Java™ 2 Platform, Standard Edition (J2SE™) Security Looks Ahead:

New Features From Cryptography to XML Security

Jeff Nisewanger, Sean Mullan, Rosanna Lee

Java Security Engineers
Sun Microsystems, Inc.

Overall Presentation Goal

Learn about the latest security features and how they are being used to build secure applications and services

Speakers' Qualifications

- Speakers are members of the J2SE™ platform Security Engineering Team at Sun Microsystems

Presentation Agenda

- Recently added security features
- Upcoming new security features
- How the security features are being used
- Possible future directions
- Q&A

J2SE™ Platform Security Features

- Secure foundation: language features
  - Strongly typed
  - Bytecode verification
  - Runtime type safety checks
  - Dynamic class loaders
- Dynamic, extensible security model
  - Fine-grained access control to protect resources
  - Security policy enforced by security manager

J2SE™ Platform Security Features (Cont.)

- APIs for customers to build secure applications
- Rich feature set, with support for
  - Cryptography
  - Secure authentication and authorization
  - Public key infrastructure
  - Secure communications
  - Web services
- Standards-based and interoperable
- Pluggable use of third party security providers

New Features and Enhancements in J2SE™ 1.4 (1.4.0, 1.4.1, 1.4.2)

- Certification path building and validation API
- Java™ GSS API with Kerberos support
- Previously separate components now bundled
  - Cryptography (JCE)
  - Authentication/Authorization Framework (JAAS)
  - SSL/TLS (JSSE)
- Dynamic policies
- Footprint, startup, performance improvements
- Enhancements for PKCS, SecureRandom

Certification Path (CertPath) API and Implementation

- Pluggable building and validation of certification paths
- Pluggable retrieval of certificates/CRLs
- PKIX compliant path building and validation (RFC 3280 [1.4.2])
- Basic support for CRL Distribution Points extension [1.4.2]
- Major performance improvements [1.4.2]

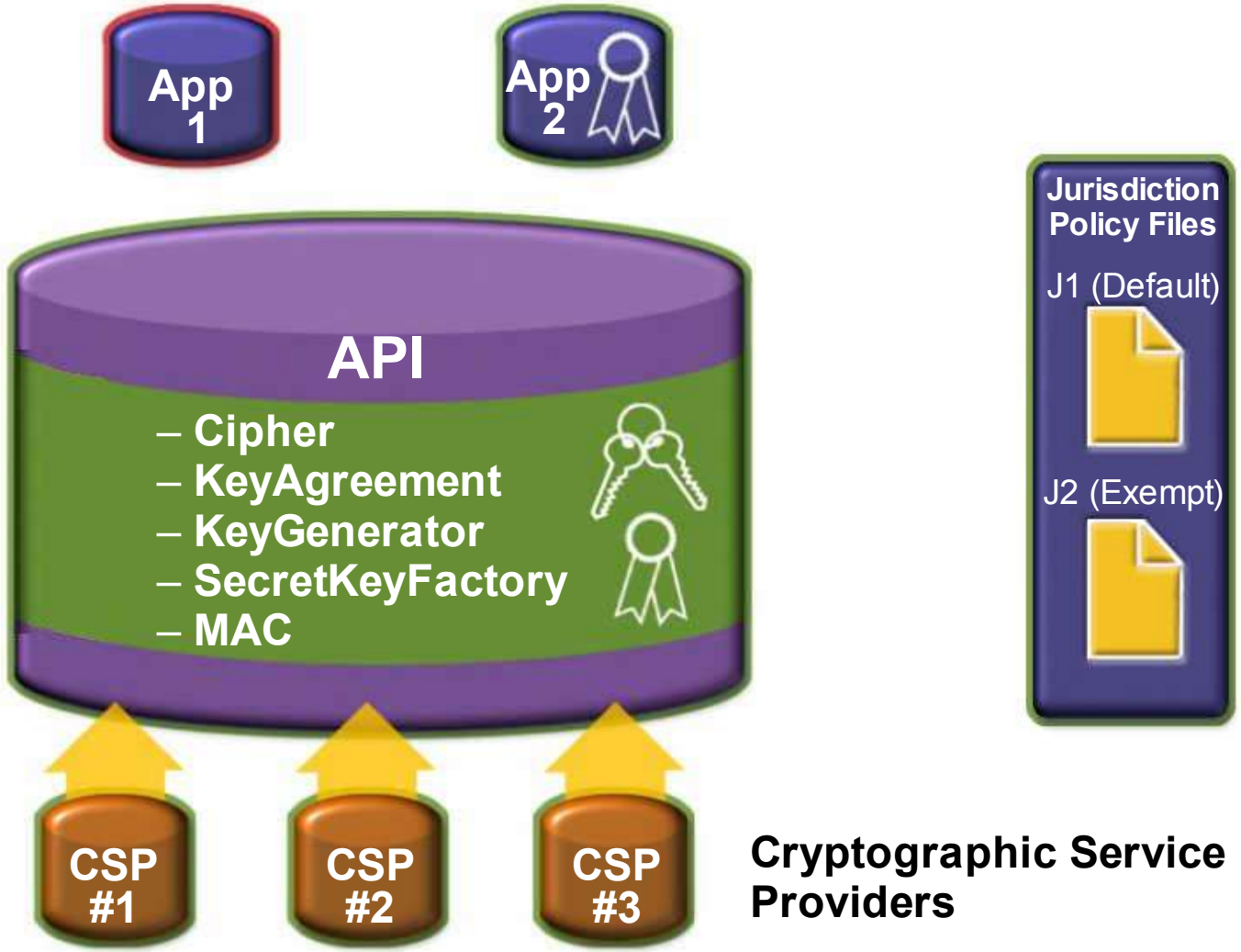
Java™ Generic Security Services API and Kerberos

- Standard Java™ language binding for GSS-API (RFC 2853)
- Supports Kerberos v5 (RFC 1510)
- Enables single sign-on in Kerberos environments
- Includes Kerberos client tools [1.4.1]
- More deployment options [1.4.2]

Java™ Cryptography Extension (JCE)

