

Mohamad Sathak A.J College of Engineering
Department of ECE

EC8702
ADHOCANDWIRELESSSENSOR NETWORKS

UNIT 1
AD HOC NETWORKS – INTRODUCTION AND ROUTING
PROTOCOLS

UNIT 1

INTRODUCTION

CELLULAR AND AD HOC WIRELESS NETWORKS

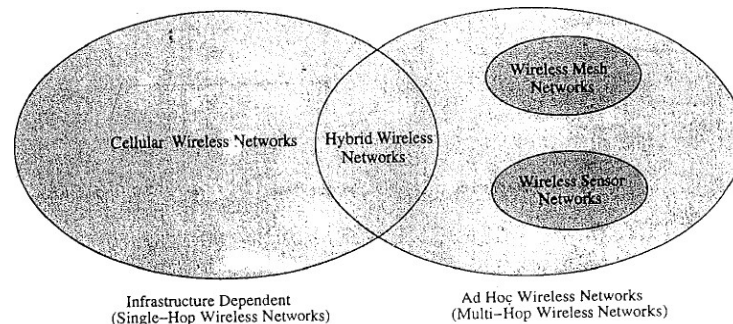


Figure : Cellular and ad hoc wireless networks.

The current cellular wireless networks are classified as the infrastructure dependent network. The path setup for a call between two nodes, say, node C to E, is completed through base station as illustrated in figure below.

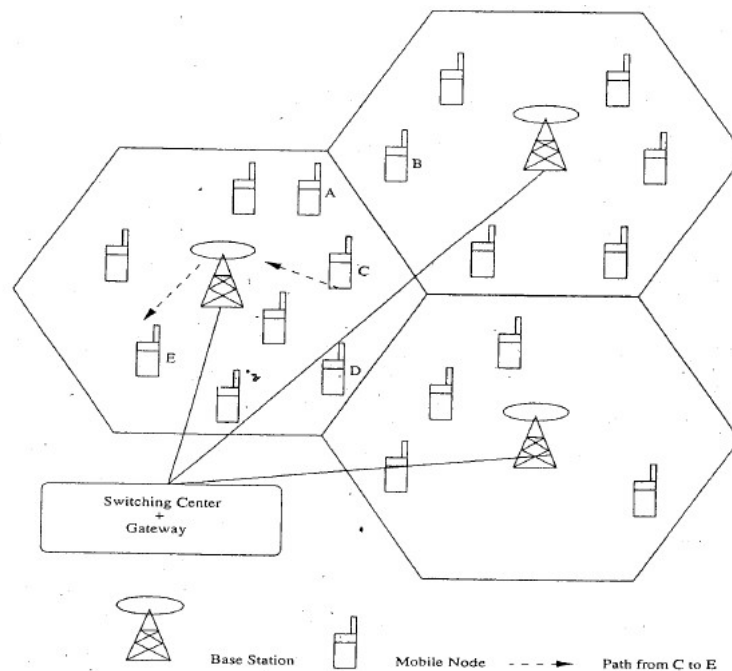


Figure 5.2 A cellular network

- Adhoc wireless networks are defined as a category of wireless network that utilize multi-hop radio replaying and are capable of operating without the support of any fixed infrastructure.
- Absence of any central co-ordinator or base station makes the routing complex.
- Adhoc wireless network topology for the cellular network shown in above figure is illustrated below.
- The path setup for a call between 2 nodes, say, node C to E , is completed through the intermediate mobile node F.
- Wireless mesh network and Wireless sensor networks are specific examples of adhoc wireless networks.

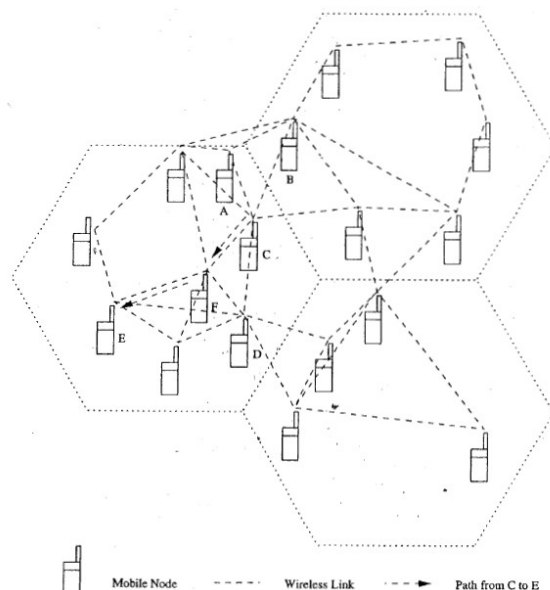


Figure 5.3: An ad hoc wireless network

- The presence of base station simplifies routing and resource management in a cellular network.
- But in adhoc networks, routing and resource management are done in a distributed manner in which all nodes co-ordinate to enable communication among them.

The following table shows the difference between cellular networks and adhoc wireless networks.

Cellular Networks	Ad Hoc Wireless Networks
Fixed infrastructure-based	Infrastructure-less
Single-hop wireless links	Multi-hop wireless links
Guaranteed bandwidth (designed for voice traffic)	Shared radio channel (more suitable for best-effort data traffic)
Centralized routing	Distributed routing
Circuit-switched (evolving toward packet switching)	Packet-switched (evolving toward emulation of circuit switching)
Seamless connectivity (low call drops during handoffs)	Frequent path breaks due to mobility
High cost and time of deployment	Quick and cost-effective deployment
Reuse of frequency spectrum through geographical channel reuse	Dynamic frequency reuse based on carrier sense mechanism
Easier to achieve time synchronization	Time synchronization is difficult and consumes bandwidth
Easier to employ bandwidth reservation	Bandwidth reservation requires complex medium access control protocols
Application domains include mainly civilian and commercial sectors	Application domains include battlefields, emergency search and rescue operations, and collaborative computing
High cost of network maintenance (backup power source, staffing, etc.)	Self-organization and maintenance properties are built into the network
Mobile hosts are of relatively low complexity	Mobile hosts require more intelligence (should have a transceiver as well as routing/switching capability)
Major goals of routing and call admission are to maximize the call acceptance ratio and minimize the call drop ratio	Main aim of routing is to find paths with minimum overhead and also quick reconfiguration of broken paths
Widely deployed and currently in the third generation of evolution	Several issues are to be addressed for successful commercial deployment even though widespread use exists in defense

APPLICATIONS OF AD HOC WIRELESS NETWORKS


Military Application

- Adhoc wireless networks can be very useful in establishing communication among a group of soldiers for tactical operations.
- Setting up of a fixed infrastructure for communication among group of soldiers in enemy territories or in inhospitable terrains may not be possible.
- In such a case, adhoc wireless networks provide required communication mechanism quickly.
- The primary nature of the communication required in a military environment enforces certain important requirements on adhoc wireless networks namely, Reliability, Efficiency, Secure communication & Support for multicast routing.

Collaborative & Distributed computing

- Adhoc wireless network helps in collaborative computing, by establishing temporary communication infrastructure for quick communication with minimal configuration among a group of people in a conference.
- In distributed file sharing application reliability is of high importance which would be provided by adhoc network.
- Other applications such as streaming of multimedia objects among participating nodes in ad hoc wireless networks require support for soft real-time communication
- Devices used for such applications could typically be laptops with add-on wireless interface cards, enhanced personal digital assistants (PDAs) or mobile devices with high processing power

Emergency Operations

- Ad hoc wireless networks are very useful in emergency operations such as search and rescue, crowd control and commando operations
- The major factors that favour ad hoc wireless networks for such tasks are  self-configuration of system with minimal overhead, independent of fixed or centralised infrastructure, the freedom and flexibility of mobility, and unavailability of conventional communication infrastructure.
- In environments, where the conventional infrastructure based communication facilities are destroyed due to a war or due to natural calamities, immediate deployment of adhoc wireless networks would be a good solution for co-ordinating rescue activities.
- They require minimum initial network configuration with very little or no delay

Wireless Mesh Network

- Wireless mesh networks are adhoc wireless network that are formed to provide an alternate communication infrastructure for mobile or fixed nodes/users, without the spectrum reuse constraint & requirement of network planning of cellular network.
- It provides many alternate paths for a data transfer session between a source & destination, resulting in quick reconfiguration of the path when the existing path fails due to node failure.
- Since the infrastructure built is in the form of small radio relaying devices, the investment required in wireless mesh networks is much less than what is required for the cellular network counterpart.
- The possible deployment scenarios of wireless mesh networks include: residential zones, highways, business zones, important civilian regions and university campuses
- Wireless mesh networks should be capable of self-organization and maintenance.
- It operates at license-free ISM band around 2.4 GHz & 5 GHz.
- It is scaled well to provide support to large number of points.
- Major advantage is the support for a high data rate, quick & low cost of deployment, enhanced services, high scalability, easy extendability, high availability & low cost per bit.

Wireless Sensor Networks:

- Sensor networks are special category of Adhoc wireless network that are used to provide a wireless communication infrastructure among the sensors deployed in a specific application domain.

- Sensor nodes are tiny devices that have capability of sensing physical parameters processing the data gathered, & communication to the monitoring system.
- The issue that make sensor network a distinct category of adhoc wireless network are the following:

Mobility of nodes:

- ✓ Mobility of nodes is not a mandatory requirement in sensor networks.
- ✓ For example, the nodes used for periodic monitoring of soil properties are not required to be mobile & the nodes that are fitted on the bodies of patients in a post-surgery ward of a hospital are designed to support limited or partial mobility.
- ✓ In general, sensor networks need not in all cases be designed to support mobility of sensor nodes.

Size of the network :

- ✓ The number of nodes in sensor network can be much larger than that in a typical ad hoc wireless network.

Density of deployment :

- ✓ The density of nodes in a sensor network varies with the domain of application.
- ✓ For example, Military applications require high availability of the network, making redundancy a high priority.

Power constraints :

- ✓ The power constraints in sensor networks are much more stringent than those in ad hoc wireless networks. This is mainly because the sensor nodes are expected to operate in harsh environmental or geographical conditions, with minimum or no human supervision and maintenance.
- ✓ In certain case, the recharging of the energy source is impossible.
- ✓ Running such a network, with nodes powered by a battery source with limited energy, demands very efficient protocol at network, data link, and physical layer.
- ✓ The power sources used in sensor networks can be classified into the following 3 categories:
 - *Replenishable Power source*: The power source can be replaced when the existing source is fully drained.
 - *Non-replenishable Power source*: The power source cannot be replenished once the network has been deployed. The replacement of sensor node is the only solution.
 - *Regenerative Power source*: Here, Power source employed in sensor network have the capability of regenerating power from the physical parameter under measurement.

Data / Information fusion :

- ✓ Data fusion refers to the aggregation of multiple packets into one before relaying it.
- ✓ Data fusion mainly aims at reducing the bandwidth consumed by redundant headers of the packets and reducing the media access delay involved in transmitting multiple packets.
- ✓ Information fusion aims at processing the sensed data at the intermediate nodes and relaying the outcome to the monitor node.

Traffic Distribution :

- ✓ The communication traffic pattern varies with the domain of application in sensor networks.
- ✓ For example, the environmental sensing application generates short periodic packets indicating the status of the environmental parameter under observation to a central monitoring station.
- ✓ This kind of traffic requires low bandwidth.
- ✓ Ad hoc wireless networks generally carry user traffic such as digitized & packetized voice stream or data traffic, which demands higher bandwidth.

Hybrid Wireless Networks

- One of the major application area of ad hoc wireless network is in the hybrid wireless architecture such as Multi-hop Cellular Network [MCN] & Integrated Cellular Adhoc Relay [iCAR].
- The primary concept behind cellular networks is geographical channel reuse.

- Several techniques like cell sectoring, cell resizing and multi tier cells increase the capacity of cellular networks.
- MCNs combine the reliability & support of fixed base station of cellular network with flexibility & multi-hop relaying adhoc wireless networks.
- Major advantages are as follows:
 - Higher capacity than cellular networks due to the better channel reuse.
 - Increased flexibility & reliability in routing.
 - Better coverage & connectivity in holes of a cell can be provided by means of multiple hops through intermediate nodes in a cell.

ISSUES IN AD HOC WIRELESS NETWORKS

The major issues that affect the design, deployment, & performance of an ad hoc wireless network system are:

- ♥ Medium Access Scheme.
- ♥ Transport Layer Protocol.
- ♥ Routing.
- ♥ Multicasting.
- ♥ Energy Management.
- ♥ Self-Organisation.
- ♥ Security.
- ♥ Addressing & Service discovery.
- ♥ Deployment considerations.
- ♥ Scalability.
- ♥ Pricing Scheme.
- ♥ Quality of Service Provisioning

Medium Access Scheme

The primary responsibility of a Medium Access Control (MAC) protocol in adhoc wireless networks is the distributed arbitration for the shared channel for transmission of packets. The major issues to be considered in designing a MAC protocol for adhoc wireless networks are as follows:

1. **Distributed Operation:**
 - The ad hoc wireless networks need to operate in environments where no centralized coordination is possible.
 - The MAC protocol design should be fully distributed involving minimum control overhead.
2. **Synchronization:**
 - The MAC protocol design should take into account the requirement of time synchronization.
 - Synchronization is mandatory for TDMA-based systems for management of transmission and reception slots.
3. **Hidden Terminals:**
 - Hidden terminals are nodes that are hidden (or not reachable) from the sender of a data transmission session, but are reachable to the receiver of the session.
4. **Exposed terminals:**
 - Exposed terminals, the nodes that are in the transmission range of the sender of an on-going session, are prevented from making a transmission.
5. **Throughput:**
 - The MAC protocol employed in adhoc wireless networks should attempt to maximize the throughput of the system.
 - The important considerations for throughput enhancement are
 - Minimizing the occurrence of collisions.
 - Maximizing channel utilization and
 - Minimizing control overhead.
6. **Access delay:**
 - The average delay that any packet experiences to get transmitted.

- The MAC protocol should attempt to minimize the delay.

7. **Fairness:**

- Fairness refers to the ability of the MAC protocol to provide an equal share or weighted share of the bandwidth to all competing nodes.
- Fairness can be either *node-based* or *flow-based*.

8. **Real-time Traffic support:**

- In a contention-based channel access environment, without any central coordination, with limited bandwidth, and with location-dependent contention, supporting time-sensitive traffic such as voice, video, and real-time data requires explicit support from the MAC protocol.

9. **Resource reservation:**

- The provisioning of QoS defined by parameters such as bandwidth, delay, and jitter requires reservation of resources such as *bandwidth*, *buffer space*, and *processing power*.

10. **Ability to measure resource availability:**

- In order to handle the resources such as bandwidth efficiently and perform call admission control based on their availability, the MAC protocol should be able to provide an estimation of resource availability at every node.
- This can also be used for making *cogestion control decisions*.

11. **Capability for power control:**

- The transmission power control reduces the energy consumption at the nodes, causes a decrease in interference at neighboring nodes, and increases frequency reuse.

12. **Adaptive rate control:**

- This refers to the variation in the data bit rate achieved over a channel.
- A MAC protocol that has adaptive rate control can make use of a high data rate when the sender and receiver are nearby & adaptively reduce the data rate as they move away from each other.

13. **Use of directional antennas:**

- This has many advantages that include
 - Increased spectrum reuse.
 - Reduction in interference and
 - Reduced power consumption.

Routing

The responsibilities of a routing protocol include exchanging the route information; finding a feasible path to a destination. The major challenges that a routing protocol faces are as follows:

1. **Mobility :**

- The Mobility of nodes results in frequent path breaks, packet collisions, transient loops, stale routing information, and difficulty in resource reservation.

2. **Bandwidth constraint :**

- Since the channel is shared by all nodes in the broadcast region, the bandwidth available per wireless link depends on the number of nodes & traffic they handle.

3. **Error-prone and shared channel :**

- The Bit Error Rate (BER) in a wireless channel is very high [10^{-5} to 10^{-3}] compared to that in its wired counterparts [10^{-12} to 10^{-9}].
- Consideration of the state of the wireless link, signal-to-noise ratio, and path loss for routing in ad hoc wireless networks can improve the efficiency of the routing protocol.

4. **Location-dependent contention :**

- The load on the wireless channel varies with the number of nodes present in a given geographical region.
- This makes the contention for the channel high when the number of nodes increases.
- The high contention for the channel results in a high number of collisions & a subsequent wastage of bandwidth.

5. **Other resource constraints :**

- The constraints on resources such as computing power, battery power, and buffer storage also limit the capability of a routing protocol.

The major requirements of a routing protocol in adhoc wireless networks are the following.

1. **Minimum route acquisition delay :**
 - The route acquisition delay for a node that does not have a route to a particular destination node should be as minimal as possible.
 - The delay may vary with the size of the network and the network load.
2. **Quick route reconfiguration :**
 - The unpredictable changes in the topology of the network require that the routing protocol be able to quickly perform route reconfiguration in order to handle path breaks and subsequent packet losses.
3. **Loop-free routing :**
 - This is a fundamental requirement to avoid unnecessary wastage of network bandwidth.
 - In adhoc wireless networks, due to the random movement of nodes, transient loops may form in the route thus established.
 - A routing protocol should detect such transient routing loops & take corrective actions.
4. **Distributed routing approach :**
 - An adhoc wireless network is a fully distributed wireless network & the use of centralized routing approaches in such a network may consume a large amount of bandwidth.
5. **Minimum control overhead :**
 - The control packets exchanged for finding a new route, and maintaining existing routes should be kept as minimal as possible.
6. **Scalability :**
 - Scalability is the ability of the routing protocol to scale well in a network with a large number of nodes.
 - This requires minimization of control overhead & adaptation of the routing protocol to the network size.
7. **Provisioning of QoS:**
 - The routing protocol should be able to provide a certain level of QoS as demanded by the nodes or the category of calls.
 - The QoS parameters can be bandwidth, delay, jitter, packet delivery ratio, & throughput.
8. **Support for time-sensitive traffic :**
 - Tactical communications & similar applications require support for time-sensitive traffic.
 - The routing protocol should be able to support both hard real-time & soft real-time traffic.
9. **Security and privacy :**
 - The routing protocol in adhoc wireless networks must be resilient to threats and vulnerabilities.
 - It must have inbuilt capability to avoid resource consumption, denial-of-service, impersonation, and similar attacks possible against an ad hoc wireless network.

Multicasting

It plays important role in emergency search & rescue operations & in military communication. Use of single-link connectivity among the nodes in a multicast group results in a tree-shaped multicast routing topology. Such a tree-shaped topology provides high multicast efficiency, with low packet delivery ratio due to the frequency tree breaks. The major issues in designing multicast routing protocols are as follows:

1. **Robustness :**
 - The multicast routing protocol must be able to recover & reconfigure quickly from potential mobility-induced link breaks thus making it suitable for use in high dynamic environments.
2. **Efficiency :**
 - A multicast protocol should make a minimum number of transmissions to deliver a data packet to all the group members.
3. **Control overhead :**
 - The scarce bandwidth availability in ad hoc wireless networks demands minimal control overhead for the multicast session.
4. **Quality of Service :**

- QoS support is essential in multicast routing because, in most cases, the data transferred in a multicast session is time-sensitive.

5. **Efficient group management :**

- Group management refers to the process of accepting multicast session members and maintaining the connectivity among them until the session expires.

6 . **Scalability :**

- The multicast routing protocol should be able to scale for a network with a large number of nodes

7. **Security :**

- Authentication of session members and prevention of non-members from gaining unauthorized information play a major role in military communications.

Transport Layer Protocol

- The main objectives of the transport layer protocols include :
 - ✓ Setting up & maintaining end-to-end connections,
 - ✓ Reliable end-to-end delivery of packets,
 - ✓ Flow control &
 - ✓ Congestion control.

Examples of some transport layer protocols are,

a. **UDP (User Datagram Protocol) :**

- It is an unreliable connectionless transport layer protocol.
- It neither performs flow control & congestion control.
- It does not take into account the current network status such as congestion at the intermediate links, the rate of collision, or other similar factors affecting the network throughput.

b. **TCP (Transmission Control Protocol):**

- It is a reliable connection-oriented transport layer protocol.
- It performs flow control & congestion control.
- Here performance degradation arises due to frequent path breaks, presence of stale routing information, high channel error rate, and frequent network partitions.

Pricing Scheme

- Assume that an optimal route from node A to node B passes through node C, & node C is not powered on.
- Then node A will have to set up a costlier & non-optimal route to B.
- The non-optimal path consumes more resources & affects the throughput of the system.
- As the intermediate nodes in a path that relay the data packets expend their resources such as battery charge & computing power, they should be properly compensated.
- Hence, pricing schemes that incorporate service compensation or service reimbursement are required.

Quality of Service Provisioning (QoS)

- QoS is the performance level of services offered by a service provider or a network to the user.
- QoS provisioning often requires ,
 - ✓ Negotiation between host & the network.
 - ✓ Resource reservation schemes.
 - ✓ Priority scheduling &
 - ✓ Call admission control.

• **QoS parameters :**

Applications	Corresponding QoS parameter
1. Multimedia application	1. Bandwidth & Delay.
2. Military application	2. Security & Reliability.
3. Defense application	3. Finding trustworthy intermediate hosts & routing.

4. Emergency search and rescue operations	4. Availability.
5. Hybrid wireless network	5. Maximum available link life, delay, bandwidth & channel utilization.
6. communication among the nodes in a sensor network	6. Minimum energy consumption, battery life & energy conservation

- **QoS-aware routing :**

- i. Finding the path is the first step toward a QoS-aware routing protocol.
- ii. The parameters that can be considered for routing decisions are,
 - Network throughput.
 - Packet delivery ratio.
 - Reliability.
 - Delay.
 - Delay jitter.
 - Packet loss rate.
 - Bit error rate.
 - Path loss.

- **QoS framework :**

- I. A framework for QoS is a complete system that attempts to provide the promised services to each user or application.
- II. The key component of QoS framework is a QoS service model which defines the way user requirements are served.

Self-Organization

- One very important property that an ad hoc wireless network should exhibit is organizing & maintaining the network by itself.
- The major activities that an ad hoc wireless network is required to perform for self-organization are,
 - ✓ Neighbour discovery.
 - ✓ Topology organization &
 - ✓ Topology reorganization (updating topology information)

Security

- 1) Security is an important issue in ad hoc wireless network as the information can be hacked.
- 2) Attacks against network are of 2 types :
 - I. **Passive attack**—Made by malicious node to obtain information transacted in the network without disrupting the operation.
 - II. **Active attack**—They disrupt the operation of network.

Further active attacks are of 2 types :

 - **External attack:** The active attacks that are executed by nodes outside the network.
 - **Internal attack:** The active attacks that are performed by nodes belonging to the same network.
- 3) The major security threats that exist in ad hoc wireless networks are as follows :
 - ✉ **Denial of service** - The attack affected by making the network resource unavailable for service to the nodes, either by consuming the bandwidth or by overloading the system.
 - ✉ **Resource consumption** - The scarce availability of resources in ad hoc wireless network makes it a easy target for internal attacks, particularly aiming at consuming resources available in the network.

The major types of resource consumption attacks are,

 - ✓ **Energy depletion :**
 - Highly constrained by the energy source
 - Aimed at depleting the battery power of critical nodes.

- ✓ Buffer overflow :
 - Carried out either by filling the routing table with unwanted routing entries or by consuming the data packet buffer space with unwanted data.
 - Lead to a large number of data packets being dropped, leading to the loss of critical information.

✉ **Host impersonation** - A compromised internal node can act as another node and respond with appropriate control packets to create wrong route entries, and can terminate the traffic meant for the intended destination node.

✉ **Information disclosure** - A compromised node can act as an informer by deliberate disclosure of confidential information to unauthorized nodes.

✉ **Interference** - A common attack in defense applications to jam the wireless communication by creating a wide spectrum noise.

Addressing and service discovery

- Addressing & service discovery assume significance in ad hoc wireless network due to the absence of any centralised coordinator.
- An address that is globally unique in the connected part of the ad hoc wireless network is required for a node in order to participate in communication.
- Auto-configuration of addresses is required to allocate non-duplicate addresses to the nodes.

Energy Management

- Energy management is defined as the process of managing the sources & consumers of energy in a node or in the network for enhancing the lifetime of a network.
- Features of energy management are :
 - Shaping the energy discharge pattern of a node's battery to enhance battery life.
 - Finding routes that consumes minimum energy.
 - Using distributed scheduling schemes to improve battery life.
 - Handling the processor & interface devices to minimize power consumption.
- Energy management can be classified into the following categories :
 - a. **Transmission power management :**
 - The power consumed by the Radio Frequency (RF) module of a mobile node is determined by several factors such as
 - * The state of operation.
 - * The transmission power and
 - * The technology used for the RF circuitry.
 - The state of operation refers to transmit, receive, and sleep modes of the operation.
 - The transmission power is determined by
 - * Reachability requirement of the network.
 - * Routing protocol and
 - * MAC protocol employed.
 - b. **Battery energy management :**
 - The battery management is aimed at extending the battery life of a node by taking advantage of its chemical properties, discharge patterns, and by the selection of a battery from a set of batteries that is available for redundancy.
 - c. **Processor power management :**
 - The clock speed and the number of instructions executed per unit time are some of the processor parameters that affect power consumption.
 - The CPU can be put into different power saving modes during low processing load conditions.
 - The CPU power can be completely turned off if the machine is idle for a long time. In such a case, interrupts can be used to turn on the CPU upon detection of user interaction or other events.
 - d. **Devices power management :**
 - Intelligent device management can reduce power consumption of a mobile node significantly.

- This can be done by the operating system(OS) by selectively powering down interface devices that are not used or by putting devices into different power saving modes, depending on their usage.

Scalability

- Scalability is the ability of the routing protocol to scale well in a network with a large number of nodes.
- It requires minimization of control overhead & adaptation of the routing protocol to the network size.

Deployment Considerations

The deployment of a commercial ad hoc wireless network has the following benefits when compared to wired networks

a) **Low cost of deployment :**

- The use of multi-hop wireless relaying eliminates the requirement of cables & maintenance in deployment of communication infrastructure.
- The cost involved is much lower than that of wired networks.

b) **Incremental deployment :**

- Deployment can be performed incrementally over geographical regions of the city.
- The deployed part of the network starts functioning immediately after the minimum configuration is done.

c) **Short deployment time :**

- Compared to wired networks, the deployment time is considerably less due to the absence of any wired links.

d) **Reconfigurability :**

- The cost involved in reconfiguring a wired network covering a Metropolitan Area Network(MAN) is very high compared to that of an ad hoc wireless network covering the same service area.

The following are the major issues to be considered in deploying an ad hoc wireless network :

a) **Scenario of deployment:**

- The scenario of deployment has significance because the capability required for a mobile node varies with the environment in which it is used.
- The following are some of the different scenarios in which the deployment issues vary widely :
 - **military deployment :**
It can be either,
 - ✓ Data-centric network : Handle a different pattern of data traffic & can be partially comprised of static nodes.
Eg : a wireless sensor network.
 - ✓ User-centric network: Consists of highly mobile nodes with or without any support from any infrastructure.
Eg : soldiers or armored vehicles carrying soldiers equipped with wireless communication devices.
 - **Emergency operations deployment :**
 - Demands a quick deployment of rescue personnel equipped with hand-held communication equipment.
 - The network should provide support for time-sensitive traffic such as voice & video.
 - Short data messaging can be used in case the resource constraints do not permit voice communication.
 - **Commercial wide-area deployment :**
 - Eg : wireless mesh networks.
 - The aim of the deployment is to provide an alternate communication infrastructure for wireless communication in urban areas & areas where a traditional cellular base station cannot handle the traffic volume.
 - **Home network deployment :**

- Deployment needs to consider the limited range of the devices that are to be connected by the network.
- Eg : short transmission range avoid network partitions.

b) Required longevity of network :

- If the network is required for a short while, battery-powered mobile nodes can be used.
- If the connectivity is required for a longer duration of time, fixed radio relaying equipment with regenerative power sources can be deployed.

c) Area of coverage :

- Determined by the nature of application for which the network is set up.
- Eg : the home area network is limited to the surroundings of a home.
- The mobile nodes' capabilities such as the transmission range & associated hardware, software, & power source should match the area of coverage required.

d) Service availability :

- Defined as the ability of an ad hoc wireless network to provide service even with the failure of certain nodes.
- Has significance in a Fully mobile ad hoc wireless network used for tactical communication & in partially fixed ad hoc wireless network used in commercial communication infrastructure such as wireless mesh networks.

e) Operational integration with other infrastructure :

- Considered for improving the performance or gathering additional information, or for providing better QoS.
- In military environment, integration of ad hoc wireless networks with satellite networks or unmanned aerial vehicles(UAVs) improves the capability of the ad hoc wireless networks.

f) Choice of protocol :

- The choice of protocols at different layers of the protocol stack is to be done taking into consideration the deployment scenario.
- A TDMA-based & insecure MAC protocol may not be the best suited compared to a CDMA-based MAC protocol for a military application.

AD HOC WIRELESS INTERNET

- Ad hoc wireless internet extends the services of the internet to the end users over an ad hoc wireless network.
- Some of the applications of ad hoc wireless internet are :
 - ✓ Wireless mesh network.
 - ✓ Provisioning of temporary internet services to major conference venues.
 - ✓ Sports venues.
 - ✓ Temporary military settlements.
 - ✓ Battlefields &
 - ✓ Broadband internet services in rural regions.
- The major issues to be considered for a successful ad hoc wireless internet are the following :
 - ❖ **Gateway :**
 - They are the entry points to the wired internet.
 - Generally owned & operated by a service provider.
 - They perform following tasks ,
 - Keeping track of end users.
 - Bandwidth management.
 - Load balancing.
 - Traffic shaping.
 - Packet filtering.
 - Width fairness &
 - Address, service & location discovery.

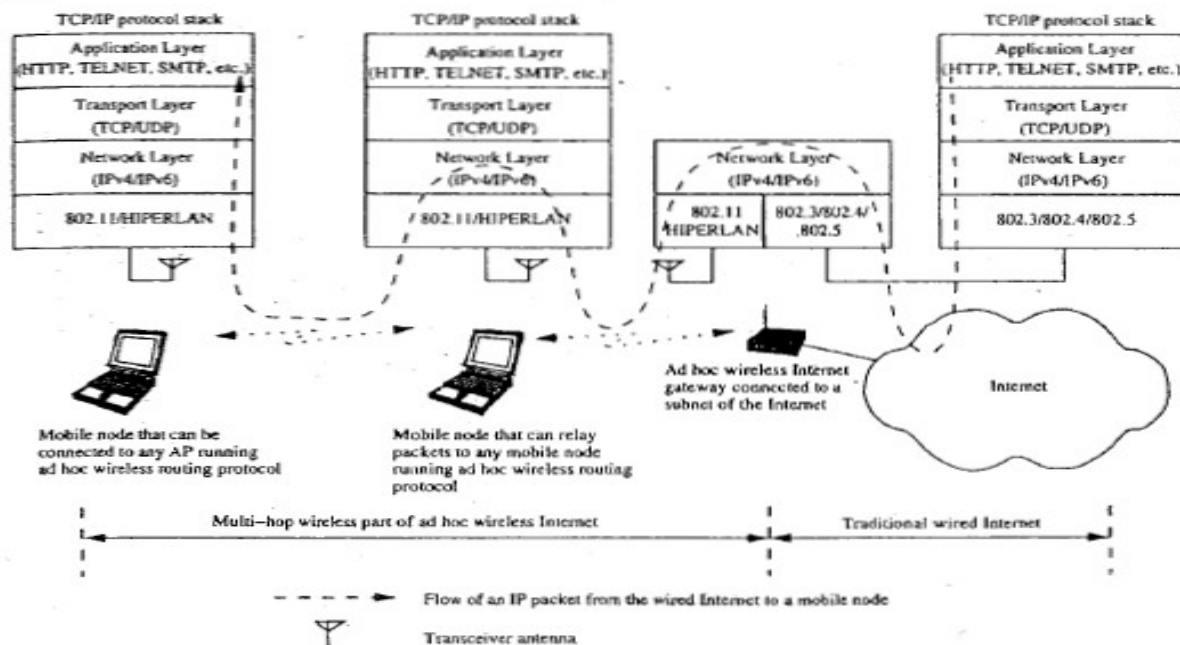


Figure 5.7 Schematic diagram of ad hoc wireless internet

❖ **Address mobility:**

- This problem is worse here as the nodes operate over multiple wireless hops.
- Solution such as Mobile IP can provide temporary alternative.

❖ **Routing:**

- It is a major problem in ad hoc wireless internet, due to dynamic topological changes, the presence of gateways, multi-hop relaying, & the hybrid character of the network.
- Possible solution is to use separate routing protocol for the wireless part of ad hoc wireless internet.

❖ **Transport layer protocol:**

- Several factors are to be considered here, the major one being the state maintenance overhead at the gateway nodes.

❖ **Load balancing:**

- They are essential to distribute the load so as to avoid the situation where the gateway nodes become bottleneck nodes.

❖ **Pricing / Billing:**

- Since internet bandwidth is expensive, it becomes very important to introduce pricing/billing strategies for the ad hoc wireless internet.

❖ **Provisioning of security:**

- Security is a prime concern since the end users can utilize the ad hoc wireless internet infrastructure to make e-commerce transaction.

❖ **QoS support:**

- ♥ With the widespread use of voice over IP (VOIP) & growing multimedia applications over the internet, provisioning of QoS support in the ad hoc wireless internet becomes a very important issue.

❖ **Service, address & location discovery:**

- Service discovery refers to the activity of discovering or identifying the party which provides service or resource.
- Address discovery refers to the services such as those provided by Address Resolution Protocol (ARP) or Domain Name Service (DNS) operating within the wireless domain.
- Location discovery refers to different activities such as detecting the location of a particular mobile node in the network or detecting the geographical location of nodes.

ROUTING

INTRODUCTION

Since the ad hoc wireless network consists of a set of mobile nodes (hosts) that are connected by wireless links, the network topology in such a network may keep changing randomly. Hence a variety of routing protocols for ad hoc wireless networks has been proposed.

ISSUES IN DESIGNING A ROUTING PROTOCOL FOR AD HOC WIRELESS NETWORKS

The major challenges that a routing protocol designed for ad hoc wireless networks faces are:

Mobility

- Network topology is highly dynamic due to movement of nodes. hence, an ongoing session suffers frequent path breaks.
- Disruption occurs due to the movement of either intermediate nodes in the path or end nodes.
- Wired network routing protocols cannot be used in adhoc wireless networks because the nodes are here are not stationary and the convergence is very slow in wired networks.
- Mobility of nodes results in frequently changing network topologies
- Routing protocols for ad hoc wireless networks must be able to perform efficient and effective mobility management.

Bandwidth Constraint

- Abundant bandwidth is available in wired networks due to the advent of fiber optics and due to the exploitation of wavelength division multiplexing (WDM) technologies.
- In a wireless network, the radio band is limited, and hence the data rates it can offer are much less than what a wired network can offer.
- This requires that the routing protocols use the bandwidth optimally by keeping the overhead as low as possible.
- The limited bandwidth availability also imposes a constraint on routing protocols in maintaining the topological information.

Error-prone shared broadcast radio channel

- The broadcast nature of the radio channel poses a unique challenge in ad hoc wireless networks.
- The wireless links have time-varying characteristics in terms of link capacity and link-error probability.
- This requires that the adhoc wireless network routing protocol interact with the MAC layer to find alternate routes through better-quality links.
- Transmissions in ad hoc wireless networks result in collisions of data and control packets.
- Therefore, it is required that ad hoc wireless network routing protocols find paths with less congestion.

Hidden and exposed terminal problems

- The hidden terminal problem refers to the collision of packets at a receiving node due to the simultaneous transmission of those nodes that are not within the direct transmission range of the receiver, but are within the transmission range of the receiver.
- Collision occurs when both nodes transmit packets at the same time without knowing about the transmission of each other.
- Ex: consider figure 7.1. Here, if both node A and node C transmit to node B at the same time, their packets collide at node B. This is due to the fact that both node A and C are hidden from each other, as

they are not within the direct transmission range of each other and hence do not know about the presence of each other.

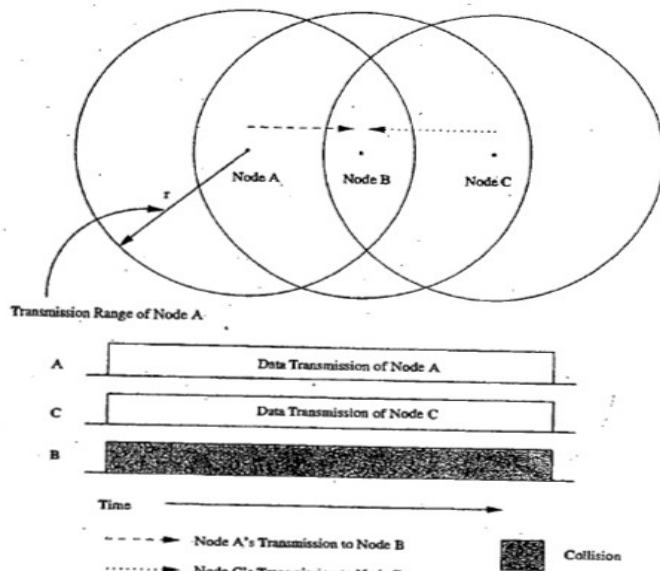


Figure 7.1. Hidden terminal problem.

- Successful transmission is a four-way exchange mechanism, RTS-CTS-Data-ACK, as illustrated in figure 7.2.

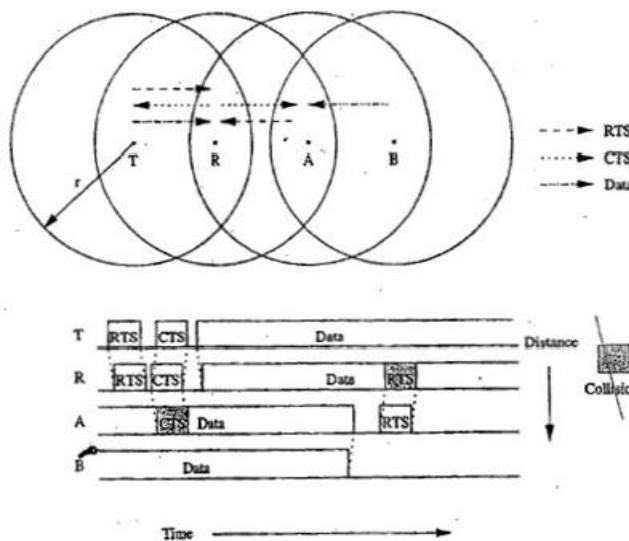


Figure 7.2. Hidden terminal problem, with RTS-CTS-Data-ACK scheme.

- Other solutions include floor acquisition multiple access (FAMA) and Dual busy tone multiple access (DBTMA).

- The exposed terminal problem refers to the inability of a node which is blocked due to transmission by a nearby transmitting node to transmit to another node.
- Ex: consider the figure 7.3. Here, if a transmission from node B to another node A is already in progress, node C cannot transmit to node D, as it concludes that its neighbor node B, is in transmitting mode and hence should not interfere with the on-going transmission. Thus, reusability of the radio spectrum is affected.

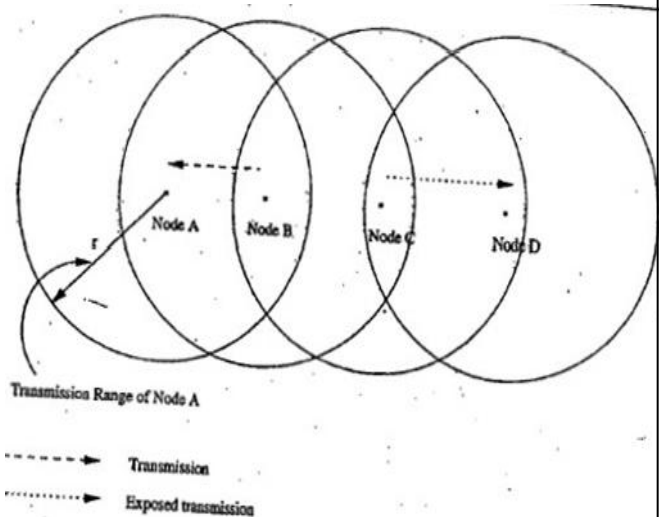


Figure 7.3. Exposed terminal problem.

MACA

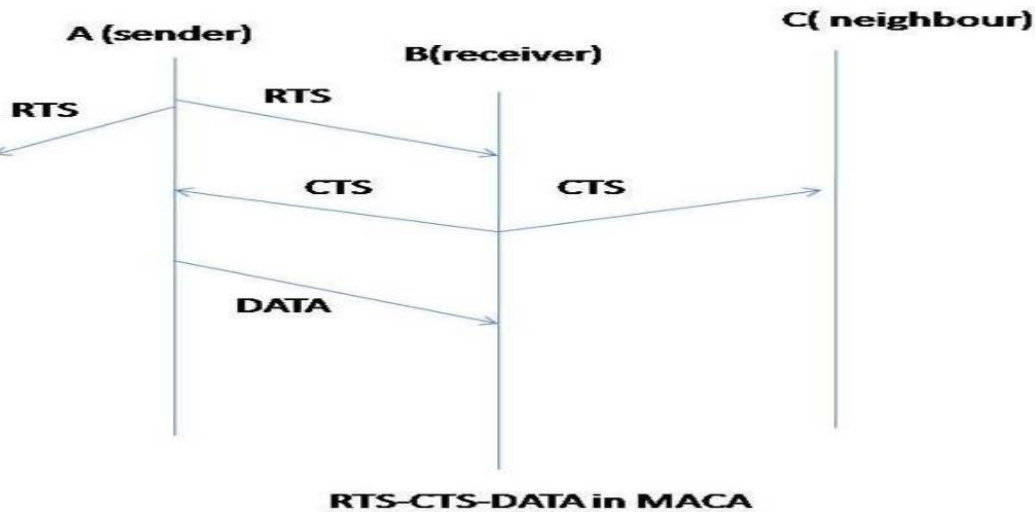


Figure II MACA Protocol

RTS/CTS (Request To Send / Clear To Send) is the optional mechanism used by the 802.11 wireless networking protocol to reduce frame collisions introduced by the hidden node problem. When a node wants to transmit data to another node, it sends out a RTS 'Request to Send' packet. The receiver node replies with a packet called CTS 'Cleared to Send' packet. After the transmitter node receives the CTS packet, it transmits the data packets

MACAW- Medium Access Collusion Avoidance for Wireless -Four way mechanism

Medium access collision avoidance for wireless (MACAW):

- o An improved version of MACA protocol.
- o Introduced to increase the efficiency.
- o Requires that a receiver acknowledges each successful reception of data packet

RTS-CTS-Data-ACK

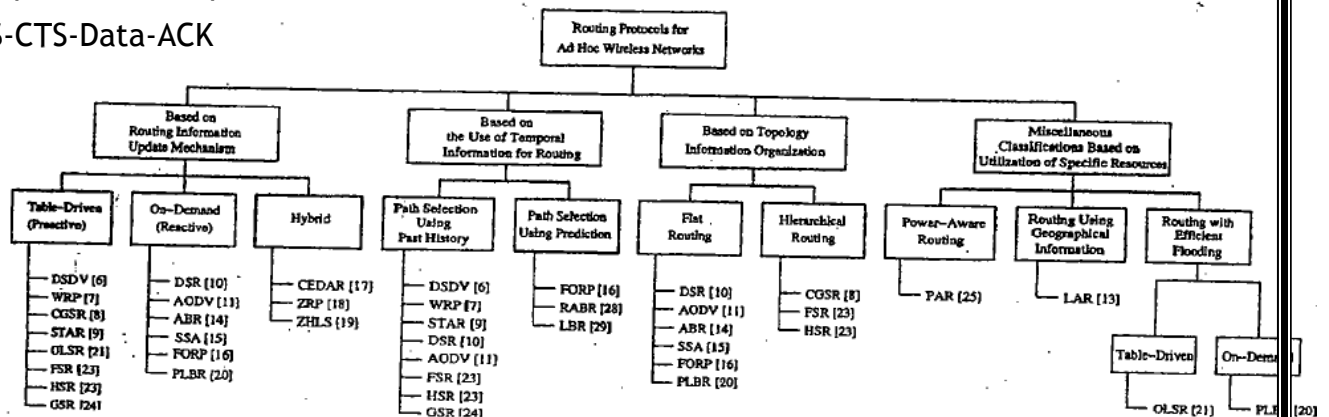


Figure 7.4. Classifications of routing protocols.

Resource Constraints

- Two essential and limited resources are battery life and processing power.
- Devices used in adhoc wireless networks require portability, and hence they also have size and weight constraints along with the restrictions on the power source.
- Increasing the battery power and processing ability makes the nodes bulky and less portable.

Characteristics of an Ideal Routing Protocol for ad hoc wireless networks

A routing protocol for ad hoc wireless networks should have the following characteristics:

- It must be fully distributed as centralized routing involves high control overhead and hence is not scalable.
- It must be adaptive to frequent topology changes caused by the mobility of nodes.
- Route computation and maintenance must involve a minimum number of nodes. Each node in the network must have quick access to routes, that is, minimum connection setup time is desired.
- It must be localized, as global state maintenance involves a huge state propagation control overhead.
- It must be loop-free and free from state routes.
- The number of packet collisions must be kept to a minimum by limiting the number of broadcasts made by each node. The transmissions should be reliable to reduce message loss and to prevent the occurrence of state routes.
- It must converge to optimal routes once the network topology becomes stable. The convergence must be quick.
- It must optimally use scarce resources such as bandwidth, computing power, memory, and battery power.
- Every node in the network should try to store information regarding the stable local topology only. Changes in remote parts of the network must not cause updates in the topology information maintained by the node.
- It should be able to provide a certain level of quality of service (QoS) as demanded by the applications, and should also offer support for time-sensitive traffic.

