



Hochschule **RheinMain**  
University of Applied Sciences  
Wiesbaden Rüsselsheim

# SECURITY

Protokolle für sichere Kommunikation

June 13, 2023

Marc Stöttinger

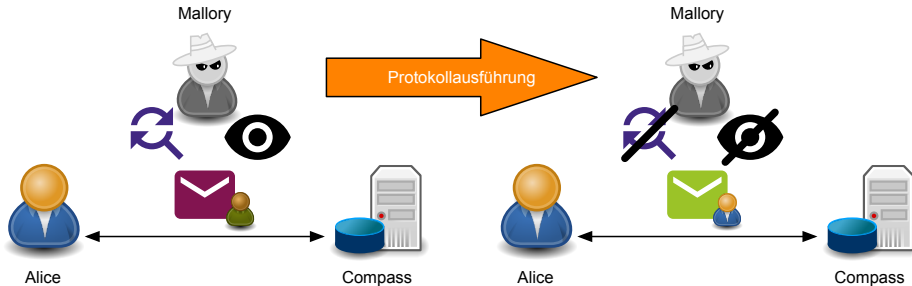


Secure communication protocols serve as the fortified gateways that protect the sanctity of our digital interactions.

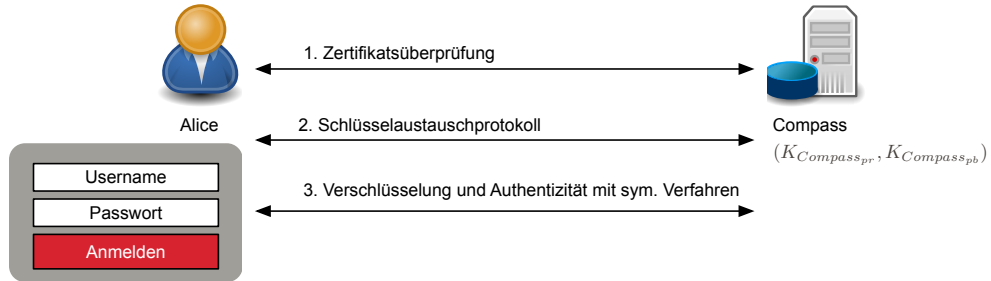
Whitfield Diffie

# MOTIVATION

- Bisher: Protokolle zur Authentifikation von Personen im Internet
- Heute: Protokolle zum Aufbau eines sicheren Kommunikationskanals
  - Start: Alle Nachrichten abhör- und manipulierbar
  - Ziel: Sicherer (vertraulicher, authentischer und integrier) Kommunikationskanal



# GROBABLAUF SICHERE KOMMUNIKATIONSPROTOKOLLE



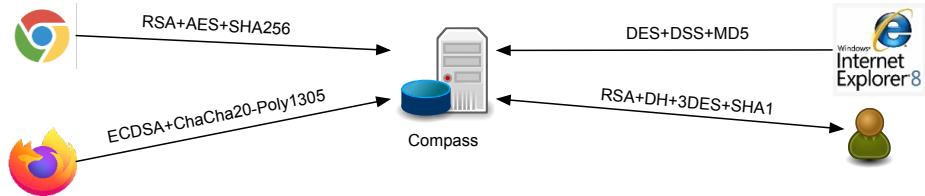
# HERAUSFORDERUNGEN FÜR SICHERE KOMMUNIKATIONSPROTOKOLLE

Herausforderungen für standardisierte, sichere Kommunikationsprotokolle

- Geräte und Anforderungen im Internet sind sehr heterogen (Leistung, Bandbreite, Plattform...)
- Einzelschritte der Protokolle müssen sicher zusammengeführt werden
- Einbettung der Protokolle im Netzwerkstack ist komplex

