Lightweight Encryption for Email

Ben Adida
ben@mit.edu
7 July 2005

joint work with
Susan Hohenberger and **Ronald L. Rivest**
MIT Cryptography and Information Security Group

Motivation

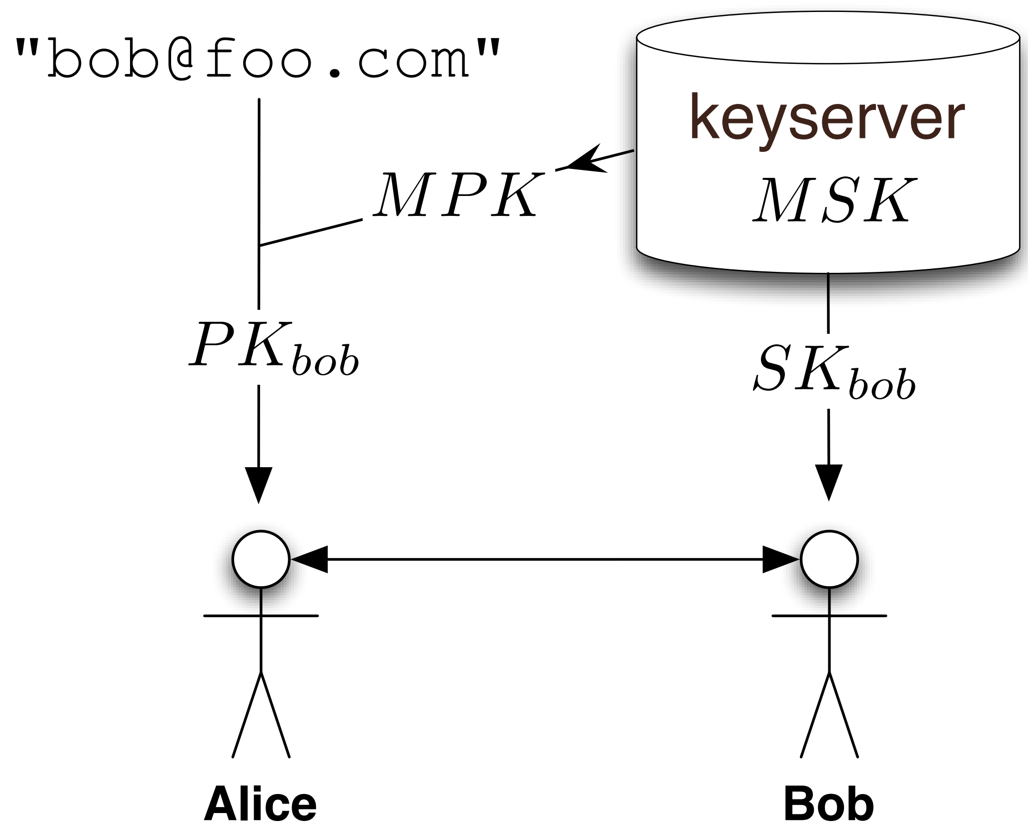
- To Improve/Restore the Usefulness of Email
- Lightweight Trust for Email Signatures [ACHR2005]
- Can we get reasonable encryption from similar simplified key management?

Lightweight Signatures

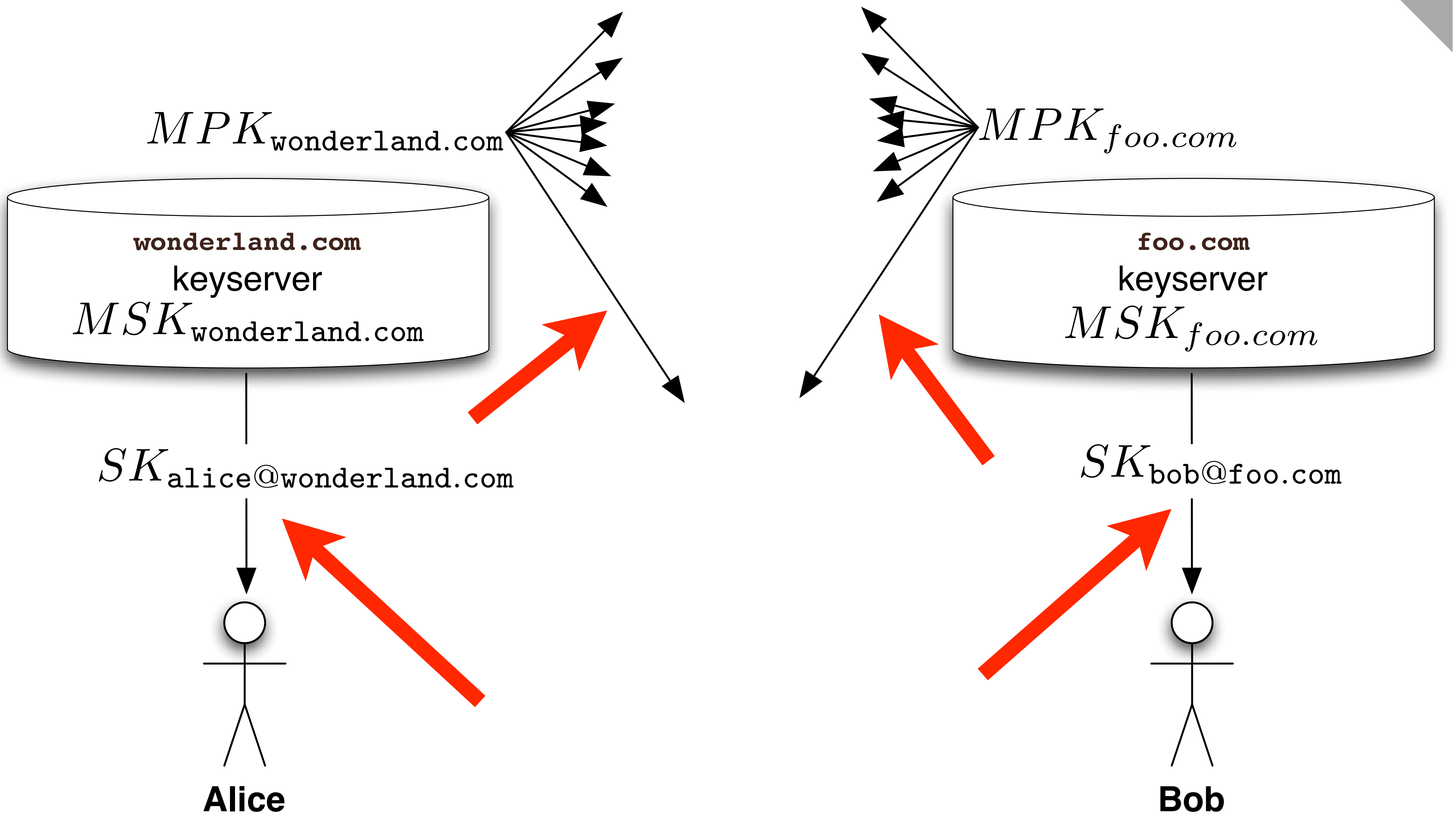
- Makes forging email from bob@foo.com as difficult as receiving Bob's email.
- No explicit user key management
- Uses only existing infrastructure

