RSA SECURITY®

Protecting the Knowledge in Knowledge-Based Authentication

Burt Kaliski, RSA Laboratories

NIST/GSA KBA Symposium

February 9-10, 2004

Authentication Access Management Encryption Digital Signatures

Opportunity and Challenge

The Opportunity

- Authenticating users based on knowledge, e.g.,
  - What city were you born in?
  - What is the name of your first pet?
- More *convenient* than passwords

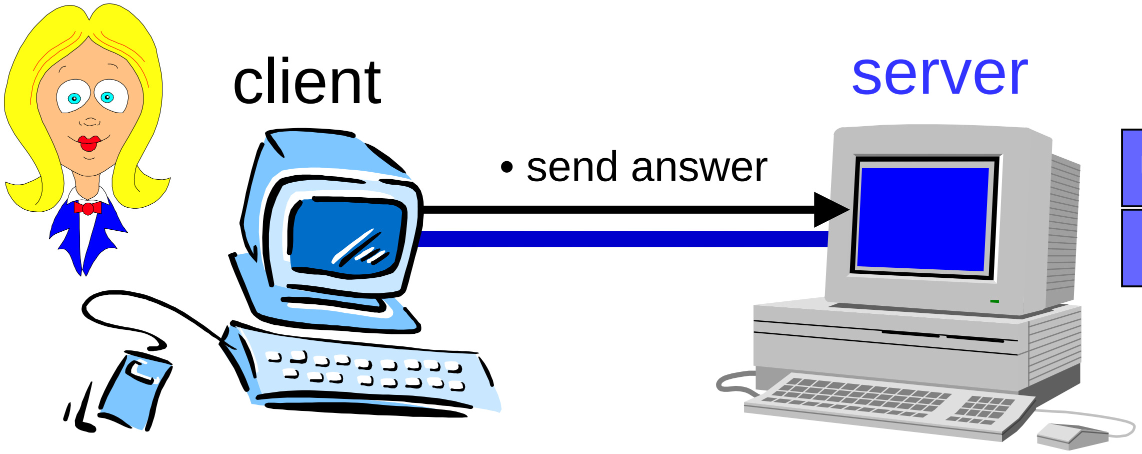
The Challenge

- Protecting that knowledge from compromise
- More *sensitive* than passwords
  - Difficult to "revoke"!

General Model

- Server asks user to provide “knowledge” *K*
- User enters *K* into client
- Client sends *K* to server
- Server verifies *K*

General Model



User Name	Answer
Sally Surfer	123 45 6789

- check answer

Answer is at risk of compromise at client, server, in transit

Protecting the Knowledge

- Client
  - Trusted platform
  - Firewall, virus checking
  - Signed applets
- Transit
  - Server certificate — or protocols like SPEKE based on weak secrets
  - Encryption
- Server
  - Trusted platform
  - Firewall, virus checking
  - Database encryption

focus of this presentation

Risk of Server Compromise

- In typical systems today, answers are stored at a single server
- Server has to *see* and *store* the answers to verify them
- Cryptography on a single server provides limited protection:
  - Hashing can generally be reversed via dictionary attack, because answers are typically searchable
  - Encryption keys often stored on same server
- Insiders and outsiders both pose a threat
  - Risk of compromise ➔ Risk of identity theft ➔ Liability

Protecting Knowledge with Secret Splitting

1. Two servers, working together, should be able to verify answers
2. Neither server should see or store the answers
3. Neither server, working alone, should be able to verify an answer
4. User shouldn’t need to do *anything* different

- Not good enough just to derive variant answers for each server, because of dictionary attacks
- Goals are met by *secret splitting* and a new *verification protocol*

