PROCESS CONTROL SYSTEMS FORUM

Lightweight Security using Identity-Based Encryption

Guido Appenzeller

Chief Technology Officer
Voltage Security Inc.

Voltage security

Identity-Based Encryption (IBE)

IBE is a new public key encryption algorithm

- A number of widely-used encryption algorithms are already available (RSA, ECC etc.)
- Why should we care about a new one?

1. IBE results in vastly simplified key management, which makes it much more lightweight to deploy
2. It can be used to build security protocols with extremely low overhead
3. Industry is adopting it, as it solves problems with existing encryption solutions

*** Confidential and Proprietary ***

Agenda

1. Identity-Based Encryption (IBE)
   - Key management
   - How IBE works
   - IBE vs. PKI
2. Lightweight Communications with IBE
   - Identity-Based Key Exchange
   - Handshake-free protocols

*** Confidential and Proprietary ***

Agenda

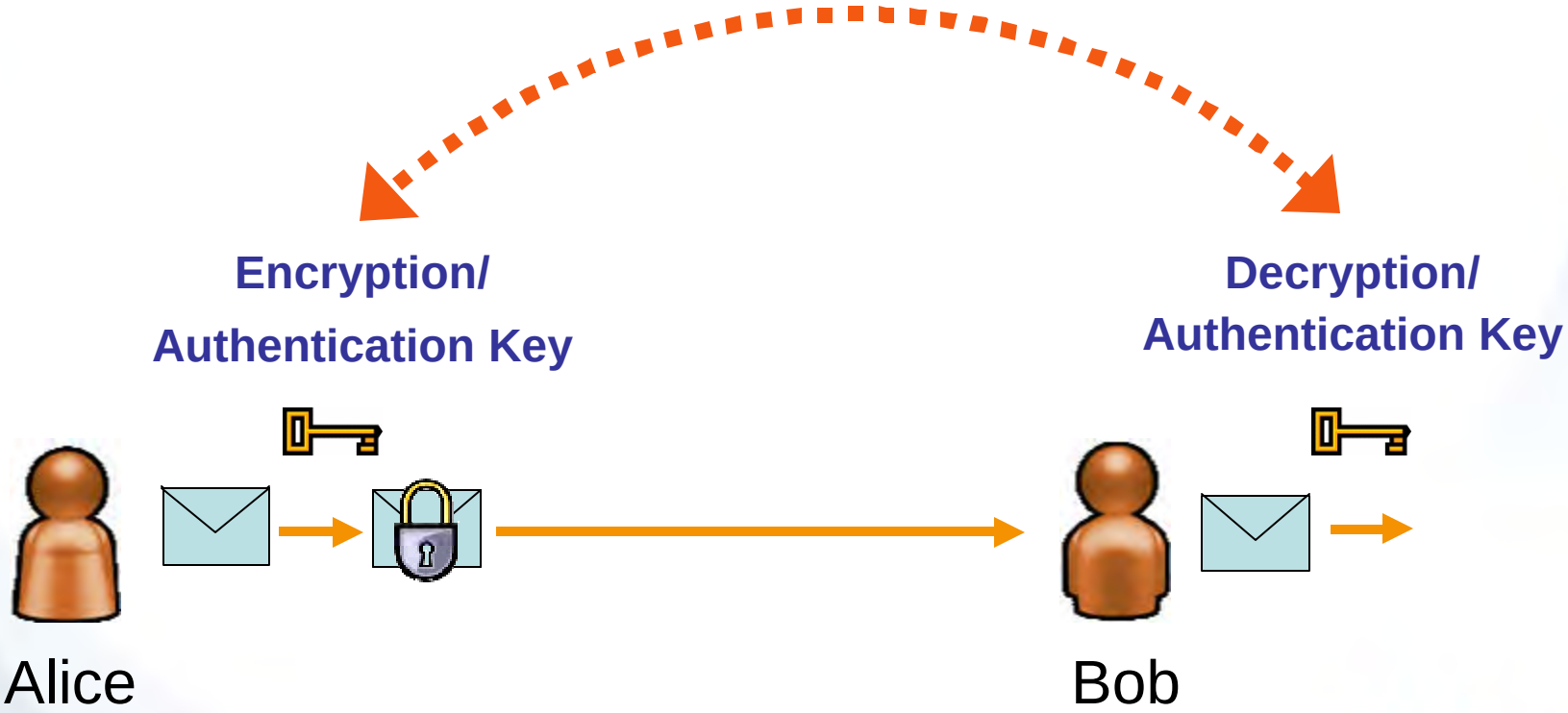
1. Identity-Based Encryption (IBE)
   - Key management
   - How IBE works
   - IBE vs. PKI
2. Lightweight Communications with IBE
   - Identity-Based Key Exchange
   - Handshake-free protocols

*** Confidential and Proprietary ***

Good News: Encryption and Signing is Easy!