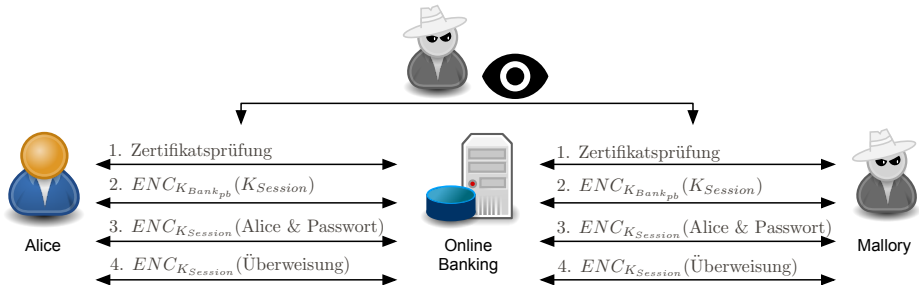
# MOTIVATION SICHERE PROTOKOLLE HETEROGENITÄT IM INTERNET

- Geräte im Internet sind sehr heterogen
  - Interoperabilität mit alten Systemen muss gewährleistet sein
  - Unterschiedliche Krypto Verfahren müssen unterstützt werden
  - Manche Anwendungen erfordern Zertifikats-basierte Authentifikation beider Parteien



# MOTIVATION SICHERE PROTOKOLLE SICHERE VERBINDUNG EINZELSCHRITTE

- Die Einzelschritte müssen sicher zusammengefügt werden
  - Ansonsten können kleinste Schwachstellen für Angriffe ausgenutzt werden
  - Beispiel: Mallory liest die Nachrichten von Alice und sendet sie erneut (Replay Angriff)



# MOTIVATION SICHERE PROTOKOLLE SICHERE VERBINDUNG EINZELSCHRITTE

- Komplexe Vorgänge in der Kommunikationstechnik werden in Schichten eingeteilt
  - OSI Modell
  - TCP/IP Modell
- Schichten werden nacheinander ausgeführt und bieten darüberliegenden Schichten bestimmte Dienste an
  - Transportschicht: Steuerung des Datenflusses
  - Internetschicht: Adressierung von Paketen
  - Netzzugriff: Zugriff auf das Netzwerk

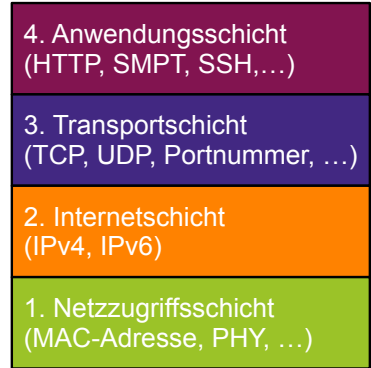
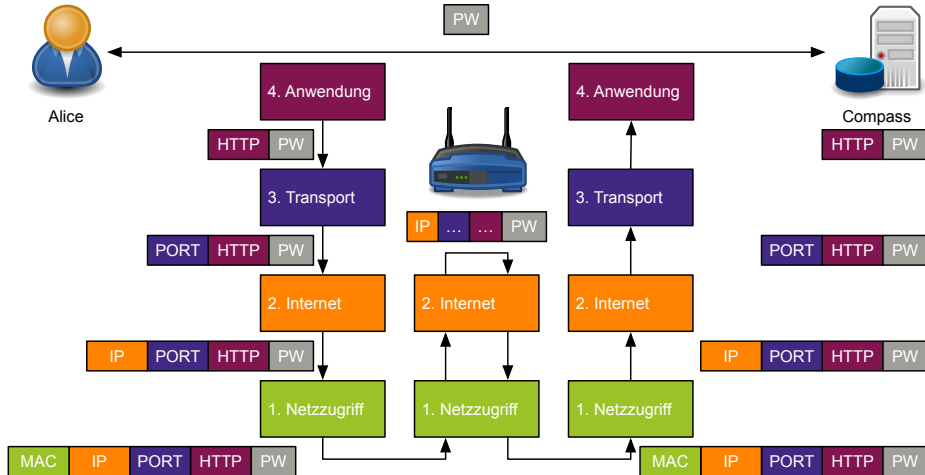