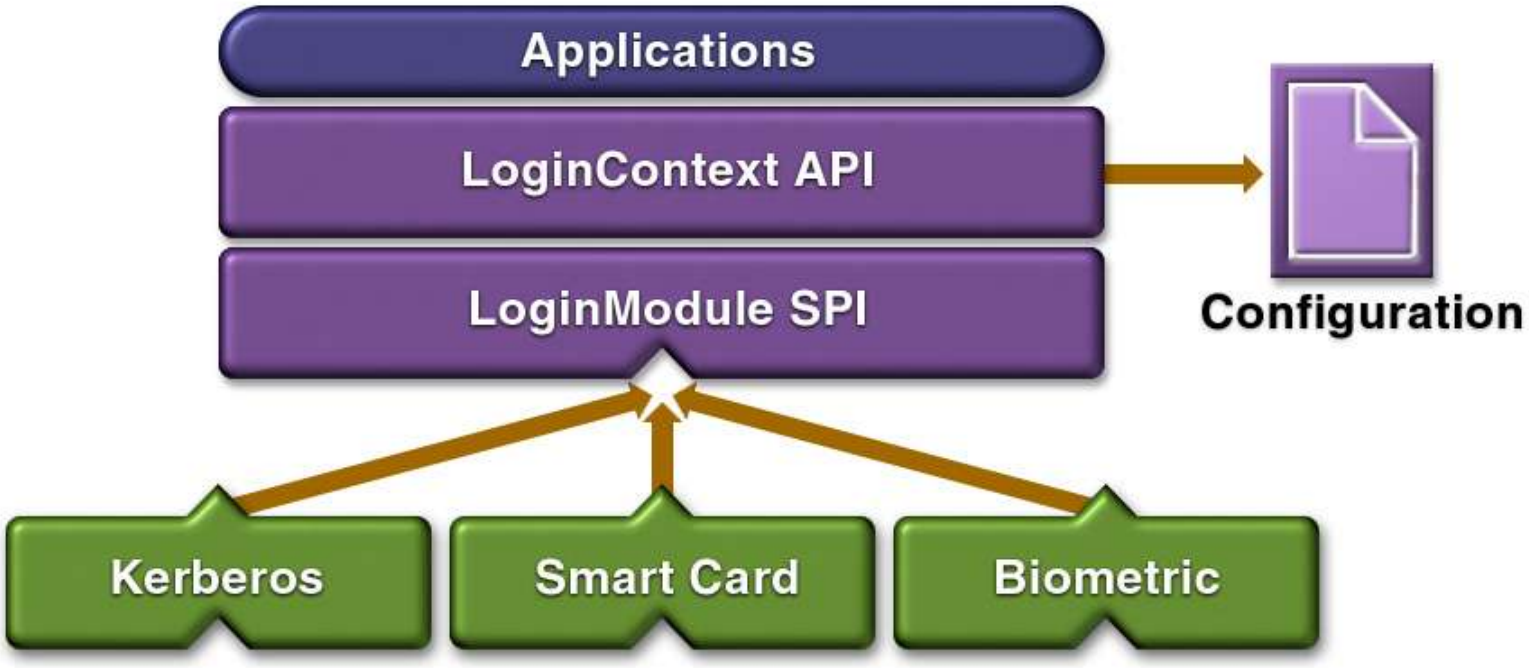
- Standard APIs and framework for encryption
- Pluggable cryptographic service providers
- Compatible with import and export regulations
- Sun's JCE provider supports rich set of algorithms
  - DES, 3DES, Blowfish, HMAC-MD5, HMAC-SHA1
  - AES [1.4.2]
  - Diffie-Hellman key agreement
  - PKCS #1, #5, and #8

JCE Architecture

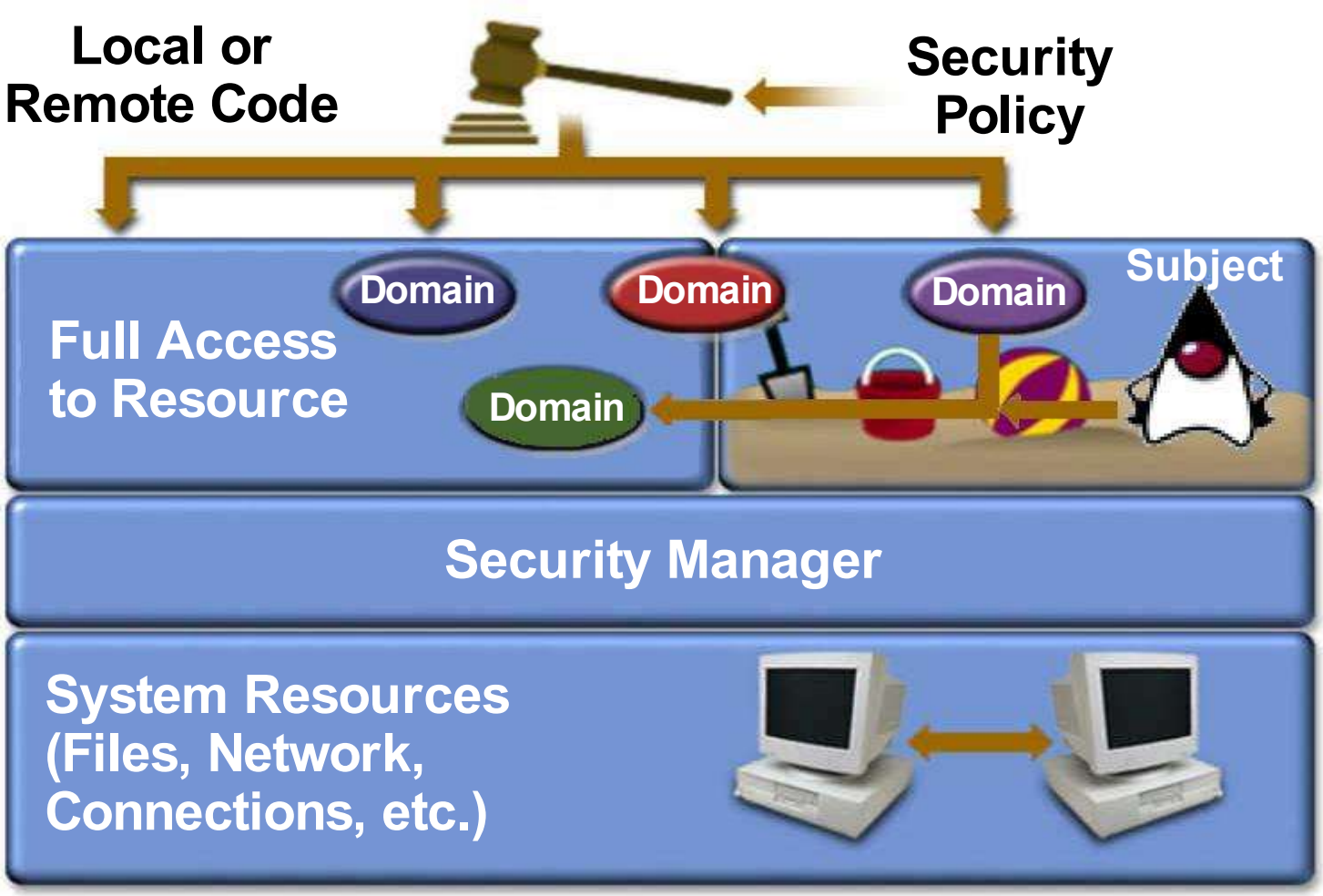


Cryptographic Service Providers

Java™ Authentication and Authorization Service (JAAS)



Subject-Based Authorization



Java™ Secure Socket Extension (JSSE)

- Standard socket API for SSL and TLS
- Provides transport level authentication, integrity, and privacy
- Supports standard cipher suites (including AES [1.4.2])
- Includes HTTPS URL handler
- Option to use PKIX CertPath trust manager [1.4.2]

Public Key Cryptography Standards (PKCS)

- PKCS 1 RSA Cryptography Standard (JCE)
- PKCS 5 Password-Based Cryptography (JCE)
- PKCS 8 Private-Key Information Syntax (JCE)
- PKCS 10 Certificate Request Syntax (Keytool)
- PKCS 12 Personal Information Exchange (KeyStore, currently read-only)

