## 1 Crypto Recap

# 1.1 Objectives:

- Confidentiality
- Integrity
- Authenticity
- Availability
- Authorization
- · Non-Repudiation, Accountability
- Freshness · Anonymity, Unlinkability
- · Intervenability, Contro
- Transparency

## 1.2 Confidentiality-Encryption

## 1.2.1 Symmetric Ciphers

- · Secret key for en- and decryption
- · Much more efficient
- Block cipher: encryps a plaintext block of fixed len e.g.: Advanced Encryption Standard
- Stream cipher: encrypts a bitstream e.g.:ChaCha20

### 1.2.2 Asymmetric Ciphers

- · Public key for encryption
- · Private key for decryption
- · Ex.: RSA-based encryption

## 1.3 Integrity, Authenticity-Signatures, MACs

### 1.3.1 MACs

- · Symmetric cryptography
- Protects data integrity & authenticity
- Ex.: Hash-based MAC

## 1.3.2 Digital Signatures

- · Asymmetric cryptography
- Signing with private key
- · Protects data integrity & authenticity
- Provices non-repudation

## 1.4 Block Cipher Modes of Operation

### 1.4.1 Electronic Code Book (ECB)

- Each plaintext block is encrypted seperatly
- Inherintly insecure! -> Smae block = Same ciphe

## 1.4.2 Cipher Block Chainning (CBC)

- · Plaintext is chained to previous ciphertext by XOR and encrypted afterwards
- Difficult to apply securely -> implementations often vulnerable

### 1.4.3 Galois Counter Mode (GCM)



## 2 Tranport Layer Security (TLS)

## 2.1 TLS handshake protocol

- · Parameter Negotiation
- Kev exchange Authentication

## 2.2 TLS record protocol

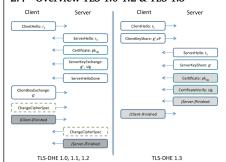
- · Protection of integrity, authenticiy and confidentiality
- . Symmetric Cryptography: e.g., block cipher, usually AES

## 2.3 Attacks

- · Attacks on the record layer
- · Attacks on the session key
- · Attacks on the private server key
- Attacks on implmentations: Timing Attacks, Heratbleed, Invalid Point Attacks
- · Attacks on TLs eco system

- Attacks on certificates and the PKI
- Attacks on the browser GUI

## 2.4 Overview TLS 1.0-1.2 & TLS 1.3



## 2.5 HKDF Key Derivation Function

## 2.6 Forward Secrecy

- · TLS 1.3 using certificate-based Authentification: forward secrecy
- TLS 1.3 using pre-shared keys (PSK):
  - PSK using elliptic-curve Diffie-Hellman: forward secrecy
  - PSK without EC-DHE (symmetric-only) and zero-round-trip data: no forward

## 2.7 Datagramm TLS

- · DTLS is identical to TLS where possible
- DTLS has to introcue new mechanisms
  - Explicit sequence numbers
  - Retransmission timer for handshake message

  - Message re-ordering for the handshake
  - Fragmentation of large handshake messages into serveral DTLS records
  - Optional replay detection - Invalid records are discarded (silently)
  - Denial-of-Service countermeasures: statless cookies, HelloRetryRequest

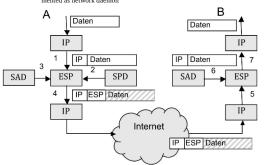
## 3 Network Layer Security: IPsec

## 3.1 TLS limitations

- · Does not protect all traffic between hosts
  - Protects only a single connection; new connection => new key agreement or session resumption necessary
  - Non-TCP traffic (resp. non-UDP trafic for DTLS) cannto be protected
- . Does not protect the TCP layer (RST attacks on TCP are possible which terminate TLS
- Application specific: applications have to be modified to use TLS/DTLS
- Using a TLS-based tunel for VPNs has disadvangates (TCP) (DTLS good option tho)

## 3.2 Overview IPsec

- · Goal: Protect IP packets cryptographically
  - Confidentiality
- Integrity & Authenticity
- · Seperation of packet protection from key exchange
  - Protocols for packet encryption and authentification: symmetric crypto, implemented in the TCP/IP stack
  - Key Exchange completly unrelated: asymmetric (+symmetric) crypto, implemented as network daemon



## 3.3 Packet Formats and Operational Modes

## 3.3.1 Authentification Header (AH)

- · Integrity & Authenticity only
- · Partially protects öuterÏP header
  - Variable fields cannot be includes
  - \* Flags, Fragment Offset, TTL, Header Checksume
  - Other fields are included in MAC computation
  - Version, IP header length, total length, identification, Protocol (must be AH), source/dest Address

## 3.3.2 Encapsulating Security Payload (ESP)

- · Confidentiality, Integrity & Authenticity
- · Does not protect öuterÏP header
- Security Parameter Index (SPI) Sequence Number Payload = inner IP packet (variable length) - encrypted and integrity-protected Padding Length Integrity Check Value - ICV (variable length)
  - SPI: SA-Identifier
  - Next Header: type of payload data (e.g. IPV4)
  - · Integrity Check Value: data for authentification / integrity protection
  - Message Authentification Code (MAC)
  - . SAs are negotiated in the key exchange
  - Sender side: Packet must be categorized to determine which SA applies (By parameters) such as destination=)
  - · Receiver side: Determines SA from IPsec header of received packet
  - SA Parameters:
    - IPsec protocol (AH or ESP)
    - Authentification alrogrithm and key
    - Encryption algorithm and key
    - IPsec mode (transport or tunnel)
    - SA lifetimne
    - Sequence number: current counter
    - On reciver side: SSliding Window"for replay protection