How do we make sure

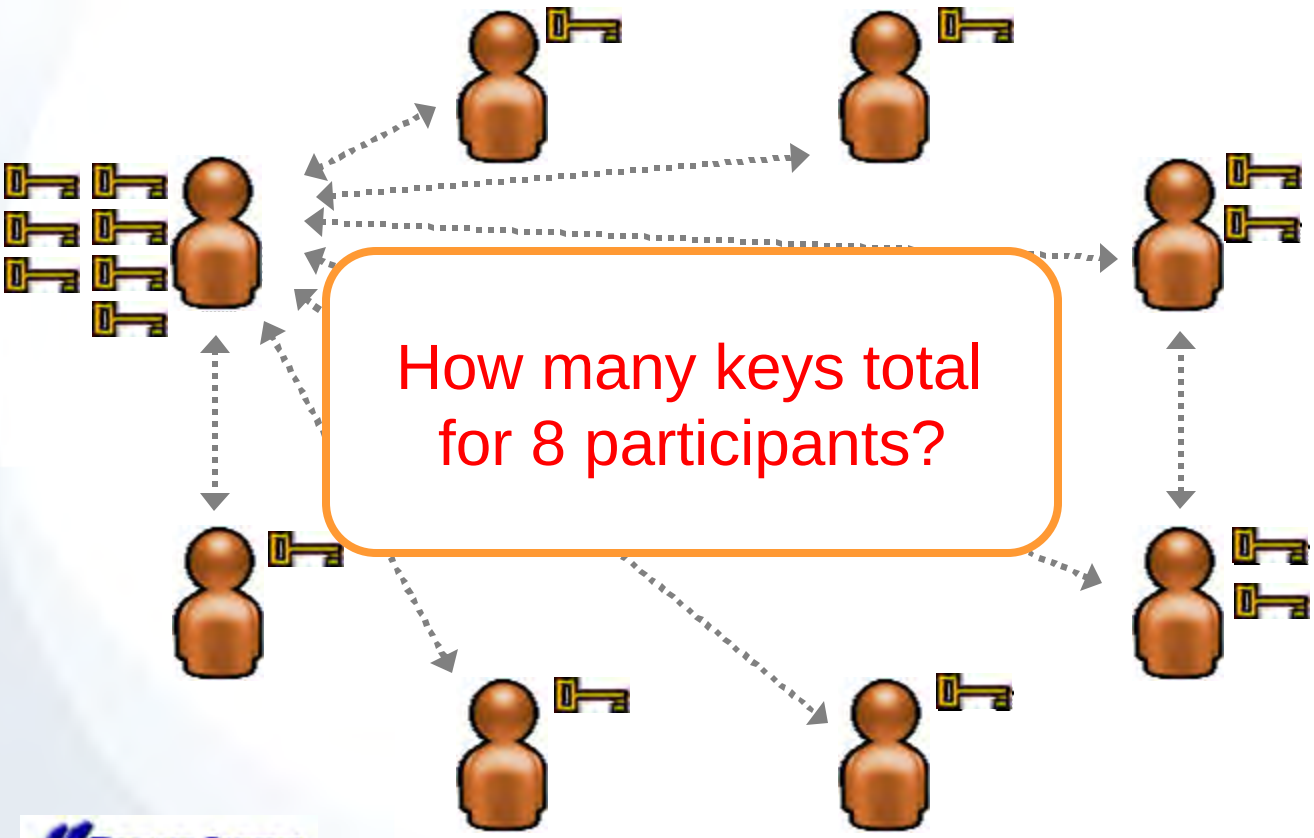
both sides have the right keys?



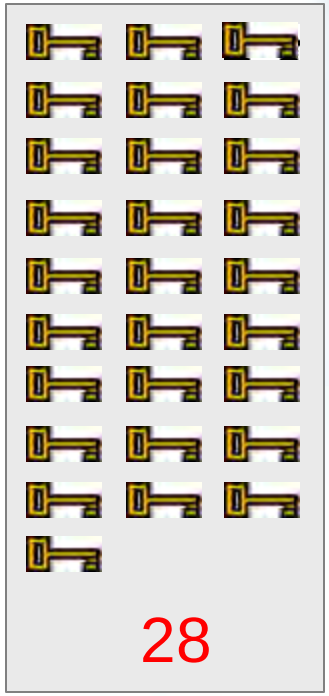
*** Confidential and Proprietary ***

Key Management with Symmetric Keys

- Encryption Key and Decyrption Key are the same



Total Keys



*** Confidential and Proprietary ***

Key Management with Symmetric Keys

- Key Management with Symmetric Keys doesn't scale
  - Network of 8 parties requires managing 28 keys
  - Network of 1000 parties requires 500,000 keys
  - Network of *N* parties requires *N(N+1)/2* keys

*** Confidential and Proprietary ***

Public Key Infrastructure (PKI)

- Today, most signing/encryption is done with Public Key Encryption Algorithms
  - Users have a Public Key and a Private Key
  - Only need one key per party, total of *N* keys for *N* parties
  - Keys are bound to users with Certificates
  - Examples: RSA, Elliptic Curve etc.

