Advanced Network Security - Summary

Lecture 2: Preliminaries

A ***security service*** is a service provided by a protocol layer of *communicating* open systems, which ensures adequate security of the systems or of data transfers. There are six fundamental security services:

1. Authentication
2. Access control
3. Confidentiality
4. Integrity
5. Non-repudiation
6. Availability

Privacy and trust are ***not*** fundamental security services, because:

- In the case of e.g. privacy, the service does not necessarily imply a communication
- In the case of e.g. trust, the definition is not strictly security-related (trust can be a measure of reliability

A ***security algorithm*** is a mathematical procedure applied to secure data. A ***security protocol*** is a sequence of operations providing one or more security services to the data or the communication, *through one or more security algorithms*. A ***security framework*** is a *series of documented processes* that define policies and procedures around the implementation and ongoing management of information security controls.

Passive attackers only listen to the channel to obtain information. Typical passive attacker models include eavesdroppers and the honest-but-curious model. ***Active attackers*** may modify and/or inject traffic as well. The Dolev-Yao, Canetti-Krawczyk and Cyber-Physical Dolev-Yao models are examples of active attacker models.

One-way hashing functions have several properties:

- ***One-wayness***: for a given hash h, it is computationally impossible to obtain the message m from which h was generated.
- ***Weak collision resistance***: given x, it is computationally hard to find $y \neq x$ such that $H(X) = H(y)$.
- ***Strong collision resistance***: it is computationally hard to find any pair of message (x, y) such that $H(x) = H(y)$.

Hashing functions can be used for message integrity verification and for signature generation and verification. However, hashing functions cannot be used to provide confidentiality, as they only allow for verifying already known information (and not for transferring information to another party).

AES has several modes of operation:

- ***Electronic Code Book***: each block is encrypted separately. This allows an attacker to alter the block order, and is prone to cryptanalysis.
- ***Cipher Block Chaining*** *(CBC)*: the previous ciphertext block is used as input for the next block. This makes the encryption dependent on ordering, but leads to error propagation as well.
- ***Cipher Feedback***: a mode where blocks depend on the past. This allows for having different output with the same input. However, this mode is not very fast.
- ***Output Feedback*** *(OFB)*: another mode where blocks depend on the past. However, this mode allows for parallelisation, which makes it much faster than Cipher Feedback mode.

- ***Counter Mode*** *(CTR)*: a mode where a counter is used as an input to the block encryption. While it is very fast and errors propagate to only 1 block of ciphertext, it does require external synchronisation protocols for the counter.

Discrete logarithm-based problems on modular arithmetic can generally be transformed to problems based on elliptic curves. This allows for obtaining a similar hardness with smaller numbers.

Lecture 3: introduction to IoT security

The Internet of Things is a network consisting of connected heterogeneous devices with capabilities to interact with the physical environment. It can also be described as devices which have a little number of functions and specific scopes.

IoT devices have limitations in the following aspects:
- Processing/computational capabilities
- Storage space
- Bandwidth
- Energy

The typical IoT networking architecture consists of 4 tiers:
1. Device tier, which consists of sensors and/or actuators which have wireless communication capabilities, have energy consumption constraints, support only a limited bitrate and exchange small packets.
2. Gateway tier, which manages the IoT network (including security) and arranges for a connection to the internet. Although this tier could be integrated into the IoT device itself, it can also be implemented on e.g. a separate Raspberry Pi.
3. Cloud tier, which provides data pre-processing, analysis and archiving, as well as support for computationally intensive tasks. Such a tier is usually hosted externally, on a large provider's platform.
4. User tier, which is used to request services from IoT devices through the cloud service provider. Different access levels can be provided, and the user tier usually consists of web or mobile apps.

Edge computing performs computation at the edge of the network, close to the user (usually: on the endpoint devices). Fog computing performs computation at nodes/hops/gateways.

The adversary may be located in any of the 4 tiers. This can lead to several different kinds of attacks:

Tier	Attacks
Device tier	Jamming
	Tampering
Gateway tier	Man-in-the-Middle
	Denial of Service
Cloud tier	Data Leakage
	Data Deletion
User tier	Unauthorised Access
	Flooding

A ***protocol suite*** is a set of network protocol layers that work together. A ***protocol stack*** is an implementation of a computer network protocol suite. Whereas the TCP/IP stack is commonly used in traditional networking, there is no single protocol stack for IoT devices. Most IoT devices

are optimised to minimise battery use, which is mostly impacted by use of the radio. Hence, transmission and reception time need to be kept to a minimum. IEEE 802.15.4 defines both a PHY layer and a MAC layer. The PHY layer arranges for modulation of the channel using Offset Quadrature Phase Shift Keying (OQPSK)[1], while the MAC layer orchestrates channel access. IEEE 802.15.4 uses multiple frequencies: 868 MHz for long range communications, and one of 16 channels of size 2 MHz within the 2.4-2.485 GHz spectrum for short range communications. The data rate is approximately 2 Mbps physically; this is roughly 250 kpbs of practical data.

The primary scope of the MAC layer is *reliability*: the limitations of nodes imply that losing a packet generally costs more than preventing the loss from occurring in the first place. A secondary scope of the MAC layer is *efficiency*: due to battery constraints in devices, nodes should communicate at a low energy cost. Several techniques are used to implement these scopes, among which:

- ***Time-Synchronized Channel Hopping*** *(TSCH)*, which is effectively a way of dividing a channel across both time and frequency. Given the channel offset of a given link, the frequency is obtained using a function which depends on the channel offset and the number of (time) slots which have elapsed since the network was deployed.
- Both dedicated and shared links can be used;
- Pseudo-random frequency changes are used;
- Scheduling of transmissions can occur either in a centralised manner or in a distributed manner.

The internet layer has two main functions:

- Addressing, which is handled by 6LoWPAN (or 6Lo) and IPv6.
- Routing, which is usually handled by RPL. Routing needs to take into account the following aspects:
  - Some of the links may be low-power and lossy;
  - Mobile nodes may move throughout the network;
  - Nodes may fail (e.g. due to the battery being drained);
  - Interference may occur.

On the transport layer, IoT devices mostly use UDP, as due to the limited battery capacity, it is hard to maintain stateful connections (i.e. TCP). Re-ordering and sequencing of packets is outsourced to other layers.

On the application layer, many capabilities can be supported. For example, the ***Constrained Application Protocol*** *(CoAP)* is a protocol which provides RESTful services over UDP. ***Message Queuing Transport Telemetry*** *(MQTT)* is a protocol based on the publish-subscribe paradigm.

