RSA
The Security Division of EMC

Key Encapsulation: A New Scheme for Public-Key Encryption

XML Security Working Group F2F, May 2009

Summary

- Most specifications of public-key encryption follow the original “encrypt/decrypt” model from 25 years ago
- New model is emerging, based on work of Shoup and others: *key encapsulation*, with better flexibility and security proofs
- Recommend transition to new model:
  - Introduction of key encapsulation into XML Encryption v1.1.

The Security Division of EMC

Original Approach

- Bob has public key / private key pair
- Alice encrypts message *M* with Bob's public key to produce a ciphertext *C*:

$$C = \mathbf{E}(\ PubKey_B,\ M\ )$$

- Bob decrypts *C* with his private key:

$$M = \mathbf{D}(\ PrivKey_B,\ C\ )$$

Limitations

- *Message length:* Length of *M* may be limited
- *Malleability:* Encryption may not protect message integrity
- *Mathematical properties:* Encryption of related messages may be related
- *Modeling:* DH (ECDH) doesn't fit well

Traditional Remedies

- Typically, some message padding is applied to address these limitations, but current approaches for RSA are less than ideal:
  - PKCS #1 v1.5 padding is *ad hoc*, doesn't provide integrity
  - OAEP provides integrity and is provably secure, but bounds aren't tight (e.g. knowledge of plaintext in RSA-OAEP reveals input to RSAEP; this is not the case with RSA-KEM)
- Message length is still bounded, and DH needs its own method

New Remedy: Two Layers

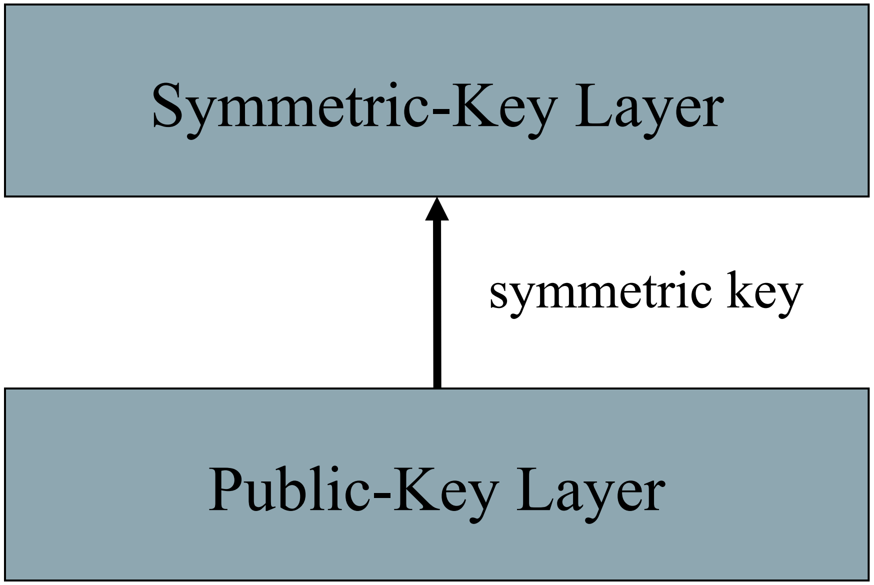
- *Public-key layer* establishes a random symmetric key
- *Symmetric-key layer* protects data with the established symmetric key and symmetric algorithm
  - data can be of any length
- Layers are independent

Addressing the Limitations

- *Modeling:* DH, RSA, other PKC all fit
- *Message length:* Length of *M* not limited
- *Malleability:* Symmetric method can provide integrity protection
- *Mathematical properties:* Symmetric keys are unrelated; symmetric method avoids mathematical properties

Don't We Do This Already?

- Many specifications (including S/MIME) have two layers:
  - message encrypted with symmetric key
  - symmetric key encrypted with RSA public key
- But the symmetric key is generated first *then* encrypted; more than needed, and results in a looser (or no) proof of security



