Give Me Letters 2, 3 and 6!
Partial Password Implementations and Attacks

David Aspinall, University of Edinburgh, UK
Mike Just, Glasgow Caledonian University, UK

Financial Cryptography and Data Security, April 2013

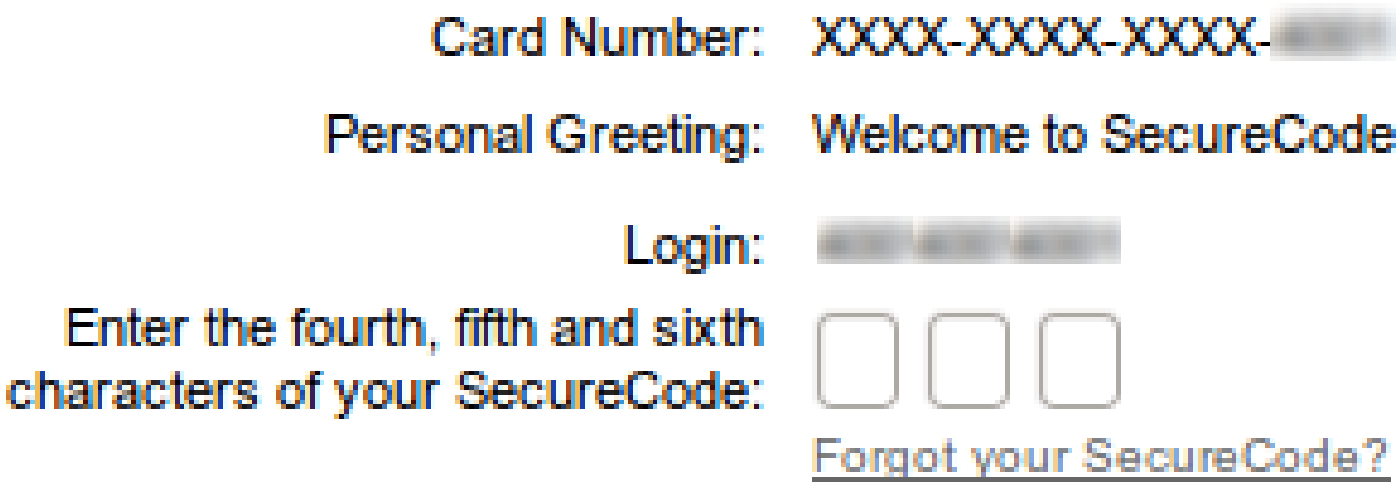
Outline

Partial Passwords

Definitions and examples

A partial password is a challenge on a subset of characters from a full password.

A partial password scheme is an authentication system using partial passwords.



Scheme

Registration User chooses a password of n characters from a set of N

Login Challenge of m positions with response:

Positions:	1	2	3	4	5	6	7
User password:	a	s	h	u	f	1	0
Correct response:		s	h			1	

Retry In case of failure, user challenged again. Number of retries usually limited.

Repeat On next login, challenge changes.

Motivations

Introduced for telephone banking: single observation by operator does not reveal whole secret.

Online, appears to impede several attacks:

- shoulder surfing
- key logging
- man-in-the-browser

Potentially, may also thwart:

- phishing
- offline attacks

Other attractions:

- easy extra authentication step (but not true 2FA)
- cheap (e.g., compared to hardware tokens)

Origins

In UK banking: first introduced for telephone banking.

Matsumoto and Imai, *Human Identification Through Insecure Channel* (Eurocrypt ’91). Related but more elaborate scheme:

- User has a password with known character set
- Challenge: word surrounded by detractor characters
- Response: *substituted* positions and detractors

Repeated several times.

Following work (e.g., Hopper & Bloom 2001): revised schemes and stronger guarantees, but showed required human computation steps are impractical.

So what about schemes actually in use?

Questions

- What are the security assumptions behind current deployment of partial passwords?
- What are good choices for the system parameters: password length, character set size, challenge size?
- How many observations does an attacker need to learn whole password or answer next challenge?
- Are weak passwords such as dictionary words safe?
- Failure mode: should the challenge be changed after failed attempts?
- Are some challenge sequences better than others?
- How usable is the scheme?