The following table gives an overview of attacks on different layers:

Layer	Attacks
Physical	Eavesdropping
	Jamming
	Energy depletion
	Side-channel attacks
MAC	Tracking
	Impersonation/Man-in-the-Middle
	Spoofing
	Message manipulation

[1] The technique of Direct Sequence Spread Spectrum (DSSS) is used to mitigate noise.

Layer	Attacks
	Denial of sleep
	Denial of service
Layer 3	Selective forwarding
	Sinkhole
	Hello Flooding
	Wormhole
	Sybil
	Cloning
Application	Tracking
	Private data leakage
	Impersonation/Man-in-the-Middle
	Unauthorised access to nodes/resources
	Code injection
	Malware

There are 5 main issues and constraints related to security in IoT devices:

1. The use of a *shared wireless channel* makes it possible for adversaries to communicate over the channels used by legitimate devices. This necessitates the use of confidentiality and authentication techniques.
2. IoT *deployments are physically accessible*. This makes it reasonably straightforward for an attack to observe and manipulate legitimate devices through e.g. software/hardware tampering, side channel attacks and jamming.
3. IoT devices are *heterogeneous*. That is, different devices may have different kinds of limitations (e.g. processing limitations, bandwidth limitations or energy capacity constraints). Still, security solutions should be able to run on all devices for interoperability to be maintained. This introduces the need for lightweight protocols which interplay with networking protocols. The problem of heterogeneity is also covered by introducing different security levels.
4. Many IoT applications are *designed in an inclusion-driven way*; that is, new devices should be able to join the network relatively easily. This also introduces the possibility for malicious devices to join. Because of this, a balance needs to be found between inclusivity and security.
5. *Securing all wireless links (i.e. the access part) might not suffice*; vulnerabilities in other parts of the network *(the network core)* may lead to issues just the same.

All in all, standardised and well-known approaches might not work in some situations. Furthermore, specific applications require specific security solutions; this implies that a one-size-fits-all solution does not exist.

The following table gives an overview of the classes of approaches for securing IoT:

Technique	Pros	Cons
Adaptation of IT protocols to IoT scenarios *(e.g. compression)*	Security primitives have already been validated and standardised	These protocols cannot always be implemented within constrained devices.
		High bandwidth cost

Technique	Pros	Cons
	Widely accepted protocols	High energy cost
Lightweight ad-hoc solutions	IoT-specific issues/constraints are considered in the design	Many non-standardised solutions
		Unreliable assessment of performance and security
	Optimised energy budget	General reluctance to implement unfamiliar solutions
Multi-level security frameworks *(i.e. change security level/ parameters depending on current conditions)*	Adapts to local/contemporary resource availability	Configuration is left to end-users
	Security has already been validated at maximum configurable level	Vulnerable to downgrade attacks

Lecture 4: IEEE 802.15.4

Protocol structure

The MAC layer has several functions:
1. Channel access
2. Channel allocation
3. Network formation
4. Synchronization
5. Link-layer security

IEEE 802.15.4 is a widely used technology for short-range, low-power communication. It is used in protocol stacks including Zigbee and Thread, and has two reference bandwidths:
1. The first reference bandwidth has a single channel at 868 MHz, and 10 more channels between 902-928 MHz.
2. The second reference bandwidth has 16 channels around 2.4 GHz.

Devices are categorized according to the following two categories:
1. Fully Function Devices (FFD), which can orchestrate the network
2. Reduced Function Devices (RFD), which can only send messages to an FFD.

Supported topologies for an IEEE 802.15.4 network include the star and mesh topologies. Devices are identified by a 64-bit address.

Channel access is 'regulated' based on the concepts of a Contention Access Period (which is based on CSMA-CA) and a Contention Free Period (which is based on coordinator-emitted messages). The rough idea behind CSMA-CA is that, whenever a transmission is attempted, this only goes through when the channel is *(sensed to be*[2]*)* idle. If the channel is busy, then a random backoff timer is started (which increases its time upon further failures).

Time is divided into slots; *"a slot groups the operations necessary to successfully transmit a packet and receive the correspondent ack"*. Scheduling of radio behavior is necessary to save (battery) power; therefore, a Radio Scheduling Table (RST) assigns radio behavior (TX, RX, TX/RX, Sleep) to every time slot and node.

An IEEE 802.15.4 packet has a maximum size of 127 bytes due to constraints at the physical layer. Accounting for the headers in the MAC layer, about 100 bytes remain available for use in the MAC payload.

---

[2] Note that this is not an accurate measurement; due to the hidden terminal problem, packets often do not arrive. For this reason, an acknowledgement (ACK) packet must be sent as well.

Protocol security

Security is optional in IEEE 802.15.4. The main downside of enabling security in this protocol is that it further reduces the available payload size, which means more packets (and hence more energy) are needed to transmit the same amount of data. If enabled, the protocol provides two security services:

1. Data confidentiality
2. Data authenticity

In addition, the protocol provides protection against replay attacks.

IEEE 802.15.4 has 8 security levels:

Security level (in bytes)	**MIC length** (in bits) ***(Message Integrity Code)***	**Provides confidentiality / encryption**	**Provides authenticity**
0 (000)	Not applicable	No	No
1 (001)	32	No	Yes
2 (010)	64	No	Yes
3 (011)	128	No	Yes
4 (100)	Reserved		
5 (101)	32	Yes	Yes
6 (110)	64	Yes	Yes
7 (111)	128	Yes	Yes

Note that the uppermost bit indicates whether encryption is used (value 1) or not (value 0), whereas the lower two bits indicate the length of the message integrity code (MIC). Furthermore, whenever encryption is used, the use of an MIC is mandatory (i.e. security level 4 is not allowed).

There are several methods for key addressing. This can be done explicitly (using KeySource and KeyIndex fields) or implicitly (without using either of those fields). The following key identifier modes exist:

Mode (in bytes)	**Description**
0x00 (00)	“Key is determined implicitly from the originator and recipient(s) of the frame, as indicated in the frame header”
0x01 (01)	“Key is determined from the Key Index field.”
0x02 (10)	“Key is determined explicitly from the 4-octet Key Source field and the Key Index field.”
0x03 (11)	“Key is determined explicitly from the 8-octet Key Source field and the Key Index field.”

CCM* is a cryptosystem based on AES-128 which is used in IEEE 802.15.4. The auxiliary security header is a variable-size header (0 to 14 bytes) which indicates the security level and provides a frame counter and key identifier.