RSA
The Security Division of EMC

Public-Key Layer: Key Encapsulation

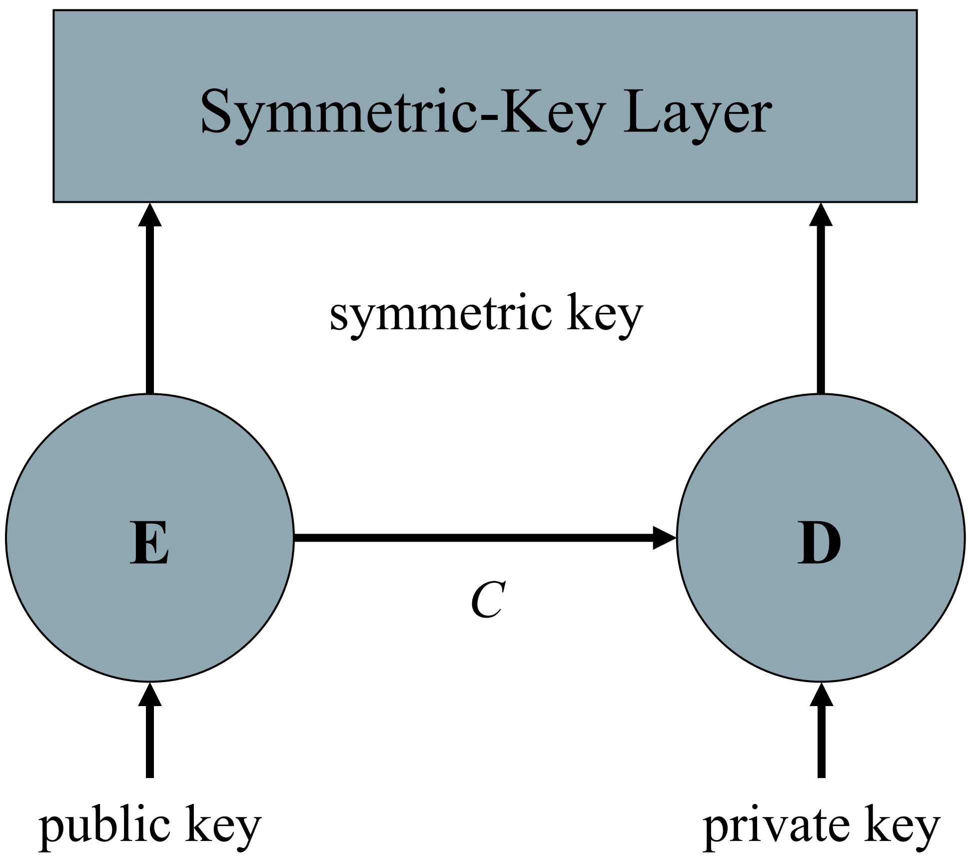
- *Encryption:* Alice generates a symmetric key W and a ciphertext C that "encapsulates" W:

$$(C, W) = \mathbf{E}(\ PubKey_{\mathrm{B}}\ )$$

- *Decryption:* Bob regenerates W from C:

$$W = \mathbf{D}(\ PrivKey_{\mathrm{B}},\ C\ )$$

Two Layers with Key Encapsulation



Encapsulation Using RSA

- Encrypt with public key (n, e):
  - $r \leftarrow_R [0, n\text{-}1]$
  - $C_0 \leftarrow r^e \bmod n$
  - $W \leftarrow \mathrm{KDF}(r)$
- Decrypt with private key (n, d)
  - $r \leftarrow C_0{}^d \bmod n$
  - $W \leftarrow \mathrm{KDF}(r)$

Key Transport Using KEM

1. Generate a random integer z ($0 \leq z \leq n-1$)

   $$z = \text{RandomInteger}\ (0, n-1)$$

2. Encrypt the random integer z using the recipient's public key (n,e)

   $$c = z^e \bmod n$$

3. Derive a key-encrypting key *KEK* of length *kekLen* bytes from z using the underlying key derivation function

   $$KEK = \text{KDF}\ (z, kekLen)$$

4. Wrap the keying data K with the key-encrypting key *KEK* using the underlying key-wrapping scheme to obtain wrapped keying data *WK*