Survey

Online banking survey: results

- Used widely in banks, online and telephone
- Elsewhere: credit cards, utilities, outside UK,. . .
- Usually part of a multi-stage authentication, alongside: names, user ids, account details, personal knowledge questions.
- Challenge sizes fixed, vary from 2-3 positions
- Challenge sequences appear random
- Mostly: ascending position challenges, no repeats
- Most repeat same challenge on retry
- Policies generally weaker than for full passwords

Parameters

	character set size, N	password length, n	challenge size, m	second credential
Cooperative	10	4	2	question
ING DiBa (DE)	10	6	2	PIN
Tesco	10	6	2	password
Smile	10	6	2	question
Nationwide	10	6	3	password
AIB	10	5	3	question
B. of Ireland (IE)	10	6	3	date of birth
Nat West, step 1	10	4	2	pp, step 2
Nat West, step 2	36	6–20	3	pp, step 1
HBoS	36	6–15	3	password
3DSecure, BoI	36	8–15	3	credit card #
Standard Life	36	8–10	3	none
Skipton	36	8–30	3	question
First Direct	36	6–30	3	question
Barclays	52	6–8	2	PIN
HSBC (CA)	62	8	3	question

NB: snapshot from Sept. 2012. Thanks to Atif Hussain for help with survey.

Guessing Attacks

Mode of attack for guessing

- online attack against each account
- suppose a fixed number of attempts allowed: β
- some background (e.g., dictionary), ideally limited
- no use of previous observations
- "trawling": use best strategy on many accounts

Two typical instances of scheme:

6 digit ***PIN***

- $N=10$, $n=6$ $m=2$, $\beta=6$

8 character ***alphanumeric***

- $N=36$, $n=8$, $m=3$, $\beta=10$

Guessing methods

1. brute-force (sample from uniform distribution)
2. position-letter frequency (ranked list per position)
3. **projection dictionary** (ranked list per challenge)
4. dependent projection (tree per challenge) *[later]*

Generate background tables by computation on:

- ordinary dictionary, e.g., `/usr/share/dict/words`
- dictionary with frequencies, e.g., RockYou

We calculate β-success rate: proportion of answers covered by the top β guesses.

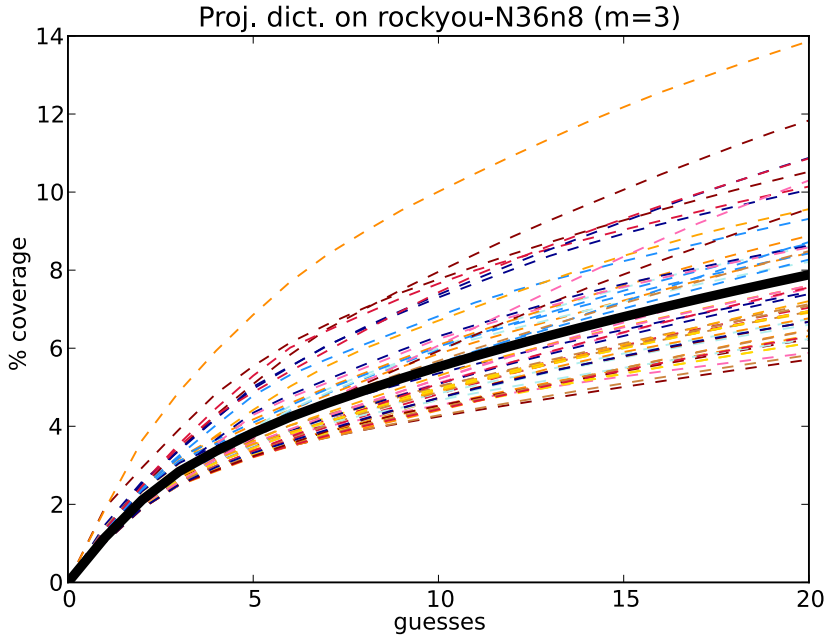
Example projection dictionary attack

Challenge 2 3 6:				*Cum.%*
1.	*a*	*s*	*o*	*1.10*
2.	*l*	*o*	*y*	*1.98*
3.	*r*	*i*	*e*	*2.79*
4.	*2*	*3*	*6*	*3.21*
5.	*a*	*r*	*e*	*3.56*

Challenge 1 2 3:				*Cum.%*
1.	*i*	*l*	*o*	*1.29*
2.	*p*	*a*	*s*	*2.42*
3.	*m*	*a*	*r*	*3.40*
4.	*b*	*a*	*b*	*4.30*
5.	*p*	*r*	*i*	*5.08*

