Challenges of Wireless Security

April 2002

Paul Fahn
Director, Sales and Business Development
Certicom Corp.
Cryptographic Algorithms

• Symmetric Key
  – DES, 3DES, RC4, AES, …
  – Encryption, decryption
  – Fast and “lightweight”

• Public Key
  – RSA, ECC, DH, DSA
  – Digital signatures, verification
  – Key management
  – Computation intensive
Crypto bottlenecks

• Security protocols start with a few public key operations, followed by many private key
  – SSL, IPsec

• Public key operations are the bottleneck

• Problem: Key sizes grow over time
  – RSA: 512 → 1024 → 2048
Security in the “Wired” World

Based on three assumptions:

1. Client has reasonable processing capabilities
2. Client has reasonable bandwidth capabilities
3. Client has reasonable storage capabilities

These do not hold in many wireless environments
Wireless: New Constraints

- Limited processing capabilities
- Bandwidth is limited
- High latency networks
- Storage capabilities are limited

Challenge:

New solutions could introduce interoperability problem with legacy systems
Wireless Constraint: Limited Processing and Power

Algorithm Choice

- RSA
  - signature and key generation very heavy
  - burdens grow as future key size increases
  - ok for server authentication

- ECC
  - more efficient for signatures and key generation
  - local key generation for non-repudiation
  - encourages use of client authentication and m-commerce

- AES
  - efficient new symmetric algorithm
  - large key size also supports increased security requirements
Wireless Constraint: Bandwidth and Latency

• Latency: minimize the number of messages
  – less “chatty” protocols. Ex: WTLS vs. SSL
  – use WTLS class III handshake as certificate request
  – short-lived certificates vs. cert validation

• Bandwidth: minimize the size of the messages
  – non-ASN.1 certificates, cert requests, etc.
  – do not send certificate chains
  Certificate URL’s
Wireless Constraint: Storage on Client Device

- Cell phones, PDA’s, SIM cards
- Minimize the code needed on the client devices
  - Platform-optimized code
  - Non-ASN.1 formats
- Reduce the size of the keys, certificates stored on the clients
  - ECC: smaller key sizes
  - WTLS/WAP certificates
  - Certificate URL’s
Elements of Traditional PKI

Message Formats
- Abstract Syntax Notation (ASN.1)
- Public Key Cryptography Standard (PKCS) 10, Certificate Management Protocol (CMP), etc.
- X.509 certificates

• Underlying Transport
  - HTTP
  - TCP – “reliable” transport
WAP PKI Server Solution

• Do not reinvent
  – Leverage off existing standards and infrastructure

• New component: “PKI Portal”
  – Sits between wireless client and existing PKI infrastructure
  – Converts the Wireless PKI protocol to wired protocols
  – Returns “certificate URL” instead of certificate
  – Support for WTLS server certificates
Wireless PKI Architecture with PKI Portal
Wireless Java: J2ME

Problems:

– Lack of security standards
– Performance issues of pure java code
  • Public key crypto algorithms
– Lack of standardized interface to native code
  • Portability threatened

Put crypto inside the KVM. API?
Questions?

Paul Fahn
Certicom Corp