# IEEE P1363.2:
# Password-based Cryptography

David Jablon
CTO, Phoenix Technologies

***NIST PKI TWG - July 30, 2003***

# What is IEEE P1363.2?

- **"Standard Specification for Password-Based Public-Key Cryptographic Techniques"**
  - **Proposed standard**
  - **Companion to IEEE Std 1363-2000**
  - **Product of P1363 Working Group**
  - **Open standards process**

# One of several IEEE 1363 standards

- Std 1363-2000
  - Sign, Encrypt, Key agreem't, using IF, DL, & EC families
- P1363a
  - Same goals & families as 1363-2000
- P1363.1: Lattice family
  - Same goals as 1363-2000, Different family
- P1363.2: Password-based
  - Same families
  - More ambitious goals

# Scope of P1363.2

- Modern “zero knowledge” password methods
  - Uses public key techniques
  - Uses two or more parties
  - Needs no other infrastructure
- Authenticated key establishment
- Resists attack on low-grade secrets
  - passwords, password-derived keys, PINs, ...

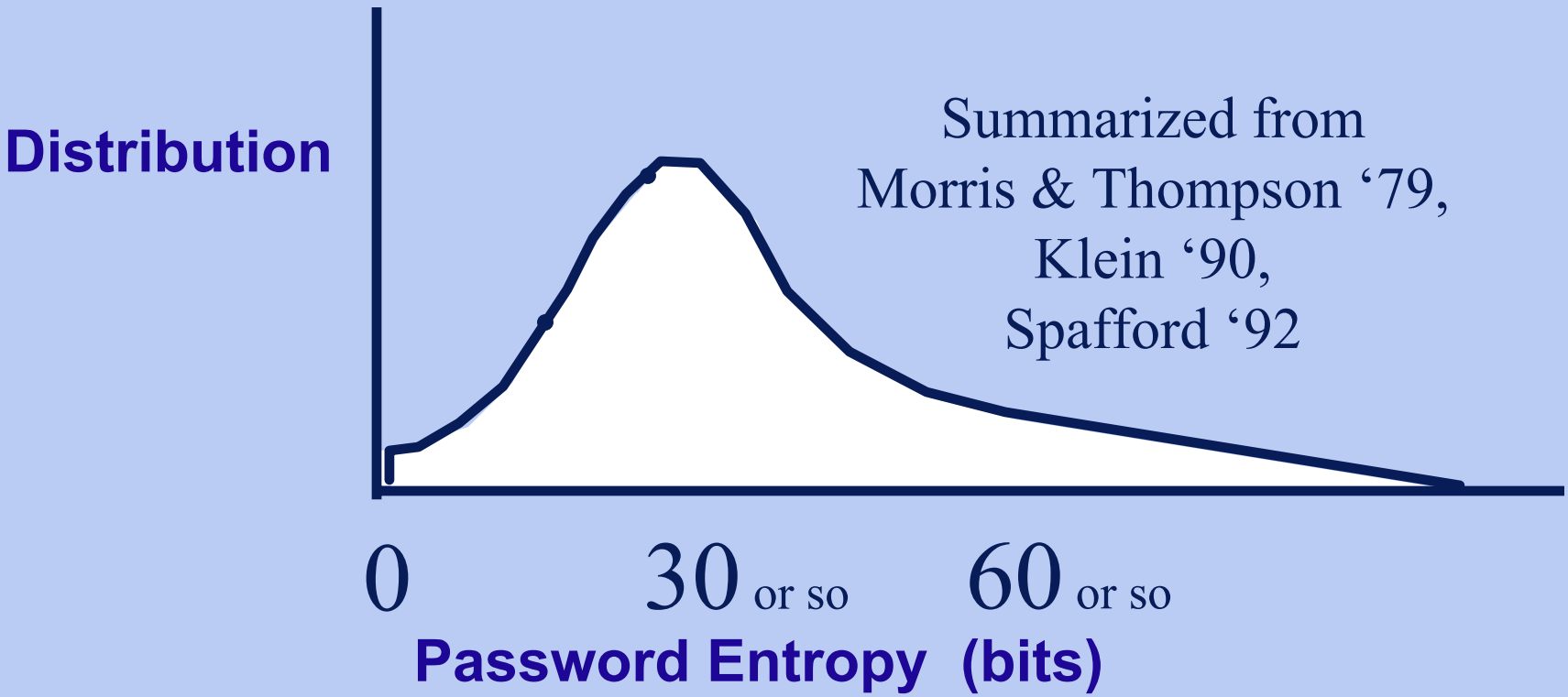
# Rationale (1)