Secret Splitting

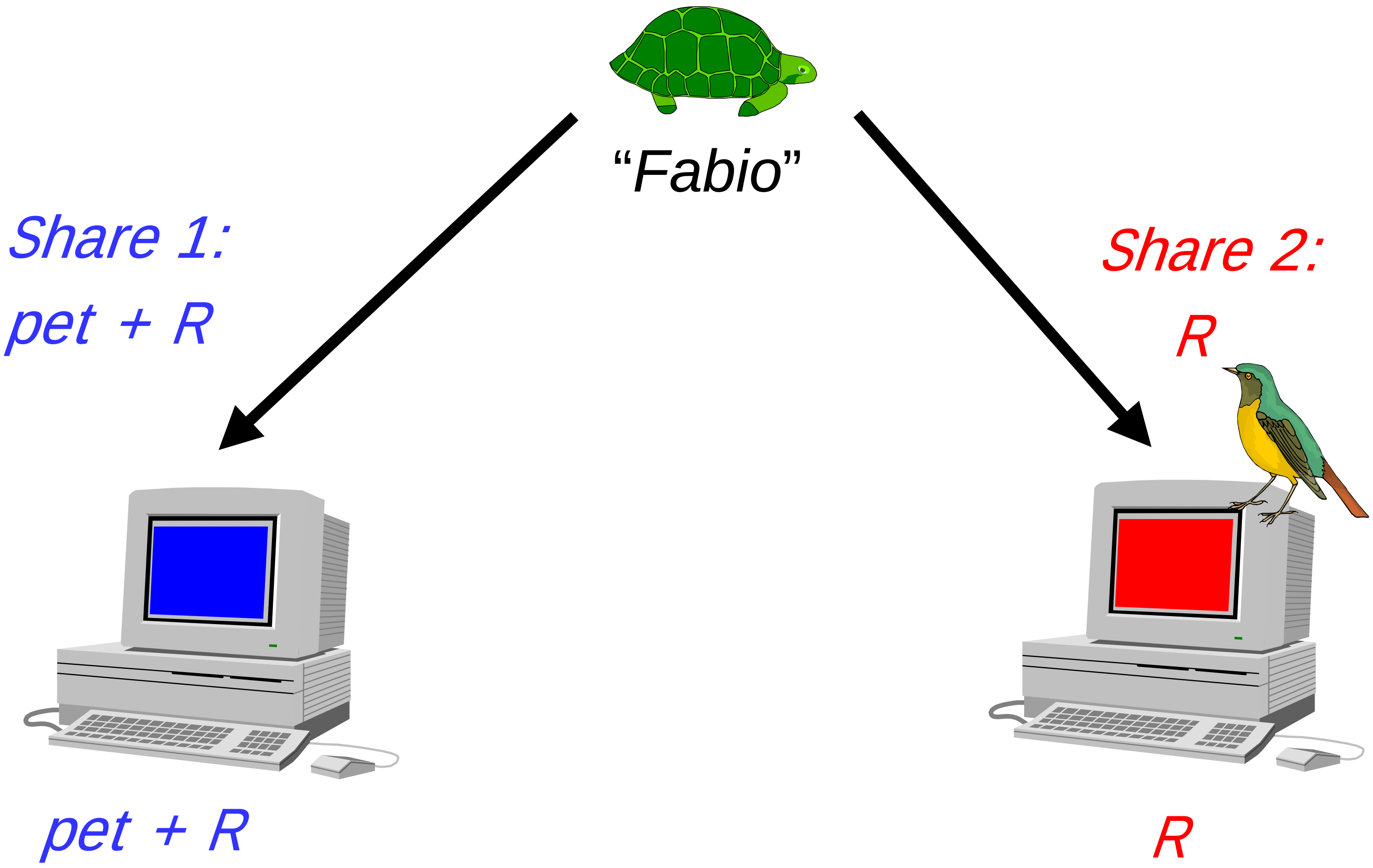
- Adi Shamir in 1979 introduced *secret splitting* as a method of protecting sensitive data
  - Data split into *n shares*
  - Shares stored at *n* servers
  - Data can be reassembled from *k* or more shares
  - If fewer than *k* shares compromised, data still secure
- Secret splitting already being applied to protect high-entropy cryptographic keys — but not previously applied in practice to low-entropy secrets such as “knowledge”