CLASSIFICATIONS OF ROUTING PROTOCOLS

A classification tree is shown below:

The routing protocol for adhoc wireless networks can be broadly classified into 4 categories based on

- Routing information update mechanism.
- Use of temporal information for routing
- Routing topology
- Utilization of specific resources.

Based on the routing information update mechanism

Ad hoc wireless network routing protocols can be classified into 3 major categories based on the routing information update mechanism. They are:

- *Proactive or table-driven routing protocols :*
 - Every node maintains the network topology information in the form of routing tables by periodically exchanging routing information.
 - Routing information is generally flooded in the whole network.
 - Whenever a node requires a path to a destination, it runs an appropriate path-finding algorithm on the topology information it maintains.
- *Reactive or on-demand routing protocols:*
 - Do not maintain the network topology information.
 - Obtain the necessary path when it is required, by using a connection establishment process.
- *Hybrid routing protocols:*
 - Combine the best features of the above two categories.
 - Nodes within a certain distance from the node concerned, or within a particular geographical region, are said to be within the routing zone of the given node.
 - For routing within this zone, a table-driven approach is used.
 - For nodes that are located beyond this zone, an on-demand approach is used.

Based on the use of temporal information for routing

The protocols that fall under this category can be further classified into two types :

- *Routing protocols using past temporal information:*
 - Use information about the past status of the links or the status of links at the time of routing to make routing decisions.
- *Routing protocols that use future temporal information:*
 - Use information about the about the expected future status of the wireless links to make approximate routing decisions.
 - Apart from the lifetime of wireless links, the future status information also includes information regarding the lifetime of the node, prediction of location, and prediction of link availability.

Based on the routing topology

Ad hoc wireless networks, due to their relatively smaller number of nodes, can make use of either a flat topology or a hierarchical topology for routing.

- *Flat topology routing protocols:*
 - Make use of a flat addressing scheme similar to the one used in IEEE 802.3 LANs.
 - It assumes the presence of a globally unique addressing mechanism for nodes in an ad hoc wireless network.
- *Hierarchical topology routing protocols:*
 - Make use of a logical hierarchy in the network and an associated addressing scheme.
 - The hierarchy could be based on geographical information or it could be based on hop distance.

Based on the utilization of specific resources

- *Power-aware routing:*
 - Aims at minimizing the consumption of a very important resource in the ad hoc wireless networks: the battery power.
 - The routing decisions are based on minimizing the power consumption either logically or globally in the network.
- *Geographical information assisted routing :*
 - Improves the performance of routing and reduces the control overhead by effectively utilizing the geographical information available.

TABLE-DRIVEN ROUTING PROTOCOLS

- These protocols are extensions of the wired network routing protocols
- They maintain the global topology information in the form of tables at every node.
- Tables are updated frequently in order to maintain consistent and accurate network state information
- Ex: Destination sequenced distance vector routing protocol (DSDV), wireless routing protocol (WRP), source-tree adaptive routing protocol (STAR) and cluster-head gateway switch routing protocol (CGSR).

Destination sequenced distance-vector routing protocol

- It is an enhanced version of the distributed Bellman-Ford algorithm where each node maintains a table that contains the shortest distance and the first node on the shortest path to every other node in the network.
- It incorporates table updates with increasing sequence number tags to prevent loops, to counter the count-to-infinity problem, and for faster convergence.
- As it is a table-driven routing protocol, routes to all destinations are readily available at every node at all times.
- The tables are exchanged between neighbors at regular intervals to keep an up-to-date view of the network topology.
- The table updates are of two types:
 - **Incremental updates:** Takes a single network data packet unit (NDPU). These are used when a node does not observe significant changes in the local topology.
 - **Full dumps:** Takes multiple NDPUs. It is done either when the local topology changes significantly or when an incremental update requires more than a single NDPU.
- Table updates are initiated by a destination with a new sequence number which is always greater than the previous one.
- Consider the example as shown in figure (a). Here node 1 is the source node and node 15 is the destination. As all the nodes maintain global topology information, the route is already available as shown in figure (b).
- Here the routing table node 1 indicates that the shortest route to the destination node is available through node 5 and the distance to it is 4 hops, as depicted in figure (b)
- The reconfiguration of a path used by an on-going data transfer session is handled by the protocol in the following way.
- The end node of the broken link initiates a table update message with the broken link's weight assigned to infinity (∞) and with a sequence number greater than the stored sequence number for that destination.
- Each node upon receiving an update with weight ∞ , quickly disseminates it to its neighbors in order to propagate the broken-link information to the whole network.
- A node always assigns an odd number to the link break update to differentiate it from the even sequence number generated by the destination.
- Figure 7.6 shows the case when node 11 moves from its current position.

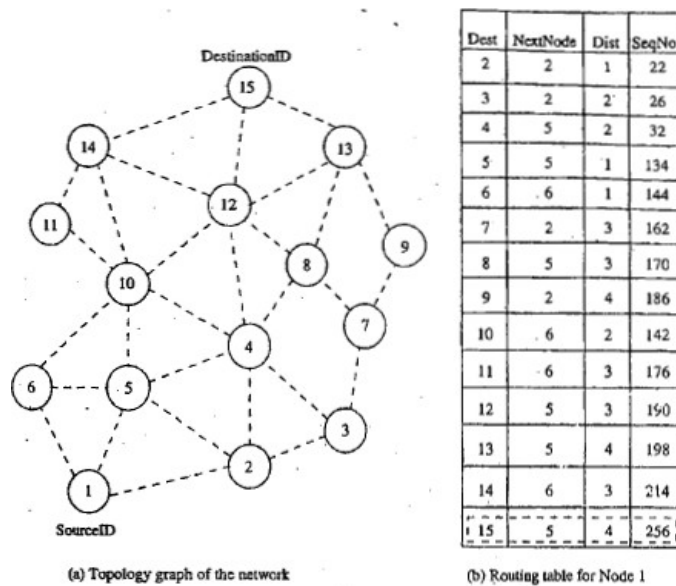


Figure 7.5. Route establishment in DSDV.

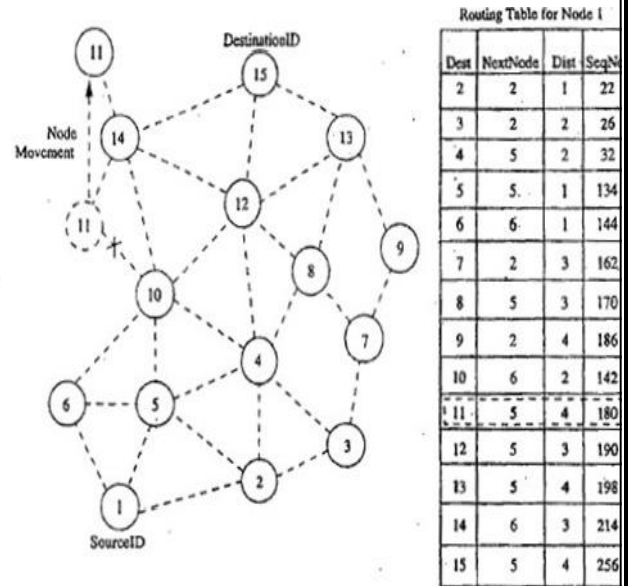


Figure 7.6. Route maintenance in DSDV.

Advantages

- Less delay involved in the route setup process.
- Mechanism of incremental update with sequence number tags makes the existing wired network protocols adaptable to ad hoc wireless networks.
- The updates are propagated throughout the network in order to maintain an up-to-date view of the network topology at all nodes.

Disadvantages

- The updates due to broken links lead to a heavy control overhead during high mobility.
- Even a small network with high mobility or a large network with low mobility can completely choke the available bandwidth.
- Suffers from excessive control overhead.
- In order to obtain information about a particular destination node, a node has to wait for a table update message initiated by the same destination node.
- This delay could result in state routing information at nodes.

Wireless Routing Protocol (WRP)

- WRP is similar to DSDV; it inherits the properties of the distributed bellman-ford algorithm.
- To counter the count-to-infinity problem and to enable faster convergence, it employs a unique method of maintaining information regarding the shortest distance to every destination node in the network and penultimate hop node on the path to every destination node.
- Maintains an up-to-date view of the network, every node has a readily available route to every destination node in the network.
- It differs from DSDV in table maintenance and in the update procedures.
- While DSDV maintains only one topology table, WRP uses a set of tables to maintain more accurate information.
- The table that are maintained by a node are:
 - **Distance table (DT):** contains the network view of the neighbors of a node. It contains a matrix where each element contains the distance and the penultimate node reported by the neighbor for a particular destination.
 - **Routing table (RT):** contains the up-to-date view of the network for all known destinations. It keeps the shortest distance, the predecessor/penultimate node, the successor node, and a flag indicating the status of the path. The path status may be a simplest (correct) path or a loop (error), or destination node not marked (null).

- **Link cost table (LCT):** contains the cost of relaying messages through each link. The cost of broken link is ∞ . It also contains the number of update periods passed since the last successful update was received from that link.
- **Message retransmission list (MRL):** contains an entry for every update message that is to be retransmitted and maintains a counter for each entry.
- After receiving the update message, a node not only updates the distance for transmitted neighbors but also checks the other neighbors' distance, hence convergence is much faster than DSDV.
- Consider the example shown in figure below, where the source of the route is node 1 and destination is node 15. As WRP proactively maintains the route to all destinations, the route to any destination node is readily available at the source node.
- From the routing table shown, the route from node 1 to node 15 has the next node as node 2. The predecessor node of 15 corresponding to this route is route 12. The predecessor information helps WRP to converge quickly during link breaks.

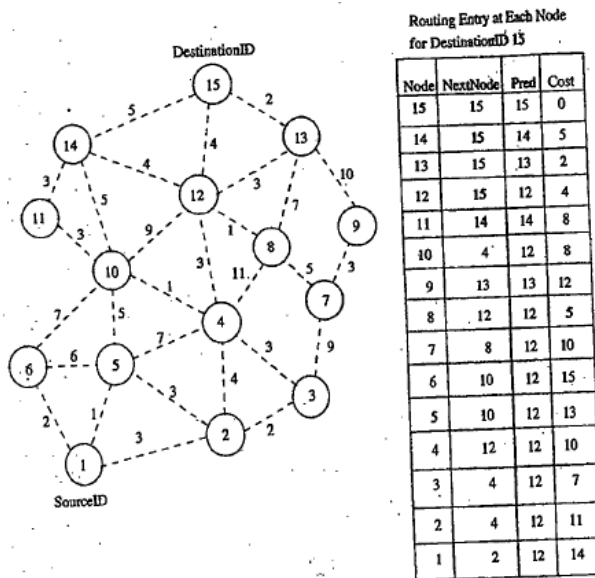


Figure 7.7. Route establishment in WRP.

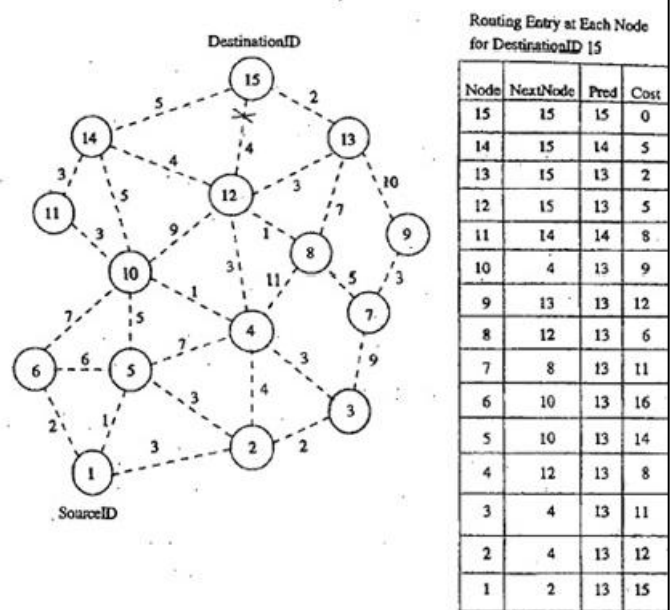


Figure 7.8. Route maintenance in WRP.

- When a node detects a link break, it sends an update message to its neighbors with the link cost of the broken link set to ∞ . After receiving the update message; all affected nodes update their minimum distances to the corresponding nodes. The node that initiated the update message then finds an alternative route, if available from its DT. Figure 7.8 shows route maintenance in WRP.

Advantages

- WRP has the same advantages as that of DSDV.
- It has faster convergence and involves fewer table updates.

Disadvantages

- The complexity of maintenance of multiple tables demands a larger memory and greater processing power from nodes in the adhoc wireless network.
- It is not suitable for highly dynamic and also for very large ad hoc wireless networks.

- According to this algorithm, a node ceases to be a cluster-head only if it comes under the range of another cluster-head, where the tie is broken either using the lowest ID or highest connectivity algorithm.
- Clustering provides a mechanism to allocate bandwidth, which is a limited resource, among different clusters, thereby improving reuse.
- A token-based scheduling is used within a cluster for sharing the bandwidth among the members of the cluster.
- CGRS assumes that all communication passes through the cluster-head. Communication between 2 clusters takes place through the common member nodes that are members of both the cluster are called *gateways*.
- A gateway is expected to be able to listen to multiple spreading codes that are currently in operation in the clusters in which the node exist as a member.
- A gateway conflict is said to occur when a cluster-head issues a token to a gateway over spreading code while the gateway is tuned to another code.
- Gateways that are capable of simultaneously communicating over two interfaces can avoid gateway conflicts.
- The performance of routing is influenced by token scheduling and code scheduling that is handled at cluster-heads and gateways, respectively.
- Every member node maintains a routing table containing the destination cluster-head for every node in the network.
- In addition to the cluster member table, each node maintains a routing table which keeps the list of next-hop nodes for reaching every destination cluster.
- The cluster routing protocol is used here.
- Figure below shows the cluster head, cluster gateways, and normal cluster member nodes in an ad hoc wireless network.

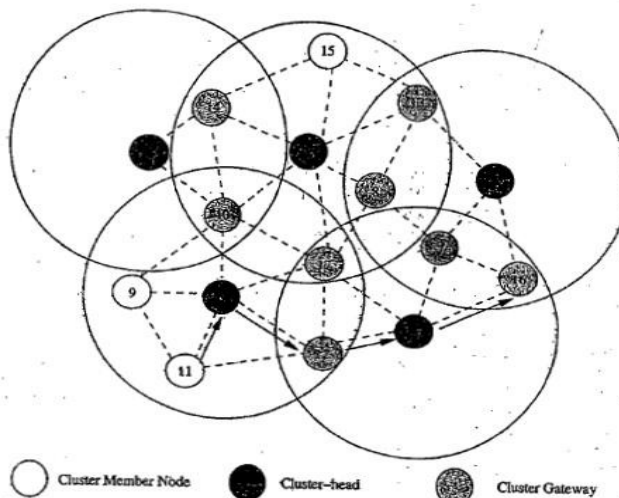


Figure 7.9. Route establishment in CGSR.


Advantages

- CGSR is a hierarchical routing scheme which enables partial coordination between nodes by electing cluster-heads.
- Better bandwidth utilization is possible.
- Easy to implement priority scheduling schemes with token scheduling and gateway code scheduling.

Disadvantages

- Increase in path length and instability in the system at high mobility when the rate of change of cluster-head is high.
- In order to avoid gateway conflicts, more resources are required.
- The power consumption at the cluster-head node is also a matter of concern.
- Lead to Frequent changes in the cluster-head, which may result in multiple path breaks.

Source-Tree Adaptive Routing Protocol (STAR)

- Key concept  least overhead routing approach (LORA)
- This protocol attempts to provide feasible paths that are not guaranteed to be optimal
- Involves much less control overhead
- In STAR protocol, every node broadcasts its source tree information
- The source tree of a node consists of the wireless links used by the node
- Every node builds a partial graph of the topology
- During initialization, a node sends an update message to its neighbors
- Each node will have a path to every destination node

- The path would be sub-optimal
- The data packet contains information about the path to be traversed in order to prevent the possibility of routing loopformation
- In the presence of a reliable broadcast mechanism, STAR assumes implicit route maintenance
- In addition to path breaks, the intermediate nodes are responsible for handling the routing loops
- The RouteRepair packet contains the complete source tree of node k and the traversed path of the packet
- When an intermediate node receives a RouteRepair update message, it removes itself from the top of the route repair path and reliably sends it to the head of the route repair path


Advantages

- Very low communication overhead
- Reduces the average control overhead

ON-DEMAND ROUTING PROTOCOLS

They execute the path-finding process and exchange routing information only when a path is required by a node to communicate with a destination

Dynamic Source Routing Protocol (DSR)

- Designed to restrict the bandwidth consumed by control packets in adhoc wireless networks by eliminating the periodic table update messages
- It is beacon-less and does not require periodic hello packet transmissions
- Basic approach  to establish a route by flooding RouteRequest packets in the network
- Destination node responds by sending a RouteReply packet back to the source
- Each RouteRequest carries a sequence number generated by the source node and the path it has traversed
- A node checks the sequence number on the packet before forwarding it
- The packet is forwarded only if it is not a duplicate RouteRequest
- The sequence number on the packet is used to prevent loop formations and to avoid multiple transmissions
- Thus, all nodes except the destination forward a RouteRequest packet during the route construction phase
- In figure 7.10, source node 1 initiates a RouteRequest packet to obtain a path for destination node 15
- This protocol uses a route cache that stores all possible information extracted from the source route contained in a data packet
- During network partitions, the affected nodes initiate RouteRequest packets
- DSR also allows piggy-backing of a data packet on the RouteRequest
- As a part of optimizations, if the intermediate nodes are also allowed to originate RouteReply packets, then a source node may receive multiple replies from intermediate nodes
- In fig 7.11, if the intermediate node 10 has a route to the destination via node 14, it also sends the RouteReply to the source node
- The source node selects the latest and best route and uses that for sending data packets
- Each data packet carries the complete path to its destination
- If a link breaks, source node again initiates the route discovery process

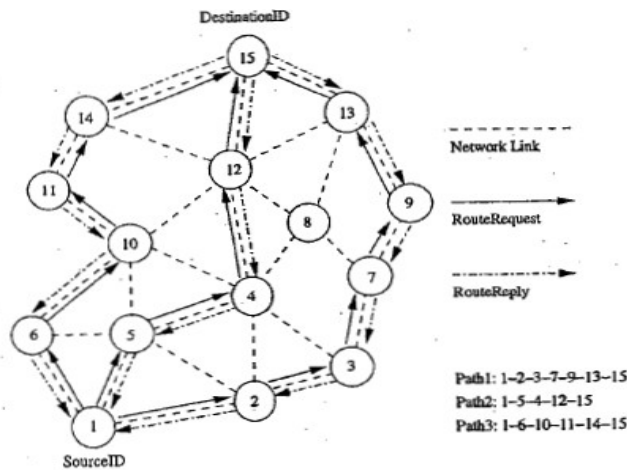


Figure 7.10. Route establishment in DSR.

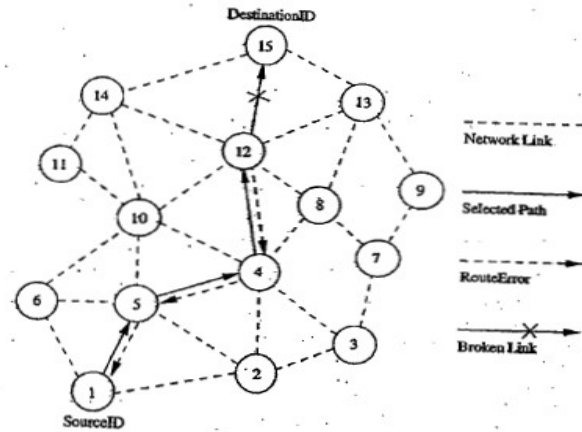


Figure 7.11. Route maintenance in DSR.

Advantages

- Uses a reactive approach which eliminates the need to periodically flood the network with table update messages
- Route is established only when required
- Reduce control overhead

Disadvantages

- Route maintenance mechanism does not locally repair a broken link
- Stale route cache information could result in inconsistencies during route construction phase
- Connection set up delay is higher
- Performance degrades rapidly with increasing mobility
- Routing overhead is more & directly proportional to path length

Ad Hoc On-Demand Distance Vector Routing Protocol

- Route is established only when it is required by a source node for transmitting data packets
- It employs destination sequence numbers to identify the most recent path
- Source node and intermediate nodes store the next hop information corresponding to each flow for data packet transmission
- Uses DestSeqNum to determine an up-to-date path to the destination
- A RouteRequest carries the source identifier, the destination identifier, the source sequence number, the destination sequence number, the broadcast identifier and the time to live field
- DestSeqNum indicates the freshness of the route that is accepted by the source
- When an intermediate node receives a RouteRequest, it either forwards it or prepares a RouteReply if it has a valid route to the destination
- The validity of the intermediate node is determined by comparing the sequence numbers
- If a RouteRequest is received multiple times, then duplicate copies are discarded
- Every intermediate node enters the previous node address and its BcastID
- A timer is used to delete this entry in case a RouteReply packet is not received
- AODV does not repair a broken path locally
- When a link breaks, the end nodes are notified
- Source node re-establishes the route to the destination if required

Advantage

- Routes are established on demand and DestSeqNum are used to find latest route to the destination
- Connection setup delay is less

Disadvantages

- Intermediate nodes can lead to inconsistent routes if the source sequence destination is very old

- Multiple Route Reply packets to single Route Request packet can lead to heavy control overhead
- Periodic beaconing leads to unnecessary bandwidth consumption

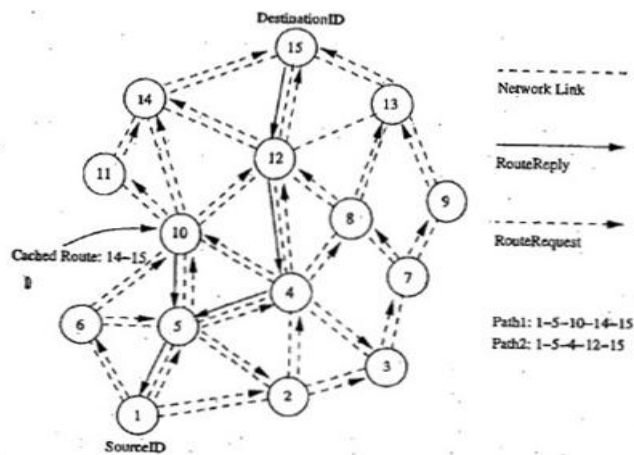


Figure 7.12. Route establishment in AODV.

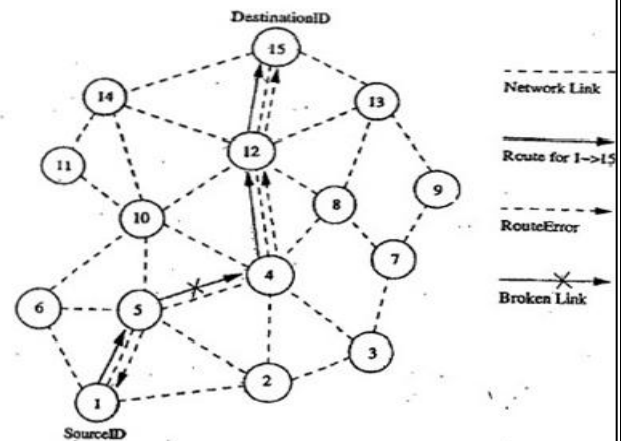


Figure 7.13. Route maintenance in AODV.

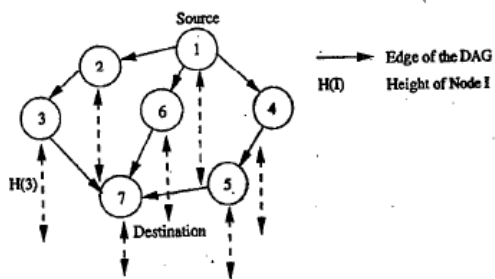
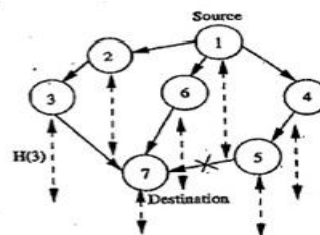
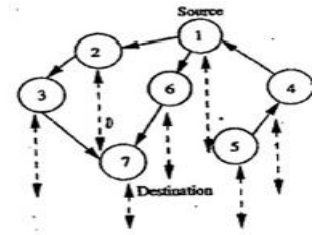


Figure 7.14. Illustration of temporal ordering in TORA.



Link break between Nodes 5 and 7



Nodes 4 and 5 reverse their links in order to update the path

Figure 7.15. Illustration of route maintenance in TORA.

Mohamad Sathak A.J College of Engineering
Department of ECE

EC8702
ADHOC AND WIRELESS SENSOR NETWORKS

UNIT II
SENSOR NETWORKS - INTRODUCTION &
ARCHITECTURES

M.KAMARAJAN ,Associate Professor

UNIT II

Introduction

What is WSN?

Wireless sensor network refers to a group of spatially dispersed and dedicated sensors for monitoring and recording the physical conditions of the environment and organizing the collected data at a central location

These are similar to wireless ad hoc networks in the sense that they rely on wireless connectivity and spontaneous formation of networks so that sensor data can be transported wirelessly.

WSNs are spatially distributed autonomous sensors to *monitor* physical or environmental conditions, such as temperature, sound, pressure, etc\

WSN is a wireless network that consists of base stations and numbers of nodes (wireless sensors). These networks are used to monitor physical or environmental conditions like sound, pressure, temperature, and co-operatively pass data through the network to the main location .

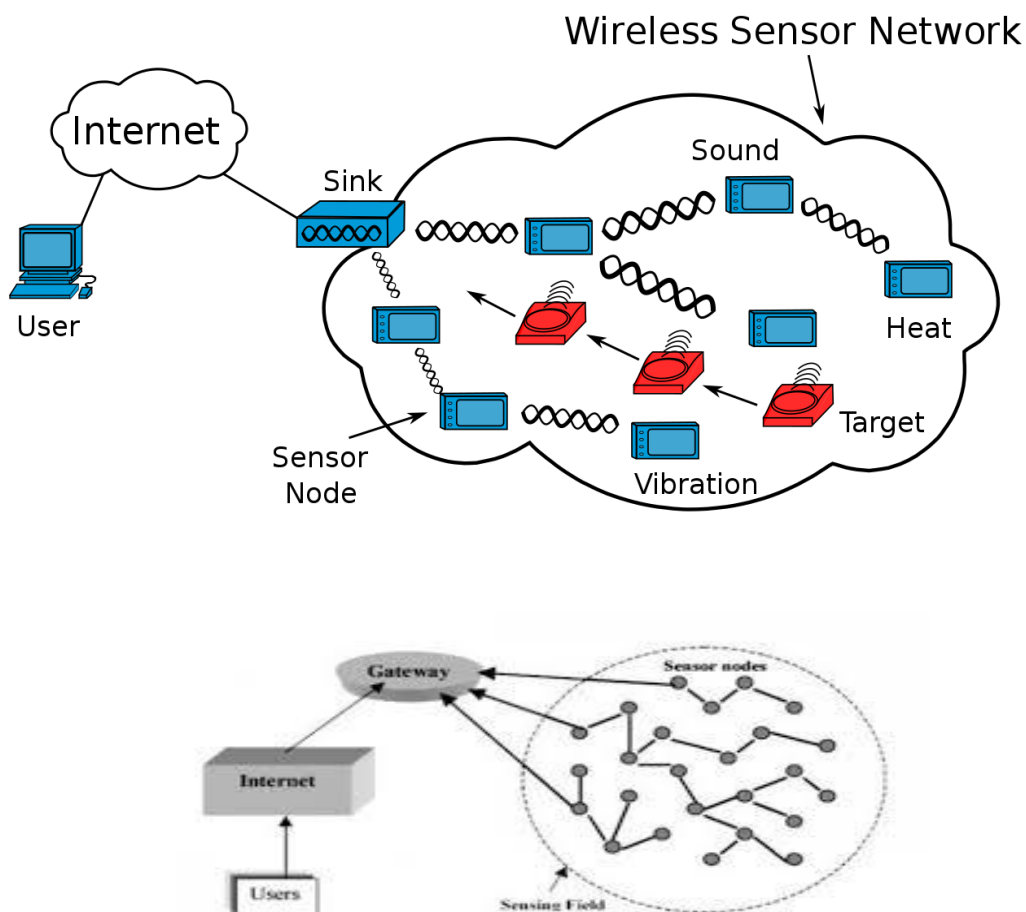


Figure: 1 Wireless Sensor Network

WSN **Sensors** are equipped with sensing, limited computation, and wireless communication capabilities

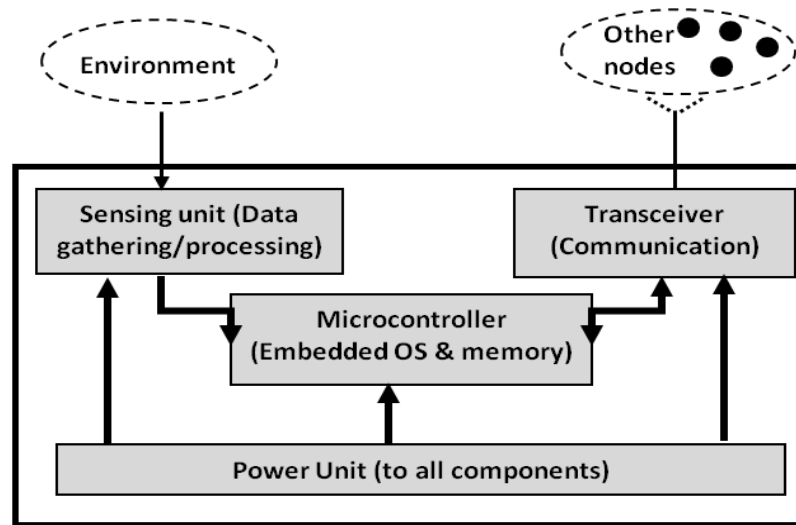


Figure 2: WSN Single Node Architecture

Comparison of WSN with ad hoc networks:

- Wireless sensor networks mainly use broadcast communication while ad hoc networks use point-to-point communication.
- Unlike ad hoc networks wireless sensor networks are limited by sensors limited power, energy and computational capability.
- Sensor nodes may not have global ID because of the large amount of overhead and large number of sensors.

WSNs Applications

- WSNs have many advantages over traditional networking techniques.
- They have an ever-increasing number of applications, such as infrastructure protection and security, surveillance, health-care, environment monitoring, food safety, intelligent transportation, and smart energy.

The applications can be divided in three categories:

1. Monitoring of objects.
2. Monitoring of an area.
3. Monitoring of both area and objects.

Monitoring Area

- Environmental and Habitat Monitoring
- Precision Agriculture
- Indoor Climate Control
- Military Surveillance
- Treaty Verification
- Intelligent Alarms

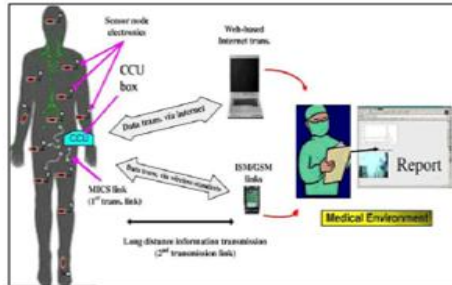
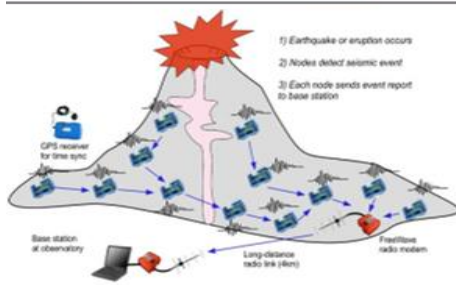
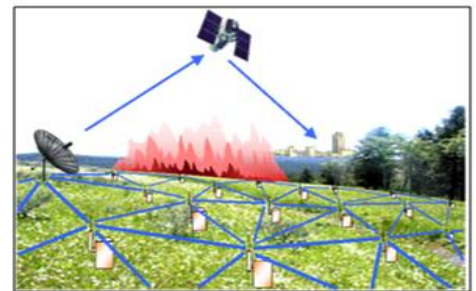
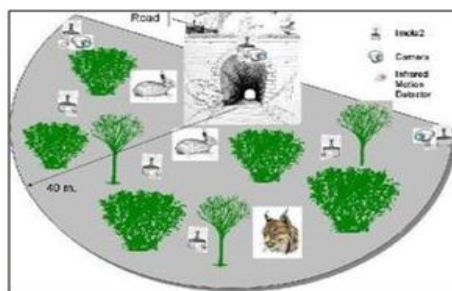
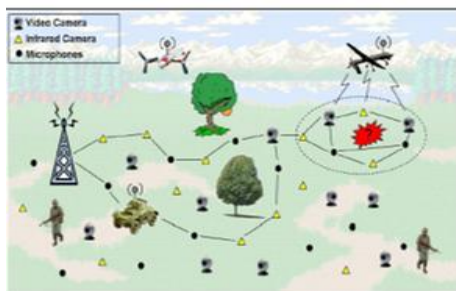
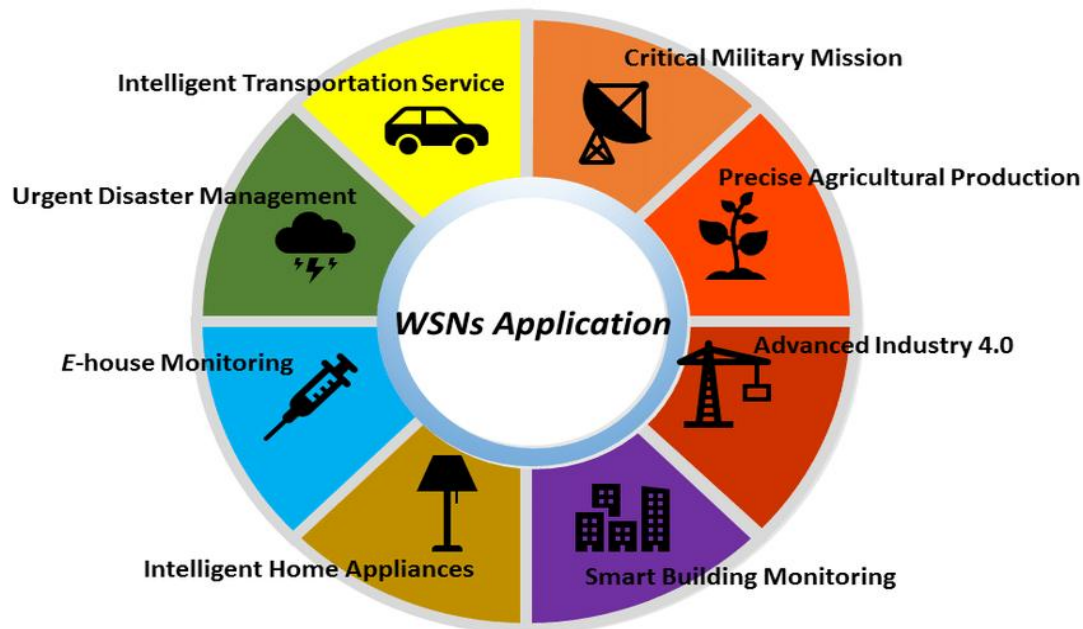


Figure 3: WSN Applications

Characteristics of Wireless Sensor Networks

Wireless Sensor Networks mainly consists of **sensors**. **Sensors** are -

- low power
- limited memory
- energy constrained due to their small size.
- Wireless networks can also be deployed in **extreme environmental** conditions and may be prone to enemy attacks.
- Although deployed in an ad hoc manner they need to be **self organized** and **self healing** and can face constant reconfiguration.

CHALLENGES FOR WSNs (T2,Page 7)

(i) Characteristic requirements

- TYPE OF SERVICE
- QUALITY OF SERVICE
- FAULT TOLERANCE:
- LIFE TIME:
- SCALIBILITY:
- WIDE RANGE OF DENSITIES
- PROGRAMMABILITY
- MAINTAINABILITY

(ii) Required mechanisms

- Multihop wireless communication
- Energy-efficient operation
- Auto-configuration
- Collaboration and in-network processing
- Data centric
- Locality
- Exploit trade-offs

- **TYPE OF SERVICE:** Provide meaningful information and actions about a given task, hence new paradigms of using such a network are required, along with new interfaces and new ways of thinking about the service of a network.
- **QUALITY OF SERVICE:** Packet delivery ratio is an insufficient. Adapted concepts like reliable detection of events or the approximation of quality is important.
- **FAULT TOLERANCE:** Since nodes may run out of energy or might be damaged, or since the wireless communication between two nodes can be permanently interrupted, it is important that the WSN as a whole is able to tolerate such faults. To tolerate node failure, redundant deployment is necessary, using more nodes than would be strictly necessary if all nodes functioned Correctly. (Nodes should be damage and failure tolerant.)
- **LIFE TIME:** WSN nodes rely on a limited supply of energy (batteries). Replacing these energy sources in the field is not practical; hence life time of WSN becomes important figure of merit.
- The lifetime of a network also has direct trade-offs against quality of service: investing more energy

can increase quality but decrease lifetime. Concepts to harmonize these trade-offs are required.

- **SCALABILITY:** Protocols used must be able to support large no. of nodes, the employed architectures and protocols must be able scale to these numbers.
- **WIDE RANGE OF DENSITIES:**
Density of network: No. of nodes per unit area– the *density of the network*
 – can vary considerably. & it vary with applications. Density can vary over time and space because nodes fail or move; the density also does not have to homogeneous in the entire network and the network should adapt to such variations.
- **PROGRAMMABILITY:** Nodes should be programmable, and their programming must be changeable during operation when new tasks become important. So it should be flexibility on changing tasks
- **MAINTAINABILITY:** Environment of a WSN and the WSN itself change, the system has to adapt. Has to maintain itself; able to interact with external maintenance mechanisms.

Required mechanisms

To realize these requirements, innovative mechanisms for a communication network have to be found, as well as new architectures, and protocol concepts

Some of the mechanisms that will form typical parts of WSNs are:

- Multihop wireless communication
- Energy-efficient operation
- Auto-configuration
- Collaboration and in-network processing
- Data centric
- Locality
- Exploit trade-offs

Multihop wireless communication: In particular, communication over long distances is only possible using prohibitively high transmission power. The use of intermediate nodes as relays can reduce the total required power. Hence, for many forms of WSNs, so-called *multihop communication* will be a necessary ingredient

Energy-efficient operation: To support long lifetimes of WSN, energy-efficient operation is necessary

Auto-configuration: A WSN will have to configure most of its operational parameters autonomously, independent of external configuration – the sheer number of nodes and simplified deployment will require that capability in most applications
 like Self location detection, able to tolerate failing nodes or to integrate new nodes etc

Collaboration and in-network processing

In some applications, a single sensor is not able to decide whether an event has happened but several sensors have to collaborate to detect an event and only the joint data of many sensors provides enough information. Information is processed in the network itself in various forms to achieve this collaboration, as opposed to having every node transmit all data to an external network and process it “at the edge” of the network.

An example is to determine the highest or the average temperature within an area and to report that value to a sink

Data centric: Traditional communication networks equipped **address-centric**. In a WSN, where nodes are typically deployed redundantly to protect against node failures or to compensate for the low quality of a single node's actual sensing equipment, the identity of the particular node supplying data becomes irrelevant. Hence, switching from an address-centric paradigm to a **data-centric** paradigm in designing architecture and communication protocols is promising.

Example: Data-centric interaction would be to request the average temperature in a given location area

Locality: The principle of locality is significant to ensure scalability.

Design Challenges:

Heterogeneity: The devices deployed maybe of various types and need to collaborate with each other.

Distributed Processing: The algorithms need to be centralized as the processing is carried out on different nodes.

Low Bandwidth Communication: The data should be transferred efficiently between sensors

Large Scale Coordination: The sensors need to coordinate with each other to produce required results.

Utilization of Sensors: The sensors should be utilized in a ways that produce the maximum performance and use less energy.

Real Time Computation: The computation should be done quickly as new data is always being generated.

Operational Challenges of Wireless Sensor Networks

- Energy Efficiency
- Limited storage and computation
- Low bandwidth and high error rates
- Errors are common
- Wireless communication
- Noisy measurements
- Node failure are expected
- Scalability to a large number of sensor nodes
- Survivability in harsh environments
- Experiments are time- and space-intensive

Why are sensor networks different?

Mobile ad hoc networks and wireless sensor networks

	WSN	MANET
Applications and equipments	<ul style="list-style-type: none"> Small sensor nodes with constrained hardware and energy supply. In general, unattended operation 	<ul style="list-style-type: none"> Powerful nodes with large batteries(laptops) In general, more elaborate Applications.e.g VoIP, with human interaction.
Redundancy	<ul style="list-style-type: none"> High 	<ul style="list-style-type: none"> Low
Data rate	<ul style="list-style-type: none"> Low 	<ul style="list-style-type: none"> High
Application specific	<ul style="list-style-type: none"> Infinite number of application in terms of devices,protocols,density etc. 	<ul style="list-style-type: none"> Although, a few scenarios not as many as in wsn.
Environment interaction	<ul style="list-style-type: none"> Lot of environmental interactions Low data rates, but also data bursts -new traffic patterns 	<ul style="list-style-type: none"> More conventional human driven applications with well understood traffic characteristics.
Scale	<ul style="list-style-type: none"> Huge amount of sensor nodes-more scalable solutions required(e.g. protocols without node identifiers) 	<ul style="list-style-type: none"> Significantly less nodes than in wsn.
Energy	<ul style="list-style-type: none"> Tighter requirements, mostly no recharge or replacement of batteries possible 	<ul style="list-style-type: none"> Energy constrained, but often energy can be recharged
Self Configurability	<ul style="list-style-type: none"> Almost equal to MANETs, but different data traffic and energy trade -offs. 	<ul style="list-style-type: none"> One of the main features in MANETs
Dependability and QoS	<ul style="list-style-type: none"> Individual node is irrelevant as long as network is working New QoS concepts necessary. 	<ul style="list-style-type: none"> Each node should be reliable Qos determined by applications such as VoIP jitter.
Data centric	<ul style="list-style-type: none"> Redundant deployment makes data centric protocols attractive. 	<ul style="list-style-type: none"> Slightly limited resources, but in general normal os and applications can run on the nodes.
Simplicity &	<ul style="list-style-type: none"> Os and s/w must be simpler than on 	<ul style="list-style-type: none"> Slightly limited resources,

Cellular Networks	Ad-hoc Networks
Fixed, pre-located cell sites and base stations.	No fixed base stations,
Slow Deployment	Very rapid deployment.
Static backbone network topology	Highly dynamic network topologies,
Single Hop	Single and Multihop Communication
Relatively favorable environment	Hostile environment (losses, noise)
Stable connectivity.	Irregular connectivity.
Detailed planning before base stations can be installed.	Ad-hoc network automatically forms and conforms to change.

Enabling Technologies for Wireless Sensor Networks

Que: Describe the enabling technologies and characteristic requirements of the wireless sensor network

Building such wireless sensor networks has only become possible with some fundamental advance sing enabling technologies. First and foremost among these technologies is the miniaturization of hardware. Smaller feature sizes in chips have driven down the power consumption of the basic components of a sensor node to a level that the constructions of WSNs can be contemplated. This is particularly relevant to microcontrollers and memory chips as such, but also, the radio modems, responsible for wireless communication, have become much more energy efficient. Reduced chip size and improved energy efficiency is accompanied by reduced cost, which is necessary to make redundant deployment of nodes affordable.

Enabling Technologies (Book Page No 16)

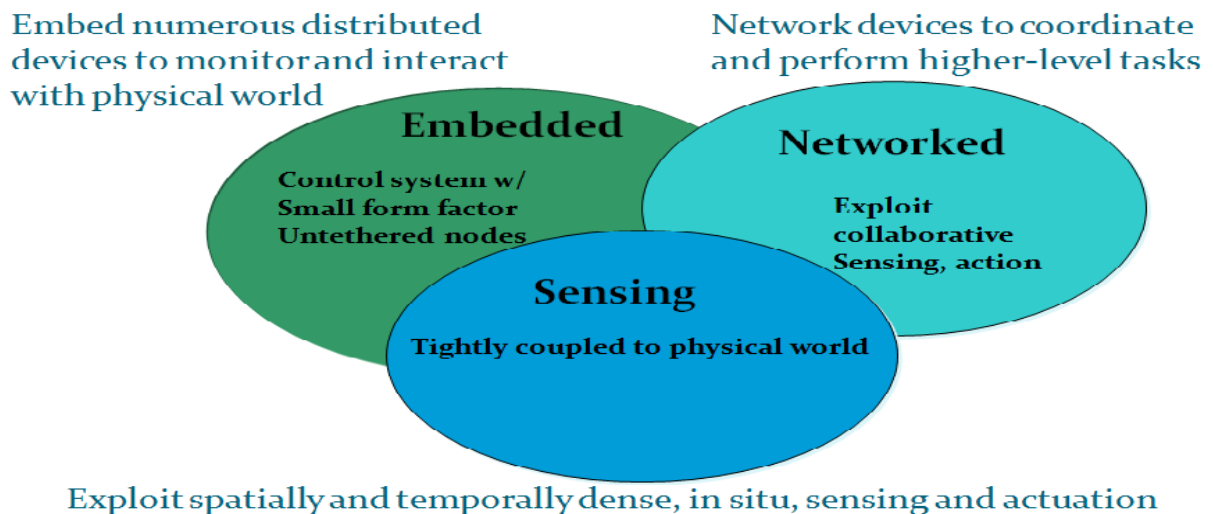


Figure 4: Enabling Technologies

- 1) Miniaturization of hardware
- 2) Reduced chip size & improved efficiency accompanied by reduced cost
- 3) The actual sensing element

These 3 parts have to be accompanied by power supply.