- Managing PKI Certificates has issues of its own
  - How do I revoke a certificate?
  - How do I find the certificate of a recipient?
  - How do I manage certificate distribution
  - What do I do if private keys are lost
  - …

*** Confidential and Proprietary ***

Identity-Based Ecryption

Basic Idea: Public-key Encryption where Identities are Public Keys

- **IBE Public Key:**

  alice@gmail.com

- **RSA Public Key:**

  Public exponent=0x10001

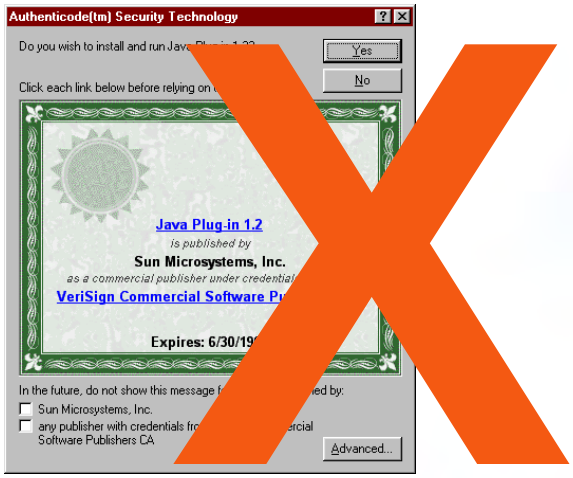
  Modulus=13506641086599522334960321627880596993888147
  56056670275244851438515265106048595338339402871 5
  05719094417982072821644715513736804197039641917 4
  30464965892742562393410208643832021103729587257 6
  23585096431105640735015081875106765946292055636 8
  55294752135008528794163773285339061097505443349 9
  9811150056977236890927563

*** Confidential and Proprietary ***

IBE does not need certificates

- Certificates bind Public Keys to Identities
  - e.g. bob@b.com has key 0x87F6…
  - Signed by a Certification Authority

- In IBE, Identity and Public Key is the same
  - No certificate needed
  - No certificate revocation
  - No certificate servers
  - No pre-enrollment



*** Confidential and Proprietary ***

Identity-Based Encryption (IBE)

- IBE is an old idea
  - Originally proposed by Adi Shamir, co-inventor of the RSA Algorithm, in 1984
- First practical implementation
  - Boneh-Franklin Algorithm published at Crypto 2001
  - Based on well-tested building blocks for encryption (elliptic curves and pairings)
- IBE is having a major impact already
  - Over 200 scientific publications on IBE/Pairings
  - Boneh-Franklin paper cited 356 times so far (Google Scholar)
  - Dan Boneh awarded 2005 RSA Conference Award for Mathematics for inventing IBE

