Security Annex Update

November 6, 2018

Dave Gluch

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8702-15-D-0002 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

DM18-1290

Objectives

Get consensus on proceeding with developing a first draft SAE standard formatted document

Discuss

- Authorization
- Action/Command Protection
- Modeling Specialized Architectures
- Vulnerability/Threat Analysis

and other topics as appropriate.

Outline

Security Policies and Requirements

- Documentation
- Verification

Security Protections

- Information/Data Protection
- Access Control and Protection
- Action/Command Protection

Security Architectures

- Specialized Architectures
- Cross Domain Solutions

Analyzing Vulnerabilities/Threats

Example Models and Analyses

E_Enabled Aircraft Models

- Commercial Transport Aircraft
- Mission-Specific Aircraft (reconnaissance)

and ALISA Security Analysis Examples are available at

https://github.com/osate/examples.git

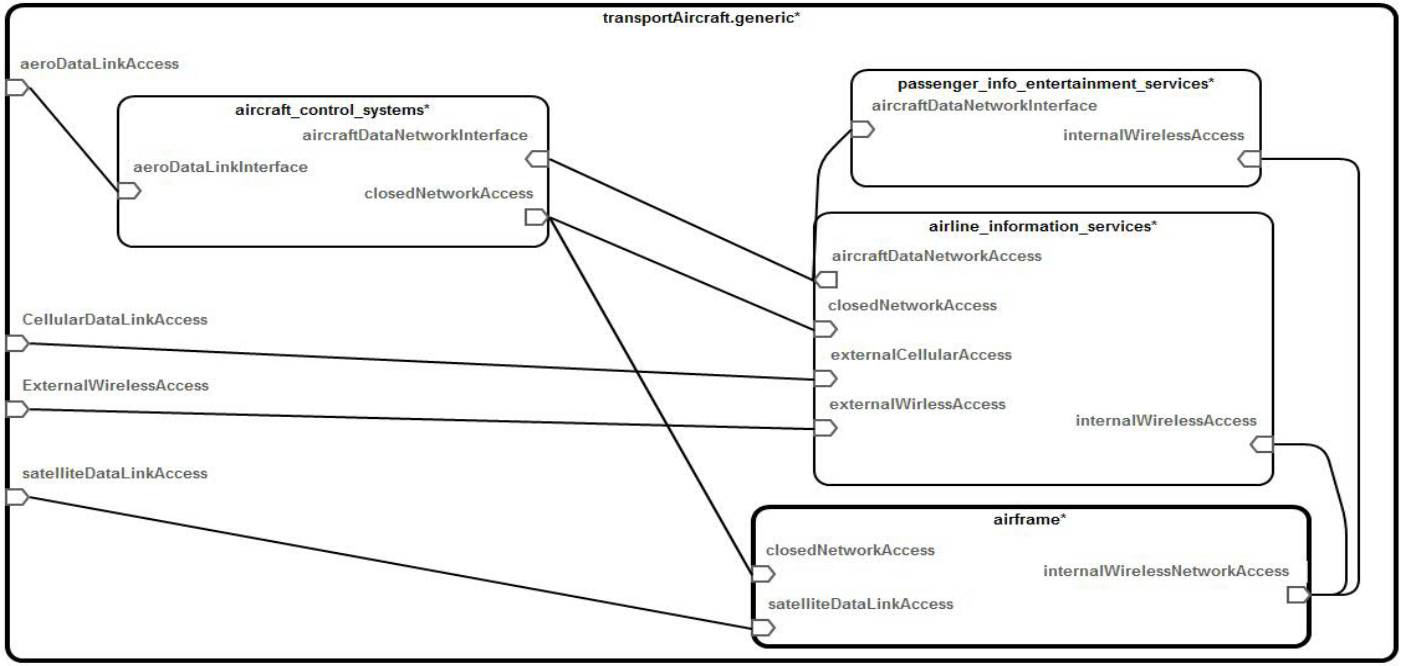
Supporting verification library files are available at