   $$WK = \text{Wrap}\ (KEK, K)$$

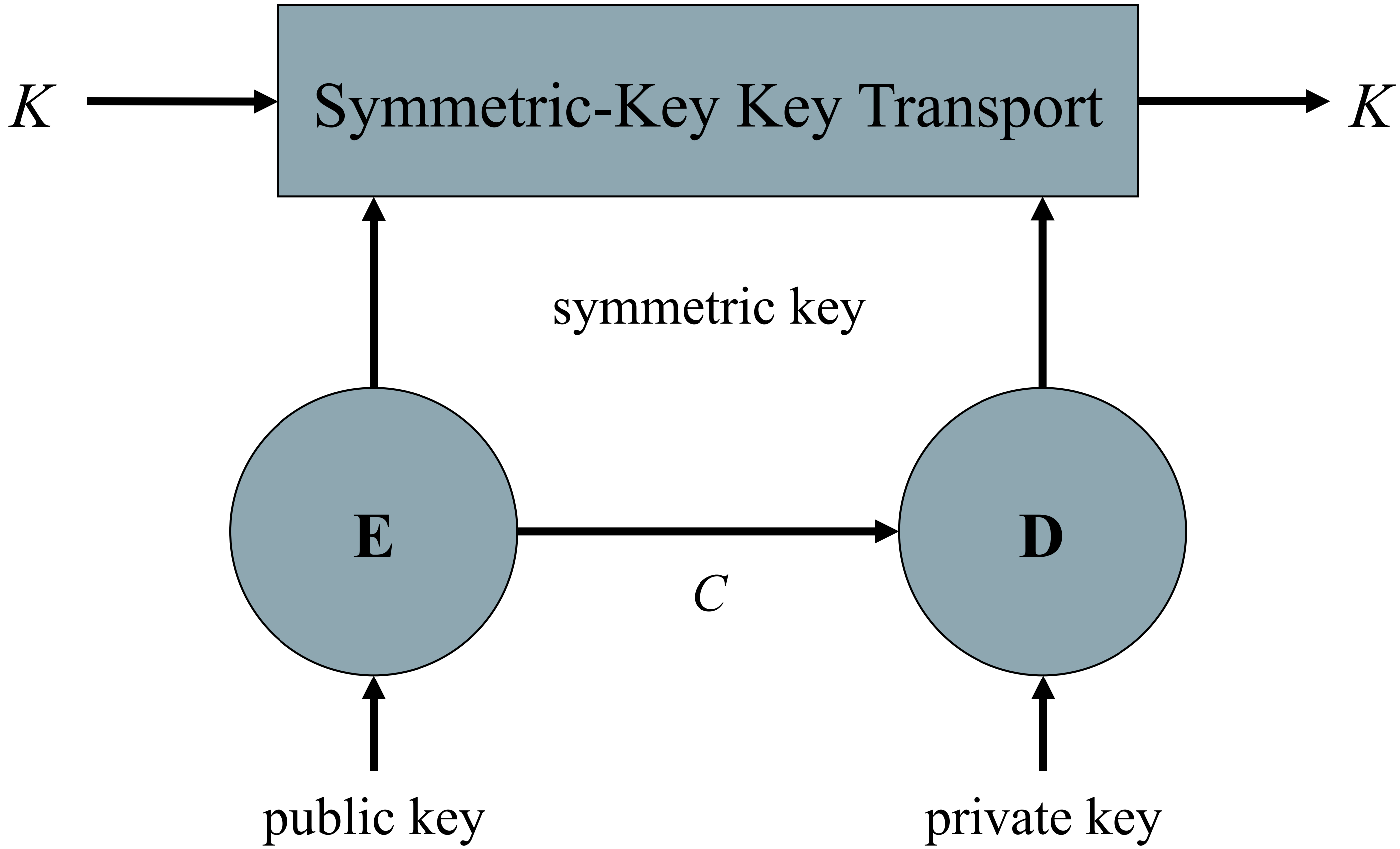
5. Concatenate the ciphertext C and the wrapped keying data *WK* to obtain the encrypted keying data *EK*

   $$EK = C \mathbin{||} WK$$

6. Output the encrypted keying data *EK*

The Security Division of EMC

Key Transport in Two Layers
(similar for message encryption)



Key Encapsulation in Standards

Standard	Status
ANSI X9F1 (X9.63, X9.44 draft)	✓
IEEE P1363 (P1363a draft, P1363b)	Proposed
ISO/IEC 18033-2 (draft)	✓
PKCS #11	Being proposed
XML Encryption	Proposed here…
S/MIME	*In WG last-call*