Figure: TC/IP Modell

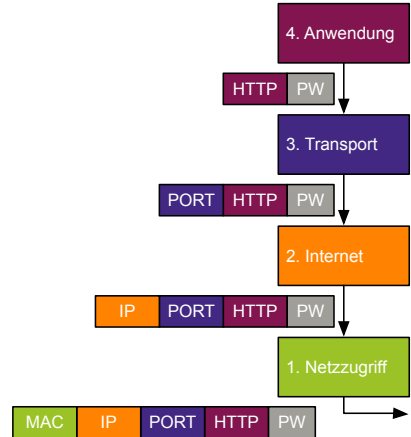


# DATENÜBERTRAGUNG IM SCHICHTENMODELL



# ABSICHERUNG DER PAKETDATEN

- **Frage:** In welcher Schicht soll die Absicherung stattfinden?
  - Je weiter unten, desto mehr Daten werden abgesichert
  - Je weiter oben, desto länger bleiben die Daten abgesichert
- Die sinnvollste Schicht zur Absicherung kann je nach Kontext und Anwendung variieren



# VARIANTE DER ABSICHERUNG DER NUTZDATEN

## → Beispiele

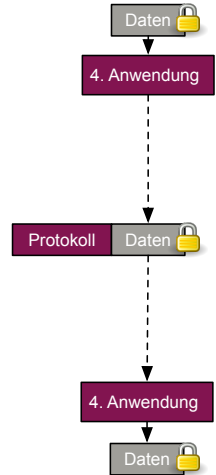
- Chat-Nachrichten oder Dateien absichern
- Kann vor oder in der Anwendungsschicht geschehen

## → Einsatzzwecke

- Mögliche Ende-zu-Ende Verschlüsselung (E2E)
- Der Anwendung wird nicht vertraut (Speicherung in der Cloud)

## → Limitierungen

- Protokoll- und Metadaten sind lesbar (wer sendet Chatnachricht)
- Applikationsspezifische Sicherheitsprotokolle nötig bei Absicherung in der Anwendung



# VARIANTE DER ABSICHERUNG NUTZ- UND PROTOKOLLDATEN

## → Beispiele

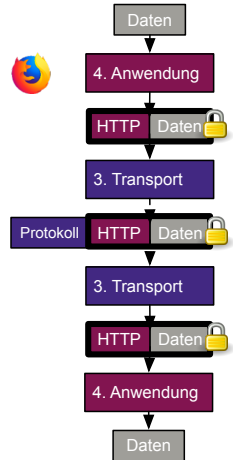
→ Webseiten, E-Mails

## → Einsatzzwecke

→ Sichere Verbindung zwischen Anwendungen inkl. Protokolldaten

## → Limitierungen

- Port, IP- und MAC Adressen les- und änderbar
- Eine sichere Verbindung je Anwendung wird benötigt
- Code im Kontext der gleichen Anwendung hat Zugriff auf die Nutzdaten (z.B. andere Webseiten)



# VARIANTE DER ABSICHERUNG PORT- UND IP ADRESSE

## → Beispiele

→ Sicheres Virtual Private Network (VPN)

## → Einsatzzwecke

→ Sichere Verbindung zwischen Rechnern

→ Absicherung Port: Rechner zu Rechner

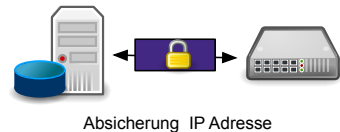
→ Absicherung IP: Rechner/Netzwerk zu Netzwerk

## → Limitierungen

→ MAC Adressen les- und änderbar

→ Nutz- und Protokolldaten sind am Ziel ungesichert

→ Komplexere Konfiguration



# VARIANTE DER ABSICHERUNG MAC ADRESSE

## → Beispiele

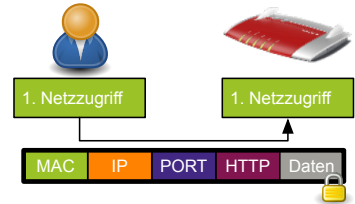
→ Sicherer Zugang zum Internet (MACSec, WPA2/3, ...)

