TapSongs: Tapping Rhythm-Based Passwords on a Single Binary Sensor

Jacob O. Wobbrock
The Information School
DUB Group
University of Washington
Seattle, WA 98195-2840
wobbrock@u.washington.edu

ABSTRACT

TapSongs are presented, which enable user authentication on a single "binary" sensor (e.g., button) by matching the rhythm of tap down/up events to a jingle timing model created by the user. We describe our matching algorithm, which employs absolute match criteria and learns from successful logins. We also present a study of 10 subjects showing that after they created their own TapSong models from 12 examples (< 2 minutes), their subsequent login attempts were 83.2% successful. Furthermore, aural and visual eavesdropping of the experimenter's logins resulted in only 10.7% successful imposter logins by subjects. Even when subjects heard the target jingles played by a synthesized piano, they were only 19.4% successful logging in as imposters. These results are attributable to subtle but reliable individual differences in people's tapping, which are supported by prior findings in music psychology.

ACM Categories & Subject Descriptors: H5.2. [Information interfaces & presentation]: User interfaces—*Input devices & strategies*. K6.5. [Management of computing & information systems]: Security & protection—*Authentication*.

General Terms: Human Factors, Security.

Keywords: User authentication, password entry, songs, rhythm, jingles, tapping, temporal strings, binary sensors, mobile devices.

INTRODUCTION

Both in research [2] and as commercial products (Figure 1), tiny devices are appearing that have no keyboards and possibly even no screens. These devices may only have a single button or touch sensor. They may be so small that loss or theft become common. If such devices store private information like addresses, phone numbers, email, or personal data, how should users log in?

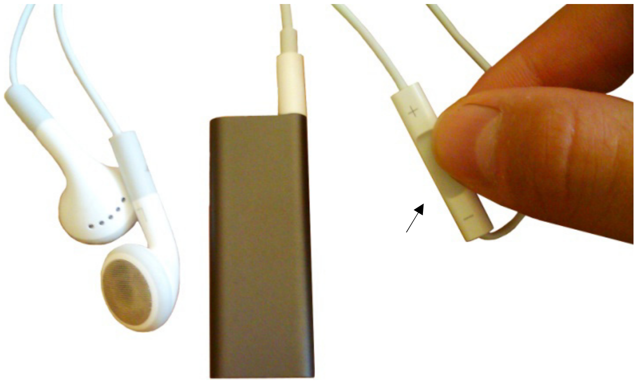

Figure 1. On the 3rd generation Apple iPod Shuffle there is no keyboard or screen. The primary input mechanism is a single button on the earbuds' cord. *TapSongs* allow rhythmic "passwords" to be entered on buttons like this, or on other "binary" sensors.

This paper presents a new method of "password" entry called a *TapSong*. Instead of text strings entered on multiple keys, a single sensor (e.g., button) can be used to tap a songlike rhythm, or *jingle*, to authenticate the user. The TapSong concept is supported by evidence from music psychology concerning humans' ability to perceive and perform rhythms [3,4], and from the mnemonic power of musical tunes, for example, as used in advertising [16]. We developed a simple pattern-matching algorithm that compares candidate jingles to user-created TapSong timing models. Our algorithm allows successful logins to further adapt TapSongs over time.

Although a security analysis is beyond the current scope, the threat model for TapSongs is similar to that of text passwords. Certainly, both can be cracked or stolen (e.g., [5,17]). But TapSongs differ from text passwords in important ways. First, a TapSong may be entered without exposing a device, e.g., by tapping anywhere on a touch screen in one's pocket. Second, if a TapSong is captured, it may be hard to portray, especially visually. Third, typing a stolen password is trivial, but tapping a stolen TapSong, even when the jingle is known with certainty, is not so easy; the performance of an attacker must be quite similar to that of another person.

Our results indicate that subjects can reliably tap their own jingle rhythms, and that individual differences, evident in prior studies of rhythm [1,15], make TapSongs promising even when compromised by eavesdropping or theft, both of which we simulate in our user study.

UIST'09, October 4–7, 2009, Victoria, British Columbia, Canada.

