Merksatz:

```
HSTS erzwingt HTTPS im Browser.
```

Mixed Content

Mixed Content bedeutet:

```
Eine HTTPS-Seite lädt unsichere HTTP-Inhalte nach.
```

Beispiel:

```
Seite selbst:
https://wiki.firma.de

eingebundenes Bild oder Script:
http://example.com/script.js
```

Das kann Sicherheitswarnungen verursachen oder blockiert werden.

Merksatz:

```
HTTPS-Seite sollte keine HTTP-Inhalte nachladen.
```

TLS und Performance

TLS braucht Rechenleistung, besonders beim Verbindungsaufbau.

Moderne TLS-Versionen sind aber effizient.

Möglichkeiten zur Optimierung:

```
- TLS 1.3
- Session Resumption
- moderne Cipher Suites
- HTTP/2 oder HTTP/3
- saubere Zertifikatskette
```

Merksatz:

```
TLS kostet etwas Aufwand,
ist aber für Sicherheit unverzichtbar.
```

TLS-Fehlersuche systematisch

Eine sinnvolle Reihenfolge:

```
1. IP-Erreichbarkeit prüfen
2. TCP-Port 443 prüfen
3. Zertifikat prüfen
4. Domainname prüfen
5. Zertifikatskette prüfen
6. Gültigkeitszeitraum prüfen
7. Systemzeit prüfen
8. TLS-Version und Cipher Suites prüfen
9. Reverse Proxy prüfen
10. Anwendungslogs prüfen
```

Merksatz:

```
Erst Verbindung,
dann TLS-Vertrauen,
dann Anwendung.
```

TLS oder DNS?

Viele TLS-Fehler hängen mit DNS zusammen.

Beispiel:

```
DNS zeigt auf falschen Server.
```

Folge:

```
Man erreicht einen Server,
aber dieser liefert ein Zertifikat für eine andere Domain.
```

Prüfung:

```
Welche IP liefert DNS?
Welcher Server antwortet?
```

```
Welches Zertifikat wird geliefert?
Passt der Name?
```

Merksatz:

```
Falsches DNS kann falsches Zertifikat verursachen.
```

TLS oder Firewall?

Wenn der TCP-Port 443 nicht erreichbar ist, kommt TLS gar nicht richtig zum Einsatz.

Dann prüft man zuerst:

```
- Routing
- Firewall
- NAT
- Portweiterleitung
- Dienst lauscht auf TCP 443
```

Wenn TCP 443 erreichbar ist, aber eine Zertifikatswarnung erscheint, prüft man TLS und Zertifikat.

Merksatz:

```
Kein TCP 443 = Schicht 4 prüfen.
Zertifikatswarnung = TLS prüfen.
```

TLS oder Anwendung?

Wenn TLS erfolgreich ist, kann die Anwendung trotzdem Fehler haben.

Beispiele:

```
- HTTP 404
- HTTP 500
- Loginfehler
- Berechtigungsfehler
- API-Fehler
- falscher Reverse-Proxy-Pfad
```

Dann ist nicht TLS das Hauptproblem, sondern die Anwendung auf höherer Schicht.

Merksatz:

> TLS erfolgreich heißt nicht:
> Anwendung korrekt.

Was TLS nicht macht

TLS macht nicht:

- IP-Adresse vergeben
- Routing übernehmen
- TCP-Port öffnen
- DNS-Namen auflösen
- Benutzerrechte festlegen
- Webanwendung fehlerfrei machen
- Server automatisch sicher konfigurieren
- Schadsoftware verhindern

TLS schützt nur einen wichtigen Teil:

> die Verbindung und Vertrauensprüfung.

Merksatz:

> TLS schützt die Übertragung,
> ersetzt aber kein Sicherheitskonzept.

Einordnung in das OSI-Modell

Thema	Schicht
IP-Adresse	3
Routing	3
TCP 443	4
TLS-Handshake	6 mit Schicht-4-/7-Bezug
Zertifikat	6 / 7

Thema	Schicht
HTTPS	7 über TLS
HTTP-Inhalt	7
DNS-Name	7
Cookie	7 mit Sitzungsbezug

Merksatz:

TCP 443 ist Schicht 4.
TLS schützt darüber.
HTTPS ist die Anwendung darüber.

Typische IHK-Fragen

In AP1 und AP2 kann zum Beispiel gefragt werden:

- Was ist TLS?
- Wofür wird TLS verwendet?
- Was ist der Unterschied zwischen HTTP und HTTPS?
- Welche Schutzziele bietet TLS?
- Was ist ein Zertifikat?
- Was prüft ein Client bei einem Zertifikat?
- Warum erscheint eine Zertifikatswarnung?
- Was ist eine Zertifizierungsstelle?
- Was ist eine Zertifikatskette?
- Was ist ein selbstsigniertes Zertifikat?
- Warum muss der Domainname zum Zertifikat passen?
- Was ist der Unterschied zwischen Verschlüsselung und Authentifizierung?
- Was ist STARTTLS?
- Was bedeutet TLS Termination?
- Warum ist TLS allein kein vollständiges Sicherheitskonzept?