- The top 5 choices for two of the $\binom{n}{m} = 56$ challenges
- Dictionary is RockYou (8-char alphanumeric) with frequencies
- 5.3m total, top 5 words in ranked dictionary covers 3.02%
- Top 5 full words:
  `password`, `iloveyou`, `princess`, 12345678, `babygirl`

Example projection dictionary attack



- This shows the coverage of guesses for increasing β
- Each line is a different challenge, bold is average
- Success rate for β=10 is 5.5% versus 3.9% without projection

Recording Attacks

Mode of attack for recording

- online, β attempts per challenge, as before
- allow recording previous k challenge-response pairs

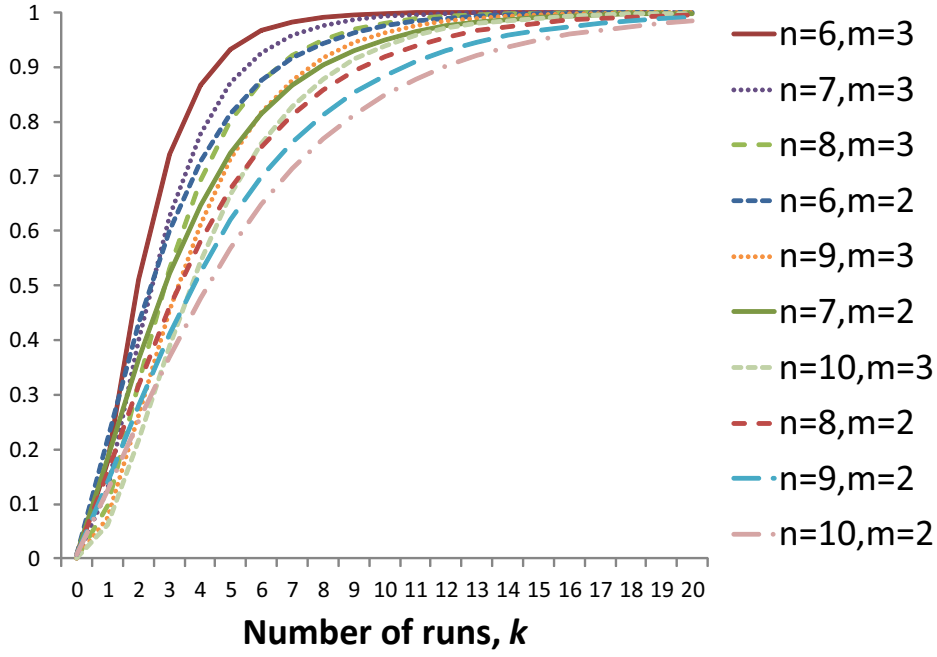
Recording methods

1. Pure recording: only answer when positions known
2. Recording+guessing: guess remainder of positions

Combinatorics: we find equations for two different success rates for increasing k. They are probabilities of:

- answering the next challenge, or
- learning the whole password.

Success rates for answering next challenge



This is a plot of

$$\sum_{j=0}^{m} \overline{s_n^m}(k,j) w_j$$

where $0 \leq j \leq m$ positions are known in a challenge after k runs.

- $\overline{s_n^m}(k,j)$: fraction of challenges with j known positions
- w_j: the β-success rate for a particular guessing method

Summary

Results for typical parameters

Attack type	**parameters**	**% success rate**	
		PINs	alphanumeric
Brute force		6	0.002
Letter position	RockYou	17.2	0.3
Dictionary	RockYou	15.3	3.9
Proj. dictionary	RockYou	30.6	5.5
Recording	k=1 (k=4)	6.7 (63.1)	1.8 (59.0)
Recording + BF Guess	k=1 (k=4)	41.1 (83.8)	9.6 (69.1)
Recording + Best Dict	k=1 (k=4)	60.2 (90.4)	25.2 (81.2)

Summary

- survey of partial password implementations
- model of partial password authentication scheme
- several attack methods, guessing and recording
- theoretical success rates measured analytically (pure recording) and empirically (using a dictionary)

Future/ongoing work:

- Better attacks (dependent case)
- Unseen challenge (Goring et al, 2007)
- Failure modes, challenge schedule and format
- General study of multi-stage authentication
- Discuss more with banks. . .