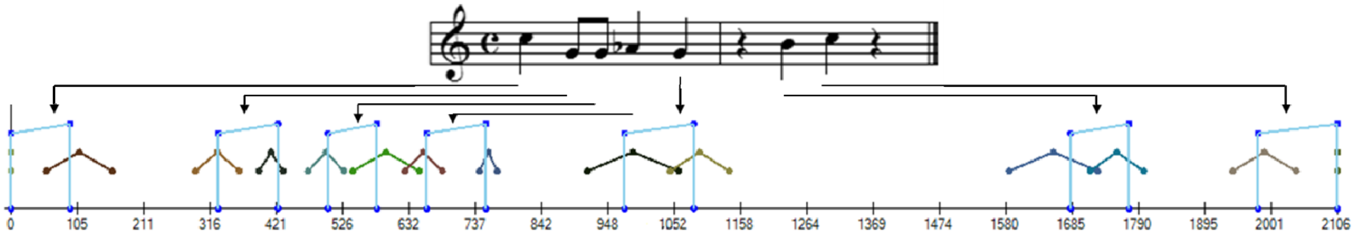

Figure 2. The famous jingle, *Shave and a Haircut, Two Bits* (Charles Hale,1899), has been used in everything from cartoons to secret knocks. A TapSong timing model has mean times and standard deviations shown with "^" marks. A matching candidate sequence is shown above it, with tap down/up events connected by crossbars falling within ±3 standard deviations of the corresponding model means. See also Figure 5.

RELATED WORK

Studies of rhythm have been conducted for over one hundred years [12]. Recent studies have isolated the brain structures responsible for enabling the perception and performance of rhythm [13]. Rhythm seems fundamental: studies show people's attempts to tap arrhythmic patterns inevitably produce rhythms [4]. Atonal sequences convey perceptible rhythms, and tapping replications by non-musicians are often not significantly different in timing from those of musicians [10]. A crucial aspect is that individual differences in tapping emerge [1,15]. For in-depth overviews of the psychology of rhythm, readers are directed to prior surveys [3,4].

Rarely has rhythm been used for computer input. The work most similar to ours is the use of rhythmic blink patterns [14]. Like our work, this research had subjects imagine songs as the basis for input. Unlike our work, however, the system used a nearest-neighbor classifier that required many training examples and did not define absolute accept/reject criteria. Also, it did not permit musical rests because it used discrete blinks for notes of all lengths, not separate down/up events defining note durations (see Figure 5). The work was applied to computer security [15] in an attempt to identify people based on the rhythm with which they blinked the same song. With TapSongs, our task is simpler: we only need to see if an inputted rhythm matches a timing model already stored on a device. Our contribution is therefore how to enter, model, and match a temporal string on a binary sensor for authentication, not how to use tapping as a biometric for identification.

Rhythm was also used for *awareLESS* input [6], where subjects made finger pressure pulses while observers tried to infer rhythms from finger motions. Atonal rhythmic tapping on the spacebar was also used as input to a music information retrieval system [9]. Finally, some commercial products attempt to increase password security by analyzing the timing with which text passwords are typed [7].

THE TAPSONG TECHNIQUE

People commonly tap the edges of tables, the covers of laptops, and paper notepads. These rhythms are often catchy phrases from songs, or *jingles*. Perhaps the most famous jingle is *Shave and a Haircut, Two Bits* (Figure 2). The key idea behind TapSongs is to allow such jingles to serve as text-less passwords that can be entered on any "binary" sensor, i.e., a sensor that simply reports down/up events. Examples are buttons, keys, flip-switches, touch screens, and simple capacitive touch-sensors.

Modeling a TapSong Rhythm

Although the human "time-sense" [12] is quite robust [13], a user will not repeatedly enter a tap sequence with the exact same timing. A TapSong timing model must capture the essential rhythm from examples and also their inherent variability. We know from prior work [8,11], for example, that events encoding longer time intervals will exhibit more variation in accordance with Weber's law.

After a user enters a small set (5-15) of tap sequences reflecting a given rhythm (e.g., Figure 2), these sequences are linearly time-warped to begin and end in sync, and then averaged so that the timing model contains the mean down or up time (T_μ) at each position. Each mean is coupled with its standard deviation (T_σ), thereby retaining the variability that is essential to matching and reflective of Weber's law.

We investigated how many examples were required for the standard deviations around each mean to stabilize. After just 5 examples, the percent change in standard deviation (ΔP) remained less than about 10% (Figure 3).