https://github.com/reteprelief/isse

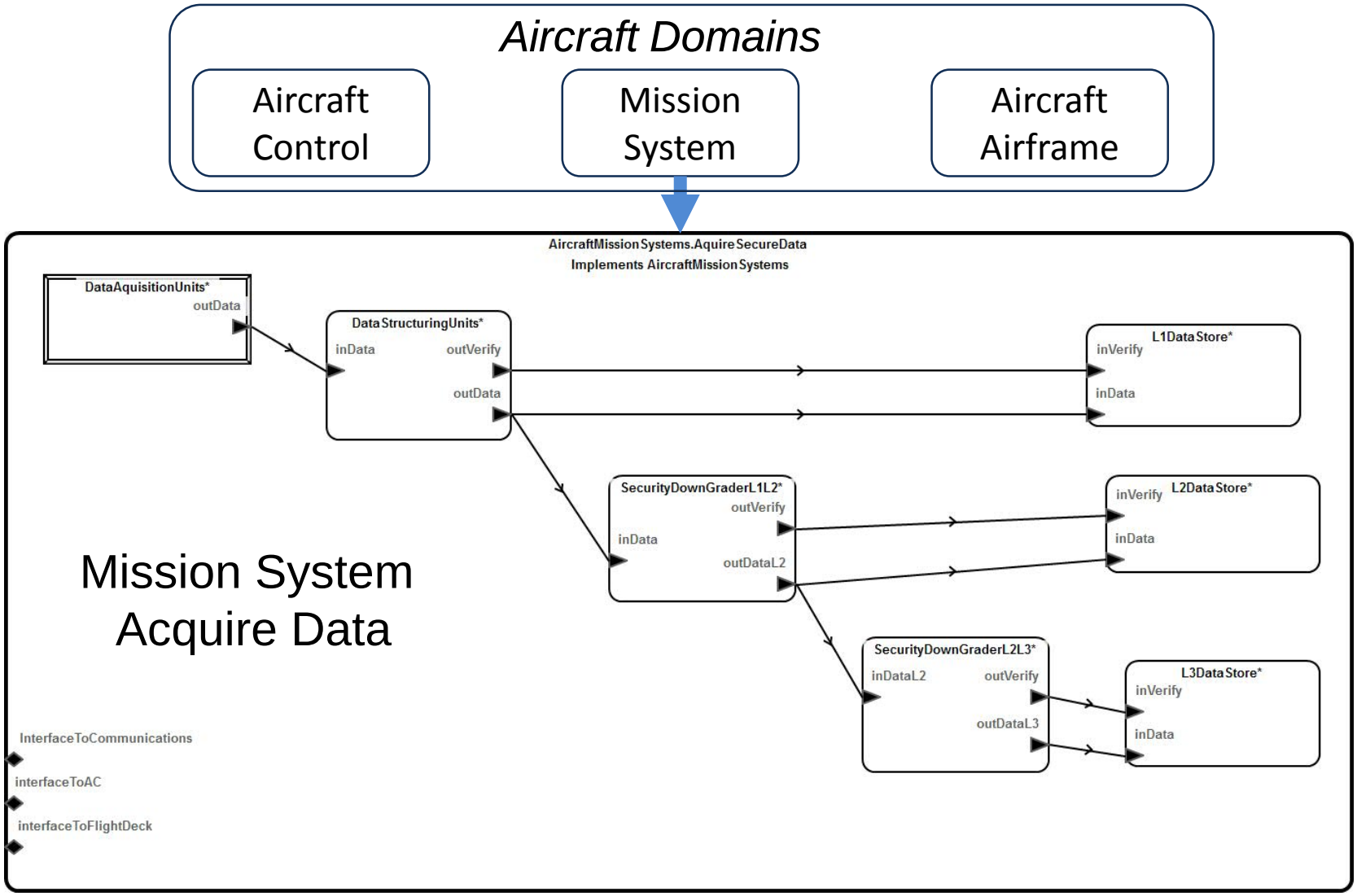
E-Enabled Transport Aircraft

Aircraft Domains

- Aircraft Control (AC)
- Airline Information Services (AIS)
- Passenger Information & Entertainment Services (PIES)