Upcoming New Features and Enhancements

- J2SE™ platform 1.5 defined by Umbrella JSR 176
- Some features available via other delivery vehicles
- Major security enhancement areas
  - Cryptography
  - XML security
  - PKI
  - Secure communications
  - Secure authentication
  - Ease of management/use

Cryptography (JCA/JCE) Enhancements

- Make API ready for Elliptic Curve Cryptography (ECC)
- NIO integration (ByteBuffers)
- JCE provider for PKCS 11 (Cryptographic Token Interface)
  - Smart cards
  - Hardware cryptographic accelerators
  - Optimized native crypto implementations
- Enhance PKCS 12 support (KeyStore)

XML Security

- XML Digital Signature API (JSR 105)
  - Java™ API for signing and validating XML signatures
  - JSR in Public Review
- XML Digital Encryption API (JSR 106)
  - Java™ API for encrypting/decrypting data in XML
  - JSR in Expert Group discussions

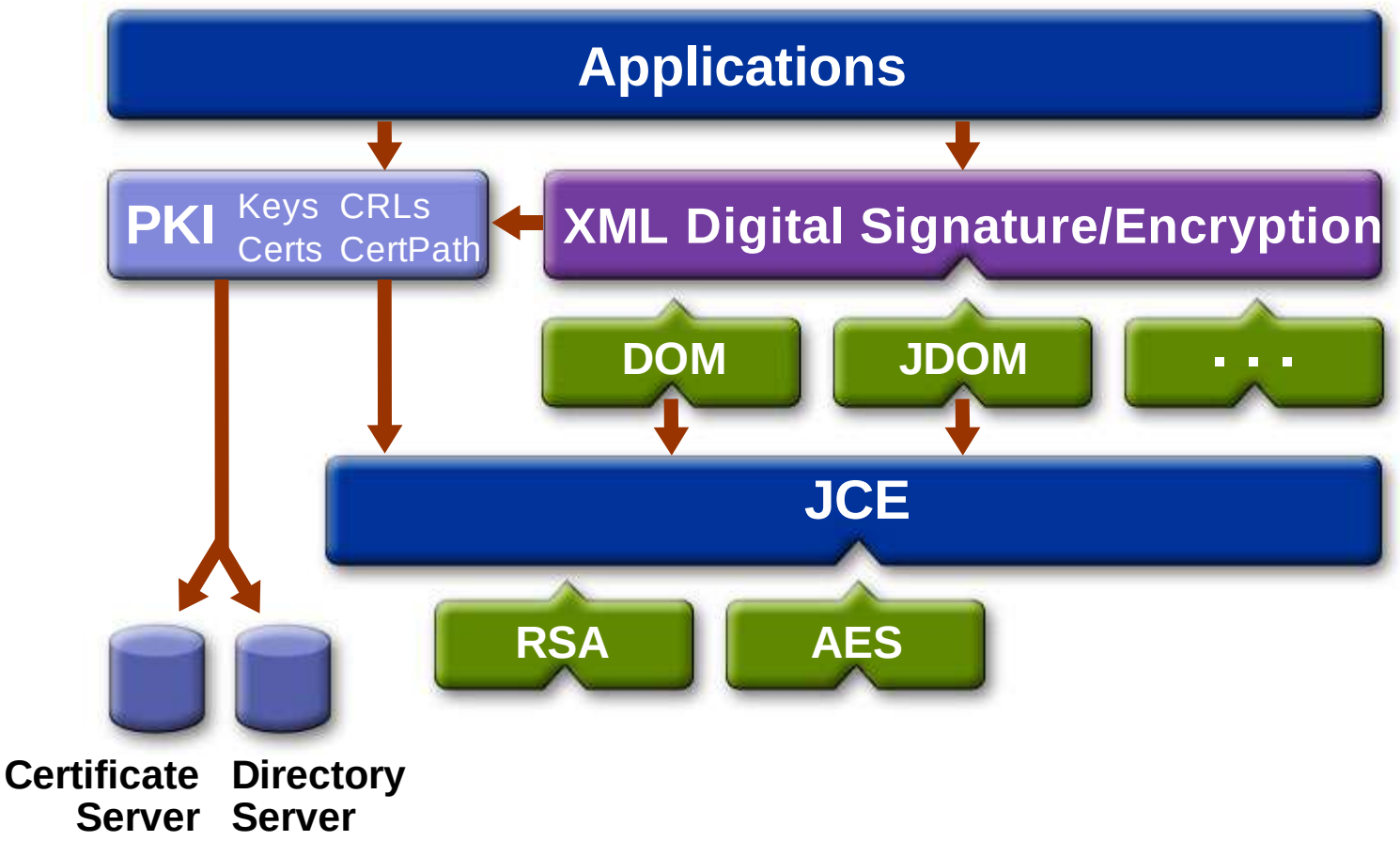
XML Digital Signature API Features

- Supports
  - W3C Recommendation, XML Signature Syntax and Processing
  - W3C Recommendation for Exclusive XML Canonicalization algorithm
  - W3C Recommendation for XPath Filter-2 Transform algorithm
- DOM-independent API
- Extensible, pluggable and provider-based

XML Digital Encryption API Features

- Supports
  - W3C Recommendation, XML Encryption Syntax and Processing
  - W3C Recommendation, Decryption Transform for XML Signature
- Design based on JSR 105 API
  - Reuse of common classes
  - Same “look and feel”

XML Security Architecture



Public Key Infrastructure Enhancements

- Certificate revocation checking via Online Certificate Status Protocol (OCSP)
- Support for additional X.509 certificate and CRL extensions
  - CRL Distribution Points
  - Issuing Distribution Point
  - Authority Information Access
  - Subject Information Access
- Minor enhancements to CertPath API

JSSE Enhancements

- SSLEngine
  - Useful for NIO, sockets and other IO abstractions
- Full pluggability
- Track the IETF TLS standard
  - TLS 1.1
  - TLS extensions
- PKIX CertPath trust manager will be default
- Kerberos cipher suites

Kerberos Enhancements

- 3DES and AES ciphers
- More deployment options
  - TGT renewals
  - Sub-session keys
  - Autoconfiguration via DNS