- Why low-grade secrets?
  - People have trouble with high-grade keys
    - storage -- memorizing
    - input -- attention to detail
    - output -- typing
  - Passwords are ubiquitous
  - Easy for people to memorize, recognize, and type.
  - Reduce security/convenience tradeoffs.

# Rationale (2)

- Why use public-key techniques?
  - Symmetric methods can't do it.
- Why new methods?
  - Different than symmetric, hash, or other PK crypto.
  - AES, SHA-1, DH, and RSA can't do it alone.

# Chosen Password Quality



# History of protocols that fail to dictionary attack (or worse)

- Clear text password $\pi \longrightarrow$
- Password as a key $E_\pi$(verifiable text) $\longrightarrow$
  - (e.g. Kerberos v4)
- Hash-based Challenge $\longleftarrow$ Random *R*
  Response Hash(*R*, $\pi$) $\longrightarrow$
- Password through server-auth. tunnel

# What's wrong with password thru browser SSL tunnel?

- User might not check SSL icon.
- User might not check certificate.
- User might not notice a misspelled name or URL. (Server spoofing attacks.)
- Mistakes in trust interpretation.
- User might enter the wrong password.

# Advantages of mutual ZKPP

- Simultaneous mutual authentication
  - Eliminates trust gap
- Active authentication
  - A step that can't be skipped
- Password not disclosed in process
  - Wrong server doesn't get other passwords

# Rough Evolution of ZKPPs

# History of P1363.2

- Field began c. 1992 with EKE
- First submission to P1363 in 1996
- Work deferred to P1363.2 supplement
- P1363.2 PAR approved in 2000
- Call for submissions through 2001
- Successive refinement of drafts

