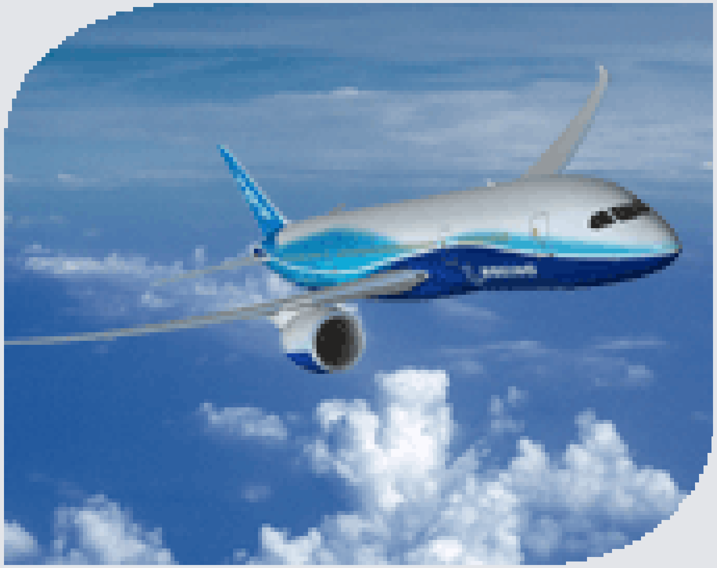
Security Architecture and Formal Analysis of an Airplane Software Distribution System

David von Oheimb [1]
Monika Maidl [1]
Richard Robinson [2]

[1] Siemens Corporate Technology, Munich
[2] Boeing Phantom Works, Seattle



ICAS/AIAA ATIO 2008 Congress
Anchorage, USA, 15 Sep 2008

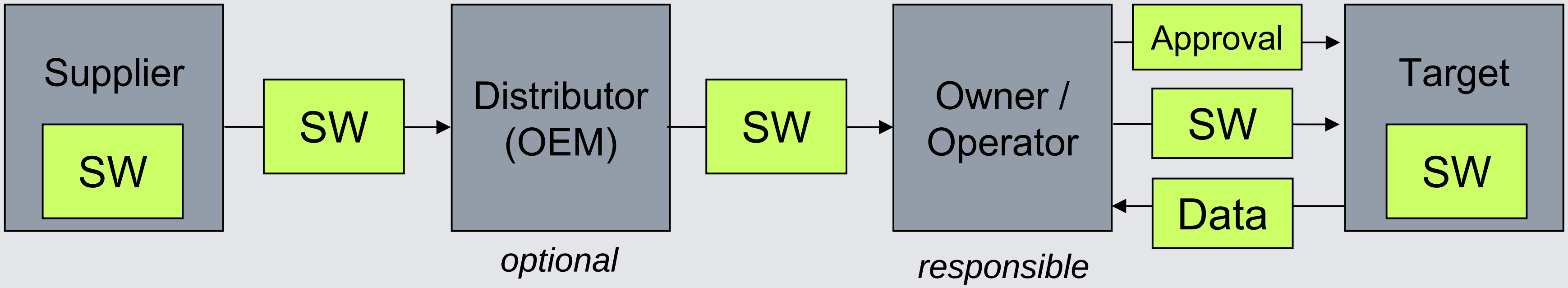
Overview

- **A**irplane **A**sset **D**istribution **S**ystem
- Hybrid security assessment and architecture
- Formal crypto protocol model
- Validation with AVISPA Tool
- Conclusion

© 2008 The Boeing Company and Siemens AG

Airplane Asset Distribution System (AADS)

System providing secure distribution of software (aka. LSAP, parts, assets) and data from software supplier to aircraft in production or in service
→ **A**irplane **A**sset **D**istribution **S**ystem (AADS)