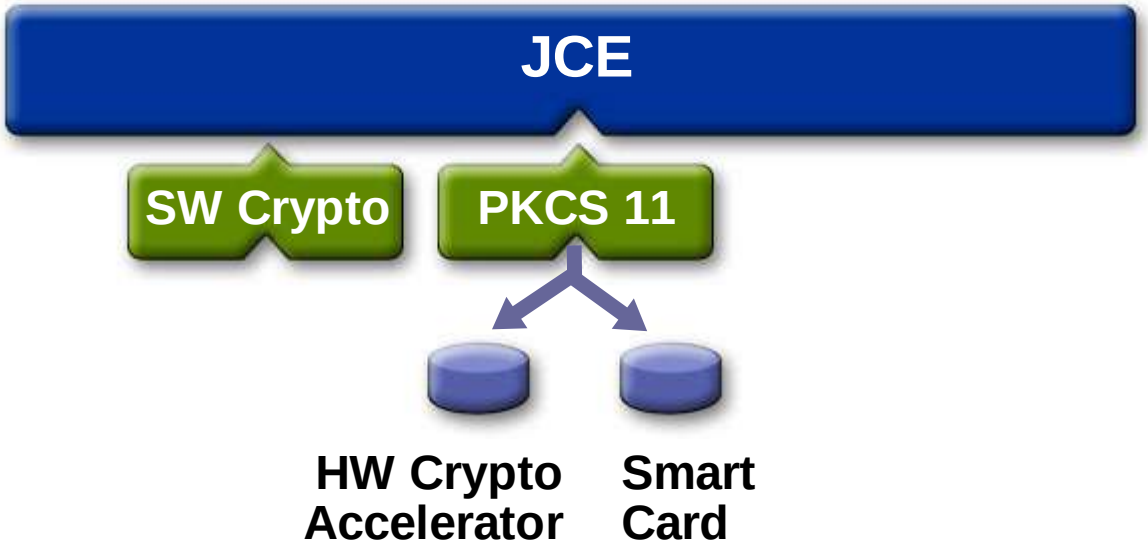
Simple Authentication and Security Layer (SASL)

- Provides pluggable authentication to network protocols (RFC 2222)
- Standard Java™ API and framework for SASL (JSR 28)
- Pluggable providers
- Provider for client/server Digest-MD5, CRAM-MD5, GSS-API/Kerberos mechanisms

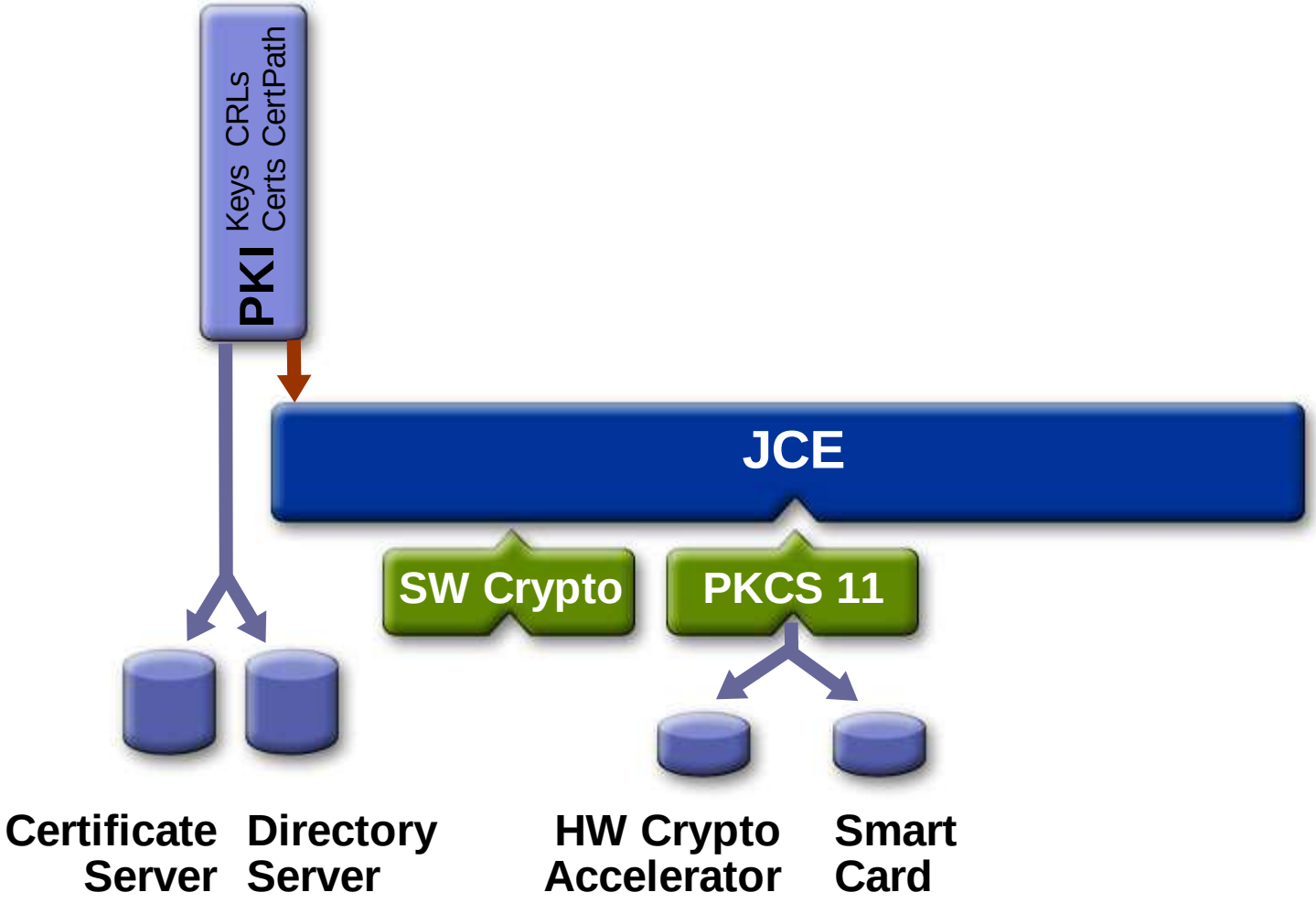
Miscellaneous Enhancements

- Secure authentication
  - Smart cards
  - Asynchronous callbacks
- Ease of management and use
  - Use of common logging facility
  - Chained exceptions
- Performance tuning