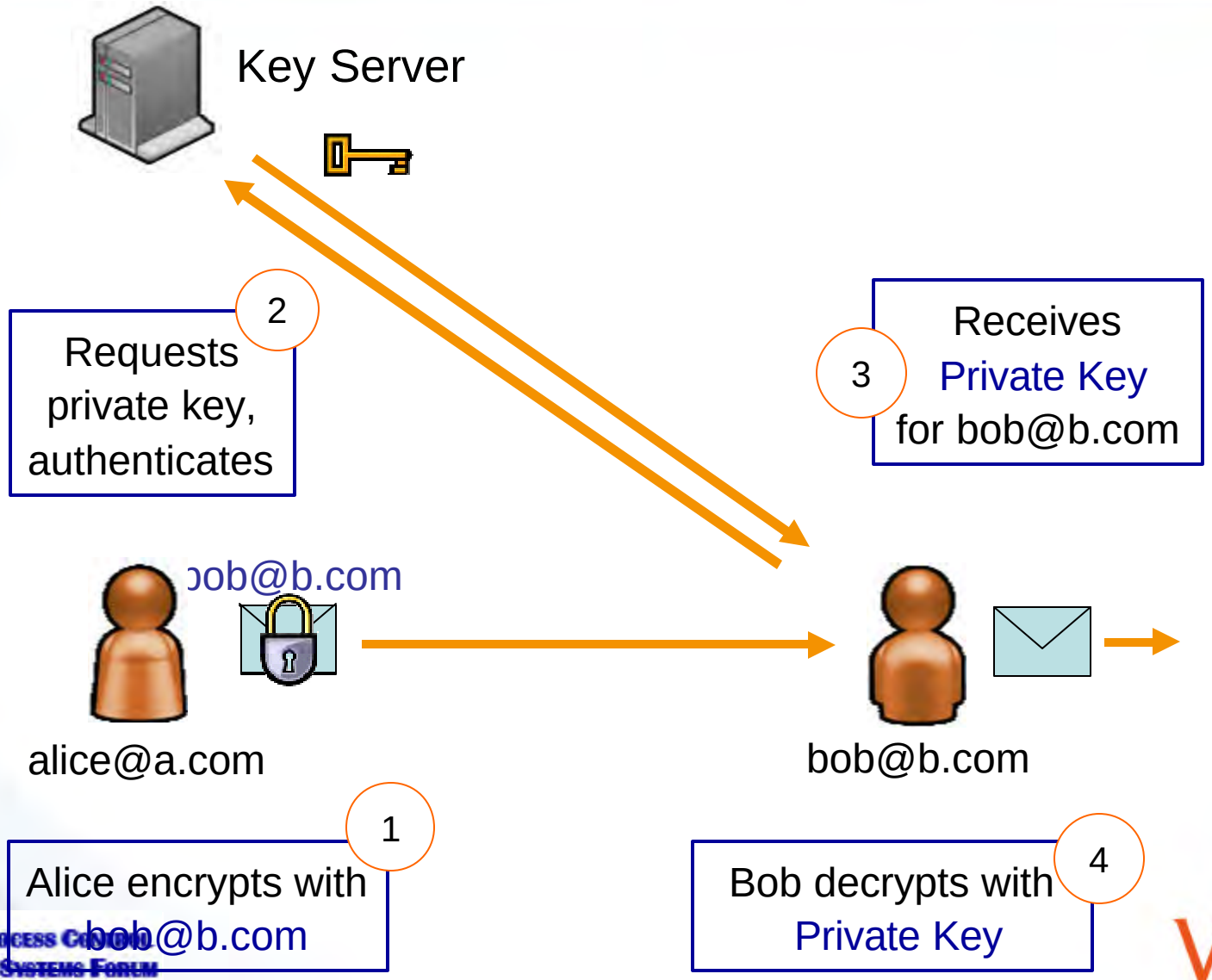
*** Confidential and Proprietary ***

IEEE 1363.3 – Pairing Based IBE Standard

- IEEE 1363 Standards Group
  - Wrote standard on RSA and Elliptic Curve Cryptography
  - Now taking steps to standardize IBE
- Pairing Based Cryptography Standard
  - IBE is the main algorithm in the standard
  - Tentative name is IEEE 1363.3
  - Formal working group expected to start work in Summer 2005
- IBE Standards kick-off workshop at the National Institute of Standards (NIST) on April 18th