## 3.4 Scenarios

- Host to host
- · Host to gateway
- · Gateway to gateway

## 3.5 The Internet Key Exchange (IKE)

- · So far, keys are already exchanges, SAs negotiated... but how?
- Authentification of the communication endpoints
- Dynamic negotiation fo algorithms and parameters
- Key establishment - Forward Secrecy
- Resistance to DoW attacks

# · Efficiency (computation, messages, round trips)

## 3.6 IKEv1

- · Phase 1: main mode vs. aggressive mode · Authentification modes:
  - Digital Signatures
- Public key encryption(PKE): two variants (rarely used)
- Pre-shared kevs(PSK)
- · Phase 2 (quick mode) With DH (=> forward secrecy)



- Phase 2, Quick Mode
  - Requires previous Phase-1 exchange Protected based on SA negotiated in Phase 1
  - DH-shares optional (forward secrecy)
  - k<sub>SPI</sub>: "master key" for ESP SA (to derive further keys)

## 3.6.1 Discussion

Complicated

Phase 1 Main Mode

- Information from serveral (complex) RFCs required
- Many options variants Gneric

I-cookie, R-cookie, I-enc-auth-data

I-cookie, R-cookie, R-enc-auth-data

Negotiate SA, keys for use in IKE (not for IPsec ESP/AH)

- Clean seperation of different phases and functionalities
- \* Can be advantagous in theory, but leads to inefficincies Intended as general handshake and key agreement protocol, not only for IPsec \* In practice: only used for IPsec

- Inneficient
- \* Requires too many round trips
- Inadequate DoS protection
- \* IKEv1 uses stateful cookie

#### 3.7 IKEv2

- Phase 2 (partially) combindes with Phase 1
  - More efficient thatn IKEv1
  - Initial perotiation of IPsec SA included
- Additional SAs can be negotiaded in further Phase 2 protocol runs Simplified specification
  - Essential information in one REC.
- No distinction main mode vs. aggressive mode
- · DoS protection using statless cookies; optional



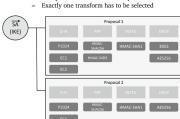
- $spi_I$ ,  $spi_R$ , SA-selection,  $g^y$ ,  $n_R$ Optional use of certificates (not shown here) Derive keys  $k_{eI}^{\downarrow}$ ,  $k_{eR}$ ,  $k_{aI}$ ,  $k_{aI}$
- TS: Traffic Selectors Describe for which traffic the proposed SAs should be applied IKE AUTE "Stand-alone" phase 2: similar to IKE\_AUTH
- $spi_I$ ,  $spi_R$ ,  $c_I := enc_{k_{eI}}(ID_I, sig_I, SA-proposals_2, TS)$ ,  $MAC_{k_{eI}}(c_I)$ New nonces, new SA negotiation
- $spi_I$ ,  $spi_R$ ,  $c_R := enc_{k_{eR}}(ID_R, sig_R, SA\text{-selection}_2, TS)$ ,  $MAC_{k_{eR}}(c_I)$ Optional DH (forward secrecy)

Derive kevs for IPsec (ESP/AH) Derive kevs for IPsec (ESP/AH)

## 3.8 SA Proposals

No signatures necessary

- Serverl proposals possible
- · Proposal contains transforms
- · Differnten options for each transform possible



# 4 Security on Layer 2 (MAC Layer, Ethernet)

# 4.1 Attacks

- MAC address spoofing ARP spoofing ARP cache poisoning

## 4.2 Network Access Control

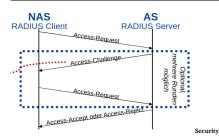
- Frontend protocols: Cient -> Network Access Server - Point-to-Point connections: PPP
  - LAN: PPPoE, EAPol.
- WLAN: WPA/WPA2/WPA3, EAPol. · Backend protocols: Network Access Server -> Authentication Server AAA protocols
  - (Authentication, Authorization, Accounting)
  - Radius, Diameter



## 4.2.1 Radius - Remote Authentification Dial-In User Service

- · Backend protocol
- Dial-In Server
  - E.g. DSL
  - Company infrastructure

Packet Format



· Old protocol

- Tries to protect sesitive data only

 Insecure cryptography · Uses only over secure protocols

Usually TLS

· RADIUS successor: Diameter

- Fmore flexible, extensible, application-aware

- SSecurity"features removed, no more insecure crypto of RADIUS => To be used over secure protocols TLS/DTLS/IPsec

## 4.2.2 Point-to-Point Protocol (PPP)

· PPP: Frontend protocol to connect two devices

- Typically used for dial-up connections (e.g. DSL)

- Transported over Layer 2 protocol (e.g. ethernet)

· Authentification mechanisms

- PAP: username/password

- CHAP: Challange Handshake Authentification Protocol

- EAP: Extensible Authentification Protocol

· PPPoE: PPP over Ethernet (e.g. used for DSL connections)

- Discovery: Client negotiates session with network access server

- PPP Session: PPP frames are encapsulated in ethernet frames; potentially several

## 4.2.3 Extensible Authentification Protocol (EAP)

· Extensible authentification framework

- Fronted protocol for lcient authentification

- Server send request (e.g. challenge, ident. request, ..)

- Client responds

- Server sends success or failure message

- EAP messages can be transported in RADIUS messages (as RADIUS attribute)

· EAPoL: EAP over LAN

### 4.2.4 Port Based Network Access Control (PNAC)

· Authentification of clients befre they can use the network (LAN)

- Client connect to the network

\* Authentification traffic is allowed

- Switch port can only be used for other purposes after successful authentification

- Re-authentification after timeouts, link-down, etc.