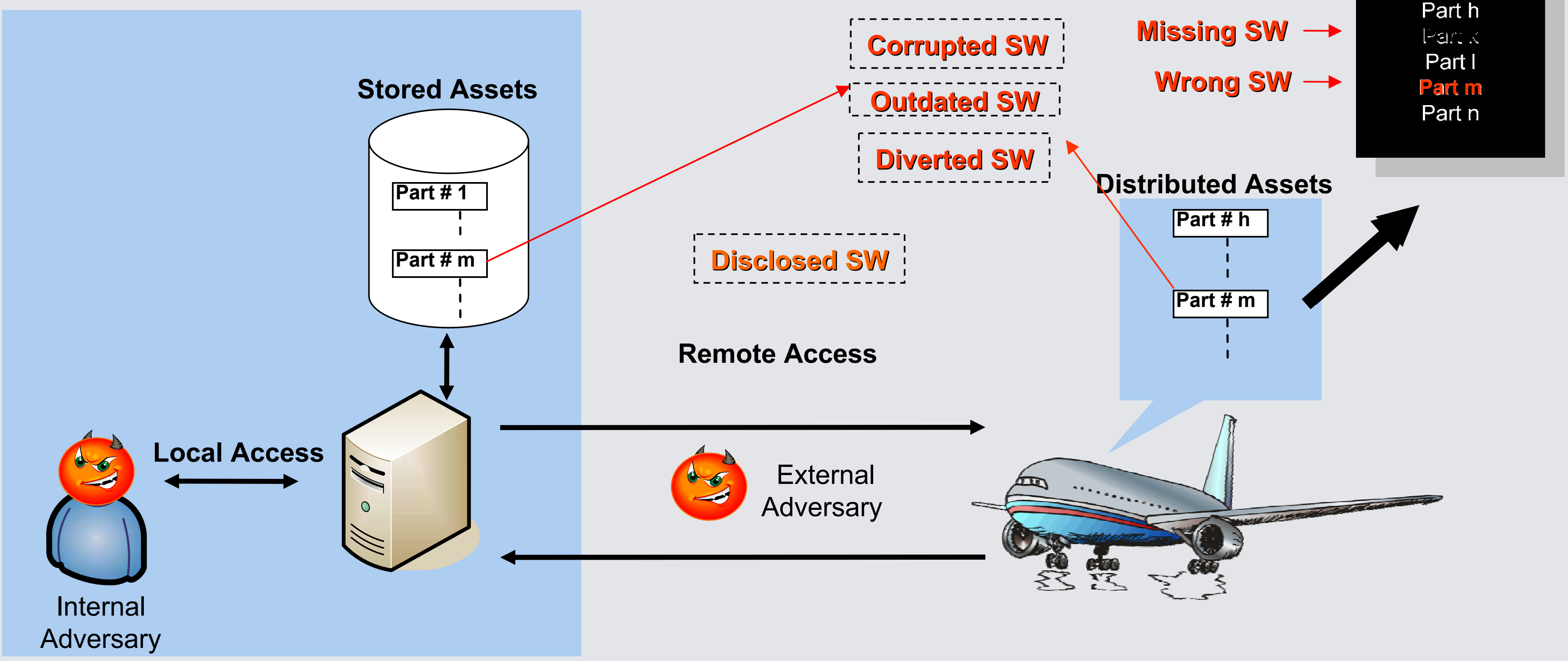
More general: IT system with networked devices in the field performing safety-critical and/or security-critical tasks.
Field devices require secure update of embedded software.

Transition from media-based (CD-ROMs etc.) to **networked transport** increases **security risks** due to transport over open, insecure networks

 © 2008 The Boeing Company and Siemens AG

Security threats

Attacker's objective: lower airplane safety margins
by tampering software that will be executed onboard an airplane



Corruption/Injection **Wrong Version** **Diversion** **Disclosure**

© 2008 The Boeing Company and Siemens AG

Overview

- **A**irplane **A**sset **D**istribution **S**ystem
- Hybrid security assessment and architecture
- Formal crypto protocol model
- Validation with AVISPA Tool
- Conclusion

© 2008 The Boeing Company and Siemens AG

Common Criteria (CC) for IT security evaluation

Common Criteria

product-oriented methodology

for **IT security assessment**

ISO/IEC standard 15408

Current version: 3.1 of end-2006

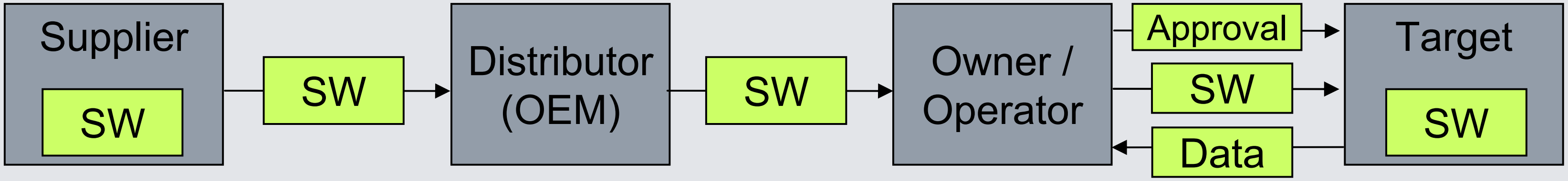
Aim: gain confidence in the security of a system

- What are the objectives the system should achieve?
- Are the measures employed appropriate to achieve them?
- Are the measures implemented and deployed correctly?