*** Confidential and Proprietary ***

How IBE works in practice
Alice sends a Message to Bob



*** Confidential and Proprietary ***

How IBE works in practice
Alice sends a Message to Bob

Key Server

Fully off-line - no connection to server required

1

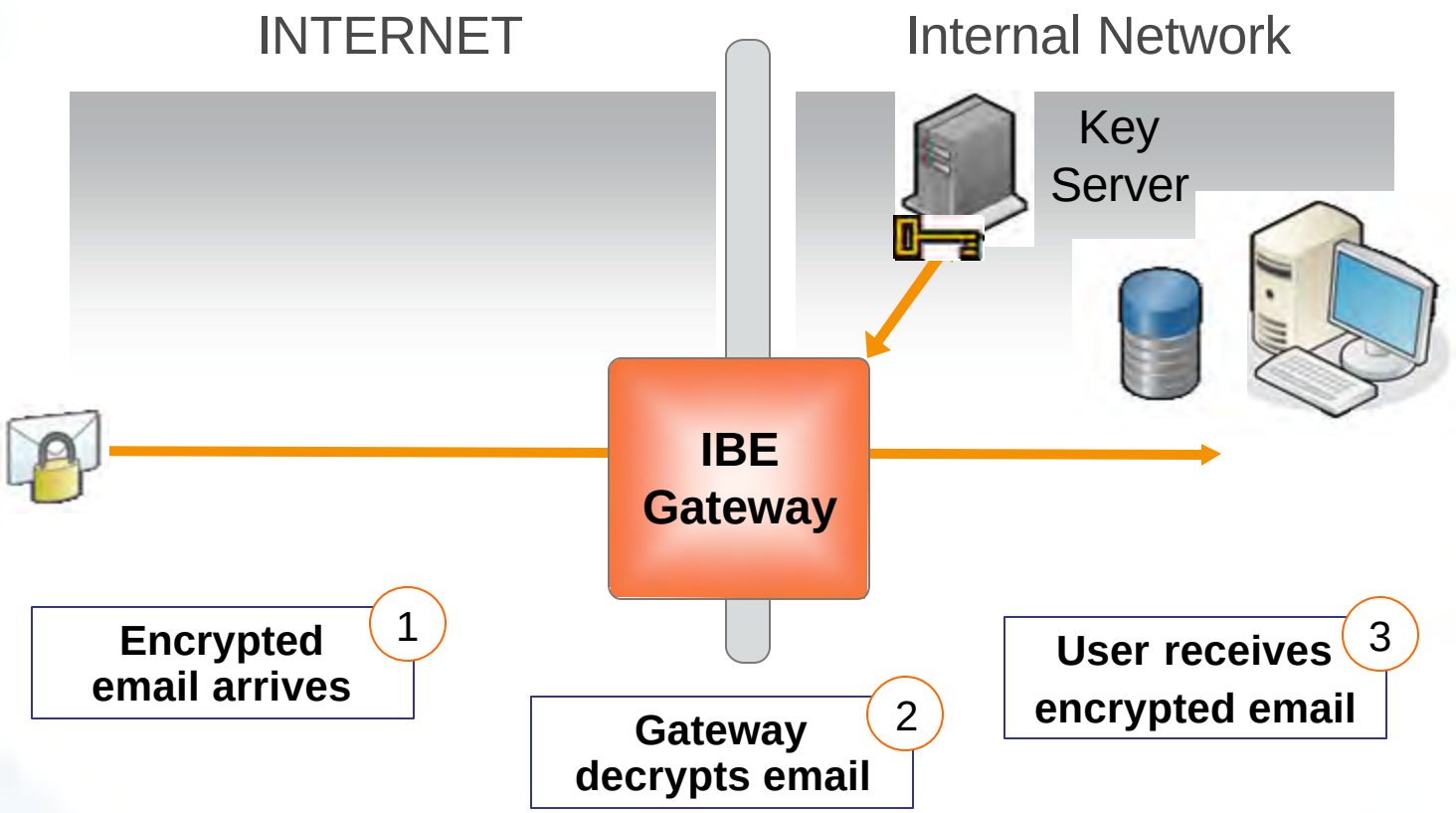
Charlie encrypts with bob@b.com

2

Bob decrypts with Private Key

*** Confidential and Proprietary ***

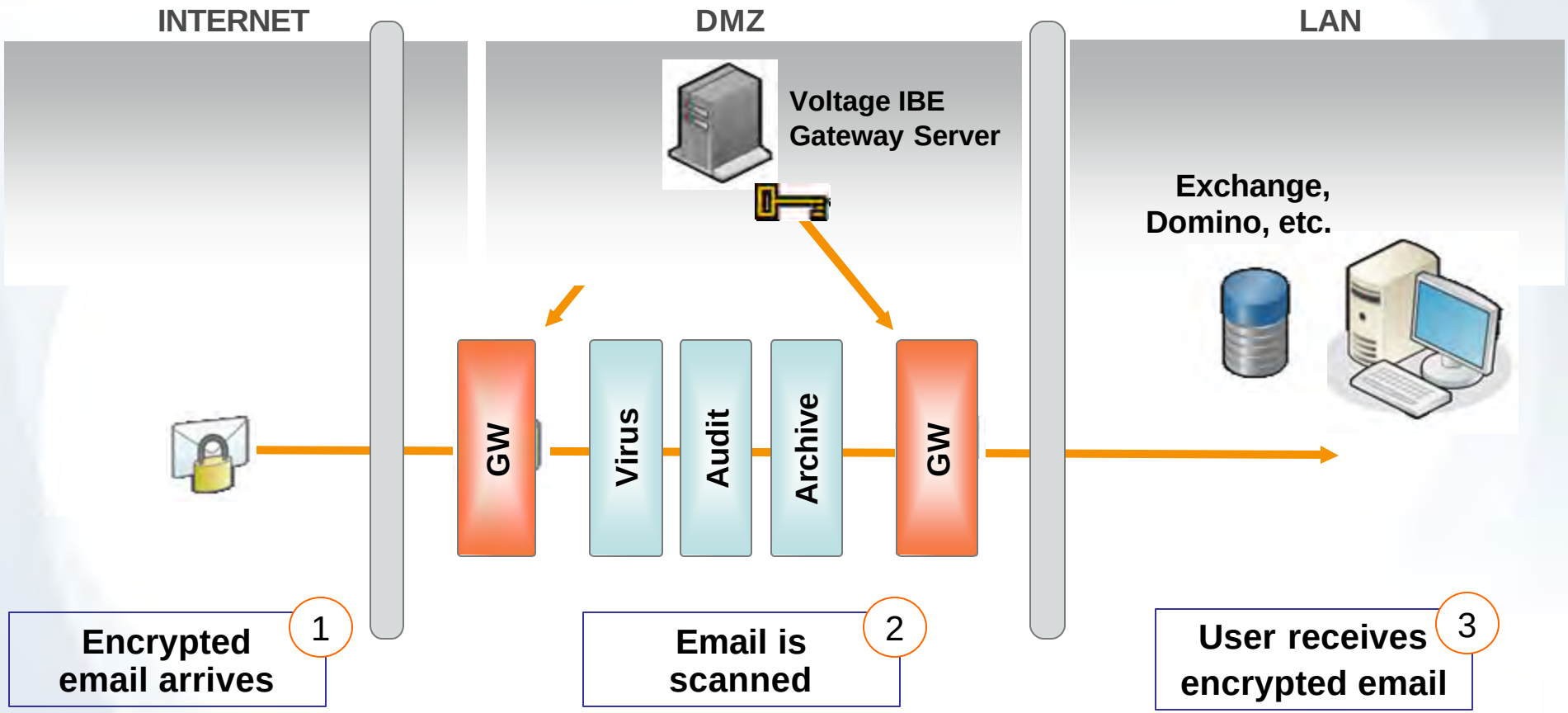
Securing Data at the Perimeter



*** Confidential and Proprietary ***

Inspecting Secured Data

IBE allows content inspection for end-to-end encrypted data



*** Confidential and Proprietary ***

User authentication

Voltage can support any type of authentication

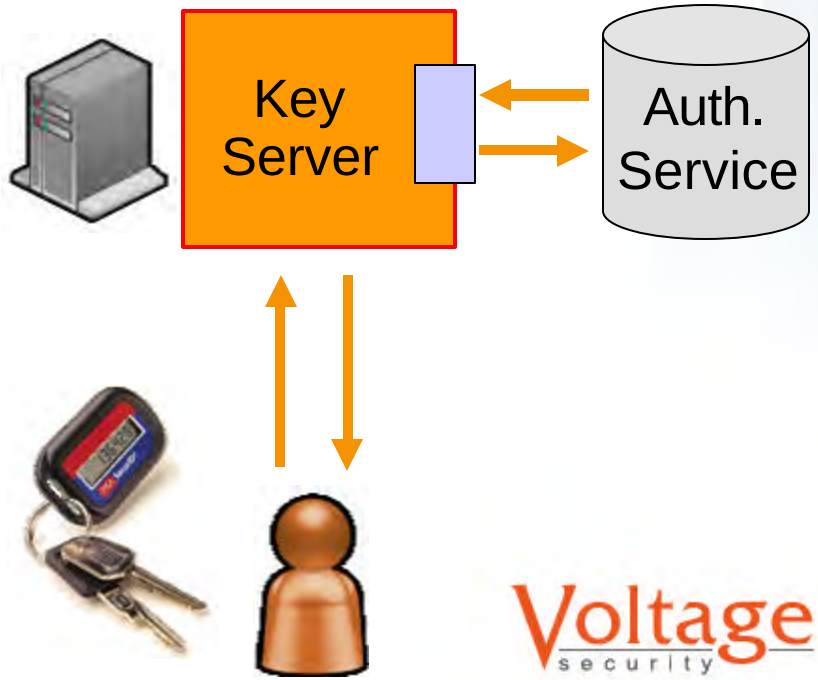
Authentication needs differs by Application