Typische Prüfungsfallen

TLS ist nicht TCP.

TLS ist nicht HTTPS.

HTTPS nutzt TLS.

HTTP ist unverschlüsselt.

HTTPS nutzt typischerweise TCP 443.

TCP-Handshake kommt vor TLS-Handshake.

Zertifikat muss zum Namen passen.

Zertifikat muss gültig sein.

Client muss der CA vertrauen.

Zertifikatskette muss vollständig sein.

Selbstsigniert heißt nicht automatisch vertrauenswürdig.

Falsche Systemzeit kann Zertifikatsfehler verursachen.

SSL ist veraltet.

TLS ist der moderne Begriff.

Verschlüsselung ist nicht dasselbe wie Authentifizierung.

TLS schützt die Übertragung,
aber nicht automatisch die Anwendung.

Wichtige Begriffe kurz erklärt

Begriff	Kurze Erklärung
TLS	Transport Layer Security
SSL	älterer Vorgänger von TLS
HTTPS	HTTP über TLS
Zertifikat	digitaler Vertrauensnachweis
CA	Zertifizierungsstelle

Begriff	Kurze Erklärung
Root-Zertifikat	Vertrauensanker
Zwischenzertifikat	Verbindung zwischen Root und Serverzertifikat
Zertifikatskette	Kette von Serverzertifikat bis Root-CA
SAN	alternative Namen im Zertifikat
Wildcard-Zertifikat	Zertifikat für mehrere Subdomains
öffentlicher Schlüssel	Schlüssel im Zertifikat
privater Schlüssel	geheimer Schlüssel des Servers
TLS-Handshake	Aushandlung sicherer Kommunikation
Cipher Suite	Kombination kryptografischer Verfahren
Session Key	temporärer Schlüssel für eine Verbindung
STARTTLS	Umschalten einer Verbindung auf TLS
mTLS	gegenseitige Zertifikatsauthentifizierung
TLS Termination	TLS endet am Proxy
TLS Passthrough	TLS wird verschlüsselt weitergeleitet
HSTS	Browser soll HTTPS erzwingen
Mixed Content	HTTPS-Seite lädt HTTP-Inhalte

IHK-sichere Kurzformulierung

TLS steht für Transport Layer Security und schützt Daten bei der Übertragung zwischen Client und Server. Es bietet Vertraulichkeit, Integrität und Authentizität. Bei HTTPS wird HTTP über TLS übertragen, typischerweise über TCP-Port 443. Ein TLS-Zertifikat hilft dem Client zu prüfen, ob er mit dem richtigen Server spricht. Dazu prüft der Client unter anderem den Domainnamen, den Gültigkeitszeitraum, die Zertifikatskette und die Vertrauenswürdigkeit der Zertifizierungsstelle. TLS ist nicht dasselbe wie TCP und nicht dasselbe wie HTTPS: TCP stellt den Transport bereit, TLS schützt die Verbindung und HTTPS nutzt TLS für Webkommunikation.

Merksätze

```
TLS = Transport Layer Security.

TLS schützt Daten bei der Übertragung.

HTTPS = HTTP über TLS.

HTTP = unverschlüsselt.
```

HTTPS = verschlüsselt.

HTTPS nutzt typischerweise TCP 443.

TCP transportiert.

TLS schützt.

HTTP ist Anwendung.

TLS bietet Vertraulichkeit.

TLS bietet Integrität.

TLS bietet Authentizität.

Zertifikat = Vertrauensnachweis.

CA = Zertifizierungsstelle.

Root-Zertifikat = Vertrauensanker.

Zertifikatskette muss vollständig sein.

Domainname muss zum Zertifikat passen.

Zertifikat darf nicht abgelaufen sein.

Client muss der CA vertrauen.

Selbstsigniert ist nicht automatisch vertrauenswürdig.

Privater Schlüssel muss geheim bleiben.

Erst TCP-Handshake,
dann TLS-Handshake,
dann HTTP.

SSL ist veraltet.

TLS ist modern.

STARTTLS schaltet auf TLS um.

TLS Termination beendet TLS am Proxy.

TLS Passthrough leitet TLS verschlüsselt weiter.

HSTS erzwingt HTTPS.

Mixed Content vermeiden.

TLS schützt die Verbindung,
ersetzt aber kein vollständiges Sicherheitskonzept.