New Verification Protocol

- Nightingale protocol from RSA Laboratories (Brainard et al., USENIX Security 2003)
  - http://developer.rsasecurity.com/labs/nightingale
- Answers split cryptographically into shares for two servers
- Two servers can verify answers together *without seeing or storing them*
  - ➔ Compromise of one server doesn't reveal secrets
- Based on Shamir secret-sharing, zero-knowledge techniques

Registering an Answer:

What was the Name of Your First Pet?



"Fabio"

Share 1:

pet + R

Share 2:

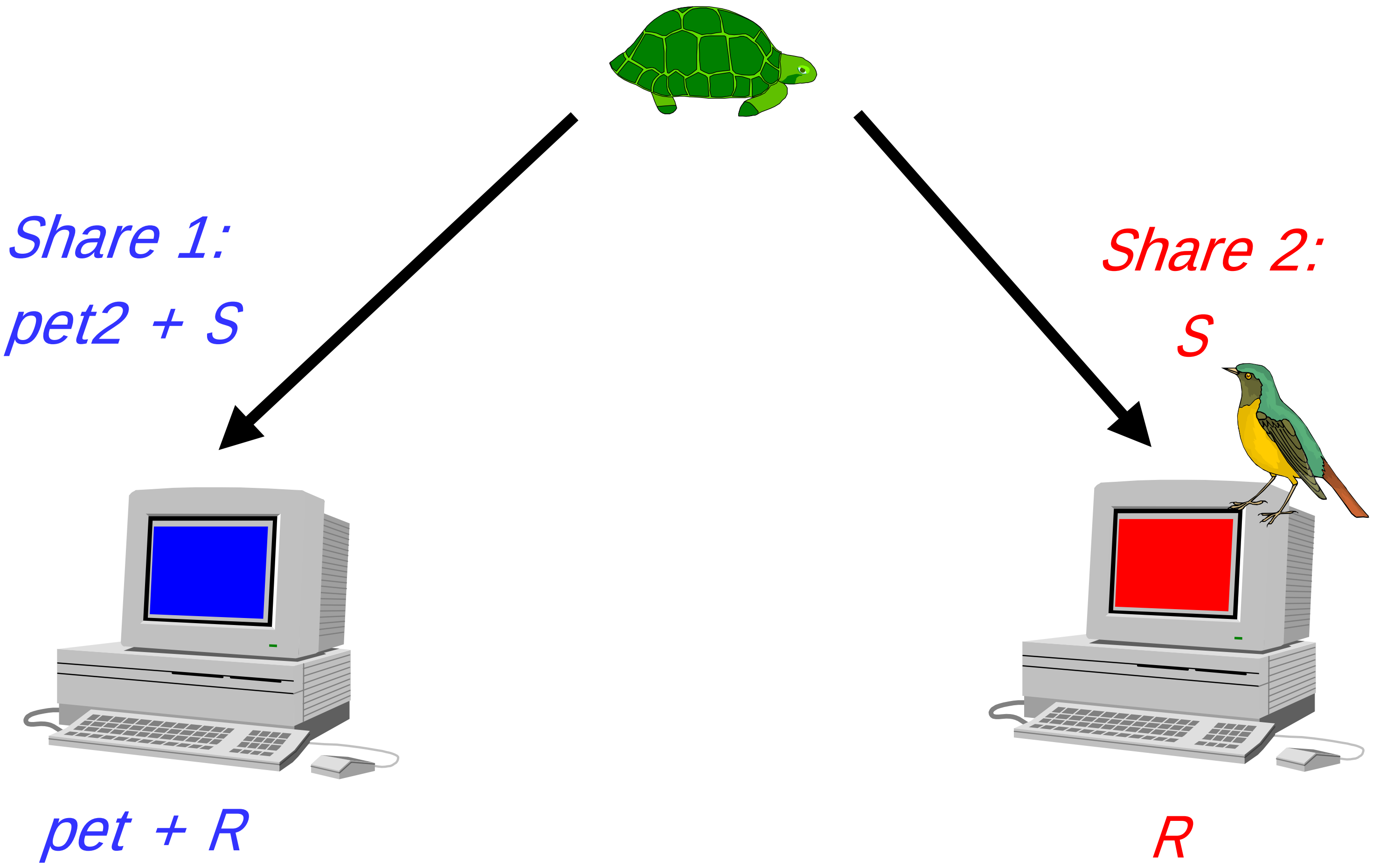
R

pet + R

R

Verifying an Answer:

What was the Name of Your First Pet?



Share 1:

pet2 + S

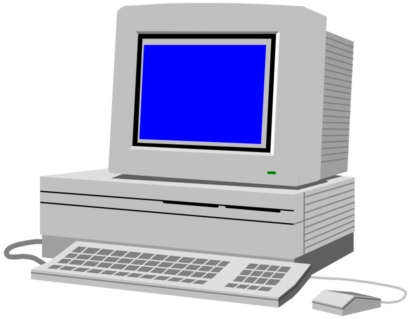
Share 2:

S

pet + R

R

Verifying an Answer (2)



Registered: $pet + R$

Login: $pet2 + S$

$$A = (pet - pet2) + (R - S)$$

R

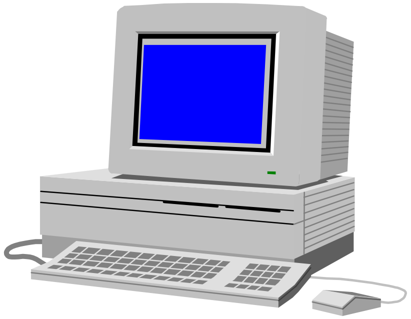
S

$$B = R - S$$

Verifying an Answer (3)