- More sensitive data, requires stronger authentication
- Identity-Based Encryption scales across all levels

External auth integration

- Leverage existing passwords, directories, portals, etc.
- Out of the box support for ad-hoc, self-provisioning auth

More on this later…



*** Confidential and Proprietary ***

The IBE Key Server

Key Server

Master Secret
s =1872361923616378

Request for Private Key
for Identity **bob@b.com**

- Key Server has “Master Secret” to generate keys
  - A random secret is picked when the server is set up
  - Each organization has a different Master Secret
  - Private key is generated from Master Secret and Identity

*** Confidential and Proprietary ***

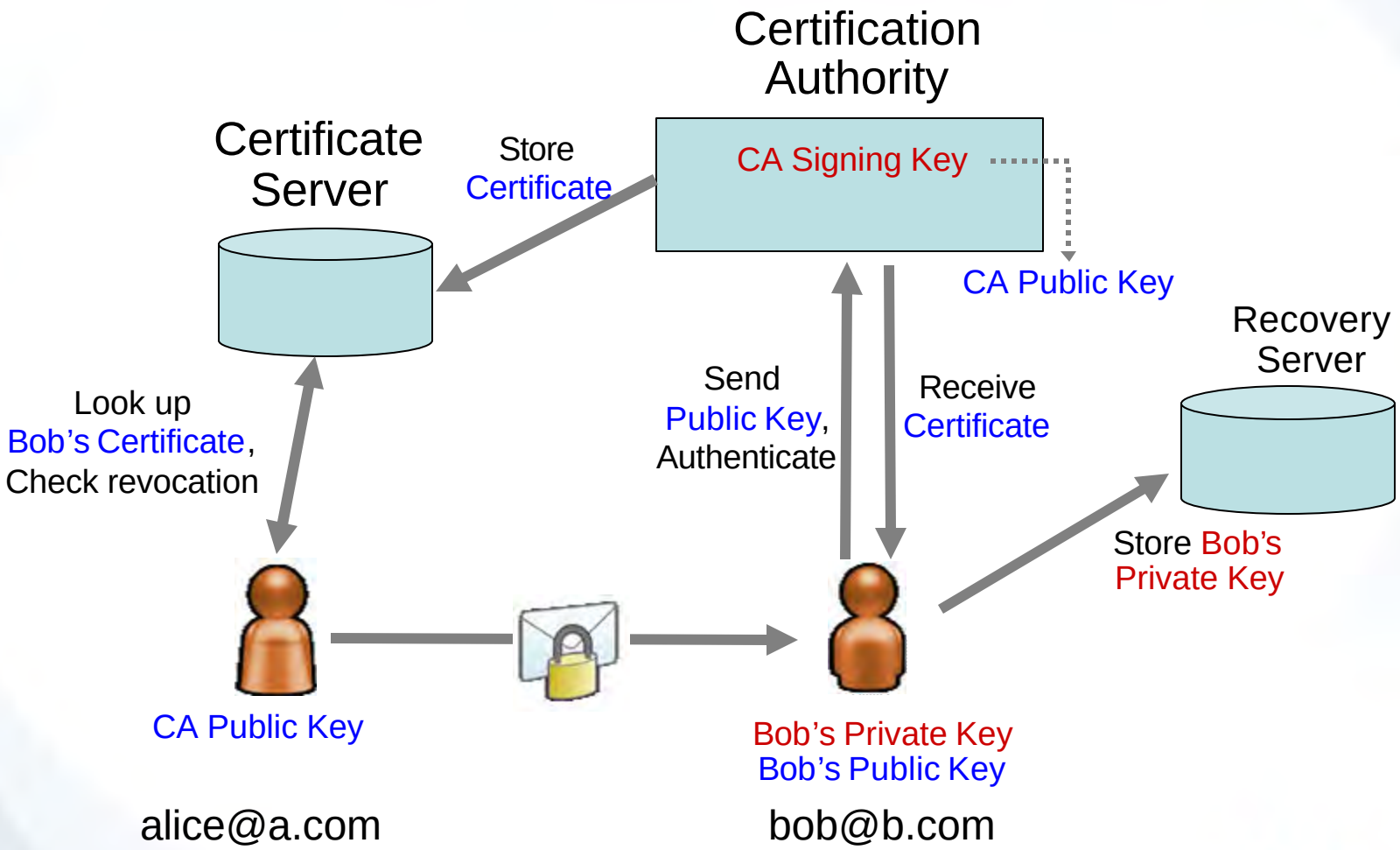
Key Revocation, Expiration and Policy

$$\underbrace{\text{bob@wellsfargo.com}}_{\text{e-mail address}} \;\|\; \underbrace{\text{week = 252}}_{\text{key validity}}$$

- Key validity enables revocation – “key freshness”
  - Every week public key changes, so every week a new private key is issued → revocation can be done on weekly basis
  - To revoke someone, simply remove him from the authentication mechanism (e.g. corporate directory)