PNAC uses EAPoL

· Terminology: Port Access Entities (PAE)

- Client: supplicant

Netowrk access server: authenticator

- Attention: "port"

### 4.3 MACsec

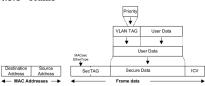
· cryptographically protect the ethernet layer Based on MACsec Kev Agreeement (MKA)

- After successful authentification (typically EAP)

- Derives key from CAK - connectivity association key (pre-shared)

- Key can be set up by EAP (e.g. EAP-TLS)

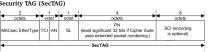
## 4.3.1 Frame



· Data cen be encryped and/or authentificated

· AES-GCM is supported

Security TAG (SecTAG)



· TCI: TAG Control Information

· AN: Association Number

· SL: Short Length (payload length; for short frames only)

· SCI: Secre Channel Identifier (edentify SA, when CA needs ore than 4)

- SA is identified by SCI (optional) and AN

## 4.4 Summary

Security on Layer 2 can protect all higher Layer

- But: only in same network

Does nt work accross Laver 3 routers

Network Access Control protects access to the network

- Client Authentification

- RADIUS/DIAMETER - FAP

- Port-Based Access Control

· MACsec protects traffic - Encryption & authentification

## 5 Wireless Security

## 5.1 WiFi Security

## 5.1.1 Historic Overview

1999: WEP (Wired Equivilaent Privacy)

- Goal: äs secure as a wired LAN"

 Insecure, various attacks known 2003: WPA (WiFi Protected Access)

- Improved protocols; most known attacks on WEP prevented

But: Requirement ofr hardware-compatibility with WEP devices => Encryption

improved, but still based on obsolete stream cipher

2004: WPA2 (still used)

- Similar to WPA, but AES-based encryption: AES-CCMP

· 2018: WPA3 (supported by new devices)

- Serveral improvements: Prevention of offline-attacks on pre-shared keys, forward secrecy, encryption for öpen"WLANs

#### 5.1.2 WPA/WPA2 Security

Personal Mode

 Pre-Shared Keys · Enterprise Mode

- EAP-TLS, PEAP EAP-TTLS

AES-CCMP

· Authenticated Encryption with Associated Data (AED)

AFS-CCM

- Authentication: CRC-MAC

- Encryption: Counter Mode (CTR)

- MAC and encyprion: computed simultaneously

## 4-Way Handshake

· Based on Pairwise Master Key (PMK)

 Personal Mode (WPA-PSK): Computed from passphrase and SSID (as ßalt") using PBKDF2 (password-based key derivation function)

Enterprise Mode: Established by key exchange protocol (e.g. EAP-TLS, PEAP)

4-Way HS to derive Pairwise Transient Key (PTK)

Exchance nonces

PTK is derived by hashing PMK, nonces, MAC addrs

Furhter key (for differnet purposes) derived from PTK

Client gets Group Temporary Key from AP (encrypted)

Message Integrity Code(MIC): MACs for integrity protection, key confirmation

 KRACK (2018): Key reinstallation attacks => meanwhile prevented by software/firmware updates

Problem: Offline attacks against passphrase

### 5.1.3 WPA 3 Improvements

Mandatory Protected Managment Frames

Prevents deauthentication attacks (DoS)

Replace PSK Authentication with SAW protocol

Simultaneous Authentication among Equals (SAE): "Dragonfly"handshake Prevents offline attacks on passphrase

Based on elliptic curve cryptography by default

· Forward Secrecy based on Diffie-Hellman

· 192-bit Security Mode (optional)

- AES-256 (GCM) SHA-384

- 284-bit elliptic curves or RSA with at least 3K bits

### 5.1.4 Simultaneous Authentication among Equals

SAE "Dragonflyäuthenicates participants and establishes PMK

Based on passphrase and (EC-)Diffie-Hellman

Can be initiated simultaneously by both parties (useful for mesh networking)

· 4-Way Handshake

- Esablishes PTK basen on PMK Same as in WPA2

- But now: PMK with much higher entropy => Offlione attacks not practical

· Hash-to-Group: "Hunting and Pecking"

 Generate point on elliptic curve from pasphrase (and MAC addresses, etc.) Cryptographic has function generates pseudo random numbers (by including a

counter in the input)

\* Both parties must use the exact same inputs in the same order

Fixed procedure to derive x-coordinate

Check if point on curve can be generated

\* If check fails: increase coutner and try again

Auth-Commit messages

Exchange ECDH shares

Auth-Confirm messages

- Key confirmation, authentication of messages

## 5.2 Bluetooth Security

Authentication: device authentication, no user authentication

Pairing/bondig: create shared keys: used in connections later on

Confidentiality: encryption of BT communication

 Message Integrity: MACs (authenticated encryption) to protect BT communication Authorization: control access to resources (based on devices, not users)

Security Modes

Mode 1: no security

Mode 2: service level (only for backward compatibility)

- Mode 3: link-level enforces security (only for backward compatibility) - Mode 4: authenticated link key using SSecure Connections", based on device

Eavesdropptin not trivial: Bluetooth uses frequency hoppting (not a security feature)

## 5.2.1 Device Pairing

Authentication and generation of link key / long term key

PIN/Legacy Pairing: enter PIN on both devices

- Key generation based on PIN, device address, and random values

Secure Simple Pairing (SSP): since Bluetooth 2.1

Numeric Comparison

\* Compare 6-digit numbers Passkev Entry

\* Read 6-digit form one device, enter on the other one

 Just Works \* User accepts connection without verification

 Out of Band (OOB) \* Transmit data using other communication channels (e.g. NFC)

## 5.2.2 Simple Secure Pairing (SSP)

Unauthenticated ECDH

· 2-Stage Authentication

- Stage 2: depends on pariing method

- Stage 2: Cryptographic authentication based on Stage 1 values and ECDH secret Key derivation to generate link key / long term key