Review

ID-Based Crypto

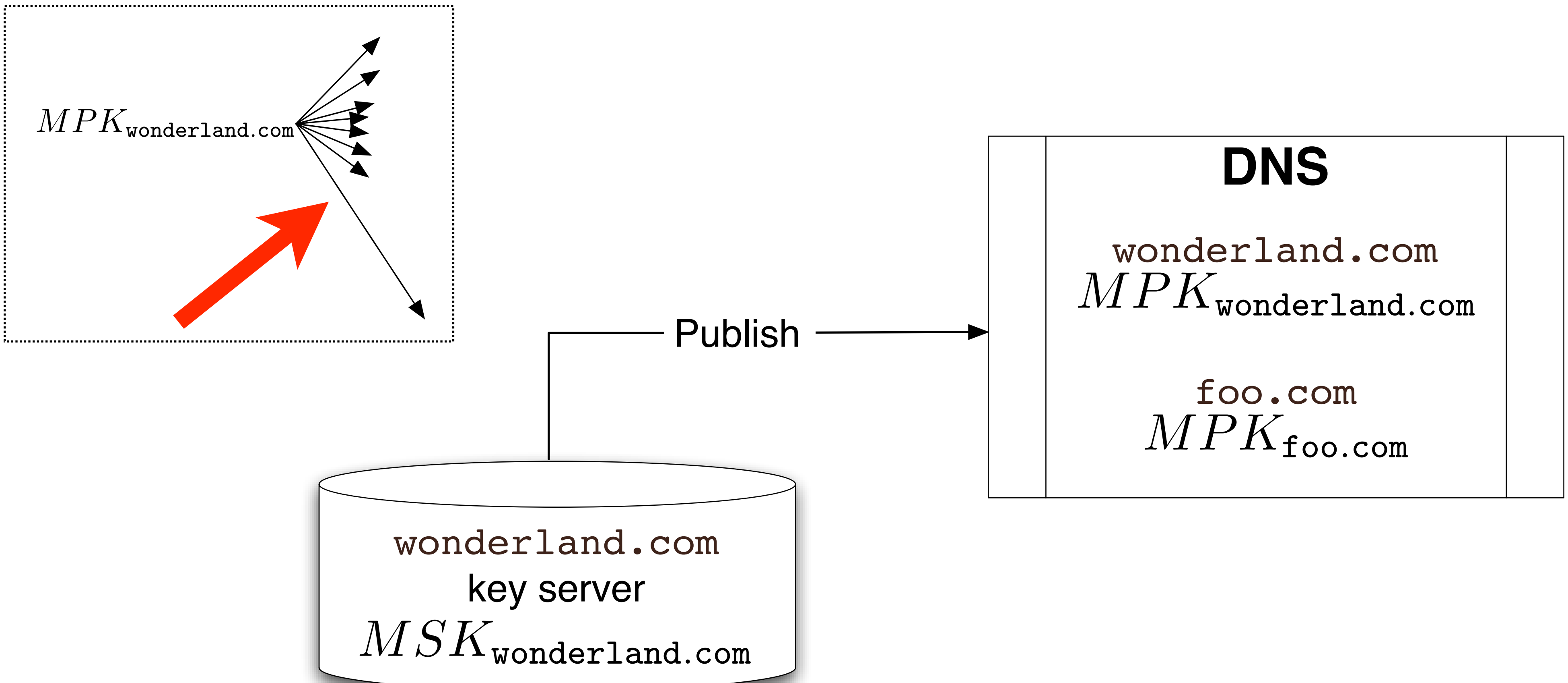


ID-based Domains



DNS to distribute Master Public Keys

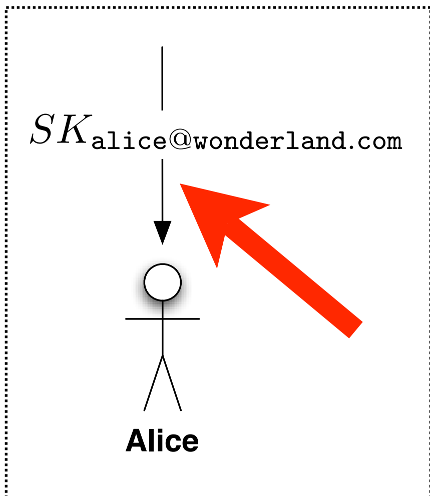
Review

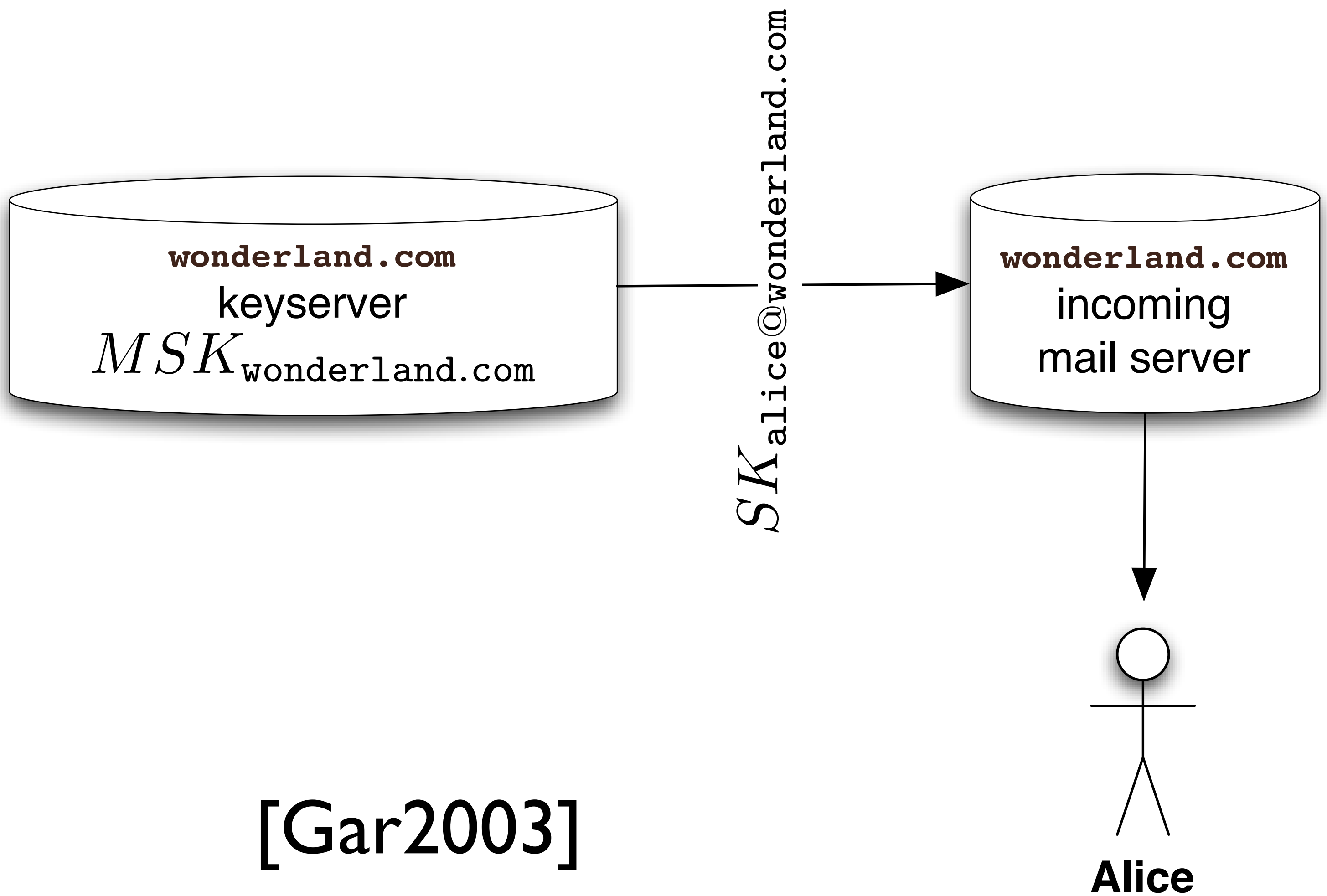


[DomainKeys]