Security operations can be performed in two ‘stages’:

1. MAC-low, in which operations are executed right before transmitting/after receiving a frame. These operations generally cannot involve pre-computations.
2. MAC-high, in which operations independent from radio operations are performed.

Performing security operations must sometimes be done at MAC-low because the authenticated MAC header depends on the sequence number, which in turn depends on whether or not transmissions were successful or not.

Longer time-slots are undesirable because they reduce throughput, increase the time needed to transmit a given packet and allow for less data to be transmitted in the same time. For this reason, it is desirable to quickly process security functionality. As a result, many devices have dedicated hardware to perform cryptographic operations. This saves both time and energy, although many devices still need quite a bit of time to complete operations (meaning time slot lengths often need to be increased).

IEEE 802.15.4 does not address key management; instead, this is delegated to upper layers of the protocol stack. Zigbee has three types of *symmetric* keys:

1. ***Network Key***, which is distributed by a Trust Center (TC) and used for broadcast communications;
2. ***Link Key***, which is used for unicast communications, negotiated at node discovery and typically obtained using a key establishment protocol;
3. ***Master Key***, which is used to make Link Key establishment *(using SSKE: Symmetric-Key Key Establishment)* confidential. It can be installed via key transport, pre-installation or user-entered data.

In table form, we can summarise this as follows:

Key type	Used for	Obtained via
Network key	Broadcast communication	Distributed by Trust Center (TC)
Link key	Unicast communication	Key establishment protocol (at node discovery time)
Master key	Making Link key establishment via SKKE confidential	Key transport, pre-installation or user-entered data (PIN)

An alternative method of key establishment in Zigbee 3.0 is Certificate-Based Key Establishment (CBKE), which is based on compressed, implicit certificates and Elliptic-Curve Qu Vanstone (ECQV) certificates. ECQV certificates have no need for explicit key signatures, which reduce the communication overhead.

In general, research in the scientific literature has different objectives, including:

1. Reduction of bandwidth overhead
2. Reduction of energy consumption
3. Reduction of processing requirements
4. Reduction of storage requirements

Lecture 5: RPL

RPL Packets

Short	Name	Description
DIO	**DODAG Information Object**	
DIS	**DODAG Information Sollicitation**	
DAO	**Destination Advertisement Object**	
DAO-Ack	**Destination Advertisement Object - Acknowledgement**	
CC	**Consistency Check**	

RPL Security header

The RPL security header contains the following fields:

- ***Counter is time*** *(T)*, which defines whether the counter field contains a timestamp;
- ***Algorithm***, which defines the encryption/MAC and signature algorithms to use. Only the value 0 is currently legal, which stands for using *CCM with AES-128* for encryption/MAC and *RSA with SHA-256* for signature;
- ***Key identifier mode*** *(KIM)*, which can be set implicitly or explicitly. This involves a Key Source and Key Index.
- ***Level*** *(LVL)*, which defines the security level. *Only the values 0-3 are legal*, whereas the values 4-7 are currently unassigned. Depending on the KIM's value, the meaning of these values is as follows:

KIM = 0,1,2

LVL	Attributes	MAC length
0	MAC-32	4
1	ENC-MAC-32	4
2	MAC-64	8
3	ENC-MAC-64	8
4-7	Unassigned	N/A

KIM = 3

LVL	Attributes	Signature length
0	Sign-3072	384
1	ENC-Sign-3072	384
2	Sign-2048	256
3	ENC-Sign-2048	256
4-7	Unassigned	N/A

- ***Flags***, which have not been specified further.

Attacks

Attack	Description	Mitigations
Blackhole	Node discards all received packets instead of forwarding them	• Use of disjoint paths • Use of encryption techniques (SF only) • Request ACKs • Send RPL packets to trusted nodes only • IDS/anomaly detection
Selective forwarding (Greyhole)	Node forwards only selected packets and drops the rest	
Sinkhole	Node attracts as much traffic as possible by advertising a better route than others	• Trust-based mechanisms • IDS • Specification-based approaches *(i.e. use of metrics typical of a protocol, such as rank or DODAG version in RPL)*
Wormhole	Two malicious nodes cooperate to create a link where (a part of) all traffic passes through	• Location-based methods • IDS • Specification-based approaches
Sybil	Node uses several different identities in a network. Can be divided into three types: a. Localised b. Sparse c. Mobile	• Keeping track of node ID and geographical info • Trust-based methods • IDS • Specification-based methods
HELLO Flood (DIO Flood)	Node sends HELLO messages with strong routing metrics/strong signal, then disappears or reduces transmission power to normal	Location-based approaches

Attack	Description	Mitigations
Rank	Node manipulates rank, which may affect network topology and the preferred parent selection mechanism. Can be effectuated by changing the advertised rank based on the rank of other nodes or by using a different objective function to decrease the rank of other nodes. This attack can be subdivided into: • ***Increased rank attacks*** • ***Decreased rank attacks*** • ***Worst parent attacks***, in which a node forwards traffic to the worst possible node instead of the best possible one.	IDS
RPL Version	Node sends DIO with higher version number to force global repair, creating topology inconsistencies and routing loops.	• ACK-based (i.e. verifying data on root node) • Trust-based • Specification-based • IDS-based
Local repair	Node sends periodic local repair messages to neighbours without having any problems. Causes recalculation of routes through malicious node, which wastes energy.	• ACK-based • Trust-based • Specification-based
RPL DIS	Node periodically delivers DIS messages using actual or fake IP addresses. Leads to energy waste.	Modify RPL to reduce overhead of new DIS message on a node
Neighbor	Node forwards DIO message to neighbouring nodes *without modification*, which creates illusion of original sender being in range of neighbouring nodes. Leads to selection of out-of-range parent, which increases end-to-end delay and leads to packet loss.	Location-based strategies *(with limited effectiveness)*
Routing table overload	Node sends bogus DAO packets with false routes, which overloads the routing table and prevents legitimate routes from being stored.	No (fully-working) mitigations available, although the attacks only work when RPL is used in storing mode.
Routing table falsification	Injection of false routes to nodes, causing delays and packet losses.	
RPL Replay	Replay of old DIO/DIS/DAO packets later on, which may lead to outdated or false routing entries, degraded routing service, lower packet delivery rate and detachment of the victim's sub-DODAG. *(Note: can also be used with secure modes active.)*	Enabling RPL replay protection mechanism.