## 5.2.3 Secure Authentication

· Paired (bonded) devices authenticate each other

· Challange-Response scheme

 128-bit random challanges - Response: HMAC of BT addresses and challanges (using link key from pairing)

\* Before Bluetooth 4.1: based on Bluetooth-specific algorithm E1

· Authenication failure: introcude delay (exponential back-off)

5.2.4 Confidentiality

· Bluetooth-specific stream cipher E0 Designed for efficiency

Serious attacks hve been published

- "Practicalin theory (but complex, hard to apply in practice)

AES-CCM

 Used since Bluetooth 4.1 - Key derived from link key (pairing) and the authentication step

5.2.5 Privacy · Privacy problem: Devices (users) can be identified by Bluetooth MAC addresses

· Mitigation: BLE private device addresses Resolvable Private Address (RPA) is changed periodically

Identity Address remains constant (but is not transmitted over the air)

- Identity Resolving Key to map RPA to Identity Address Especially imprtant to discoverable devices (which advertice identity info)

5.2.6 5.x Security · No major changes to security protocols and algorithms

Bluetooth 5.0

- PHY improvements, no relevant security changes

 Bluetooth 5.1 - HCI support for debug keys (should not be relevant in production systems) Bluetooth 5.2: adds new features (Extended Attributes, Isochronous Communication,

- Isochronous communication: connection-oriented or connection-less \* Group communication: group keys need to be established

\* Broadcast Authenication

Bluetooth 5.3: Key Size Negotiation - Enables host to define minimun key size

5.2.7 BLUFFS · BLUFFS: New attacks against bluetooth

- Breaks Forward Secrecy and Future Secrecy

- Enables man-in-the-middle attacks, impersonation if one session key compromi-Forces weak key: spec allows minimus of 7 Bytes entropy (56 Bits)

\* Brute-force attack: offline, parallelizable

\* Forces reuse of compromised key

Attack against bluetooth spec (BR/EDR: "Bluetooth Classic" versions 4.2 to 5.4):

All compliant devices are affected

Published and presented at ACM CCS 2023

## 5.2.8 implementations Vulnerabilities

- On Windows, IOS, Linux, Android

· Android (2018): implementation flaws in L2CAP and SMP

BleedingTooth(2020): several bugin in Linux

- Bluetooth controllers: SoC firmware Vulnerabilities(Link Manager)

· Windows (2021): BT Driver Elevation of Privilege

- Estimation 1400 bluetooth chips/modules affected

· Many attacks in the past

Bluetooth versions before 2.1 are basically completely insecure

- But the implementations not necessarily!

Bluetooth 5.2 architecture similar to 4.x

## 6 Security Mobile Networks

♦ 2<sup>nd</sup> generation (2G): GSM 1990s

Generation 2.5: GPRS (packet-switched data traffic)

♦ 3<sup>rd</sup> generation (3G): UMTS 2000s

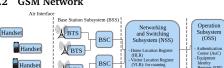
► Improved security: mutual authentication, better crypto algorithms

♦ 4<sup>th</sup> generation (4G): LTE 2010s

▶ Higher bandwidth

Changes to security

## ... even better, of course ;-) ..



RTS: Base Tranceiver Station

- Stable channel between communication partners is reserved Seperate channel for signaling (different time slices)

6.2.1 Secuirty

· Goal: Ät least as secure as a landline"

· Security functions

\* Shared secren in AuC and SIM \* Authentication via challange-response protocol

\* Ägreementön short term session keys for encryption Confidentiality Protection

\* Encryption of user data (üser plane") \* Encryption of signalling ("control plane")

\* Only for signalling \* Often implicitly by encryption

· SIM card introduced with GSM

BlueBorn(2017); Collection of implementation Vulnerabilities

Buffer overflow, integer overflows,

Remote Memory Disclousure

- Can even lead to arbitrary code execution in kernal mode

BrakTooth(2021)

5.2.9 Summary

· Complex protocol stack, not easy to implement

- on cryptography algorithms

Bluetooth versions sinde 4.2 are relativly secure (...but: "BLUFFS"!)

- Introduces new features and minor security improvements

6.1 Overview

First generation: analog telephony networks (voice only)

► Digital, circuit-switched, SIM-based; introduced cryptographic protection

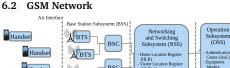
► Generation 2.75: EDGE (increased bandwidth)

Much higher bandwidth

Generation 3.5: HSDPA/HSUPA (increased bandwidth)

► Completely new IP-based core network

⋄ 5<sup>th</sup> generation (5G): 2020-?



BSC: Base Station Controller

· Circuit-switched network, designed for voice communication

Slow expensive GSM introduces SMS: small texts can be transferred in signaling channel

- Autentication and Key Agreement (AKA)

Integrity Protection

6.2.2 (U)SIM

- SIM: Subscriber Identity Module International mobile subscriber identity (IMSI) \* Temporary mobile subscriber identity (TMSI): derived from IMSI

\* Typically implemented on a JavaCard

Security token (smartcard) of the network operator

Packet Data

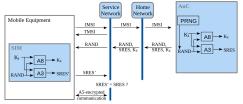
Network (PDN)

- Identifies the network subscriber (customer of the MNO)
- Subscriber kev K; (128 bit)
- Standardized crypto ⇒ roaming possible
- Independent from mobile equipment (ME)
  - \* International Mobile Equipment Identifier (IMEI) identifies the device SIM can be used with any ME (but: SIM-Lock prevents changing SIM for
- USIM: Universal SIM → UMTS (typically implemented in UICC, together with (GSM-)