Review

Email-Based Authentication

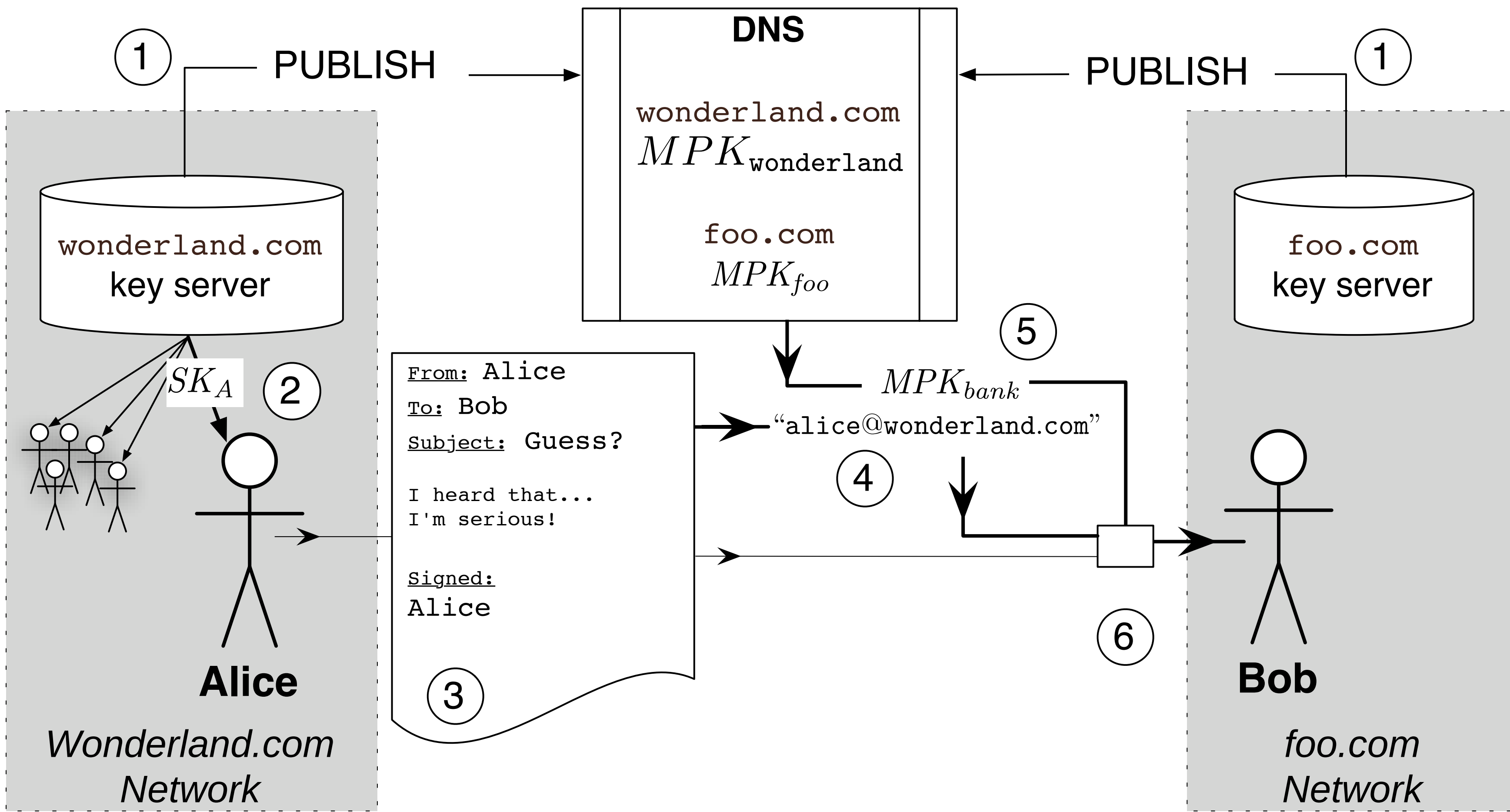




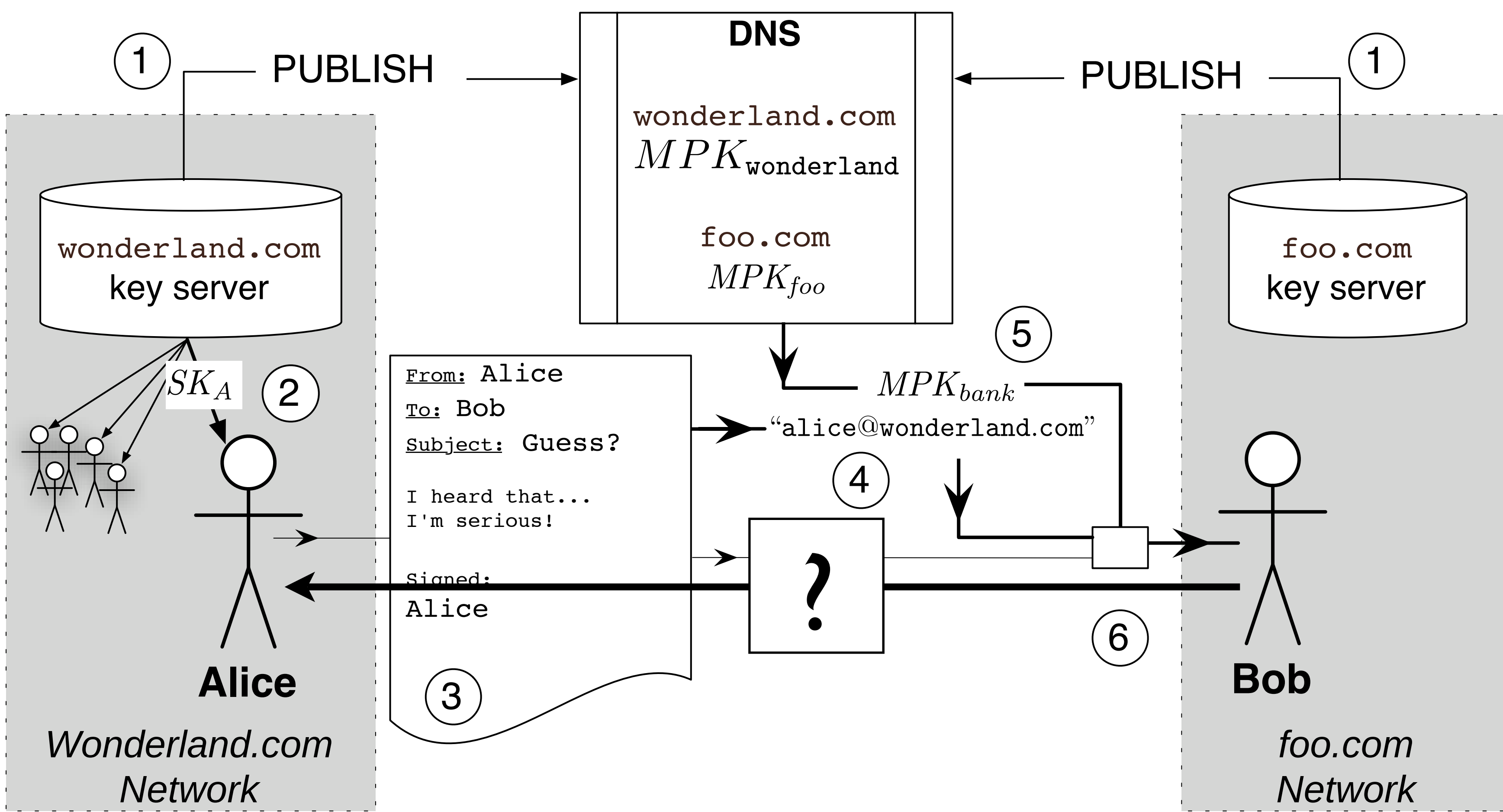
[Gar2003]