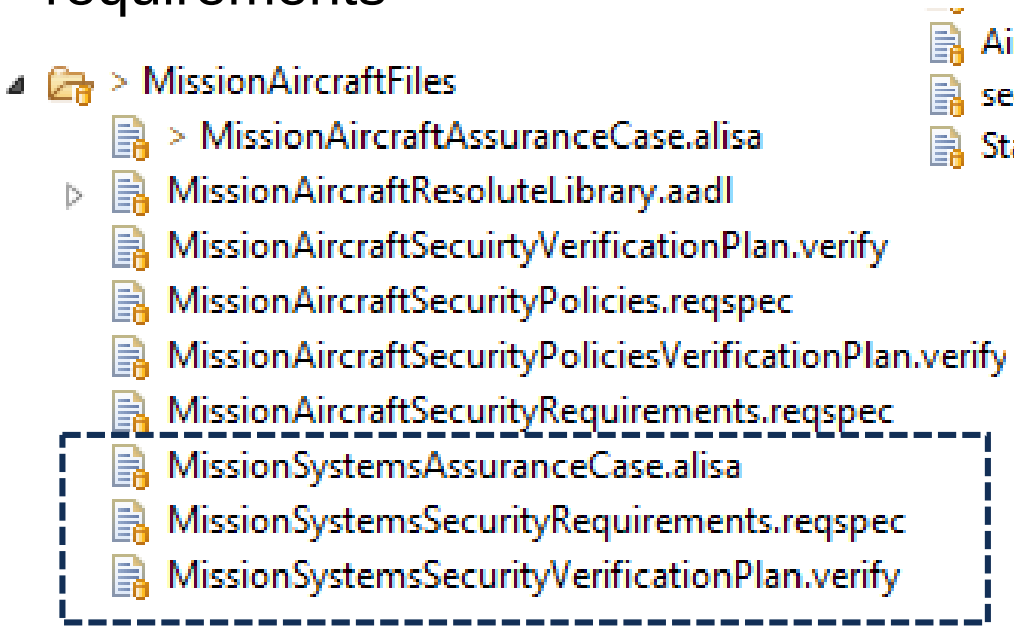
Mission-Specific Aircraft

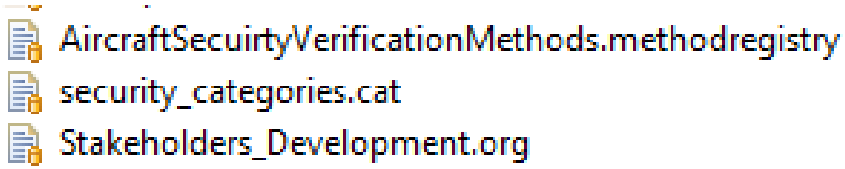


Security Policies and Requirements Overview - 1

Documentation of policies and requirements.

- ReqSpec of ALISA
- Naming convention to distinguish between policies and requirements





Security Policies and Requirements Overview - 2

Policies and Requirements Verification – ALISA

- Verification plans (.verify)
- Assurance cases (.alisa)
- Results (.assure)

MissionSystemsAssuranceCase.alisa
MissionSystemsSecurityRequirements.reqspec
MissionSystemsSecurityVerificationPlan.verify

> assure
> MissionAircraftAssuranceCase.assure
> MissionSystemsAssuranceCase.assure
> TransportAircraftAssuranceCase.assure