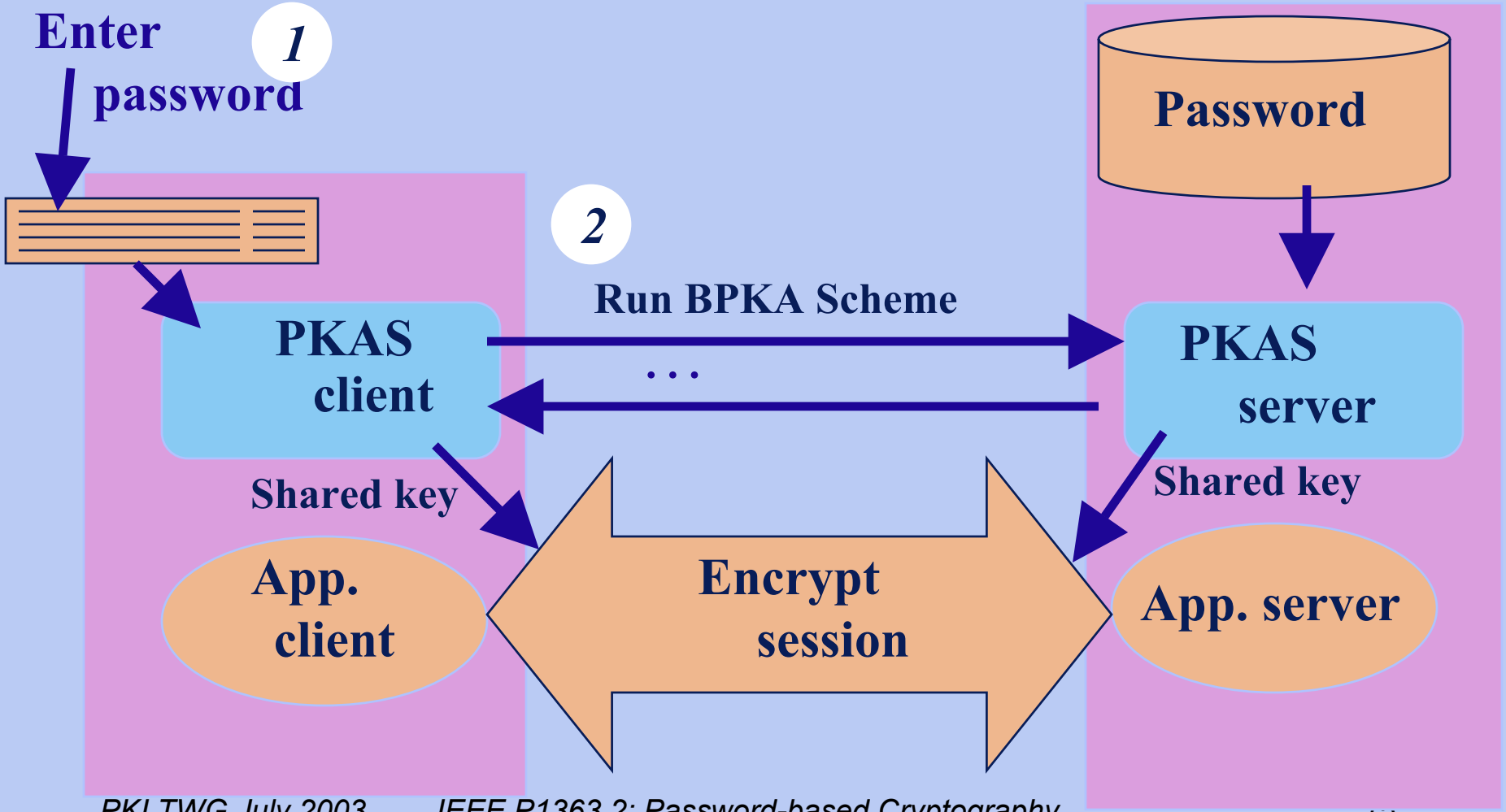
# Focus of P1363.2

- Zero-knowledge password proofs
  - Password authenticated key agreement
    - Balanced
    - Augmented
  - Password authenticated key retrieval
- Use DL and EC (elliptic curve) families