This requires, depending on application, high capacity batteries that last for long times, that is, have only a negligible self-discharge rate, and that can efficiently provide small amounts of current

A sensor node has a device *energy scavenging*.

What is energy harvesting or *energy scavenging* in wireless sensor networks?

Energy Harvesting-based WSNs (EHWSNs) are the result of endowing WSN nodes with the capability of extracting **energy** from the surrounding environment. **Energy harvesting** can exploit different sources of **energy**, such as solar power, wind, mechanical vibrations, temperature variations, magnetic fields, etc.

WSN Application Examples

Que: What are the applications of wireless sensor networks and explain with an example each

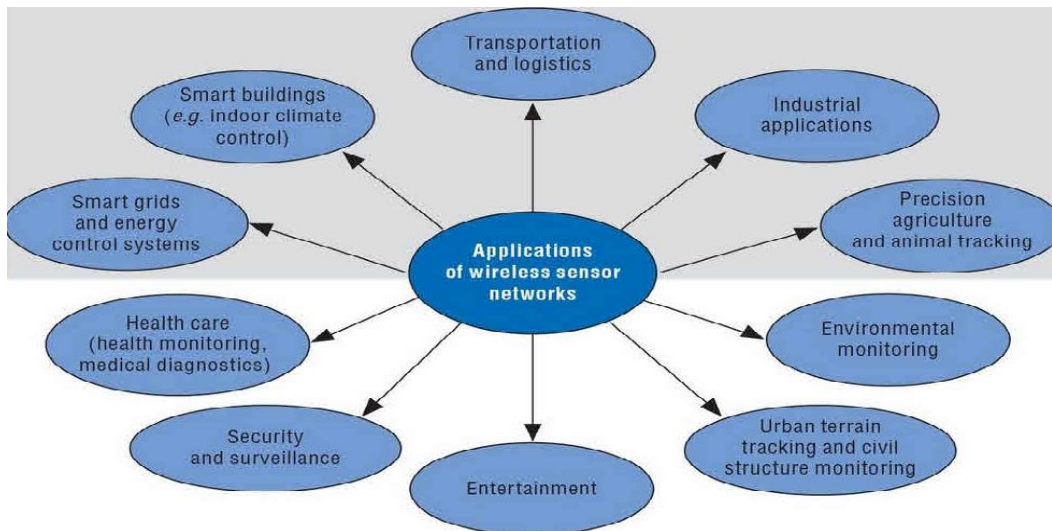


Figure :5 WSN Applications

- **DISASTER RELIEF APPLICATIONS:** Wildfire, sensors produce a “temperature map” of the area.
- **ENVIRONMENT CONTROL AND BIODIVERSITY MAPPING:** Chemical pollutants, surveillance of the marine ground floor and survey on no. of plant and animal species that live in a particular habitat.
- **INTELLIGENT BUILDINGS:** monitoring of temperature, airflow, humidity and other physical parameters in a building - reduce energy consumption. Also mechanical stress levels of buildings.
- **MACHINESURVEILLANCE & PREVENTIVE MAINTANANCE:** Sensor nodes are fixed at difficult-to- reach areas of machinery to detect vibration patterns.
- **FACILITY MANAGEMENT:** Detection of intruders, tracking vehicle’s position, scanning chemical leakage in chemical plants.
- **PRECISION AGRICULTURE:** Fertilizing by placing humidity/soil composition sensors into fields. Also pest control, livestock breeding can be improved.
- **MEDICINE & HEALTH CARE:** Long-term surveillance of patients and automatic drug administration. Also used in patient-doctor tracking system.
- **LOGISTICS:** Used for simple tracking of goods during transportation or to facilitate inventory tracking in stores and warehouses.
- **TELEMATICS:** Used to gather information about traffic conditions at a higher resolution(intelligent roadside)

Note: Also refer in introduction part WSN Applications

Single-Node Architecture

Question: Explain in detail about the single node architecture in wireless sensor networks

Outline

Hardware components

Energy consumption of sensor nodes

Operating systems and execution environments

Some examples of sensor nodes

Hardware Components:

- Sensor node hardware overview
- Controller(Microcontrollers versus microprocessors, FPGAs, and ASICs)
- Memory
- Communication device
- Sensors and actuators
- Power supply of sensor nodes

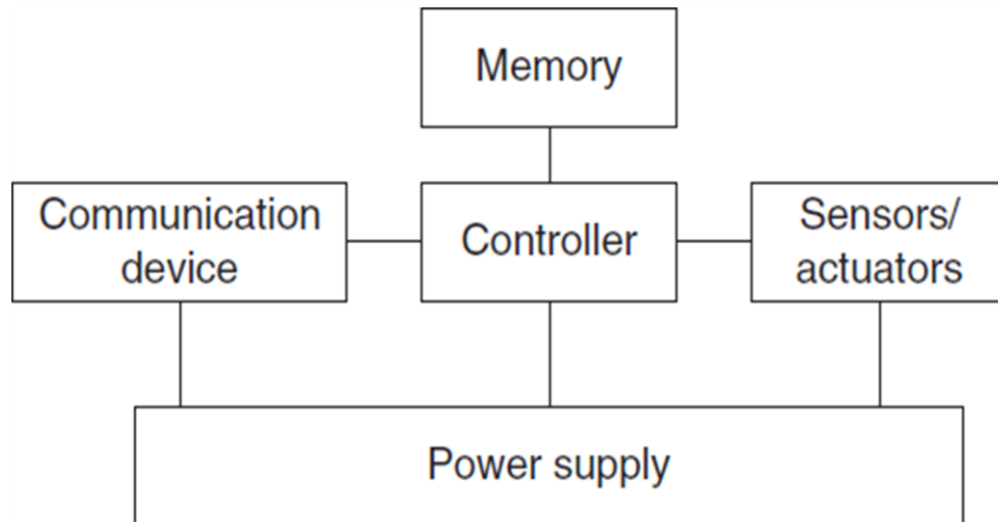


Figure 6: WSN Single Node Architecture

Five main components of basic sensor node

Controller: A controller to process all the relevant data, capable of executing arbitrary code.

Memory: Some memory to store programs and intermediate data; usually, different types of memory are used for programs and data.

Sensors and actuators: The actual interface to the physical world: devices that can observe or control physical parameters of the environment.

Communication: Turning nodes into a network requires a device for sending and receiving information over a wireless channel

Power supply: As usually no tethered power supply is available, some form of batteries are necessary to provide energy. Sometimes, some form of recharging by obtaining energy from the environment is available as well (e.g. solar cells).

CONTROLLER

- Collects data from the sensors, processes the data, decides when and where to send it, receives data from other sensor nodes, and decides on the actuator's behavior.
- Executes various programs, ranging from time-critical signal processing and communication protocols to application programs.
- Microcontrollers, microprocessors, FPGAs, and ASICs can be used as controllers
- **Microcontroller:**
 - Flexible in connecting with other devices (like sensors)
 - Instruction set amenable to time-critical signal processing
 - Low power consumption
 - Built in memory
 - Freely programmable
 - Can reduce power consumption by going to sleep state
 - No memory management unit, hence reduced memory functionality.
- **Digital Signal Processors (DSP):**
 - Specialized programmable processors
 - Highly useful in signal processing applications
 - In WSN DSP can be used to process data coming from a simple analog, wireless communication device to extract a digital data stream.
 - But in WSN signal processing functions are not overly complicated.
- **Field-Programmable Gate Arrays (FPGAs)**
 - Can be easily reprogrammed to changing environment
 - Consumes more time and energy compared to $m C$
- **Application Specific Integrated Circuits (ASICs)**
 - Specialized processor.
 - Trade off between flexibility and energy efficiency & performance.
 - A microcontroller requires software development, ASICs provide the same functionality in hardware, resulting in potentially more costly hardware development.
- Hence microcontrollers are more suitable for WSNs
- Eg. Microcontrollers Intel StrongARM, Texas Instruments MSP 430 and Atmel ATmega

MEMORY

- **Random Access Memory (RAM):**
 - Store intermediate sensor readings, packets from other nodes
 - Comparatively faster
 - Volatile in nature
- **Read-only Memory (ROM) or Electrically Erasable Programmable Read-only Memory (EEPROM):**
 - Store program code
- **Flash Memory:**
 - Allows data to be erased or written in blocks instead of only a byte at a time
 - Serve as intermediate storage of data in case RAM is insufficient or when the power supply of RAM should be shut down for some time
 - Longer delay and more consumption of energy

COMMUNICATION DEVICE

- Used to exchange data between individual nodes.
- **Choice of transmission medium**
 - There are different mediums for wireless communication. Usual choice includes are,
 - Radio frequencies
 - Optical communication
 - Ultrasound
 - Magnetic inductance- used very specific cases

RF-based Communication :

- Most suitable for WSN applications; since it provides
 - Relatively long range and high data rates
 - Acceptable error rates at reasonable energy expenditure and
 - No line of sight between sender and receiver required
 Frequencies used: between about 433 MHz and 2.4 GHz

TRANSCEIVERS:

For actual communication, both a transmitter and a receiver are required in a sensor node. The essential task is to convert a bit stream coming from a microcontroller (or a sequence of bytes or frames) and convert them to and from radio waves

- Device capable of doing both transmission and reception.
- But usually, only half duplex operation is realized.

Transceiver tasks and characteristics

1. Service to upper layer:

Offer certain services to the upper layers, most notably to the Medium Access Control (**MAC**) layer.

The service is mostly **packet oriented**. Sometimes, a transceiver only provides a **byte interface or even only a bit interface to the microcontroller**

In any case, the transceiver must provide an interface that somehow allows the MAC layer to initiate frame transmissions and to hand over the packet from, say, the main memory of the sensor node into the transceiver (or a byte or a bit stream, with additional processing required on the microcontroller).

2. Power consumption and energy efficiency:

Energy efficiency is the energy required to transmit and receive a single bit.

Transceivers should be switchable between different states (Eg. Idle, sleep and active).

3. Carrier frequency and multiple channels

Transceivers are available for different carrier frequencies.

Some provides several carrier frequencies ("channels"). For example, for certain MAC protocols (FDMA or multichannel CSMA/ ALOHA techniques

4. State change times and energy:

Transceiver can operate in different modes: sending or receiving, use different channels, or be in different power-safe states.

Important figures of merit are the time and the energy required to change between two states.

5. Data rates:

The gross data rate is determined by carrier frequency and used bandwidth together with modulation and coding. Typical values are a few tens of kilobits per second - considerably less than in broadband wireless communication, but usually sufficient for WSNs. Different data rates can be achieved, for example, by using different modulations or changing the symbol rate

6. Modulations

Transceivers can support one or several of on/off-keying (ASK, FSK etc.)

7. **Coding:** Some transceivers allow various coding schemes to be selected.
Transmission power control - Some transceivers can directly provide control over the transmission power to be used; some require some external circuitry for that purpose. Maximum output power is usually determined by regulations.
8. **Noise figure:** The **noise figure NF of an element is defined as the ratio of the Signal-to-Noise Ratio (SNR) ratio SNR_I at the input of the element to the SNR ratio SNR_O at the element's output:**

$$NF = SNR_I / SNR_O$$

It describes the degradation of SNR due to the element's operation and is typically given in dB: $NF \text{ dB} = SNR_I \text{ dB} - SNR_O \text{ dB}$
9. **Gain:**
 Ratio of the output signal power to the input signal power
 Expressed in dB
10. **Power Efficiency**

Efficiency of radio front end: Ratio of the radiated power to the overall power consumed by the front end.
Efficiency of power amplifier: Ratio of the output signal's power to the power consumed by the overall power amplifier.
11. **Receiver sensitivity** (given in dBm):
 The minimum signal power at the receiver needed to achieve a prescribed E_b/N_0 or a prescribed bit/packet error rate.
12. **Out of band emission:**
 The inverse to adjacent channel suppression is the out of band emission of a transmitter.
13. **Range**
 - It is considered in absence of interference
 - It depends on:
 - the transmission power
 - the antenna characteristics
 - the attenuation caused by the environment
 - the used carrier frequency
 - the modulation/coding scheme that is used
 - the bit error rate that one is willing to accept at the receiver
 - the quality of the receiver- sensitivity.
14. **Frequency stability:**
 The **frequency stability** denotes the degree of variation from nominal center frequencies when environmental conditions of oscillators like temperature or pressure change. In extreme cases, poor frequency stability can break down communication links, for example,
 When one node is placed in sunlight whereas its neighbor is currently in the shade

15. Blocking performance:

- Achieved bit error rate in the presence of an interferer.

16. Carrier sense and RSSI:

- To sense whether the wireless channel, the carrier, is busy (another node is transmitting).
- **RSSI** (Received Signal Strength Indicator): The signal strength at which an incoming data packet has been received.

The precise semantics of this carrier sense signal depends on the implementation. For example, the IEEE 802.15.4 standard distinguishes the following modes:

- The received energy is above threshold; however, the underlying signal does not need to comply with the modulation and spectral characteristics.
- A carrier has been detected, that is, some signal which complies with the modulation.
- Carrier detected and energy is present.

Also, the signal strength at which an incoming data packet has been received can provide useful information (e.g. a rough estimate about the distance from the transmitter assuming the transmission power is known); a receiver has to provide this information in the Received Signal Strength Indicator (RSSI).

17. Frequency stability: The **frequency stability** denotes the degree of variation from nominal center frequencies when environmental conditions of oscillators like temperature or pressure change. In extreme cases, poor frequency stability can break down communication links, for example, when one node is placed in sunlight whereas its neighbor is currently in the shade.

18. Voltage range: Transceivers should operate reliably over a range of supply voltages. Otherwise, inefficient voltage stabilization circuitry is required.

Transceiver Design Considerations

Question: Explain the transceiver design consideration in WSN

Transceiver structure

A fairly common structure of transceivers is into the Radio Frequency (RF) front end and the Base band part:

- The **radio frequency front end** performs analog signal processing in the actual radio frequency band, whereas
- The **baseband processor** performs all signal processing in the digital domain and communicates with a sensor node's processor or other digital circuitry.

RF front end: The **RF front end performs analog signal processing in the actual radio frequency band**, for example in the 2.4 GHz Industrial, Scientific, and Medical (ISM) band;

- Power Amplifier (PA): accepts up-converted signals from the IF or base band part and amplifies them for transmission over the antenna.
- Low Noise Amplifier (LNA): amplifies incoming signals without significantly reducing SNR. Always active and can consume a significant fraction of transceiver's energy.
- Local oscillators or voltage-controlled oscillators and mixers: used for frequency conversion from the RF spectrum to intermediate frequencies or to the

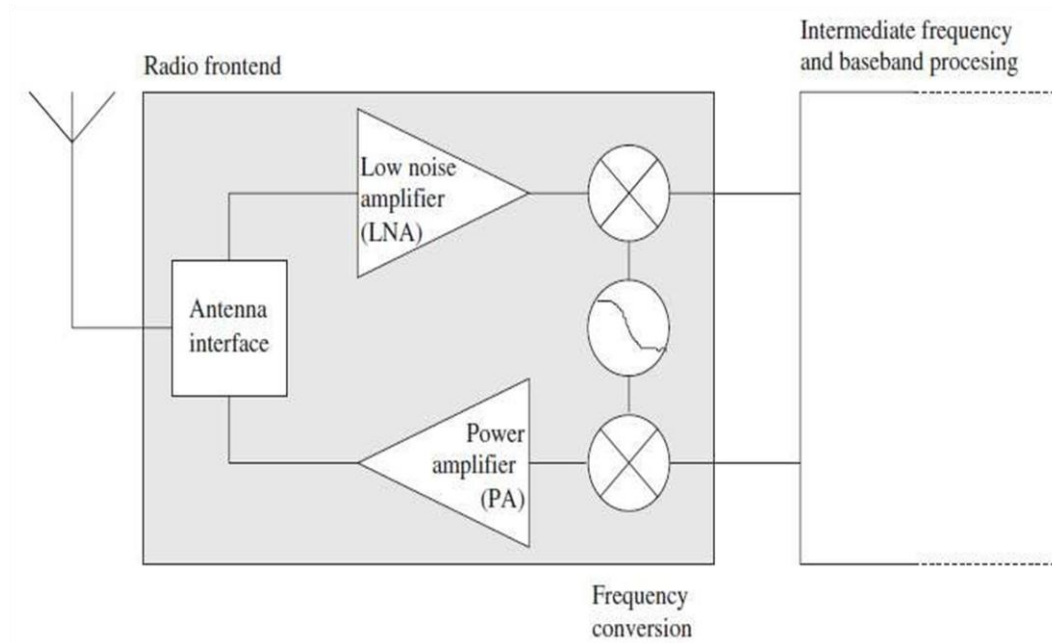


Figure 7: RF Front End

Transceiver Operational States

Question: Explain the transceiver operational states.

Many transceivers can distinguish four operational states

• **Mostly four states:**

1. **Transmit :** The transmit part of the transceiver is active and the antenna radiates energy.
2. **Receive :** The receive part is active
3. **Idle:** Transceiver is ready to receive but is not currently receiving anything. In this state, many parts of the receive circuitry are active, and others can be switched off.
4. **Sleep :** Significant parts of the transceiver are switched off. sleep states differ in the amount of circuitry switched off and in the associated recovery times and startup energy

Wakeup radio

1. **Wakeup receivers:** receiver specialized to notify an incoming packet; upon such an event, the main receiver can be turned on and perform the actual reception of the packet.

Need for Wakeup Radio

Their only purpose is to wake up the main receiver without needing (a significant amount of) power to do state a target power consumption of less than 1 μ W.

2. **Spread-spectrum transceivers:**

- _ **Simple transceiver like ASK,FSK has limited performance**
- _ High performance transceivers with reduced interference.
- _ Complex hardware and high cost.
- _

3. Ultrawideband communication:

1. Very large bandwidth is used to directly transmit digital sequence as very short impulses which occupy few Hertz up to the range of several GHz
2. Sender and receiver has to be synchronized.
3. Overlapping with conventional radio system can occur
4. Small transmitting power is needed.
5. Very high data rate can be realized over a short distance
6. can easily penetrate obstacles such as doors.
7. UWB transmitters are simpler where as receivers are more complex

Non Radio Frequency Wireless Communication

1. Optical

- Optical links can be used between sensor nodes for communication
- Very small energy per bit required for both generating and detecting optical light
- Communication can take place concurrently with only negligible interference.
- Communicating peers need to have a line of sight connection
- Strongly influenced by weather conditions

2. Ultrasound

- Used where Radio or optical waves can not penetrate the surrounding medium
- Travels relatively long distances at comparably low power.
- Used for surveillance of marine ground floor erosion
- Used as a secondary means of communication with a different propagation speed

Examples of radio transceivers RFM TR1000 family, Hardware accelerators (Mica motes), Chipcon CC1000 and CC2420 family, Infineon TDA 525x family, Ember EM2420 RF transceiver LMX3162, Conexant RDSSS9M

IEEE 802.15.4/Ember EM2420 RF transceiver

National Semiconductor LMX3162

Sensors and actuators Question: Explain the three categories of sensor with example and actuators

Without the actual sensors and actuators, a wireless sensor network would be beside the point entirely.

Passive, omnidirectional sensors These sensors can measure a physical quantity at the point of the sensor node without actually manipulating the environment by active probing – in this sense, they are passive.

Moreover, some of these sensors actually are self-powered in the sense that they obtain the energy they need from the environment – energy is only needed to amplify their analog signal.

There is no notion of “direction” involved in these measurements.

Typical examples for such sensors includes

- Thermometer
- light sensors
- vibration,
- microphones
- humidity
- mechanical stress or tension in materials,
- chemical sensors sensitive for given substances,
- smoke detectors,
- air pressure, and so on.

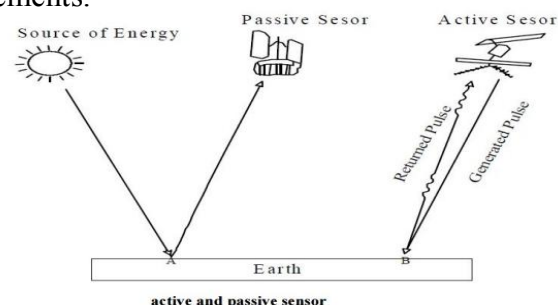


Figure 8: Passive and Active Sensor

Passive, narrow-beam sensors These sensors are passive as well, but have a well-defined notion of direction of measurement. A typical example is a camera, which can “take measurements” in a given direction, but has to be rotated if need be.

Active sensors This last group of sensors actively probes the environment, for example, a sonar or radar sensor or some types of seismic sensors, which generate shock waves by small

Overall, most of the theoretical work on WSNs considers passive, omnidirectional sensors.

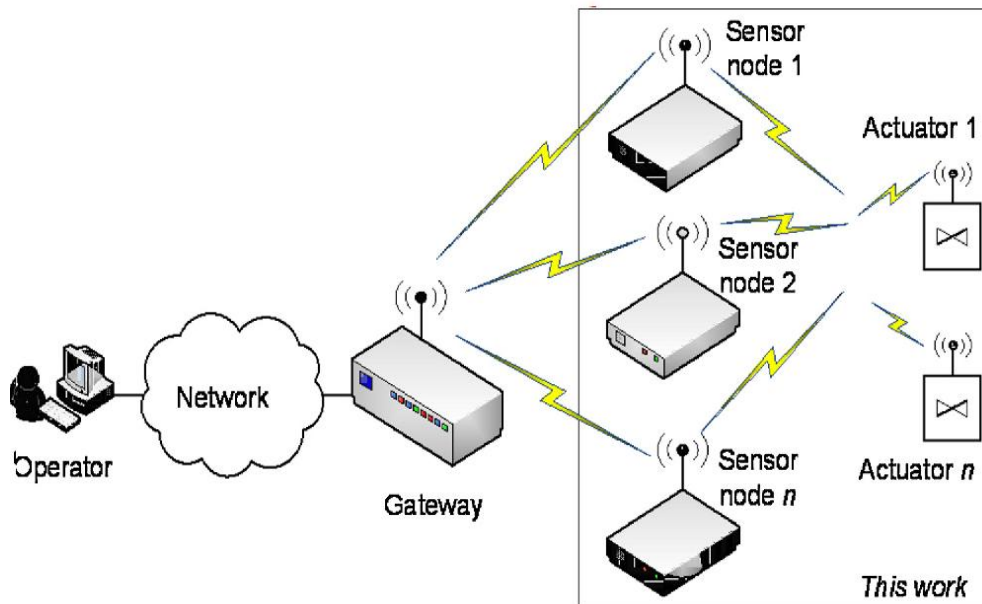


Figure 9: Integration of Sensors and Actuators

What is actuator in WSN?

In the context of sensor networks, any output device. **Actuators** allow a **WSN** node to influence its environment, providing a feedback channel through which its decisions can be enacted

What are the differences between sensors and actuators within wireless communications network?

Sensor nodes are usually low-cost, low-power, small devices equipped with limited **sensing**, data processing and **wireless communication** capabilities, while **actuator** nodes typically have stronger computation and **communication** powers and more energy budget that allows longer battery life

Power supply of sensor nodes

Traditional batteries

The power source of a sensor node is a battery, either non rechargeable (“primary batteries”) or, if an energy scavenging device is present on the node, also rechargeable (“secondary batteries”).

Capacity, Capacity under load. Self-discharge, Efficient recharging

Energy consumption of sensor nodes

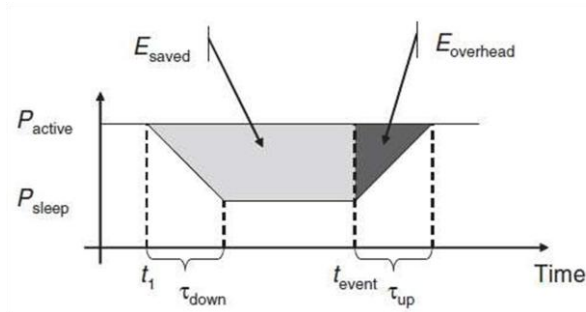
Question: Discuss about the operation states with different power consumption

The energy consumption of a sensor node must be tightly controlled for efficient operation

The main consumers of energy

are the controller, the radio front ends, to some degree the memory, and, depending on the type, the sensors

- **Energy savings and Overheads for sleep mode**



- At time t_1 : Time at which the decision whether or not a component is to be put into sleep mode should be taken to reduce power consumption from P_{active} to P_{sleep} .
- t_{event} : Time at which next event occurs
- Total energy wasted by idling at active time: $E_{active} = P_{active}(t_{event} - t_1)$
- τ_{down} : Time taken by the node to reach sleep mode from active mode
- Average power consumption during τ_{down} : $(P_{active} + P_{sleep})/2$.
- Energy consumed until t_{event} : P_{sleep}
- Energy required in sleep mode = $\tau_{down}(P_{active} + P_{sleep})/2 + (t_{event} - t_1 - \tau_{down})P_{sleep}$
- Energy required in active mode: $(t_{event} - t_1)P_{active}$
- energy saved in sleep mode:
 - $E_{saved} = (t_{event} - t_1)P_{active} - (\tau_{down}(P_{active} + P_{sleep})/2 + (t_{event} - t_1 - \tau_{down})P_{sleep})$.
- Energy consumed for turning the node active from sleep mode: $E_{overhead} = \tau_{up}(P_{active} + P_{sleep})/2$
- Switching to sleep mode is beneficial only if $E_{overhead} < E_{saved}$ or, equivalently, if the time to the next event is sufficiently large:

$$(t_{event} - t_1) > \frac{1}{2} \left(\tau_{down} + \frac{P_{active} + P_{sleep}}{P_{active} - P_{sleep}} \tau_{up} \right).$$

1. Microcontroller energy consumption

- Embedded controllers commonly implement the concept of multiple operational states

- **Examples:**

1. Intel Strong ARM

Provides three sleep modes:

1. Normal mode: All parts of the processor are fully powered. Power consumption is up to 400 mW.
2. Idle mode: Clocks to the CPU are stopped; clocks that pertain to peripherals are active. Any interrupt will cause return to normal mode. Power consumption is up to 100 mW.
3. Sleep mode: Only the real-time clock remains active. Wakeup occurs after a timer interrupt and takes up to 160ms. Power consumption is up to 50 μ W.

2. Texas Instruments MSP 430

- One Fully operational mode: consumes about 1.2 mW
- There are four sleep modes.
 - LPM4, deepest sleep mode: Consumes 0.3 μ W. Only the controller is woken up by external interrupts.
 - LPM3, next higher mode: A clock is used for scheduled wake ups.

Consumes about 6 μ W.

3. Atmel ATmega

- The Atmel ATmega 128L has six different modes of power consumption.
- Power consumption varies between 6 mW and 15 mW in idle and active modes and is about 75 μ W in power-down modes.
- **Dynamic voltage scaling**
- Another technique that is used for saving power is to use a continuous notion of functionality /power adaptation by adapting the speed with which a controller operates instead of using different operational states.
- The supply voltage can be reduced at lower clock rates still guaranteeing correct operation.

Memory:

- On-chip memory of a microcontroller and FLASH memory consumes lesser energy; hence more appropriate for WSNs.

Radio transceivers

It Has two important tasks: transmitting and receiving data between a pair of nodes.

Can operate in different modes

A radio transceiver has essentially two tasks: transmitting and receiving data between a pair of nodes. To accommodate the necessary low total energy consumption, the transceivers should be turned off most of the time and only be activated when necessary – they work at a low **duty cycle**. **But this incurs additional complexity, time and power overhead that has** to be taken into account.

Radio transceivers

Models for the energy consumption per bit for both sending and receiving

Modeling energy consumption during transmission

Energy consumed by a transmitter is due to two sources due to

RF signal generation, which mostly depends on chosen modulation and target distance and hence on the transmission power P_{tx} , that is, the power radiated by the antenna.

A second part is due to electronic components necessary for frequency synthesis, frequency conversion, filters, and so on

For discussion, let us assume that the desired transmission power P_{tx} is known

P_{tx} is a function of system aspects like energy per bit over noise E_b/N_0 , the bandwidth efficiency η_{BW} , the distance d and the path loss coefficient γ .

The transmitted power is generated by the amplifier of a transmitter. Its own power consumption P_{amp} depends on its architecture, but for most of them, their consumed power depends on the power they are to generate.

$$P_{amp} = \alpha_{amp} + \beta_{amp} P_{tx}. \quad (2.4)$$

where α_{amp} and β_{amp} are constants depending on process technology and amplifier architecture [559].

As an example, MIN and CHANDRAKASAN [563] report, for the μ AMPS-1 nodes, $\alpha_{amp} = 174 \text{ mW}$ and $\beta_{amp} = 5.0$. Accordingly, the efficiency of the power amplifier η_{PA} for $P_{tx} = 0 \text{ dBm} = 1 \text{ mW}$ radiated power is given by

$$\eta_{PA} = \frac{P_{tx}}{P_{amp}} = \frac{1 \text{ mW}}{174 \text{ mW} + 5.0 \cdot 1 \text{ mW}} \approx 0.55 \%.$$

In addition to the amplifier, other circuitry has to be powered up during transmission as well, for example, baseband processors. This power is referred to as P_{txElec} .

The energy to transmit a packet n -bits long (including all headers) then depends on how long it takes to send the packet, determined by the nominal bit rate R and the coding rate R_{code} , and on the total consumed power during transmission.

If, in addition, the transceiver has to be turned on before transmission, startup costs also are incurred (mostly to allow voltage-controlled oscillators and phase-locked loops to settle). Equation (2.5) summarizes these effects.

$$E_{tx}(n, R_{code}, P_{amp}) = T_{start} P_{start} + \frac{n}{R R_{code}} (P_{txElec} + P_{amp}). \quad (2.5)$$

To elucidate, the energy E_{rcvd} required to receive a packet has a startup component $T_{start} P_{start}$ similar to the transmission case when the receiver had been turned off (startup times are considered equal for transmission and receiving here); it also has a component that is proportional to the packet time $n / R R_{code}$.

. During this time of actual reception, receiver circuitry has to be powered up, requiring a (more or less constant) power of P_{rxElec} – for example, to drive the LNA in the RF front end.

$$E_{rcvd} = T_{start} P_{start} + \frac{n}{R R_{code}} P_{rxElec} + n E_{decBit}.$$

The decoding energy is relatively complicated to model, as it depends on a number of hardware and system parameters

Power consumption of sensor and actuators

Providing any guidelines about the power consumption of the actual sensors and actuators is next to impossible because of the wide diversity of these devices. For some of them – for example, passive light or temperature sensors – the power consumption can perhaps be ignored in comparison to other devices on a wireless node (report a power consumption of 0.6 to 1 mA for a temperature sensor). For others, in particular, active devices like sonar, power consumption can be quite considerable and must even be considered in the dimensioning of power sources on the sensor node, not to overstress batteries,

Network Architecture Question: Describe the network architecture of wireless sensor networks with diagrammatic illustration.

For this question First Draw the diagram then explain 1. Sensor Network Scenarios (Source and Sinks) 2. Gateway Concepts 3. Sensor nodes 4. Optimization of Networks 5. Mobility

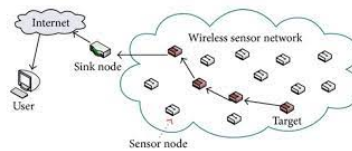
Sensor network scenarios

Optimization goals and figures of merit

Design principles for WSNs

Service interfaces of WSNs

Gateway concepts



Sensor network scenarios:

Types of Sources and Sinks

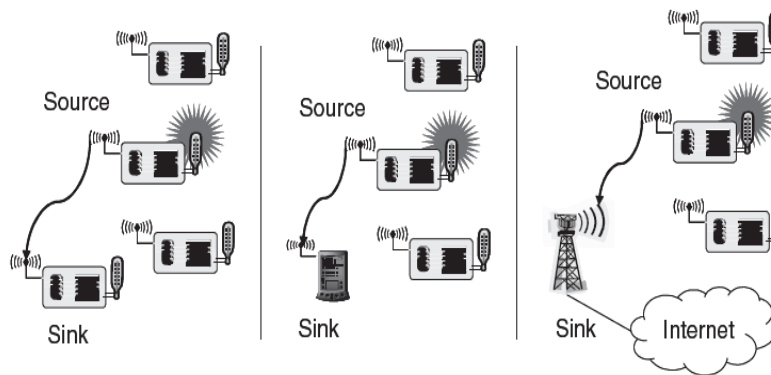
In wireless **sensor** networks (WSNs), **all** the data collected by the **sensor nodes** are forwarded to a **sink node**. Therefore, the placement of the **sink node** has a great impact on the energy consumption and lifetime of WSNs. **Sink nodes** are used to collect and pre-process data which gathered from regular **sensor nodes**

Sources: A source is any entity in the network that can provide information, that is, typically a sensor node; it could also be an actuator node that provides feedback about an operation. (Any entity that provides data/measurements)

- **Sinks:** the entity where information is required

There are essentially three options for a sink:

1. it could belong to the sensor network as such and be just another sensor/actuator node or it could be an entity outside this network
2. The sink could be an actual device
3. It could also be merely a gateway to another larger network such as the Internet, where the actual request for the information comes from some node “far away” and only indirectly connected to such a sensor network.



Three types of sinks in a very simple, single-hop sensor network

Figure 10: Three Types of Sinks

- **Applications: Usually, machine to machine, often limited amounts of data**

Single-hop vs. multi-hop networks

The simple, direct communication between source and sink is not always possible, Specifically in WSNs, which are intended to cover a lot of ground (e.g. in environmental or agriculture applications) or that operate in difficult radio environments with strong attenuation (e.g. in buildings).

To overcome such limited distances, an obvious way out is to use relay stations, with the data packets taking multi hops from the source to the sink. This concept of multihop networks in figure is particularly attractive for WSNs as the sensor nodes themselves can act as such relay nodes, foregoing the need for additional equipment.

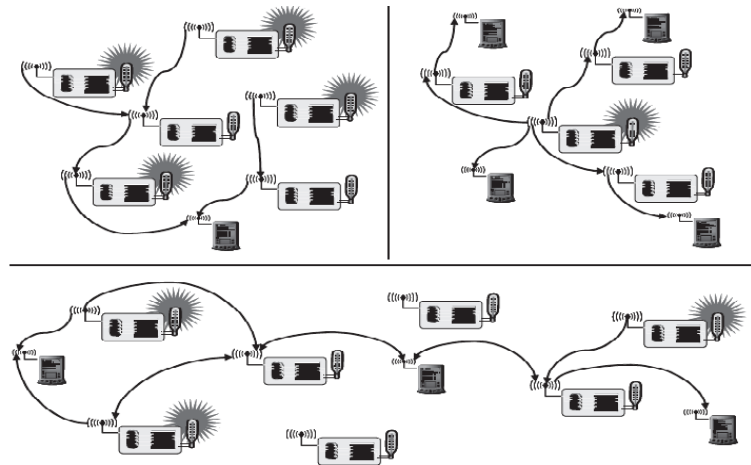
While multi hopping is an evident and working solution to overcome problems with large distances or obstacles, it has also been claimed to improve the energy efficiency of communication

In multihop attenuation of radio signals is at least quadratic in most environments (and usually larger), it consumes less energy to use relays instead of direct communication:

When targeting for a constant SNR at all receivers (assuming for simplicity negligible error rates at this SNR), the *radiated* energy required for direct communication over a distance d is cda (c some constant, $a \geq 2$ the path loss coefficient); using a relay at distance $d/2$ reduces this energy to $2c(d/2)a$.

WSN: Multiple sinks, multiple sources

In many cases, there are multiple sources and/or multiple sinks present. In the most challenging case, multiple sources should send information to multiple sinks, where either all or some of the information has to reach all or some of the sinks. Figure below illustrates these combinations.



Multiple sources and/or multiple sinks. Note how in the scenario in the lower half, both sinks and active sources are used to forward data to the sinks at the left and right end of the network

Figure 11: Multiple Source and Sinks

Types of mobility

WSN Node mobility

The wireless sensor nodes themselves can be mobile. In the face of node mobility, the network has to reorganize itself frequently enough to be able to function correctly.

- A node participating as source/sink (or destination) or a relay node might move around
- Deliberately, self-propelled or by external force; targeted or at random
- Happens in both WSN and MANET

WSN Sink mobility—The information sinks can be mobile (Figure 3.4). While this can be a special case of node mobility, the important aspect is the mobility of an information sink that is not part of the sensor network, for example, a human user requested information via a PDA while walking in an intelligent building.

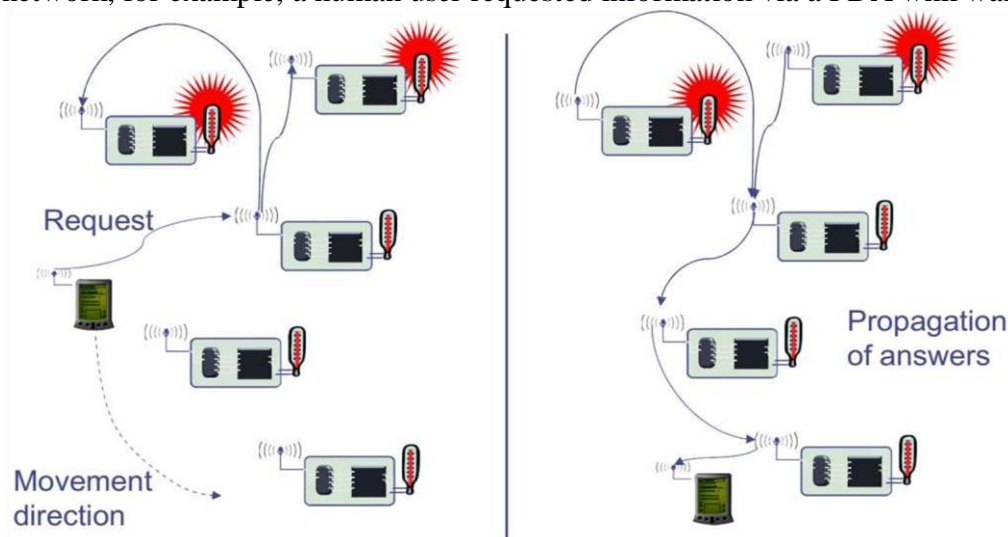
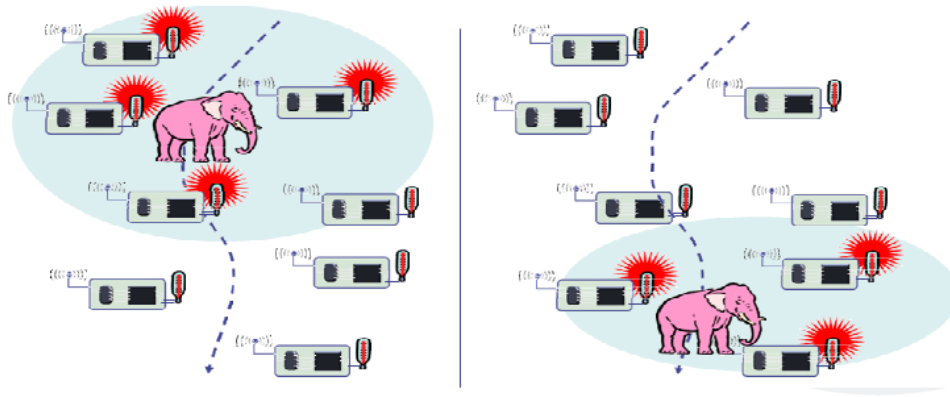


Figure 12: A mobile sink moves through a sensor network as information is being retrieved on its behalf

WSN Event mobility



Area of sensor nodes detecting an event – an elephant [378] – that moves through the network along with the event source (dashed line indicate the elephant's trajectory; shaded ellipse the activity area following or even preceding the elephant)

Figure 13: Event Mobility

In applications like event detection and in particular in tracking applications, the cause of the events or the objects to be tracked can be mobile.

In such scenarios, it is (usually) important that the observed event is covered by a sufficient number of sensors at all time. Hence, sensors will wake up around the object, engaged in higher activity to observe the present object, and then go back to sleep

Optimization goals and figures of merit Explain how optimization goals and figure of merits achieved in WSN with list of factors used to optimize the wireless sensor .

- **Quality of service**

QoS can be regarded as a low-level, networking-device-observable attribute – bandwidth, delay, jitter, packet loss rate – or as a high-level, user-observable, so-called subjective attribute like the perceived quality of a voice communication or a video transmission. high-level QoS attributes in WSN highly depend on the application. Some generic possibilities are:

- **Event detection/reporting probability:**
 - The probability that an event that actually occurred is not detected/not reported to an information sink that is interested in such an event
 - Eg: Not reporting a fire alarm to a surveillance station
- **Event classification error:** Events are not only to be detected but also to be classified, the error in classification must be small.
- **Event detection delay:** The delay between detecting an event and reporting it to any/all interested sinks
- **Missing reports:** In applications that require periodic reporting, the probability of undelivered reports should be small.

Approximation accuracy

- The Tracking applications must average/max absolute or relative error with respect to the actual function that occurs in function approximation applications
- E.g. approximating the temperature as a function of location for a given area
- **Tracking accuracy:**
 - not miss an object to be tracked.
 - The reported position should be as close to the real position as possible, and the error should be small.
- **Energy efficiency**
- **Energy per correctly received bit:** Average energy spent, counting all sources of energy consumption at all possible intermediate hops, to transport one bit of information from the source to the destination
- **Energy per reported (unique) event:** The same event is sometimes reported from various sources; this metric is usually normalized to unique event.
- **Delay/energy trade-offs:** Trade-off exists between delay and energy overhead
- **Network lifetime:** The time for which the network is operational / the time during which it is able to fulfill its tasks. Possible definitions are:
 - **Time to first node death:** Time at which the first node in the network run out of energy or fail and stop operating
 - **Network half-life:** Time at which 50% of the nodes have run out of energy and stopped operating.
 - **Time to partition:** Time at which the first partition of the network into two (or more) disconnected parts occur
- **Time to loss of coverage:** The time when for the first time any spot in the deployment region is no longer covered by any node's observations.
- **Time to failure of first event notification:** Time at which a node or a part of the network fails to deliver an event

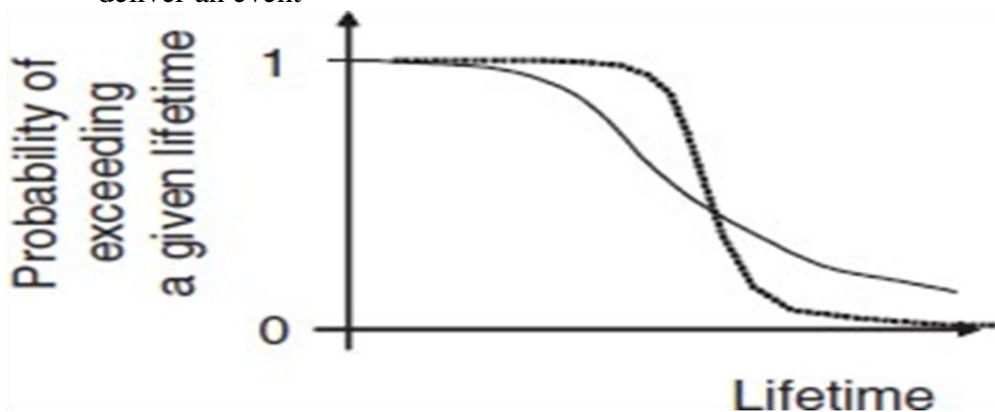


Figure 14: Two probability curves of a node exceeding a given lifetime – the dotted curve trades off better minimal lifetime against reduced maximum lifetime

Scalability

Ability to maintain performance characteristics irrespective of the size of the network

Robustness

WSN should not fail just because a limited number of nodes run out of energy, or because their environment changes and break up existing radio links between two nodes

These failures have to be compensated; for example, by finding other routes.

Text Book

- Holger Karl , Andreas willig, —Protocol and Architecture for Wireless Sensor Networks, John wiley publication, Jan 2006.

Mohamad Sathak A.J College of Engineering
Department of ECE

EC8702
ADHOC AND WIRELESS SENSOR NETWORKS

UNIT III

WSN NETWORKING CONCEPTS AND PROTOCOLS

UNIT III

WSN NETWORKING CONCEPTS AND PROTOCOLS

Topics

MAC Protocol for WSN
 Low Duty-cycle Protocols & Wakeup concepts
 Contention based Protocols (PAMAS)
 Schedule based Protocols (LEACH, IEEE 802.15.4 MAC protocol)
 Routing Protocols
 Energy Efficient Protocols
 Challenges and Issues in Transport layer protocol

FUNDAMENTALS OF MAC

What is MAC Protocol? What is MAC Protocol in WSN?

In IEEE 802 LAN/MAN standards, Medium Access Control (MAC) protocols solve a seemingly simple task: they coordinate the times where a number of nodes access a shared communication medium.

Medium Access Control (MAC) protocols is the first protocol layer above the Physical Layer (PHY) and consequently MAC protocols are heavily influenced by its properties. The fundamental task of any MAC protocol is to regulate the access of a number of nodes to a shared medium in such a way that certain application-dependent performance requirements are satisfied.