## → Einsatzzwecke

→ Sichere Verbindung zum Router bzw. nächsten Hop  
→ Absicherung des lokalen Netzwerkes

## → Limitierungen

→ Schlüssel müssen auf Geräte verteilt werden, damit sie Zugang zum Netzwerk erhalten  
→ Kommunikation nur abgesichert bis zum Internetzugang



# ÜBERSICHT ABSICHERUNG DER PAKETDATEN

→ Welche Paketdaten sollen abgesichert werden?

MAC	IP	PORT	HTTP	PW
-----	----	------	------	----

Abgesicherte Daten	Einsatzzweck	Limitierungen	Protokolle
Nutzdaten	Sichere Ende-zu-Ende (E2E) Kommunikation für Anwendung (z.B. eMail oder WhatsApp)	Protokolldaten lesbar (HTTP GET/POST), Anwendungsspezifisch	Signal
+ Protokolldaten (z.B. HTTP)	Sichere Verbindung zu einer Anwendung (z.B. Webserver)	Eine sichere Verbindung pro Dienst wird benötigt (z.B. Unternehmens-IT)	Transport Layer Security (TLS)
+ Port und IP	Sichere Verbindung zu einem Host/Netzwerk (z.B. VPN)	Komplexe Netzwerkadministration, Absicherung geht nicht bis zu Anwendung	Internet Protocol Security (IPSec)
+ MAC Adressen	Absicherung des lokalen Netzwerkes (z.B. im Fahrzeug)	Komplexe Netzwerkadministration aufgrund vorher verteilter Schlüssel	WPA2/3, MACsec (MAC Security)

# TRANSPORT LAYER SECURITY (TLS – 1/2)

- TLS ist das meist genutzte Protokoll für sichere Kommunikation im Internet
  - Früher bekannt als Security Socket Layer (SSL)
- Browser über HTTPs
- eMail Clients über SMTP/IMAP/POP3
  - Früher bekannt als Security Socket Layer (SSL)
- TLS speichert **Zustandsinformationen** in **Sitzungen**, von denen mehrere gleichzeitig aktiv sein können (z.B. eine Sitzung pro Webseite)

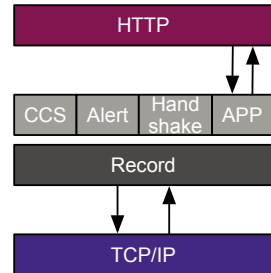
Standard	Nutzungs- zeitraum	Unterstützende Webseiten (Dez22)
SSL1.0	1994 - ?	-
SSL2.0	1995 - 2011	0,2%
SSL3.0	1996 - 2015	2,1%
TLS1.0	1999 - 2021	343,0%
TLS1.1	2006 - 2021	37,0%
TLS1.2	2008+	99,9%
TLS1.3	2018+	58,9%



# TRANSPORT LAYER SECURITY (TLS – 2/2)

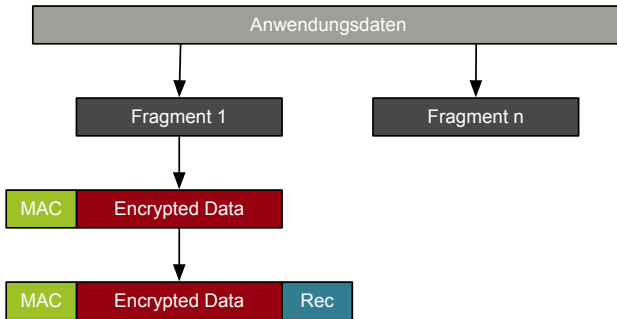
→ TLS liegt zwischen Anwendungs- und Transportschicht und besteht aus fünf verschiedenen Protokollen