Lightweight Sigs

Review



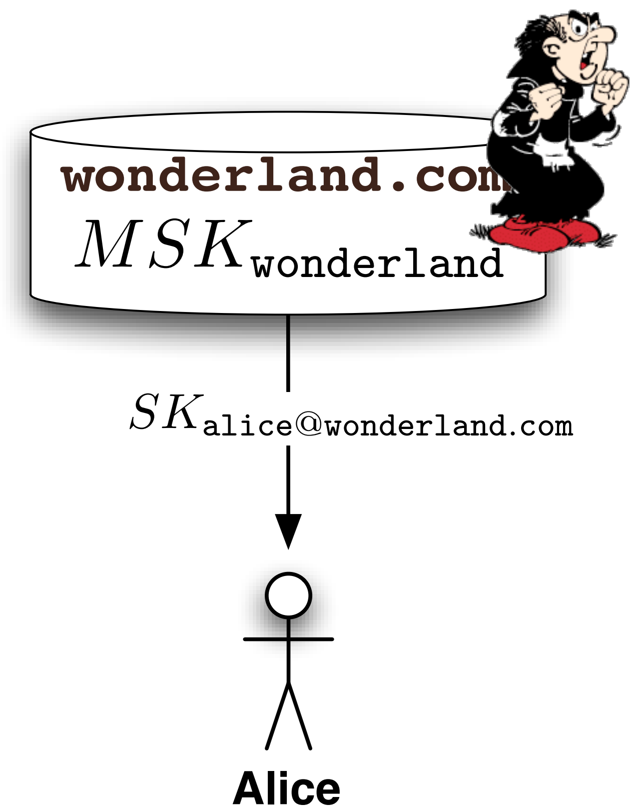
For Encryption?



Threat Model

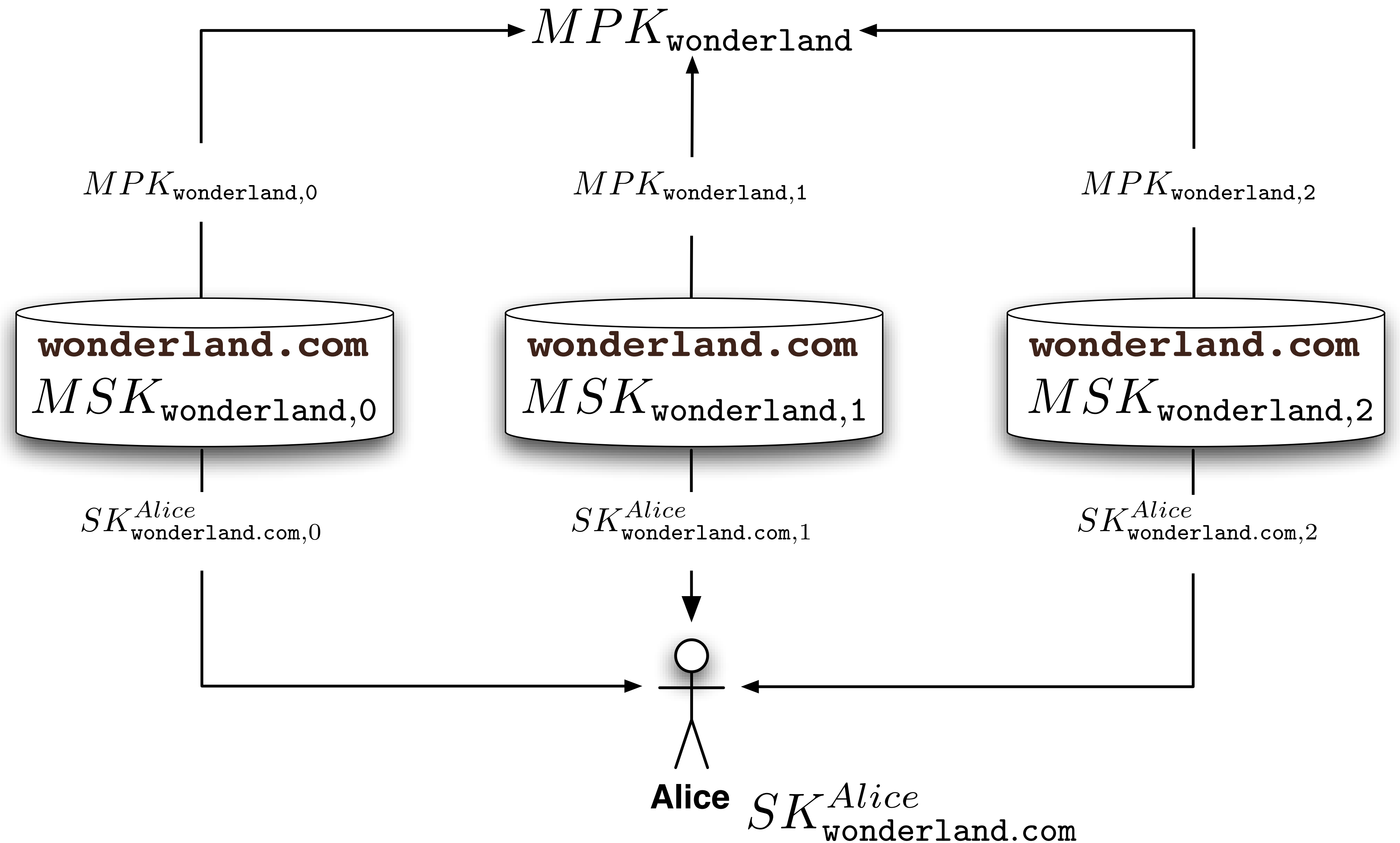
- Assume your incoming mail server won’t actively spoof/attack you.
- **Signatures**
  If the MSK is compromised, simply change the MSK/MPK (DNS updates).
- **Encryption**
  Different story....

