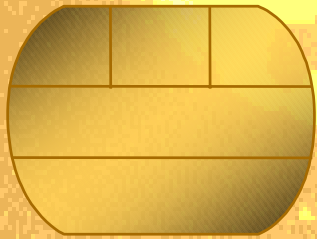


Open Platform Development

OPEN PLATFORM SECURITY



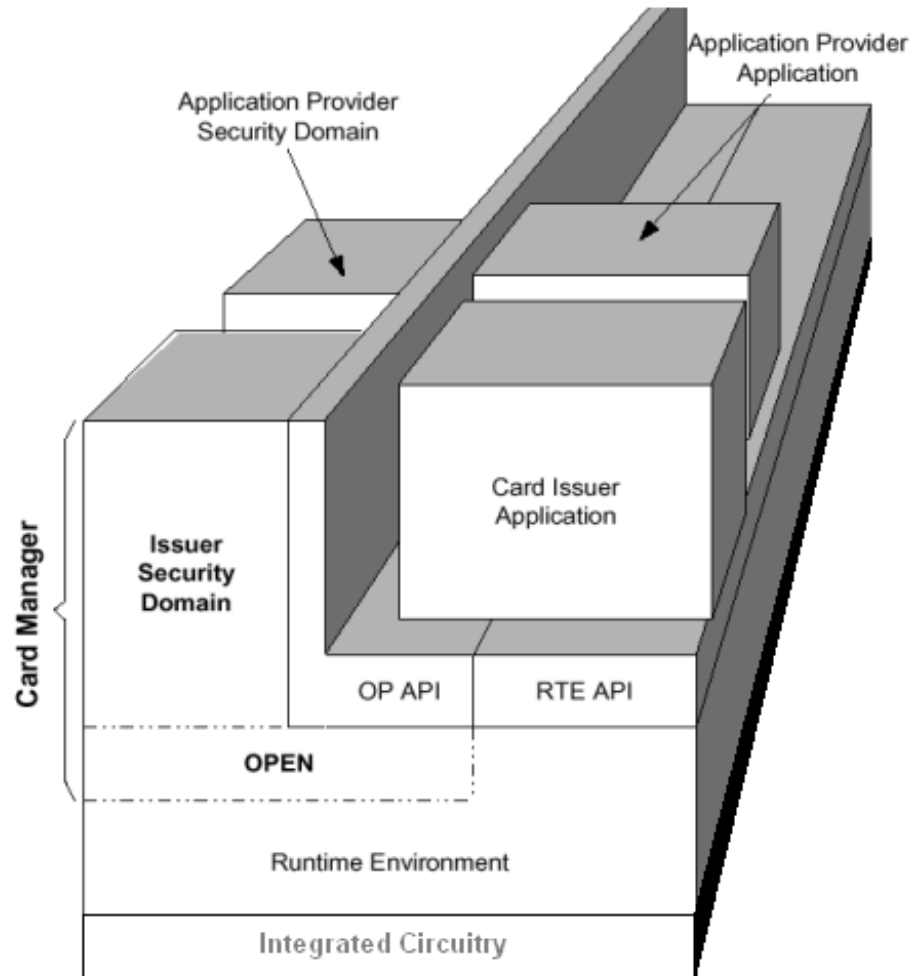
Mark Kekicheff
Forough Kashef
David Brewer



Introduction

- Open Platform is a cross industry standard for dynamic multi-application smart cards
 - ▶ *<http://www.globalplatform.org>*
- JavaCard™ 2.1.1 JCRE Specification
- Open Platform = JCRE “Installer” + card management & security services extras
- Need to demonstrate trustworthiness of OP
- Open Platform Protection Profile (OP3)
 - ▶ *<http://www.visa.com/openplatform>*