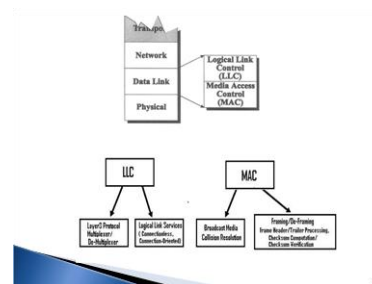
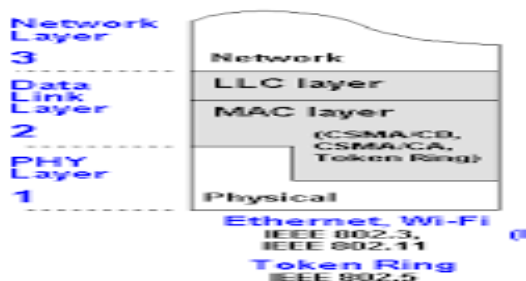
OSI

Within the OSI reference model, the MAC is considered as a part of the Data Link Layer (DLL), but there is a clear division of work between the MAC and the remaining parts of the DLL.

The MAC protocol determines for a node the points in time when it accesses the medium to try to transmit a data, control, or management packet to another node (unicast) or to a set of nodes (multicast, broadcast).

Two important responsibilities of the remaining parts of the DLL

are error control and flow control. Error control is used to ensure correctness of transmission and to take appropriate actions in case of transmission errors and flow control regulates the rate of transmission to protect a slow receiver from being overwhelmed with data.



FUNDAMENTALS OF (WIRELESS) MAC PROTOCOLS

Requirements and design constraints for wireless MAC protocols

The most important **performance requirements** are:

1. Throughput
2. efficiency
3. stability
4. fairness
5. low access delay (time between packet arrival and first attempt to transmit it)
6. low transmission delay (time between packet arrival and successful delivery), as well as a low overhead.
7. The overhead in MAC protocols can result from per-packet overhead (MAC headers and trailers), collisions, or from exchange of extra control packets
8. Collisions: It can happen if the MAC protocol allows two or more nodes to send packets at the same time. Collisions can result in the inability of the receiver to decode a packet correctly, causing the upper layers to perform a retransmission.
9. For time-critical applications, it is important to provide deterministic or stochastic guarantees on delivery time or minimal available data rate. Sometimes, preferred treatment of important packets over unimportant ones is required, leading to the concept of **priorities**

The main issues need to be addressed while designing a MAC protocol for WSN

The operation and performance of MAC protocols is heavily influenced by the properties of the underlying physical layer. Since WSNs use a wireless medium, they inherit all the well-known problems of wireless transmission. One problem is time-variable, and sometimes quite high, error rates, which is caused by physical phenomena like slow and fast fading, path loss, attenuation, and man-made or thermal noise . Depending on modulation schemes, frequencies, distance between transmitter and receiver, and the propagation environment, instantaneous bit error rates in the range of 10^{-3} . . . 10^{-2} can easily be observed

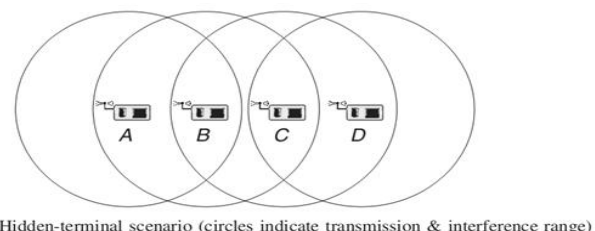
- **Hidden and exposed terminal problems:**
 - **Hidden nodes:**
 - **Hidden stations:** Carrier sensing may fail to detect another station. For example, A and D.
 - **Fading:** The strength of radio signals diminished rapidly with the distance from the transmitter. For example, A and C.
 - **Exposed nodes:**
 - **Exposed stations:** B is sending to A. C can detect it. C might want to send to E but conclude it cannot transmit because C hears B.
 - **Collision masking:** The local signal might drown out the remote transmission.
- **Error-Prone Shared Broadcast Channel**
- **Distributed Nature/Lack of Central Coordination**
- **Mobility of Nodes:** Nodes are mobile most of the time.
- **Bandwidth efficiency** is defined as the ratio of the bandwidth used for actual data transmission to the total available bandwidth. The MAC protocol for ad-hoc networks should maximize it.
- **Quality of service** support is essential for time-critical applications. The MAC protocol for ad-hoc networks should consider the constraint of ad-hoc networks.
- **Synchronization** can be achieved by exchange of control packets.

Hidden-terminal problem-Explanation

The hidden-terminal problem occurs specifically for the class of Carrier Sense Multiple Access (CSMA) protocols, where a **node senses the medium before starting to transmit a packet**. If the medium is found to be busy, the node defers its packet to avoid a collision and a subsequent retransmission.

Consider the example in Figure 5.1. Here, we have three nodes A, B, and C that are arranged such that A and B are in mutual range, B and C are in mutual range, but A and C cannot hear each other. Assume that A starts to transmit a packet to B and sometime later node C also decides to start a packet transmission. A carrier-sensing operation by C shows an idle medium since C cannot hear A's signals. When C starts its packet, the signals collide at B and both packets are useless. Using simple CSMA in a hidden-terminal scenario thus leads to needless collisions

A carrier sensing may fail to detect another station



Hidden-terminal scenario (circles indicate transmission & interference range)

Figure 1: Hidden node Problem

In the **exposed-terminal scenario**, B transmits a packet to A, and some moment later, C wants to transmit a packet to D . Although this would be theoretically possible since both A and D would receive their packets without distortions, the carrier-sense operation performed by C suppresses C 's transmission and bandwidth is wasted. Using simple CSMA in an exposed terminal scenario thus leads to needless waiting.

Two solutions to the hidden-terminal and exposed-terminal problems are busy-tone solutions and the RTS/CTS handshake used in the IEEE 802.11 WLAN standard

Another important problem arises when there is no dedicated frequency band allocated to a wireless sensor network and the WSN has to share its spectrum with other systems. Because of license-free operations, many wireless systems use the so-called ISM bands, with the 2.4 GHz ISM band being a prime example. This specific band is used by several systems, for example, the IEEE 802.11/IEEE 802.11b WLANs , Bluetooth , and the IEEE 802.15.4 WPAN. Therefore, the issue of coexistence of these systems arises

IMPORTANT CLASSES OF MAC PROTOCOLS (Fundamentals of MAC)

Fixed assignment protocols

In this class of protocols, the available resources are divided between the nodes such that the resource assignment is long term and each node can use its resources exclusively without the risk of collisions.

Long term means that the assignment is for durations of minutes, hours, or even longer, as opposed to the short-term case where assignments have a scope of a data burst, corresponding to a time horizon of perhaps (tens of) milliseconds

Typical protocols of this class are TDMA, FDMA, CDMA, and SDMA

- TDMA (time Division Multiple Access)
 - Subdivides the time axis into superframes
 - Each superframe is subdivided to time slots
 - Slots assigned to nodes and each node can xmit periodically
 - Requires tight time synchronization (-)
- FDMA (Frequency Division Multiple Access)
 - Frequency band is divided to a number of subchannels
 - Each subchannel is assigned to a node
 - Requires frequency synchronization
 - FDMA xceiver is more complex than TDMA xceiver
- CDMA (Code Division Multi Access) Codes are used to separate a spreaded signal

Demand assignment protocols

In demand assignment protocols, the exclusive allocation of resources to nodes is made on a short-term basis, typically the duration of a data burst.

This class of protocols can be broadly subdivided into centralized and distributed protocols. In central control protocols (examples are the HIPERLAN/2 protocol, DQRUMA, or the MASCARA protocol, polling schemes can also be subsumed under this class), the nodes send out requests for bandwidth allocation to a central node that either accepts or rejects the requests.

. In case of successful allocation, a confirmation is transmitted back to the requesting node along with a description of the allocated resource,

An example of *distributed demand assignment protocols* are *token-passing protocols* like

IEEE 802.4 Token Bus

Key points

- Resources are allocated on a short term basis
- Centralized and distributed versions are possible
- Central
 - Nodes request a resource (e.g. time slot) from a central server
 - Waits for ACK and then transmits
 - Polling by central station is possible
 - **Piggybacking may be used (request on data packets)**
 - Central server to be switched on always
 - Central node requires a lot of energy

Central node may be rotated (LEACH)

Distributed Demand Assignment Protocols

- Token passing (IEEE 802.4) may be used
- Token is passed among stations in a logical ring
- Ring management needed
 - Include/exclude nodes from the ring
 - Correct lost tokens
- In WSNs, maintenance is difficult
 - Channel errors
 - Node receiver must be switched on all the time (energy ..) due to variable token delivery times

Random access protocols

The nodes are uncoordinated, and the protocols operate in a fully distributed manner. Random access protocols often incorporate a random element, for example, by exploiting random packet arrival times, setting timers to random values, and so on. One of the first and still very important random.

access protocols is the ALOHA or slotted ALOHA protocol, developed at the University of Hawaii [5]. In the pure ALOHA protocol, a node wanting to transmit a new packet transmits it *immediately*. There is no coordination with other nodes and the protocol thus accepts the risk of collisions at the receiver.

Key points

- Fully distributed
- ALOHA (more on this later)
- CSMA based
 - Listen to the medium
 - If idle, transmit
 - If busy, wait (p-persistent, non-persistent etc.)
- RTS/CTS based on MACAW protocol (more later)

CSMA FOR UNDERSTANDING

In the class of **CSMA protocols** a transmitting node tries to be respectful to ongoing transmissions. First, the node is required to listen to the medium; this is called **carrier sensing**. If the medium is found to be idle, the node starts transmission. If the medium is found busy, the node defers its transmission for an amount of time determined by one of several possible algorithms. For example, in **nonpersistent CSMA**, the node draws a random waiting time, after which the medium is sensed again. Before this time, the node does not care about the state of the medium. In different **persistent CSMA** variants, after sensing that the medium is busy, the node awaits the end of the ongoing transmission and then behaves according to a **backoff algorithm**. In many of these backoff algorithms, the time after the end of the previous frame is subdivided into time slots. In p -persistent CSMA, a node starts transmission in a time slot with some probability p and with probability $1 - p$ it waits for another slot.³ If some other node starts to transmit in the meantime, the node defers and repeats the whole procedure after the end of the new frame. A small value of p makes collisions unlikely, but at the cost of high access delays. The converse is true for a large value of p .

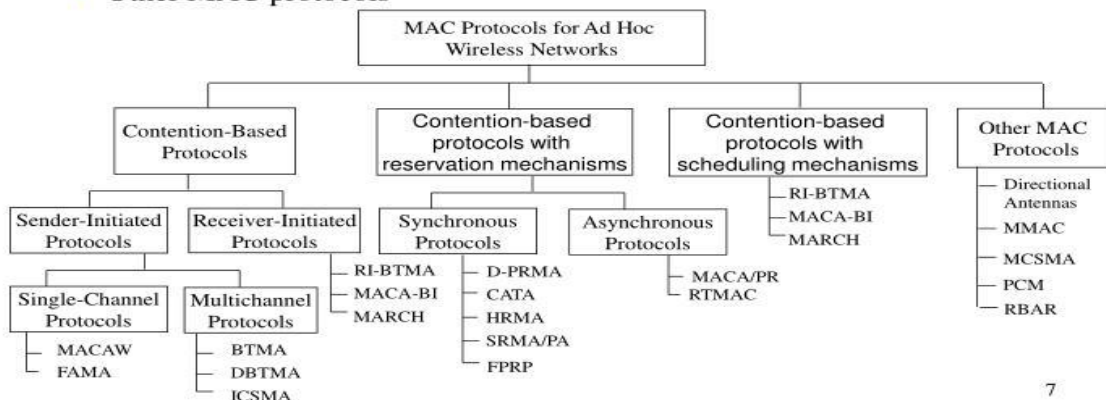
Carrier-sense protocols are susceptible to the hidden-terminal problem since interference at the receiver cannot be detected by the transmitter. This problem may cause packet collisions. The energy spent on collided packets is wasted and the packets have to be retransmitted.

Several approaches have appeared to solve or at least to reduce the hidden-terminal problem;

Two important ones: the busy-tone solution and the RTS/CTS handshake

Classifications of MAC protocols

- Ad hoc network MAC protocols can be classified into three types:
 - Contention-based protocols
 - Contention-based protocols with reservation mechanisms
 - Contention-based protocols with scheduling mechanisms
 - Other MAC protocols



MAC protocols for wireless sensor networks

Explain the MAC protocols for WSn (First Write what is MAC protocol, Energy Problems on WSN MAC, Requirements and design constraints for wireless MAC protocols)

Design goals of a MAC Protocol

- Design goals of a MAC protocol for ad hoc wireless networks
 - The operation of the protocol should be distributed.
 - The protocol should provide QoS support for real-time traffic.
 - The access delay, which refers to the average delay experienced by any packet to get transmitted, must be kept low.
 - The available bandwidth must be utilized efficiently.
 - The protocol should ensure fair allocation of bandwidth to nodes.
 - Control overhead must be kept as low as possible.
 - The protocol should minimize the effects of hidden and exposed terminal problems.
 - The protocol must be scalable to large networks.
 - It should have power control mechanisms.
 - The protocol should have mechanisms for adaptive data rate control.
 - It should try to use directional antennas.
 - The protocol should provide synchronization among nodes.

6

NEED FOR MAC LAYER

- Controlling when to send a packet and when to listen for a packet are perhaps the two most important operations in a wireless network
 - Especially, idly waiting wastes huge amounts of energy
- schemes for medium access control that are
 - Suitable to mobile and wireless networks
 - Emphasize energy-efficient operation
 - The **MAC layer** is responsible for moving data packets to and from one Network Interface Card (NIC) to another across a shared channel. ... The **MAC** sublayer uses **MAC** protocols to ensure that signals sent from different stations across the same channel don't collide.

MAC vs. LLC

- MAC deals with when and how to access medium
 - unicast and broadcast message sending
- LLC main task is Error and Flow Control (next chapter)
- Energy efficiency is important

- Put a node to SLEEP whenever possible
- Contention based and Scheduling based approaches

IEEE 802.15.4 combines both

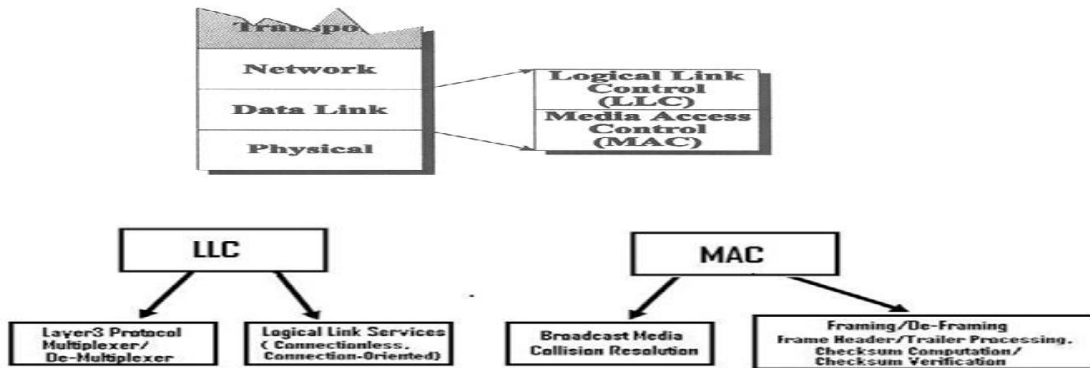


Figure 2: MAC VS LLC

1. MEDIUM ACCESS CONTROL PROTOCOL

MAC Goals

- Performance Requirements
 - throughput
 - stability
 - low access delay
 - low overhead

Overhead due to :

- per-packet overhead (headers , trailers)
- collisions – happens when 2 nodes xmit
- exchange of extra control packets
- use priority based xmission for real-time

Medium access in wireless networks is difficult mainly because of

- Impossible (or very difficult) to send and receive at the same time
- Interference situation at receiver is what counts for transmission success, but can be very different from what sender can observe
- High error rates (for signaling packets) compound the issues
- Requirement
- As usual: high throughput, low overhead, low error rates, ...
- Additionally: energy-efficient, handle switched off devices!

2. Requirements for MAC in WSNs

- Energy consideration was not a primary concern before
- Operation influenced by physical layer
- WSN medium has high error rate, path loss, noise, attn.

- Event based and time based applications may require different MAC protocols
- Hidden terminal problem should be addressed

3. Multiple Access Control (MAC) Protocols

- MAC allows multiple users to share a common channel.
- ***Conflict-free protocols*** ensure successful transmission. Channel can be allocated to users statically or dynamically.
- Only static conflict-free protocols are used in cellular mobile communications
 - Frequency Division Multiple Access (FDMA): provides a fraction of the frequency range to each user for all the time
 - Time Division Multiple Access (TDMA) : The entire frequency band is allocated to a single user for a fraction of time
 - Code Division Multiple Access (CDMA) : provides every user a portion of bandwidth for a fraction of time
 - ***Contention based protocols*** must prescribe ways to resolve conflicts
 - Static Conflict Resolution: Carrier Sense Multiple Access (CSMA)
 - Dynamic Conflict Resolution: the Ethernet, which keeps track of various system parameters, ordering the users accordingly

4. Energy Problems on WSN MAC

- Collision
- Over Hearing
- Idle Listening
- Protocol Overhead

Collisions

Collisions incur useless receive costs at the destination node, useless transmit costs at the source node, and the prospect to expend further energy upon packet retransmission. Hence, collisions should be avoided, either by design (fixed assignment/TDMA or demand assignment protocols) or by appropriate collision avoidance/hidden-terminal procedures in CSMA protocols. However, if it can be guaranteed for the particular sensor network application at hand that the load is always sufficiently low, collisions are no problem

Protocol overhead

Protocol overhead is induced by MAC-related control frames like, for example, RTS and CTS packets or request packets in demand assignment protocols, and furthermore by per-packet overhead like packet headers and trailers.

Idle listening

A node being in idle state is ready to receive a packet but is not currently receiving anything. This readiness is costly and useless in case of low network loads; for many radio modems, the idle state still consumes significant energy. Switching off the transceiver is a solution; however, since mode changes also cost energy, their frequency should be kept at “reasonable” levels. TDMA-based protocols offer an implicit solution to this problem, since a node having assigned a time slot and exchanging (transmitting/receiving) data *only during this slot can safely switch off its transceiver in all other time slots*

5. Requirements for MAC in WSNs

- Energy consideration was not a primary concern before
- Operation influenced by physical layer
- WSN medium has high error rate, path loss, noise, attn.
- Event based and time based applications may require different MAC protocols
- Hidden terminal problem nad Exposed node problem should be addressed

6.RTS/CTS HANDSHAKE IN MAC

The **RTS/CTS handshake** as used in IEEE 802.11 is based on the MACAW protocol

Main options to shut up senders

- Receiver informs potential interferers *while* a reception is on-going
 - By sending out a signal indicating just that
 - Problem: Cannot use same channel on which actual reception takes place
 - ! Use separate channel for signaling
 - *Busy tone* protocol
- Receiver informs potential interferers *before* a reception is on-going
 - Can use same channel
 - Receiver itself needs to be informed, by sender, about impending transmission
 - Potential interferers need to be aware of such information, need to store it

Receiver informs interferers before transmission – MACA

- ▶ Sender B asks receiver C whether C is able to receive a transmission
Request to Send (RTS)
- ▶ Receiver C agrees, sends out a *Clear to Send (CTS)*
- ▶ Potential interferers overhear either RTS or CTS and know about impending transmission and for how long it will last
 - Store this information in a *Network Allocation Vector*
- ▶ B sends, C acks

- *MACA protocol* (used e.g. in *IEEE 802.11*)

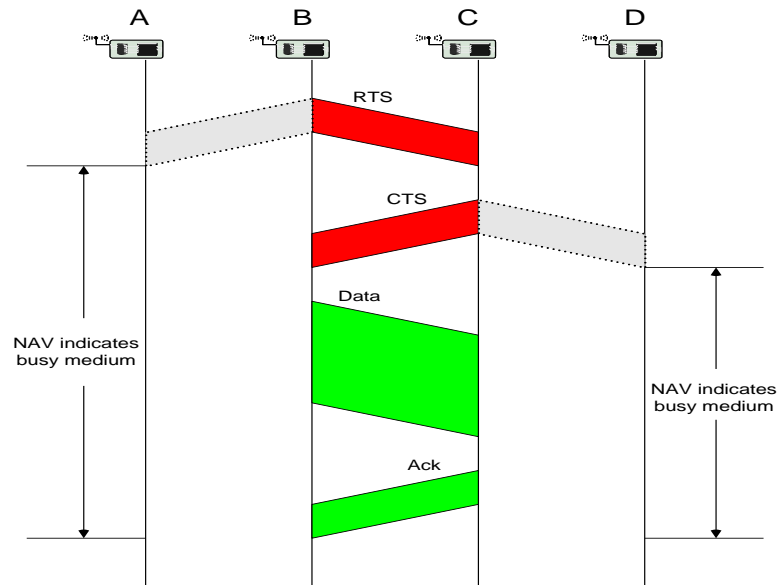
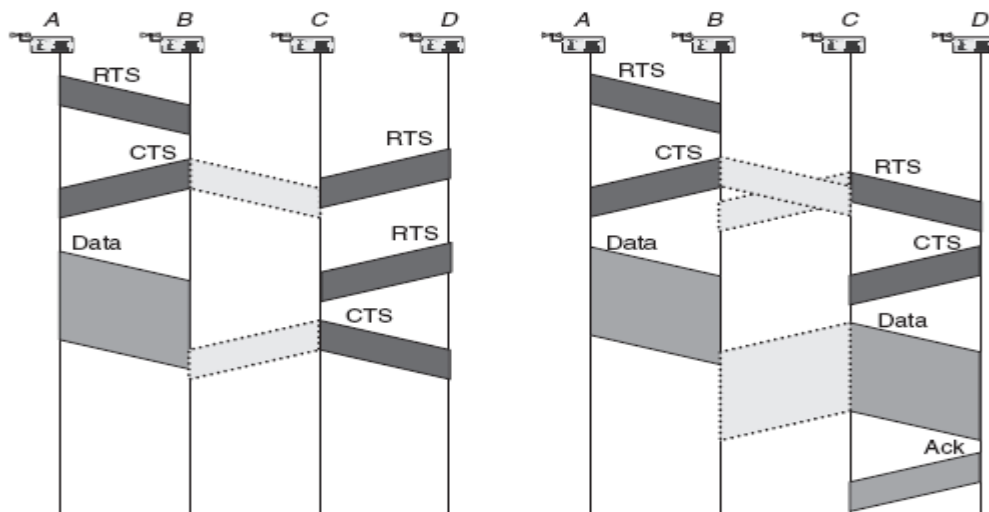


Figure 3: RTS/CTS Hand shake in IEEE802.11

- RTS/CTS ameliorate, but do not solve hidden/exposed terminal problems
- Example problem cases:



Two problems in RTS/CTS handshake [668]

Figure 3:

LOW DUTY CYCLE & WAKEUP CONCEPTS: (STEM,S-MAC, Mediation Device Protocol, Walkup Radio Concepts)

Low duty cycle protocols try to avoid spending (much) time in the idle state and to reduce the communication activities of a sensor node to a minimum. In an ideal case, the sleep state is left only when a node is about to transmit or receive packets. A concept for achieving this, the wakeup radio

In several protocols, a **periodic wakeup** scheme is used. Such schemes exist in different flavors. One is the **cycled receiver** approach, illustrated in Figure below. 4

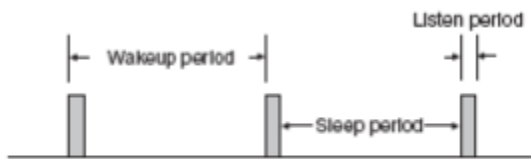


Figure 4 Periodic wakeup scheme

In this approach, nodes spend most of their time in the sleep mode and wake up periodically to *receive* packets from other nodes. Specifically, a node *A* listens onto the channel during its **listen period** and goes back into sleep mode when no other node takes the opportunity to direct a packet to *A*.

A potential transmitter *B* must acquire knowledge about *A*'s listen periods to send its packet at the right time – this task corresponds to a *rendezvous*. This rendezvous can, for example, be accomplished by letting node *A* transmit a short beacon at the beginning of its listen period to indicate its willingness to receive packets.

Another method is to let node *B* send frequent request packets until one of them hits *A*'s listen period and is really answered by *A*. However, in either case, node *A* only *receives* packets during its listen period. If node *A* itself wants to transmit packets, it must acquire the target's listen period.

A whole cycle consisting of sleep period and listen period is also called a **wakeup period**. **The ratio of the listen period length to the wakeup period length is also called the node's duty cycle.**

Some important observations:

By choosing a small duty cycle, the transceiver is in sleep mode most of the time, avoiding idle listening and conserving energy.

- By choosing a small duty cycle, the traffic directed from neighboring nodes to a given node concentrates on a small time window (the listen period) and in heavy load situations significant competition can occur.
- Choosing a long sleep period induces a significant **per-hop latency**, since a prospective transmitter node has to wait an average of half a sleep period before the receiver can accept packets.

In the multihop case, the per-hop latencies add up and create significant end-to-end latencies.

- Sleep phases should not be too short lest the start-up costs outweigh the benefits.

The Sparse Topology and Energy Management (STEM)

STEM protocol does not cover all aspects of a MAC protocol but provides a solution for the idle listening problem. STEM targets networks that are deployed to wait for and report on the behavior of a certain event,

IDEA:

- Periodic wake up and listen
- Sleep when no packet to send and receive

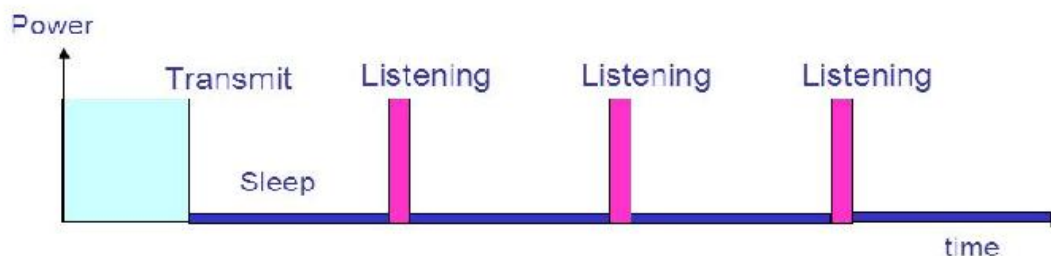


Figure 5 STEM

STEM Explanation

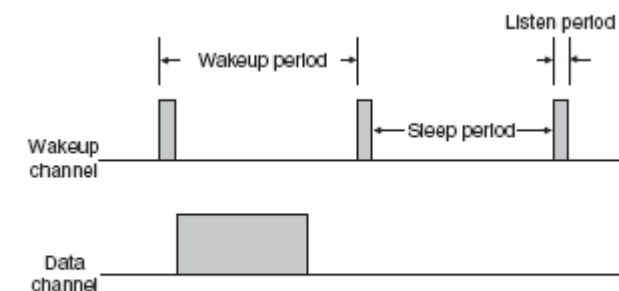


Figure 6. STEM duty cycle for a single node

Two different channels are used, requiring two transceivers in each node: the **wakeup channel** and the **data channel**. The data channel is always in sleep mode, except when transmitting or receiving data packets. The underlying MAC protocol is executed solely on the data channel during the transfer states. On the wakeup channel the time is divided into fixed-length **wakeup periods** of length T .

A wakeup period is subdivided into a **listen period** of length $TR_x - T$ and a sleep period, where the wakeup channel transceiver enters sleep mode, too. If a node enters the listen period, it simply switches on its receiver for the wakeup channel and waits for incoming signals.

If nothing is received during time TR_x , the node returns into sleep mode. Otherwise the transmitter and receiver start a packet transfer on the data channel. There are two different variants for the transmitter to acquire the receiver's attention:

There are two different variants for the transmitter to acquire the receiver's attention:

STEM –B and STEM –T

STEM-B:

Strategy : sensor nodes wakes a neighbour by transmitting a beacon
(no RTS/CTS)

advantages:

- Lower Latency

Disadvantages:

- More complex
- High energy consumption

STEM-T:

Strategy : sensor nodes wakes a neighbour by transmitting a tone of sufficient length that destination will have a high probability of sensing

- Busy tone contains no destination address

Disadvantages:

- High latency
- Results in overhearing

The S-MAC (Sensor-MAC)

What is SMAC?

The S-MAC (Sensor-MAC) protocol provides mechanisms to circumvent idle listening, collisions, and overhearing. As opposed to STEM, it does not require two different channels. S-MAC adopts a periodic wakeup scheme, that is, each node alternates between a fixed-length listen period and a fixed-length sleep period according to its schedule.

What is the drawback of SMAC?

S-MAC has one major drawback: it is hard to adapt the length of the wakeup period to changing load situations, since this length is essentially fixed, as is the length of the listen period.

S-MAC adopts a periodic wakeup scheme, that is, each node alternates between a fixed-length listen period and a fixed-length sleep period according to its **schedule**, compare Figure .6. However, as opposed to STEM, the listen period of S-MAC can be used to receive *and transmit* packets.

S-MAC attempts to coordinate the schedules of neighboring nodes such that their listen periods start at the same time. A node x 's listen period is subdivided into three different phases:

- ▶ Idea: Switch nodes off, ensure that neighboring nodes turn on simultaneously to allow packet exchange

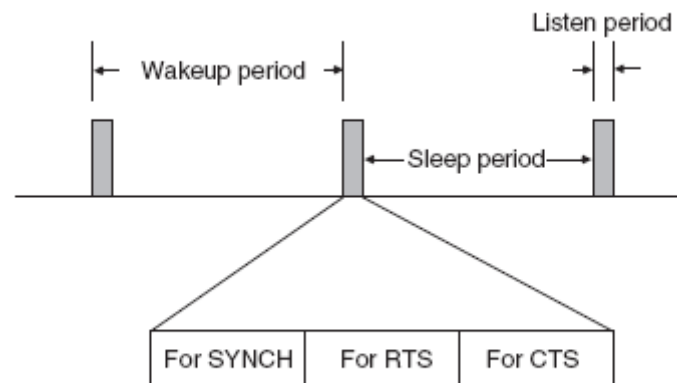


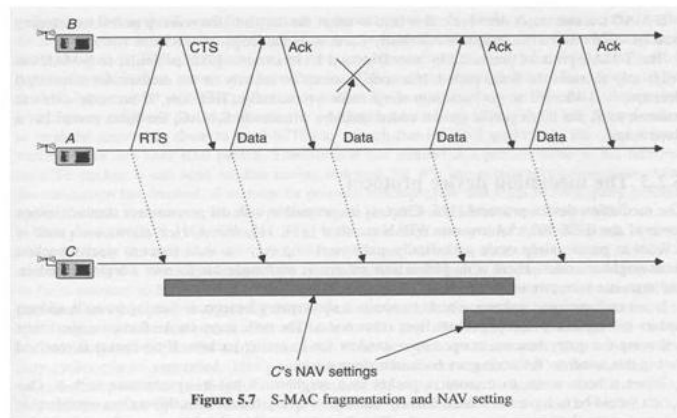
Figure .6 S-MAC principle

- Only in these **active periods**, packet exchanges happen
- Need to also exchange wakeup schedule between neighbors
- When awake, essentially perform RTS/CTS
- ▶ Use SYNCH, RTS, CTS phases

- Set of nodes periodically become active/sleep in a synchronized fashion. This set of nodes is called Virtual Cluster
- Active periods are divided into two periods, one for exchanging SYNC packets and other for exchanging DATA packets
 - Active periods are fixed to a pre-recalculated size optimized for expected traffic.
- Collisions are avoided by RTS/CTS mechanism

SMAC fragmentation and NAV setting

- S-MAC adopts a message passing concept
 - long messages are broken into small frames
 - only one RTS/CTS communication for each messages
 - each frame is acknowledged separately
 - each frame contains the information about the message length
- The NAV (not available) variable of suppressed neighbors is adjusted appropriately
- Problems: Fairness



The approach taken by S-MAC reduces the latency of complete messages by suppressing intertwined transmissions of other packets. Therefore, in a sense, this protocol is unfair because single nodes can block the medium for long time

S-MAC has one major drawback: it is hard to adapt the length of the wakeup period to changing load situations, since this length is essentially fixed, as is the length of the listen period

MEDIATION DEVICE PROTOCOL

What is the mediation device protocol?

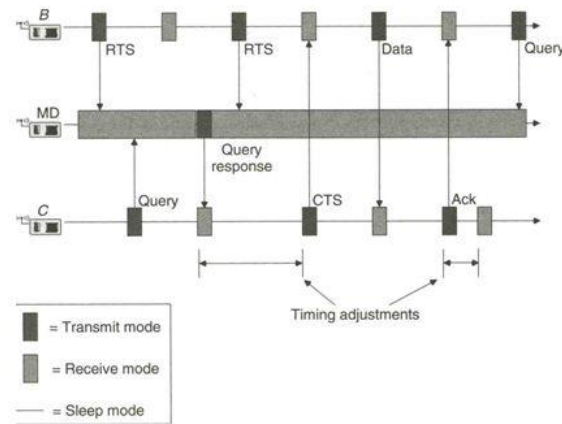
The mediation device protocol is compatible with the peer-to-peer communication mode of the IEEE 802.15.4 low-rate WPAN standard.

It allows each node in a WSN to go into sleep mode periodically and to wake up only for short times to receive packets from neighbor nodes.

There is no global time reference, each node has its own sleeping schedule, and does not take care of its neighbors sleep schedules

Mediation Device Protocol

- Goal: Avoid useless listening on the channel for messages
- Uses: mediation device (MD) which is available all the time
- Protocol
 - Sender B sends RTS to MD
 - MD stores this information
 - Receiver C sends query to MD
 - MD tells receiver C when to wake up
 - C sends CTS to B (now in sync)
 - B sends data
 - C acknowledges
 - C returns to old timing
- Main disadvantage:
 - MD has to be energy independent
 - Solution: Distributed Mediation Device Protocol
 - Nodes randomly wake up and serve as mediation device
- Problem: no guarantees on full coverage of MD



34

What are the advantages of Mediation device protocol?

This protocol has some advantages.

First, it does not require any time synchronization between the nodes, only the mediation device has to learn the periods of the nodes.

Second, the protocol is asymmetric in the sense that most of the energy burden is shifted to the mediation device, which so far is assumed to be power unconstrained. The other nodes can be in the sleep state most of the time and have to spend energy only for the periodic beacons. Even when a transmitter wants to synchronize with the receiver, it does not have to wait actively for the query beacon, but can go back to sleep and wait for the mediation device to do the synchronization work. This way very low duty cycles can be supported.

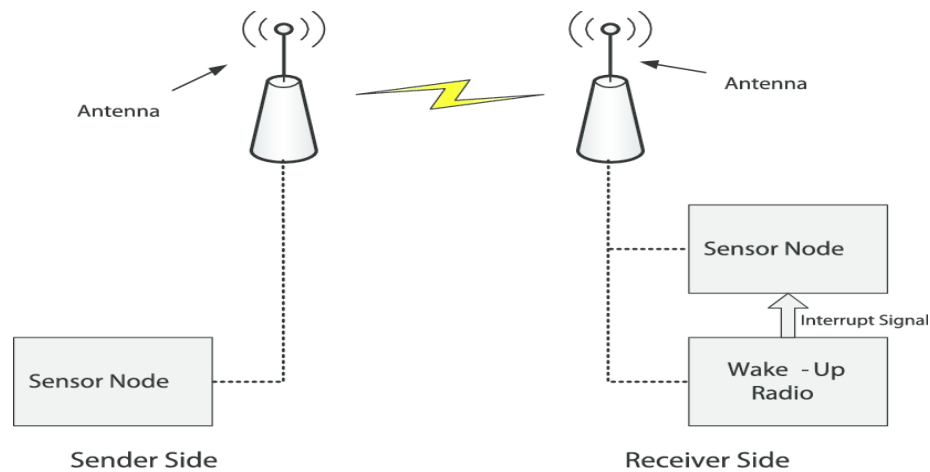
What are the drawback of MDP

This protocol has also some drawbacks:

The nodes transmit their query beacons without checking for ongoing transmissions and, thus, the beacons of different nodes may collide repeatedly when nodes have the same period and their wakeup periods overlap

The main drawbacks, however, are the assumptions that: (i) the mediation device is energy unconstrained, which does not conform to the idea of a “simply thrown out” wireless sensor network, and (ii) there are sufficient mediation devices to cover all nodes. The distributed mediation device protocol deals with these problems in a probabilistic manner

Concept of Wakeup radio



What is the advantage of Wakeup radio concept?

The wakeup radio concept has the significant advantage that only the low-power wakeup transceiver has to be switched on all the time while the much more energy consuming data transceiver is non sleeping if and only if the node is involved in data transmissions. Furthermore, this scheme is naturally traffic adaptive , that is, the MAC becomes more and more active as the traffic load increases

However, there are also some drawbacks.

1. There is no real hardware yet for such an ultralow power wakeup transceiver. Second, the range of the wakeup radio and the data radio should be the same. If the range of the wakeup radio is smaller than the range of the data radio, possibly not all neighbor nodes can be woken up. On the other hand, if the range of the wakeup radio is significantly larger, there can be a problem with local addressing schemes . These schemes do not use globally or network wide-unique addresses but only locally unique addresses, such that no node has two or more one-hop neighbors with the same address.

CONTENTION-BASED PROTOCOL (CBP)

1.CSMA 2. PMAS

A **contention-based protocol** (CBP) is a communications protocol for operating wireless telecommunication equipment that allows many users to use the same radio channel without pre-coordination. The "listen before talk" operating procedure in [IEEE 802.11](#) is the most well known contention-based protocol.

A protocol that allows multiple users to share the same spectrum by defining the events that must occur when two or more transmitters attempt to simultaneously access the same channel and establishing rules by which a transmitter provides reasonable opportunities for other transmitters to operate.

Such a protocol may consist of procedures for initiating new transmissions, procedures for determining the state of the channel (available or unavailable), and procedures for managing retransmissions in the event of a busy channel

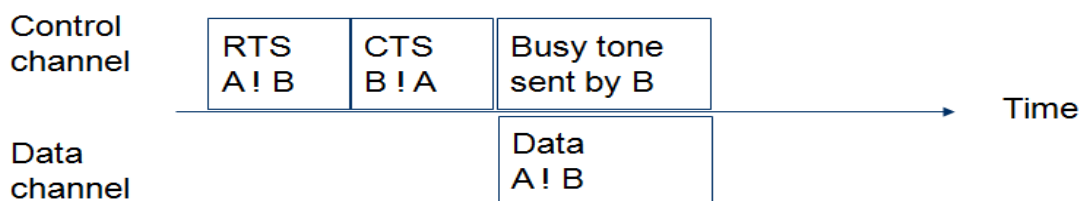
THE PAMAS PROTOCOL (POWER AWARE MULTIACCESS WITH SIGNALING)

It is originally designed for ad hoc networks. It provides a detailed overhearing avoidance mechanism while it does not consider the idle listening problem. The protocol combines the busy-tone solution and RTS/CTS handshake similar to the MACA protocol (MACA uses no final acknowledgment packet).

Feature: Feature of PAMAS is that it uses two channels: a **data channel** and a **control channel**. All the signaling packets (RTS, CTS, busy tones) are transmitted on the control channel, while the data channel is reserved for data packets.

Power Aware Multiaccess with Signaling – PAMAS

- Idea: combine busy tone with RTS/CTS
 - Results in detailed overhearing avoidance, does not address idle listening
 - Uses separate **data** and **control channels**
- Procedure
 - Node A transmits RTS on control channel, does not sense channel
 - Node B receives RTS, sends CTS on control channel if it can receive and does not know about ongoing transmissions
 - B sends busy tone as it starts to receive data



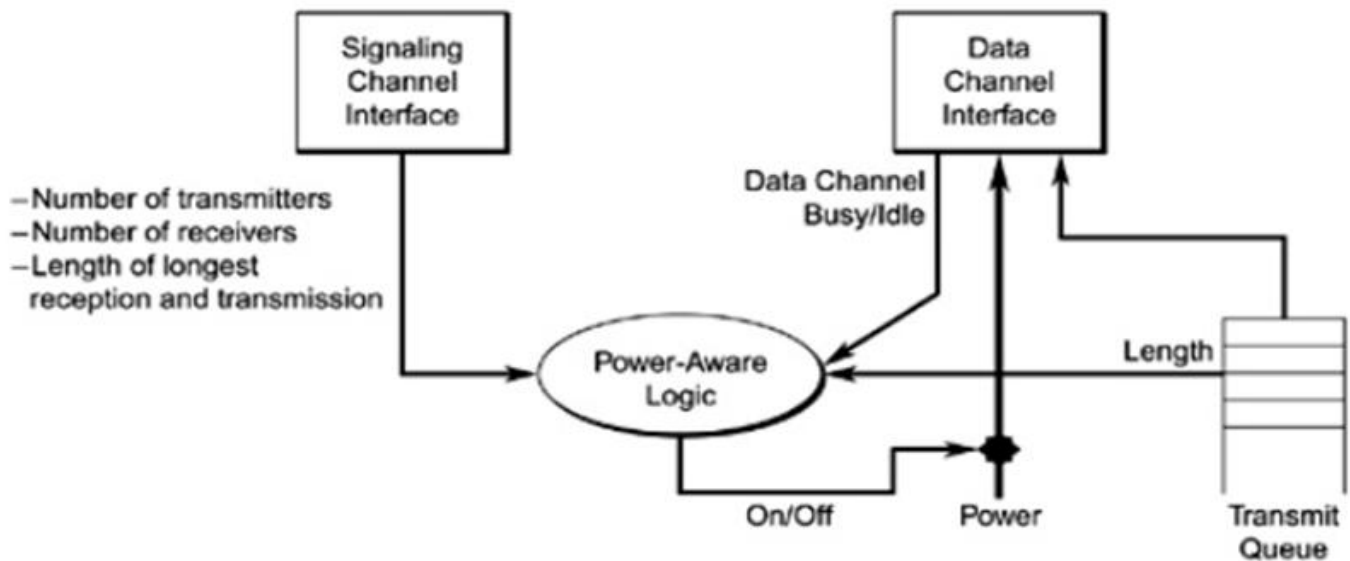
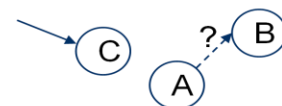
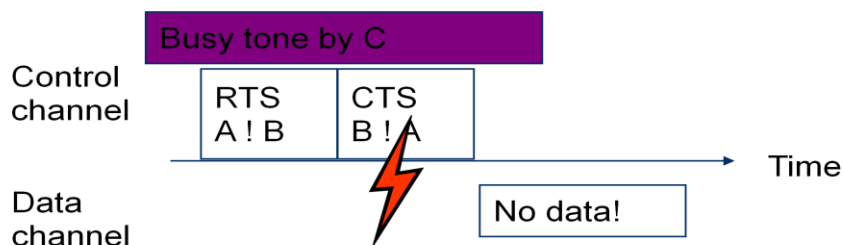


Figure: PMAS with Signaling

PAMAS – Already ongoing transmission

- Suppose a node C in vicinity of A is already receiving a packet when A initiates RTS
- Procedure
 - A sends RTS to B
 - C is sending busy tone (as it receives data)
 - CTS and busy tone collide, A receives no CTS, does not send data



Similarly: Ongoing transmission near B destroys RTS by busy tone

SCHEDULE-BASED PROTOCOLS

Scheduled based MAC protocol is divided into two categories. The first one is traditional TDMA MAC protocol or centralized TDMA protocols.

Second category is distributed TDMA protocols. In centralized TDMA protocols; the nodes are scheduled centrally by different time slots.

Fundamental advantage of schedule based protocols

,Firstly by employing TDMA schemes, which explicitly assign transmission and reception opportunities to nodes and let them sleep at all other times. A second fundamental advantage of schedule-based protocols is that transmission schedules can be computed such that no collisions occur at receivers and hence no special mechanisms are needed to avoid hidden-terminal situations

However, these schemes also have downsides. First, the setup and maintenance of schedules involves signaling traffic, especially when faced to variable topologies. Second, if a TDMA variant is employed, time is divided into comparably small slots, and both transmitter and receiver have to agree to slot boundaries to actually meet each other and to avoid overlaps with other slots, which would lead to collisions. However, maintaining time synchronization involves some extra signaling traffic.

The schedule based protocols are:

LEACH Low-energy Adaptive Clustering Hierarchy

TRAMA Traffic-adaptive medium access protocol (TRAMA)

Self-Organizing Medium Access Control for Sensor Networks (SMACS) protocol

LEACH Low-energy Adaptive Clustering Hierarchy

What is Leach protocol in WSN?

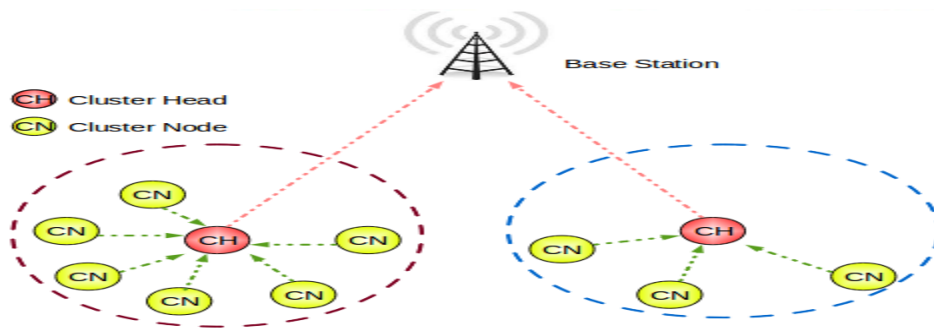
Low-energy adaptive clustering hierarchy ("LEACH") is a TDMA-based MAC protocol which is integrated with clustering and a simple routing protocol in wireless sensor networks (WSNs)

Why we are using leach protocol in sensor networks?