 © 2008 The Boeing Company and Siemens AG

Hybrid security assessment

- AADS usually are complex distributed systems with many components



- Highest CC evaluation assurance levels (EAL 6-7) require formal analysis

General problems:

- Complete formal analysis too costly
- CC offer only limited support ("CAP") for modular system evaluation

Pragmantic approach:

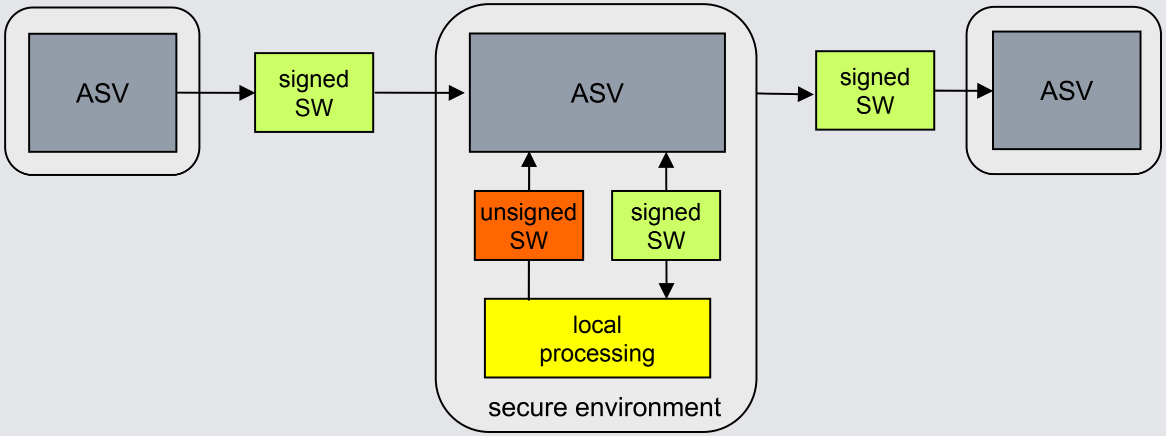
- Define **confined security kernel** with generic component: ASV
- **Asset Signer Verifier (ASV)** handles digital signatures at each node
- Evaluate ASV according to Common Criteria EAL4 (non-formal)
- Analyze the interaction of ASVs in a formal way (→ crypto protocol)

© 2008 The Boeing Company and Siemens AG

Asset Signer Verifier (ASV)

Each node in AADS runs an ASV instance, used for:

- Introducing unsigned software into the AADS, by digitally signing and optionally encrypting it
- Verifying the signature on software received from other ASVs, checking integrity, authenticity and authorization of the sender
- Approving software by adding an authorized signature
- Delivering software out of the AADS after successfully verifying it



© 2008 The Boeing Company and Siemens AG

Overview

- **A**irplane **A**sset **D**istribution **S**ystem
- Hybrid security assessment
- Formal crypto protocol model
- Validation with AVISPA Tool
- Conclusion

© 2008 The Boeing Company and Siemens AG

Formal modeling: Alice-Bob notation

```
SUP - {Asset.{h(Asset).DIS}_inv(KSUP).CertSUP}_KDIS -> DIS
DIS - {Asset.{h(Asset).DIS}_inv(KSUP).CertSUP
            .{h(Asset).OP }_inv(KDIS).CertDIS}_KOP  -> OP
OP  - {Asset.{h(Asset).DIS}_inv(KSUP).CertSUP
            .{h(Asset).OP }_inv(KDIS).CertDIS
            .{h(Asset).TD }_inv(KOP ).CertOP }_KTD  -> TD
```

`A - M -> B`	message M sent from A to B
`Asset`	a software item including its identity
`h(M)`	the hash value (i.e. crypto checksum) of content M
`M.N`	the concatenated contents of M and N
`{M}_inv(K)`	content M digitally signed with private key K
`{M}_K`	content M encrypted with public key K