Methods

- Resolute
- Java

AircraftSecuirtyVerificationMethods.methodregistry

Security Policy and Requirements Examples

```
requirement SecureMissionDataPolicy: "All mission data must be secured to levels as
defined by DoD security classifications and other mission-specific classifications
of that data."

[

description "All data must be secured. This includes standard DoD classifications and related
security procedures as well as mission-specific classifications and related
procedures."

]

requirement SecureMissionDataClassify: "All mission data must be classified using
DoD or mission-specific classifications."

[

decomposes MissionAircraftSecurityPolicies.AccessControlPolicy
MissionAircraftSecurityPolicies.SecureMissionDataPolicy

]
```

ALISA Verification Plan and Assurance Case

```
claim SecretLevel : "Checks that the information security level is secret"
[
activities
check: AircraftSecurityVerificationMethods.VerifySecret ()
]

claim ExposedEncryption :"check encryption on exposed components"
[
activities
checkencrypt: SecurityVerificationMethods.ExposedConnectionEncrypted()
]
```

```
assurance case MissionSystemsAssuranceCase for
AircraftMissionSystems_pkg::AircraftMissionSystems
[
   assurance plan MissionSystemsSecurityVerificationPlan for
AircraftMissionSystems_pkg::AircraftMissionSystems.AquireSecureData [
assure MissionSystemsSecurityVerificationPlan
assure subsystem DataAquisitionUnits
description "This is the plan for the mission systems including data acquisition,
classification, and downgrading."]

]
```

ALISA Analysis

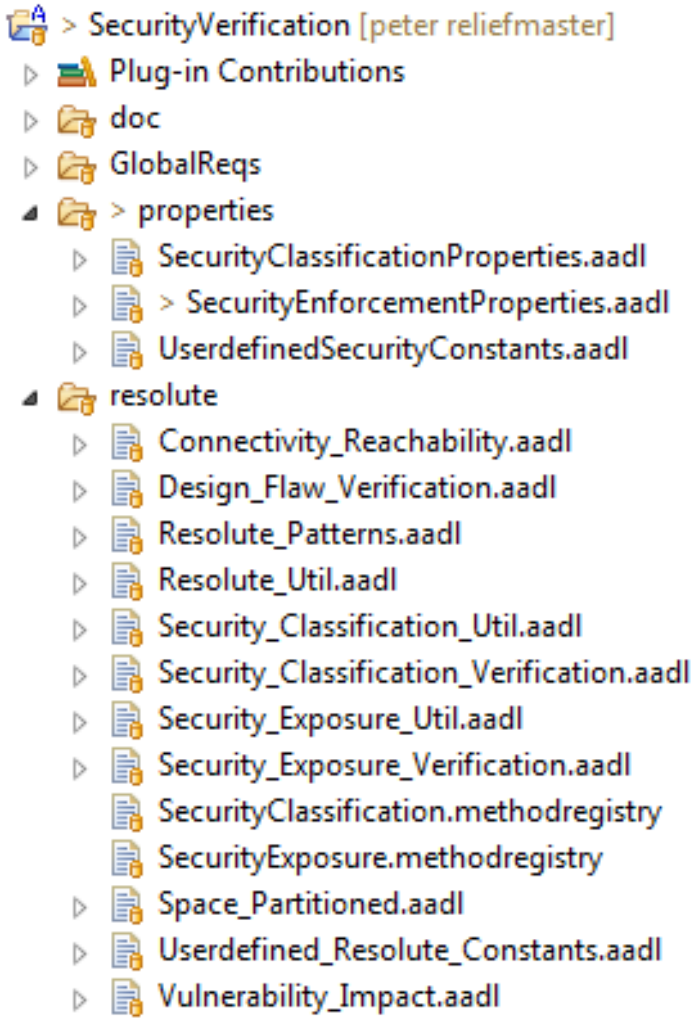
An assurance case output for the Mission System.