1. **Change Cipher Spec (CCS)**: Aushandlung der genutzten Krypto Verfahren
2. **Alert Protocol**: Fehlerbehandlung und Verbindungsabbruch
3. **Handshake**: Aushandlung der Sitzungsinformationen und des Sitzungsschlüssels
4. **Application**: Transparente Kommunikation mit Anwendung
5. **Record Layer**: Teilt Daten in Fragmente und sorgt für deren Absicherung



# TLS - RECORD LAYER PROTOKOLL

→ Das Record Layer Protokoll fragmentiert Anwendungsdaten transparent und nutzt symmetrische Kryptographie, um die Sicherheit der Daten zu gewährleisten



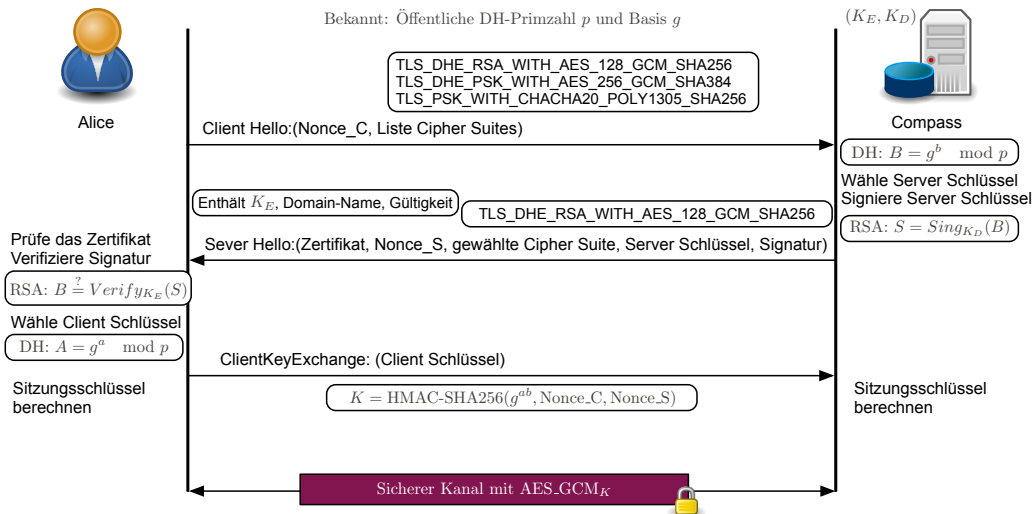
Der TLS Record enthält:

- Typ des überliegenden Protokolls (CCS, Alert, Handshake, Applikation)
- TLS Versionsinformationen
- Länge der Nutzdaten

# TLS - HANDSHAKE PROTOKOLL

- Pro Sitzung müssen verschiedene Informationen ausgehandelt werden
  - Verwendete kryptographische Verfahren
  - Wer muss sich authentifizieren? (Keiner, nur Server, Alice und Server)
  - Symmetrischer Schlüssel für Record Layer Protokoll (sog. Sitzungsschlüssel)
- Kryptographische Verfahren werden mittels der Cipher Suite ausgehandelt
  - Beispiel: TLS\_**DHE**\_**RSA**\_WITH\_**AES\_128\_GCM**\_SHA256
- Eine Cipher-Suite definiert
  - **Schlüsselaustausch** (Diffie-Hellman mit Schlüssellöschung – sog. DH Ephemera)
  - **Authentifizierung** (RSA Signaturen)
  - **Verschlüsselung** (AES-128 GCM)
  - **Hashfunktion** (SHA256)

# HANDSHAKE PROTOKOLL TLS1.2 RSA SIGNATUR UND DH SCHLÜSSELAUSTAUSCH



# DISKUSSION IN KLEINEN GRUPPEN

## Sicherheit von TLS

- Wie wird verhindert, dass ein Angreifer die Nachrichten einer alten Sitzung einspielt, um einen Replay-Angriff durchzuführen?
- Wieso muss der Server Schlüssel  $B$  signiert werden?
- Wieso kann ein Angreifer den Sitzungsschlüssel  $K$  nicht berechnen?