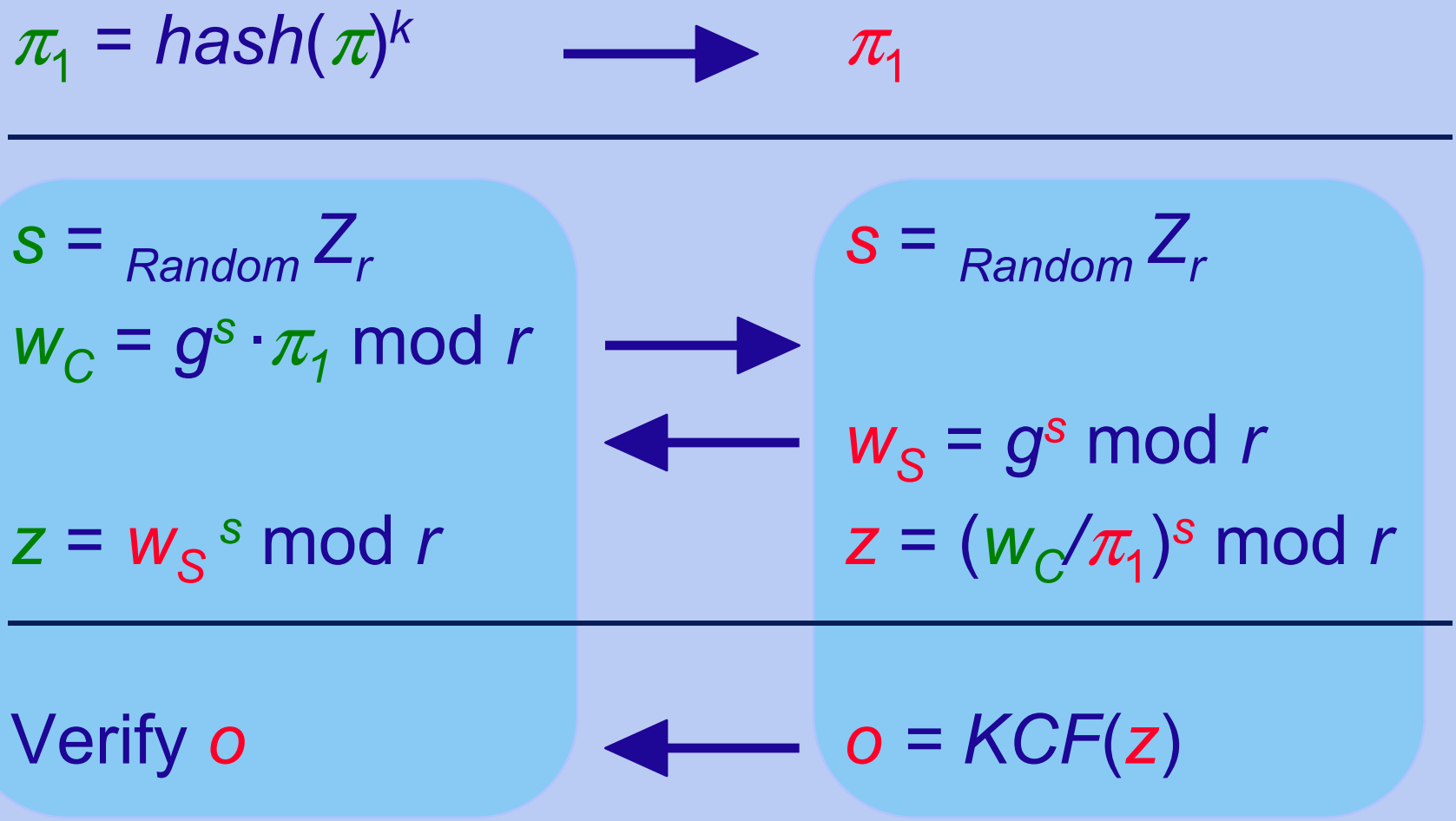
# Balanced PKA Scheme (BPKAS)

- Alice and Bob share same password
  - or same password-derived value
- Mutual ZK proof of password
- Derive shared authenticated key
- Examples: *EKE, PAK, SPEKE*