Problems | Properties | AADL Property Values | Assurance View

Assurance C...	Filter
TransportAir	<all>
MissionAircr	<all>
MissionSyste	<all>

Evidence	Pass	Fail	Error	Todo	Description
Case MissionSystemsAssuranceCase	2	2			
Plan MissionSystemsSecurityVerificationPlan(AircraftMissionSystems.AquireSecureData)	2	2			
Claim missionDataEncrypt		1			All mission data must be encrypted using AES-256 encryption.
Evidence VerifyEncryption		1			XXX[F]
Claim missionDataHighLevel	1				Raw mission data must be encrypted to the most secure level.
Claim SecretLevel		1			Component must have secret information security level.
Evidence check		1			component AircraftMissionSystems_AquireSecureData_Instance : A...
Claim ExposedEncryption(connection1)	1				All exposed connections must have encryption

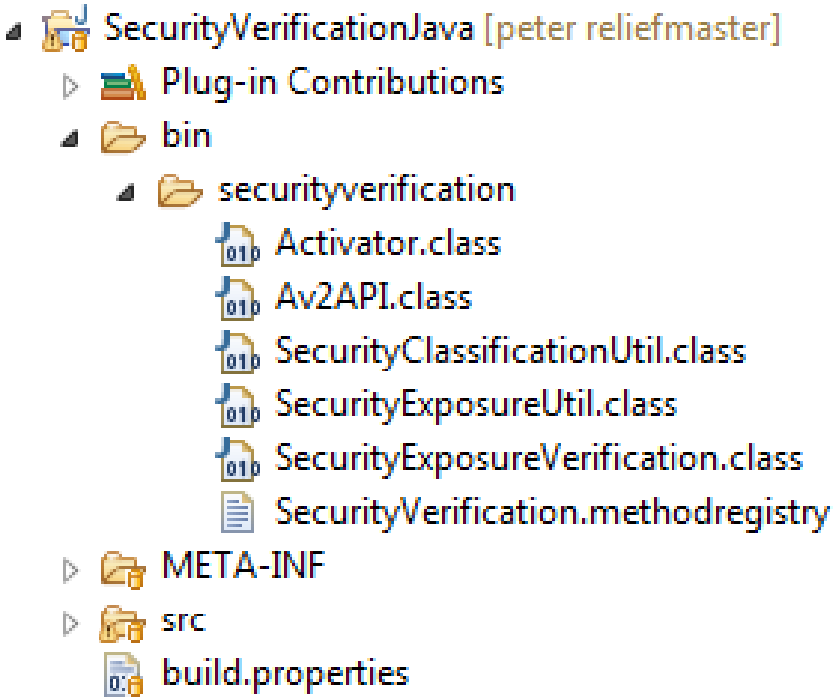
Resolute Security Verification



Resolute Security

- Properties
- Claims
- Functions
- Utilities

Java Security Verification – Methods, Classes, Utilities




```
verification methods
SecurityVerificationMethods [
method
ExposedComponentConnectionsEncrypted
(component)report: "Check that all
connections owned by a given component
are encrypted if going over exposed
physical components" [
java
securityverification.SecurityExposureVeri
fication.allExposedConnectionsEncrypted
]
method
ExposedConnectionEncrypted(connection)
boolean: "Check that all connections
owned by a given component are encrypted
if going over exposed physical
components" [
java
securityverification.SecurityExposureVeri
fication.exposedConnectionEncrypted
    ]
]
```

Outline

Policies and Requirements

- Documentation
- Verification

Protections

- Information/Data Protection
- Access Control and Protection
  - Authentication
  - Authorization
- Action/Command Protection

Security Architectures

- Specialized architectures
- Cross Domain Solutions

Analyzing Vulnerabilities/Threats

Protections - Overview

Information/Data Protection

- Properties
- Verification ALISA, Resolute, Java

Access Control and Protection

- Authentication
  - Properties
  - Verification methods (ALISA – claim, activity, methods)
  - Resolute claims
- Authorization
  - Properties
  - (work in progress)