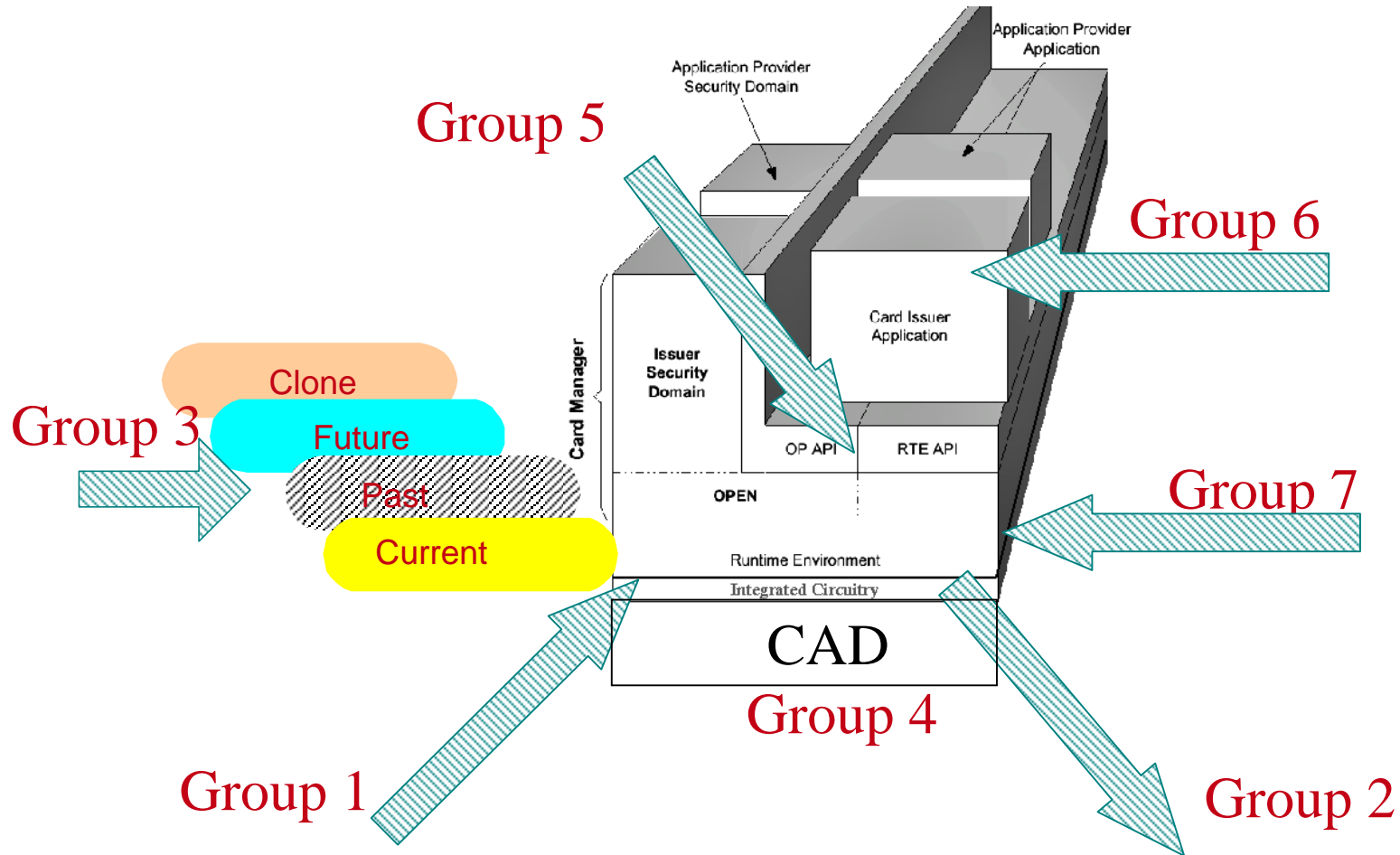
The Card Specification



Security Assumptions

- OP card is merely a component
- Need to trust
 - ▶ *back-office systems*
 - ▶ *cryptographic key management*
 - ▶ *card/chip operating environment (COE)*
 - ▶ *off-card security procedures (actors and roles)*
- Assumptions expose vulnerabilities that OP card cannot protect itself against

Security Threats



Security Functions

- Extensive access control rules (discretionary and mandatory)
- Intrusion detection, Secure recovery
- Cryptography
 - ▶ *host-card authentication, key confidentiality, message authentication, message encryption, MAC chaining*
 - ▶ *receipt generation and DAP / token verification*
- Application code verification

OP3

■ Usual structure 5.2.1.2 STATIC KEY DISTRIBUTION (FCS_CKM.2+1)

■ Bags of refinement, lots of iterations

■ Packages for OP options

■ Appendices on COE and applications

The TSF shall distribute **static keys** in accordance with a specified cryptographic key distribution method ([assignment: *the relevant APDU command and in association with the appropriate Security Domain*]) that meets the following: OPCS ref e. FCS_CKM.2+1.1

This component is necessary to support the various cryptographic operations that in turn support O.SECURE_COMMUNICATIONS, O.REPLAY_DEFENSE and O.AUTHORIZED_LOADING.