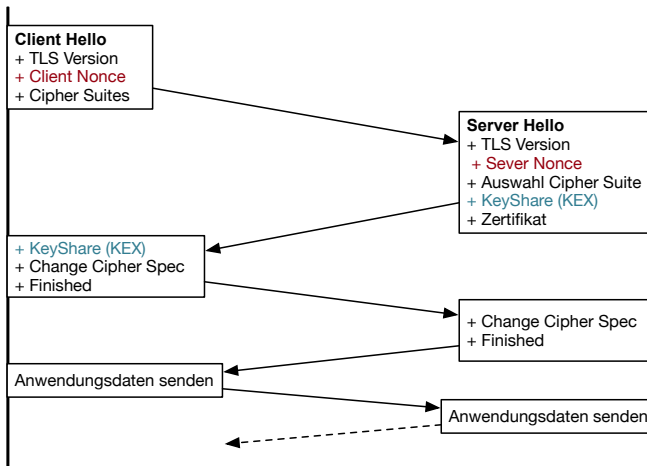
# HANDSHAKE PROTOKOLL TLS1.2



Alice



Compass



# HANDSHAKE PROTOKOLL TLS1.3



Alice

**Client Hello**

- + TLS Version
- + **Client Nonce**
- + Cipher Suites
- + Unterstützte Versionen und Param.
- + **KeyShare(KEX)**
- + Pre-Shared Keys
- + Vorzeitige Daten

- + Finished
- Anwendungsdaten senden

**Server Hello**

- + TLS Version
- + **Sever Nonce**
- + Auswahl Cipher Suite
- + Unterstützte Versionen und Param.
- + **KeyShare (KEX)**
- + Change Cipher Spec
- + Zertifikat
- + Finished

Anwendungsdaten senden



Compass

# TLS1.3 VS. TLS1.2

Hauptunterschied zwischen TLS1.3 [RFC8446] und TLS 1.2 [RFC5246]:

- Unsichere veraltete Verfahren wurden rausgenommen
  - Die Cipher Suite wurde auf fünf Sets reduziert
  - Kein statischer Schlüsselaustausch erlaubt
  - Schlüsselaustauschverfahren nur noch mit (EC)DHE, PSK-only und PSK mit (EC)DHE
- Verschlüsselung der Kommunikation nach Handshake Nachricht ServerHello
- Kryptographische Verfahren basieren auf Elliptischen Kurven und gehören zum Basisset
- Reduktion des Handshake-Protokolls zum schnelleren Aufbau des gesicherten Kommunikationskanals



# SICHERHEITSPROBLEME BEI TLS

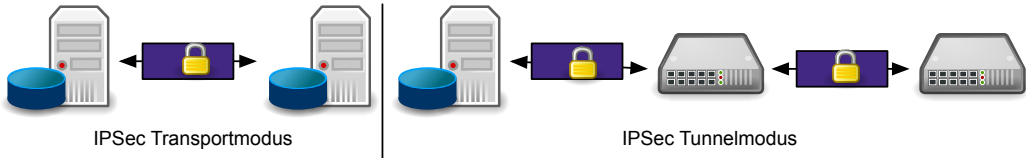
- Viele bekannte Angriffe gegen alte TLS / SSL Versionen
  - Bleichenbacher ( $\leq$ SSL3.0): Angriffe auf RSA Padding Verfahren [Bleichenbacher]
  - Beast ( $\leq$ TLS1.2): Angriff auf Cipher-Block-Chaining (CBC) Initialisierungsvektor [BEAST]
  - Poodle ( $\leq$ TLS1.0): Angriff auf Padding Verfahren in CBC [POODLE]
- Häufiger Angriffsvektor Downgrade: Angreifer bringt Opfer und Server dazu, eine alte TLS Version oder anfällige Cipher-Suite zu nutzen
  - Gegenmaßnahme: Abschalten alter TLS Versionen und Cipher-Suites
- Implementierungsfehler in TLS Bibliotheken
  - Heartbleed: Softwarefehler, der Auslesen zufälliger Bereiche im RAM ermöglichte [HB]