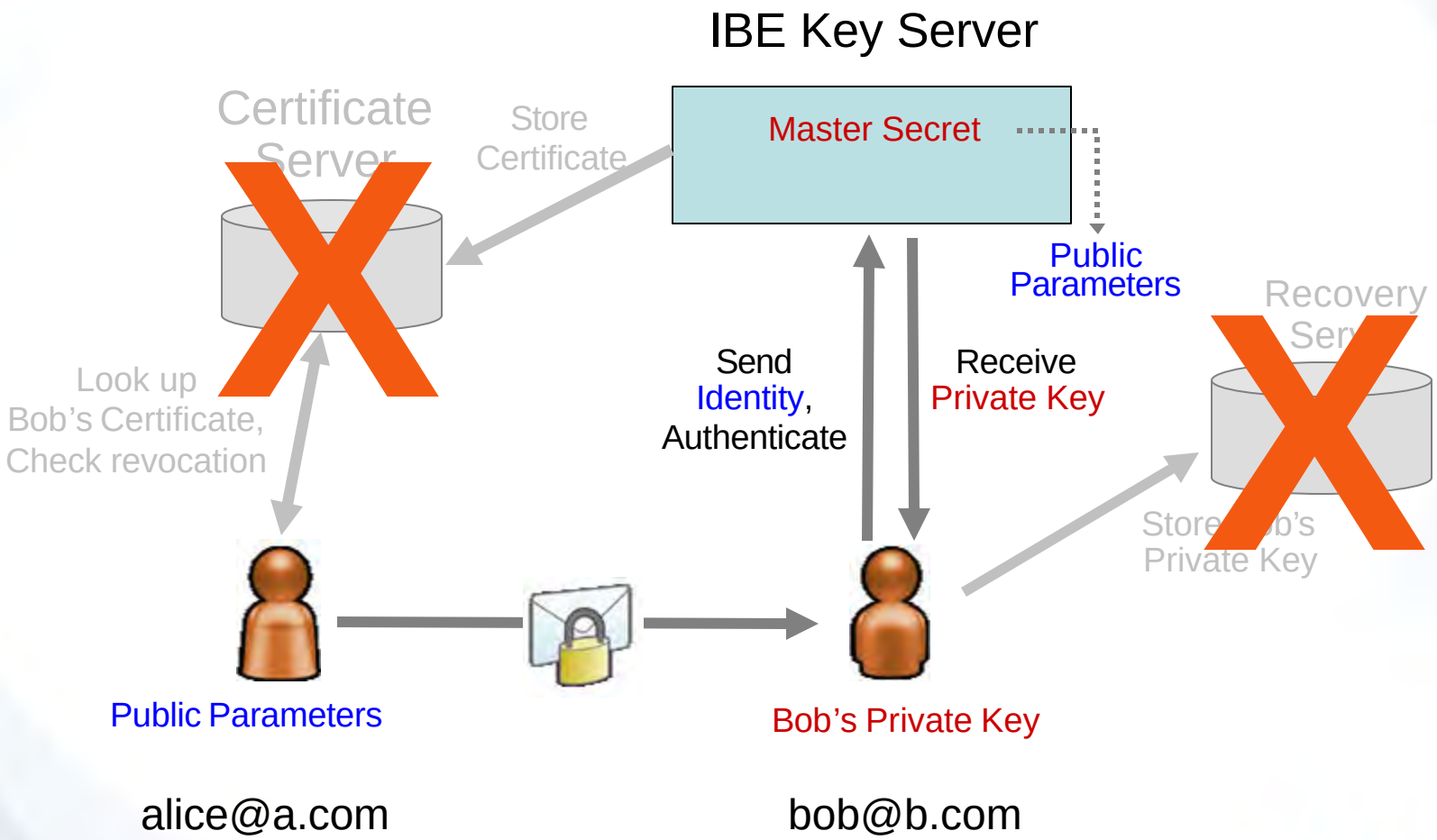
*** Confidential and Proprietary ***

Key Management - Public Key Infrastructure
Certificate Server binds Identity to Public Key



*** Confidential and Proprietary ***

Key Management - IBE
Binding is done by mathematics



*** Confidential and Proprietary ***

Key Management – IBE vs. PKI

- IBE has no Certificates and Certificate management
  - No certificate server, certificate lookups
  - No certificate (or key) revocation, CRLs, OCSP etc.
    - Instead, IBE uses short-lived keys.
- Unlike PKI, IBE requires no pre-enrollment
  - In PKI, recipient must generate key pair before sender can encrypt message
  - IBE is Ad-Hoc capable, a sender can send message at any time
- IBE eliminates encryption key recovery/escrow server
  - Most PKI applications require access to private keys (e.g. Lost keys, Financial Audit, Virus Filtering etc.)
  - Key server can generate any key on the fly

*** Confidential and Proprietary ***

Hardware support for IBE

- IBE Smart Cards
  - Two major smart card vendors now support IBE
  - Encryption/Decryption done natively on the card
  - Under one second per operation

- Special Server Hardware
  - Protect Master Secret with a Hardware Security Module (HSM)
  - Key generation takes place inside the HSM
  - Authentication may also take place inside HSM
  - Even if you steal the server, you can't get the master secret or issue keys