Lecture 6: Application & Transport layer

Differences between TLS and DTLS

There are several differences between TLS and DTLS:

1. DTLS uses explicit sequence numbers, while TLS does not (since TCP already 'arranges' the sequence of packets);
2. DTLS allows for receiving messages out of order and silently discarding messages, while TLS closes the connection[3] if a single packet is received out of order (as TCP should have ensured this could not happen);
3. DTLS adds acknowledgement messages for better loss recovery. The acknowledgements are built using other messages;
4. DTLS messages can be fragmented and re-assembled (unlike in TLS, where TCP prevents this from happening);
5. DTLS adds HelloRetryRequest[4] messages, which aim to avoid abuse of datagram transport in ways that support Man-in-the-Middle attacks and Denial-of-Service attacks.

DTLS Stack

Four types of protocols use the record layer in DTLS:

1. The DTLS handshake, which is based on TLS v1.3, with the following changes:
    1. The handshake header is modified to deal with message loss, reordering and fragmentation;
    2. Retransmission timers are introduced to handle message loss;
    3. A new ACK content type has been added to ensure reliable message delivery of handshake messages;
2. ChangeCipherSpec
3. The Alert protocol
4. Application Data Protocols

OSCoAP

Object Security for CoAP (OSCoAP) aims to provide application layer security using a minimal number of packets. To this end, it does not secure the communication session. Instead, the application layer data is secured using pre-exchanged key material. Resources are pre-signed, while encryption is performed using keys dercved from client access secrets.

OSCoAP provides replay protection, while not requiring cipher negotiation. Servers may perform updates offline (without using their radio). Clients perform symmetric decryption and ECDSA verifications, which are more lightweight than a DTLS message exchange.

Attacks against CoAP

Using DTLS or OSCoAP does not prevent any of the following attacks:

1. CoAP (possibly selective) blocking[5] attacks
2. CoAP request delay attacks (since servers do not necessarily reject delayed packets)
3. CoAP relay attacks
    1. Recording packets to then relay them to the server, which gives the idea that the server and client are closer together than they actually are. This can be used to steal cars.

---

[3] Note: this aspect of DTLS can be used to launch a Denial-of-Service attack, either by consuming resources allocated for (spoofed) session maintenance or through traffic amplification after spoofing a source address.

[4] Denial-of-Service attacks are prevented by including a stateless cookie in a HelloRetryRequest message. If such a cookie is received, the client must send a new ClientHello with the cookie added as an extension. The server only proceeds with the handshake if the received cookie is verified to be valid.

[5] Note that the source and destination MAC/IP addresses are visible even when DTLS/OSCoAP is used.

Mitigations include measuring the consistency of the round trip time or the use of signed GPS coordinates.
4. CoAP response delay and mismatch
5. CoAP request fragment re-arrangement

MQTT

MQTT is a publish-subscribe-based protocol. It has three concepts:
1. Brokers, which host lists of topics and the subscribers for those topics. They redirect updates on topics to subscribers;
2. Subscribers, who subscribe to topics of interest and receive updates from topics;
3. Publishers, who publish updates on topics. They may coincide with subscribers.

MQTT-SN is an adaptation of MQTT to constrained networks. It has a reduced message size (maximum 128 bytes instead of 24 MB), it uses UDP (or non-IP solutions) instead of TCP, and it may buffer messages at the server to allow battery-operated devices to sleep.
It is possible to bridge MQTT and MQTT-SN using a gateway.

MQTT can be secured to provide the following security services:
1. Authentication of users and devices, which can either use a username and password or a challenge-response mechanism
2. Authorization of access to server resources, which is done on the broker
3. Integrity of control packets and the application data contained in them, which can be done by including hash values in application messages
4. Confidentiality of control packets and the application data contained in them

Intrusion detection systems may consider many different anomalies related to e.g. connection, authentication or topic scanning. In MQTT-SN, closing the connection based on anomaly detection is not possible due to the use of DTLS.

Lecture 7: jamming

There are several types of generic jamming:

Based on	Jamming type	Description	Detection	Mitigation	Advantages	Disadvantages
Frequency	**Spot jamming**	Focus all transmission power on a single frequency.	Easy: same channel is always noisy	Change channel		
	Sweep jamming	Jam multiple frequencies, one at a time	Harder than spot jamming	Change channel/ wait for frequency to be free	Can affect multiple frequencies	Limited effectiveness, as only one frequency is effected at a time
	Barrage jamming	Jam a range of frequencies at the same time	Easy: channel always has noise	May be hard, if few frequencies remain unjammed.		
Signal	**Noise jamming**	Inject random noise on wireless channel.	Easy if channel is usually quiet; Hard if channel is usually busy.	Impossible if power is sufficient		
	Deceptive jamming	Inject spoofed/ replayed messages on link	Usually hard		Usually hard to detect, may even lead to energy depletion.	Requires previous knowledge on channel
Time	**Constant jamming**	Jam at all times, regardless of time/channel activity.	Easy		Easy to implement, very effective	Inefficient, easy to detect, easy to localise
	Proactive jamming	Randomly select group of frequencies to be jammed per time in advance, only jam those	Could be harder, depending on scenario		More energy-efficient than constant jamming, harder to detect & localise	Does not guarantee success in 100% of cases
	Reactive jamming	Jam only when activity is detected on channel.	Very hard		Very hard to detect & localise; energy-efficient & very effective (depending on scenario)	Hard to realise & expensive due to need for specialised hardware

An attacker may choose their desired type of jamming based on their budget, the scenario, the need for stealthiness[6] and their skills.

Friendly & Cooperative jamming

Friendly jamming is a form of jamming used to aid oneself. Since the jammer is the only one who ca accurately determine when the jamming will occur (in advance), the jammer may share this information with their authorised people so that they (and only they) can communicate in non-jammed intervals.

Cooperative jamming is a form of jamming where jamming is focused in a certain direction (i.e. by positioning an antenna or by using a directional antenna) to prevent an eavesdropper from listening in on benign communication.

Jamming detection strategies

Jamming detection can be based on the following metrics:

Type of metric	Metric	Relative value in jamming situation
Channel Quality Indicator	**Signal-to-Noise Ratio (SNR)**	low
	Noise floor level	high
Traffic Quality Indicator	**Bit Error Rate (BER)**	high
	Packet Delivery Ratio (PDR)	low
	Throughput (in bits/sec)	low

Issues in jamming detection include the following:
- It may be difficult to distinguish between unintentional interference/noise and jamming;
- Even if jamming is easy to detect, it is not necessarily straightforward to avoid.

Jamming detection can be carried out either by the networked devices themselves or by a dedicated device. One major issue with the latter is that such a device would need to communicate with other devices during jamming situations to orchestrate a mitigation.