## 6.2.3 Crypto Algorithms

- · Authentication: A3
  - Variants: COMP128 (broken)
  - Later: COMP128-2 (only 54 key bits), COMP128-3 (64 key bits), COMP128-4
- · Kev Generation: A8
  - Variants: COMP128. COMP128-2. COMP128-3. COMP128-4
- · Encryption: A5 (stream cipher)
  - A5/0 (no encryption), A5/1 (broken), A5/2 (weakened version of A5/1)
  - = Later: A5/3 (KASIIMI = also broken)

#### 6.2.4 Authentication



- · The Service Network sends a RAND (random challenge) to the Mobile Equipment
- · The SIM card computes:
  - SRES': The signed response, using the authentication algorithm A3 and the subscriber key  $(K_i)$
- Kc: The session key for encryption, generated using algorithm A8 and K:
- The Authentication Center (AuC) in the Home Network performs the same computations (using  $K_i$  stored in its database) to generate SRES.
- . The network compares the SRES' received from the ME with its own SRES. If they match, authentication is successful.
- Communication is encrypted using the generated K<sub>C</sub> and the A5 stream cipher.

- · IMSI: International Mobile Subscriber Identity, a unique identifier for the subscriber.
- TMSI: Temporary Mobile Subscriber Identity, a pseudonym for the IMSI used for privacy.
- K<sub>i</sub>: Subscriber key, stored on the SIM and at the network's Authentication Center.
- PRNG: Pseudo-Random Number Generator, used to generate the RAND challenge.

### 6.2.5 IMSI Catcher

- Man-in-the-Middle attack
  - IMSI Catcher: A device (e.g., Stingray) impersonates a legitimate Base Station to intercept communications
  - Step-by-step process:
    - \* The IMSI catcher pretends to be a Base Station with a strong signal, forcing nearby mobile devices to connect.
    - The mobile device sends its IMSI (International Mobile Subscriber Iden tity) to the catcher for authentication.
    - \* The IMSI catcher forwards authentication requests to the real Base Sta tion, acting as a mobile phone.
    - Once connected, the IMSI catcher can:
      - Intercept data and voice traffic.
      - · Disable encryption by negotiating an unencrypted or weaker cipher mode.
      - Track users by capturing their IMSI or location data.
  - This method exploits the lack of mutual authentication in GSM networks (i.e., the mobile phone does not verify the Base Station's authenticity).

## 6.3 UMTS

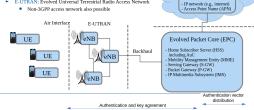
## 6.3.1 Security Improvements

- 3GPP (3rd Generation Partnership Project)
- Responsible for specifications, including issuing 5G standards.
- Old GSM SIM cards can still be used:
  - Backwards compatibility supports GSM authentication protocols.
- Circuit and Packet Switching: Both methods available for communication.
- · Improved Crypto:
  - 3GPP publishes cryptographic algorithms.
  - Encryption is applied not only on the air, enhancing security.
- · New Encryption Algorithms:
  - Initially introduced KASUMI, used for encryption and CBC-MAC.
- Later replaced by more secure algorithms, as KASUMI was broken Mutual Authentication
- Both user and network authenticate each other (not mandatory). · Main Problems:
- - Backwards compatibility creates vulnerabilities
- Fallback to GSM weakens overall security.

## 6.4 LTE- Long Term Evolution

### LTE - Long Term Evolution

- ► UE: User Equipment
- eNB: eNodeB (evolved NodeB) ~ Basestation
- ► E-UTRAN: Evolved Universal Terrestrial Radio Access Network





## 6.4.1 Protocol Stack

## 6.4.2 LTE Protocol Stack



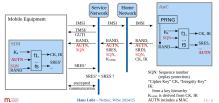
- · Control Plane:
  - Handles signaling, network resource management, and routing.
  - Ensures communication setup and maintenance

#### · User Plane:

- Responsible for transferring user data (e.g., voice, video, and application data). Integrates directly with applications via the TCP/IP stack.
- Key Protocols (Cellular Stack):
  - NAS (Non-Access Stratum):
    - \* Manages mobility and session-related signaling between the UE and the core network (e.g., authentication, location updates).
  - RRC (Radio Resource Control):
  - \* Handles signaling at the radio level, such as connection setup, handovers.
  - and state management. - PDCP (Packet Data Convergence Protocol):
    - \* Provides header compression, encryption, and integrity protection for
  - user and control plane data. RLC (Radio Link Control):

  - \* Ensures data reliability through retransmissions and error correction. MAC (Medium Access Control):
  - \* Manages resource allocation and data multiplexing for multiple UEs.
  - PHY (Physical Laver): Deals with the actual transmission of radio signals over the air interface.
- · Integration with Applications:
  - The LTE stack works alongside the TCP/IP stack for application-level data transfer:
    - Application protocols (e.g., HTTP, VoIP) run on top of TCP/UDP.
    - \* IP ensures routing and addressing across networks.
    - \* PDCP compresses and encrypts packets before passing them to the lower layers for radio transmission

### 6.4.3 Authentication



- · Authentication Process:
  - The Home Network sends a random challenge (RAND), an authentication token (AUTN), and a sequence number (SQN) to the Mobile Equipment (ME).

- The SIM card in the ME uses the subscriber key (K<sub>i</sub>) and predefined functions  $(f 1, f 2, \dots, f 5)$  to compute:
  - SRES': Signed response for authentication
  - CK: Cipher key for encryption.
- IK: Integrity key for data integrity.
- The ME sends SRES' back to the network for verification.
- If SRES' matches the expected SRES, authentication is successful, and encrypted communication begins

#### · Acronyms:

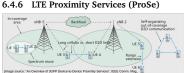
- RAND: Random number used for the challenge.
- AUTN: Authentication token, includes a MAC (Message Authentication Code)
- SQN: Sequence number for replay protection.
- Ki: Subscriber key shared between SIM and the Home Network.
- CK: Cipher key, used for encrypting data.
- IK: Integrity key, used for ensuring data integrity.
- KASME: Key derived from CK and IK, used for session management. Security Features:
- Protects against replay attacks using SQN.
- Ensures mutual authentication with AUTN and SRES.
- Enables encrypted communication using CK.