If *pet* = *pet2*, then *A* = *B!*

Otherwise, *A* and *B* are different



$$A \overset{?}{=} B$$

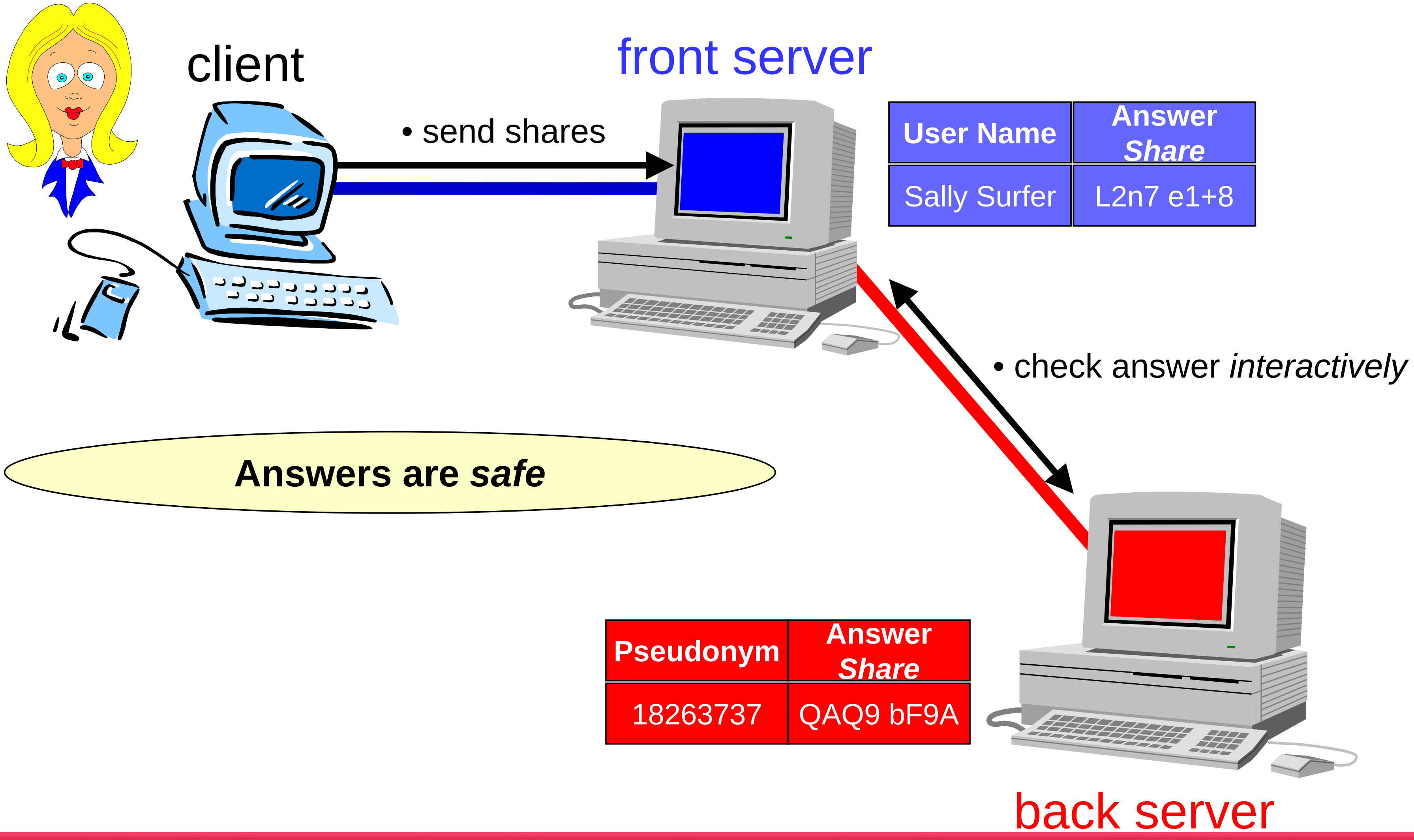
Zero Knowledge Protocol

Revised Model with New Protocol

- Front server asks user to provide “knowledge” *K*
- User enters *K* into client
- Client splits *K* into shares
- Client sends shares to servers
  - for simplicity, can “tunnel” one server’s share through other server by encrypting with that server’s public key
- Front and back servers together verify shares *interactively*

- Servers don’t see or store answer …
  - ➔ No single point of server compromise!
- Result: Convenience *and* protection for KBA

Revised Model with New Protocol



Answers are *safe*

User Name	Answer *Share*
Sally Surfer	L2n7 e1+8

Pseudonym	Answer *Share*
18263737	QAQ9 bF9A

Conclusions

- Knowledge-based authentication is convenient, and it is likely that many applications will use it, especially as standards are defined
- The more applications that use KBA, the more “knowledge” will be handled by servers, and thus the greater risk of compromise, somewhere
- New cryptographic protocols can help improve the protection of knowledge stored at these servers, just as new standards improve the quality of the knowledge itself

A Final Thought: The Threshold Dilemma

- Servers should “lock out” an account after some threshold of unsuccessful authentication attempts
- The threshold dilemma:
  - If too high, attacker can easily get into *some* accounts, without locking any, by guessing a little against all of them
  - If too low, attacker can easily lock *all* accounts!
- More than just a threshold defense is needed. Examples:
  - *IP address tracing* to detect repeat attempts from one source
  - *Client puzzles* to increase attacker’s computational cost
  - *CAPTCHAs* to make automated “bot” attacks more difficult
- Much more to think about in the full solution

Contact Information

- Burt Kaliski
Director, RSA Laboratories
bkaliski@rsasecurity.com
+1 781 515 7073
www.rsasecurity.com