Threat #1:
MSK compromise

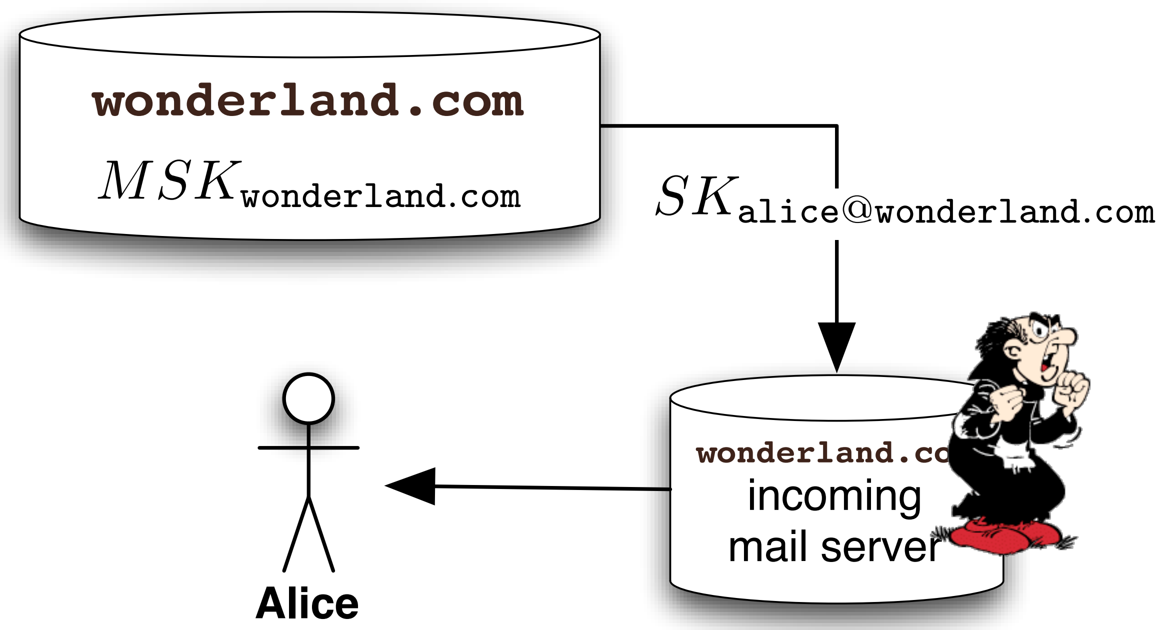


- all **past** encrypted emails are immediately compromised.
- if the MSK compromise is discreet, then all **future** encrypted emails are also compromised. (hacking into a keyserver).