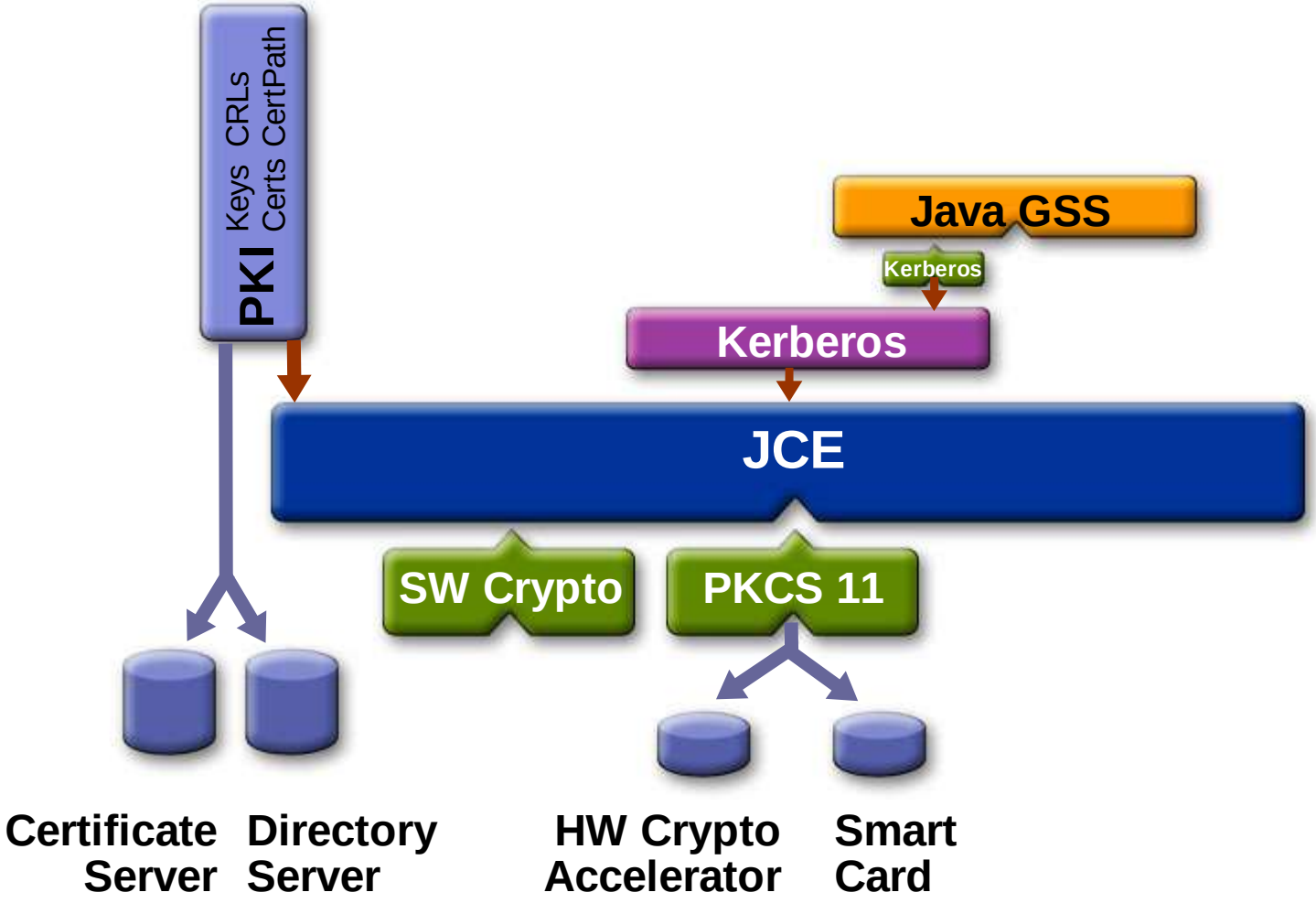
The Big Picture



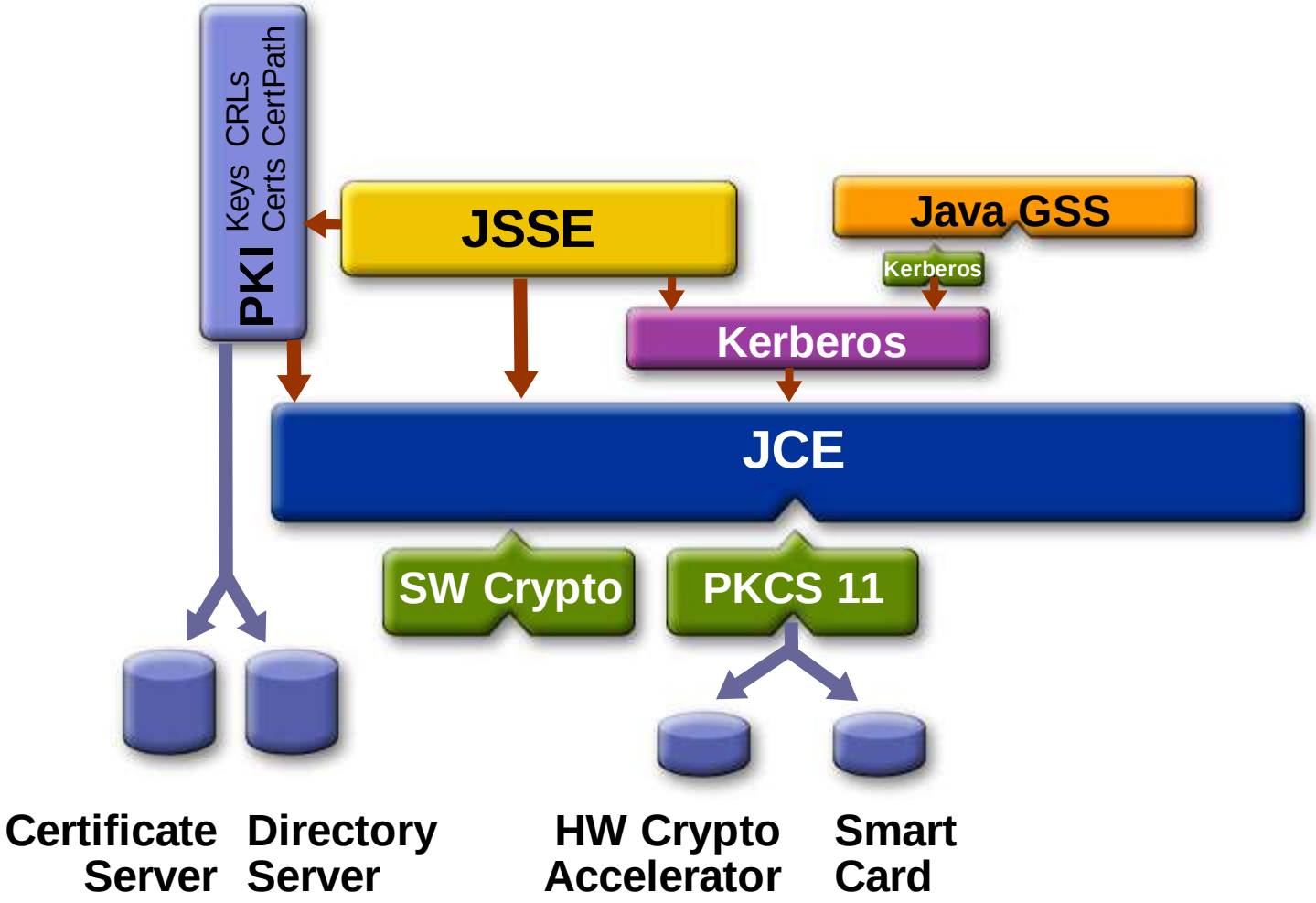
The Big Picture



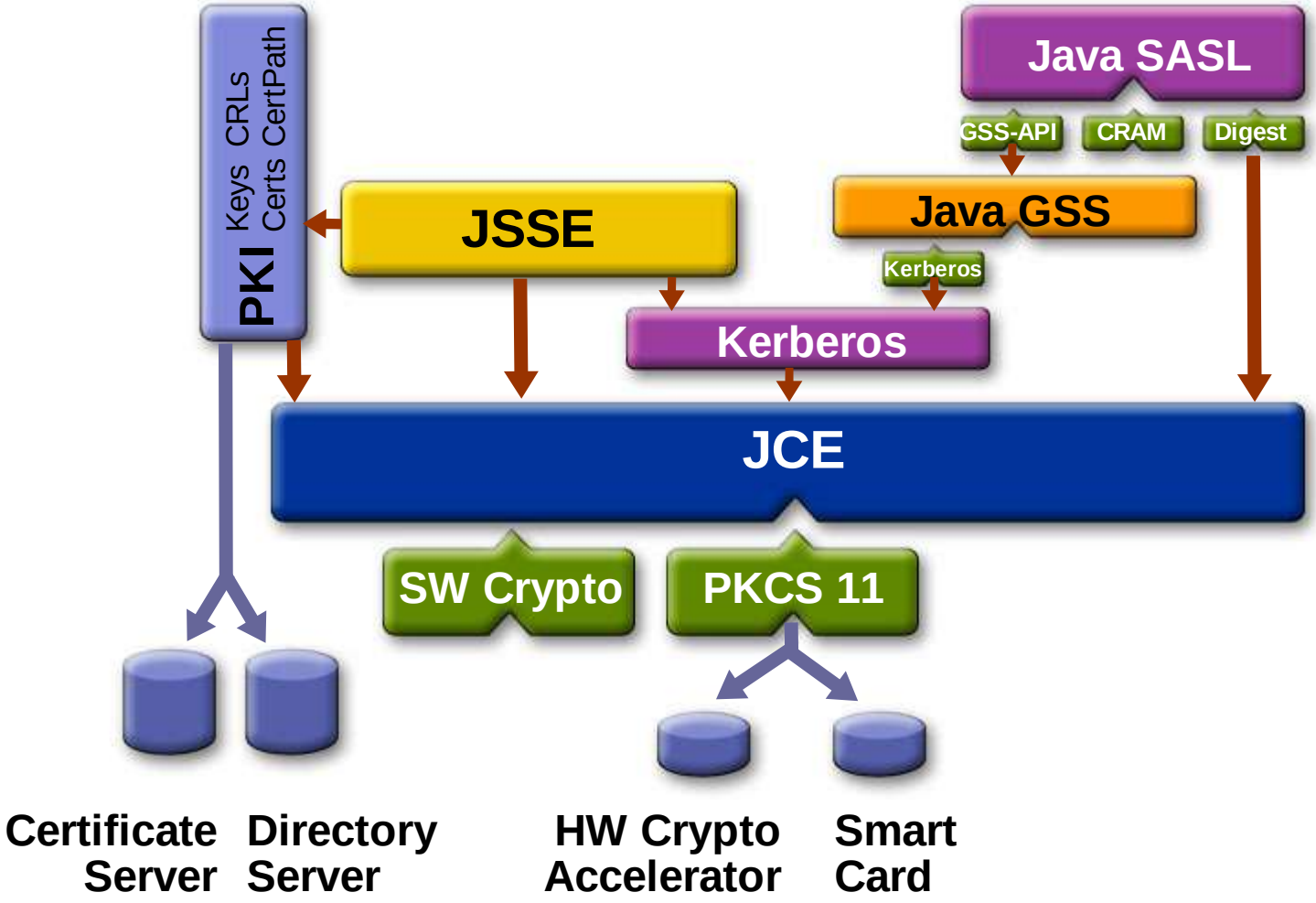
The Big Picture



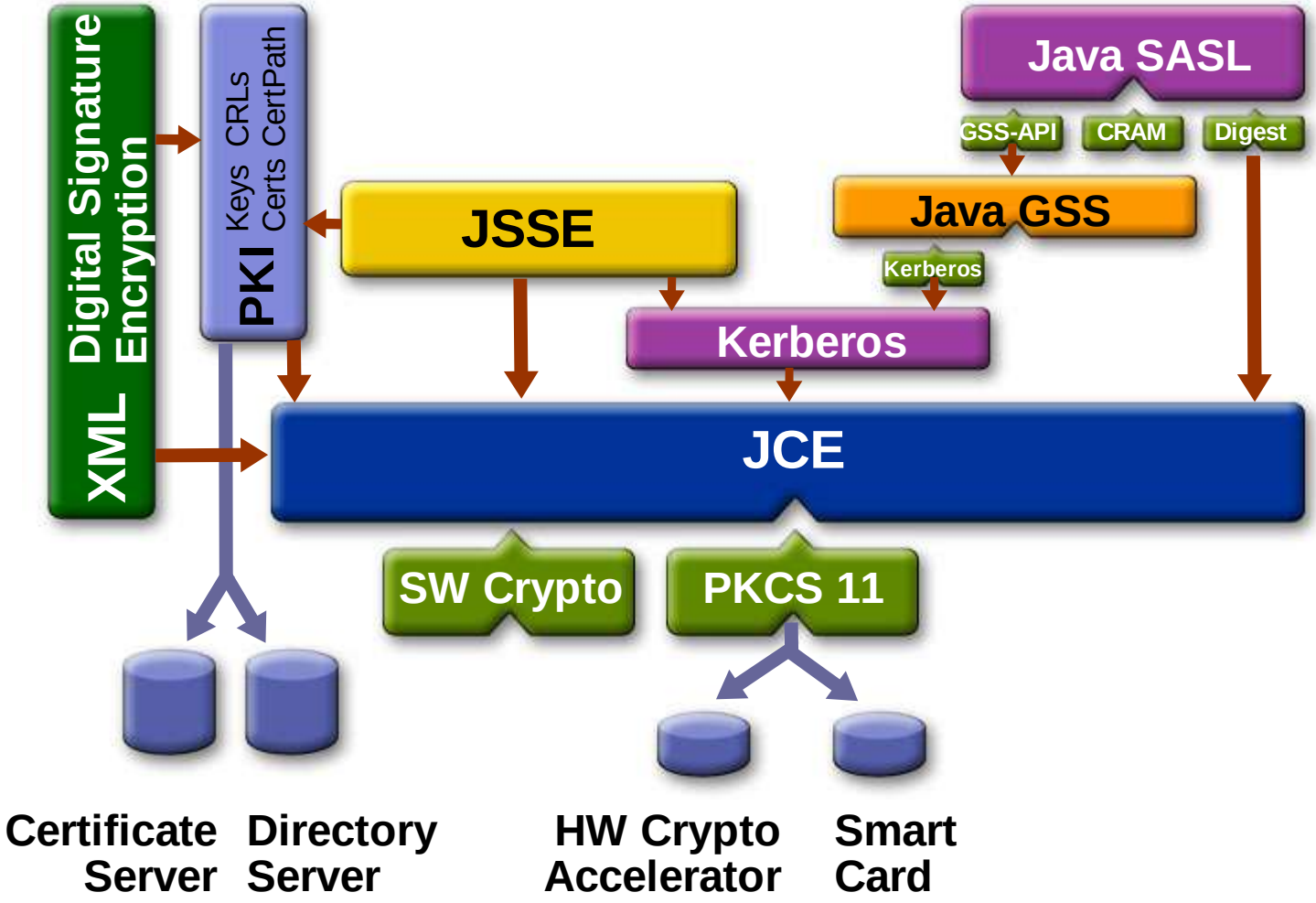
The Big Picture



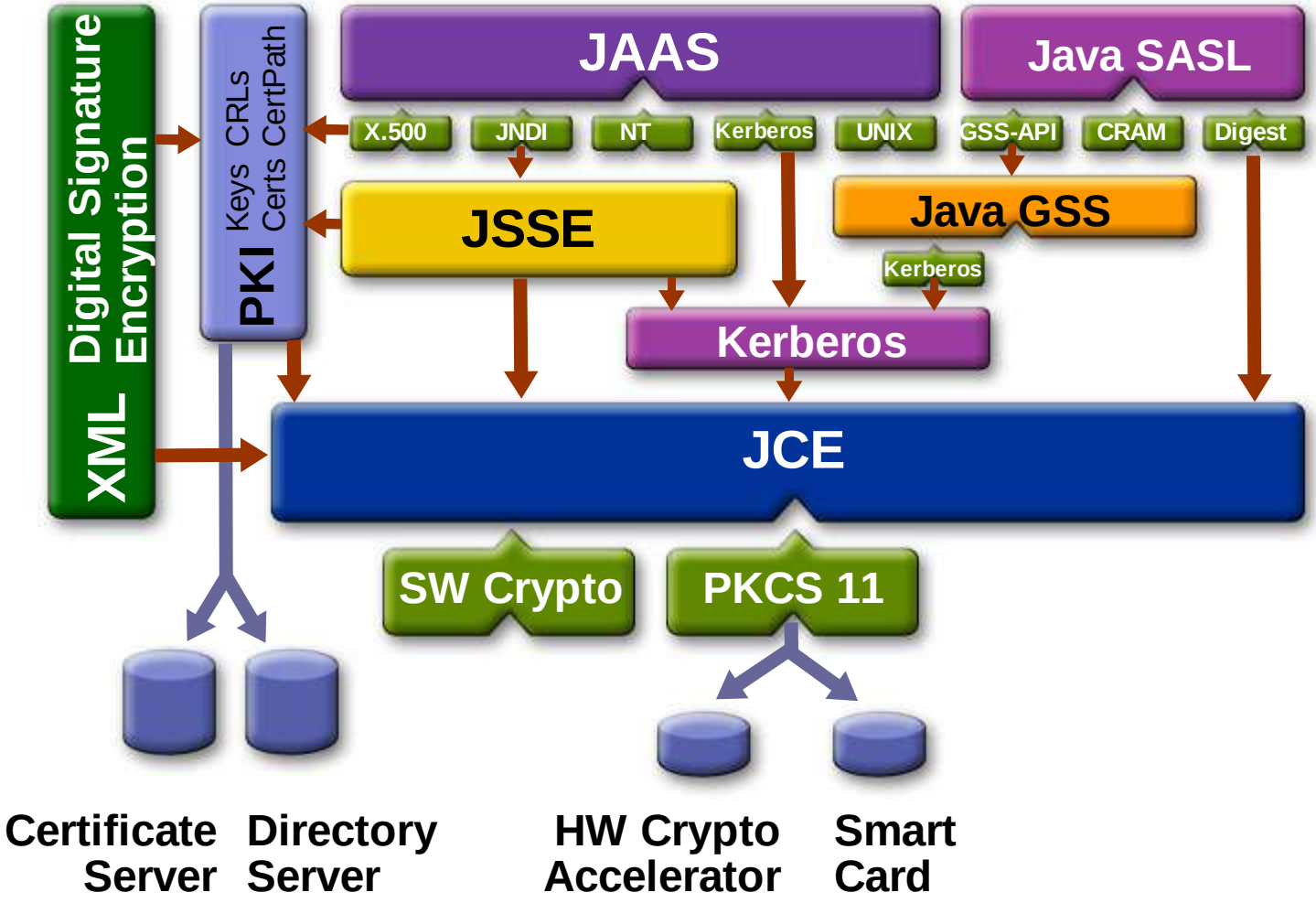
The Big Picture



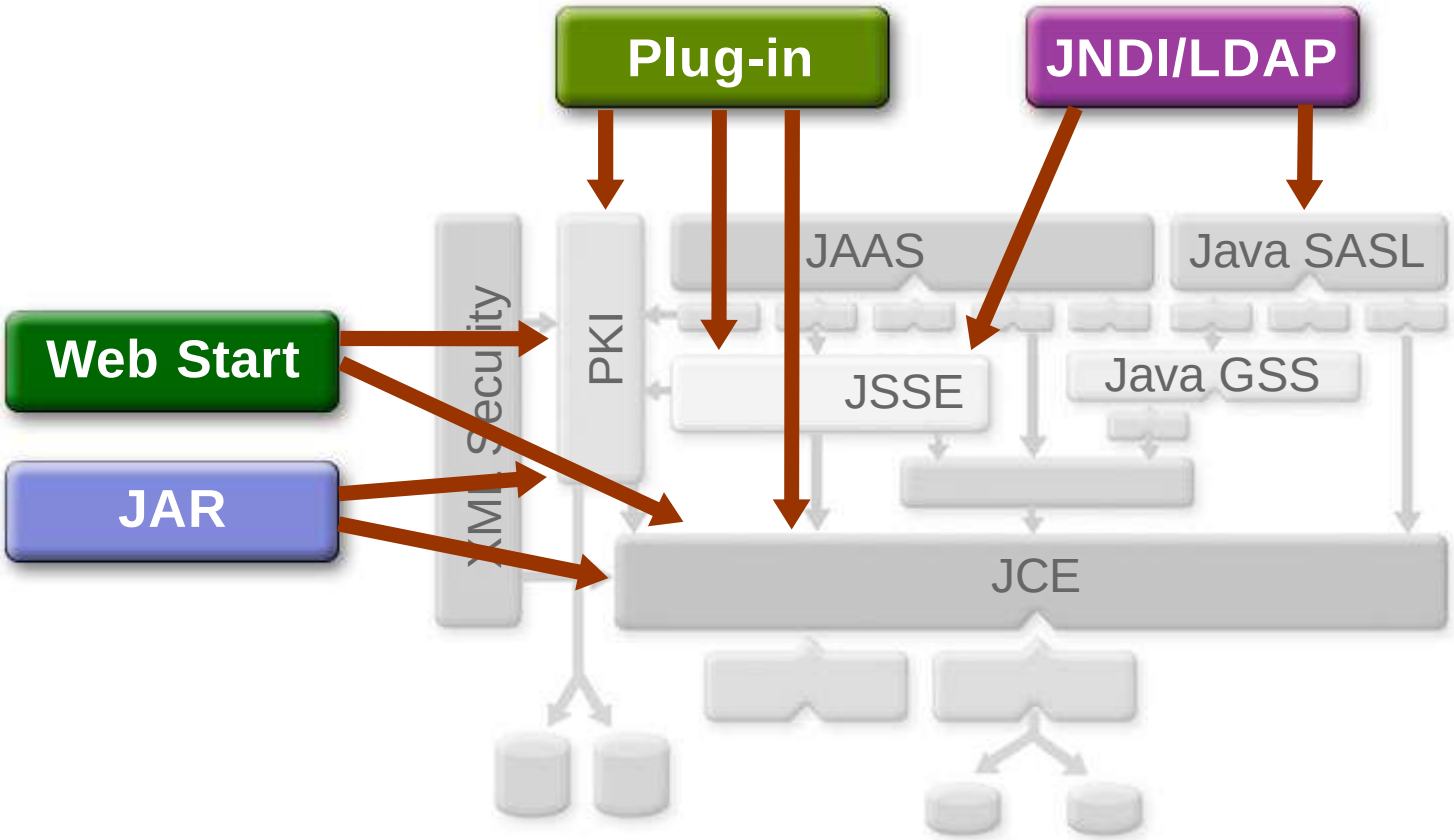
The Big Picture



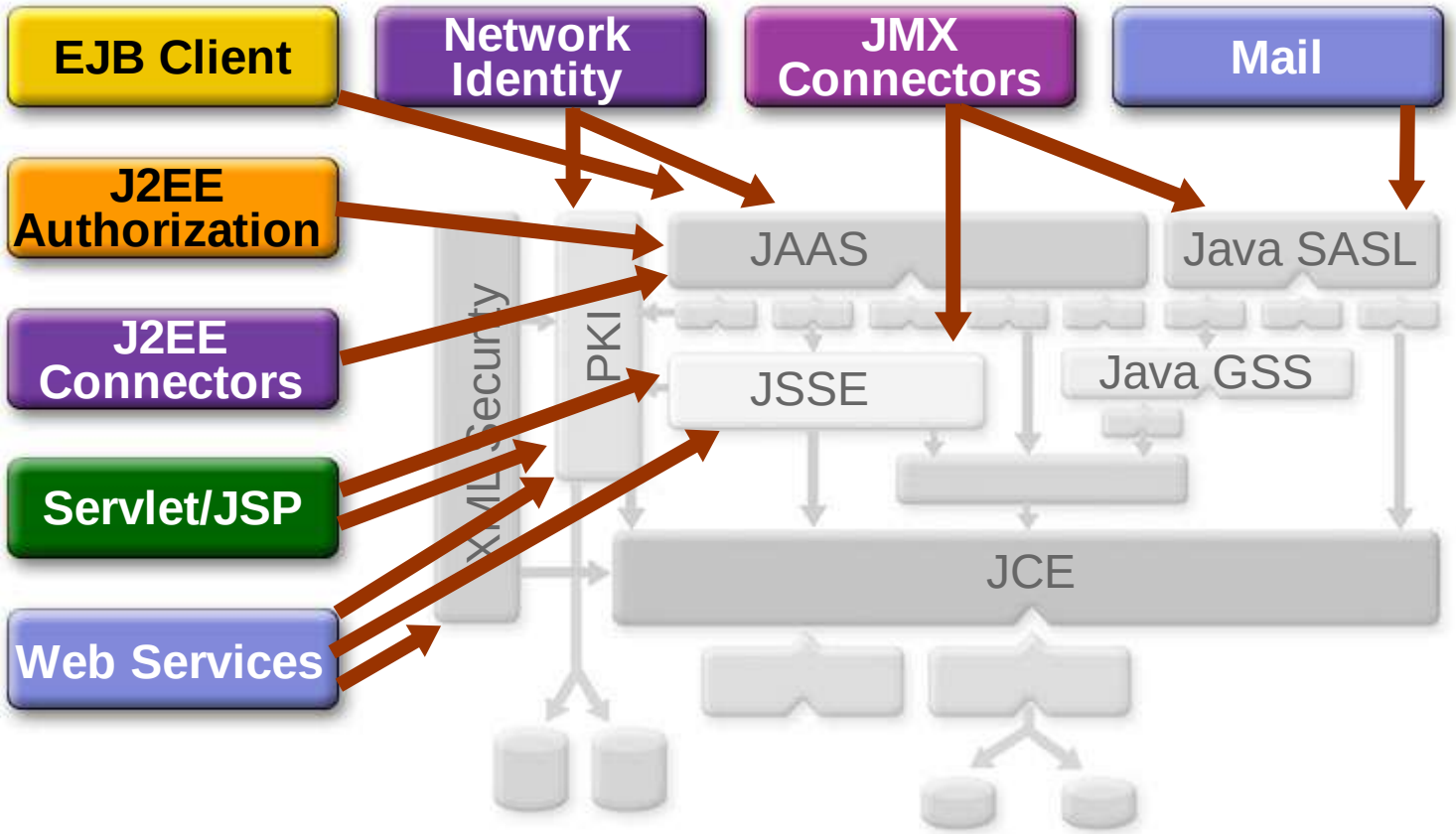
The Big Picture



How Security Is Used in the JRE



How Security Is Used in Enterprise Applications



Beyond J2SE™ 1.5: Secure Communications

- Java™ GSS API
  - Pluggability
  - SPNego and PKInit mechanisms