Jamming avoidance strategies

Direct Sequence Spread Spectrum (DSSS) is a technique which spreads the signal over a much larger bandwidth, but reduces the power level. This makes it more difficult to distinguish the data from random noise.

Antenna polarisation is a technique in which the direction of EM fields is used to naturally ‘filter’ out signals. Downsides include the need for dedicated antennas and synchronisation mechanisms to make nodes agree on the polarisation to use.

Directional transmissions increase the range of antennas and make interference/eavesdropping more difficult, but require dedicated hardware and may lead to more complex MAC-layer protocols and multipath routing.

Frequency Hopping Spread Spectrum (FHSS) is a technique in which the communication channel is varied over time. The transmitter and receiver can coordinate the channel they use through a shared function. *FHSS is similar to TSCH, although FHSS occurs at the PHY-layer, while TSCH occurs at the MAC-layer.*

A way to avoid reactive jamming is to use hardly-detectable communication schemes. For example, by giving packets a postamble instead of a preamble, it becomes more difficult for the jammer to decide when to jam. However, this does lead to increased energy consumption.

---

[6] In e.g. a war situation, the jammer does not want to be localised (and attacked).

An alternative method introduces a side channel which is based on interpreting the existence of jammed signals as a 1-bit.

If jamming only affects a portion of the network, then jamming can be avoided by routing around the jammed portion of the network.

In general, preventing jamming is difficult in IoT scenarios, as hardware-based solutions require (expensive) dedicated hardware, whereas software-based solutions consume too much energy. One approach which might work in practice is to use TSCH to enable FHSS, and then hope that the adversary cannot cover all frequencies at all times.

Lecture 8: BLE and Wi-Fi

Bluetooth Mesh

Bluetooth 5.0 has 8 security fundamentals:

Security fundamental	Description
Encryption and Authentication	All Bluetooth mesh messages are encrypted and authenticated.
Separation of Concerns	Network security, application security and device security are addressed independently.
Area Isolation	A Bluetooth mesh network can be divided into subnets, each cryptographically distinct and secure from the others.
Key Refresh	Security keys can be changed during the life of the Bluetooth mesh network via a Key Refresh procedure.
Message Obfuscation	Message obfuscation makes it difficult to track messages sent within the network, and, as such, provides a privacy mechanism to make it difficult to track nodes.
Replay Attack Prevention	Bluetooth mesh security protects the network against replay attacks.
Trashcan Attack Prevention	Nodes can be removed from the network securely, in a way which prevents trashcan attacks.
Secure Device Provisioning	The process by which devices are added to the Bluetooth mesh network to become nodes is a secure process.

Bluetooth uses the AES-CCM and SALT (s1) cryptographic algorithms. AES-CCM uses an authentication tag of 16 bytes (seemingly), whereas AES-CCM*, which was seen in IEEE 802.15.4 allows authentication tags of multiple sizes.

Bluetooth devices can be addressed using public addresses (based on organization identifiers assigned by IEEE) or random addresses, which can be:
- Static, in which the address does not change until power-cycling the device;
- Resolvable, where the address can be used to derive the long-term address;
- Non-resolvable, which prevents tracking using the address.

Bluetooth message has three types of keys:
1. ***Network Key*** *(NetKey)*: a mandatory key shared with all nodes in the same subnetwork, which is created and distributed during provisioning. It is used to provide network-layer security;
2. ***Device Key*** *(DevKey)*: a mandatory key shared with another Bluetooth device, which is created and distributed during provisioning. It is used to provide link-layer security;
3. ***Application Key*** *(AppKey)*: a mandatory key shared with all devices which support the same application. It is created and distributed by the provisioner through a key-binding process, and is derived by *NetKey*. The application key is used to provide access-layer security (i.e. access control).

Joining a Bluetooth mesh network uses a group-based approach based on the publish-subscribe paradigm. The process in which a new node joins is called provisioning. Provisioning processes are defined as part of the Generic Access Profiles (GAP), and consist of several phases:

1. ***Beaconing***, in which the unassociated Bluetooth device advertises itself in a broadcast advertisement;
2. ***Invitation***, in which the provisioner sends an invite. The newly registering device responds by indicating its provisioning capabilities. These capabilities may cover aspects such as supported elements, a set of security algorithms, out-of-bounds availability of a public key[7], as well as in- and output capabilities;
3. ***Exchanging public keys***, in which ECDH is used to create a secure link key. The link key is then used in AES-128 to exchange secure messages.
4. ***Authentication***, in which the provisioner authenticates the new device. This can be done in any of the following ways:
    1. ***Output OOB***, in which the unprovisioned device outputs a random number.This number then needs to be given as input to the provisioner;
    2. ***Input OOB***, which reverses the roles in output OOB. Thus, the provisioner generates a number, which needs to be given as input to the unprovisioned device;
    3. ***Static OOB*** (also known as ***No OOB***), which both devices generate a random number and continue.

    The authentication phase concludes by performing a ***Confirmation Value Check***, which checks whether the the values exchanged were correctly shared between the two parties. This uses the values exchanged in a sequence of cryptographic algorithms (AES-CCM and SALT).
5. ***Provisioning Data Distribution***, in which the secure link established between the provisioner and the unprovisioned device is used to derive and distribute provisioning data. Most importantly, this includes the network and device keys[8]. Other exchanged parameters include a key index, flags, an IV index (for randomness) and a unicast address to be used by the device.
    Cryptographic parameters are generated and distributed using AES-CCM, which uses the session key, a session nonce and plaintext provisioning data. After the device has obtained the provisioning data (in particular, the network and device keys), the device is ready to join the network.

Bluetooth packets have an MTU of 47 bytes. Their packet payload (including header) may be at most 39 bytes; after removing the 2 bytes for the header, this leaves at most 37 bytes of payload.

Bluetooth defines 2 ***security modes***:

1. Security via Encryption, which can be subdivided into 4 levels:
    1. No security
    2. Unauthenticated pairing with encryption
    3. Authenticated pairing with encryption
    4. Authenticated LE secure connections pairing with encryption
2. Security via Data Signing
    1. Unauthenticated pairing with data signing
    2. Authenticated pairing with data signing

Wi-Fi (IEEE 802.11)

IEEE 802.11 channels overlap in blocks of 4.

An IEEE 802.11 frame has at most 2304 bytes of payload. Together with headers & trailers, the length of a frame may be at most 2338 bytes.

In IEEE 802.11 security, the AAA server is in charge of authentication and authorization. The AAA server is a management entity in which access policies are defined and implemented.