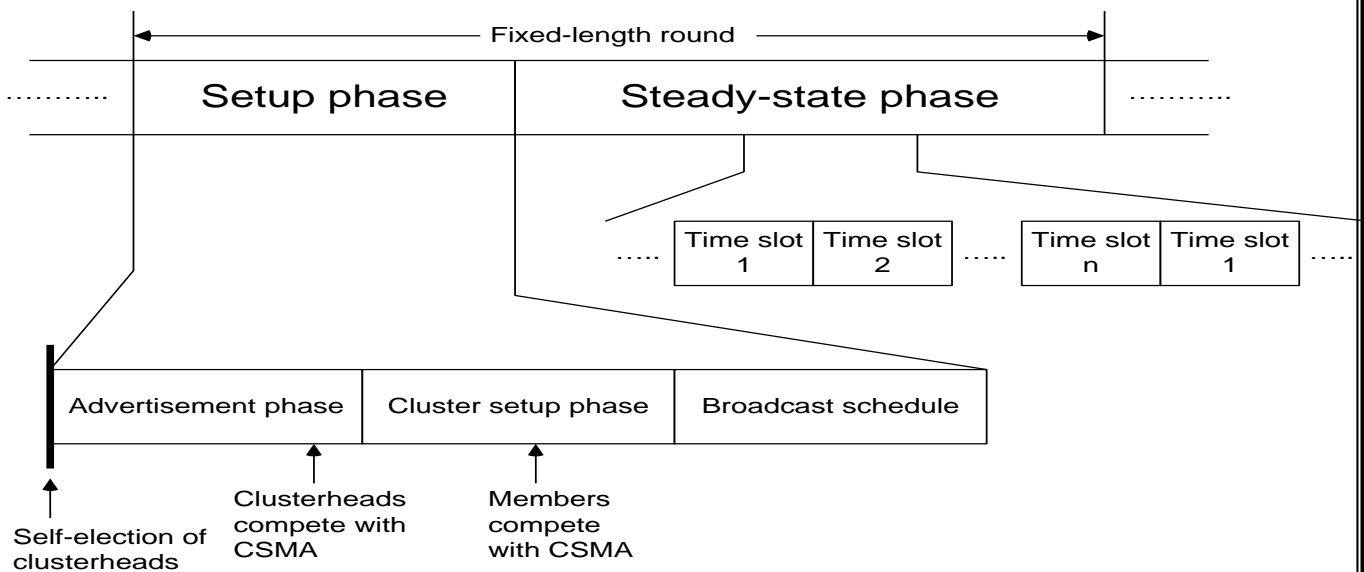
To increase network lifetime, energy consumption is considered as one of an essential performance metric. ... The common pioneer hierarchical routing protocol for wsn such as Low Energy Adaptive Cluster Hierarchical Routing (LEACH) is also proposed to improve the energy efficiency of WSN

LEACH

- Given: dense network of nodes, reporting to a central sink, each node can reach sink directly
- Idea: Group nodes into “clusters”, controlled by *clusterhead*
 - Setup phase;
 - About 5% of nodes become clusterhead (depends on scenario)
 - Role of clusterhead is rotated to share the burden
 - Clusterheads advertise themselves, ordinary nodes join CH with strongest signal
 - Clusterheads organize
 - CDMA code for all member transmissions
 - TDMA schedule to be used within a cluster
- In steady state operation
 - CHs collect & aggregate data from all cluster members
 - Report aggregated data to sink using CDMA



LEACH Rounds



Set-up phase

- Cluster heads assign a TDMA schedule for their members where each node is assigned a time slot when it can transmit.
- Each cluster communications using different CDMA codes to reduce interference from nodes belonging to other clusters.

TDMA intra-cluster

CDMA inter-cluster

- Spreading codes determined randomly
- Broadcast during advertisement phase

In steady state operation

- CHs collect & aggregate data from all cluster members
- Report aggregated data to sink using CDMA

Advantages

- Increases the lifetime of the network
- Even drain of energy
- Distributed, no global knowledge required
- Energy saving due to aggregation by CHs

Disadvantages

- LEACH assumes all nodes can transmit with enough power to reach BS if necessary (e.g., elected as CHs)
- Each node should support both TDMA & CDMA
- Need to do time synchronization
- Nodes use single-hop communication

The IEEE 802.15.4 MAC protocol

The standard covers the physical layer **140** MAC protocols and the MAC layer of a low-rate Wireless Personal Area Network (WPAN).

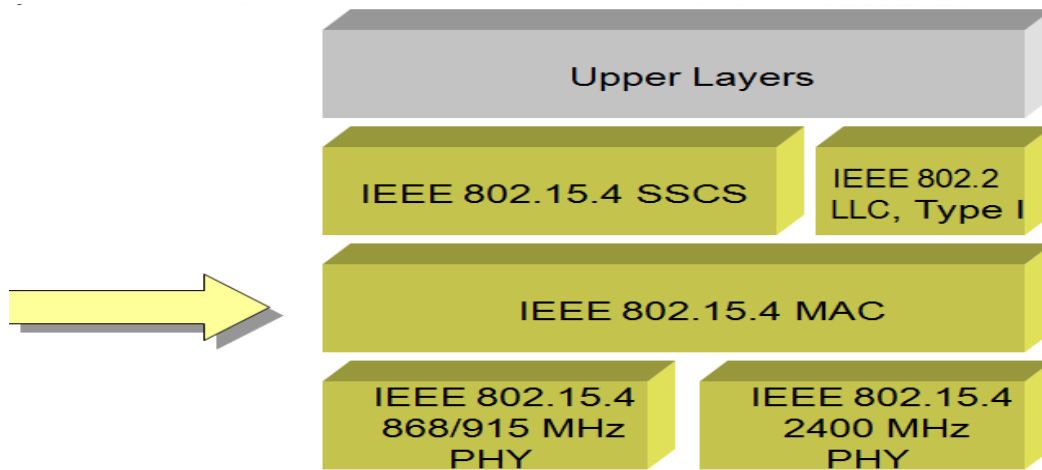
The targeted applications for IEEE 802.15.4 are in the area of wireless sensor networks, home automation, home networking, connecting devices to a PC, home security, and so on. Most of these applications require only low-to-medium bitrates (up to some few hundreds of kbps), moderate average delays without too stringent delay guarantees, and for certain nodes it is highly desirable to reduce the energy consumption to a minimum.

The physical layer offers bitrates of 20 kbps (a single channel in the frequency range 868–868.6 MHz), 40 kbps (ten channels in the range between 905 and 928 MHz) and 250 kbps (16 channels in the 2.4 GHz ISM band between 2.4 and 2.485 GHz with 5-MHz spacing between the center frequencies).

There are a total of 27 channels available, but the MAC protocol uses only one of these channels at a time; it is not a multichannel protocol. The MAC protocol combines both schedule-based as well as contention-based schemes

Network architecture and types/roles of nodes

802.15.4 Architecture



Over View

- Star and peer-to-peer topologies
- Optional frame structure
- Association
- CSMA-CA channel access mechanism
- Packet validation and message rejection
- Optional guaranteed time slots
- Guaranteed packet delivery
- Facilitates low-power operation
- Security

The standard distinguishes on the MAC layer two types of nodes:

- A Full Function Device (FFD) can operate in three different roles: it can be a **PAN coordinator** (PAN = Personal Area Network), a simple **coordinator** or a **device**.
- A Reduced Function Device (RFD) can operate only as a device

IEEE 802.15.4 Device Classes

- Full function device (FFD)
 - Any topology
 - PAN coordinator capable
 - Talks to any other device
 - Implements complete protocol set
- Reduced function device (RFD)

- Limited to star topology or end-device in a peer-to-peer network.
 - Cannot become a PAN coordinator
 - Very simple implementation
 - Reduced protocol set
- Network Device: An RFD or FFD implementation containing an IEEE 802.15.4 medium access control and physical interface to the wireless medium.
 - Coordinator: An FFD with network device functionality that provides coordination and other services to the network.
 - PAN Coordinator: A coordinator that is the principal controller of the PAN. A network has exactly one PAN coordinator.

A device must be associated to a coordinator node (which must be a FFD) and communicates only with this, this way forming a **star network**. Coordinators can operate in a peer-to-peer fashion and multiple coordinators can form a Personal Area Network (PAN). The PAN is identified by a 16-bit **PAN Identifier** and one of its coordinators is designated as a PAN coordinator.

A coordinator handles among others the following tasks:

- It manages a list of associated devices. Devices are required to explicitly associate and disassociate with a coordinator using certain signaling packets.

- It allocates short addresses to its devices. All IEEE 802.15.4 nodes have a 64-bit device address.

When a device associates with a coordinator, it may request assignment of a 16-bit short address to be used subsequently in all communications between device and coordinator. The assigned address is indicated in the association response packet issued by the coordinator.

- In the beamed mode of IEEE 802.15.4, it transmits regularly **frame beacon** packets announcing the PAN identifier, a list of outstanding frames, and other parameters. Furthermore, the coordinator can accept and process requests to reserve fixed time slots to nodes and the allocations are indicated in the beacon.
- It exchanges data packets with devices and with peer coordinators.

- Duty-cycle control using superframe structure
 - Beacon order and superframe order
 - Coordinator battery life extension
- Indirect data transmission
- Devices may sleep for extended period over multiple beacons
- Allows control of receiver state by higher layers

Superframe structure

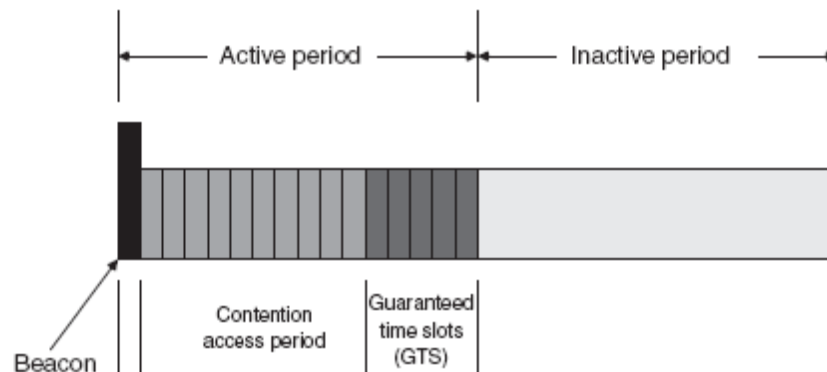


Figure 5.14 Superframe structure of IEEE 802.15.4

The coordinator of a star network operating in the **beaconed mode** organizes channel access and data transmission with the help of a superframe structure displayed in Figure 5.14. All superframes have the same length. The coordinator starts each superframe by sending a frame beacon packet. The frame beacon includes a **superframe specification** describing the length of the various components of the following superframe:

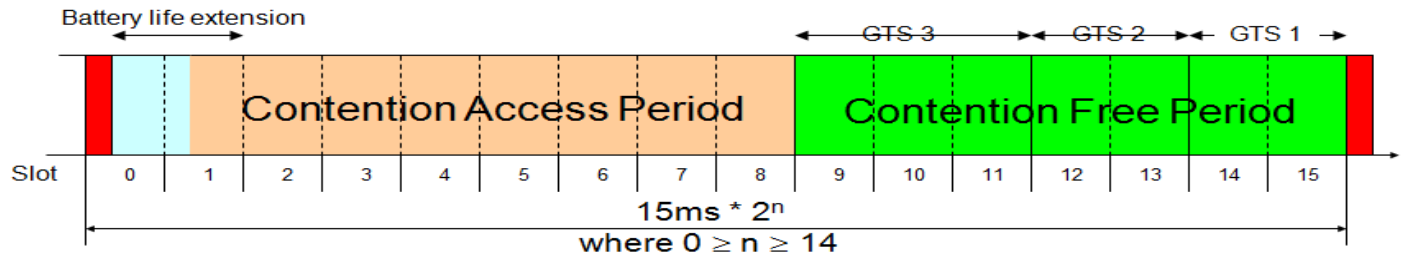
- The superframe is subdivided into an **active period** and an **inactive period**. During the inactive period, all nodes including the coordinator can switch off their transceivers and go into sleep state. The nodes have to wake up immediately before the inactive period ends to receive the next beacon. The inactive period may be void.
- The active period is subdivided into 16 time slots. The first time slot is occupied by the beacon frame and the remaining time slots are partitioned into a **Contention Access Period (CAP)** followed by a number (maximal seven) of contiguous **Guaranteed Time Slots (GTSs)**.

The length of the active and inactive period as well as the length of a single time slot and the usage of GTS slots are configurable.

The coordinator is active during the entire active period. The associated devices are active in the GTS phase only in time slots allocated to them; in all other GTS slots they can enter sleep mode. In the CAP, a device can shut down its transceiver if it has neither any own data to transmit nor any data to fetch from the coordinator.

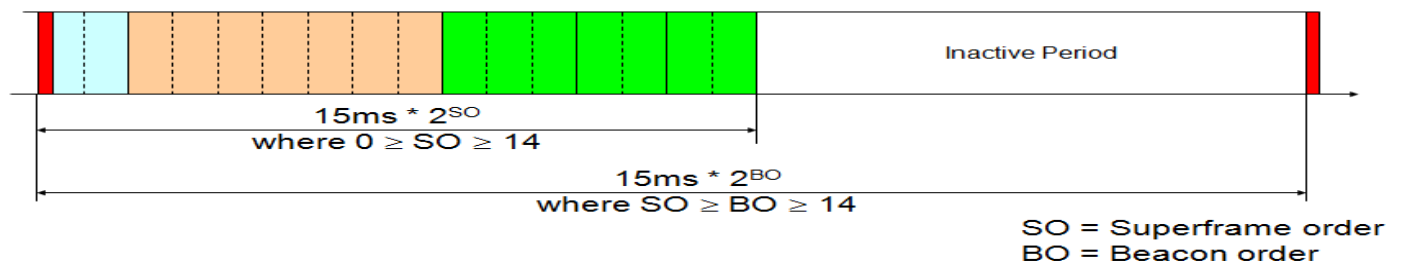
It can be noted already from this description that coordinators do much more work than devices and the protocol is inherently asymmetric. The protocol is optimized

Optional Frame Structure



- | | | |
|-------------------------|--|--|
| Network beacon | | Transmitted by PAN coordinator. Contains network information, frame structure and notification of pending node messages. |
| Beacon extension period | | Space reserved for beacon growth due to pending node messages |
| Contention period | | Access by any node using CSMA-CA |
| Guaranteed Time Slot | | Reserved for nodes requiring guaranteed bandwidth [$n = 0$]. |

Optional Frame Structure



- Superframe may have inactive period

Data Service

- Data transfer to neighboring devices
 - Acknowledged or unacknowledged
 - Direct or indirect
 - Using GTS service
- Maximum data length (MSDU) $aMaxMACFrameSize$ (102 bytes)

Data Transfer Message Diagram(Direct and Indirect)

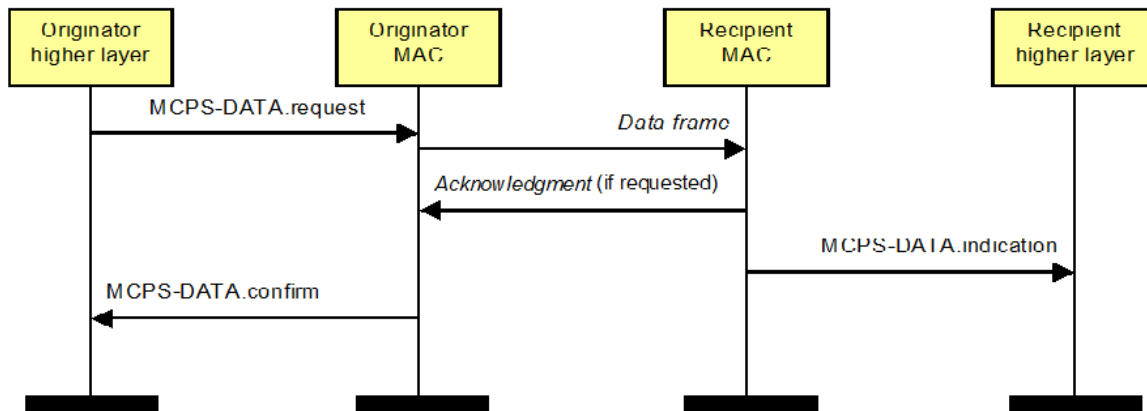
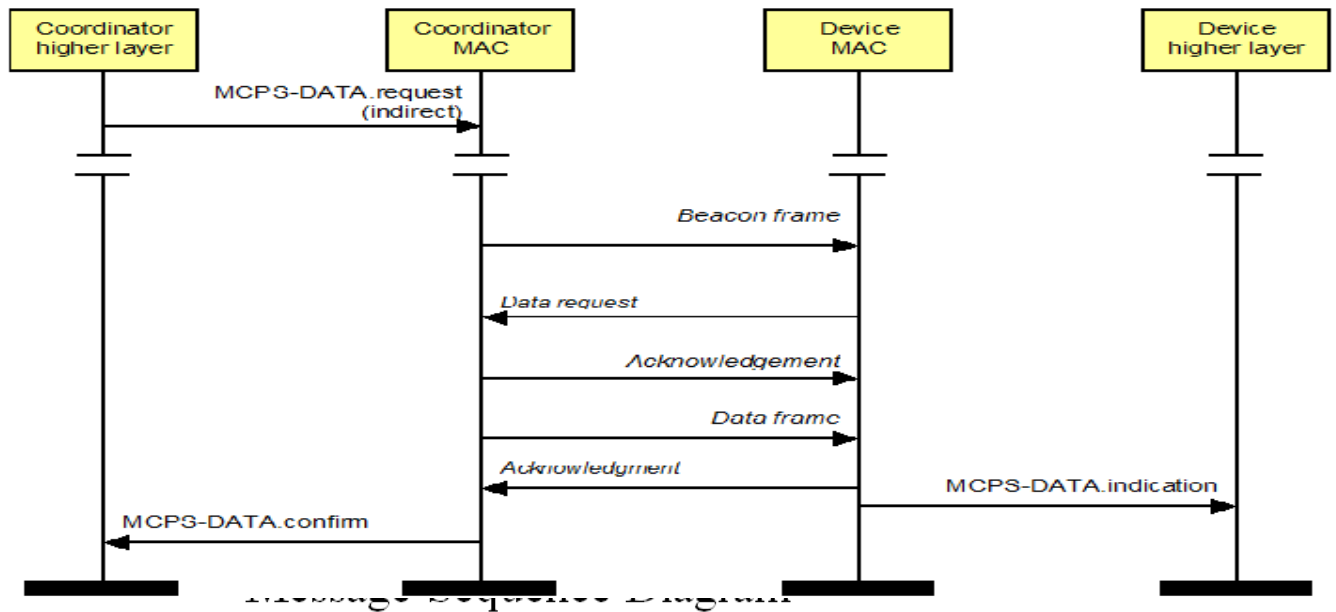


Table 5.1 Summary of important WSN MAC protocols

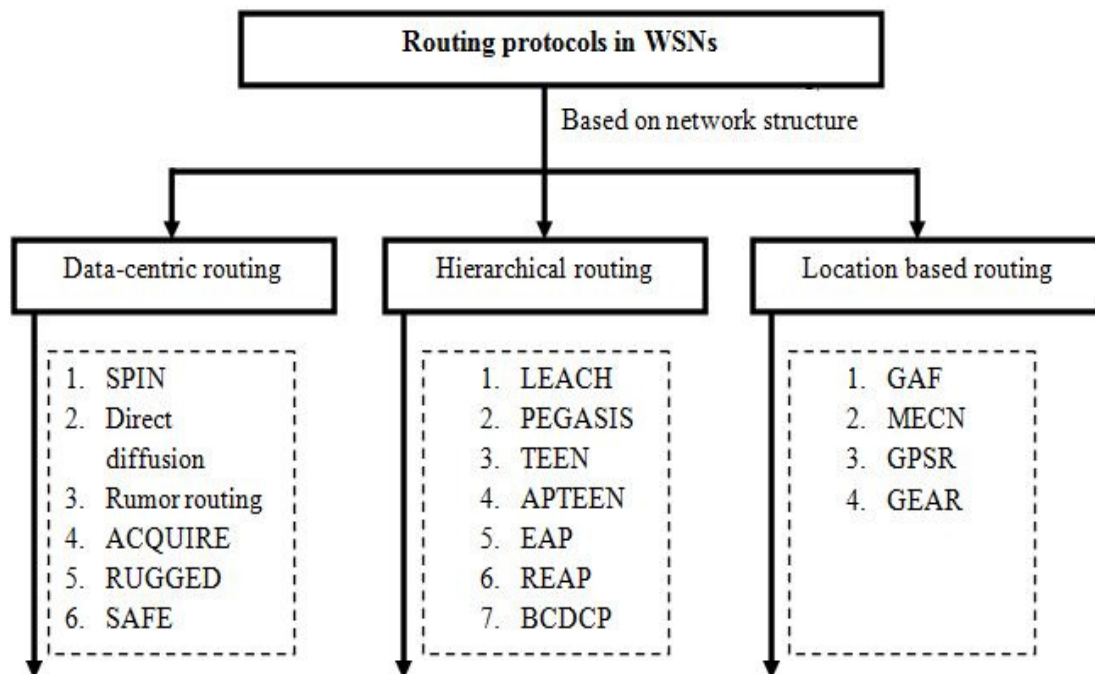
Protocol	References	Flat/ clustered	# of required channels	Idle listening avoidance	Overhearing avoidance	Collision avoidance	Overhead
LEACH	[346]	Rotating clusters	1	By TDMA	By TDMA	By TDMA	Cluster election/ formation
STEM	[742]	Both	2	Periodic sleep	STEM-B	Depends on MAC	Depends on MAC, wake up beacons
S-MAC	[914, 915]	Flat	1	Periodic sleep	Through NAV	RTS/CTS	RTS/CTS, SYNCH, virtual cluster init.
Mediation device	[115, Chap. 4]	Flat	1	Periodic sleep	Implicit	No	Periodic mediator service, query beacons, RTS/CTS
Wakeup radio	[667, 931]	Flat	≥ 2	Wakeup signal	Wakeup signal	Multichannel CSMA	Extra wakeup radio
CSMA protocols	[888]	Flat	1	-	Sleep during backoff	RTS/CTS	RTS/CTS
PAMAS	[668]	Flat	2	-	Yes	RTS/CTS, busy tone	Signaling channel
SMACS	[778, 780]	Flat	Many	By TDMA	By TDMA	By TDMA	Neighborhood discovery, channel setup
TRAMA	[672]	Flat	1	By scheduling	By scheduling	By scheduling	Neighbor protocol, schedule transmission

CLASSIFICATION OF WSN ROUTING PROTOCOLS

Three categories

- ▶ Data-centric
- ▶ Hierarchical
- ▶ Location based routing

Classification of routing protocols



What is energy efficient routing?

Energy-efficient routing protocols exploit application requirements in order to save precious **energy** of the sensor nodes resulting in extended lifetime of the network . **Routing** protocols for delay sensitive applications tend to reduce the number of transmissions in order to save **energy**.

An Energy Efficient ANT Based Routing algorithm (EEABR)

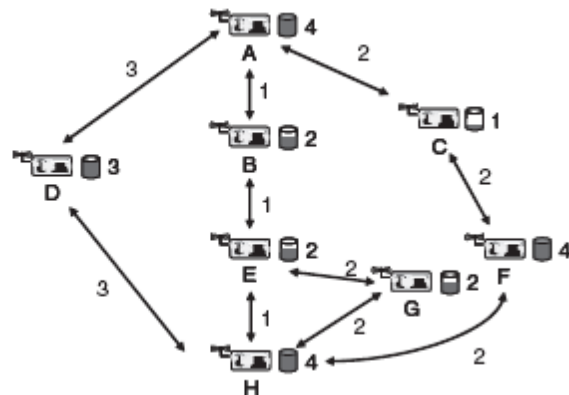
- ▶ Routing protocol is based on ANT colony
- ▶ highly adaptive, efficient and scalable
- ▶ ANTS travel through the WSN looking for path between sensor nodes and a destination node
- ▶ that are both short in length and energy efficient
- ▶ Forward (FANT) and backward ANTS (BANT).
- ▶ A forward ANT is launched periodically from every node
- ▶ ANT stores the identifiers of all the nodes it visits
- ▶ selection probability is a trade-off between visibility(Energy) and actual trail intensity

- ▶ BANT sent back along the path stored

EECDA (Energy Efficient Clustering and Data Aggregation) Protocol

- After the CHs election, a **path with maximum sum of residual energy** would be selected for data communication instead of the **path with minimum energy consumption**.
- Therefore, each CH first aggregates the received data and then transmits the aggregated data to the Base Station (BS).
- The main contributions of EECDA protocol is to provide longest stability and improves the network lifetime

Minimize energy per packet (or per bit) The most straightforward formulation is to look at the total energy required to transport a packet over a multihop path from source to destination (including all overheads). The goal is then to minimize, for each packet, this total amount of energy by selecting a good route. Minimizing the hop count will typically not achieve this goal as routes with few hops might include hops with large transmission power to cover large distances – but be aware of distance-independent, constant offsets in the energy-consumption model. Nonetheless, this cost metric can be easily included in standard routing algorithms. It can lead to widely differing energy consumption on different nodes.



Various example routes for communication between nodes A and H, showing energy costs per packet for each link and available battery capacity for each node

In the example of above Figure the minimum energy route is A-B-E-H, requiring 3 units of energy. The minimum hop count route would be A-D-H, requiring 6 units of energy.

Why energy efficiency is important in WSN routing?

Energy efficient routing protocols are required to minimize the utilization of the power resources and prolonging the network lifetime path while transferring data

POWER-AWARE ROUTING PROTOCOLS

Power-Aware Routing Metrics

The limitation on the availability of power for operation is a significant bottleneck.

Hence, the use of routing metrics contributes to the efficient utilization of energy and increases the lifetime of the network

- **Minimal energy consumption per packet**
 - This metric aims at minimizing the power consumed by a packet in traversing from source node to the destination node.
 - The energy consumed by a packet when traversing through a path is the sum of the energies required at every intermediate hop in that path.
 - This metric doesn't balance the load
 - Disadvantages
 - Selection of path with large hop length
 - Inability to measure the power consumption in advance
 - Inability to prevent the fast discharging of batteries at some nodes
- **Maximize network connectivity**
 - This metric attempt to balance the routing load among the cut set (the subset of the nodes in the network, the removal of which results in network partitions).

It is difficult to achieve a uniform battery draining rate for the cut set.
- **Maximum variance in Node power levels**
 - This metric proposes to distribute the load among all nodes in the network so that the power consumption pattern remains uniform across them.
 - This problem is very complex when the rate and size of the data packets vary
- **Minimum cost per packet**
 - In order to maximize the life of every node in the network, this routing metric is made as a function of the state of the node's battery.
 - A node's cost decreases with an increase in its battery charge and vice versa.
 - Cost of node can be easily computed
 - Advantage □ congestion handling & cost calculation
- **Minimize maximum node cost**
 - This metric minimizes the maximum cost per node for a packet after routing a number of packets or after a specific period.
 - This delays the failure of a node, occurring due to higher discharge because of packet forwarding

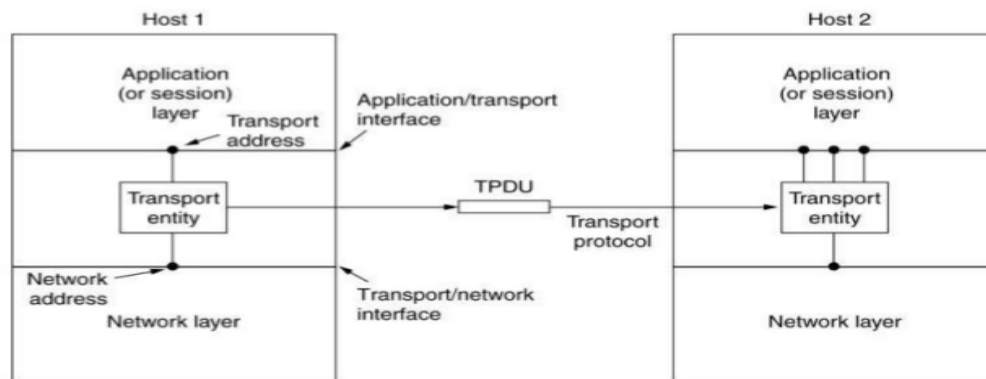
CHALLENGES AND ISSUES IN TRANSPORT LAYER PROTOCOL

INTRODUCTION

The objectives of transport layer protocol include the setting up of an end-to-end connection, end-to-end delivery of data packets, flow control, congestion control.

Transport Layer

The ultimate goal of the transport layer is to provide efficient, reliable, and cost-effective service to its users, normally processes in the application layer. To achieve this goal, the transport layer makes use of the services provided by the network layer. The hardware and/or software within the transport layer that does the work is called the **transport entity**.



ISSUES IN DESIGNING A TRANSPORT LAYER PROTOCOL FOR AD HOC WIRELESS NETWORKS

1. **Induced Traffic:**
 - In a path having multiple link, the traffic at any given link (or path) due to the traffic through neighbouring links (or paths) is referred to as induced traffic.
 - This is due to the broadcast nature of the channel and the location-dependent contention on the channel
 - Induced Traffic affects the throughput achieved by the transport layer protocol.
2. **Induced throughput unfairness:**
 - This refers to the throughput unfairness at the transport layer due to the throughput/delay unfairness existing at the lower layer such as the n/w and MAC layers.
 - A transport layer should consider these in order to provide a fair share of throughput across contending flows
3. **Separation of congestion control, reliability and flow control:**
 - A transport layer protocol can provide better performance if end-to-end reliability, flow control and congestion control are handled separately.
 - Reliability and flow control are end-to-end activities, whereas congestion can at times be a

- local activity
- Objective □ minimisation of the additional control overhead generated by them
- 4. **Power and Band width constraints:**
 - Nodes in ad hoc wireless networks face resource constraints including the two most important resources: (i) power source and (ii) bandwidth
 - The performance of a Transport layer protocol is significantly affected by these resource constraints
- 5. **Interpretation of congestion:**
 - Interpretation of network congestion as used in traditional networks is not appropriate in ad hoc networks.
 - This is because the high error rates of wireless channel, location-dependent contention, hidden terminal problem, packet collisions in the network, path breaks due to mobility of nodes, and node failure due to drained battery can also lead to packet loss in ad hoc wireless networks
- 6. **Completely decoupled transport layer:**
 - Another challenge faced by Transport layer protocol is the interaction with the lower layers.
 - Cross-layer interaction between the transport layer and lower layers is important to adapt to the changing network environment
- 7. **Dynamic topology:**
 - Experience rapidly changing network topology due to mobility of nodes
 - Leads to frequent path breaks, partitioning and remerging of networks & high delay in re-establishment of paths

Performance is affected by rapid changes in network topology

DESIGN GOALS OF A TRANSPORT LAYER PROTOCOL FOR ADHOC WIRELESS NETWORKS

- ✓ The protocol should maximize the throughput per connection.
- ✓ It should provide throughput fairness across contending flows.
- ✓ It should incur minimum connection set up and connection maintenance overheads.
- ✓ It should have mechanisms for congestion control and flow control in the network.
- ✓ It should be able to provide both reliable and unreliable connections as per the requirements of the application layer.
- ✓ It should be able to adapt to the dynamics of the network such as rapid changes in topology.
- ✓ Bandwidth must be used efficiently.
- ✓ It should be aware of resource constraints such as battery power and buffer sizes and make efficient use of them.
- ✓ It should make use of information from the lower layers for improving network throughput.
- ✓ It should have a well-defined cross-layer interaction framework.
- ✓ It should maintain End-to-End Semantics.

CLASSIFICATION OF TRANSPORT LAYER SOLUTIONS

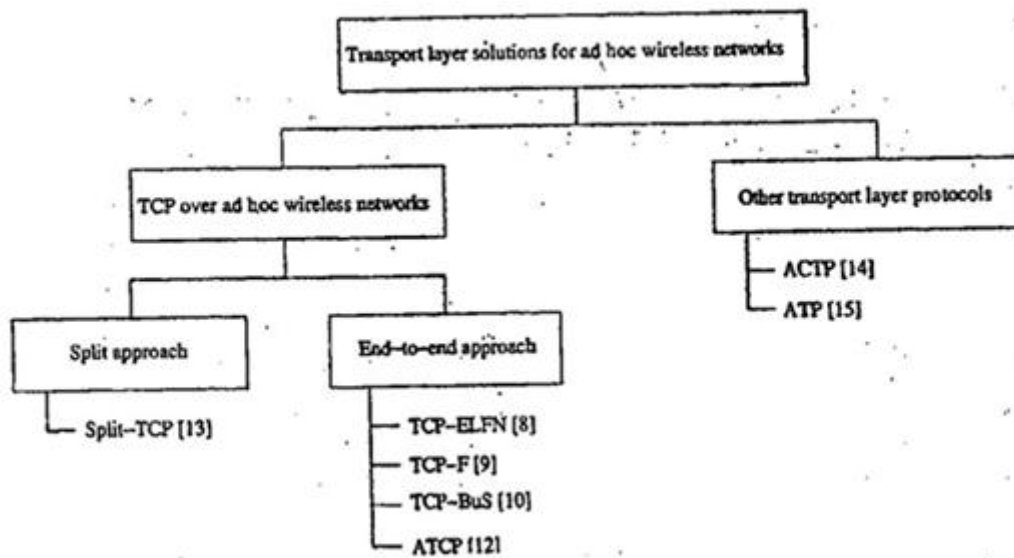


Figure 9.1. Classification of transport layer solutions.

Mohamad Sathak A.J College of Engineering
Department of ECE

EC8702
ADHOC AND WIRELESS SENSOR NETWORKS

UNIT IV
SENSOR NETWORK SECURITY

UNIT IV

Introduction

Outline:

- **Learn Ad hoc network and Sensor Network Security**
- **Understand the different security aspects**
- **Have an in-depth knowledge on sensor network attacks and its issues related to network security**
- **Understand the transport layer and security issues possible in Ad hoc and Sensor networks**

Topics

- **Network Security Requirements,**
- **Issues and Challenges in Security Provisioning,**
- **Network Security Attacks,**
- **Layer wise attacks in wireless sensor networks,**
- **Possible solutions for**
 - **Jamming,**
 - **Tampering,**
 - **Black Hole Attack,**
 - **Flooding Attack.**
- **Key Distribution and Management,**
- **Secure Routing – SPINS,**

Reliability Requirements in Sensor Networks

Network security is one of the most pressing concerns in all wireless networks, including wireless sensor networks.

Fundamentals

Network designers have to be aware of and decide about suitable mechanisms to implement one or more of the following general **security goals**

NETWORK SECURITY GOALS OR NETWORK SECURITY REQUIREMENTS

Network Security Requirements:

The requirements listed below should in fact be met by security protocols for other types of networks also.

- ✓ Confidentiality
- ✓ Integrity
- ✓ Accountability
- ✓ Availability
- ✓ Controlled access
- ✓ Non-repudiation

Confidentiality Information should only be revealed to authorized entities; any other entity should not be able to discover the information from eavesdropping or from reading memories.

The eavesdropping attack is a serious security threat to a Wireless Sensor Network (WSN)

Data integrity: The data sent by the source node should reach the destination node as it was sent: unaltered. In other words, it should not be possible for any malicious node in the network to tamper with the data during transmission.

Accountability: The entity requesting a service, triggering an action, or sending a packet must be uniquely identifiable.

Availability: (Legitimate entities should be able to access a certain service/information and to enjoy proper operation.)

(The network should remain operational all the time. It must be robust enough to tolerate link failures and also be capable of surviving various attacks mounted on it. It should be able to provide the guaranteed services whenever an authorized user requires them.)

Controlled access: A service or information access should only be granted to authorized entities.

Non-repudiation: Non-repudiation is a mechanism to guarantee that the sender of a message cannot later deny having sent the message and that the recipient cannot deny having received the message. Digital signatures, which function as unique identifiers for each user, much like a written signature, are used commonly for this purpose

ISSUES AND CHALLENGES IN SECURITY PROVISIONING

Issues

Here some of the **issues** to be considered while designing a security concern for ad hoc wireless networks are discussed.

- ✓ Induced traffic
- ✓ Induced throughput unfairness
- ✓ Separation of congestion control, reliability, and flow control
- ✓ Power and bandwidth constraints
- ✓ Misinterpretation of congestion
- ✓ Completely decoupled transport layer
- ✓ Dynamic topology
- ✓
- **Induced traffic:** Unlike wired networks, ad hoc wireless networks utilize multi-hop radio relaying. A link-level transmission affects the neighbor nodes of both the sender and receiver of the link. In a path having multiple links, transmission at a particular link affects one upstream link and one downstream link. This traffic at any given link (or path) due to the traffic through neighboring links (or paths) is referred to as induced traffic. This is due to the broadcast nature of the channel and the location-dependent contention on the channel. This induced traffic affects the throughput achieved by the transport layer protocol.
- **Induced throughput unfairness:** This refers to the throughput unfairness at the transport layer due to the throughput/delay unfairness existing at the lower layers such as the network and MAC layers. For example, an ad hoc wireless network that uses *IEEE 802.11 DCF* as the MAC protocol may experience throughput unfairness at the transport layer as well. A transport layer protocol should consider these in order to provide a fair share of throughput across contending flows.
- **Separation of congestion control, reliability, and flow control:** A transport layer protocol can provide better performance if end-to-end reliability, flow control, and congestion control are handled separately. Reliability and flow control are end-to-end activities, whereas congestion can at times be a

- local activity. The transport layer flow can experience congestion with just one intermediate link under congestion. Hence, in networks such as ad hoc wireless networks, the performance of the transport layer may be improved if these are separately handled. While separating these, the most important objective to be considered is the minimization of the additional control overhead generated by them.
- **Power and bandwidth constraints:** Nodes in ad hoc wireless networks face resource constraints including the two most important resources: (i) power source and (ii) bandwidth. The performance of a transport layer protocol is significantly affected by these constraints.
- **Misinterpretation of congestion:** Traditional mechanisms of detecting congestion in networks, such as packet loss and retransmission timeout, are not suitable for detecting the network congestion in ad hoc wireless networks. This is because the high error rates of wireless channel, location-dependent contention, hidden terminal problem, packet collisions in the network, path breaks due to the mobility of nodes, and node failure due to a drained battery can also lead to packet loss in ad hoc wireless networks. Hence, interpretation of network congestion as used in traditional networks is not appropriate in ad hoc wireless networks.
- **Completely decoupled transport layer:** Another challenge faced by a transport layer protocol is the interaction with the lower layers. Wired network transport layer protocols are almost completely decoupled from the lower layers. In ad hoc wireless networks, the cross-layer interaction between the transport layer and lower layers such as the network layer and the *MAC* layer is important for the transport layer to adapt to the changing network environment.
- **Dynamic topology:** Some of the deployment scenarios of ad hoc wireless networks experience rapidly changing network topology due to the mobility of nodes. This can lead to frequent path breaks, partitioning and remerging of networks, and high delay in reestablishment of paths. Hence, the performance of a transport layer protocol is significantly affected by the rapid changes in the network topology.

Challenges in Security Provisioning

The following are the Challenges to be met while make a security for ad hoc wireless networks:

- ✓ Throughput maximization
- ✓ Minimum connection setup and connection maintenance overheads
- ✓ Congestion control and flow control
- ✓ Reliable and unreliable connections
- ✓ Protocol
- ✓ Bandwidth availability
- ✓ Effective, scalable, and protocol-independent interaction
- ✓ End-to-end semantics

The protocol should maximize the throughput per connection. It should provide throughput fairness across contending flows.

- The protocol should incur minimum connection setup and connection maintenance overheads. It should minimize the resource requirements for setting up and maintaining the connection in order to make the protocol scalable in large networks.
- The transport layer protocol should have mechanisms for congestion control and flow control in the network.

- It should be able to provide both reliable and unreliable connections as per the requirements of the application layer.
- The protocol should be able to adapt to the dynamics of the network such as the rapid change in topology and changes in the nature of wireless links from uni-directional to bidirectional or vice versa... The protocol should be aware of resource constraints such as battery power and buffer sizes and make efficient use of them. The transport layer protocol should make use of information from the lower layers in the protocol stack for improving the network throughput.
- One of the important resources, the available bandwidth, must be used efficiently.
- It should have a well-defined cross-layer interaction framework for effective, scalable, and protocol-independent interaction with lower layers.
- The protocol should maintain end-to-end semantics.

ISSUES AND CHALLENGES IN SECURITY PROVISIONING

A detailed discussion on how each of the above mentioned characteristics causes difficulty in providing security in ad hoc wireless networks is given below.

- ✓ **Shared broadcast radio channel**
- ✓ **Insecure operational environment**
- ✓ **Lack of central authority**
- ✓ **Lack of association**
- ✓ **Limited resource availability**
- ✓ **Physical vulnerability**
- ✓ Designing a foolproof security protocol for ad hoc wireless is a very challenging task. This is mainly because of certain unique characteristics of ad hoc wireless networks, namely, shared broadcast radio channel, insecure operating environment, lack of central authority, lack of association among nodes, limited availability of resources, and physical vulnerability. A detailed discussion on how each of the above mentioned characteristics causes difficulty in providing security in ad hoc wireless networks is given below.
- ✓ **Shared broadcast radio channel:** Unlike in wired networks where a separate dedicated transmission line can be provided between a pair of end users, the radio channel used for communication in ad hoc wireless networks is broadcast in nature and is shared by all nodes in the network. Data transmitted by a node is received by all nodes within its direct transmission range. So a malicious node (**node** seeking to deny service to other **nodes** in the network.) could easily obtain data being transmitted in the network. This problem can be minimized to a certain extent by using directional antennas.
- ✓ **Insecure operational environment:** The operating environments where ad hoc wireless networks are used may not always be secure. One important application of such networks is in battlefields. In such applications, nodes may move in and out of hostile and insecure enemy territory, where they would be highly vulnerable to security attacks.
- ✓ **Lack of central authority:** In wired networks and infrastructure-based wireless networks, it would be possible to monitor the traffic on the network through certain important central points (such as routers, base stations, and access points) and implement security mechanisms at such points. Since ad hoc wireless networks do not have any such central points, these mechanisms cannot be applied in ad hoc wireless networks.
- ✓ **Lack of association:** Since these networks are dynamic in nature, a node can join or leave the network at any point of the time. If no proper authentication mechanism is used for associating nodes with a network, an intruder would be able to join into the network quite easily and carry out his/her attacks.

- ✓ **Limited resource availability:** Resources such as bandwidth, battery power, and computational power (to a certain extent) are scarce in ad hoc wireless networks. Hence, it is difficult to implement complex cryptography-based security mechanisms in such networks.
- ✓ **Physical vulnerability:** Nodes in these networks are usually compact and hand-held in nature. They could get damaged easily and are also vulnerable to theft. (**Physical vulnerability** is defined as any flaw or weakness in a data system or its hosting environment that can enable a **physical** attack on the system. There are different types of **physical** security attacks to data system)

NETWORK SECURITY ATTACKS

Core Reason for Network Security Attacks

- ✓ Misinterpretation of packet loss
- ✓ Frequent path breaks
- ✓ Effect of path length
- ✓ Misinterpretation of congestion window
- ✓ Asymmetric link behaviour
- ✓ Uni-directional path
- ✓ Multipath routing
- ✓ Network partitioning and remerging
- ✓ The use of sliding-window-based transmission

Misinterpretation of packet loss: Traditional *TCP* was designed for wired networks where the packet loss is mainly attributed to network congestion. Network congestion is detected by the sender's packet *RTO* period. Once a packet loss is detected, the sender node assumes congestion in the network and invokes a congestion control algorithm. Ad hoc wireless networks experience a much higher packet loss due to factors such as high bit error rate (*BER*) in the wireless channel, increased collisions due to the presence of hidden terminals, presence of interference, location-dependent contention, uni-directional links, frequent path breaks due to mobility of nodes, and the inherent fading properties of the wireless channel.

Frequent path breaks: Ad hoc wireless networks experience dynamic changes in network topology because of the unrestricted mobility of the nodes in the network. The topology changes lead to frequent changes in the connectivity of wireless links and hence the route to a particular destination may need to be recomputed very often. The responsibility of finding a route and re-establishing it once it gets broken is attached to the network layer (Chapter 7 discusses network layer routing protocols in detail). Once a path is broken, the routing protocol initiates a route reestablishment process. This route reestablishment process takes a significant amount of time to obtain a new route to the destination. The route reestablishment time is a function of the number of nodes in the network, transmission ranges of nodes, current topology of the network, bandwidth of the channel, traffic load in the network, and the nature of the routing protocol. If the route reestablishment time is greater than the *RTO* period of the *TCP* sender, then the *TCP* sender assumes congestion in the network, retransmits the lost packets, and initiates the congestion control algorithm. These retransmissions can lead to wastage of bandwidth and battery power. Eventually, when a new route is found, the *TCP* throughput continues to be low for some time, as it has to build up the congestion window since the traditional *TCP* undergoes a slow start.

Effect of path length: It is found that the *TCP* throughput degrades rapidly with an increase in path length in string (linear chain) topology ad hoc wireless networks..

Misinterpretation of congestion window: *TCP* considers the congestion window as a measure of the rate of transmission that is acceptable to the network and the receiver. In ad hoc wireless networks, the congestion

control mechanism is invoked when the network gets partitioned or when a path break occurs. This reduces the congestion window and increases the *RTO* period. When the route is reconfigured, the congestion window may not reflect the transmission rate acceptable to the new route, as the new route may actually accept a much higher transmission rate. Hence, when there are frequent path breaks, the congestion window may not reflect the maximum transmission rate acceptable to the network and the receiver.