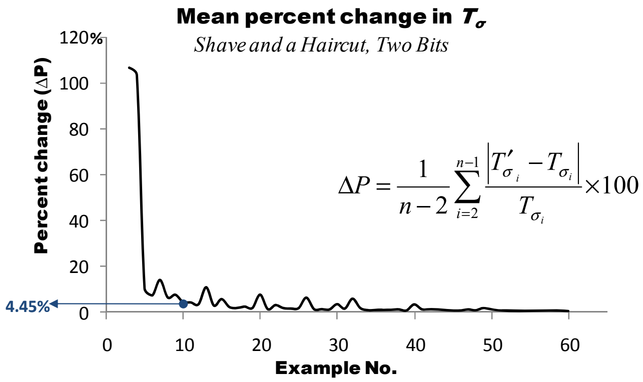

Figure 3. The mean percentage change ΔP of the standard deviation of event times T_σ as the timing model for *Shave and a Haircut, Two Bits* (Figure 2) absorbs each successive sequence. T'_{σ_i} is the standard deviation around the i^{th} input event in timing model T', which immediately follows T. The first and last input events are ignored because linear time-warping forces them to align.

Although it takes less than 2 minutes to create a TapSong timing model, we could avoid the need for training by using premade standard deviations around each tap, scaling them according to Weber's law. Alternatively, we could avoid having to train a TapSong in the first place if a user is allowed to select a song from a list, or even to upload a jingle from an audio file. Rhythm-extraction software could provide a model that reflects the timing of the jingle's notes, again with appropriate standard deviations. In any case, as the user logs in over time, the premade values will be replaced as a TapSong adapts (described below).

One benefit of TapSongs is that merely knowing a song title does not necessarily reveal its rhythm, or even its number of taps. Musical phrases come in many variations, which as time series appear quite distinct. For example, a variation of *Shave and a Haircut, Two Bits* is shown in Figure 4.

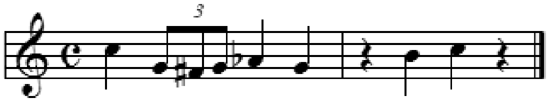

Figure 4. A variation of *Shave and a Haircut, Two Bits* that involves a triplet. It would not match the timing model shown in Figure 2.

Logging In with a TapSong

When a user taps a rhythm in an attempt to log in, the tap sequence is first time-warped such that it begins and ends with the timing model (see Figure 2). The time-warp is linear, not dynamic, which ensures that that the temporal relationships among rhythmic events are preserved. Then the candidate sequence C and TapSong timing model T are said to be a match if they satisfy the three addends of Eq. 1:

$$\left(|C| = |T|\right) \wedge \left(\tfrac{2}{3}T_{ms} \le C_{ms} \le \tfrac{4}{3}T_{ms}\right) \wedge \left(\forall i : \left|C_i - T_{\mu_i}\right| \le 3T_{\sigma_i}\right) \quad (1)$$

The first condition is that the number of tap events agrees in C and T. The second condition requires that the *unwarped* duration (in ms) of C to be within one-third of T. The third condition requires that every candidate down/up event C_i is no more than three standard deviations from its corresponding model mean T_{μ_i}. (Recall that standard deviation T_{σ_i} is specific to mean T_{μ_i}.) An example of such comparisons is shown in Figure 5.

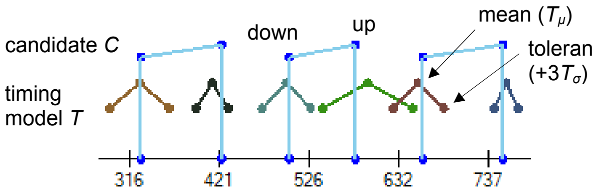

Figure 5. An excerpt from Figure 2 showing events from the candidate C being compared to corresponding means (T_μ) and standard deviations (T_σ) in timing model T. In this excerpt, all events C_i fall within ±3T_{σ_i} of each T_{μ_i}, permitting authentication.