Action/Command Protection (work in progress)

Information/Data Protection

Security Classification and Levels as Properties

- Information security levels (Principal and Caveat)
- Primary and Secondary Access Classifications
  - Each has Principal Classification and Caveat

Encryption as Properties

- Encryption Scheme (Encryption Type)

Protected Containment

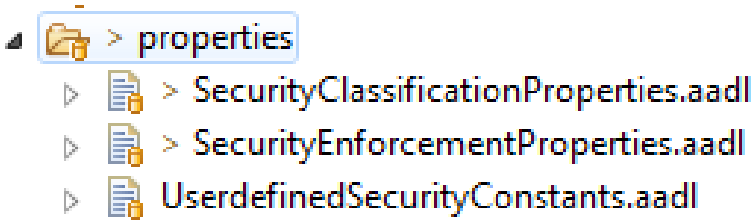
Verification

- ALISA assurance cases and methods
- Resolute
- Java

Information/Data Protection Properties

Classification Properties

- data
- access components



Encryption

- properties
- data, connections, and hardware (storage and transmission)
  - connection and supporting hardware must be consistent
- key management

Verification

- Resolute security functions and claims in libraries
- Java methods

Information/Data Protection - Security Classifications

Information security level properties include a primary security classification (e.g. Top Secret, Secret, Confidential) and a caveat (e.g. control markings) statement.

```
Information_Security_Level: aadlstring applies to (all);
Information_Security_Caveat: aadlstring applies to (all);
```

There is a primary and secondary security clearance. Each includes a principal security classification (e.g. Top Secret, Secret, Confidential) and an supplemental statement (e.g. specialized authorizations or restrictions).

```
Security_Clearance: aadlstring applies to (system, device);
Security_Clearance_Supplement: aadlstring applies to (system, device);
Secondary_Security_Clearance: aadlstring applies to (system, process,
thread, data, processor, memory);
Secondary_Security_Clearance_Supplement: aadlstring applies to (system,
process, thread, data, processor, memory);
```

Information/Data Protection – Encryption Properties

```
encryption: aadlboolean applies to (all);

encryption_scheme : Security_Properties::encryption_type applies to (all);
encryption_type : type record
(
encryption_form : enumeration (symmetric, asymmetric, hybrid,
                   authenticated_encryption, no_encryption, AEAD);

algorithm : enumeration (tripledes, des, rsa, blowfish, twofish, aes, D_H,
ECC, clear);

private_key: aadlstring;  -- maybe better as an integer?
public_key: aadlstring;
single_key: aadlstring;
authentication_type: enumeration (EtM, MtE, E_and_M, AEAD);
MAC_key : aadlstring;
```

Information/Data Protection - Analysis

Resolute and Java Verification Libraries

```
method ForallConnectionsAcrossProcessorsConnectionIsEncrypted (root):
"For all connections: encrypted when across different processors" [
resolute
Security_Exposure_Verification.Security_Exposure_Verification_public.Resolute.Resolute.forall_conne
ctions_across_processors_connection_is_encrypted
]
```

- securityverification
  - Activator.class
  - Av2API.class
  - SecurityClassificationUtil.class
  - SecurityExposureUtil.class
  - SecurityExposureVerification.class
  - SecurityVerification.methodregistry

- resolute
  - Connectivity_Reachability.aadl
  - Design_Flaw_Verification.aadl
  - Resolute_Patterns.aadl
  - Resolute_Util.aadl
  - Security_Classification_Util.aadl
  - Security_Classification_Verification.aadl
  - Security_Exposure_Util.aadl
  - Security_Exposure_Verification.aadl
  - SecurityClassification.methodregistry
  - SecurityExposure.methodregistry
  - Space_Partitioned.aadl
  - Userdefined_Resolute_Constants.aadl
  - Vulnerability_Impact.aadl