Asymmetric link behaviour: The radio channel used in ad hoc wireless networks has different properties such as location-dependent contention, environmental effects on propagation, and directional properties leading to asymmetric links. The directional links can result in delivery of a packet to a node, but failure in the delivery of the acknowledgment back to the sender. It is possible for a bidirectional link to become uni-directional for a while. This can also lead to *TCP* invoking the congestion control algorithm and several retransmissions.

Uni-directional path: Traditional *TCP* relies on end-to-end ACK for ensuring reliability. Since the ACK packet is very short compared to a data segment, ACKs consume much less bandwidth in wired networks. In ad hoc wireless networks, every *TCP* ACK packet requires *RTS-CTS-Data-ACK* exchange in case *IEEE 802.11* is used as the underlying *MAC* protocol.

This can lead to an additional overhead of more than 70 bytes if there are no retransmissions. This can lead to significant bandwidth consumption on the reverse path, which may or may not contend with the forward path. If the reverse path contends with the forward path, it can lead to the reduction in the throughput of the forward path. Some routing protocols select the forward path to be also used as the reverse path, whereas certain other routing protocols may use an entirely different or partially different path for the ACKs. A path break on an entirely different reverse path can affect the performance of the network as much as a path break in the forward path.

Multipath routing: There exists a set of QoS routing and best-effort routing protocols that use multiple paths between a source-destination pair. There are several advantages in using multipath routing. Some of these advantages include the reduction in route computing time, the high resilience to path breaks, high call acceptance ratio, and better security. For *TCP*, these advantages may add to throughput degradation. These can lead to a significant amount of out-of-order packets, which in turn generates a set of duplicate acknowledgments (DUPACKs) which cause additional power consumption and invocation of congestion control.

Network partitioning and remerging: The randomly moving nodes in an ad hoc wireless network can lead to network partitions. As long as the *TCP* sender, the *TCP* receiver, and all the intermediate nodes in the path between the *TCP* sender and the *TCP* receiver remain in the same partition, the *TCP* connection will remain intact. It is likely that the sender and receiver of the *TCP* session will remain in different partitions and, in certain cases, that only the intermediate nodes are affected by the network partitioning.

The use of sliding-window-based transmission: *TCP* uses a sliding window for flow control. The transmission of packets is decided by the size of the window, and when the ACKs arrive from a destination, further packets are transmitted. This avoids the use of individual fine-grained timers for transmission of each *TCP* flow. Such a design is preferred in order to improve scalability of the protocol in high-bandwidth networks such as the Internet where millions of *TCP* connections may be established with some heavily loaded servers.

The use of a sliding window can also contribute to degraded performance in bandwidth-constrained ad hoc wireless networks where the *MAC* layer protocol may not exhibit short-term and long-term fairness. For

example, the popular *MAC* protocols such as *CSMA/CA* protocol show short-term unfairness, where a node that has captured the channel has a higher probability of capturing the channel again. This unfairness can lead to a number of *TCP ACK* packets being delivered to the *TCP* sender in succession, leading to a burstiness in traffic due to the subsequent transmission of *TCP* segments.



Figure 1: Network Attack

CATEGORIES OF SECURITY ATTACKS

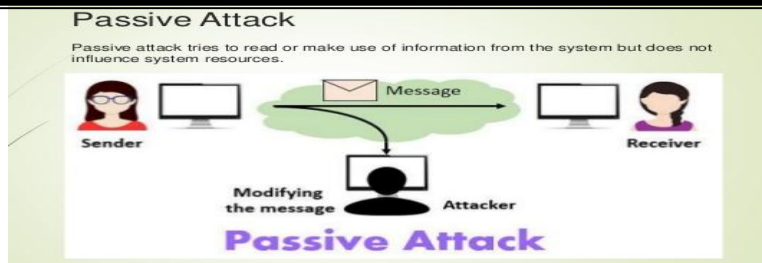
- Security attacks divided into two categories:



Attacks on ad hoc wireless networks can be classified into two broad categories, namely, *passive* and *active* attacks.

A passive attack does not disrupt the operation of the network; the adversary snoops the data exchanged in the network without altering it. Here, the requirement of confidentiality can be violated if an adversary is also able to interpret the data gathered through snooping. Detection of passive attacks is very difficult since the operation of the network itself does not get affected. One way of overcoming such problems is to use powerful encryption mechanisms to encrypt the data being transmitted, thereby making it impossible for eavesdroppers to obtain any useful information from the data overheard.

Figure 2: Passive Attack



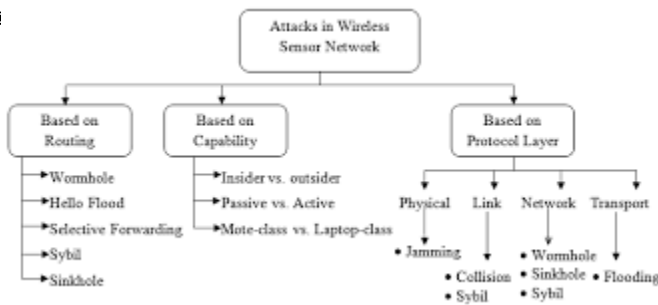
An active attack attempts to alter or destroy the data being exchanged in the network, thereby disrupting the normal functioning of the network. Active attacks can be classified further into two categories, namely, *external* and *internal* attacks. External attacks are carried out by nodes that do not belong to the network.

These attacks can be prevented by using standard security mechanisms such as encryption techniques and firewalls.¹ Internal attacks are from compromised nodes that are actually part of the network. Since the adversaries are already part of the network as authorized nodes, internal attacks are more severe and difficult to detect when compared to external attacks.



Figure 3: Active Attack

Active Attack	Passive Attack
Access and modify information	Access information
System is harmed	No harm to system
Easy to detect than prevent	Difficult to detect than prevent
Threat to Integrity, Availability	Threat to Confidentiality
Masquerading, Repudiation, DOS	Snooping, Traffic analysis



ATTACKS IN WIRELESS SENSOR NETWORKS

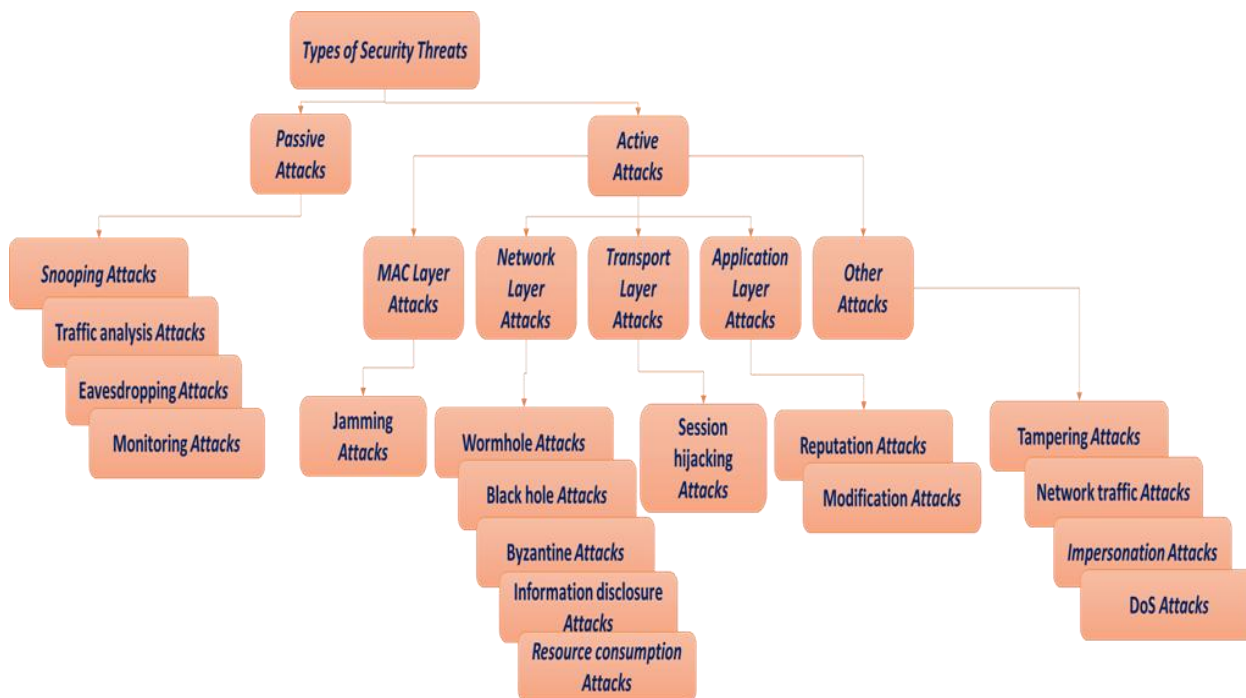


Figure 4: Types of Attack

Active attack

Some active attacks are spoofing attack, Wormhole attack, Modification, Denial of services, Sinkhole, and Sybil attack.

- Spoofing:** When a malicious node mis-presents his identity, so that the sender changes the topology.
- Modification:** When a malicious node performs some modification in the routing route, so that the sender sends the message through the long route. This attack causes communication delay between sender and receiver.
- Wormhole:** This attack is also called the tunnelling attack. In this attack an attacker receives a packet at one point and tunnels it to another malicious node in the network. So that a beginner assumes that he found the shortest path in the network.

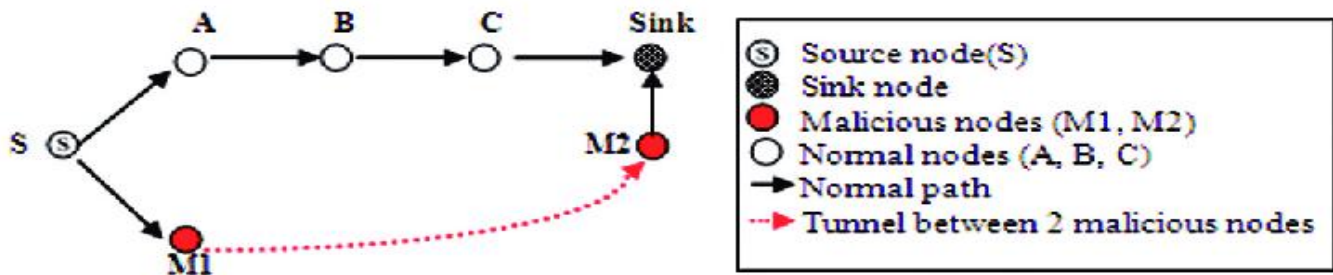


Figure 5: Wormhole attack with use of tunnel between two nodes

- d. **Fabrication:** A malicious node generates the false routing message. This means it generate the incorrect information about the route between devices
- e. **Denial of services :** In denial of services attack, malicious node sending the message to the node and consume the bandwidth of the network. The main aim of the malicious node is to be busy the network node. If a message from unauthenticated node will come, then receiver will not receive that message because he is busy and beginner has to wait for the receiver response.
- f. **Sinkhole:** Sinkhole is a service attack that prevents the base station from obtaining complete and correct information. In this attack, a node tries to attract the data to it from his all neighboring node. Selective modification, forwarding or dropping of data can be done by using this attack
- g. **Sybil:** This attack related to the multiple copies of malicious nodes. The Sybil attack can be happen due to malicious node shares its secret key with other malicious nodes. In this way the number of malicious node is increased in the network and the probability of the attack is also increases. If we used the multipath routing, then the possibility of selecting a path malicious node will be increased in the network.

Passive attack

The names of some passive attacks are traffic analysis, Eavesdropping, and Monitoring .

- a. **Traffic analysis** In the traffic analysis attack, an attacker tries to sense the communication path between the sender and receiver. An attacker can found the amount of data which is travel from the route of sender and receiver. There is no modification in data by the traffic analysis.
- b. **Eavesdropping** This is a passive attack, which occurred in the mobile ad-hoc network. The main aim of this attack is to find out some secret or confidential information from communication. This secrete information may be privet or public key of sender or receiver or any secrete data.
- c. **Monitoring** In this attack in which attacker can read the confidential data, but he cannot edit the data or cannot modify the data.

Advance attacks

- a. **Black hole attack:** Black hole attack is one of the advance attacking which attacker uses the routing protocol to advertise itself as having the best path to the node whose packets it want to intercept. An hacker use the flooding based protocol for listing the request for a route from the initiator, then hacker create a reply message he has the shortest path to the receiver . As this message from the hacker reached to the initiator before the reply from the actual node, then initiator wills consider that, it is the shortest path to the receiver. So that a malicious fake route is create.

- b. **Rushing attack:** In rushing attack, when sender send packet to the receiver, then attacker alter the packet and forward to receiver. Attacker performs duplicate sends the duplicate to the receiver again and again. Receiver assumes that packets come from sender so the receiver becomes busy continuously.
- c. **Replay attack:** In this attack a malicious node may repeat the data or delayed the data. This can be done by originator who intercept the data and retransmit it. At that time, an attacker can intercept the password.
- d. **Byzantine attack:** A set of intermediate node works between the sender and receiver and perform some changes such as creating routing loops, sending packet through non optimal path or selectively dropping packet, which result in disruption or degradation of routing services.
- e. **Location disclosure attack:** Malicious node collects the information about the node and about the route by computing and monitoring the traffic.
- f. So malicious node may perform more attack on the network.

Eavesdropping attack

Eavesdropping attacks occur through the interception of network traffic. By eavesdropping, an attacker can obtain passwords, credit card numbers and other confidential information that a user might be sending over the network. Eavesdropping can be passive or active:

Passive eavesdropping — A hacker detects the information by listening to the message transmission in the network.

Active eavesdropping — A hacker actively grabs the information by disguising himself as friendly unit and by sending queries to transmitters. This is called probing, scanning or tampering.

Detecting passive eavesdropping attacks is often more important than spotting active ones, since active attacks requires the attacker to gain knowledge of the friendly units by conducting passive eavesdropping before.

Data encryption is the best countermeasure for eavesdropping.

Denial-of-service (DoS) and distributed denial-of-service (DDoS) attacks

A denial-of-service attack overwhelms a system's resources so that it cannot respond to service requests. A DDoS attack is also an attack on system's resources, but it is launched from a large number of other host machines that are infected by malicious software controlled by the attacker.

Unlike attacks that are designed to enable the attacker to gain or increase access, denial-of-service doesn't provide direct benefits for attackers. For some of them, it's enough to have the satisfaction of service denial. However, if the attacked resource belongs to a business competitor, then the benefit to the attacker may be real enough. Another purpose of a DoS attack can be to take a system offline so that a different kind of attack can be launched.

One common example is session hijacking, which I'll describe later

There are different types of DoS and DDoS attacks; the most common are TCP SYN flood attack, teardrop attack, smurf attack, ping-of-death attack and botnets.

Wormhole attack: In this attack, an attacker receives packets at one location in the network and tunnels them (possibly selectively) to another location in the network, where the packets are resent into the network [16]. This tunnel between two colluding attackers is referred to as a wormhole.

It could be established through a single long-range wireless link or even through a wired link between the two colluding attackers. Due to the broadcast nature of the radio channel, the attacker can create a wormhole even for packets not addressed to itself. Though no harm is done if the wormhole is used properly for efficient relaying of packets, it puts the attacker in a powerful position compared to other nodes in the network, which the attacker could use in a manner that could compromise the security of the network.

If proper mechanisms are not employed to defend the network against wormhole attacks, most of the existing routing protocols for ad hoc wireless networks may fail to find valid routes.

Blackhole attack: In this attack, a malicious node falsely advertises good paths (*e.g.*, shortest path or most stable path) to the destination node during the path-finding process (in on-demand routing protocols) or in the route update messages (in table-driven routing protocols).

The intention of the malicious node could be to hinder the path-finding process or to intercept all data packets being sent to the destination node concerned.

Byzantine attack: Here, a compromised intermediate node or a set of compromised intermediate nodes works in collusion and carries out attacks such as creating routing loops, routing packets on non-optimal paths, and selectively dropping packets. Byzantine failures are hard to detect. The network would seem to be operating normally in the viewpoint of the nodes, though it may actually be exhibiting Byzantine behavior. Information disclosure: A compromised node may leak confidential or important information to unauthorized nodes in the network. Such information may include information regarding the network topology, geographic location of nodes, or optimal routes to authorized nodes in the network.

Resource consumption attack: In this attack, a malicious node tries to consume/waste away resources of other nodes present in the network. The resources that are targeted are battery power, bandwidth, and computational power, which are only limitedly available in ad hoc wireless networks. The attacks could be in the form of unnecessary requests for routes, very frequent generation of beacon packets, or forwarding of stale packets to nodes. Using up the battery power of another node by keeping that node always busy by continuously pumping packets to that node is known as a sleep deprivation attack.

Routing attacks: There are several types attacks mounted on the routing protocol which are aimed at disrupting the operation of the network. In what follows, the various attacks on the routing protocol are described briefly.

- **Routing table overflow:** In this type of attack, an adversary node advertises routes to non-existent nodes, to the authorized nodes present in the network. The main objective of such an attack is to cause an overflow of the routing tables, which would in turn prevent the creation of entries corresponding to new routes to authorized nodes. Proactive routing protocols are more vulnerable to this attack compared to reactive routing protocols.

- **Routing table poisoning:** Here, the compromised nodes in the networks send fictitious routing updates or modify genuine route update packets sent to other uncompromised nodes. Routing table poisoning may result in sub-optimal routing, congestion in portions of the network, or even make some parts of the network inaccessible.

Packet replication: In this attack, an adversary node replicates stale packets. This consumes additional bandwidth and battery power resources available to the nodes and also causes unnecessary confusion in the routing process.

Route cache poisoning: In the case of on-demand routing protocols (such as the *AODV* protocol [18]), each node maintains a route cache which holds information regarding routes that have become known to the node in the recent past. Similar to routing table poisoning, an adversary can also poison the route cache to achieve similar objectives.

Rushing attack: On-demand routing protocols that use duplicate suppression during the route discovery process are vulnerable to this attack [19]. An adversary node which receives a *RouteRequest* packet from the source node floods the packet quickly throughout the network before other nodes which also receive the same *RouteRequest* packet can react. Nodes that receive the legitimate *RouteRequest* packets assume those packets to be duplicates of the packet already received through the adversary node and hence discard those packets. Any route discovered by the source node would contain the adversary node as one of the intermediate nodes. Hence, the source node would not be able to find secure routes, that is, routes that do not include the adversary node. It is extremely difficult to detect such attacks in ad hoc wireless networks.

Transport Layer Attacks

This section discusses an attack which is specific to the transport layer in the network protocol stack.

Session hijacking: Here, an adversary takes control over a session between two nodes. Since most authentication processes are carried out only at the start of a session, once the session between two nodes gets

established, the adversary node masquerades as one of the end nodes of the session and hijacks the session.

Application Layer Attacks

This section briefly describes a security flaw associated with the application layer in the network protocol stack.

Repudiation: In simple terms, repudiation refers to the denial or attempted denial by a node involved in a communication of having participated in all or part of the communication. As mentioned in Section 9.8, non-repudiation is one of the important requirements for a security protocol in any communication network.

Other Attacks

This section discusses security attacks that cannot strictly be associated with any specific layer in the network protocol stack.

Multi-layer Attacks

Multi-layer attacks are those that could occur in any layer of the network protocol stack. Denial of service and impersonation are some of the common multi-layer attacks. This section discusses some of the multi-layer attacks in ad hoc wireless networks.

Denial of Service: In this type of attack, an adversary attempts to prevent legitimate and authorized users of services offered by the network from accessing those services. A denial of service (DoS) attack can be carried out in many ways. The classic way is to flood packets to any centralized resource (*e.g.*, an access point) used in the network so that the resource is no longer available to nodes in the network, resulting in the network no longer operating in the manner it was designed to operate. This may lead to a failure in the delivery of guaranteed services to the end users. Due to the unique characteristics of ad hoc wireless networks, there exist many more ways to launch a DoS attack in such a network, which would not be possible in wired networks.

DoS attacks can be launched against any layer in the network protocol stack [20]. On the physical and MAC layers, an adversary could employ jamming signals which disrupt the on-going transmissions on the wireless channel. On the network layer, an adversary could take part in the routing process and exploit the routing protocol to disrupt the normal functioning of the network. For example, an adversary node could participate in a session but simply drop a certain number of packets, which may lead to degradation in the QoS being offered by the network. On the higher layers, an adversary could bring down critical services such as the key management service (key management will be described in detail in the next section). Some of the DoS attacks are described below.

Jamming: In this form of attack, the adversary initially keeps monitoring the wireless medium in order to determine the frequency at which the receiver node is receiving signals from the sender. It then transmits signals on that frequency so that error-free reception at the receiver is hindered. Frequency hopping spread spectrum (*FHSS*) and direct sequence spread spectrum (*DSSS*) (described in detail in the first chapter of this book) are two commonly used techniques that overcome jamming attacks.

SYN flooding: Here, an adversary sends a large number of SYN packets^[8] to a victim node, spoofing the return addresses of the SYN packets. On receiving the SYN packets, the victim node sends back acknowledgment (SYN-ACK) packets to nodes whose addresses have been specified in the received SYN packets. However, the victim node would not receive any ACK packet in return. In effect, a half-open connection gets created. The victim node builds up a table/data structure for holding information regarding all pending connections. Since the maximum possible size of the table is limited, the increasing number of half-open connections results in an overflow in the table. Hence, even if a connection request comes from a legitimate node at a later point of time, because of the table overflow, the victim node would be forced to reject the call request.

Distributed DoS attack: A more severe form of the DoS attack is the distributed DoS (DDoS) attack. In this attack, several adversaries that are distributed throughout the network collude and prevent legitimate users from accessing the services offered by the network.

Impersonation: In impersonation attacks, an adversary assumes the identity and privileges of an authorized node, either to make use of network resources that may not be available to it under normal circumstances, or to disrupt the normal functioning of the network by injecting false routing information into the network. An adversary node could masquerade as an authorized node using several methods.

It could by chance guess the identity and authentication details of the authorized node (target node), or it could snoop for information regarding the identity and authentication of the target node from a previous communication, or it could circumvent or disable the authentication mechanism at the target node. A *man-in-the-middle* attack is another type of impersonation attack.

Here, the adversary reads and possibly modifies, messages between two end nodes without letting either of them know that they have been attacked. Suppose two nodes *X* and *Y* are communicating with each other; the adversary impersonates node *Y* with respect to node *X* and impersonates node *X* with respect to node *Y*, exploiting the lack of third-party authentication of the communication between nodes *X* and *Y*.

Device Tampering

Unlike nodes in a wired network, nodes in ad hoc wireless networks are usually compact, soft, and hand-held in nature. They could get damaged or stolen easily.

PASSIVE ATTACK

- In a passive attack the attack monitors the transmissions to obtain message content or monitors traffic flows, but does not modify the message
- There are many types of passive attacks:
 - *Eavesdropping (tapping)*
 - *Traffic Analysis*
 - *Sniffing and snooping*
 - *Spoofing*
 - *Monitoring*
- *Passive attacks* are relatively scarce from a classification perspective, but can be carried out with relative ease, particularly if the traffic is not encrypted. There are two types of passive attacks:
- ***eavesdropping (tapping)***: the attacker simply listens to messages exchanged by two entities. For the attack to be useful, the traffic must not be encrypted. Any unencrypted information, such as a password sent in response to an HTTP request, may be retrieved by the attacker.
- ***traffic analysis***: the attacker looks at the metadata transmitted in traffic in order to deduce information relating to the exchange and the participating entities, e.g. the form of the exchanged traffic (rate, duration, etc.). In the cases where [encrypted data](#) are used, traffic analysis can also lead to attacks by [cryptanalysis](#), whereby the attacker may obtain information or succeed in unencrypting the traffic.
- **Snooping and Replay Attacks.** **Snooping attacks** involve an intruder listening to traffic between two machines on your network. If traffic includes passing unencrypted passwords, an unauthorized individual can potentially access your network and read confidential data.
- Sniffing and snooping should be synonyms. They refer to listening to a conversation. For example, if you login to a website that uses no encryption, your username and password can be sniffed off the
- network by someone who can capture the network traffic between you and the web site.
- **Spoofing** refers to actively introducing network traffic pretending to be someone else. For example, spoofing is sending a command to computer A pretending to be computer B. It is typically used in a scenario where you generate network packets that say they originated by computer B while they really originated by computer C. Spoofing in an email context means sending an email pretending to be someone else.
- **Sniffing:**
Any eavesdropping on existing traffic can be called sniffing, for example you can sniff your own traffic using a network sniffer, I think the WireShark is a good tool for this purpose. WireShark does not change the packets and only capture them and display them, this is the meaning of sniffing.
- **Spoofing:**
When someone(or something) try to introduce himself as another person (or another object), this called spoofing, for example there is IP Spoofing, DNS Spoofing etc in IP Spoofing suppose person A send a packet with source address B not A (not himself IP address) to another host. this is a simple IP Spoofing. And in DNS Spoofing A person tries to put his own IP address as the name of a victim host in the DNS server, this is also a simple DNS Spoofing
- **Monitoring and Eavesdropping:** This is the most common attack which compromises with privacy. Here, the adversary may easily find out the contents of communication just by snooping to the data. Eavesdropping can be effective against protection of privacy, whenever, there is traffic of packet flow containing the control information about the configuration of sensor network. This traffic may contain

most important detailed information through the location server.

- [Deng J, Han R, Mishra S: Countermeasures against traffic analysis attacks in wireless sensor networks. In Proceedings of IEEE/Create Net International Conference on Security and Privacy for Emerging Areas in Communication Networks (SecureComm). Athens, Greece; 2005:113.]
- Deng et al. (2005) identified two classes of passive traffic analysis attacks that can be applied to WSNs, namely, rate monitoring attack and time correlation attack. In the rate monitoring attack, an attacker monitors the packet transmission rate of nodes close to the attacker and moves gradually closer to the nodes that have a higher packet sending rate, eventually reaching the sink.
- Active attacks contrast with [passive attacks](#), in which an unauthorized party monitors networks and sometimes scans for open ports and vulnerabilities. The purpose is to gain information about the target and no data is changed. However, passive attacks are often preparatory activities for active attacks.

ACTIVE ATTACKS

Active attacks take a wider variety of forms, with an almost endless number of possibilities. In an active attack, the attacker is involved in a communication, either by sending or modifying messages. The main types of active attacks are as follows:

–*replay*: this attack consists of recording a series of messages exchanged by two entities, typically a client (the victim) and a server, in order to play them back as-is to the same server with the aim of obtaining access to protected resources, for example. This attack type works on encrypted conversations, unless additional countermeasures have been taken. These countermeasures generally take the form of random number exchanges or time stamping.

–*denial-of-service*: in this case, the attacker aims to exhaust the network or system resources of a machine. One well-known variant is the *distributed denial of service* (DDoS), where a large number of zombie (malware-compromised) machines are used to generate a very large amount of traffic for a given target.

–*man in the middle* (MITM): in this case, the attacker relays communications between victims, in each case

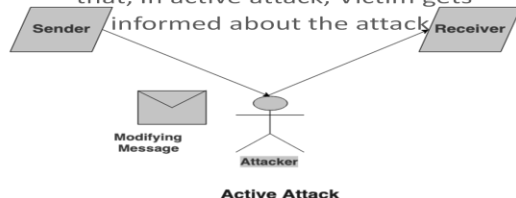
pretending to be the other legitimate correspondent. The attacker therefore intercepts all messages and is able to modify them before transmission to the true recipient, as shown in Figure 3.1. MITM attacks are hard to prevent from a theoretical perspective. When designing a protocol including countermeasures, these measures lead the protocol to question the identity of the correspondent during the [authentication process](#) itself; this prevents production of a proof of identity. By definition, all password-based protocols, including OTPs, are therefore vulnerable to MITM attacks.

DIFFERENCE BETWEEN ACTIVE ATTACK AND PASSIVE ATTACK

Active Attacks:

The attacker efforts to change or modify the content of messages. Active Attack is danger for Integrity as well as availability.

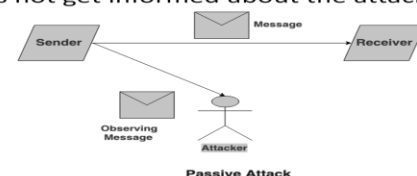
Due to active attack system is always damaged and System resources can be changed. The most important thing is that, In active attack, Victim gets



Passive Attacks:

The attacker observes the content of messages or copy the content of messages. It is danger for Confidentiality.

Due to passive attack, there is no any harm to the system. The most important thing is that In passive attack, Victim does not get informed about the attack.



Active

Active attacks are the type of attacks in which, The attacker efforts to change or modify the content of messages. Active Attack is danger for Integrity as well as availability. Due to active attack system is always damaged and System resources can be changed. The most important thing is that, In active attack, Victim gets informed about the attack.

Attacks:**Passive**

Passive Attacks are the type of attacks in which, The attacker observes the content of messages or copy the content of messages. Passive Attack is danger for Confidentiality. Due to passive attack, there is no any harm to the system. The most important thing is that In passive attack, Victim does not get informed about the attack.

Attacks:

Active Attack	Passive Attack
Modification in Information	Does Not
Integrity & Availability	Confidentiality
Attention is on Detection	Attention is on prevention
System is Always Damaged	No any Harm to the System
Victim gets Informed	Victim does not get informed
System resources can be change	Not change
Influence the services	Information and messages in the system or network are acquire
Information collected and used during executing	Information like passwords, messages by itself
Restrict from entering systems	Easy to prohibited

Difference between Active Attack and Passive Attack

1. In active attack, Modification in information take place.

While in passive attack, Modification in the information does not take place.

2. Active Attack is danger for Integrity as well as availability.

Passive Attack is danger for Confidentiality.

3. In active attack attention is on detection.

While in passive attack attention is on prevention.

4. Due to active attack system is always damaged.

While due to passive attack, there is no any harm to the system.

5. In active attack, Victim gets informed about the attack.

While in passive attack, Victim does not get informed about the attack.

6. In active attack, System resources can be changed.

While in passive attack, System resources are not change.

7. Active attack influence the services of the system.

While in passive attack, information and messages in the system or network are acquired.

8. In active attack, information collected through passive attacks are used during executing.

While passive attack are performed by collecting the information such as passwords, messages by itself.

9. Active attack is tough to restrict from entering systems or networks.

Passive Attack is easy to prohibited in comparison to active attack.

Types of active attacks

-Layer based

- Layer Attacks on WSN Securing wireless ad-hoc networks is a highly challenging issue.
- Attacks can occur in different layers of the network protocol stack.
- Layer based Types of active attacks definitions
 - Attacks at Physical Layer
 - Attacks at Data Link Layer
 - Attacks at Network Layer
 - Attacks at Transport Layer
 - Attacks at Application Layer

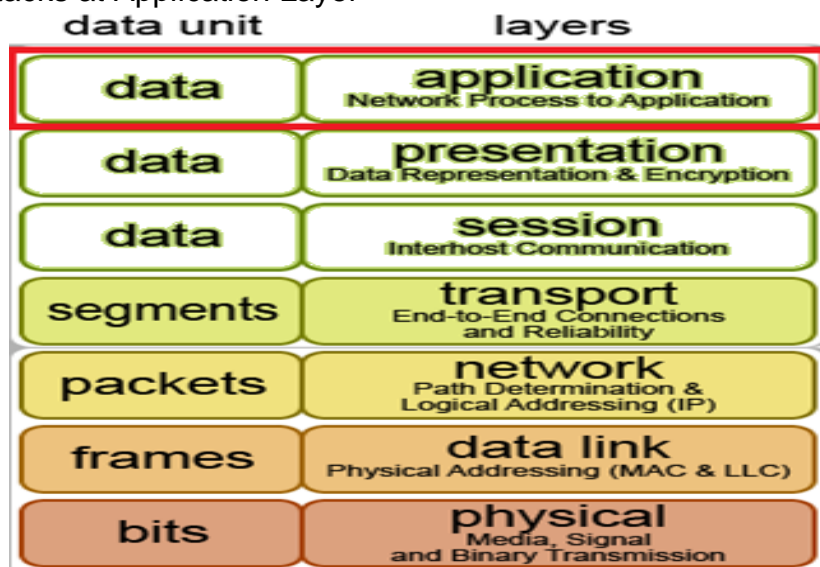


Figure 6: 7 Layer

Layer Attacks on WSN Securing wireless ad-hoc networks is a highly challenging issue. Understanding possible form of attacks is always the first step towards developing good security solutions.

Security of communication in WSN is important for secure transmission of information.

Security means protecting the privacy (confidentiality), availability, integrity and non-repudiation.

Security implies the identification of potential attacks from unauthorized access, use, modification or destruction.

The characteristics of MANETs make them susceptible to many new attacks.

These attacks can occur in different layers of the network protocol stack.

3.3.1. Attacks at Physical Layer Some of the attacks identified at physical layer include eavesdropping, interference, and jamming etc.

3.3.2. Attacks at Data Link Layer The data link layer can classified attacks as to what effect it has on the state of the network as a whole.

3.3.3. Attacks at Network Layer The basic idea behind network layer attacks is to inject itself in the active path from source to destination or to absorb network traffic.

3.3.4. Attacks at Transport Layer Some of the attacks identified at physical layer include TCP/UDP, Session Hijacking, and SYN Flooding etc.

3.3.5. Attacks at Application Layer Some of the attacks identified at physical layer include Repudiation attacks, malicious code attacks, data corruptions attacks etc.

ENHANCING SECURITY VIA PHYSICAL LAYER

- The classic hierarchical structure of communication protocol stack and the examples of security mechanisms deployed at each layer.

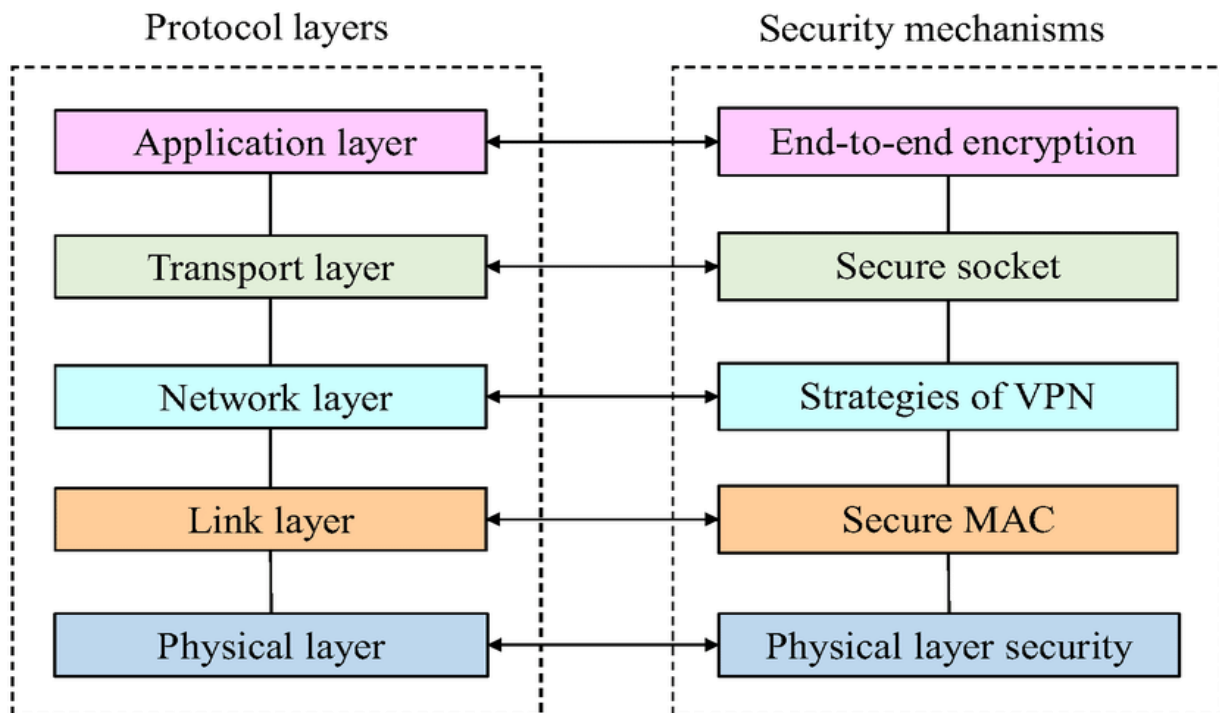


Figure 8: Layer with Security

ATTACKS AT PHYSICAL LAYER

Timing attack: This kind of attack is based on analyzing the time required for executing encryption/decryption algorithms to obtain the key data.

Eavesdropping This is a passive attack, which occurred in the mobile ad-hoc network. The main aim of this attack is to find out some secret or confidential information from communication. This secret information may be private or public key of sender or receiver or any secret data.

Jamming attacks are a subset of denial of service (DoS) attacks in which malicious nodes block legitimate communication by causing intentional interference in networks.

Solutions:

Access Restriction
Encryption.

Some of Attacks occurs at Physical Layer

- Timing attack
- Eavesdropping,
- Interference and Jamming etc.,

Solutions:

- ✓ Access Restriction
- ✓ Encryption.

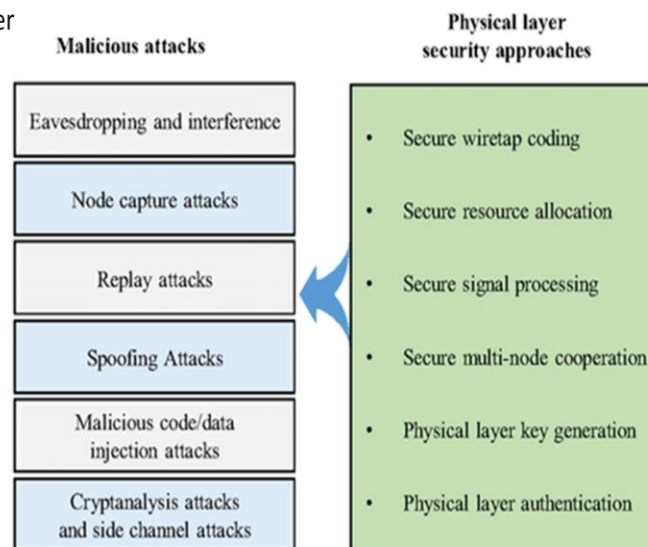


Figure 9: Attacks in Physical Layer

Timing attack: This kind of attack is based on analyzing the time required for executing encryption/decryption algorithms to obtain the key data.

Eavesdropping: This is a passive attack, which occurred in the mobile ad-hoc network. The main aim of this attack is to find out some secret or confidential information from communication. This secret information may be private or public key of sender or receiver or any secret data.

Jamming attacks are a subset of denial of service (DoS) attacks in which malicious nodes block legitimate communication by causing intentional interference in networks.

Solutions:

Access Restriction
Encryption.

ATTACK IN DATA LINK LAYER

Flooding Attacks

These attacks may be better characterized as brute-force flooding attacks.

Sending a steady flood of bogus BPDUs (Bridge Protocol Data Unit) forces continuous spanning

tree recalculation, thereby creating a DoS condition on the computational power of the switches. The bigger the attack bandwidth the more chances it stands to succeed. There are several versions of this attack.

Topology Engagement Attacks

In this category of attacks, a station being serviced by bridges maliciously claims an active role in the tree topology

ATTACKS AT DATA LINK LAYER

Some of the Attacks occurs at Data Link Layer

- Flooding Attacks
- Topology Engagement Attacks

Solutions:

- ✓ Misbehavior Detection.
- ✓ Identity Protection

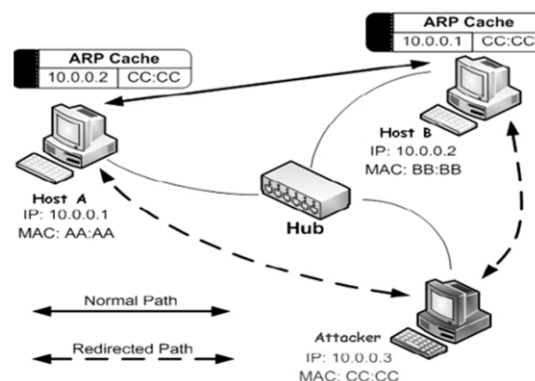


Figure 10: Attacks in Data Layer

ATTACKS IN NETWORK LAYER

- The **attacks** of the **network layer** are:
 - IP spoofing,
 - Hijacking,
 - Smurf,
 - Wormhole,
 - Blackhole,
 - Sybil and sinkhole
 - it was observed that are many other attacks that effectives physical layer
 - Eavesdropping,
 - Jamming
 - Network Injection.
- Solutions:
 - Routing Access Restriction.
 - False Routing Information Detection.

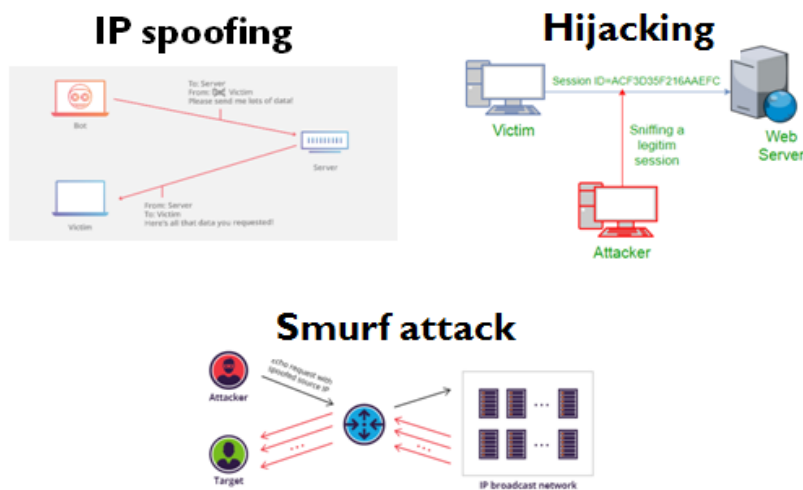


Figure 11: Attacks of the network layer

The **attacks** of the **network layer** are: IP spoofing, hijacking, smurf, wormhole, blackhole, sybil and sinkhole. ... During this study it was observed that are many other **attacks** that effectes physical **layer** such as eavesdropping, jamming and **network** injection.

What is IP spoofing?

IP spoofing is the creation of Internet Protocol (IP) packets which have a modified source address in order to either hide the identity of the sender, to impersonate another computer system, or both. It is a technique often used by bad actors to invoke [DDoS attacks](#) against a target device or the surrounding infrastructure.

Sending and receiving IP packets is a primary way in which networked computers and other devices communicate, and constitutes the basis of the modern internet. All IP packets contain a header which precedes the body of the packet and contains important routing information, including the source address. In a normal packet, the source [IP address](#) is the address of the sender of the packet. If the packet has been spoofed, the source address will be forged.

IP Spoofing is analogous to an attacker sending a package to someone with the wrong return address listed. If the person receiving the package wants to stop the sender from sending packages, blocking all packages from the bogus address will do little good, as the return address is easily changed. Relatedly, if the receiver wants to respond to the return address, their response package will go somewhere other than to the real sender. The ability to spoof the addresses of packets is a core vulnerability exploited by many DDoS attacks.

DDoS attacks will often utilize spoofing with a goal of overwhelming a target with traffic while masking the identity of the malicious source, preventing mitigation efforts. If the source IP address is falsified and continuously randomized, blocking malicious requests becomes difficult. IP spoofing also makes it tough for law enforcement and cyber security teams to track down the perpetrator of the attack.

spoofing is also used to masquerade as another device so that responses are sent to that targeted device instead. Volumetric attacks such as [NTP Amplification](#) and [DNS amplification](#) make

use of this vulnerability. The ability to modify the source IP is inherent to the design of [TCP/IP](#), making it an ongoing security concern.

Tangential to DDoS attacks, spoofing can also be done with the aim of masquerading as another device in order to sidestep authentication and gain access to or “hijack” a user’s session.

How to protect against IP spoofing (packet filtering)

While IP spoofing can’t be prevented, measures can be taken to stop spoofed packets from infiltrating a network. A very common defense against spoofing is ingress filtering, outlined in BCP38 (a Best Common Practice document). Ingress filtering is a form of packet filtering usually implemented on a [network edge](#) device which examines incoming IP packets and looks at their source headers.

If the source headers on those packets don’t match their origin or they otherwise look fishy, the packets are rejected. Some networks will also implement egress filtering, which looks at IP packets exiting the network, ensuring that those packets have legitimate source headers to prevent someone within the network from launching an outbound malicious attack using IP spoofing.

Hijacking is a type of network security attack in which the attacker takes control of a communication - just as an airplane **hijacker** takes control of a flight - between two entities and masquerades as one of them.

For **example**, the time between you first log into your bank account, and then log off after your operation, is a **session**. During a **session hijacking**, a malicious hacker places himself in between your computer and the website's server (Facebook for instance), while you are engaged in an active **session**.

A **Smurf attack** is a form of a distributed denial of service (DDoS) **attack** that renders computer networks inoperable. The **Smurf** program accomplishes this by exploiting vulnerabilities of the Internet Protocol (IP) and Internet Control Message Protocols (ICMP). This attacking technique is a DoS (Denial of Service) attack that happen on the network layer. These attacks are very easy to implement. The idea of this attack is to overload a server with packets.

The attacker will send a high number of packets from a spoofed IP address to the server. The main goal of these attacks is to disable the service the network is providing. Many techniques of attacking are used to achieve this goal. When the attacker wants to realize a Smurf attack, he will transmit to the intended victim a large number of Internet Control Message Protocol (ICMP) by using an IP broadcast address.