Splitting Keys

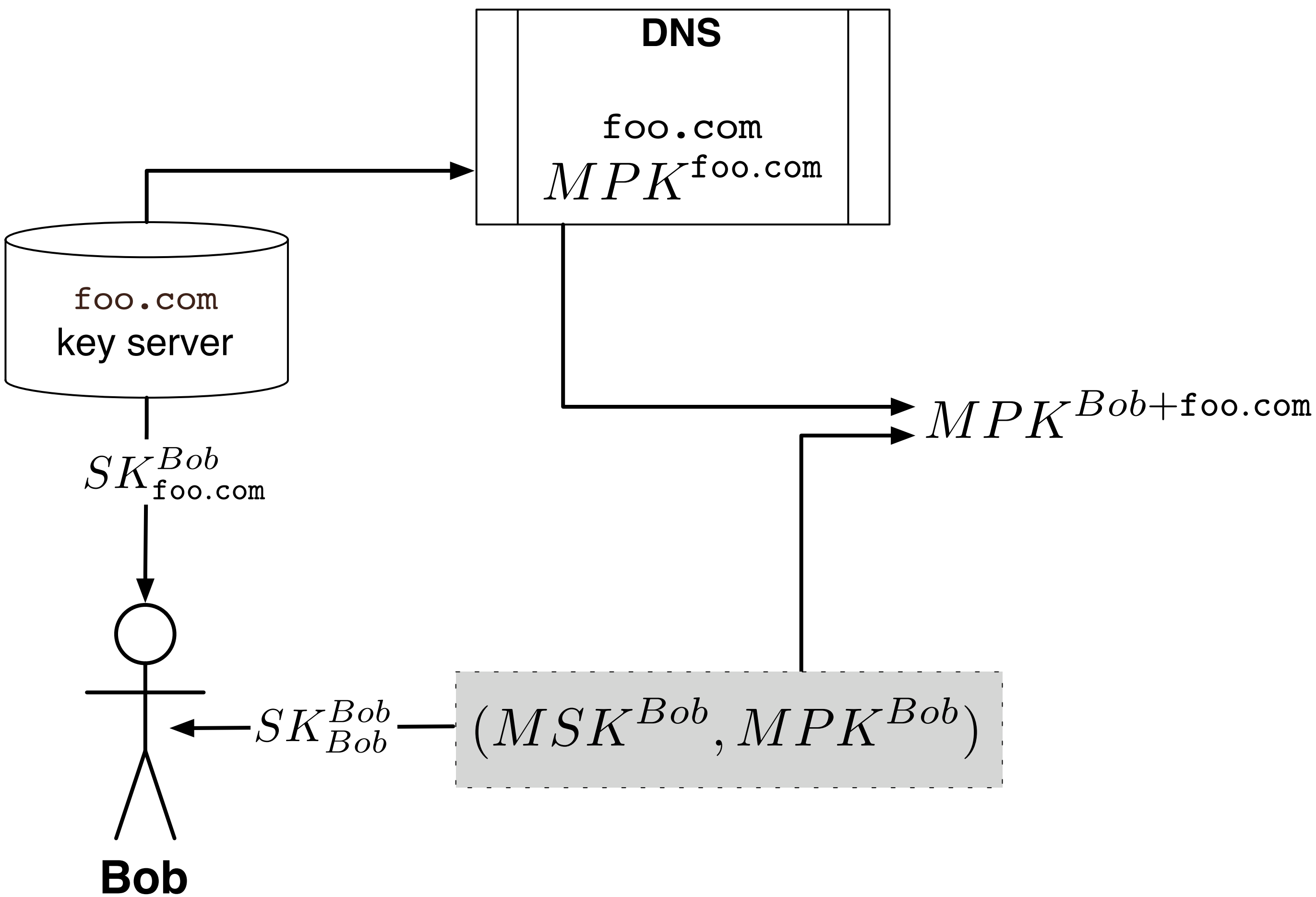


Threat #2: Corrupt Mail Server



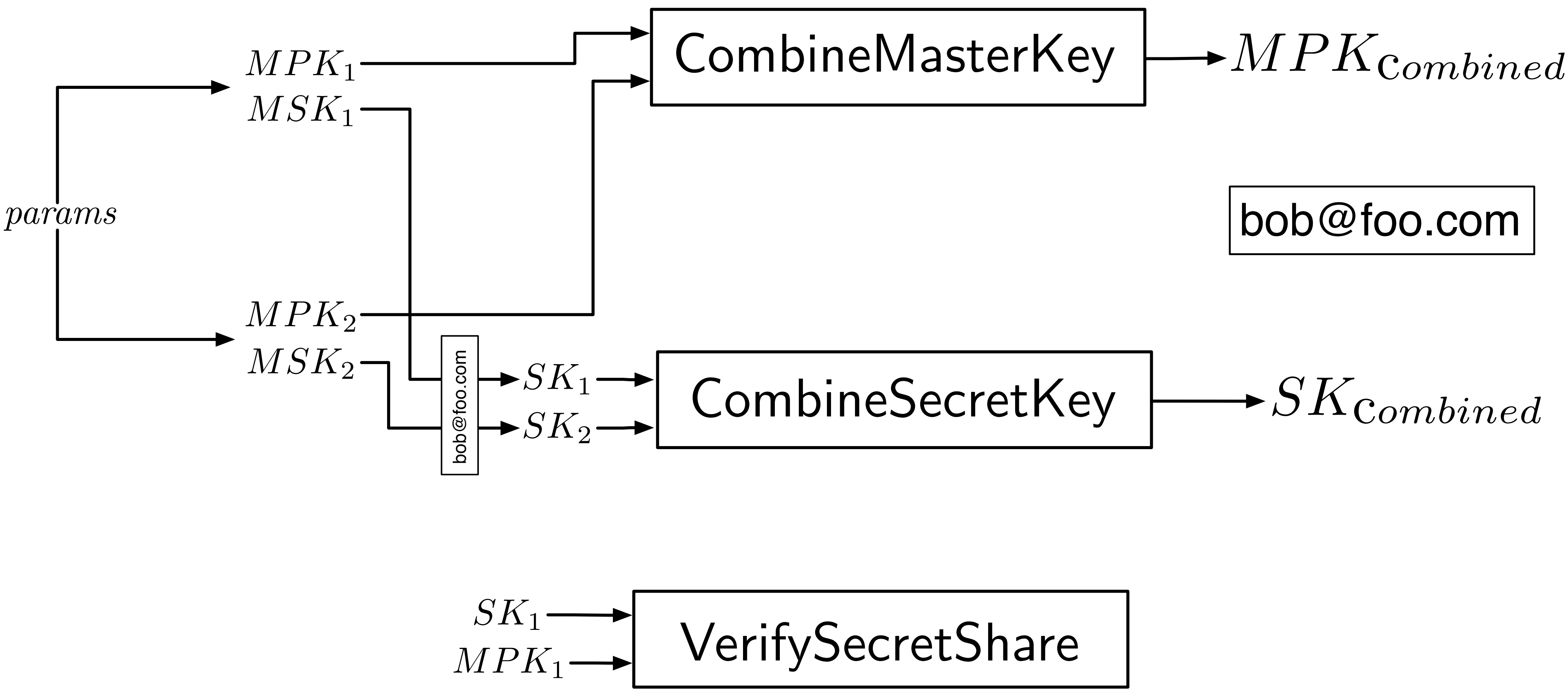
- a corrupt incoming mail server can decrypt and read all secret key material.
- a passive corrupt mail server can intercept all emails.
- even MSK splitting doesn't help.

Recombining Keys



- Bob generates a new MPK/MSK pair
- The combined SK matches the combined MPK.
- The combined MPK provides **certification** and **protection**.
- The second MPK component needs no certification!

Single Core Solution



Building These Features on

Boneh-Franklin and **Waters**

Identity-Based Encryption

Bilinear Maps

Review

$G_1, G_2,$ both of prime order q

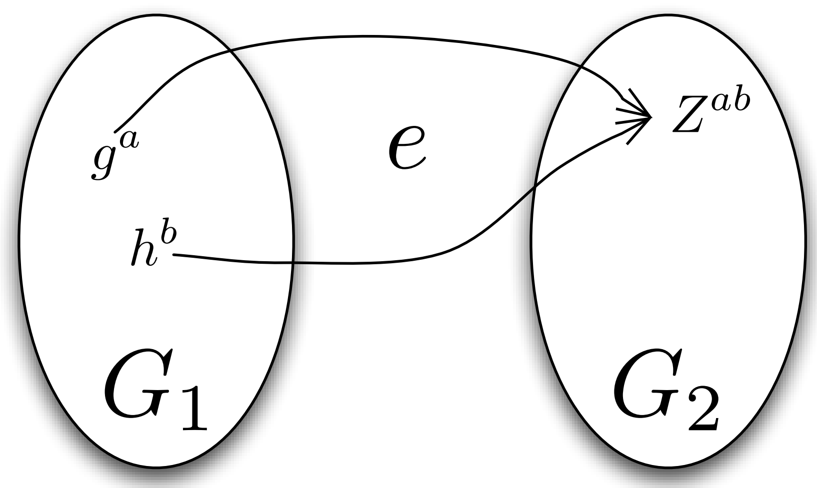
$e : G_1 \times G_1 \to G_2$

g, h generate G_1

$Z = e(g, h)$ generates G_2

$e(g^a, h^b) = e(g, h)^{ab}$

$e(ug, h) = e(u, h)e(g, h)$



Boneh-Franklin Keys