---

[7] Note: this does ***not*** refer to out-of-bound verification of the exchanged key materials. Instead, it refers to out-of-bound availability of the public key of the unprovisioned device, as shared using e.g. a QR code.

[8] This device key is for communication between the provisioner and the now-provisioned device.

IEEE 802.11i specifies the security architecture for IEEE 802.11 LANs. This includes aspects such as authentication, data integrity, data confidentiality, and key management. The WEP protocol contained major weaknesses. The WPA protocol solved most of WEP's issues, while maintaining compatibility with existing hardware. ***Robust Security Network*** (RSN) defines robust mechanisms for authentication, access control and data confidentiality. RSN is the final form of IEEE 802.11.

In RSN, a protocol is used to achieve *mutual authentication* between the user and to generate temporary keys for use on the wireless link between the client and the AP. Access control enforces the use of the authentication function and can be used with a variety of authentication protocols. MAC-level data is encrypted and provided along with a message integrity code that ensures data has not been altered.

The following gives an overview of the services provided by RSN, and the protocols which implement these services:

Service	Protocol
Access Control	IEEE 802.11 Port-based Access Control
Authentication & Key Generation	Extensible Authentication Protocol (EAP)
Confidentiality, Data Origin Authentication, Integrity & Replay Protection	TKIP
	CCMP
Confidentiality	TKIP (RC4)
	CCM (AES-CTR)
	NIST Key Wrap
Integrity & Data Origin Authentication	HMAC-SHA-1
	HMAC-MD5
	TKIP (Michael MIC)
	CCM (AES-CBC-MAC)
Key Generation	HMAC-SHA-1
	RFC 1750

IEEE 802.11 security is only concerned with secure communication between the STA (i.e. client device) and the AP (access point). IEEE 802.11i does ***not*** provide end-to-end security. This can only be provided through additional higher-layer protocols.

IEEE 802.11i has 5 operational phases:

1. The ***discovery phase***, in which a wireless node securely discovers an AP of interest and shares mutual capabilities. The AP uses beacons and probes responses to advertise its presence and security policy. The STA selects cipher suites and authentication mechanisms from the ones proposed by the AP. At the end of the phase, an 802.1x controlled port has been formed, which remains blocked.
2. The ***authentication phase***, in which the STA and AS (authentication server) prove their identities to each other. The AP blocks non-authentication traffic until the authentication transaction is successful, and does not do anything but forwarding the transaction between STA and AS. Authentication is done via the Extensible Authentication Protocol (EAP). IEEE 802.1x distinguished three elements:

1. the ***supplicant***, which is the client wishing to perform authentication
2. the ***authenticator*** *(i.e. access point)*, which receives requests from the supplicant and proxies traffic to the authentication server
3. the ***authentication server*** *(AS)*, which oversees the actual authentication, and may overlap with the authenticator

The protocol is based on a port-based approach. An ***uncontrolled port*** allows the exchanged of packets between supplicant and the AS, regardless of authentication state. A ***controlled port*** allows the exchange of packets between a supplicant and other LAN systems only if authorized by the supplicant's state. The RADIUS protocol is used between the AP and the AS, and the ports are still blocked after the authentication phase.

3. The ***key management phase***, in which a variety of cryptographic keys are generated and distributed to the STA. There are two types of keys: pairwise keys, which protect unicast communication between the STA and AP and are generated during the 4-ways handshake, as well as group keys, which are used for multicast communications. The group keys can be updated during a group key handshake.

In the 4-ways handshake, several messages with nonces and MACs are exchanged to agree on a pairwise temporal key (or PTK). The first message sends a nonce from AP to STA (to ensure freshness). The second message sends another nonce back (to ensure freshness the other way around) and demonstrates the STA is 'online'. The third message demonstrates the AP is 'online', while the fourth message is an ACK. After concluding the 4-wats handshake, the 802.1x controlled port is unblocked.

The group key handshake is straightforward: effectively, the PTK from the (pairwise) 4-ways handshake is used to share a newly established key with the STA.

WPA-Enterprise might have the AS establish a master session key/authentication, authorization and accounting key (MSK/AAA), and send this to the AP; all keys needed by the STA to communicate securely with the AP are derived from this key. WPA-Personal does not use an MSK/AAA.

4. The ***protected data transfer phase***, which aims to protect data exchange. This can be done using TKIP (in WPA) or CCMP (in WPA2). TKIP stands for temporal key integrity protocol, whereas CCMP stands for Counter Mode-CBC MAC Protocol. CCMP is intended for newer devices and provides *message integrity* (using CBC-MAC) and *data confidentiality* (using AES in counter mode with a 128-bit key).
5. The ***connection termination phase***, in which the connection is torn down and restored to the original state.

WPA3 has a more robust authentication process in which the 4-ways handshake is replaced by the Simultaneous Authentication of Equals (SAE) protocol. This protocol provides:

- Natural password selection: it allows users to choose passwords that are easier to remember and enter;
- Ease of use: it delivers enhanced protections without changing the way users connect to a network;
- ***Forward secrecy***: it protects data traffic even if a password is compromised after the data has been transmitted
- It provides robustness against offline dictionary attacks
- It provides enhanced protection to users who choose bad passwords
- The scheme is based on ECDH

Generally speaking, WPA3 improves on the too-short keys used in WPA2.

WPA3 Enterprise provides increased cryptographic robustness while enhancing the network's resiliency (against eavesdropping and forging of WPA2 management frames).

Lecture 9: Side and Covert channels

A ***side channel*** is a 'channel' which leaks (additional) physical information outside of the main information channel. A ***covert channel*** is a side channel which can be controlled and used to serve the controller's purposes. Covert channels can be used both for benign and for malicious purposes: while they can be used to exfiltrate information, they can also be used to communicate despite the presence of an adversary *(e.g. a jammer)*.

To decide which device transmits in which timeslot under reactive jamming, a protocol based on covert channels similar to the following can be used:

1. Assign a timeslot to every device. This should be done out-of-band, i.e. before any transmissions occur;
2. When a device wants to transmit over the channel *(i.e. 'requests to send')*, it sends a message. The time at which this message is sent is then known as timeslot 0;
3. After such a jammed signal is received, all devices send a message in their assigned timeslot, except for the device which wants to transmit over the channel. This causes the channel to be jammed during all timeslots except for the one assigned to the device which requested to send. From this, it can be derived which device wants to send;
4. The protocol concludes by sending an acknowledgement packet after some delay to inform devices of the correct completion of the protocol.