In contrast to FCS_CKM.2+2 (see section 5.8.1.4), in which keys generated on card are distributed to off-card entities, this component receives off card generated keys and distributes them to a Security Domain as appropriate.

Security API Definition: This component is utilized by the [select: *decryptVerifyKey, decryptData*] security API to decrypt a key received by the application within a Secure Channel. It is available to applications to load their own keys encrypted by their Security Domain's K_{SGK} .

The appropriate Security Domain could be the Issuer Security Domain.

Optional Components

- Assignments and selections for implementation choices

- Use packages for functional choices
 - ▶ *Basic package +*
 - ▶ *Delegated Management*
 - ▶ *DAP Verification*
 - ▶ *CVM*
 - ▶ *etc.*

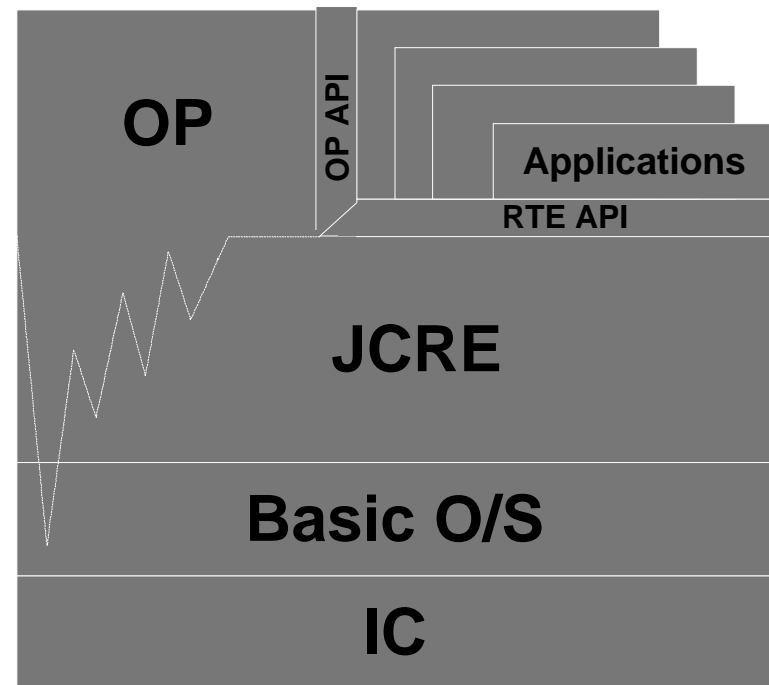
Packages

- Different assumptions, policies, objectives and functions - common threats

3.3	<i>Organizational Security Policies</i>	3-14
3.3.1	Organizational Security Policies Applicable to All TOE Configurations	3-14
3.3.1.1	P.GENERAL.....	3-14
3.3.1.2	P.ROLES.....	3-16
3.3.1.3	P.CRYPTOGRAPHY.....	3-16
3.3.1.4	P.CARD_MANAGER	3-16
3.3.1.5	P.SECURITY_DOMAIN	3-16
3.3.1.6	P.OP_API.....	3-17
3.3.1.7	P.STATE_TRANSITION.....	3-17
3.3.2	Organizational Security Policies Peculiar to the Delegated Management Package	3-18
3.3.2.1	P.DELEGATED_MANAGEMENT	3-18
3.3.3	Organizational Security Policies Peculiar to the DAP Verification Package	3-18
3.3.3.1	P.LOAD_FILE_VERIFICATION	3-18
3.3.4	Organizational Security Policies Peculiar to the Global PIN Package	3-18
3.3.4.1	P.GLOBAL_PIN	3-18
3.3.5	Organizational Security Policies Peculiar to the Intrusion Detection Package	3-19
3.3.5.1	P.INTRUSION_DETECTION	3-19
3.3.6	Organizational Security Policies Peculiar to the Application Code Verification Package	3-19
3.3.6.1	P.APPLICATION_CODE_VERIFICATION	3-19
3.3.7	Organizational Security Policies Peculiar to the DAP/ Token Generation Package	3-19
3.3.7.1	P.DAP/TOKEN_GENERATION.....	3-19