## 6.4.4 Crypto Algorithms

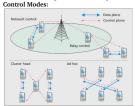
- Authentication Algorithms:
  - f 1, f 2: Message Authentication Codes (MACs), inherited from UMTS.
  - Operator-specific (MNO-dependent); 3GPP recommends AES-based algorith-
- . Key Generation Algorithms (Key Derivation):
  - f 3, f 4, f 5: Used for deriving keys (e.g., K<sub>c</sub>, K<sub>i</sub>), also inherited from UMTS.
  - Operator-specific (MNO-dependent); 3GPP recommends AES-based algorith-

- · Encryption Algorithms:
  - EEA0: "Null cipher" (no encryption). - 128-EEA1: SNOW 3G (stream cipher)
  - 128-EEA2: AES-based encryption algorithm (strong and widely used)
- 128-EEA3: ZUC (Chinese stream cipher). · Integrity Algorithms (EIA):
  - Similar to encryption, used to ensure data integrity during transmission.

- Fact: Mobile Network Operators (MNOs) can always track their users (this is by design).
- Additional Problem: Eavesdropping on radio communication.
  - IMSI: A unique identifier for a user:
    - GSM: Used by the UE only when no TMSI is available
  - \* LTE: IMSI replaced by GUTI (Globally Unique Temporary Identifier). - TMSI/GUTI-
  - \* Can (and should) be updated frequently to avoid tracking by third par-
  - \* Changes must be unpredictable to prevent tracking or identification
  - Research findings \* 19 out of 28 MNOs used very simple, predictable patterns, such as mo
    - notonic counters \* Earlier research found no or very rare changes, allowing tracking.



- ProSe: Cellular Device-to-Device (D2D) Communication
  - Enables communication between devices directly, bypassing base stations.
  - Can operate with or without network coverage
  - Commonly used in public safety networks, such as emergencies or disasters
  - Applications:
    - Peer-to-peer applications.
    - Location-based services
- Range extension through device relays.



- Network-Controlled Mode:
  - \* ProSe communication is managed via the LTE network for reliability and resource allocation.

\* Manages intra-cluster communication and relays data to the LTE network

 Cluster Head Mode: \* A device (cluster head) acts as a local coordinator for nearby devices.

- or other clusters
- Ad Hoc Mode:

  - \* Suitable for completely out-of-coverage scenarios
- Key Management:
  - on the USIM.
  - MIKEY (Multimedia Internet Keying):
    - \* Used for VoIP key exchange.

- . 5G: Represents the next and current generation of mobile networks.
- . Deployment: Has already started, though some specifications are still evolving.
- Performance:

  - Includes advanced Quality-of-Service (QoS) features.
- Industry 4.0: Automation and IoT for manufacturing and logistics.
- Private Networks: Operated by non-traditional operators, such as manufacturing industries.

- AMF (Access and Mobility Management Function): Manages device access and mo-
- anchor point for security in the Serving Network.
- ARPF (Authentication Credential Repository and Processing Function): Stores and
- SEPP (Security Edge Protection Proxy): Secures inter-operator communication (e.g., roaming) by protecting signaling messages exchanged between networks.
- · IMSI SUPI (Subscription Permanent Identifier):
  - SUPI: Replaces IMSI and is never transmitted unencrypted over the air.
  - transmission
- TMSI 5G-S-TMSI: Mandatory to change after paging (e.g., during basestation handover)
- GUTI 5G-GUTI: Requirements for dynamic GUTI changes to improve privacy.
- IMSI Catcher Detection:
- Measurement Reports: Enable network-based and UE-assisted detection.
- · Security in 5G Standalone Mode:
  - Many advanced features are available only in 5G SStandalone"mode. Current usage is mainly 5G NSA (Non-Standalone) for cooperation with 4G

- 6.5.2 Authentication and Key Agreement

  - FAP-AKA'-
  - \* Defined in RFC5448.

  - \* Fits into the EAP (Extensible Authentication Protocol) framework. 5G AKA:
- Privacy Concerns:
  - \* David Basin et al., Ä Formal Analysis of 5G Authentication", ACM CCS 2018. - Practical attacks require targeting specific users: \* Chlosta, Rupprecht, Pöpper, Holz, "5G SUCI-Catchers: Still catching them

all?". ACM WiSec 2021

- · Network Operators and 3rd Parties Offering SServices":
  - Other Services: Additional functionalities provided by the network.
  - TLS Support: Ensures secure communication between entities. OAuth 2.0-based Authorization: Allows secure access control using access to-
  - of Service) properties.
  - Uses FAP-based Authorization for access control

- \* Devices communicate directly without centralized control.
- Authorization: User Equipment (UE) must be authorized unless pre-configured
- - \* Supports different modes (e.g., pre-shared keys or PKI-based).
  - \* ProSe MIKEY Key (PMK) Temporary ProSe Group Key (PGK). \* PGK ProSe Traffic Key (PTK) ProSe Encryption Key (PEK).