To achieve this, the attackers use a program called “smurf” that builds a network packet which appears at the attacked server as it is coming from the trusted IP address. When the attacked server will receive this ICMP packets, by default the server will response to the request. The “smurf” program will generate the necessary amount of ICMP requests to overload the victim with ICMP requests and responses until this device will not be able to provide the necessary services on the network.

Wormhole attacks These attacks are the most severe attacks and complicated attacks in wireless network. Wormhole attacks are very hard to detect and to protect from them. Even when all the communication on the wireless network provides authenticity and confidentiality, a wormhole attack can happen. The attackers will record the packets at one point of the network and retransmits them to another point of the network using private highspeed network, and then replays them into the

network from that point. These kind of attacks are a serious threat against network routing protocols.

Blackhole attack This attack is a type of Denial of Service attack. The main idea of this attack is to drop the incoming and outgoing information between the receiver and the source. In blackhole attack, the attacker will capture all the packets and discard them instead of forwarding them to the destination. The effectiveness of the network will be decreased during this attack, while important packets will not reach the destination. Network parameters such as delay and throughput will be changed during the blackhole attack. The delay will be increased because the packets will not be delivered to the destination. The throughput will be become very less, while it will be used from the black hole attacker

Sybil attack is a method of attacking by stealing or fabricating the identities of other devices on the network. The attacker will use these identities to operate as multiple identities to other network devices. Usually, this method of attack is used against routing algorithms. There are two types of Sybil attacks: external and internal. Using different security mechanisms, external attack can be prevented. To stop an internal attack is used the method of mapping between identity and entity by one to one. Unfortunately, this attack is able to overlap the mapping by creating multiple identities.

Sinkhole attack Malicious device on the network advertises itself to the routing protocols as having the best path to the destination. Some protocols will try to verify if this path is the shortest path to communicate with the destination by using acknowledgment packets. Using these packets, the protocol will understand if this path has reliability, and if the latency is low. The sinkhole attack can transmit false report attacks or reply route messages, making in this way the malicious device look attractive path to forward their packets to the destination

ATTACKS AT TRANSPORT LAYER

- The **attacks** of the **transport layer** are
- TCP Attacks
- UDP Attacks

Solutions:

- ✓ Limit number of connections from a particular node.
- ✓ Header or full packet authentication

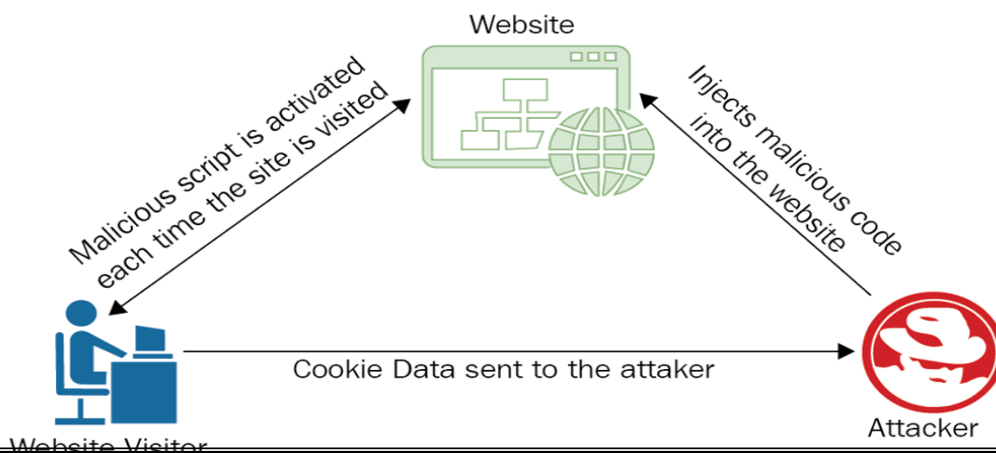


Figure 12: Attacks of the transport layer

The attacks of the transport layer are: TCP sequence prediction, UDP & TCP flooding. During this study it was observed that there are many other **attacks** that effect the physical **layer** such as eavesdropping, jamming and network injection.

TCP flooding attacks. This method of attacking the transport layer is known also as ping flooding. This attack is denial of service (DoS) attack. Attacker sends to the victim a huge number of ping requests. The victim will respond to these ICMP echo requests by sending ping replies. This process will continue until the victim will be blocked replying this ping requests and responses. This attack is one of the oldest attacks of the wireless networks. The method of attacking is not effective as before, because it requires a high bandwidth to saturate the network with ping requests and replies.

UDP flooding attacks. UDP protocol of the transport layer will be attacked, also by flooding attacks. Before the attacking the victim, the attacker will spoof the IP address of other legitimate devices, to hide his identity. The attacker will send to a random or specified port of the victim system a high number of UDP packets.

After this request, victim system will analyze the request and determine what response to reply for this request. If the requested application cannot be offered from the system, the system will reply with the message: "Destination Unreachable". The attacker will send UDP packets to the victim to deplete the network bandwidth, crash the system of the victim, or at least to degrade the performance of the system attacked.

TCP sequence prediction attack. To deliver the packets ordered at the destination, TCP transport protocol uses a sequence number for each packet transmitted. At beginning, the attacker will listen the communications between two hosts (host 1 and host 2). One of this hosts is the victim of the attack, let's suppose that the victim will be host 1. Then, the attacker will send request packets to host 2 with the spoofed IP address of the trusted device. The attacker will flood host 2 until a Denial of Service attack will happen and the communication between host 1 and host 2 will stop. After this step, the attacker can issue the packets with correct sequence number, which host 1 is expecting from host 2. Attacker will send the packet with correct sequence number to host A with the spoofed IP address of host B. This packet can damage the network by asking the victim to run malicious scripts or to execute different commands.

APPLICATION LAYER ATTACKS,

Also known as **Layer 7 Attacks** after the OSI model,

- BGP Hijacking
- Slowloris, Slow Post, Slow Read
- HTTP(s) Flooding
- Low and Slow Attack
- Large Payload POST
- Mimicked User Browsing

Solutions:

- ✓ Data Integrity Protection.
- ✓ Data Confidentiality Protection.

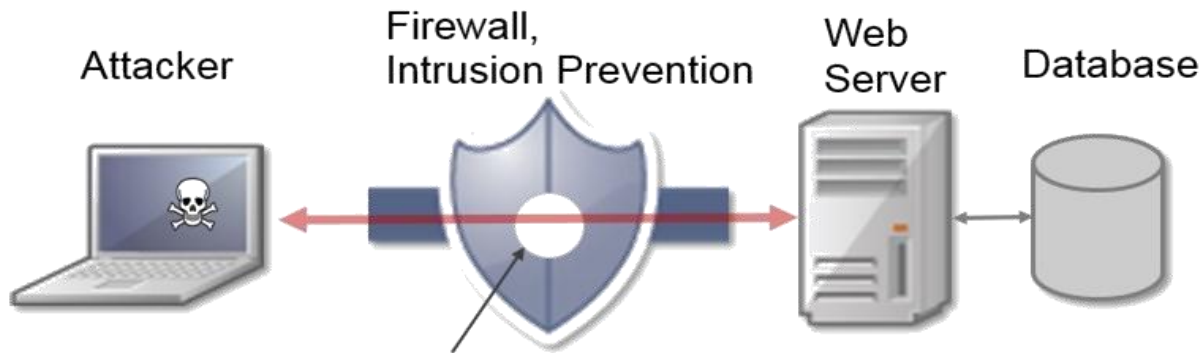


Figure 13: Application Layer Attacks

Application Layer Attacks

Application Layer attacks target some aspect of an application or service at Layer-7. These are the deadliest kind of attacks as they can be very effective with as few as one attacking machine generating a low traffic rate (this makes these attacks very difficult to pro-actively detect and mitigate). Application layer attacks have come to prevalence over the past three or four years and simple application layer flood attacks (HTTP GET flood etc.) have been some of the most common denial of service attacks seen in the wild.

Today's sophisticated attackers are blending volumetric, state exhaustion and application-layer attacks against infrastructure devices all in a single, sustained attack. These cyber attacks are popular because they difficult to defend against and often highly effective.

The problem doesn't end there. According to Frost & Sullivan, DDoS attacks are "increasingly being utilized as a diversionary tactic for targeted persistent attacks." Attackers are using DDoS tools to distract the network and security teams while simultaneously trying to inject advanced persistent threats such as malware into the network, with the goal of stealing IP and/or critical customer or financial information.

Application Layer Attacks, also known as **Layer 7 Attacks** after the OSI model, include **attacks** that target vulnerabilities in a server's web services like Apache, IIS, NGINX, and so on, as well as floods using GET and POST methods over HTTP/S. This DDoS vector accounts for about 20 percent of all DDoS **attacks**

Application layer DDoS attacks are designed to attack the application itself, focusing on specific vulnerabilities or issues, resulting in the application not being able to deliver content to the user. Application layer attacks are designed to attack specific applications, the most common is web servers, but can include any application such SIP voice services and BGP.

Such attacks are usually low-to-mid volume since they have to conform to the protocol the application is using, which often involves protocol handshakes and protocol/application compliance. This means that these attacks will primarily be launched using discrete intelligent clients, usually [Internet of Things \(IoT\)](#) devices, and cannot be spoofed.

What Are the Different Types of Application Layer Attack?

When looking at [DDoS trends](#) over time, attacks are cyclical in nature. Attackers develop new attack types and vectors, which are used to launch a new wave of attacks. As defenders become more proficient in stopping these new attacks, the attackers develop new types of attacks and the cycle repeats itself.

The proliferation of insecure IoT devices in recent years has been a boon to the DDoS attackers as there are now a nearly unlimited number of intelligent devices which can be used to launch more advanced application layer attacks.

What is a BGP Hijacking?

The Border Gateway Protocol (BGP) is used to direct traffic across the Internet, allowing networks to exchange “reachability information” to facilitate reaching other networks. BGP hijacking is a form of [application-layer DDoS attack](#) that allows an attacker to impersonate a network, using a legitimate network prefix as their own. When this “impersonated” information is accepted by other networks, traffic is inadvertently forwarded to the attacker instead of its proper destination.

What is a Slowloris Attack?

Slowloris is an [application layer DDoS attack](#) which uses partial HTTP requests to open connections between a single computer and a targeted Web server, then keeping those connections open for as long as possible, thus overwhelming and slowing down the target. This type of [DDoS attack](#) requires minimal bandwidth to launch and only impacts the target web server, leaving other services and ports unaffected. Slowloris attacks can target many type of Web server software, but has proven highly-effective against Apache 1.x and 2.x.

What is a Slow Post Attack?

In a Slow Post DDoS attack, the attacker sends legitimate HTTP POST headers to a Web server. In these headers, the sizes of the message body that will follow are correctly specified. However, the message body is sent at a painfully low speed. These speeds may be as slow as one byte every two minutes.

Since the message is handled normally, the targeted server will do its best to follow specified rules. As in a [Slowloris attack](#), the server will subsequently slow to a crawl. Making matters worse, when attackers launch hundreds or even thousands Slow POST attacks at the same time, server resources are rapidly consumed, making legitimate connections unachievable.

What is a Slow Read Attack?

A slow read DDoS attack involves an attacker sending an appropriate HTTP request to a server, but then reading the response at a very slow speed, if at all. By reading the response slowly – sometimes as slow as one byte at a time – the attacker prevents the server from incurring an idle connection timeout. Since the attacker sends a Zero window to the server, the server assumes the client is actually reading the data and therefore keeps the connection open. This has the cumulative effect of consuming server resources, thus preventing legitimate requests from going through.

A Slow Read attack is characterized by a very low number for the TCP Receive Window size, while at the same time draining the attacker’s TCP receive buffer slowly. This in turn creates a condition where the data flow rate is extremely low.

What is an HTTP Flooding Attack?

An HTTP flood attack utilizes what appear to be legitimate HTTP GET or POST requests to attack a web server or application. These [flooding attacks](#) often rely on a botnet, which is a group of Internet-connected computers that have been maliciously appropriated through the use of malware such as a Trojan Horse.

What is a Low and Slow Attack?

A low and slow attack, also known as a slow-rate attack, involves what appears to be legitimate traffic at a very slow rate. This type of [state exhaustion attack](#) targets application and server resources and is difficult to distinguish from normal traffic. Common attack tools include [Slowloris](#), Sockstress, and R.U.D.Y. (R U Dead Yet?), which create legitimate packets at a slow rate, thus allowing the packets to go undetected by traditional mitigation strategies.

Low and slow attacks are often HTTP focused, but can also involve Long-Lived TCP sessions (slow transfer rates) that attack any TCP-based service.

What is a Large Payload Post Attack?

A Large Payload Post is a class of [HTTP DDoS attack](#) where the attacker abuses XML encoding used by web servers. In this type of attack, a web server is sent a data structure encoded in XML, which the server then attempts to decode, but is compelled to use an excessive amount of memory, thus overwhelming the system and crashing the service.

These types of [DDoS attacks](#) are also referred to as "Oversize Payload Attacks" or "Jumbo Payload Attacks."

What is Mimicked User Browsing?

A Mimicked User Browsing DDoS attack involves botnets that pose as legitimate users attempting to access a website. A sufficiently high volume of these bots will ultimately overwhelm the target website causing it to crash, or making it impossible for legitimate traffic to get through. The common motive behind such [DDoS attacks](#) may be financial or political.

POSSIBLE SOLUTIONS FOR JAMMING ATTACK

To address jamming problem

- localization,
- detection and
- countermeasure mechanisms

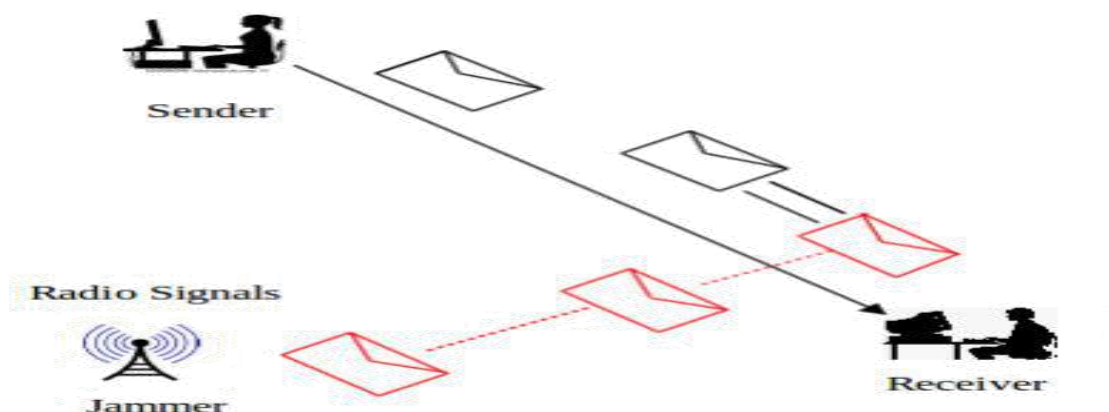


Figure 14: Jamming Attack

Because of the proliferation of wireless technologies, jamming in wireless networks has become a major problem due to the ease in blocking communication in wireless networks.

Jamming attacks are a subset of denial of service (DoS) attacks in which malicious nodes block legitimate communication by causing intentional interference in networks.

TAMPERING

- TAMPERING

- Tampering means changing or deleting a resource without authorization.
- Data tampering or data manipulation is a way that a hacker or a malicious user gets into a web site and changes, deletes or to access unauthorized files.
- Data tampering indirectly by using a script exploit to mask itself as a user input from a page.
- Data tampering through;
 - Cookie tampering,
 - HTML form field tampering
 - URL query string tampering
 - Password cracking tampering
 - HTTP header tampering

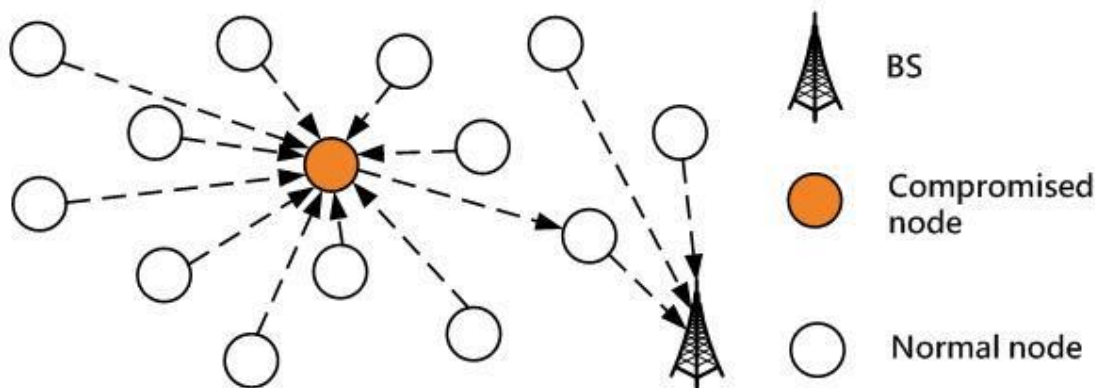


Figure 15: Tampering

Tampering: it is the result of physical access to the node by an attacker; the purpose will be to recover cryptographic material like the keys used for ciphering

Tampering means changing or deleting a resource without authorization. A web application is an application

that is accessed through a web browser over the internet. Data tampering in web applications simply means a way in which a hacker or a malicious user gets into a web site and changes, deletes or to access unauthorized files. A hacker or malicious user can also tamper indirectly by using a script exploit that is the hacker would get the script to execute by masking it as a user input from a page or as a web link.

Data tampering or data manipulation can usually be done through the following ways: Cookies, HTML Form Fields, URL Query Strings, HTTP Headers and Password Cracking.

Data tampering or data manipulation is a way that a hacker or a malicious user gets into a web site and changes, deletes or to access unauthorized files. Hackers or malicious users can cause data tampering indirectly by using a script exploit to mask itself as a user input from a page.

Data tampering can be done either through; cookie tampering, that is, modifying cookies to allow access for files and documents; HTML form field tampering which deals with hackers changes values in HTML forms; URL query string tampering which deals with changing values in HTML forms accessed through the address bar of a user; Password cracking tampering which consists of a hacker using an application or tool which allows him/her to obtain the unknown password for

unauthorized access to a computer system; HTTP header tampering which allows malicious users to modify data through available proxies.

Possible solutions for DATA tampering

Data tampering could become the greatest cybersecurity threat organizations face

- **Data tampering**
 - **Anti-Tamper**
 - Copy-On-Write (COW)

Data tampering or data manipulation can usually be done through the following ways

- **Cookies**
- **HTML Form Fields**
- **URL Query Strings**
- **HTTP Headers and**
- **Password Cracking**
- Cyber Command and Chief of the Central Security Service, has said that **data tampering could become the greatest cybersecurity threat organizations face**. Data tampering could be an act of revenge by a disgruntled employee, industrial espionage by a competitor or the work of hactivists or a rogue nation state. Whatever the root cause, the prospects of such a security breach are alarming.
- Data **tampering** is the act of deliberately modifying (destroying, manipulating, or editing) data through unauthorized channels. ... With data at rest, a system application can suffer a **security** breach and an unauthorized intruder could deploy malicious code that corrupts the data or underlying programming code.
- **COOKIE TAMPERING**
- Cookies are used as a mechanism to store user details and preferences and other data including session tokens. Cookies that are persistent and non-persistent, insecure or secure can be altered by the user and sent to the server with Uniform Resource Locator requests, therefore any malicious user or hacker can modify cookie content to his advantage allowing the attacker to access the files needed.
- **HTML FORM FIELDS TAMPERING**
- When a user makes selections or changes on a web or an HTML page, the selection is stored as form field values which are then delivered to the application as an HTTP request. HTML usually stores field values as Hidden Fields, which are not shown to the screen of the user but are collected and submitted as strings or parameters during form submissions. Whether these form fields can be hidden, pre-selected or free form, they can all be tampered or manipulated by the hacker to submit whatever
 - values he/she chooses.
- **URL QUERY STRINGS TAMPERING**
- URL tampering comes with all of the problems associated with Hidden Form Fields. One of two methods is used by the HTML forms to submit their results, either POST or GET. Usually the method GET is used, showing all form element names and their values in the query string of the next URL that the hacker sees. Hackers find tampering with query strings easier than tampering with hidden form fields. All that the hacker has to do is look at the URL in the user's address bar.
 - For example; a web page allows the authenticated user to select one of his pre-populated accounts from a drop-down box and debit the account with a fixed unit amount. His/her choices are recorded by pressing the submit button. The page is actually storing the entries

in form field values and submitting them using a form submit command. The command sends the following HTTP request:
<http://www.victim.com/example?accountnumber=12345&debitamount=1>, now all what the hacker has to do is could construct his/her own account number and change the parameters like the following:

<http://www.victim.com/example?accountnumber=67891&creditamount=999999999>.

(Curphey M, Smith T et al. (The Open Web Application Security Project), 2002)

- **Anti-Tamper (AT)** is defined as the Systems Engineering and System Security Engineering activities intended to prevent and/or delay exploitation of critical technologies in U.S. weapon systems, training devices, and maintenance support equipment.+
- A **possible solution** to this **data tampering** problem is something called copy-on-write (COW). Each time a **database** is modified, delta snapshots are taken. ... While the **database** may become encrypted, restoring the file system to a pre-attack state would end any downtime and retrieve lost **data**
- **Story behind the Risks of Data Tampering and How to Prevent It**
- Most businesses are aware of the risks associated with data theft or exposure. The recent Equifax breach, which compromised the sensitive information of nearly half the U.S. population, is only the most recent in a series of cybercrimes in which massive amounts of data were exfiltrated.
- But what if data wasn't stolen but *modified*. What if, for example, someone tampered with the quality assurance data of a manufacturing plant? Or a bank's account balances? Or the patient information held by a hospital? How long might it take an organization to discover that its data had been modified? How would the organization recover?
- Admiral Michael S. Rogers of the U.S. Navy, who serves as Director of the National Security Agency, Commander of the U.S. Cyber Command and Chief of the Central Security Service, has said that **data tampering could become the greatest cybersecurity threat organizations face**. Data tampering could be an act of revenge by a disgruntled employee, industrial espionage by a competitor or the work of hactivists or a rogue nation state. Whatever the root cause, the prospects of such a security breach are alarming.
- **One type of data tampering has unfortunately become commonplace — ransomware**. In a ransomware attack, cybercriminals encrypt an organization's data and demand payment of a ransom to obtain the decryption key. According to data from Quick Heal Security Labs, more than 25,000 ransomware infections were reported daily on Windows system in the third quarter of 2017 alone.
- **What's more, many cyberattacks involve some kind of data tampering**. Hacker often insert new files that perform some malicious activity, change a configuration file to gain control of a system, or delete or modify system log files to cover their tracks.
- Clearly, it's important that organizations be able to identify successful and unsuccessful attempts to change critical files. But how do you go about it? That's the function of a security control known as file integrity monitoring (FIM).
- Also known as change monitoring, **FIM is the process of examining critical files to see if, when and how they change**. FIM systems compare the current state of a file to a known, good baseline, typically using a cryptographic algorithm to generate a mathematical value called a checksum. Files may be monitored at predefined intervals, randomly or in real time.
- Given the large amount of data stored by organizations today, monitoring all files typically isn't practical. FIM systems can be resource-intensive, particularly when it comes to very large files and those that change constantly. That said, it's important to monitor any files that a hacker might seek to compromise, or that might cause downtime or data loss if a legitimate user makes an error

- With that in mind, FIM systems generally are used to monitor user identities, privileges and credentials, security settings, operating system and application files, configuration files, and encryption key stores. Monitoring of log files is especially important, and should ensure that only systems and applications write data to logs and that log files are frequently collected and stored in a separate management system. Organizations may also use FIM to monitor files that contain sensitive content such as customer information and trade secrets.
- In our next post, we'll describe what to look for in a FIM solution and how to integrate FIM into your security strategy.

PREVENTION AND COUNTER-MEASURES

- To use a firewall and windows security
- Guarding against script exploits
- Appropriate and safe steps against malicious executable code.
- Counter-Measures to prevent data tampering
 - Data signing and harsing
 - using digital signatures,
 - using strong authorization,
 - using tamper resistant protocols
 - using secure communication links
 - using strong and powerful firewalls
 - using complicated passwords and
 - blocking IP addresses for failed login attempts
- Using access controls to protect data
- Using role based security

PREVENTION AND COUNTER-MEASURES

- A primary defense against data tampering is to use a firewall and windows security to lock down important files, directories and other resources. The web application should also run with minimum privileges. Guarding against script exploits by not trusting any information that comes from a user or even from a database. Appropriate and safe steps should be taken when getting information from untrusted sources, to make sure it does not contain any malicious executable code.
- Counter-Measures to prevent data tampering are done through the following ways: by using data signing and harsing, using digital signatures, using strong authorization, using tamper resistant protocols across communication links, using secure communication links with protocols that provide message integrity, also by using strong and powerful firewalls, and long passwords that consist of alphanumeric characters, by also blocking IP addresses for a certain period of time which will cause repeated failed login attempts by the attacker.
- Also by using access controls to protect data in persistent stores to ensure that only authorized users can access and modify the data, and by using role based security to define which users can view data and which users can modify data.

BLACK HOLE ATTACK

- Blackhole attack is a type of denial-of-service attack
- Black hole attack is one of the possible attacks in WSN.

- The attacker drops packets selectively
- Intermediate malicious node will suffer from partial or total data loss.

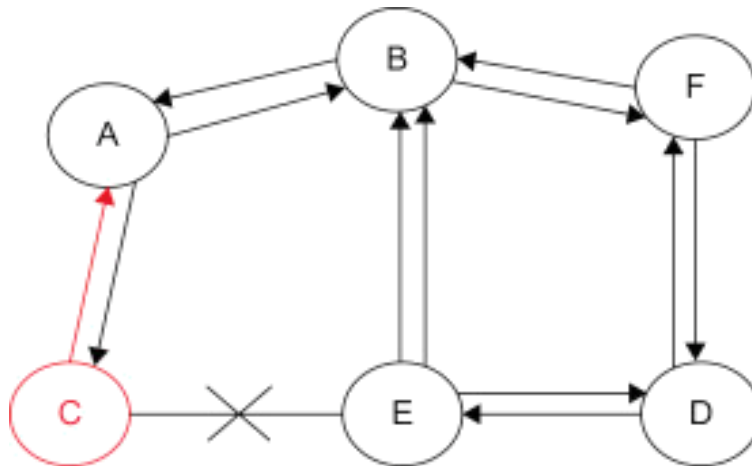


Figure16: Black Hole Attack

In computer networking, a packet drop **attack** or **blackhole attack** is a type of denial-of-service **attack** in which a router that is supposed to relay packets instead discards them. This usually occurs from a router becoming compromised from a number of different causes.

Black hole attack is one of the possible **attacks in WSN**. In **black hole attack** a malicious node sends the RREP (Route Reply) message to the source node as a shortest path to the destination node then the sender node send a data packet to the malicious node in the network. In this type of attack, the attacker drops packets selectively, or all control and data packets that are routed through him. Therefore, any packet routed through this intermediate malicious node will suffer from partial or total data loss.

Possible solutions for black hole attack

- Routing Protocol Enhancements
- Safe Route
- Find and Secure Alternate Route
- Neighboring Nodes Maintenance
- Malicious Paths Identification
- Timely Data Transmission
- Link Break Detection
- We propose a solution that is an enhancement of the basic AODV routing protocol, which will be able to avoid black holes. To reduce the probability it is proposed to wait and check the replies from all the neighboring nodes to find a safe route. According to this proposed solution the requesting node without sending the DATA packets to the reply node at once, it has to wait till other replies with next hop details from the other neighboring nodes. After receiving the first request it sets timer in the 'TimerExpiredTable', for collecting the further requests from different nodes. It will store the 'sequence number', and the time at which the packet arrives, in a 'Collect Route Reply Table'

- (CRRT). The time for which every node will wait is proportional to its distance from the source. It calculates the 'timeout' value based on arriving time of the first route request. After the timeout value, it first checks in CRRT whether there is any repeated next hop node. If any repeated next hop node is present in the reply paths it assumes the paths are correct or the chance of malicious paths is limited.
- After selecting the route between the source and the destination and during data transmission, if any node participating in the route moves, then the node that tries to send data will detect a link break. Then it tries to salvage the packet, that is, it searches in its cache to find an alternate route to reach the destination. If there is any route, then it will send data through that new route.

FLOODING ATTACK

- **Flooding attack** involves the generation of spurious messages to increase traffic on the network for consuming server's or network's resources
- **DoS and DDoS attacks can be divided into three types:**
 - Volume Based Attacks
 - UDP floods,
 - ICMP floods, and
 - spoofed-packet floods
 - Protocol Attacks
 - SYN floods
 - fragmented packet attacks
 - Ping of Death
 - Smurf DDoS
- - Application Layer Attacks

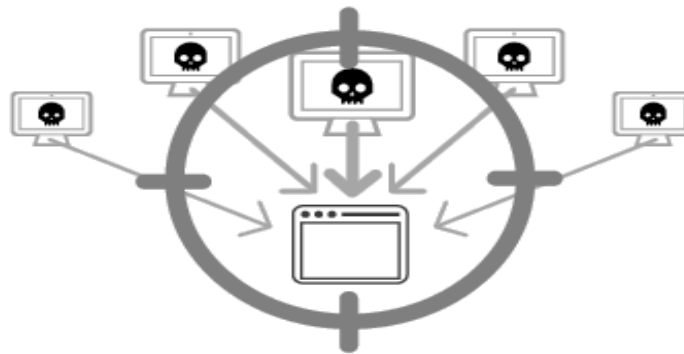


Figure 17(a) : Flooding Attack

HTTP Flood Attack

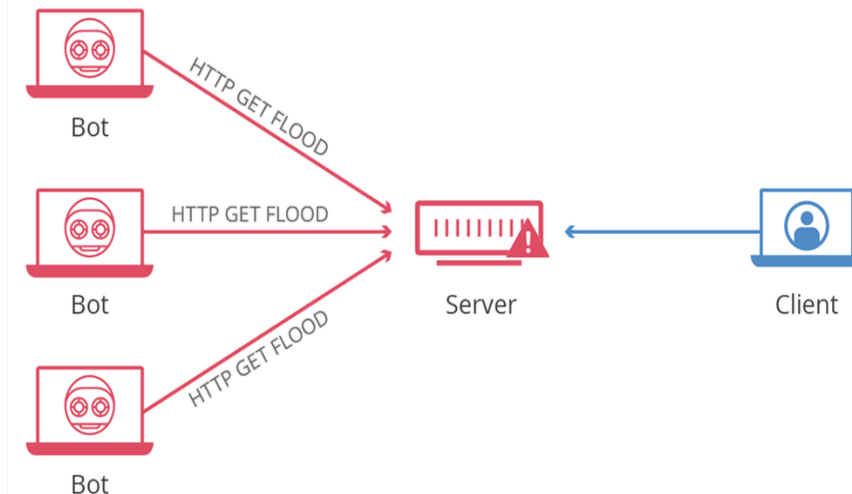


Figure 17(b) : Flooding Attack

An **HTTP flood attack** is a type of volumetric distributed denial-of-service (DDoS) **attack** designed to overwhelm a targeted server with HTTP requests. Once the target has been saturated with requests and is unable to respond to normal traffic, denial-of-service will occur for additional requests from actual users. distributed denial-of-service (DDoS) **attack**:

A **distributed denial-of-service (DDoS) attack** occurs when multiple systems flood the bandwidth or resources of a targeted system, usually one or more web servers. Such an **attack** is often the result of multiple compromised systems (for example, a botnet) flooding the targeted system with traffic.

Distributed denial of service attack (DDoS) definition

A distributed denial of service (DDoS) attack is a malicious attempt to make an online service unavailable to users, usually by temporarily interrupting or suspending the services of its hosting server.

A DDoS attack is launched from numerous compromised devices, often distributed globally in what is referred to as a [botnet](#). It is distinct from other denial of service (DoS) attacks, in that it uses a single Internet-connected device (one network connection) to flood a target with malicious traffic. This nuance is the main reason for the existence of these two, somewhat different, definitions. Broadly speaking, DoS and DDoS attacks can be divided into three types:

Volume Based Attacks

Includes UDP floods, ICMP floods, and other spoofed-packet floods. The attack's goal is to saturate the bandwidth of the attacked site, and magnitude is measured in bits per second (Bps).

Protocol Attacks

Includes SYN floods, fragmented packet attacks, Ping of Death, Smurf DDoS and more. This type of attack consumes actual server resources, or those of intermediate communication equipment, such as firewalls and load balancers, and is measured in packets per second (Pps).

Application Layer Attacks

Includes low-and-slow attacks, GET/POST floods, attacks that target Apache, Windows or OpenBSD vulnerabilities and more. Comprised of seemingly legitimate and innocent requests, the goal of these attacks is to crash the web server, and the magnitude is measured in Requests per second (Rps).

Common DDoS attacks types

Some of the most commonly used DDoS attack types include:

UDP Flood

A UDP flood, by definition, is any DDoS attack that floods a target with User Datagram Protocol (UDP) packets. The goal of the attack is to flood random ports on a remote host. This causes the host to repeatedly check for the application listening at that port, and (when no application is found) reply with an ICMP 'Destination Unreachable' packet. This process saps host resources, which can ultimately lead to inaccessibility.

ICMP (Ping) Flood

Similar in principle to the UDP flood attack, an ICMP flood overwhelms the target resource with ICMP Echo Request (ping) packets, generally sending packets as fast as possible without waiting for replies. This type of attack can consume both outgoing and incoming bandwidth, since the victim's servers will often attempt to respond with ICMP Echo Reply packets, resulting a significant overall system slowdown.

SYN Flood

A SYN flood DDoS attack exploits a known weakness in the TCP connection sequence (the "three-way handshake"), wherein a SYN request to initiate a TCP connection with a host must be answered by a SYN-ACK response from that host, and then confirmed by an ACK response from the requester. In a SYN flood scenario, the requester sends multiple SYN requests, but either does not respond to the host's SYN-ACK response, or sends the SYN requests from a spoofed IP address. Either way, the host system continues to wait for acknowledgement for each of the requests, binding resources until no new connections can be made, and ultimately resulting in [denial of service](#).

Ping of Death

A ping of death ("POD") attack involves the attacker sending multiple malformed or malicious pings to a computer. The maximum packet length of an IP packet (including header) is 65,535 bytes. However, the Data Link Layer usually poses limits to the maximum frame size – for example 1500 bytes over an Ethernet network. In this case, a large IP packet is split across multiple IP packets (known as fragments), and the recipient host reassembles the IP fragments into the complete packet. In a Ping of Death scenario, following malicious manipulation of fragment content, the recipient ends up with an IP packet which is larger than 65,535 bytes when reassembled. This can overflow memory buffers allocated for the packet, causing denial of service for legitimate packets.

Slowloris

Slowloris is a highly-targeted attack, enabling one web server to take down another server, without affecting other services or ports on the target network. Slowloris does this by holding as many connections to the target [web server open for as long as possible](#). It accomplishes this by creating connections to the target server, but sending only a partial request. Slowloris constantly sends more HTTP headers, but never completes a request. The targeted server keeps each of these false connections open. This eventually overflows the maximum concurrent connection pool, and leads to denial of additional connections from legitimate clients.

NTP Amplification

In NTP amplification attacks, the perpetrator exploits publically-accessible Network Time Protocol (NTP) servers to overwhelm a targeted server with UDP traffic. The attack is defined as an amplification assault because the query-to-response ratio in such scenarios is anywhere between

1:20 and 1:200 or more. This means that any attacker that obtains a list of open NTP servers (e.g., by a using tool like Metasploit or data from the Open NTP Project) can easily generate a devastating high-bandwidth, high-volume DDoS attack.

HTTP Flood

In an HTTP flood DDoS attack, the attacker exploits seemingly-legitimate HTTP GET or POST requests to attack a web server or application. HTTP floods do not use malformed packets, spoofing or reflection techniques, and require less bandwidth than other attacks to bring down the targeted site or server. The attack is most effective when it forces the server or application to allocate the maximum resources possible in response to every single request.

Zero-day DDoS Attacks

The “Zero-day” definition encompasses all unknown or new attacks, exploiting vulnerabilities for which no patch has yet been released. The term is well-known amongst the members of the hacker community, where the practice of trading zero-day vulnerabilities has become a popular activity.

POSSIBLE SOLUTIONS FOR JAMMING, TAMPERING, BLACK HOLE ATTACK, FLOODING ATTACK

- **Volume Based Attacks**
- **Protocol Attacks**
- **Application Layer Attacks**

Solutions mitigate DDoS damage

seamlessly and comprehensively protects websites against all three types of DDoS attacks, addressing each with a unique toolset and defense strategy:

Volume Based Attacks

counters these attacks by absorbing them with a global network of scrubbing centers that scale, on demand, to counter multi-gigabyte DDoS attacks.

Protocol Attacks

mitigates this type of attack by blocking “bad” traffic before it even reaches the site, leveraging visitor identification technology that differentiates between legitimate website visitors (humans, search engines etc.) and automated or malicious clients.

Application Layer Attacks

mitigates Application Layer attacks by monitoring visitor behavior, blocking known bad bots, and challenging suspicious or unrecognized entities with JS test, Cookie challenge, and even CAPTCHAs.

all these scenarios, applies its [DDoS protection](#) solutions outside of your network, meaning that only filtered traffic reaches your hosts.

Moreover, maintains an extensive DDoS threat knowledge base, which includes new and emerging attack methods. This constantly-updated information is aggregated across our entire network – identifying new threats as they emerge, detecting known malicious users, and applying remedies in real-time across all websites.

KEY DISTRIBUTION AND MANAGEMENT

The objective of key management is to maintain keying relationships and keying material in a manner that counters relevant threats. In practice an additional objective is conformance to a relevant security policy

- Delivers a **key** to two parties
- Sort of mechanism or protocol need for security

- Sort of mechanism or protocol
- The objective of key management is to maintain keying relationships and keying material in a manner that counters relevant threats
- In practice an additional objective is conformance to a relevant security policy
- **Key Management** and **Distribution**. **Key distribution** is the function that delivers a **key** to two parties who wish to exchange secure encrypted data. Some sort of mechanism or protocol is needed to provide for the secure **distribution** of **keys**.
- Keys are used as part of encryption and authentication functions to lock and unlock messages. While a particular encryption *algorithm* is often published and well known, the keys used to make each encryption unique must be kept secure and private. But there are logistics problems in exchanging
- keys. If you send an encrypted message to a friend, your friend will need a key to decrypt the message. The process of getting that key to your friend may be compromised. This topic describes methods for exchanging keys in secure ways over open networks like the Internet.
- The one thing to avoid in any key exchange is obvious: never send the actual key over the network in the open. If Alice and Bob need to exchange keys, they may be able to do so over the phone (if it's a relatively short alphanumeric string). They could also meet in person or use a public-key scheme as described later. In any case, once they have a "shared secret key," they can use it for authentication and to establish trust.
- Manual key exchange methods (security considerations)
- Public keys and certificates
- Diffie-Hellman Key Exchange
- IKE (Internet Key Exchange)
- ISAKMP (Internet Security Association and Key Management Protocol)
- OAKLEY
- SKEME
- Key Recovery

KEY MANAGEMENT

- Key management is the set of techniques and procedures supporting the establishment and maintenance of keying relationships between authorized parties.
- A keying relationship is the state wherein communicating entities share common data(keying material) to facilitate cryptography techniques. This data may include public or secret keys, initialization values, and additional non-secret parameters

KEY MANAGEMENT TECHNIQUES AND PROCEDURES

- Initialization
- Generation, Distribution, and Installation
- Controlling
- Update, Revocation, and Destruction
- Storage, Backup/Recovery, and Archive

Key management encompasses techniques and procedures supporting:

1. initialization of systems users within a domain;
2. generation, distribution, and installation of keying material;
3. controlling the use of keying material;
4. update, revocation, and destruction of keying material; and
5. storage, backup/recovery, and archival of keying material.

SECURITY POLICY & THREATS

- Security policy explicitly or implicitly defines the threats a system is intended to address. Security policy may affect the stringency of cryptographic requirements, depending on the susceptibility of the environment in questions to various types of attack.
- **Threats**
 - Compromise of confidentiality of secret keys
 - Compromise of authenticity of secret or public keys.
 - Unauthorized use of public or secret keys
- Security policy explicitly or implicitly defines the threats a system is intended to address. Security policy may affect the stringency of cryptographic requirements, depending on the susceptibility of the environment in questions to various types of attack.
- Threats 1. compromise of confidentiality of secret keys 2. compromise of authenticity of secret or public keys. 3. unauthorized use of public or secret keys

KEY MANAGEMENT TECHNIQUES

Public-key techniques

Primary advantages offered by public-key techniques for applications related to key management include:

1. simplified key management
2. on-line trusted server not required
3. enhanced functionality

Simplified key management

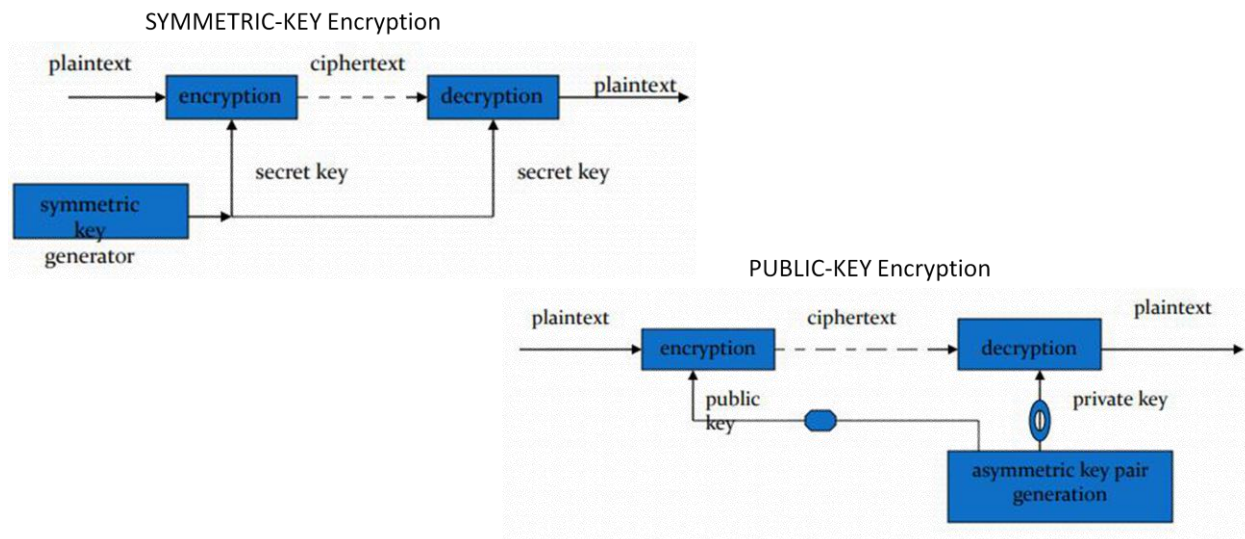


Figure 18: Simplified key management

Symmetric-key certificates: Symmetric-key certificates provide a means for a KTC(Key Translation Center) to avoid the requirement of either maintaining a secure database of user secrets (or duplicating such a database for multiple servers), or retrieving such keys from a database upon translation requests.