TapSong Adaptation over Time

The choice of ±3T_σ was made for reasons both pragmatic and theoretical. Pragmatically, we found that this much tolerance made it possible to match all input events with a properly executed tap sequence without being too forgiving. The other reason was theoretical. Users may speed up as they become more familiar with their TapSong [11], or they may gradually transition from tapping staccato to legato. Therefore, TapSongs must adapt over time to subtle but reliable timing changes. With each successful login, TapSongs can absorb a new sequence into their model, computing new means (T_μ) and standard deviations (T_σ). However, we must be careful: we do not want to grow or shrink a TapSong's standard deviations *solely* by virtue of our mathematical policy. For example, if ±1T_σ were the criterion for absorption, we would only ever *shrink* our standard deviations, making it harder to log in over time! We must therefore adopt a tolerance range such that the existing standard deviations will be unaffected except by consistent trends in user behavior.

We know from prior research [8] that human timing error will be Gaussian around our timing means. A Gaussian distribution reaches approximately zero about three standard deviations from its mean (Figure 6), so if we use ±3T_σ as our criterion for absorption, we will preserve the existing standard deviation of event times. We confirmed this outcome with numerous Monte Carlo simulations.

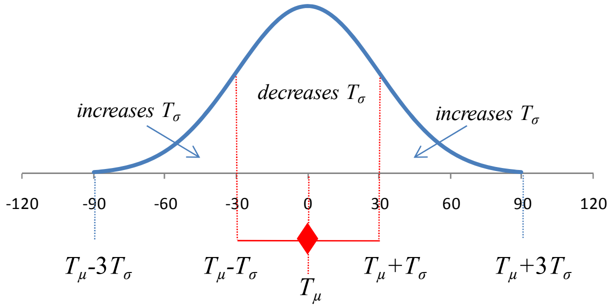

Figure 6. A mean input event time at T_μ = 0 ms with surrounding standard deviation T_σ = 30 ms. Allowing TapSongs whose events all fall within T_μ ± 3T_σ = ±90 ms to be absorbed into the timing model will retain T_σ = 30 ms as the standard deviation, assuming a normal distribution of events around the mean [8].

Differentiating Among Multiple Matches

TapSongs are like text passwords in that a single TapSong will be tied to a single device. In such cases, a tap sequence only needs to be compared to one TapSong timing model, and a match is definable in absolute terms (Eq. 1). However, there may be cases when a tap sequence must be compared to a *set* of timing models, such as when logging into a shared device. In such cases, it will be necessary to *score* comparisons in the off-chance that a candidate matches multiple TapSongs. We devised a distance measure for which $D = 0$ would mean a perfect match in Eq. 2:

$$D = \frac{1}{n-2}\sum_{i=2}^{n-1}\frac{\left|C_i - T_{\mu_i}\right|}{3T_{\sigma_i}} \quad (2)$$

Eq. 2 calculates the average percent deviation between C and T, excluding the first and last events. A distance of $D = 0.50$ indicates that on average, a candidate's events fell at 50% of the tolerable deviation from the timing means.

USER STUDY

We conducted a user study of 10 subjects (6 female). The mean age was 36.0 years (*SD* 13.7). Five subjects could play a musical instrument. In Part I, subjects were randomly assigned one of 15 jingles consisting of 6-8 notes. Examples were opening lines from *Itsy Bitsy Spider*, *Jingle Bells*, *Old MacDonald*, and *London Bridge*. Subjects listened to a synthesized piano melody and then tapped its rhythm on a mouse button 12 times to create a TapSong timing model (< 2 minutes). Then they logged in against this model 25 times with adaptation enabled. Subjects were 83.2% (*SD* 14.2%) successful, giving a rate of true positives.

Parts II and III of the experiment examined false positives. In Part II, subjects aurally and visually eavesdropped from 3 feet away while the experimenter tapped each of the 15 jingles. Subjects were not told the jingles' names. Although in practice, tap down/up events may be difficult to overhear (e.g., a finger lifting from a touch screen is nearly silent), the experimenter used a loud-clicking mouse button that made down/up events audible. Timing models were created earlier by the experimenter using 12 examples per jingle based on the rhythms of the synthesized piano melodies. After each of the 15 logins by the experimenter, the subject attempted to replicate. Mean login success was only 10.7% (*SD* 11.4%). About 77.4% (*SD* 27.5%) of these entries had the correct number of taps, but login success was still low at 12.0% (*SD* 12.5%). Subjects felt eavesdropping was difficult because they did not know the song names or melodies, making it impossible to "play" the underlying tunes in their heads while listening to the experimenter's taps.