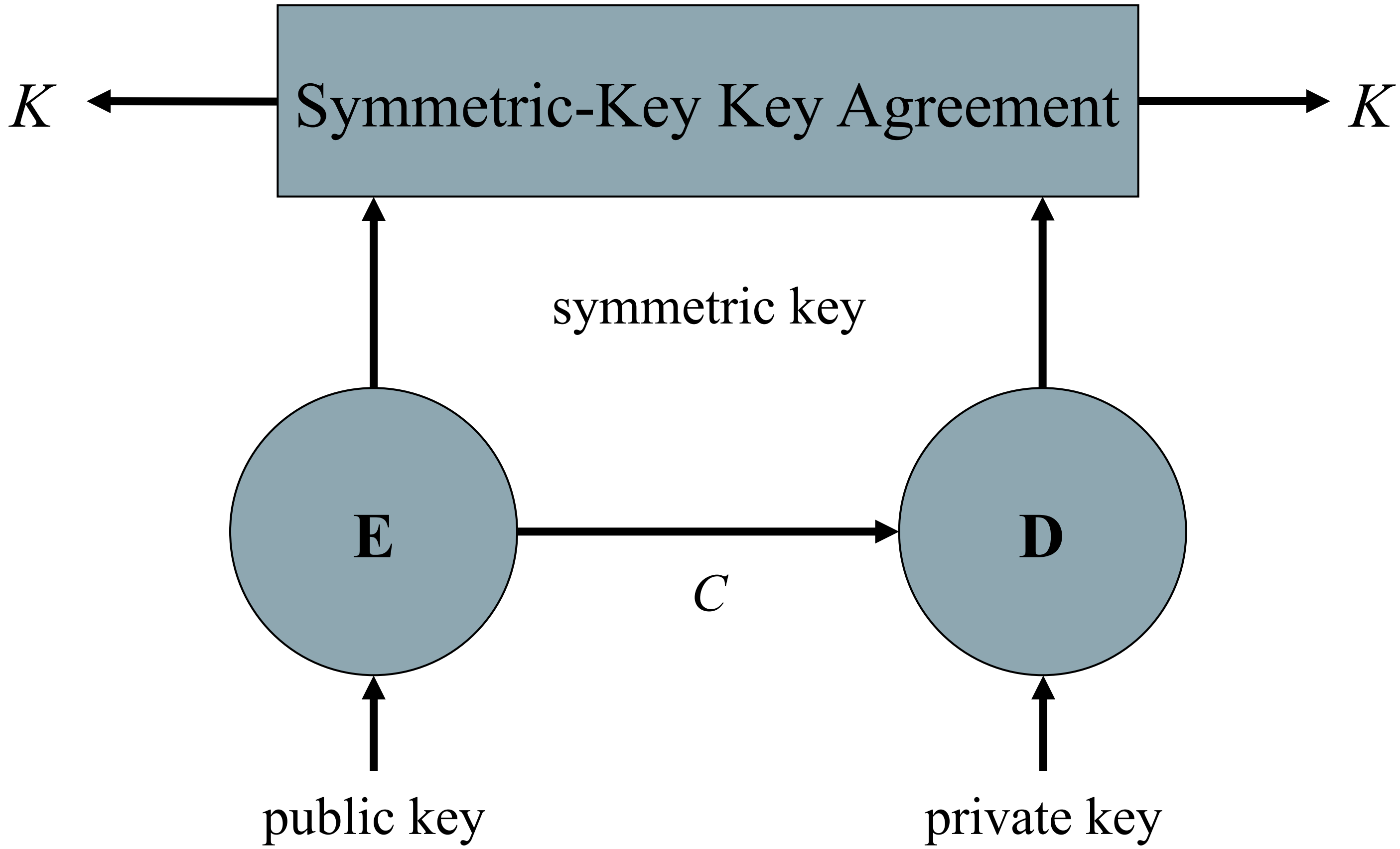
Conclusions & Proposal

- Key encapsulation is a convenient way of positioning public-key cryptography
- Specific suggestion for XMLSec: Include KEM as new key transport method in XMLEnc 2.0 (?)
  - RSA-KEM, ECDH-KEM
- Will entail: Defining schema for defining key encapsulation method (RSA-KEM, ECDH-KEM), key derivation function, key length and key wrapping scheme

Related Research & Information

- Zheng-Seberry, Bellare-Rogaway proposed RSA-based schemes with two layers (early 1990s)
- Shoup: KEM for ISO proposal (2001)
- Handschuh *et al.*: GEM (2002)
- http://www.rsa.com/rsalabs/node.asp?id=3D2147
- Jakob Jonsson's paper comparing security bounds of OAEP and KEM:
  - http://eprint.iacr.org/2002/034.pdf

Key Agreement in Two Layers
(one key-pair case)



RSA
The Security Division of EMC

Encapsulation Using DH

- Encrypt with public key (p, q, g, y):
  - $r \leftarrow_R [1, q\text{-}1]$
  - $C_0 \leftarrow g^r \bmod p$
  - $Z \leftarrow y^r \bmod p$
  - $W \leftarrow \mathrm{KDF}(C_0 \mid\mid Z)$
- Decrypt with private key (p, q, g, x)
  - $Z \leftarrow C_0^x \bmod p$
  - $W \leftarrow \mathrm{KDF}(C_0 \mid\mid Z)$