KEY MANAGEMENT LIFE CYCLE



Figure 19: Key management life cycle

Key management life cycle

1. user registration
2. user initialization
3. key generation
4. key installation
5. key registration
6. normal use
7. key backup
8. key update
9. archival
10. key de-registration and destruction
11. key recovery
12. key revocation

AUTOMATIC KEY DISTRIBUTION

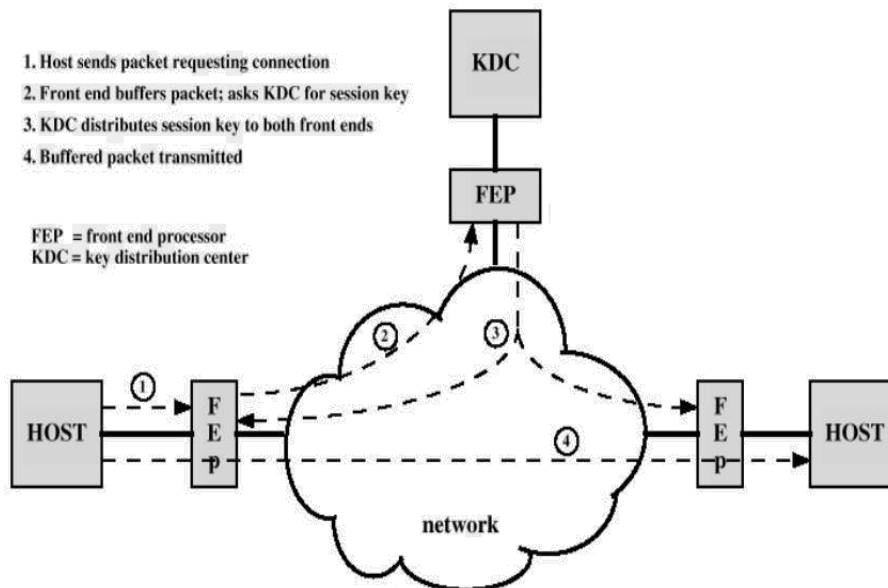


Figure 20: Automatic Key Distribution

Session Key

Used for duration of one logical connection

Destroyed at end of session

Used for user data

Permanent key

Used for distribution of keys

Key distribution center

Determines which systems may communicate

Provides one session key for that connection

Front end processor

Performs end to end encryption

Obtains keys for host

Key Distribution

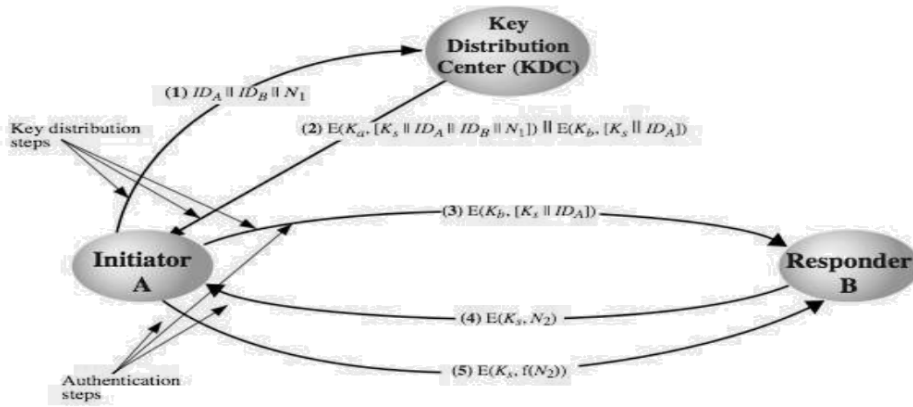


Figure 21: Key Distribution

given parties A and B have various **key distribution** alternatives:

'A' can select key and physically deliver to 'B'

third party can select & deliver key to 'A' & 'B'

if 'A' & 'B' have communicated previously can use previous key to encrypt a new key

if 'A' & 'B' have secure communications with a third party 'C', 'C' can relay key between 'A' & 'B'

Key Distribution Issues

- hierarchies of KDC's required for large networks, but must trust each other
- session key lifetimes should be limited for greater security
- use of automatic key distribution on behalf of users, but must trust system
- use of decentralized key distribution
- controlling key usage

Simple Secret Key Distribution

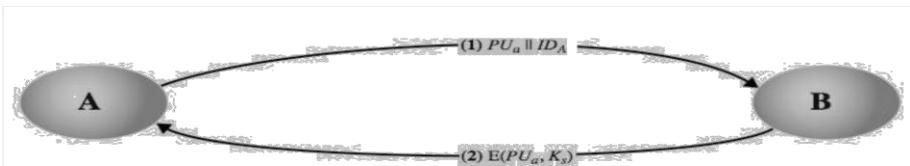


Fig 1. Simple Secret Key Distribution

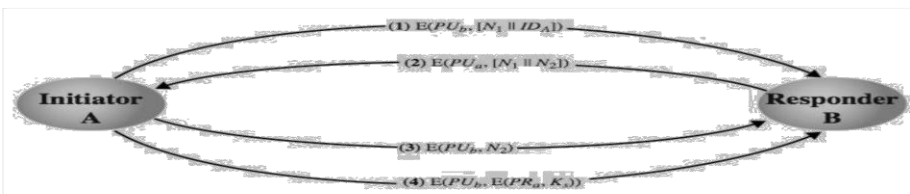


Fig . Secret Key Distribution with Confidentiality and Authentication

Figure 22(1): Simple Secret Key Distribution

Merkle proposed this very simple scheme

- allows secure communications
- no keys before/after exist

Fig 22(2). Secret Key Distribution with Confidentiality and Authentication

Distribution of Public Keys

Distribution of Public Keys can be considered as using one of:

- public announcement
- publicly available directory
- public-key authority
- public-key certificates

Public Announcement

- *Users distribute public keys to recipients or broadcast to community at large
 - eg. append PGP keys to email messages or post to news groups or email list
- * major weakness is forgery
 - anyone can create a key claiming to be someone else and broadcast it
- * until forgery is discovered can masquerade as claimed user

Publicly Available Directory

- * can obtain greater security by registering keys with a public directory
- * directory must be trusted with properties:
 - contains {name,public-key} entries
 - participants register securely with directory
 - participants can replace key at any time
 - directory is periodically published
 - directory can be accessed electronically
- * still vulnerable to tampering or forgery

Public-Key Authority

- improve security by tightening control over distribution of keys from directory has properties of directory and requires users to know public key for the directory then users interact with directory to obtain any desired public key securely does require real-time access to directory when keys are needed
 - may be vulnerable to tampering

Public-Key Certificates

- Øcertificates allow key exchange without real-time access to public-key authority
- Øa certificate binds **identity** to **public key**
- Øusually with other info such as period of validity, rights of use etc
- Øwith all contents **signed** by a trusted Public-Key or Certificate Authority (CA)
- Øcan be verified by anyone who knows the public-key authorities public-key

Layer Attack and Countermeasure

Network Layer	Security Attack	Countermeasure
Physical	Jamming Tampering	Spread-spectrum, frequency hopping Tamper-proof design
Link	Collisions Exhaustion Unfairness	Error-correcting codes Data rate limits, time division multiplexing Short frames
Network and Routing	Spoofing, altering, replaying Sinkholes Wormholes Sybil Selective forwarding HELLO attack Acknowledge spoofing	Authentication, link-layer encryption Authentication, link-layer encryption Authentication, geographic routing, tight synchronisation Authentication, public key cryptography Authentication, link-layer encryption, multipath routing Authentication, bidirectional link and identity verification Authentication
Transport	Flooding Desynchronization	Client puzzles, authenticated broadcast Authentication
Application	Stimuli attack Packet injection	Authentication Authentication

Sensor Node Hardware – Berkeley Motes, Programming Challenges, Node-level software platforms – TinyOS, nesC, CONTIKIOS, Node-level Simulators – NS2 and its extension to sensor networks, COOJA, TOSSIM, Programming beyond individual nodes – State centric programming.

Sensor Node Hardware

Sensor node hardware can be grouped into three categories, each of which entails a different set of trade-offs in the design choices.

1. Augmented general-purpose computers: Examples include low power PCs, embedded PCs (e.g., PC104), custom-designed PCs (e.g., Sensoria WINS NG nodes),¹ and various personal digital assistants (PDA). These nodes typically run off-the-shelf operating systems such as Win CE, Linux, or real-time operating systems and use standard wireless communication protocols such as Bluetooth or IEEE 802.11.

Because of their relatively higher processing capability, they can accommodate a wide variety of sensors, ranging from simple microphones to more sophisticated video cameras. Compared with dedicated sensor nodes, PC-like platforms are more power hungry. However, when power is not an issue, these platforms have the advantage that they can leverage the availability of fully supported networking protocols, popular programming languages, middleware, and other off-the-shelf software

2. Dedicated embedded sensor nodes: These platforms typically use commercial off-the-shelf (COTS) chip sets with emphasis on small form factor, low power processing and communication, and simple sensor interfaces. Because of their COTS CPU, these platforms typically support at least one programming language, such as C.

However, in order to keep the program footprint small to accommodate their small memory size, programmers of these platforms are given full access to hardware but barely any operating system support. A classical example is the Tiny OS platform and its companion programming language, nesC.

System-on-chip (SoC) nodes: Examples of SoC hardware include smart dust, the BWRC picoradionode [187], and the PASTA node.³ Designers of these platforms try to push the hardware limits by fundamentally rethinking the hardware architecture trade-offs for a sensor node at the chip design level. The goal is to find new ways of integrating CMOS, MEMS, and RF technologies to build extremely low power and small footprint sensor nodes that still provide certain sensing, computation, and communication capabilities. Since most of these platforms are currently in the research pipeline with no predefined instruction set, there is no software platform support available.

Examples of Sensor Node

Berkeley Motes(Mica motes)

What is Berkeley Motes in WSN?

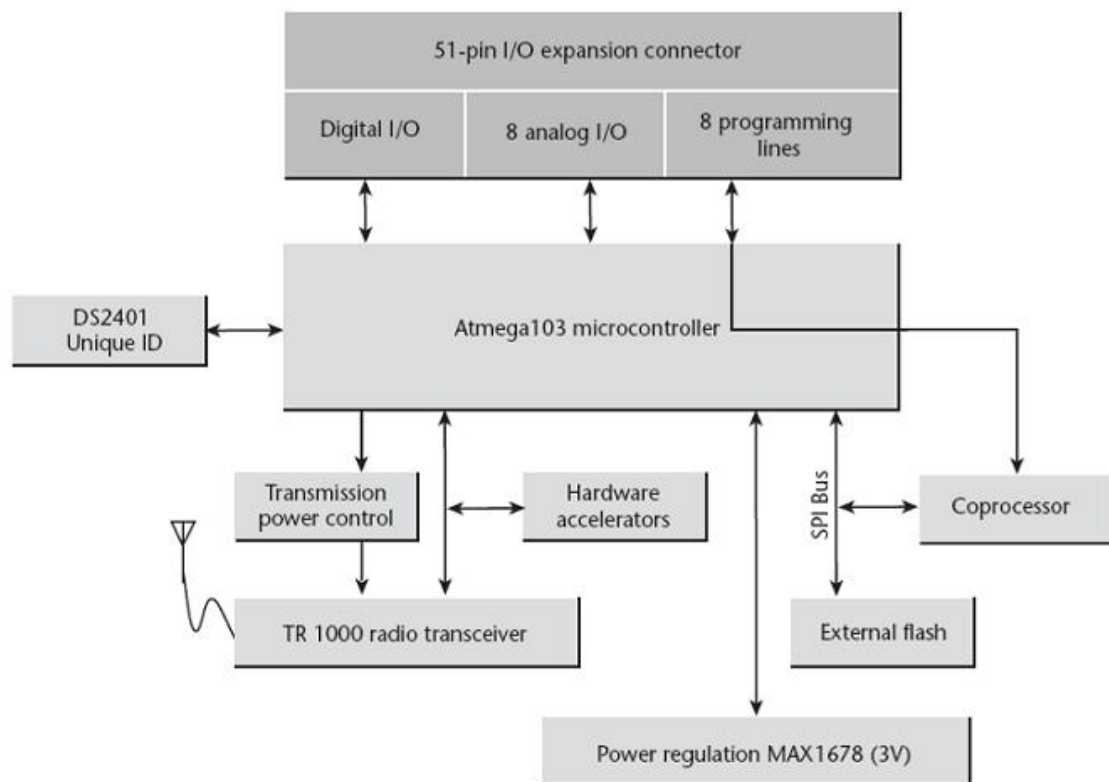
The Berkeley Motes are a **family of embedded sensor nodes sharing roughly the same architecture**. The MICA motes have a two-CPU design. The main microcontroller (MCU), an Atmel ATmega103L, takes care of regular processing. A separate and much less capable coprocessor is only active when the MCU is being reprogrammed.

1. The main microcontroller (MCU), an Atmel ATmega103L, takes care of regular processing. A separate and much less capable coprocessor is only active when the MCU is being reprogrammed. The ATmega103L MCU has integrated 512 KB flash memory and 4 KB of data memory.

2. Given these small memory sizes, writing software for motes is challenging. Ideally, programmers should be relieved from optimizing code at assembly level to keep code footprint small.

3. However, high-level support and software services are not free. Being able to mix and match only necessary software components to support a particular application is essential to achieving a small footprint.

MICA MOTE ARCHITECTURE



The memory inside the MCU, a MICA mote also has a separate 512 KB flash memory unit that can hold data. Since the connection between the MCU and this external memory is via a low-speed serial peripheral interface (SPI) protocol, the external memory is more suited for storing data for later batch processing than for storing programs.

The RF communication on MICA motes uses the TR1000 chip set (from RF Monolithics, Inc.) operating at 916 MHz band. With hardware accelerators, it can achieve a maximum of 50 kbps raw data rate. MICA motes implement a 40 kbps transmission rate.

The transmission power can be digitally adjusted by software through a potentiometer (Maxim DS1804). The maximum transmission range is about 300 feet in open space.

MICA motes support a 51 pin I/O extension connector. Sensors, actuators, serial I/O boards, or parallel I/O boards can be connected via the connector.

A sensor/ actuator board can host a temperature sensor, a light sensor, an accelerometer, a magnetometer, a microphone, and a beeper.

The serial I/O (UART) connection allows the mote to communicate with a PC in real time. The parallel connection is primarily for downloading programs to the mote.

POWER CONSUMPTION OF MICA MOTES

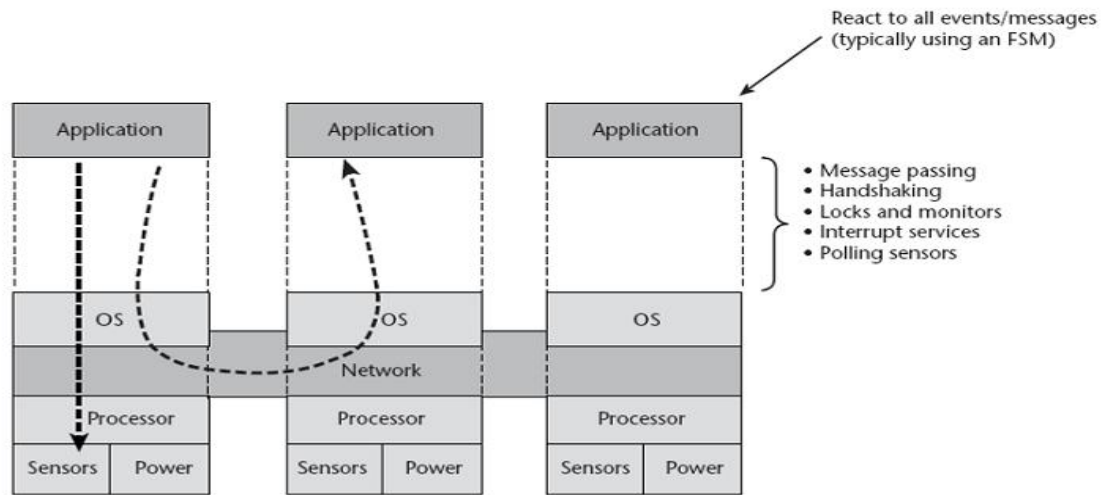
It is interesting to look at the energy consumption of various components on a MICA mote. A radio transmission bears the maximum power consumption. However, each radio packet (e.g., 30 bytes) only takes 4 ms to send, while listening to incoming packets turns the radio receiver on all the time.

The energy that can send one packet only supports the radio receiver for about 27 ms. Another observation is that there are huge differences among the power consumption levels in the active mode, the idle mode, and the suspend mode of the MCU.

It is thus worthwhile from an energy-saving point of view to suspend the MCU and the RF receiver as long as possible.

SENSOR NETWORK PROGRAMMING CHALLENGES

Traditional embedded system programming interface



Sensor networks, the application programmers need to explicitly deal with message passing, event synchronization, interrupt handing, and sensor reading.

As a result, an application is typically implemented as a finite state machine (FSM) that covers all extreme cases: unreliable communication channels, long delays, irregular arrival of messages, simultaneous events, and so on.

In a target tracking application implemented on a Linux operating system and with directed diffusion routing, roughly 40 percent of the code implements the FSM and the glue logic of interfacing computation and communication

For resource-constrained embedded systems with real-time requirements, several mechanisms are used in embedded operating systems to reduce code size, improve response time, and reduce energy consumption.

Microkernel technologies modularize the operating system so that only the necessary parts are deployed with the application.

Real-time scheduling allocates resources to more urgent tasks so that they can be finished early.

Event-driven execution allows the system to fall into low-power sleep mode when no interesting events need to be processed.

At the extreme, embedded operating systems tend to expose more hardware controls to the programmers, who now have to directly face device drivers and scheduling algorithms, and optimize code at the assembly level.

Although these techniques may work well for small, stand-alone embedded systems, they do not scale up for the programming of sensor networks for two reasons.

Sensor networks are large-scale distributed systems, where global properties are derivable from program execution in a massive number of distributed nodes.

Distributed algorithms themselves are hard to implement, especially when infrastructure support is limited due to the ad hoc formation of the system and constrained power, memory, and bandwidth resources.

As sensor nodes deeply embed into the physical world, a sensor network should be able to respond to multiple concurrent stimuli at the speed of changes of the physical phenomena of interest

NODE-LEVEL SOFTWARE PLATFORMS

A node-level platform can be a node centric operating system, which provides hardware and networking abstractions of a sensor node to programmers, or it can be a language platform, which provides a library of components to programmers.

A typical operating system abstracts the hardware platform by providing a set of services for applications, including file management, memory allocation, task scheduling, peripheral device drivers, and networking.

Tiny OS and Tiny GALS are two representative examples of node-level programming tools

TinyOS – An Operating System for Sensor Networks

Introduction (TinyOS)

What is TinyOS?

TinyOS is an embedded, component-based operating system and platform for low-power wireless devices, such as those used in wireless sensor networks, smartdust, ubiquitous computing, personal area networks, building automation, and smart meters.

TinyOS has a component-based programming model, codified by the NesC language, a dialect of C.

It is a programming framework for embedded systems

Building an application-specific OS into each application.

About 15K in size, of which the base OS is about 400 bytes; the largest application.

A database-like query system

Commands, Events and Task (TinyOS)

A TinyOS program is a graph of components, each of which is an independent computational entity that exposes one or more interfaces.

Components have three computational abstractions: *commands*, *events*, and *tasks*.

Commands and events are mechanisms for inter-component communication, while tasks are used to express intra-component concurrency.

Command – Request

Events – Services

Task - Rather than performing a computation immediately, commands and event handlers may post a task.

Interfaces

These interfaces define how the component directly interacts with other components.

An interface generally models some service (e.g., sending a message) and is specified by an interface type.

Interfaces are bidirectional and contain both *commands* and *events*

Collections of related functions

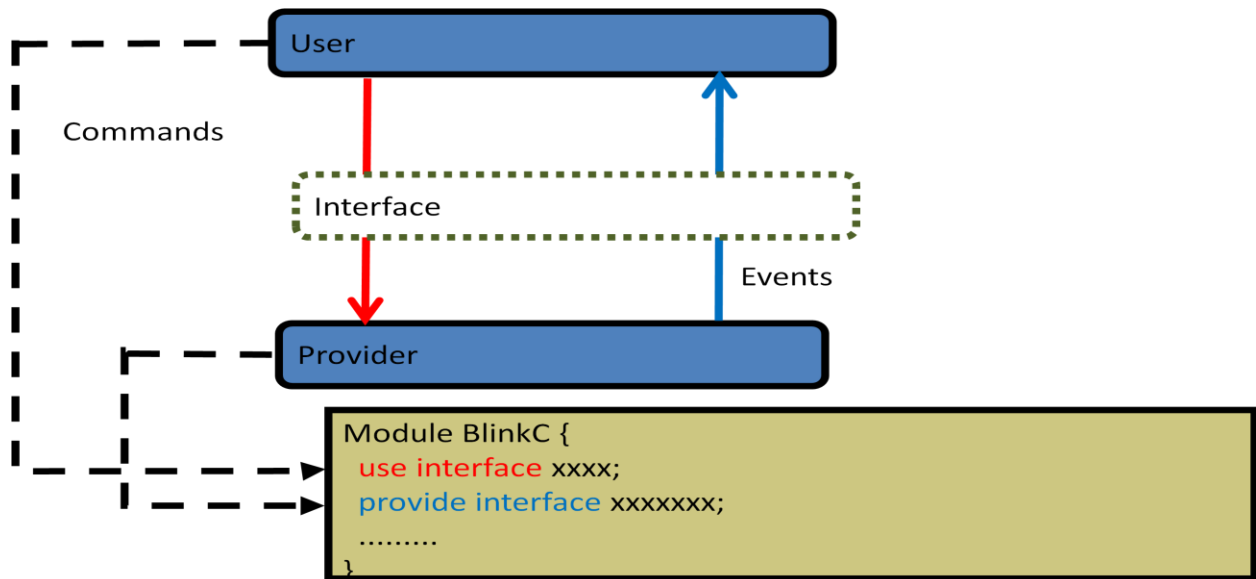
- Define how components connect
- Interfaces are bi-directional: for A->B
 - Commands are from A to B
 - Events are from B to A

Can have parameters (types

```
interface Timer<tag> {  
  command void startOneShot(uint32_t period);  
  command void startPeriodic(uint32_t period);  
  event void fired();  
}
```

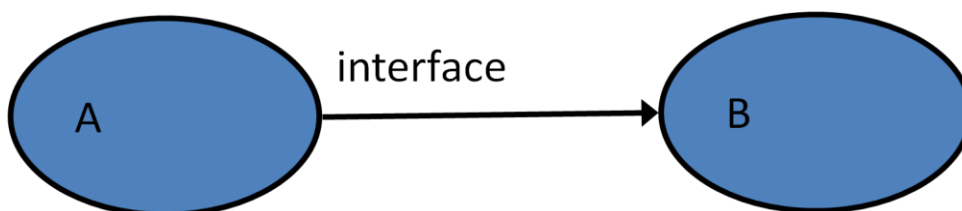

Interface (provide and use)

If a component calls a command, it uses the interface and the component which implements this command is the provider of this interface. If a component calls an event function, it is the provider of the event, and the component which implements this event function is the user of the event. Remember that the command and event functions are dual.



TinyOS Components

- TinyOS and its applications are in nesC dialect with extra features
- Basic unit of nesC code is a component
- Components connect via interfaces Connections called "wiring"
- Components
- A component is a file (names must match)
- Modules are components that have variables and executable code
- Configurations are components that wire other components together



Tasks

TinyOS has a single stack: long-running computation can reduce responsiveness

Tasks: mechanism to defer computation

Tells TinyOS “do this later”

Tasks run to completion

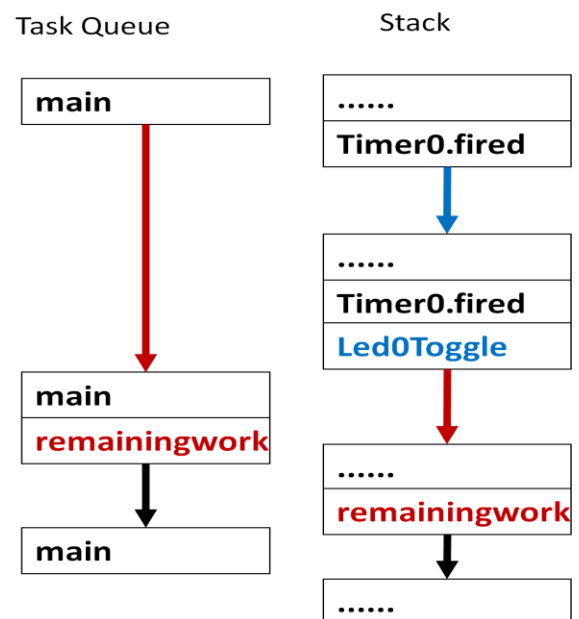
TinyOS scheduler runs them one by one in the order they post

Keep them short!

Interrupts run on stack, can post tasks

TinyOS Execution Model

```
XXXXXX;  
event void Timer0.fired()  
{  
    XXXXXX;  
    XXXXXX;  
    XXXXXX;  
    XXXXXX;  
    call Leds.led0Toggle();  
    XXXXXX;  
    XXXXXX;  
    post remainingwork();  
}  
XXXXXX;  
remainingwork(){xxxx;};  
XXXXXX;
```



Let's look at an example, At the left side is a code snippet. The timer event function calls the LED function in the middle and post a task at the end. At the beginning, the task queue only contains the main task, which is the scheduler. While the timer event function is called, this function is put into the stack and the execution of this function starts.

When the LED function is called, it is pushed into the stack, but the task queue is unchanged since both the timer function and the LED function are within the same task. At the end of the timer function, it posts a task remainingwork. The task queue is appended with a this newly posted task.

When the LED and Timer function finish, both of them are popped up. And the current task execution is finished, then the scheduler picks the next task, i.e., the remainingwork task, to execute. The remaining task is removed from the task queue and it is pushed into the stack for execution.

Applications

- TinyOS platform: an environmental monitoring system, a declarative query processor, and magnetometer-based object tracking.
- Sensor networks enable data collection at a scale and resolution that was previously unattainable, opening up many new areas of study for scientists.
- These applications pose many challenges, including
 - low-power operation
 - Robustness
 - due to remote placement
 - extended operation.

Tiny OS application example—Field Monitor, where all nodes in a sensor field periodically send their temperature and photo sensor readings to a base station via an ad hoc routing mechanism

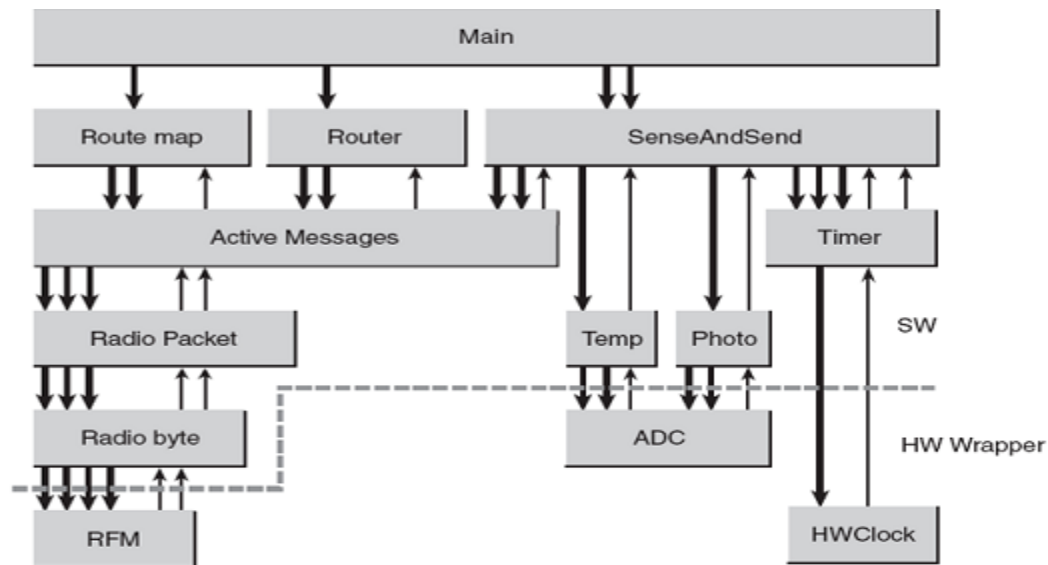
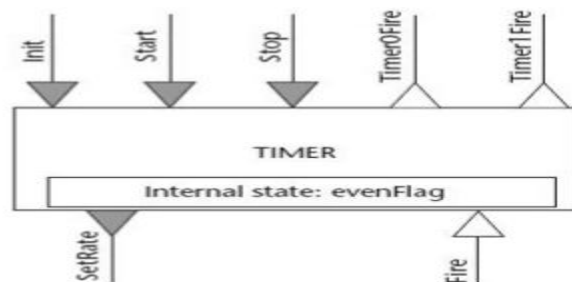


Figure 7.5 The FieldMonitor application for sensing and sending measurements.

Blocks represent Tiny OS components and arrows represent function calls among them. The directions of the arrows are from callers to callees



The Timer component and its interfaces.

This component is designed to work with a clock, which is a software wrapper around a hardware clock that generates periodic interrupts. The method calls of the Timer component are shown in the figure as the arrowheads.

An arrowhead pointing into the component is a method of the component that other components can call.

An arrowhead pointing outward is a method that this component requires another layer component to provide.

The absolute directions of the arrows, up or down, illustrate this component's relationship with other layers.

For example, the Timer depends on a lower layer HWClock component. The Timer can set the rate of the clock, and in response to each clock interrupt it toggles an internal Boolean flag, `evenFlag`, between true (or 1) and false (or 0). If the flag is 0, the Timer produces a `timer0Fire` event to trigger other components; otherwise, it produces a `timer1Fire` event. The Timer has an `init()`

method that initializes its internal flag, and it can be enabled and disabled via the `start` and `stop` calls. A program executed in Tiny OS has two contexts, tasks and events, which provide two sources of concurrency. Tasks are created (also called posted) by components to a task scheduler.

The default implementation of the Tiny OS scheduler maintains a task queue and invokes tasks according to the order in which they were posted. Thus tasks are deferred computation mechanisms. Tasks always run to completion without preempting or being preempted by other tasks. Thus tasks are nonpreemptive. The scheduler invokes a new task from the task queue only when the current task has completed. When no tasks are available in the task queue, the scheduler puts the CPU into the sleep mode to save energy.

The ultimate sources of triggered execution are events from hardware: clock, digital inputs, or other kinds of interrupts. The execution of an interrupt handler is called an event context.

ÇA split-phase execution, sending a packet will block the entire system from reacting to new events for a significant period of time. In the Tiny OS implementation, the `send()` command in the AM component returns immediately. When the packet is indeed sent, the AM component will notify its caller by a `sendDone()` method call.

nesC

nesC is an extension of C to support and reflect the design of Tiny OS v1.0 and Above. provides a set of language constructs and restrictions to implement Tiny OS components and applications. used to build applications for the TinyOS platform.

NesC has two types of components: modules and configurations. Modules provide code and are written in a dialect of C with extensions for calling and implementing commands and events.

A module declares private state variables and data buffers, which only it can reference.

Configurations are used to wire other components together, connecting interfaces used by components to interfaces provided by others.

Contiki-OS

Contiki is an:

- open source
- highly portable
- multi-tasking operating system

For:

- Memory efficient
- Networked embedded systems

And wireless sensor networks

Contiki is an operating system for networked, memory-constrained systems with a focus on low-power wireless Internet of Things devices.

Contiki OS is an open-source operating system which is Linux based and developed for Internet of Things (IoT) devices.

It has also powerful tools for building complicated wireless communication systems. Contiki OS has been especially developed for low-powered WSN apps.

In other words, it has been developed for WSN applications that are able to work with AA type batteries for years.

Where Contiki Used?

Contiki has been used in a variety of projects from

- road tunnel fire monitoring,
- intrusion detection, water monitoring to surveillance networks

Contiki features

- TCP/IP communication with uIP stack
- Loadable modules
- Event-driven kernel
- Protothreads
- Protocol-independent radio network with the Rime stack
- Cross-layer network simulation with Cooja
- Networked shell
- Memory efficient flash-based Coffee file system
- Software-based power profiling
- Coffee file system :
 - Coffee is a very simple, relatively small and easy to use file system that you are most likely going to be very familiar with if you have done any C file access in the past. The notion is the same as on a normal PC: you open a file, read and write to it and close it. Contiki will take care of the underlying flash memory, giving you more time to focus on the real issues.
- Software-based power profiling
- Power consumption is the most important metric in wireless sensor networks because reduced power consumption leads to increased network lifetime. Energy has been reduced by using more efficient protocols for topology management, routing, and radio medium access.

CONTIKI-OS

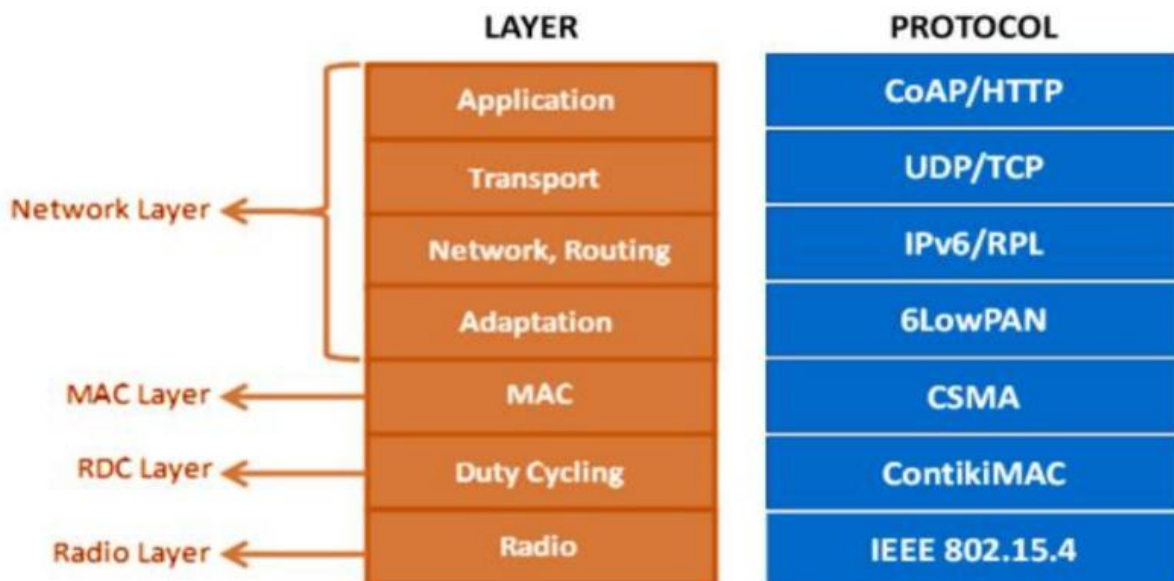
A standard Contiki configuration for a microcontroller is 2kB RAM and 40 kB ROM. Besides that, Contiki can provide communication over IPv4, IPv6 and Rime Network Stack.

Contiki directory in OS, also provides access to system source codes, sample application codes, practical applications, and driver codes for many node types, specific microcontroller files and important tools like Cooja.

Contiki provides the opportunity for developers to use existing samples directly or modify them. With these features, researchers and developers would have effective development environment

CONTIKI OPERATING SYSTEM AND ITS STRUCTURE

Contiki OS, which is C programming language based and open source, has been developed for lightweight, flexible and low-powered wireless sensor networks



Conclusion

Contiki OS which is a lightweight, open source operating system developed for WSN application.

It has powerful tools for building complicated wireless communication systems.

Rime Network Stack is very important as it presents a lightweight network stack which is very convenient for low-powered WSN's.

Protothread mechanism is one of important factors which makes difference. Likewise, flexible structure of Contiki OS and ability to use in many WSN platforms like cc2538, skymote, MicaZ, Zolertia Z1 etc. increase its preferability

Comparison Criterion	Contiki OS 3.0	TinyOS 2.0
Source model	Open source	Open source
System(Dynamic/Static)	Dynamic	Static
System(Monolithic/Modular)	Modular	Monolithic
Networking support	IPv4, IPv6, Rime	Active message
Programming language	C	NesC
Multithreading support	Yes (have Protothread mechanism)	Partial (through Tiny Threads)
Simulator	Cooja, MSPSim, Netsim	TOSSIM, Power Tossim

Simulators

Ns2

Toosim.

What is Ns2?

- Ns2 l is a discrete event simulator also for networking research and also work at packet level research. It also also simulates both wired and also wireless network.
- Ns2 was Developed by UC Berkeley and also is currently maintained by usc.

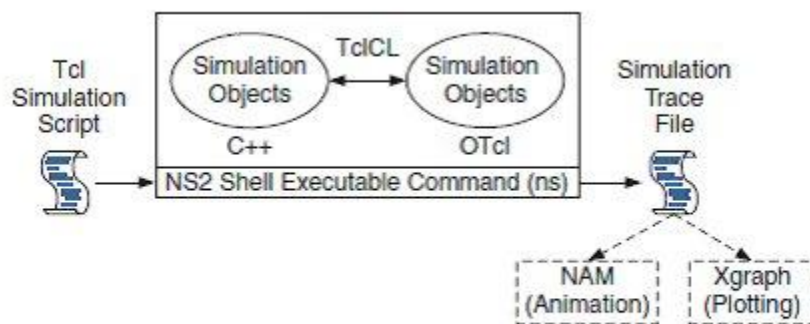
Components of ns-2:

- Nam.
- Ns2.

- Post processing like simple trace analysis, often also in AWK, perl or OTCL.
- Pre-processing like traffic and also topology generators.
- **Features of NS2**
- 1. It is a discrete event simulator for networking research.
- 2. It provides substantial support to simulate bunch of protocols like TCP, FTP, UDP, https and DSR.
- 3. It simulates wired and wireless network.
- 4. It is primarily Unix based.
- 5. Uses TCL as its scripting language.
- 6. Otcl: Object oriented support
- 7. Tclcl: C++ and otcl linkage
- 8. Discrete event scheduler

Basic Architecture

NS2 consists of two key languages: C++ and Object-oriented Tool Command Language (OTcl). While the C++ defines the internal mechanism (i.e., a backend) of the simulation objects, the OTcl sets up simulation by assembling and configuring the objects as well as scheduling discrete events. The C++ and the OTcl are linked together using TclCL



Basic architecture of NS.

“data” / control separation : C++ for “data”: per packet processing, core of ns fast to run, detailed, complete control

OTcl for control: Simulation scenario configurations, Periodic or triggered action, Manipulating existing C++ objects q fast to write and change

FUNCTIONALITY OF NS2

The main functionality of NS-2 is implemented in C++, while the dynamics of the simulation (e.g., time-dependent application characteristics) is controlled by TCL(Tool Command Language) scripts.

Procedures to create simulating script:

- Create the event scheduler.
- Turn on tracing.
- Create network.
- Setup routing.
- Insert errors.
- Create transport connection.
- Create traffic.
- Transmit also in application-level data.

Advantages of simulation:

- Cheaper.
- Can simulate detail also at arbitrary level.
- Generality.
- Find bugs also in advance.

Sample code for how to set node configuration in ns-2:

- `set val(chan) Channel/WirelessChannel`
- `set val(prop) Propagation/TwoRayGround`
- `set val(netif) Phy/WirelessPhy`
- `set val(mac) Mac/802_11`
- `set val(ifq) Queue/DropTail/PriQueue`
- `set val(ll) LL`
- `set val(ant) Antenna/OmniAntenna`
- `set val(x) 1300`
- `set val(y) 1300`
- `set val(ifqlen) 201`
- `set val(seed) 0.0`
- `set val(routingprotocol) AMMNET`
- `set val(nn) 50`
- `set val(nnn) 49`
- `set val(cp) cbr`
- `set val(sc) scen`
- `set val(stop) 27.3`
- `set val(stop1) 16.0`
- `set val(traffic) tcp`
- `set val(rxPower) 0.1`
- `set val(txPower) 0.1`
- `set val(energymodel) EnergyModel`
- `set val(initialenergy) 1000`
- `set val(sleeppower) 0.00005`
- `set val(ener) 500`

- `set val(energy) cal`
- `set val(freq) node`

TOSSIM

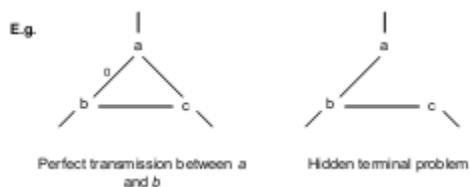
TOSSIM is a **discrete event simulator for TinyOS sensor networks**. Instead of compiling a TinyOS application for a mote, users can compile it into the TOSSIM framework, which runs on a PC. This allows users to debug, test, and analyze algorithms in a controlled and repeatable environment

TOSSIM - Overview

- Like TinyOS applications, TOSSIM is component based and event-driven
- Low-level components may be reimplemented to capture mote behavior at a fine grain
- Simulation is based on a virtual clock

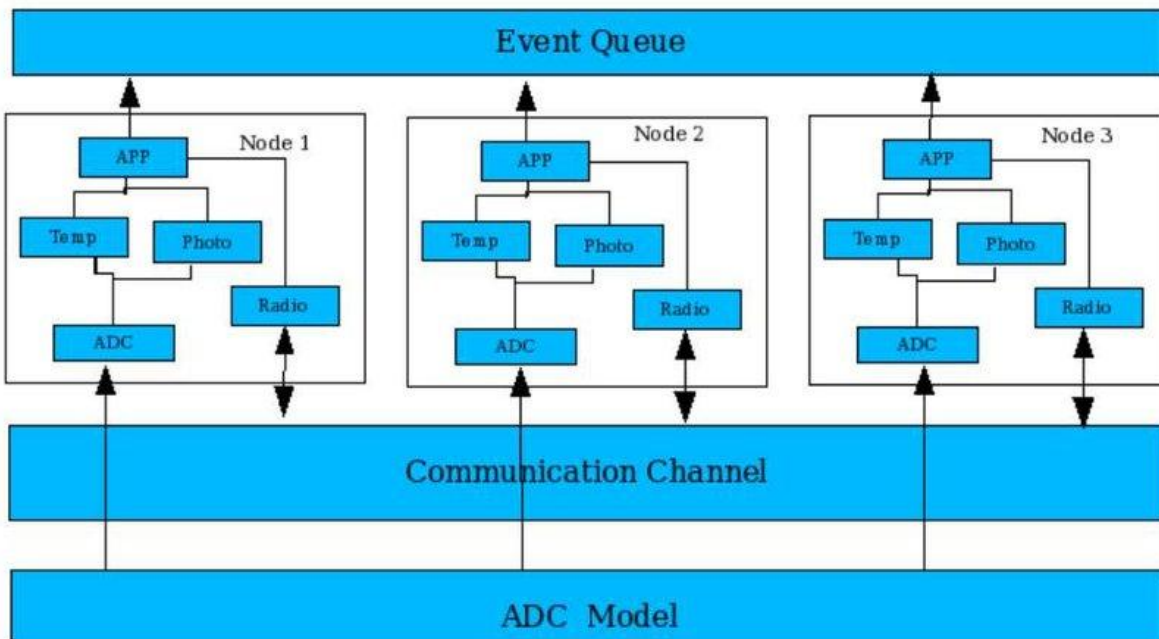
Uses a directed graph to represent a network

- Each vertex represents a node
- Each edge has a bit error probability
- Abstraction can capture most conditions



TOSSIM – Features

- Single compiler support for simulation and hardware
 - Added a compiler option to the nesC Compiler to compile an application for simulation instead of mote hardware
- Visualization tool (TinyViz)
 - Allows simulations to be visualized, controlled and analyzed through a GUI
 - Allows for plugins for users to interact with simulation



The architecture of TOSSIM. The block diagram represents the various components of TinyOS and the network simulated by TOSSIM.

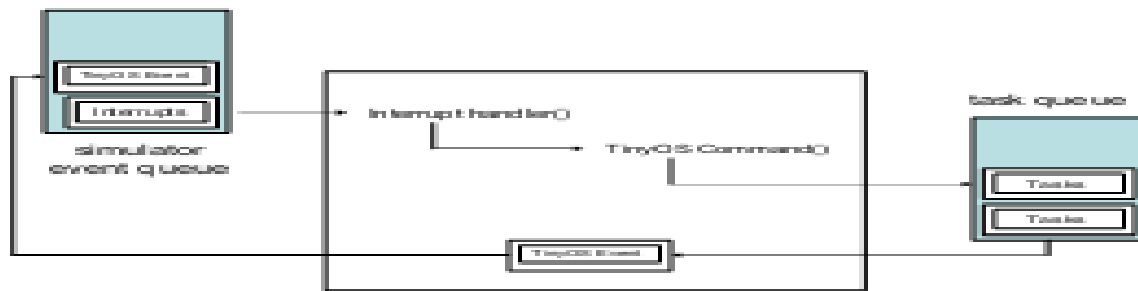
The architecture of TOSSIM Parts

- Compiling component graphs into simulation infrastructure
- Discrete event queue
- Hardware abstraction of component
- Radio and ADC models
- Communication services

TOSSIM – Event Simulation

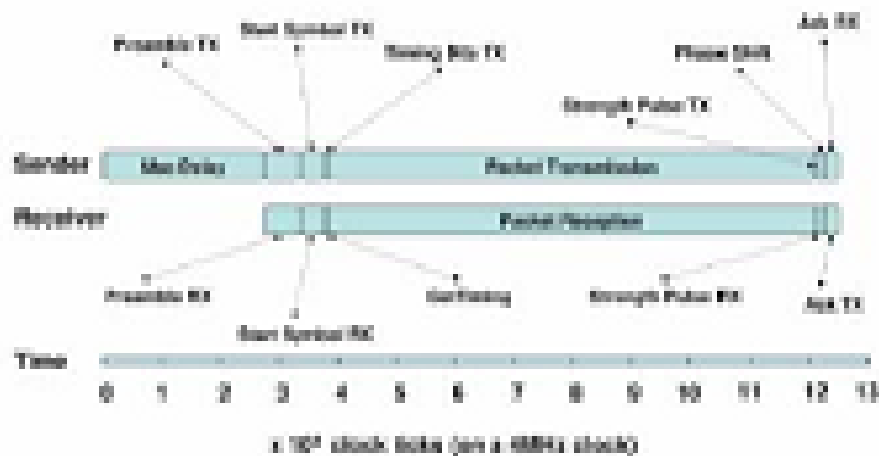
- Simulator event cycle

- Simulator event cycle



TOSSIM – Network Simulation

- Receiver radio clock need to be adjusted to synchronize with sender signal
- Adjustments to radio bit rates are made by changing the period between radio clock events
- Network stack is simulated at bit level



Cooja is an emulator

Emulator is a hardware or software system that enables one computer system (called the host) to behave like another computer system (called the guest):

- ✓ Eg: Cooja enabling your laptop to behave like a Z1 mote.

A system that typically enables the host system to run software or use peripheral devices designed for the guest system:

Eg. Cooja enabling your laptop to run the RPL protocol, LIBP and/or other IoT protocols of interest

What is Cooja?

Cooja is a Contiki network emulator

- An extensible Java-based simulator capable of emulating Tmote Sky (and other) nodes
- The code to be executed by the node is the exact same firmware you may upload to physical nodes
- Allows large and small networks of motes to be simulated Motes can be emulated at the hardware level
- Slower but allows for precise inspection of system behaviour Motes can also be emulated at a less detailed level Faster and allows simulation of larger networks
- Cooja is a highly useful tool for Contiki development .
- It allows developers to test their code and systems long before running it on the target hardware .
- Developers regularly set up new simulations to
- debug their software
- to verify the behaviour of their systems

Main steps

Open a terminal window to start Cooja

2. Create a new simulation to run Contiki in simulation and wait for Cooja to start and compile itself

3. Set simulation options

4. Create a new mote type

5. Add motes to the simulation

6. Open a terminal Cooja is a highly useful tool for Contiki development

It allows developers to test their code and systems long before

(a) running it on the target hardware

ii. Developers regularly set up new simulations to

(a) debug their software