# How a BPKA Protocol works



# DL BPKAS-PAK (variant of EKE)



# DL BPKAS-SPEKE

$g_1 = hash(\pi)^k$ → $g_1$

---

$s = {}_{Random} Z_r$

$w_C = g_1^{\,s} \bmod r$

→ $w_C$

← $w_S$

$z = w_S^{\,s} \bmod r$

$s = {}_{Random} Z_r$

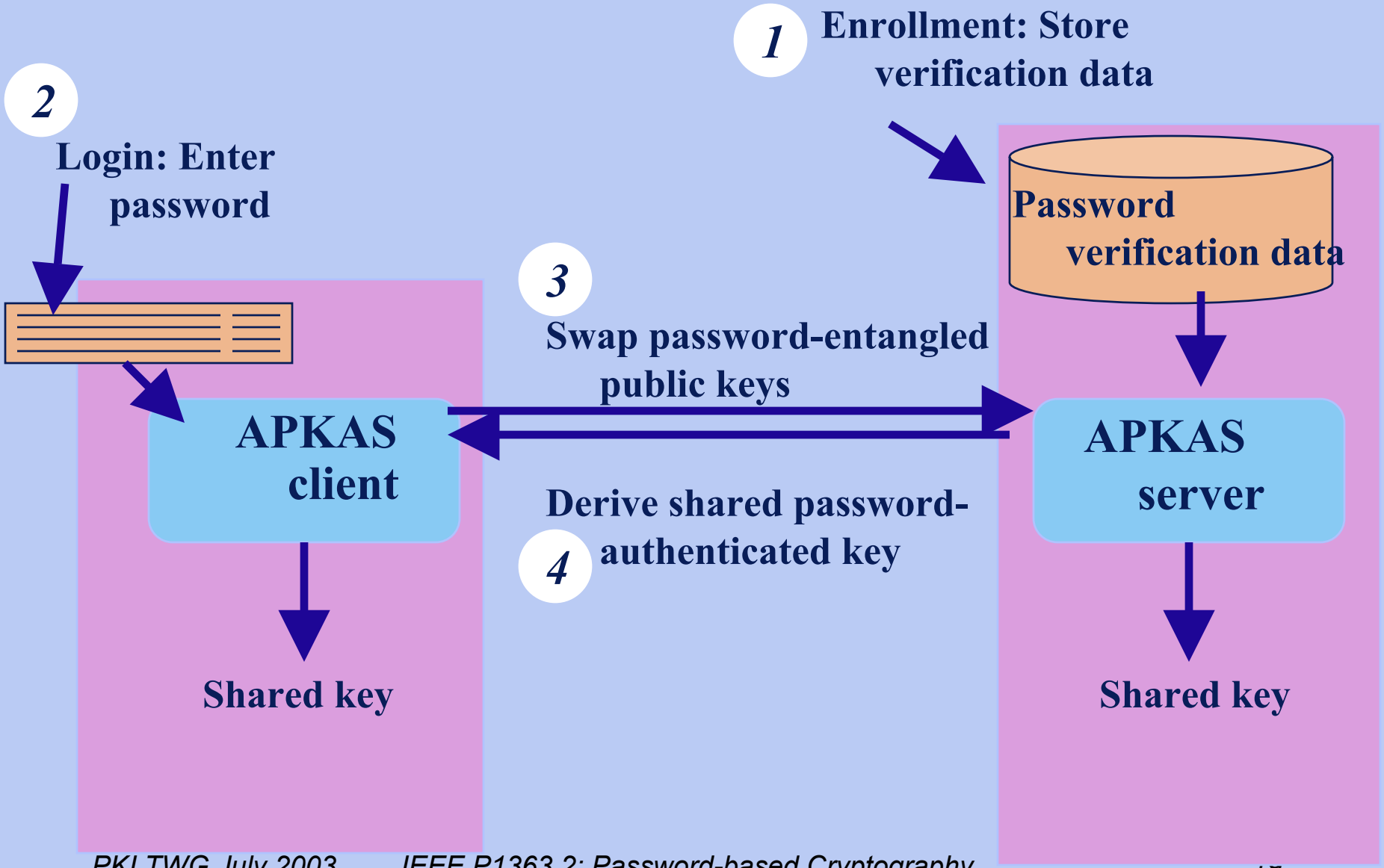
$w_S = g_1^{\,s} \bmod r$

$z = w_C^{\,s} \bmod r$

# Augmented PKA Scheme (APKAS)

- Bob has verification for Alice's password
  - constructed as public key for password
- Mutual ZK proof of password / verification data
  - Alice proves knowledge of password
  - Bob proves knowledge of verification data
- Derive shared authenticated key
- Examples: *B-SPEKE, PAK-Z, SRP*

# How an APKAS Works



# DL APKAS-SRP

$u = hash(\pi)$

$v = g_q{}^u \bmod q$

**($v$ is built using a one-way function, but client can't log in using $v$)**

$v$ →

---

$s = {}_R Z_q$

$w_C = g_q{}^s \bmod q$ →

← $w_S = v + g_q{}^s \bmod q$

$s = {}_R Z_q$

$t = hash(w_S) \bmod 2^{32}$

$z = (w_S - v)^{(s + tu)} \bmod q$

$t = h(w_C) \bmod 2^{32}$

$z = (w_C \cdot (v^{\,t}))^s \bmod q$

---

$o = KCF(z)$ → *Verify o*

# Applications

- General password authentication & Secure connection establishment
- Authenticated key retrieval
  - Roaming protocols
- Wireless connection authentication
  - Provisioning credentials
  - 802.11 wireless key establishment

# Summary of IEEE P1363.2

- IEEE proposed standard
  - work in progress
- Reference for password-based techniques
- Solves important problems
  - with human participants
- Fills a gap in other crypto standards

# Contact Information

- IEEE P1363
  - http://grouper.ieee.org/groups/1363
- Phoenix
  - http://speke.com
- Me
  - David_Jablon@phoenix.com