Access Control and Protection

Authentication

```
Authenticator: aadlboolean applies to (all);

AuthenticationTypeAccess: enumeration (NoValue, single_password,
smart_card,  ip_addr, two_factor, multi_layered, bio_metric) applies to
(all);

AuthenticationProtocol: enumeration (NoValue, cert_services, EAP, PAP,
SPAP, CHAP, MS_CHAP, Radius, IAS, Kerberos, SSL, NTLM)  applies to (all);
```

Authorization Protocols

- establish access rights and actions of an authenticated entity

Work in Progress

Should we include public key infrastructure (PKI) - public key and certificate modeling? Perhaps only internal key management within an architecture?

Action/Command Protection

Model, assess, and assure access control of execution of actions/commands

- Assess Control Flow
- Trusted Execution Instances

This may require modeling of control flow including subprograms.
Not sure how much is applicable within an architecture model.

Work in Progress

Outline

Policies and Requirements

- Documentation
- Verification

Protections

- Information/Data Protection
- Access Control and Protection
  - Authentication
  - Authorization
- Action/Command Protection

Security Architectures

- Specialized architectures
- Cross Domain Solutions

Analyzing Vulnerabilities/Threats

Security Architectures

Modeling Specialized Architectures

- AADL core modeling capabilities and ARINC 653
- Potentially specialized security constructs
- Example models
  - MILS and Cross Domain
  - Protected Containment
  - Secure Virtualization
  - Secure kernels
  - Cross Domain Solutions

Intent is to develop examples and guidance for modeling and analysis utilizing core AADL and properties, etc. of the security annex.

Analyzing Security Architectures

- ALISA, Resolute
- Adventium MILS Tool

Work in Progress

Vulnerability/Treat Analysis

Vulnerability - system state that could be exploited by an attacker involving

- Design
- Procedures and Operation
- Constraints

AADL

- Vulnerabilities Property
- EMV2

Architecture Analysis may be layered

- Attack surfaces
- Within the system architecture
  - access paths/traces
  - attack trees
  - chain of events

Should we consider a state machine based model for analysis of vulnerabilities, their impacts, attack surfaces?

Work in Progress

Vulnerabilities Property

```
Vulnerabilities: list of record(
Name : aadlstring; -- short identification phrase for the vulnerability
Description : aadlstring;     -- description of the vulnerability
CrossReference : aadlstring;  -- cross reference to an external document
Phases : list of aadlstring;  -- operational phases in which the vulnerability may be
exploited
Environment : aadlstring;     -- description of operational environment
Threat : aadlstring;          -- description of the circumstances under
                                 which the vulnerability may be exploited
Loss : aadlstring; -- description of the loss that may result
Risk : aadlstring;  -- description of risk
Severity : EMV2::SeverityRange ; -- actual risk as severity
Likelihood : EMV2::LikelihoodLabels;    -- actual risk as likelihood/probability
Probability: EMV2::ProbabilityRange;    -- probability of a exploitation (i.e. realization
of loss)
 AcceptableSeverity : EMV2::SeverityRange; -- acceptable risk as severity
  AcceptableLikelihood : EMV2::LikelihoodLabels; --acceptable risk as likelihood/probability
  DevelopmentAssuranceLevel : EMV2::DALLabels;     -- level of rigor in development
   VerificationMethod : aadlstring; -- verification method to address the vulnerability or
threat
   SecurityReport : aadlstring;     -- analysis/assessment of hazard
   Comment : aadlstring;
          )  applies to (all);
```

Summary

Develop a Draft Annex Standard Document

- Description and Guidelines
- Representative Examples (E-Enable Aircraft Models)
- Libraries of claims and methods

Annex Summary

- Core AADL with security-specific properties
- ALISA for policies and requirements capture
- Verification of requirements and other analyses with
  - ALISA assurance cases (Resolute and Java based methods)
  - Resolute Claims
- Custom plugins (modeling and analysis) - TBD