  - SPKM and LipKey mechanisms
- JSSE
  - Track the IETF TLS standard
  - JAAS-aware key manager

Beyond J2SE™ 1.5: Secure Authentication and Authorization

- Enhanced role/group support
- XACML policy provider
- LDAP policy provider
- Signed policy information
- Policy in JAR

Beyond J2SE™ 1.5: PKI

- PKCS (JSR 74)
  - PKCS 7 Cryptographic Message Syntax
  - PKCS 9 Selected Attribute Types
  - PKCS 10 Certificate Request Syntax
  - PKCS 12 Personal Information Exchange
- XML Trust Service APIs (JSR 104)
- PKIX delegated path discovery and validation

Beyond J2SE™ 1.5: Miscellaneous

- Ease of management and use
  - Enhance deployment options
  - Better integration with operating environment
  - Better tools
- ECC provider
- JAR enhancements

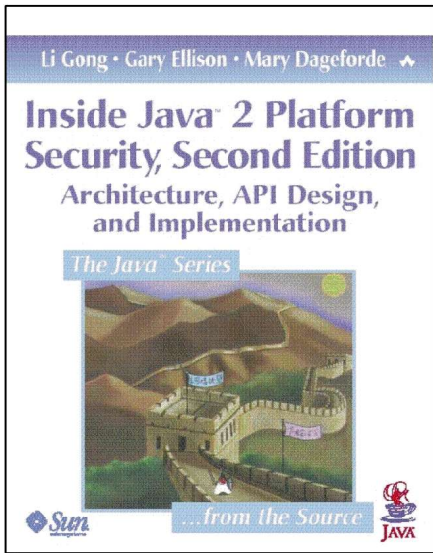
Summary

- J2SE™ platform 1.4 offers a secure foundation for building secure applications and services
- J2SE™ platform 1.5 will make major enhancements in the security area
- More security enhancements to come: give us feedback

More Information

- Web site
  - java.sun.com/security
- JSR
  - www.jcp.org/jsr/overview
- Contacts
  - java-security@sun.com for feedback
  - www.sun.com/developer/support for support

New Edition of Java™ 2 Platform Security Book



- Inside Java 2 Platform Security: Architecture, API Design, and Implementation, Second Edition
  - By Li Gong, Gary Ellison, and Mary Dageforde
- java.sun.com/docs/books

Security BOFs

- BOF-2269: Meet the J2SE™ Security Engineering Team
- BOF-2236: Implementing Security via the JAAS and the Java™ GSS API
- BOF-2214: XML Digital Signature and Encryption APIs (JSRs 105 and 106)
- BOF-2226: JSSE™ Software: Tips, Tricks, and Q&A

Q&A

JavaOne℠

Sun's 2003 Worldwide Java Developer Conference℠

java.sun.com/javaone/sf