Review

Public Parameters: G_1, G_2, q, g, H

$$MSK = s \in \mathcal{Z}_q$$

$$MPK = g^s \in G_1$$

$$PK_{ID} = H(ID)$$

$$SK_{ID} = H(ID)^s$$

Splitting & Recombining Boneh-Franklin Keys

[BF2000]

$$MSK_1 = s_1 \qquad MSK_2 = s_2$$

$$MPK_1 = g^{s_1} \qquad MPK_2 = g^{s_2}$$

$$SK_1 = H(ID)^{s_1} \qquad SK_2 = H(ID)^{s_2}$$

$\boxed{\text{CombineMasterKey}}$ $$MPK = MPK_1 \cdot MPK_2 = g^{s_1+s_2}$$

$\boxed{\text{CombineSecretKey}}$ $$SK = SK_1 \cdot SK_2 = H(ID)^{s_1+s_2}$$

$$\text{Effective } MSK = s_1 + s_2$$

Waters Keys

Review

Public Parameters: G_1, G_2, q, g, h, F

$$MSK = h^s$$

$$MPK = g^s$$

$$PK_{ID} = F(ID)$$

$$SK_{ID} = (h^s F(ID)^r, g^r)$$

Splitting & Recombining Waters Keys

$$MSK_1 = h^{s_1} \qquad MSK_2 = h^{s_2}$$

$$MPK_1 = g^{s_1} \qquad MPK_2 = g^{s_2}$$

$$SK_1 = (h^{s_1} F(ID)^{r_1}, g^{r_1}) \qquad SK_2 = (h^{s_2} F(ID)^{r_2}, g^{r_2})$$