Network traffic analysis is one way to use side channel information for benign purposes. We consider a situation where remote-controlled drones[9] need to be detected in no-fly zones. Variables which can be used in machine learning (ML) approaches to detect drones in traffic include:

- packet interarrival time (i.e. the time between the arrival of two packets);
- packet size

These parameters give a reasonably accurate overview of whether a drone is present, and, if so, what commands are being given to them.
An adversary might[10] make the packet size more consistent or transmit bogus packets to prevent these forms of analysis from working. However, such solutions require more energy and may hence not always be applied successfully.

The time between keystrokes being entered (also known as ***inter-keystroke timings***) can be used to infer a person's identity for authentication or authorization, or to infer the text (e.g. words) they are typing. Although keystroke entries can be detected from the acoustic signals they produce, it is also possible to determine timing from the emission of EM signals from wireless keyboards. Such an attack may work quite well even at distances of up to 10 meters.

While RF antennas can be used to intentionally cause EM emissions, wires in most products are not perfectly shielded, and, as a result, leak a magnetic field. This field depends on hardware features and could, for instance, be used to identify brands of USB drives, or, more specifically, individual USB drives. Such a technique is fully passive and does not need software modification of the target device. However, it does need specific EM antennas for reading out the signal.
By using the time and frequency of the emitted signals, it is possible to accurately distinguish between 15 brands of USB drives. Such a method also works reasonably accurately (94.6%) to distinguish between 15 USB drives of the same brand.
One way to prevent analysis of unintentional EM emissions is to improve the shielding of the device. However, the cost-effectiveness of doing so is limited.

Drone can also be detected using the sound they emit in an analysis of acoustic signals. This allows for determining the weight of the load carried by drones, and leads to accuracies above 90%.

---

[9] We note that communication between the drone and controller is encrypted at the link layer. The adversary may also use masquerading techniques such as a dynamically changing MAC address to hide the drone's identity.

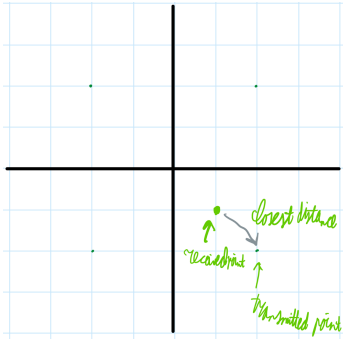
[10] More generally, an adversary would want to make their traffic indistinguishable.

Lecture 10: Global Navigation Satellite Systems (GNSS)

GNSS systems use Medium-Earth Orbit satellites. The main usages of such systems include positioning, navigation and timing. A network of GNSS satellites broadcasts timing and orbital information. In general, the architecture of a GNSS system consists of three segments:

1. Space segment, which consists of satellites;
2. Control segment, which consists of a network of master control, data upload and monitoring stations located across the world and allows operators to control the positions of satellites;
3. User segment, which consists of equipment that receives signals from at least four satellites to determine a position based on the time and orbital location of the satellites.

GNSS systems use modulation across I and Q components, as well as phase shift keying techniques to provide robustness against noise and interference. In effect, this can be seen as similar to an error correction technique where the closest 'allowed' point in a two-dimensional plane is taken to be the measured value.
To synchronize with the signal, GNSS signals provide PRN (pseudo random noise) sequences as part of their output. This also allows devices to identify satellites.
A major problem with GNSS systems is their low bit rate of approximately 50 bits/second.



A graphical interpretation of how error correction may work in GNSS.

Major security issues with GNSS systems originate from the following facts:

1. the power level received by the user is very low, as the transmitter is located very far away;
2. GNSS frames do not contain any security-related fields.

The low power level makes jamming and spoofing viable attacks against a GNSS system. In practice, jamming is difficult to prevent, as most GNSS (hardware) modules[11] do not report errors and simply report the last-seen location instead[12]. Spoofing is also easy; a power ratio of 1.1 suffices to lift the receiver onto the spoofed signal (i.e. emitting a signal which is 10% stronger than the real one will trick the receiver). There are two forms of GNSS spoofing:

1. ***Meaconing***, in which legitimate GNSS signals are captured and re-transmitted after a delay;
2. ***Security Code Estimation and Replay*** *(SCER)*, in which individual satellite signals are forged after a delay. This could modify the position and/or timing computed by the receiver.

It should be noted, though, that launching a successful GNSS attack in practice may be costly, difficult and time-consuming.

In general, there are four categories of security solutions for GNSS threats *(in particular: spoofing)*:

1. ***Cryptographic techniques***, which use symmetric and/or asymmetric encryption to authenticate and encrypt messages. This is mainly done in new constellations, especially those used for military purposes. Known techniques include spreading code encryption, which uses pre-generated code re-used in a random order and uses a master encryption key to encrypt broadcast messages. The encryption key is changed periodically and delivered to each receiver via a pre-shared key. Another technique is Navigation Message Authentication (NMA) and Encryption (NME), which periodically embeds a signature in the message. This allows anyone to read the signature, while certified receivers can authenticate message using a key. A problem for NMA is that only a relatively small portion of the message can be used for the signature. For civil use, the main issue with cryptographic techniques is that they would require *backward compatibility* with old receivers, which is practically impossible to achieve.

---

[11] In addition, hardware modules often compensate for issues in the signal, which obfuscates any information that could be used to detect an attack.

[12] Note that false positives are quite common; for example, signals are regularly lost when a car travels through a tunnel.

2. ***Signal-processing based techniques***, which processes the raw signal from the antenna to san for anomalies in e.g. spatial consistency of satellite locations, Doppler shifts[13,14], power-based metrics, or metrics based on an analysis of the IQ components of signals.
3. ***Correlation with other sources***, which checks the consistency of other sources of location/timing information. Possible source include:
    1. Checking civilian GNSS messages for (phase) consistency with military messages;
    2. Inertial Measurement Units (IMU), which include accelerometers and magnetometers;
    3. Internet Synchronization Protocols such as Network Time Protocol (NTP), Precision Time Protocol (PTP) or eLoran;
    4. Additional networks which broadcast GPS information, such as 4G LTE network, ADS-B or AIS.
4. ***Radio spectrum and antenna-based mitigation schemes***. One such technique uses multiple receiving antennas to detect anomalies. Another technique uses multiple antennas to determine the difference between the angles of arrival of signals, and uses this to determine whether the expected difference in angles matches the measured value. Finally, a technique can be designed based on the information obtained from multiple (moving) antennas, where anomalies can be described in terms of the differences between signals obtained from various antennas.