© 2008 The Boeing Company and Siemens AG

Formal modeling: AADS node structure

```
SUP  - {Asset.{h(Asset).DIS}_inv(KSUP).CertSUP}_KDIS -> DIS
DIS  - {Asset.{h(Asset).DIS}_inv(KSUP).CertSUP
             .{h(Asset).OP }_inv(KDIS).CertDIS}_KOP   -> OP
OP   - {Asset.{h(Asset).DIS}_inv(KSUP).CertSUP
             .{h(Asset).OP }_inv(KDIS).CertDIS
             .{h(Asset).TD }_inv(KOP ).CertOP }_KTD   -> TD
```

SUP: software supplier with private key **inv(KSUP)**

DIS: software distributor with private key **inv(KDIS)**

OP : target operator with private key **inv(KOP)**

TD : target device with private key **inv(KTD)**

Signatures comprise hash value of asset and identity of intended receiver

Signatures are applied in parallel (rather than nested or discarded)

© 2008 The Boeing Company and Siemens AG

Formal modeling: approvals and certificates

```
SUP - {Asset.{h(Asset).DIS}_inv(KSUP).CertSUP}_KDIS -> DIS
DIS - {Asset.{h(Asset).DIS}_inv(KSUP).CertSUP
            .{h(Asset).OP }_inv(KDIS).CertDIS}_KOP  -> OP
OP  - {Asset.{h(Asset).DIS}_inv(KSUP).CertSUP
            .{h(Asset).OP }_inv(KDIS).CertDIS
            .{h(Asset).TD }_inv(KOP ).CertOP }_KTD  -> TD
```

- Certificate of a node relates its identity with its public key, e.g. certificate of supplier `SUP`: **`CertSUP`** `= {SUP.KSUP}_inv(KCA)`
- Certificate authority (CA) with private key `inv(KCA)`
- Certificates are self-signed or signed by CA
- Locally stored sets of public keys of trusted ASVs and CAs
- Approval information partially modelled: **`operator`** specifies **`target`**

 © 2008 The Boeing Company and Siemens AG

Overview

- **A**irplane **A**sset **D**istribution **S**ystem
- Hybrid security assessment and architecture
- Formal crypto protocol model
- Validation with AVISPA Tool
- Conclusion

Formal validation: goals

Show asset authenticity, integrity and confidentiality*:*

- assets accepted by target have indeed been sent by the supplier
- assets accepted by target have not been modified during transport
- assets remain secret among the ASV instances
- asset authenticity and integrity also hop-by-hop

Correct destination covered*:*

- Name of the intended receiver in signed part, checked by target.
  Signature of the operator acts as installation approval statement.

Correct version partially covered:

- Integrity of version info, *checks delegated* to ASV local environment.

© 2008 The Boeing Company and Siemens AG

Formal validation: remarks

Modelling:

- Alice-Bob notation not detailed and precise enough
- Use the specification language of the AVISPA Tool: HLPSL
- Asset Signer Verifier (ASV) as parameterized role, multiple instances
- AADS as communication protocol linking different ASV instances
- Multiple protocol sessions describing individual SW transports

Checking:

- At the level of detail of the model, **all goals are met**
- Modelcheckers at their complexity limits, due to
  - parallel signatures, only the latest one being checked
  - multiple instances of central nodes (e.g. manufacturer)
  - …?

© 2008 The Boeing Company and Siemens AG

Overview

- **A**irplane **A**sset **D**istribution **S**ystem
- Hybrid security assessment and architecture
- Formal crypto protocol model
- Validation with AVISPA Tool
- Conclusion

© 2008 The Boeing Company and Siemens AG

Conclusion

- Challenges for AADS development
  - complex, heterogeneous, distributed system
  - security is critical for both flight safety and airline business
- Experience with AADS evaluation
  - Common Criteria most widely accepted methodology available
  - Problem of compositional security evaluation not solved
  - Use formal analysis where cost/benefit ratio is best
  - Highly precise design and documentation: assumptions, requirements
  - Shape system architecture to support security evaluation
- Future steps
  - Trust management aspects including Public Key Infrastructure (PKI)
  - Configuration management with installation instructions and reports

© 2008 The Boeing Company and Siemens AG