*** Confidential and Proprietary ***

Agenda

1. Identity-Based Encryption (IBE)
   - Key management
   - How IBE works
   - IBE vs. PKI
2. Lightweight Communications with IBE
   - Identity-Based Key Exchange
   - Handshake-free protocols

*** Confidential and Proprietary ***

Lightweight Communication between Devices with IBE

- In an IBE System, public keys are not limited to email addresses
  - Any Identifier can be used as public key
  - e.g. IP addresses, Device IDs,
  - If I know your Device ID, I can encrypt data to you
- We can combine long and short-lived keys
  - E.g. Devices and Controllers get long-lived keys
  - People (service personal) get short-lived keys

*** Confidential and Proprietary ***

IBE based Key Exchange

- With IBE, you can do a key exchange doesn't require communications
  - Shared secret can only be generated if you know the other ID and your private key:

$$S(ID_A, K_{Priv}B) = S(ID_B, K_{Priv}A)$$

  - From this you can generate symmetric Encryption Key *s* and MAC Key *h* using hashing
  - No lookup to certificate server or naming server required
  - No latency from handshake

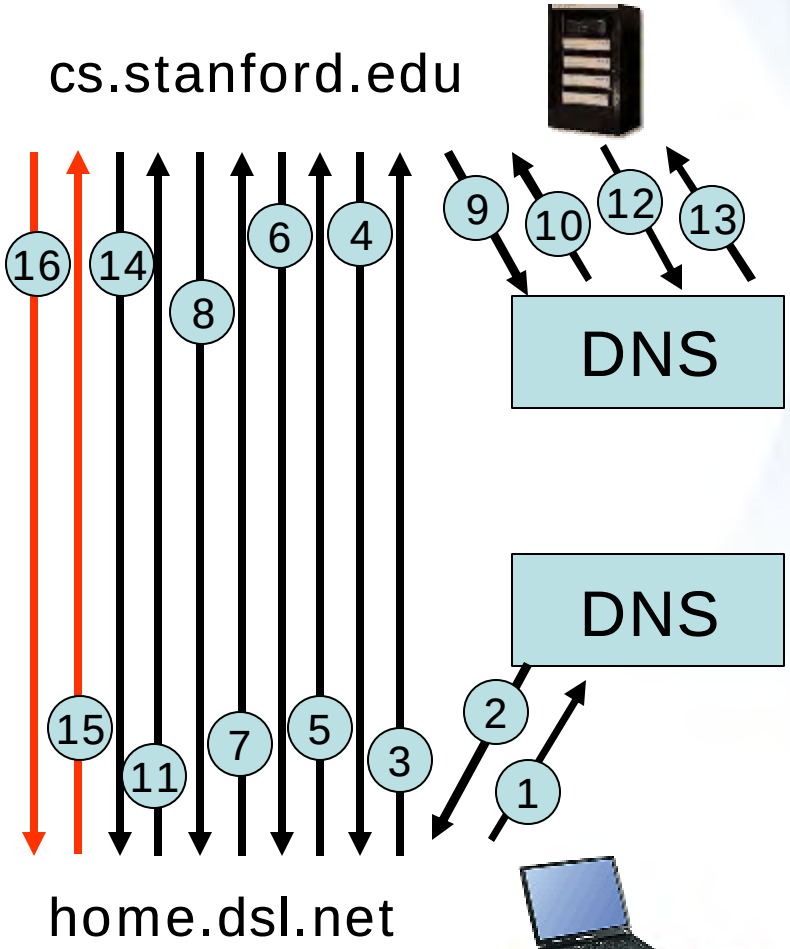
*** Confidential and Proprietary ***

Example: IPSec with Opportunistic Encryption

RFD draft-richardson-ipsec-opportunistic

Example: ping –c 1 71.64.64.64

1. DNS request: txt record for B
2. DNS reply: txt record for B
3. ISKAMP phase 1, Main Mode
4. ISKAMP phase 1, #2
5. ISKAMP phase 1, #3
6. ISKAMP phase 1, #4
7. ISKAMP phase 1, #5
8. ISKAMP phase 1, success
9. DNS request: txt record for A
10. DNS reply: txt record for A
11. ISKAMP phase 2 negotiation
12. DNS reverse lookup for A
13. DNS reply
14. ISKAMP phase 2, #4
15. First encrypted packet
16. Encrypted return packet



*** Confidential and Proprietary ***

IBE based Network Layer Security

Low overhead, no handshake required

- Outer packet format

IP header	$HMAC_h[Encr.\ Pld]$	$Encr_s[Payload]$
40 bytes	20 bytes	[...]

- Payload Format

Random	*len*	*proto*	-	IP Packet Data
20 bytes	2	1	1	[...]

  - IP header unencrypted
  - *proto* is original IP next header protocol
  - *length* is original length (encrypted length is padded for block cypher)
  - Overhead: 44 bytes per packet

*** Confidential and Proprietary ***

Conclusion

- IBE is a new public key encryption algorithm
  - Identities (e.g. names, device IDs) are public keys
- IBE reduces the overhead associated with encryption and signing infrastructure
  - No certificates, certificate lookup required
  - Simplified revocation and expiration
  - Key generation at the key server
- IBE allows you to build lightweight secure protocols
  - Handshake-free key exchange
  - No need for external secure naming service

*** Confidential and Proprietary ***

*** Confidential and Proprietary ***