## 6.5 5G: The 5th Generation Mobile Network

- Much faster than 4G.
- Provides significantly lower latency and higher reliability. Advancements:
- Builds on software-defined networking principles, initiated with LTE.
- · New Application Scenarios:
  - Device-to-Device Communication: Key for vehicular networks and ProSe.
- 6.5.1 5G Security: Relevant Entities (Functions)
- bility in the Serving Network. . SEAF (Security Anchor Functionality): Handles initial authentication and acts as the
- AUSF (Authentication Server Function): Processes authentication in the Home Network, replacing the HSS from LTE.
- processes authentication keys, like the AuC in LTE.
- - SUCI (Subscription Concealed Identifier): \* SUPI is encrypted using the network's public key to create SUCI for secure
- · Authentication (AKA): Protocol improvements to better protect security features and enhance user pri-
- Makes IMSI/SUCI catchers harder to deploy without being detected.
- Authentication Ontions
  - Rased on pre-shared keys
  - \* Derived from LTE AKA for backward compatibility.
  - Flaws in 5G AKA found via formal protocol analysis:
- 6.5.3 Service-Based Interfaces
  - Network Functions (NF): Core network features exposed as services.
- Network Slices:
  - Provides virtualized and isolated network partitions with specific QoS (Quality

## 7 Firewalls, Intrusion Detection and Prevention

#### 7.1 Firewalls

- · Firewalls separate networks
  - Restricts traffic: enforces policy
  - Logging
  - Implemented in SW and/or HW)

#### 7.1.1 Packet Filtering

- · Packet filtering firewalls:
  - Policies defined by filtering rules (source, destination, port, protocol, etc.).
  - Rules decide actions: accept, block, log, or modify packets. Advanced filters can modify packets (e.g., NAT or port translation).

#### Advantages:

- Cheap, widely available, and efficient.
- Examples: Linux netfilter, Windows firewall.
- Simple implementation; no modification to applications required.

#### Limitations:

- Susceptible to IP and port spoofing attacks.
- Rules limited to individual packets and lack comprehensive state awareness.
- Cannot enforce application-specific filtering: lacks deeper protocol knowledge
- Usually not capable of rejecting specific packets within a particular application

### · Niche Aspects:

- Filters can reject IP spoofing by blocking packets from internal addresses arriving at external interfaces.
- Access control by ports can restrict services to specific source addresses (e.g., SMTP limited to internal IPs).
- Can filter based on packet type (TCP UDP) or flags (e.g., SYN for connection

## 7.1.2 Connection Tracking and Stateful Firewalls

## Connection Tracking

- Tracks packets belonging to the same logical "connection" (not limited to TCP)
- Examples:
  - \* UDP-based connections (e.g., DTLS).
  - \* DNS requests/responses and ICMP echo/replies.
- Supports protocols with separate control and data connections (e.g., active FTP). Enables NAT functionality by rewriting addresses for complex protocols.

#### · Stateful Firewalls:

- Maintain state information about active connections.
- Rules can dynamically adapt based on connection states (e.g., allow replies to outgoing requests).

### 7.1.3 Linux Firewall: Netfilter

Netfilter, part of the Linux kernel, is a framework for packet filtering and network address translation (NAT). It operates using a series of hooks in the network stack where rules, organized into tables and chains, are applied.

### Chains and Their Functions:

- . Input Chain: Handles packets destined for the local system (e.g., filtering SSH traffic).
- . Output Chain: Manages packets generated locally and leaving the system (e.g., limiting outgoing ICMP traffic).
- · Forward Chain: Processes packets passing through the system but not destined for it (e.g., traffic routing in a gateway).
- · Prerouting Chain: Alters incoming packets before routing decisions (e.g., DNAT for
- · Postrouting Chain: Alters outgoing packets after routing decisions (e.g., SNAT for source IP modification).

#### Key Components:

- Tables: Contain chains for specific tasks:
  - Filter Table: Default table for packet filtering (Input, Output, Forward chains). - NAT Table: Used for network address translation (Prerouting, Postrouting
- Mangle Table: For modifying packet headers. · Tools:
  - iptables: Legacy tool for managing rules.
  - nftables: Modern replacement offering simplified syntax and better perfor-

Use Case Example: A packet enters the system through the Prerouting Chain, is routed to the appropriate interface (via Forward Chain if being forwarded or Input Chain if destined locally), and exits through the Postrouting Chain after necessary modifications. Netfilters modularity and flexibility make it a powerful tool for managing network traffic.

## 7.1.4 Application-Level Gateways

An Application-Level Gateway (ALG) acts as an intermediary between clients and applications, offering fine-grained control and monitoring of traffic specific to application protocols. It provides a specialized interface tailored for applications, such as SMTP or HTTP, allowing for filtering, logging, and user profiling at the application layer.

- . Connection Handling: Clients connect to the gateway, which filters and analyzes the traffic before forwarding it to the application server.
- Protocol Awareness: Equipped with detailed knowledge of application-specific protocols, enabling in-depth traffic inspection and filtering.
- . State Management: Tracks the state of connections or application data to enforce security or logging policies effectively.
- In-depth analysis and traffic filtering based on application logic.
- Enables user profiling, attack detection, and intrusion prevention.

- · Provides application-specific caching and potential for secure communication
- App extensions for additional features, such as authentication, without modifying the

- · Performance Overhead: The inspection and filtering process can reduce throughput.
- Maintenance Effort: Requires frequent updates to stay compatible with evolving application protocols
- · Limited Scope: Only supports predefined applications, limiting flexibility in heteroge-

Use Case: ALGs are frequently employed in enterprise environments for managing email servers (e.g., SMTP gateways) or for secure access to critical applications by acting as a filter and

#### 7.1.5 Proxy Firewalls

Proxy firewalls act as a generic proxy on the transport layer, enabling advanced traffic control and enhancing communication security

#### **Kev Features:**

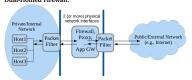
- Connection State Tracking: Monitors and manages the state of connections.
- · Authentication and Security: Adds user authentication and communication security.
  - Frequently used in corporate environments:
    - \* Outgoing traffic is routed only via the proxy.
  - \* User authentication is required before access
  - \* Only specific ports or applications are allowed
- · Flexibility: Can extend functionality toward application-level filtering (e.g., web provies or caches)

#### Comparison to Application-Level Gateways

- · Proxy firewalls are more general than application-level gateways.
- · The distinction between proxy firewalls and application gateways is not always clear.