# INTERNET PROTOCOL SECURITY (IPSEC)

- IPSec ist eine Familie von Protokollen, zur sicheren Kommunikation, die auf der Internetschicht arbeiten
  - **Internet Key Exchange (IKE)**: Protokoll zum Schlüsselaustausch und Überprüfung der Authentizität der Endgeräte
  - **Authentication Header (AH)**: Authentizität und Integrität der Kommunikation
  - **Encapsulation Security Payload (ESP)**: Vertraulichkeit, Authentizität und Integrität der Kommunikation

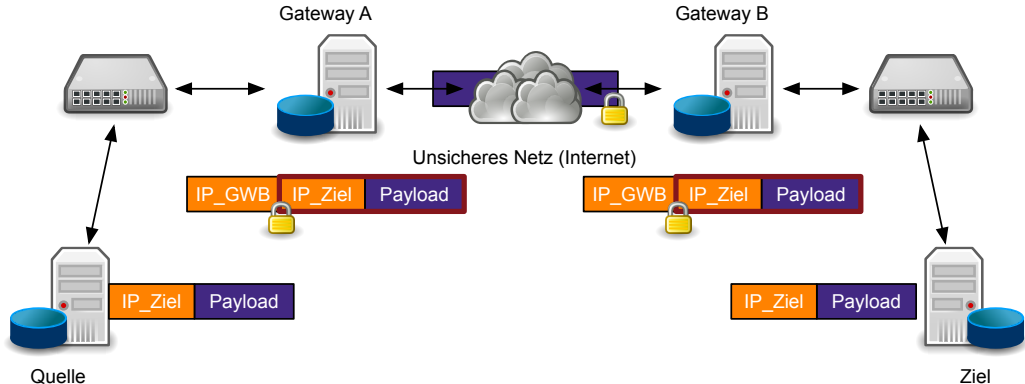
# INTERNET PROTOCOL SECURITY (IPSEC)

- IPSec unterstützt zwei verschiedene Modi
  - **Transportmodus**: Sichere Verbindung zweier Geräte
  - **Tunnelmodus**: Sichere Verbindung in Netzwerke (Virtual Private Network - VPN)
- AH und ESP unterscheiden sich je nachdem, ob sie für den Transport- oder Tunnelmodus eingesetzt werden



# ESP IM TUNNELMODUS ARCHITEKTUR

ESP im Tunnelmodus verschlüsselt die Ziel IP und den Payload und leitet das Paket an das Ziel Gateway weiter



