Notes from Airborne Networking side session:

Special considerations for an Airborne Network:

-- Limited space for equipment on aircraft

-- Aircraft power frequently not as "clean" as ground power

-- Challenges of RF networks (data drop-outs common, system must be very fault tolerant)

-- System must be "securable" when parked on the ramp (i.e. aircraft not always parked in secure facilities where everyone has a system-high clearance)

-- Airborne networks have rapidly changing nodes as aircraft transition in and out of the network

-- Network must be self-forming

-- System must have a high level of automation. There will be no network technicians on the plane.

-- Inherently lower data rates than terrestrial networks

-- LPI considerations for transmission

-- Antenna design/placement. Trade-off between punching holes in airframes and providing adequate sky coverage

-- Not likely to be using the latest and greatest hardware due to the time required to design/certify/field airborne systems

-- Open question: where will the gateway be (air, ground)?

Reason MLS is critical for airborne networks:

-- Aircraft platforms present a finite resource platform: physical space and computing resources are extremely limited.

> -- Need to optimize utilization of resources
>
> -- Large volumes of sensitive data transition networks of varying classification levels. Even something as simple as geo-coordinates can be classified. Other examples of sensitive data: imagery, electronic warfare data, weapons characteristics, crypto

-- MLS can enhance data sharing with coalition partners

-- MLS can help speed the accreditation process. In some cases, accreditation is now costing more and taking longer than design. In one stated example, the hardware had to be recapped before accreditation was complete.

Thoughts on MLS/Airborne Networks:

-- MLS to free up space on aircraft by sharing resources. Will require a bottom-to-top re-architecture using MLS principles for maximum benefit.

-- Can't do grey matter fusion on aircraft. System will require a high level of automation.

-- Any solution must be very tolerant to high latency and data drop-outs

-- Need to concentrate on separating infrastructure from application

-- OSI model considerations: physical layer, network layer, application layer, etc.

    -- A transport layer that's not aware of the physical layer will have problems

-- Bandwidth requirements for security at each level must be closely monitored

- Must approach MLS from a system level. There is a lot of community focus on the kernel and not enough about other aspects of the system.

-- Open questions:

    - How are we going to deal with audits? (a special concern because of the limited hardware resources)

    - What will be the platform for MLS: size, cooling requirements?

    - How much can you achieve with legacy systems?