Alice and Bob in Love:

Cryptographic Communication Using Natural Entropy

Joseph Bonneau

University of Cambridge
Computer Laboratory

17th International Workshop on Security Protocols
April 2, 2009

Outline

1. Natural Entropy
2. Protocol
3. Experimental Results
4. Discussion Questions

Human Memory and Entropy

- Evolved to remember emotion, experience
- Can't remember high-entropy crypto keys
- Many pairs of people naturally share a huge entropy pool
    - Lovers
    - Siblings
    - Close friends

Human Challenge-Response

What was the name of the family who lived in the Hill House in Fond-du-Lac, Wisconsin?

Human Challenge-Response

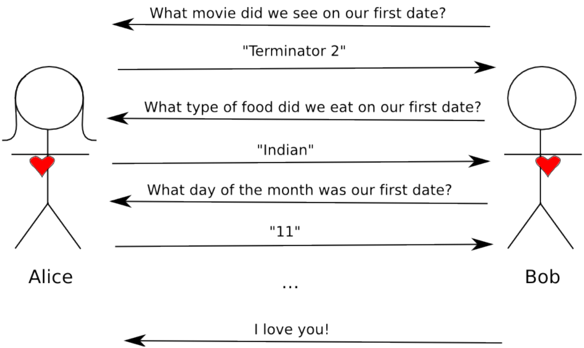
Calvin: i came here for a vacation and i was robbed by some gang
Calvin: i want you to loan me $900
Calvin: you can have the money send via western union

Evan: ok well i want to help you, since we're friends
Evan: ok one question
Evan: what was the name of our high school mascot?

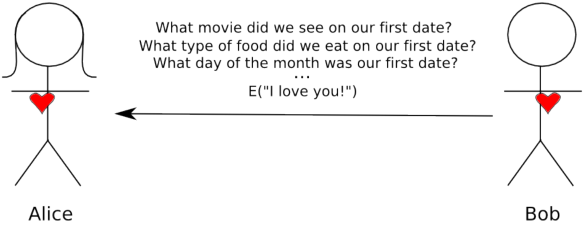
Calvin: Shawnee Mission Northwest High '01

Evan: good luck finding someone stupid
Evan: bye now

Human Challenge-Response



Human Challenge-Response, 1-way?



Applications

- Emergency distress
- Drafting a will
- Password backup

Goals

- Extract cryptographically secure amount of entropy ($\geq$64 bits)
- Minimal recipient sophistication
- Maximise use of available entropy
- Maximise decryption probability

Non-Goals

- Performance
  - Memory overhead
  - Encryption/Decryption processing
- Sender simplicity
  - Grandmother can receive, not send
- Anonymity/Steganography

Building Blocks

- Password Backup Systems
  - Carl Ellison, Chris Hall, Randy Milbert, and Bruce Schneier. "Protecting Secret Keys with Personal Entropy." *Future Generation Computer Systems*, 2000.
    - Use traditional secret-sharing
  - Nyklas Frykholm and Ari Juels. "Error-tolerant Password Recovery." *Computer and Communications Security*, 2001.
    - Use error-correcting code
- Personal Knowledge Questions studied empirically
  - Mostly in the context of online "re-authentication"

Improvements

- Flexible
  - Arbitrary entropy in answers
  - Arbitrary recall probability
- Key Strengthening

Question Generation

- Sender picks a set Q of questions $\{q_0, q_1, \ldots, q_m\}$
  - Also specify answers $A = \{a_0, a_1, \ldots, a_m\}$
- For each quesion q_i, annotate:
  - Entropy for attacker, H_i
  - Recall probability for recipient, r_i
  - Optional: multiple-choice answers

Example

<question>
<entropy>**3**</entropy>
<recall>**0.95**</recall>
<prompt>**What type of restaurant did we go to before a concert at St. John's?**</prompt>
<option>**Chinese**</option>
<option>**Sushi**</option>
<option>**Italian**</option>
<option>**Lebanese**</option>
<option>**Brazilian**</option>
<option>**Mexican**</option>
<option>**Thai**</option>
<option>**Indian**</option>
<answer>**Thai**</answer>
</question>

Encryption

(NB: Protocol tweaked from pre-proceedings paper)

Encryption

- Critical step - Designate subsets of keys which can decrypt:

$$A^* = \{A_i \in A : \text{ knowledge of } A_i \text{ shall enable decryption}\}$$

- Secret-sharing by brute-force
- Will add storage, work overhead proportional to $|A^*|$
  - In practice, this won't kill us

Encryption

- For each decrypting subset A_i, store an offset O_i to recover the master key K_{M}:

$$K_i^0 = \bigoplus_{a_j \in A_i} \mathbf{H}(a_j || j)$$

$$K_i^1 = \mathbf{H}^{2^s}(K_i^0)$$

$$O_i = K_i^1 \oplus K_{\mathrm{M}}$$

- Encryption requires $|A^*|$ storage, $|A^*| \cdot 2^s$ work

Encryption

- Alice sends the following to Bob:
  - $\mathbf{E}_{K_{\mathrm{M}}}(M||A||Q||O)$
  - $\mathbf{MAC}_{K_{\mathrm{M}}}(\mathbf{E}_{K_{\mathrm{M}}}(M||A||Q||O))$
  - Q
  - O
- Decryption straightforward
  - requires searching over $|A^*|$