CombineMasterKey $MPK = MPK_1 \cdot MPK_2 = g^{s_1+s_2}$

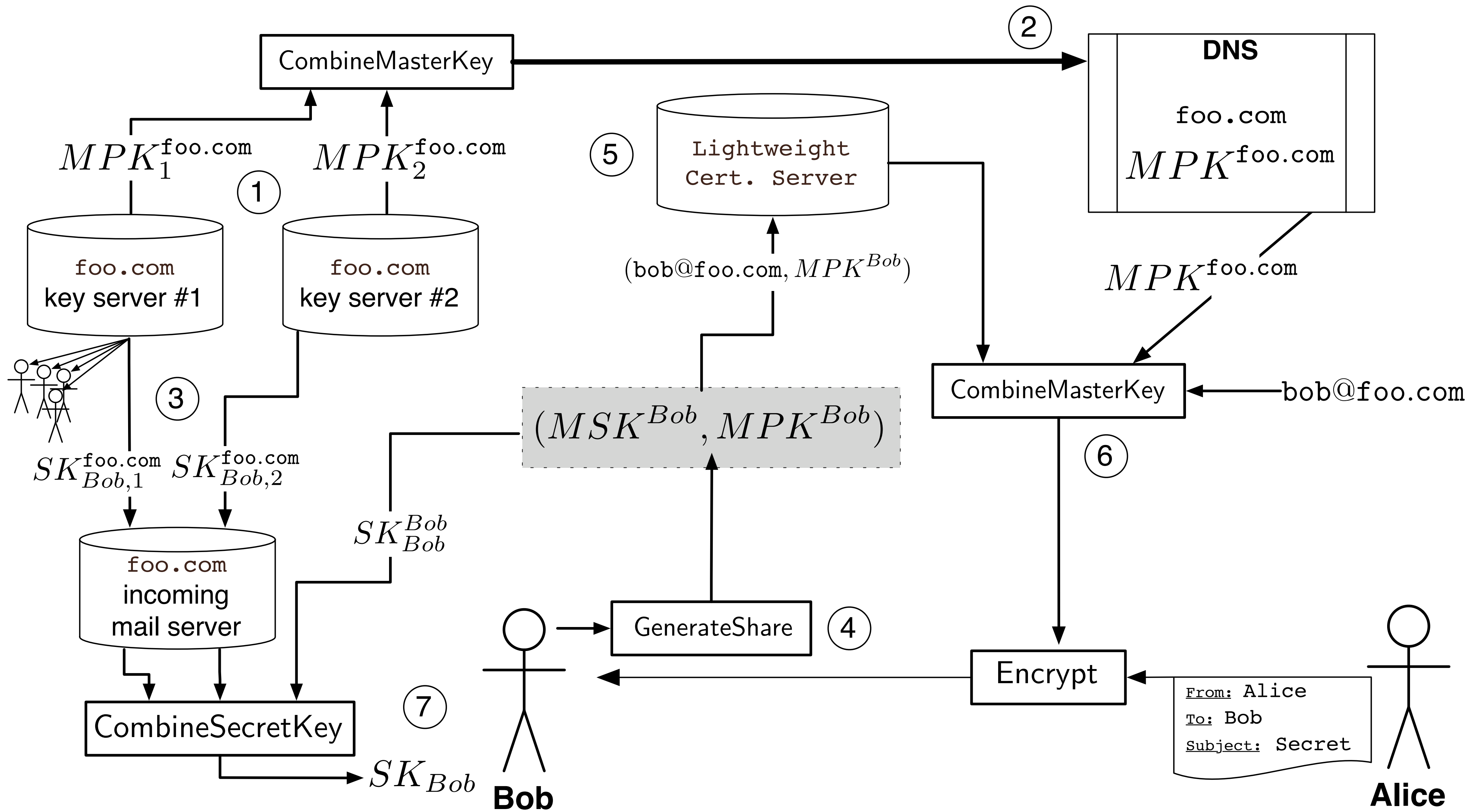
CombineSecretKey
$$\begin{aligned} SK &= (h^{s_1} F(ID)^{r_1} \cdot h^{s_2} F(ID)^{r_2}, g^{r_1} \cdot g^{r_2}) \\ &= (h^{s_1+s_2} F(ID)^{r_1+r_2}, g^{r_1+r_2}) \end{aligned}$$

$$\text{Effective } MSK = g^{s_1+s_2}$$

Additional Details

- **Malicious Share Generation**: NIZK Proof of Knowledge of MSK share
- **Malicious SK Distribution**: k-out-n shares using Lagrange coefficients [GJKR99]

Putting it All Together



Alice's Point of View

- **Finding Bob's Public Key**: automatic: a lookup, a computation against MPK. No trust decision necessary.
- **Decryption Key Management**: automatic, just upgrade the mail client
- **Key Revocation, etc...**: automatic, with upgraded mail client

Automation!

Summary

- Lightweight key infrastructure is not enough for encryption
- To protect against MSK compromise: **key splitting**
- To protect against mail server compromise: **key recombination**
- Both can be accomplished with the same trick on Boneh-Franklin and Waters keys

Questions?

SRUTI '05
Steps to Reducing Unwanted Traffic on the Internet Workshop
USENIX
July 7, 2005 • Cambridge, MA

Backup Slides

Another Solution