Deriving Security for Mixed IT System Architectures from Evaluated Products

6th ICCC 2005, Tokyo/Japan

David Ochel

atsec information security

Objective

- Security does not stop with the evaluation of individual products:
  - Product evaluations are an important input for the information security management of a consumer's global systems.

=> How can consumers, developers, and the CC community contribute to the enhancement of system security?

Agenda

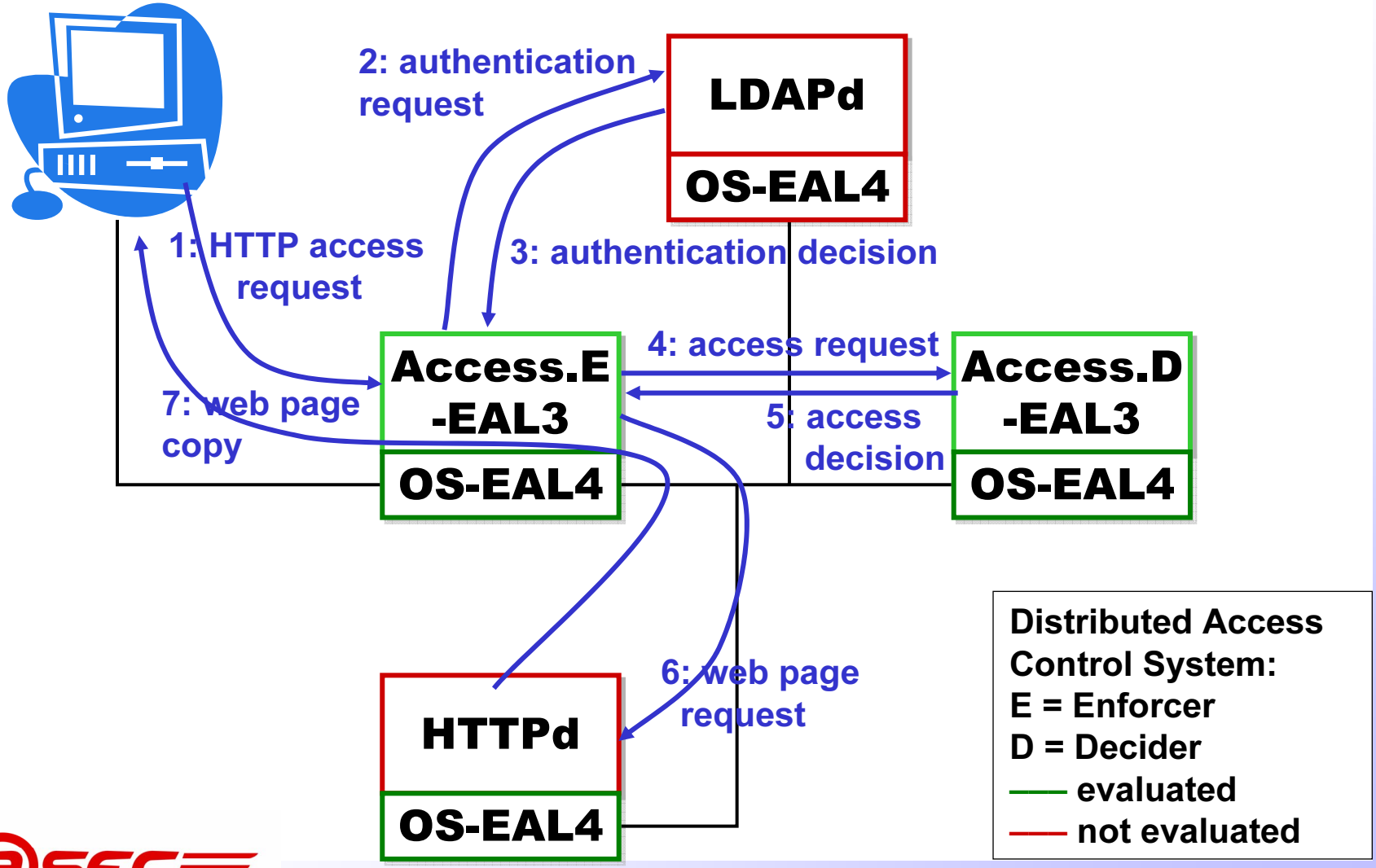
- Introduce Example System
  - Architecture and Information Flow
  - Some Security Functions
  - Some Potential Threat Vectors
  - Potential Remedies
- Identify Opportunities to Enhance System Security
- Questions and Answers

(This slide set contains animations, some content will be lost in printed versions.)

Example System – Distributed Access Control Enforcement

- Problem:
  - Enforce a common access control policy throughout a number of systems
  - Example: limited to web-based access
- Example Solution: Access Control Framework
  - Central, application-independent policy server computes access control decisions ("Access.**D**" component)
    - in addition, in our example system a central LDAP daemon hosts the user registry and performs authentication of users
  - Distributed, application-dependent resource managers enforce access control decisions ("Access.**E**" component)
    - can be arbitrary resource; in our example: web proxies
  - Compare to ISO/IEC 10181-3: separation between decision and enforcement functions

Example System – Architecture & Information Flow



Some Security Functions

no SF (?)