## Firewall Architectures: Dual-Homed Firewall vs. DMZ

· Dual-Homed Firewall:



- Concept: Firewall with two interfaces separating internal and external networks.
- Operation: Traffic flows through the firewall, using packet filters or proxies for
- Advantages:
  - \* Strong physical isolation.
  - \* Centralized traffic control
- \* Flexible integration of security mechanisms.
- Disadvantages:
  - \* Requires additional hardware resources.
  - Limited scalability for complex networks.
- Use Cases: Suitable for small/medium networks needing strict isolation. Demilitarized Zone (DMZ):

### Firewall, Proxy App GW Host1 Public/External Network (e.g., Internet) Filter Host2 (public) Web Server (public)

- Concept: Segregated network zone for public-facing services (e.g., web servers).
- Operation: Packet filters manage traffic to/from DMZ, preventing direct internal
- Advantages
  - \* Protects internal networks from compromised public-facing services.
  - Granular traffic control for specific services.
  - Limits exposure of internal systems
- Disadvantages:
- Adds configuration and maintenance complexity. - Use Cases: Ideal for hosting public services (e.g., web hosting, e-commerce).
- Key Differences:
  - Dual-Homed focuses on isolation, while DMZ separates public services from in-
  - DMZ adds a dedicated network segment for external-facing servers.

## 7.2 Intrusion Detection and Prevention System (IDPS)

- · Intrusion Detection Systems (IDSs):
  - Detect attempts to attack or intrusions
  - Focused on monitoring and logging suspicious activities.

  - React to detected intrusions to prevent successful attacks.
  - Goal: Stop attacks in progress by blocking or responding.

- IDPS = IDS + Prevention: Combines detection and prevention to ensure proactive and reactive defense mechanisms
- · Historical Work:
  - James Anderson: Computer Security Threat Monitoring and Surveillance (1980).
  - Dorothy Denning: An Intrusion Detection Model (1987)

#### Why Use IDPS?

- Identify incidents effectively.
- Support incident response to minimize damage.
- Identify gaps in security policy and practices.
- · Deter insiders from violating policies.

### 7.2.1 IDPS Functionality (High-Level)

- · Record Events:
  - Logging and accounting for detected activities.
  - Observations useful for detecting and analyzing incidents.
- Integration with Security Information and Event Management (SIEM) systems. Notify Administrators:
  - Alert administrators to take appropriate action
- · Produce Reports:
  - Summarize activities and detected threats for analysis
- Automated First-Level Reaction:
  - Stop attacks or intrusion attempts automatically.
  - Dynamically change the security environment, e.g., reconfigure firewalls.

## 7.2.2 Intrusion Detection Model (Denning, 1987)

- · Components:
  - Subjects: Entities initiating actions.
  - Objects: Resources being accessed or affected.
  - Audit Records: Logs of subject actions for monitoring.
  - Profiles: Behavioral characterizations based on metrics. - Anomaly Records: Highlight abnormal activities.
  - Activity Rules: Define responses to anomalies (e.g., profile updates, intrusion

#### reporting) Metrics:

- Event Counters
- Interval Timers - Resource Measures
- Models:
  - Statistical Models: Identify statistical anomalies
    - \* Mean and Standard Deviation Model
    - Multivariate Correlation Model
    - Markov Process Model
  - Time Series Model
  - Operational Models: Compare metrics against predefined thresholds. - Modern Approaches: Incorporate Machine Learning/AI for advanced anomaly detection

- 7.2.3 Types of IDPS
- · Network-Based Intrusion Detection System (NIDS):
  - Monitors and analyzes network traffic - Often integrated with firewalls (tightly or loosely).
  - Includes specialized Wireless IDPS for wireless networks.
- Incorporates Network Behavior Analysis (NBA) for deeper traffic insights.
- Host-Based Intrusion Detection System (HIDS):
  - Monitors activity on a specific host (server or end-user device). - Analyzes system and application logs, network activity, running processes, and
- system calls. Tracks file accesses and configuration changes for anomaly detection.

- 7.2.4 IDPS Architecture
- Sensors (Agents): Monitor and analyze activity.
- . IDPS Management Server: Collects information from sensors and manages them.
- Database: Serves as a repository for audit records. . Admin Console: Provides a user interface for administrators

# 7.2.5 Detection Methodologies: Categorization

- Signature-Based Intrusion Detection
- Uses fixed patterns to identify known threats (e.g., hash values of malware). Limited to known threats and can be evaded by intelligent attackers.
- Anomaly-Based Intrusion Detection Detects previously unknown threats by modeling normal/abnormal behavior.
- Requires a training /learning phase for data collection and profiling.
- Often uses Machine Learning or AI; main focus of current research Specification-Based Intrusion Detection (Stateful Protocol Analysis)
- Compares observations to models/specifications of legitimate behavior.
- Can be integrated into application-level gateways.
- 7.3 Summary
  - Essential building block for network security.
  - Alone cannot effectively prevent attacks. Various types:
- \* Application-level gateways/proxies. Example: Linux firewall (netfilter, iptables/nftables).
- Intrusion Detection and Prevention Systems (IDPS) - Detect and prevent intrusion attempts

\* Packet filters.

- Types:
  - \* Host-based vs. Network-based IDPS

- Detection methodologies
  - \* Signature-based
  - \* Behavior-based (e.g., anomaly detection).
- Specialized IDPS for specific systems (e.g., industrial automation, vehicle net-