Lecture 10: Automatic Dependent Surveillance - Broadcast (ADS-B)

Formerly, communication in avionics mostly occurred via radar-based systems, which were costly and inaccurate in situations with high aircraft densities. ADS-B is a communication method which can be split into two parts:

1. ***ADS-B Out***, which is mandatory since 2020 and broadcasts information about the position, speed and altitude of the sending aircraft at a data rate of 1 Mbps;
2. ***ADS-B In***, which might become mandatory in 2025 and allows for receiving (and perhaps forwarding) of information from other aircraft.

There are two competing ADS-B standards:

- Universal Access Transceiver (UAT), which uses the 978 MHz frequency, requires fitting new hardware and is only used in/near Europe and the US;
- Extended Squitter 1090 (ES1090), which uses the 1090 MHz and 1030 MHz frequencies and can be integrated into legacy transponders.

The ES1090 standard has a 112 bit packet size, of which 56 bits can be used for ADS-B data.

Many threats apply to ADS-B systems:

- Eavesdropping;
- Selective jamming may lead to aircraft ‘disappearance’;
- Replay attacks;
- Aircraft may appear in the vicinity of other aircraft, causing sudden high-risk maneuvers;
- Aircraft may be spoofed to appear all around a ground station, which can prevent landings and risk incidents;
- Aircraft may appear in no-fly zones, leading to diplomatic incidents.

ADS-B is insecure because it was designed in the 1980s, with an emphasis on aspects such as reliability, accuracy, range, operational capacity and channel occupancy. In a sense, safety (of people) is considered significantly more important than security (of communications and systems). Securing ADS-B is challenging because of limitations related to the following aspects:

- ***Current network properties***, which are very specific: the traffic is unidirectional, uses a broadcast setup, has opportunistic communication links (in which messages are delivered without connections being established and without guarantee of reception), and has a very limited message size;

---

[13] This can be done by e.g. simulating the positions of satellites and checking whether this is consistent with the received signals.

[14] Note: detecting anomalies in the Doppler shift might not work if the attacker adjusts their signal to the location of the receiver.

- ***Reliability***: as the used modulation technique is not really robust against noise and interference, the percentage of lost messages can raise above 50%, especially for larger distances *(e.g. several hundred kilometers)*;
- ***Cost-effectiveness***: modifications should be integrated without requiring hardware modifications, and should require minimal software updates and maintenance;
- ***Legacy requirements***: designing and deploying ADS-B took 40 years. As a result, deploying a new version would introduce a lot of *(costly and time-consuming)* review;
- Throughput.
- ***Openness***: as the safety of the aircraft is essential, everyone should know where it is. This implies encryption cannot be used and authentication may be desired.

Roughly speaking, there are 4 types of solutions to make ADS-B communication secure/trusted:

1. ***Public key cryptography solutions***, which have a major problem: distributing, updating and managing keys is difficult, while including asymmetric signatures is very difficult in the limited message size. In other words, this solution would require message fragmentation, lead to high overhead and bandwidth use, while encryption can only apply to data, and a loss of compatibility and openness are to be expected when implementing such solutions. All in all, implementing a PKI-based solution is impossible;
2. ***Retroactive key publication strategies***, the main instance of which is the ***TESLA*** protocol *(or μTESLA, which is optimized for constrained environments)*. Their main idea is to choose a random ***master key***, as well as a non-invertible (e.g. hash) function to generate keys from. The final key in the hash chain will then become the ***root key***, which can be published. The symmetric slot keys (i.e. the i^{th} key when counted from the root key) are used to provide a MAC on messages. This MAC can be verified later on when a key is published retroactively, i.e. some time after it is used. The correctness of the key can be verified by applying the hash function until the root key is obtained. Advantages and disadvantages of TESLA include:
   + Security is based on time synchronization, which can be achieved through GPS;
   + The use of symmetric cryptography leads to shorter digests than a PKI-based solution;
   + No complex PKI is required;
   + The protocol can recover from lost/jammed packets *(since the next slot key also chains back to the root key, and can be used to obtain any missed slot keys as well)*;
   - The key chain needs to be re-initialized periodically, which introduces management overhead;
   - A trusted third party is needed to generate and manage master keys. Governmental organizations may ***not*** be willing to assume this position.
3. ***Non-cryptographic physical-layer schemes***, which aim to detect anomalies in known hardware, software, or wireless channel imperfections. One of the downsides of hardware fingerprinting methods is that they may depend on atmospheric conditions. Channel fingerprinting may use characteristics such as received signal strength (RSS), channel impulse response (CIR) or carrier phase. While channel-based techniques can easily be implemented with limited overhead, they often require bidirectional communication, are never tested in highly dynamic environments and suffer from the problem that the channel's features are inconsistent throughout the transmission of a given packet. Finally, the technique of ***uncoordinated frequency hopping***, which is based on frequency hopping spread spectrum (FHSS), improves robustness against eavesdropping and jamming by randomly having both parties select the channel on which they transmit/receive during a given time, and only *(implicitly)* accepting packets if the channels match when a packet is transmitted/received. The main downside of this technique is that it leads to packet loss, unless the sequence of channels to use is exchanged in advance.
4. ***Secure location verification techniques***, which aim to ensure the integrity of air traffic communications by verifying the plausibility of location claims in ADS-B messages. Such techniques can either use additional information on sources or verify the source messages. Example techniques include:
   1. ***Multilateration***, which uses multiple viewpoints to receive messages, and checking whether the signals match based on (physical) properties such as arrival time. This has the disadvantages of being susceptible to messages which arrive from multiple paths (e.g. bouncing against buildings), having decreased accuracy when far away from the source and not giving an accurate picture of the altitude (as most receivers are at similar heights).

On the other hand, advantages of this technique include it being easy to deploy, non-invasive and effective;

2. ***Distance bounding*** techniques, which make use of the fact that signals cannot travel faster than light. Advantages include that secure protocols exist, and the security is based on physical limitations. Major disadvantages include the dependence on mutual communication links *(which ADS-B does not have)*, the fact that precision generally requires multiple protocol iterations *(which is difficult with aircraft moving at high velocity)* and the fact that distance bounding is not very usable over long distances;
3. ***Data fusion and trust management***, which broadly describes techniques which combine several sources of data to check for anomalies. While this does not cause overhead on the transmission side, it does increase cost for the redundant systems needed in the operation and for integrating the features.
4. ***Traffic modeling***, which uses historical flight data to predict future behavior of aircraft. Possible consistency checks include checks for suddenly disappearing aircraft, inconsistent distances, an overly high mobility or an unreasonable density.

Nowadays, data fusion techniques and traffic modeling are most commonly used. In particular, data fusion techniques which consider multi-lateration are used.