• access control enforcement
• security function (SF) management

Access.E -EAL3
OS-EAL4

LDAPd
OS

• client authentication
• user ID mapping
• protection of system's user registry
• SF management

• access control decisions
• SF management

Access.D -EAL3
OS-EAL4

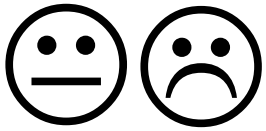
• File system DAC
• Resource Management
• Separation
• application TSF support
• SF management

no SF (?)

HTTPd
OS-EAL4

Some Potential Threat Vectors

malicious client:
- exploit Access.E proxy functionality
- circumvent Access.E by exploiting routing malfunction in OS

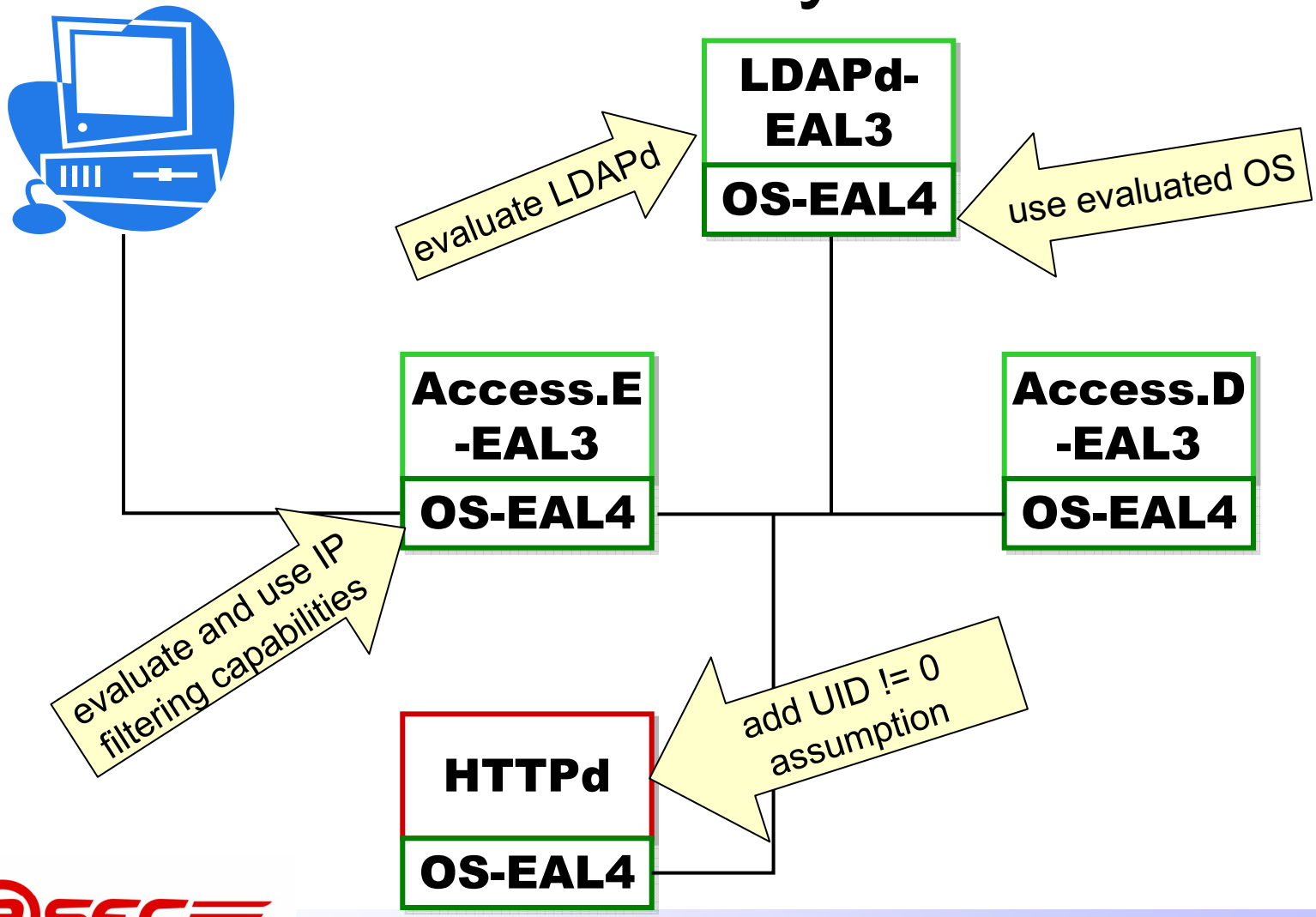


careless/malicious developer:
- authentication malfunction

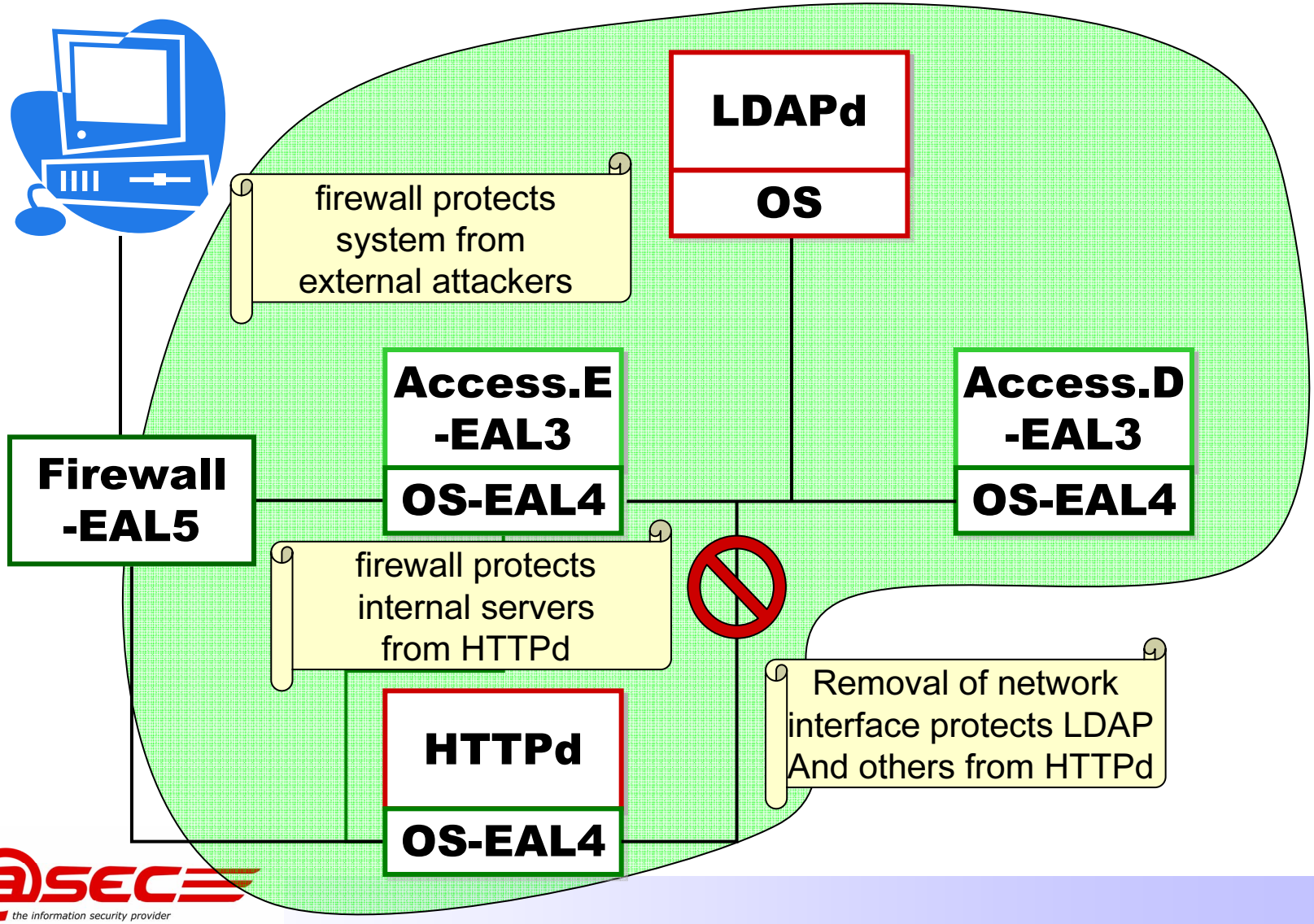
administrator error:
- configuration allows SF circumvention, e.g., HTTPd runs as root

malicious developer/user:
- exploit other systems in local network

Remedy by optimized use of evaluated systems



Remedy by system architecture



Roundup: Developer Opportunities

- Enhance evaluated configurations of products:
  - Harmonize TOE configurations so that they do not “bite” each other
    - more difficult when several developers are involved
  - Evaluate all security functionality and relevant interfaces offered by the product
  - Allow flexible configurations
  - Address TOE integration aspects
    - for example, define functional requirements for the IT environment, and meet those specified by others
  - Work with consumers in determining useful configuration

Roundup: Consumer Opportunities

When assembling systems:

- Where available, use evaluated products for systems with security functionality
- Identify threats
  - to (evaluated) security functions
  - to user data
- Perform risk analysis and mitigate by
  - system architecture
  - organizational policies
- Report problems with the evaluated configuration to the developer

Roundup: CC Community Opportunities

- Encourage developers to evaluate the complete set of security functions rather than subsets
- Encourage developers to address TOE integration issues
  - similar to component evaluation approaches: provide input for the consumer's system-wide risk analysis
- Develop framework for easy and effective system evaluations
  - again, there may be analogies to the concept of re-using component evaluation results

Questions?

mailto:david@atsec.com