Part III of the experiment simulated a stolen password. For each of the 15 jingles, subjects were told the jingle's name and played its piano melody before logging in. Still, login success was only 19.4% (*SD* 11.5%). Most entries had the correct number of taps (91.0%, *SD* 17.8%), but it seems that individual differences arising partly in how staccato or legato notes were tapped resulted in TapSongs being difficult to match by someone other than their author.

CONCLUSION

TapSongs allow text-less user authentication on a single binary sensor. TapSongs are user-specific, adaptable, and implementable on almost any hardware. It seems that individual differences arising in people's rhythmic tapping give TapSongs some ability to distinguish between their authors and imposters.

Future work should formally quantify TapSong "password strength," which depends on many factors, including number of notes, heterogeneity of note and rest lengths, and variance in the timing model. Security can be strengthened or weakened by the designer by adjusting Eq. 1, or by the user by creating TapSong timing models from intentionally more or less varied input sequences. Future work should also examine TapSong memorability, especially for TapSongs that have not been entered for days or weeks.

ACKNOWLEDGEMENTS

Thanks to Daniel Avrahami, Parmit K. Chilana, Chandrika Jayant and Joseph Tennis. This work was supported in part by Intel Research Seattle and Microsoft Research.

REFERENCES

[1] Aschersleben, G. and Prinz, W. (1995) Synchronizing actions with events: The role of sensory information. *Perception and Psychophysics 57* (3), 305-317.

[2] Baudisch, P. and Chu, G. (2009) Back-of-device interaction allows creating very small touch devices. *Proc. CHI '09*. New York: ACM Press, 1923-1932.

[3] Clarke, E.F. (1999) Rhythm and timing in music. In *The Psychology of Music (2nd)*, D. Deutsch (ed.). San Diego: Academic Press, 473-500.

[4] Fraisse, P. (1982) Rhythm and tempo. In *The Psychology of Music (1st)*, D. Deutsch (ed.). New York: Academic Press, 149-180.

[5] Kuo, C., Romanosky, S. and Cranor, L.F. (2006) Human selection of mnemonic phrase-based passwords. *Proc. SOUPS '06*. New York: ACM Press, 67-78.

[6] Manabe, H. and Fukumoto, M. (2007) AwareLESS authentication: Insensible input based authentication. *Extended Abstracts CHI '07*. New York: ACM Press, 2561-2566.

[7] Marriott, M. (2000) New encryption strategy: Typing rhythm. *New York Times*. Technology, June 29, G-9.

[8] Mates, J., Müller, U., Radil, T. and Pöppel, E. (1994) Temporal integration in sensorimotor synchronization. *J. Cognitive Neuroscience 6* (4), 332-340.

[9] Peters, G., Anthony, C. and Schwartz, M. (2005) Song search and retrieval by tapping. *Proc. AAAI '05*. Menlo Park, CA: AAAI Press, 1696-1697.

[10] Snyder, J. and Krumhansl, C.L. (2001) Tapping to ragtime: Cues to pulse finding. *Music Perception 18* (4), 455-489.

[11] Sternad, D., Dean, W.J. and Newell, K.M. (2000) Force and timing variability in rhythmic unimanual tapping. *J. Motor Behavior 32* (3), 249-267.

[12] Stevens, L.T. (1886) On the time-sense. *Mind 11* (43), 393-404.

[13] Thaut, M.H., Kenyon, G.P., Schauer, M.L. and McIntosh, G.C. (1999) The connection between rhythmicity and brain function. *IEEE Engineering in Medicine and Biology 18* (2), 101-108.

[14] Westeyn, T. and Starner, T. (2004) Recognizing song-based blink patterns: Applications for restricted and universal access. *Proc. FGR '04*. Washington, D.C.: IEEE Computer Society, 717-722.

[15] Westeyn, T., Pesti, P., Park, K.-H. and Starner, T. (2005) Biometric identification using song-based blink patterns. *Proc. HCI Int'l '05*. Mahwah, NJ: Lawrence Erlbaum.

[16] Yalch, R.F. (1991) Memory in a jingle jungle: Music as a mnemonic device in communicating advertising slogans. *J. Applied Psychology 76* (2), 268-275.

[17] Zhuang, L., Zhou, F. and Tygar, J.D. (2005) Keyboard acoustic emanations revisited. *Proc. CCS '05*. New York: ACM Press, 373-382.