COE Specification

- Tamper resistant
- Resistant to DPA, etc.
- Facilitates OP recovery
- Reports exceptions to OP
- Prevents bypass, etc. of OP security
- Enforces applet separation
- Secure data erasure



Class A Security Target

The OP API

- OP Security services
 - ▶ *Secure channel, CVM, access to lifecycle states, etc.*
- How can an Application PP/ST invoke them without having them re-evaluated?
- Refine ADV-RCR.1 (maps API to TSFs during traceability analysis)
- Include specific assumptions (defined in OP3) in Application PP/ST

Security API Definition: This component is utilized by the [select: *decryptVerifyKey, decryptData*] security API to decrypt a key received by the application within a Secure Channel. It is available to applications to load their own keys encrypted by their Security Domain's K_{SSK} .

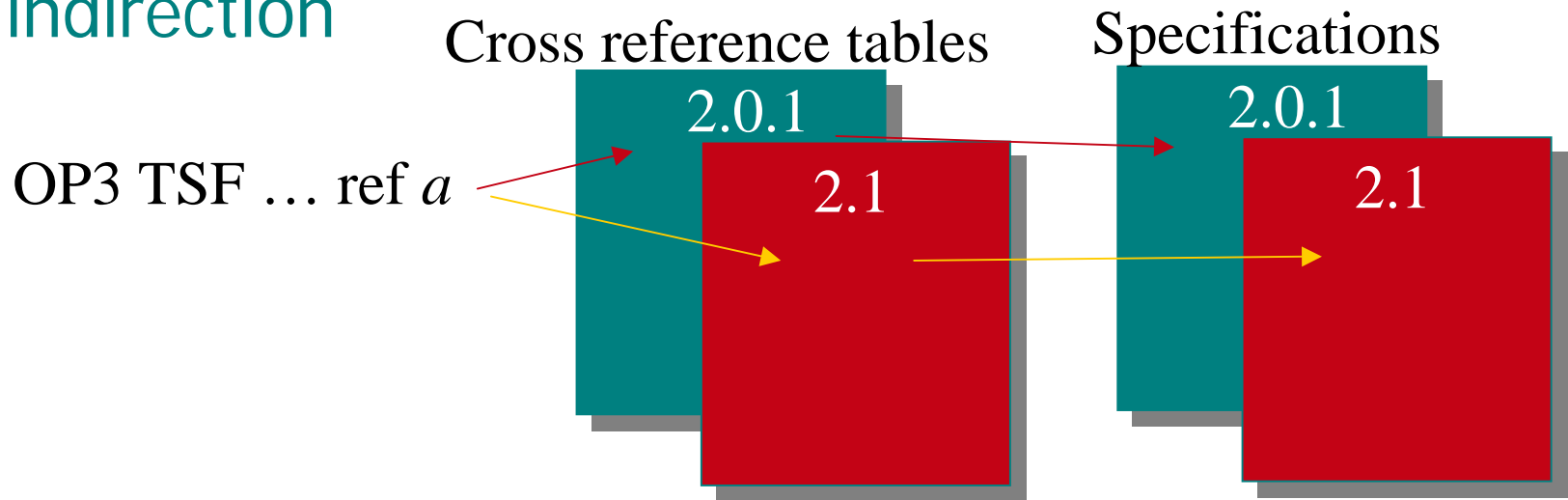


Application Code Verification

- Use FDP_ACC and FDP_ACF
- Orange Book: *clearance/classification* known
- AVC: calculate attribute by applying each rule to each Executable Module in Load File
- All rules must pass
- ST instantiation of P.APPLICATION_CODE_VERIFICATION defines AVC algorithms

Updating the OPCS

- OP3 is resilient to OPCS changes via reference indirection



- Works even for functionality changes
- But not if new TSFs are needed

Summary

- OP = secure dynamic multi-application card management framework
- OP3:
 - ▶ *very detailed protection profile*
 - ▶ *uses packages to handle options*
 - ▶ *requires COE/integration Protection Profile*
 - ▶ *provides security services to applets*
- Effective/practical security management
 - ▶ *MRA*
 - ▶ *separate evaluations possible*
 - ▶ *reconfigurable smart cards*