Optimisation

- How to pick A^*?
- For any set candidate subset $\tilde{A}^* \subset \text{powerset}(A)$ can compute:
  - Minimum entropy brute force path for attacker
  - Estimated success probability for recipient
- Given a desired value for either, can find optimal A^* easily

Structure

- 1 sender (me)
- 8 receivers whom I've had a close relationship with
    - Mother
    - Father
    - Brother
    - Sister
    - Girlfriend
    - Ex-Girlfriend
    - College Roommate
    - High School Friend

Sender Process

- 60 minutes spent per recipient
- Questions created prior to discussing research with subjects
- No external aids (ie photo albums) used
- Chose A^* to yield 64 bits of entropy
- All messages had estimated decryption probability > 0.99

Entropy Estimates

Answer Category	Entropy (bits)
Color	3
TV Title	4
University	5
Movie Title	6
First Name	8
Last Name	10

Recipient Process

- 24 hours to respond
  - All reported ∼10 minutes to complete
- All recipients given other recipients' questions
  - Simulation of inside attacker

Message Stats

Receiver	$\lvert Q \rvert$	H_{total}	$p_{success}$	$\lvert A^* \rvert$
Mother	13	88	0.997	306
Father	14	95	0.998	2,027
Brother	17	98	0.999	9,332
Sister	13	87	0.994	518
Girlfriend	16	89	0.999	3,318
Ex-girlfriend	15	84	0.997	189
Ex-Roommate	13	93	0.999	808
HS Friend	15	101	0.999	10,762
Average	**14.4**	**91.9**	**0.998**	**3,408**

Actual Success Rates

- 6 of 8 messages successfully decrypted
- Overall, 75% of questions answered correctly
    - Predicted 95% . . .

Results

Receiver	$\|Q\|$	Correct	Input	Forgot	Result	Guessed
Mother	13	8	3	2	✖	3
Father	14	7	4	3	✖	3
Brother	17	13	2	2	✔	4
Sister	13	10	2	1	✔	2
Girlfriend	16	14	2	0	✔	0
Ex-girlfriend	15	13	1	1	✔	0
Ex-Roommate	13	10	0	3	✔	1
HS Friend	15	10	0	5	✔	1
Average	**114**	**83**	**14**	**17**		**14**

Error types

- Spelling
  - Vowels Only - '**Rachel**' vs. '**Rachael**'
  - Complex - '**Fruit and Fibre**' vs. '**Fruit 'N Fibre**'
- Phrasing
  - Synonyms - '**shoes**' instead of '**boots**'
  - Grammar - '**ride a bike**' instead of '**riding a bike**'
- Actual Forgetfulness
  - $\frac{1}{3}$ indicated directly as '**don't know**'
  - One answer provided wrong by sender!

Error Breakdown

Result	Frequency
Correct	74%
Vowel errors	3%
Spelling errors	2%
Synonyms	7%
Forgotten	14%

Normalisation

- Expecting some issues, normalisation implemented
  - Conversion to lower case
  - Removal of all punctuation, white-space
  - elimination of ‘the,’ ‘and,’ trailing ‘s’
- Prevented some errors, but not enough
- Normalisation has some limits . . .

Conclusions

- Encryption is possible using natural entropy
  - Appears to be secure
- Usability is terrible for sender
  - Very hard to come up with questions
- Reliability is also lacking
  - hard to accurately predict recall probability

Authentication

- Some implicit authentication
- Encrypt all answers along with message
- Much weaker than confidentiality level
- Adversary can use *any* known information to fool Bob
  - Dumpster diving
  - Malware
  - E-mail/social network account compromise

Experimental Design

- Sample size $N=8$ is insufficient
- Difficult to run a larger study
  - Need fairly sophisticated senders
  - Need sender's actual close relations
- Is the data collected PII?

Experimental Design

- How to model a “real” attacker?
  - Participants unlikely to be highly motivated
- Extremely time-consuming
  - Every question requires different investigation

Privacy Concerns

- How much is given up if the questions Q get published?

Better Sender Interface

- Standard classes of question - mostly useless
- Estimating recall - probably impossible
- Estimating entropy - very difficult
- Standardised multiple choice answers - might help

Normalisation

- More aggressive normalisation possible
  - Soundex & variants
  - User defined (ie only consider first 4 characters)
- Intuition - Hard to get fancy without leaking information
  - Eventually doing homomorphic encryption

Estimating Entropy Automatically

- Realistically only works for multiple choice
- Variation within answer categories
  - *What was the name of our waiter in Dallas?* - high entropy
  - *Which co-worker of yours plays the violin?* - lower entropy
- Requires huge amount of domain-specific knowledge
  - *Where did we stay driving from Phoenix to LA?*

Fuzzy Matching

- "Close" answers mean something
  - *What year did Alissa and Mike get married?* **'2008'**
  - **'2007'** is much better than **'1997'**
- Not quite like normalisation-want to give partial credit
- Cheap solution: divide answer character by character
  - Close answer can still miss badly, ie **'2000'** vs **'1999'**
- Multiple questions, encode close answers at lower entropy level
  - Destroys performance

More Memorable Items

- Humans even better at dealing with images, sounds, smell
- Huge entropy pool available
- Difficult to encode
- Difficult for sender to come up with

Thank You

jcb82@cl.cam.ac.uk