Analysis of Reduction in Passcode Guessability as a Function of Conversion from Alphanumeric to Numeric Codes

TR 29.3665
July 1, 2003

James R. Lewis
Wallace J. Sadowski

IBM Pervasive Computing

Boca Raton, Florida

Abstract

With the introduction of telephone access to web databases, it is possible that users who already have an alphanumeric passcode for a graphical user interface will want to use the same passcode when accessing the database through a speech user interface. Speech recognition for alphanumeric entry is notoriously poor due to the acoustic confusability of certain clusters of letters (for example, 'S' and 'F'). Thus, it is reasonable to let callers enter their passcodes using the telephone keypad, substituting numbers for letters as required. The purpose of this analysis was to investigate the extent to which a passcode entered in this way becomes more guessable and, consequently, less secure. The results of the analysis indicated that alphanumeric passcodes that might be converted to numeric passcodes should contain at least four characters and, if possible, the enabling system should allow passcodes of variable length.

ITIRC Keywords

IVR
Speech interface
Telephony
VRU
Password
Passcode
Security

Contents

Introduction

Normally, a system designed for use with a telephone would not allow alphanumeric passcodes, but in the last few years more and more telephony systems make use of the same databases as those used by websites, for which the users of the websites already have alphanumeric passcodes. This leads to a need for the entry of alphanumeric passcodes (passwords) with a telephone keypad. There are several existing methods for interpreting telephone key presses as text, used primarily for the entry of names and addresses in phone-based address books or for the typing of instant messages and e-mail. The two most common methods are multipress and multikey (Lewis, Potosnak, & Magyar, 1997).

In multipress, users must press a key the correct number of times to acquire the desired number/letter. For example, to get 'C' a user would press the 2 key four times (cycling through '2', 'A', 'B', 'C'). In multikey, users press the '*', '0', or '#' key to indicate which of the letters they want from a key. For example, to get 'C' a user would press #2. The multikey solution does not provide any obvious way to get numbers instead of letters, and has difficulty handling the keys that have four rather than three letters on keypads that show 'Q' and 'Z'. For this reason, multipress is more commonly used than multikey.

For the production of text that another person will see, it is important to disambiguate the letter, so users must tolerate the clumsiness of the multipress user interface. For the entry of passcodes, which no other human will see, such disambiguation at the user input side is not necessary. Thus, it is possible to offer to users a simpler way to enter an alphanumeric passcode by having the system convert the alphanumeric passcode to a numeric code, substituting numbers for letters following the patterns on the telephone keypad.

The business scenario in which this can happen is when a provider of information services already has a passcode system for their website, and has allowed users to select alphanumeric passcodes because the legacy device for passcode entry is a standard keyboard. The company now wants to make some of the information from its website available via an IVR system in which the user interface device is now a telephone. If the system substitutes numbers for letters following the patterns on the telephone keypad, all the user needs to do is to press a single numeric key for each letter in the passcode. For example, if the passcode for entry into the website was ‘a g z 7 c’, the system would convert the passcode for telephone keypad entry to ‘2 4 9 7 2’. The user wouldn't need to know that the system was performing this conversion, and would only have to memorize a single passcode for entry via either keyboard or telephone keypad.

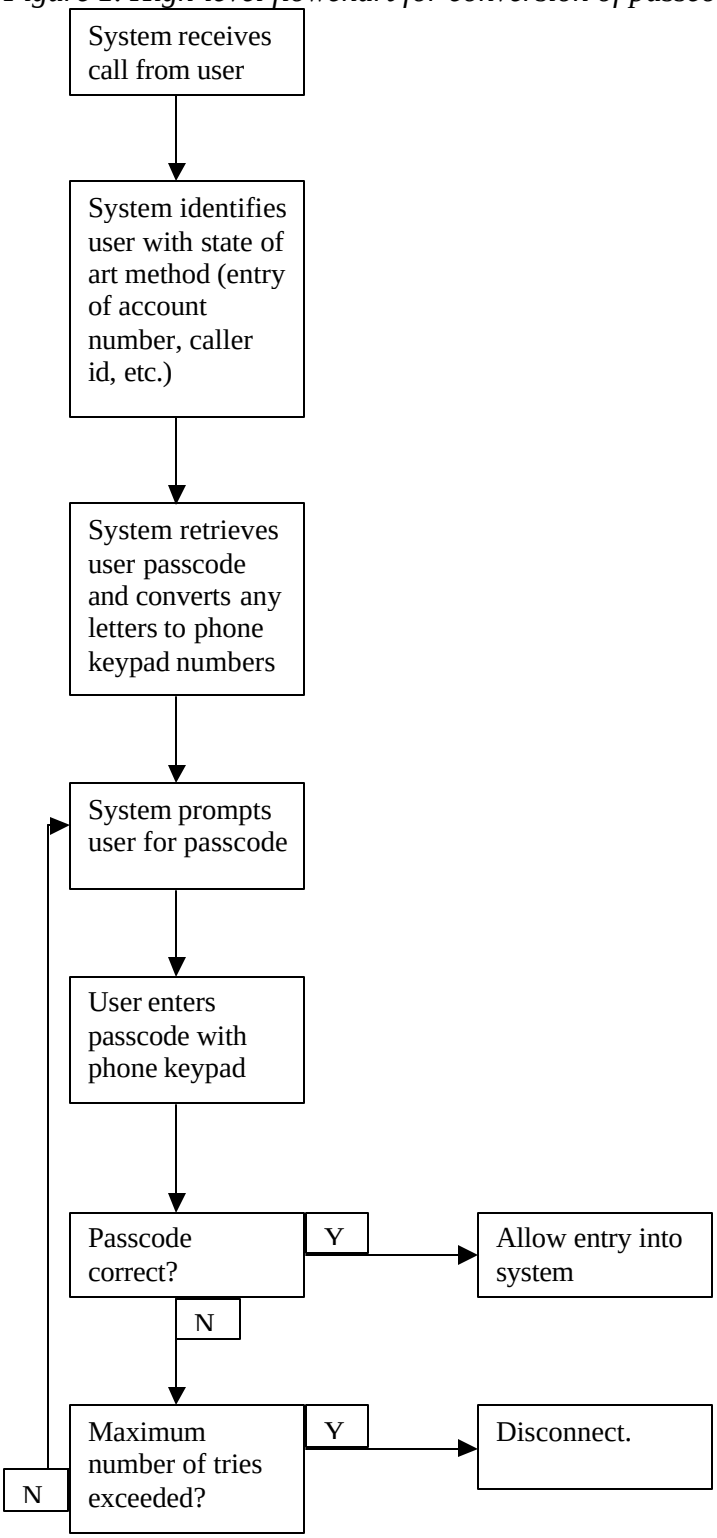
Table 1 shows the conversions:

Table 1. Mapping of letters to numbers on a telephone keypad

Letter	Number
A, B, C	2
D, E, F	3
G, H, I	4
J, K, L	5
M, N, O	6
P, Q, R, S	7
T, U, V	8
W, X, Y, Z	9

Also, here is a high-level flowchart of one way to instantiate the process:

Figure 1. High-level flowchart for conversion of passcodes from alphanumeric to numeric



Analysis and Results

.

Table 2 shows the increase in 'guessability' due to converting an alphanumeric passcode to a numeric passcode, but as long as the minimum number of passcode characters is four or greater, the guessability of the passcode will be very low (no more likely than 1/10,000).

Table 2. Comparison of guessability as a function of content and length for fixed length passcodes

# Chars	AlphaNum (CaseSens)	AlphaNum	Num	Num Odds
1	0.016129032258064500	0.027777777777777800	0.100000000000000000	1/10
2	0.000260145681581686	0.000771604938271605	0.010000000000000000	1/100
3	0.000004195898090027	0.000021433470507545	0.001000000000000000	1/1,000
4	0.000000067675775646	0.000000595374180765	0.000100000000000000	1/10,000
5	0.000000001091544768	0.000000016538171688	0.000010000000000000	1/100,000
6	0.000000000017605561	0.000000000459393658	0.000001000000000000	1/1,000,000
7	0.000000000000283961	0.000000000012760935	0.000000100000000000	1/10,000,000
8	0.000000000000004580	0.000000000000354470	0.000000010000000000	1/100,000,000
9	0.000000000000000074	0.000000000000009846	0.000000001000000000	1/1,000,000,000
10	0.000000000000000001	0.000000000000000274	0.000000000100000000	1/10,000,000,000

If the passcode is of variable length, then the guessability of the converted passcode becomes even lower. For example, suppose a system allows passcodes to be from 4 to 10 characters in length. Because there are 7 possible lengths, the odds of guessing at any given length are divided by 7, as shown in Table 3.

Table 3. Comparison of guessability as a function of content and length for variable length passcodes

# Chars	AlphaNum (CaseSens)	AlphaNum	Num	Num Odds
4	0.000000009667967949	0.000000085053454395	0.000014285714285714	1/70,000
5	0.000000000155934967	0.000000002362595955	0.000001428571428571	1/700,000
6	0.000000000002515080	0.000000000065627665	0.000000142857142857	1/7,000,000
7	0.000000000000040566	0.000000000001822991	0.000000014285714286	1/70,000,000
8	0.000000000000000654	0.000000000000050639	0.000000001428571429	1/700,000,000
9	0.000000000000000011	0.000000000000001407	0.000000000142857143	1/7,000,000,000
10	0.000000000000000000	0.000000000000000039	0.000000000014285714	1/70,000,000,000

Thus, it is clear that even with the increase in guessability caused by conversion from alphanumeric to numeric, the passcodes will still be very secure as long as the minimum passcode contains no fewer than four characters.

Recommendation

For callers who have legacy alphanumeric passcodes that contain more than three characters, allow the entry of the passcodes as numeric passcodes using the telephone keypad when using telephony versions of web applications.

References

Lewis, J. R., Potosnak, K. M., and Magyar, R. (1997). Keys and keyboards. In M. Helander, T. K. Landauer, and P. V. Prabhu (Eds.), *Handbook of Human-Computer Interaction* (pp. 1285-1315). Amsterdam: North-Holland.

.