# ESP IM TRANSPORT- UND TUNNELMODUS

Paket abgesichert im  
Transportmodus

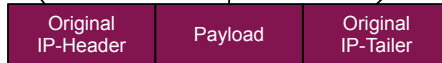
Verschlüsselt und Authentisch



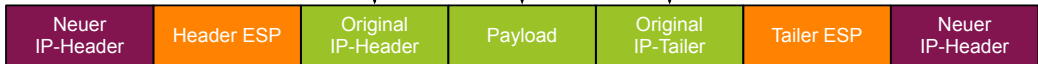
Authentisch

Ungesichert

Original Paket  
auf Schicht 2



Paket abgesichert im  
Tunnelmodus

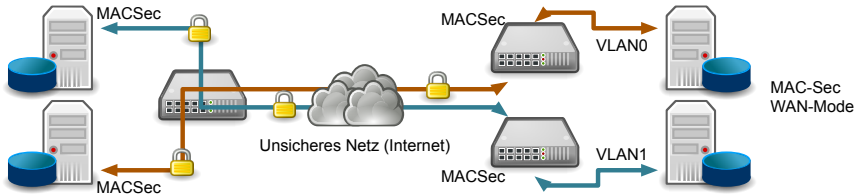
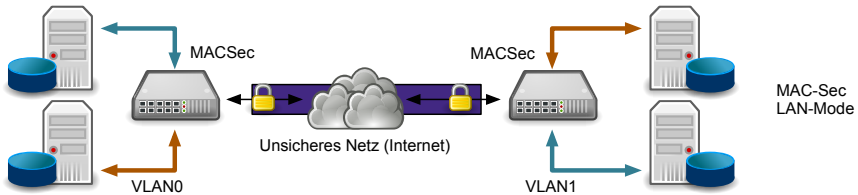


# MEDIA ACCESS CONTROL SECURITY (MACSEC)

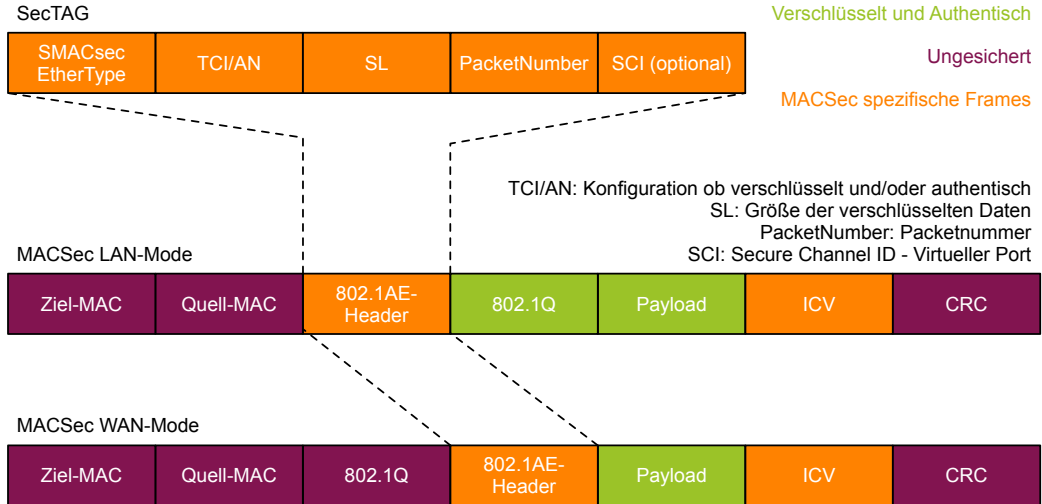
- MACSec ist in IEEE 802.AE standardisiert zur sicheren Kommunikation, die auf der Netzzugriffsschicht arbeitet
  - MACSec basiert auf einem Standard Ethernet Frame und wird um zwei Felder erweitert
    - **MACsec Security Tag (SecTAG)**: Kontrollfeld mit Konfigurationsinformationen
    - **Integrity Check Value (ICV)**: Authentizitätstoken 16 Byte
  - der MACSec Frame kann mit AES-GCM gesichert werden und somit verschlüsselt und authentisch sein
  - Schlüssel für die Absicherungen können statisch vorab geteilt werden (PSK) oder über einen Schlüsselservers mit Authentisierungsprotokollen (EAP) via IEEE 802.1X

# MACSEC MODUS

MACSec hat zwei Betriebsmodi, welche sich auf die Nutzung von VLANs auswirken.



# MACSEC PAKETSTRUKTUR





# ZUSAMMENFASSUNG

- Herausforderungen für Protokolle zur sicheren Kommunikation
- Einsatzzwecke und Limitierungen bei der Absicherung in verschiedenen TCP/IP Schichten
- Sichere Kommunikationsprotokolle der verschiedenen TCP/IP Schichten
- Grobe Funktionsweise von TLS, um einen sicheren Kommunikationskanal zu etablieren
- IPSec Protokollfamilie sowie den Tunnel- und Transportmodus
- Konzept hinter dem IPSec Tunnelmodus
- Konzept hinter MACSec