Timing Attacks on Cognitive Authentication Schemes

Mario Čagalj, Toni Perković, *Member, IEEE,* Marin Bugarić

***Abstract*—Classical password/PIN-based authentication methods have proven to be vulnerable to a broad range of observation attacks (such as key-logging, video-recording or shoulder surfing attacks). In order to mitigate these attacks, a number of solutions have been proposed, most of them being cognitive authentication schemes (challenge-response protocols that require users to perform some kind of cognitive operations).**

In this paper we show successful passive side-channel timing attacks on two cognitive authentication schemes, a well-known Hopper-Blum (HB) protocol and a US patent Mod10 method, previously believed to be secure against observation attacks. As we show, the main security weakness of these methods comes from detectable variations in the user's cognitive load that results from cognitive operations during the authentication procedure. We carried out theoretical analysis of both Mod10 and HB methods, as well as an experimental user study of Mod10 method with 58 participants to validate the results of our timing attacks.

We also propose security enhancements of these schemes aimed to mitigate the timing side-channel attacks. The proposed enhancements show the existence of a strong tradeoff between security and usability, indicating that the security of cognitive authentication schemes comes at a non-negligible usability cost (e.g., increased overall login time). For this reason, the designers of new cognitive authentication schemes should not ignore possible threats induced by side-channel timing attacks.

***Keywords*—Authentication, cognitive authentication schemes, human factors, observation attack, side-channel timing attack**

I. Introduction

Internet services such as social networks, e-banking, email, cloud services, blogs, all require some form of user authentication. Despite the availability of advanced authentication technologies such as smart cards, biometrics or USB tokens [1], passwords and PINs are still the most prevalent form of user authentication. This is primarily due to simplicity and low cost of their creation, maintenance and revocation. At the same time, traditional password/PIN-based authentication systems are vulnerable to different forms of *observation attacks*, ranging from a simple shoulder-surfing [2] to a more advanced keylogging and camera recording attacks [3].

A number of solutions have been proposed to strengthen password/PIN-based solutions against such observation attacks. Most solutions use challenge-response protocols where users are required to perform some form of a *cognitive task* (e.g., small additions, recognition, visual recall) [4], [5], [6], [7], prior to being granted an access. More specifically, in each authentication session, the user gives back the response r_i to a number of challenges c_i based on the knowledge of the shared key s_i; the user performs a cognitive operation $f(c_i, s_i)$ to calculate the response r_i. While mitigating certain observations attacks, these *cognitive authentication* schemes inevitably incur a higher cognitive load on a user (due to cognitive operations) compared to traditional password-based authentication. For this reason, besides the protocol security, the designers of such protocols put a lot of attention on the usability (e.g., try to minimize the overall user's load). However, subtle variations in the difficulty of cognitive operations while calculating the response $r_i = f(c_i, s_i)$ could lead to the asymmetry in the user's cognitive load. Although the designers often neglect these subtle variations, they could be used for *cognitive-asymmetry side-channel* attacks. As we show in this paper, cognitive asymmetry can be exploited in a passive side-channel timing attack to partially or completely recover secret PINs/passwords. Please note that this is different from the timing attacks that exploit asymmetry of physical user interfaces such as keyboards [8] or asymmetry of virtual interfaces (i.e., asymmetric graphical layouts), as found in Undercover [5], [9].

To demonstrate cognitive-asymmetry side-channels, in this paper we analyze the security of two cognitive authentication schemes that were designed to mitigate shoulder-surfing attacks. More precisely, we present successful side-channel timing attacks on two authentication schemes: a Mod10 [10] scheme[1] and a well-known Hopper-Blum (HB) protocol [12]. Our attacks exploit detectable variations in the user's cognitive load during the course of the authentication protocol (variations in user's reaction times when responding to different challenges).

For the Mod10 authentication scheme, that is based on a secret PIN and modulo 10 arithmetic, we show that it is possible to reduce the entropy of each PIN digit by approximately 60%. We validated the results of timing attack through a user study involving 58 participants. In the case of the Hopper-Blum protocol, the main security result in [12] states that the protocol is secure against computationally bounded

T. Perković is the corresponding author with the Department of Electrical Engineering, FESB, University of Split, 21000 Croatia. E-mail: toperkov@fesb.hr.

M. Čagalj and M. Bugarić are affiliated with the Department of Electrical Engineering, FESB, University of Split, 21000 Croatia. E-mail: {mcagalj, marin.bugaric}@fesb.hr.

[1]The Mod10 scheme, originally proposed in a US patent [10], has been suggested for use several times in the literature due to its simplicity and fast login times (around 10 seconds) [11], [6].

eavesdropping adversaries. As we will show, this proof holds in the setting where the adversary does not obtain the timing information. Based on the assumptions of our theoretical model, a passive attacker can completely recover the user's secret after observing a reasonably small number of authentication rounds.

HB and Mod10 schemes are representatives of authentication paradigms that require users to perform some form of a cognitive task, such as addition, visual recall and counting, multiplication, modulo operations etc. However, there is a plethora of protocols that have similarities with HB and Mod10 schemes, such as Foxtail [13], Asghar, Pieprzyk and Wang (APW) [14] protocols and virtual password scheme [15], where users are also required to perform a cognitive operation $f(c_i, s_i)$ to compute the response r_i. We believe that these schemes are also vulnerable to cognitive-asymmetry side channels. Moreover, we stress here the HB protocol has been extended for use in RFID systems [16] and that it should be explored whether our attack is applicable in this context too (under the assumptions of our authentication model).

We further show that eliminating the timing side channels in Mod10 and HB schemes necessarily comes at a non-negligible usability cost (i.e., longer authentication times and/or longer PINs/passwords). This is an important result since a usability score of a given authentication scheme is correlated with its authentication times and the PIN/password size. Therefore, when designing and evaluating a new scheme, designers cannot neglect cognitive-asymmetry side-channel timing attacks.

II. The Attacker Model

In this paper we consider a *passive attacker* who eavesdrops on all public communication between the user and the end system (public challenges and/or responses). The attacker also has the capability of recording the user's reaction time during the course of the login procedure by, for example, using key-logging malware, a simple camera or an accelerator within a smartphone as a high bandwidth side channel [17].

III. Timing Attack on Mod10 Method

Mod10 scheme implements a *one-time pad* paradigm. To enter the ith digit d_i of their PIN, the users receive a challenge c_i (one digit long) selected uniformly at random from the set $(0, \ldots, 9)$ via a protected channel (e.g., earphones), add two digits modulo 10 ($r_i = (c_i + d_i) \bmod 10$) and enter back the outcome r_i via the public channel (e.g., a numerical pad). The main intuition that led us to investigate cognitive-asymmetry side channels in Mod10 scheme is simple: during the authentication process, a user will take less time to respond to a challenge if the challenge digit c_i is 0, 1 or 2.

The Mod10 scheme was originally proposed in a US patent [10]. The security of Mod10 scheme against the timing attack recently analyzed in ASIACCS'13 paper [18] (named NumPad scheme) does not reveal weakness to the cognitive-asymmetry side channels. However, our preliminary study of the Mod10 scheme in [6] led to the discovery of variations in the user's cognitive load (response times) while responding to different challenges, which allows the attacker to reduce the PIN space. The main intuition why NumPad scheme proposed in [18] does not find any weaknesses to side-channel timing attack is because NumPad timing attack analysis is based on observation of limited number of features, i.e. the average response times per PIN digit that have large variances (induce high noise). To eliminate the effect of noise in our study we focus on observing only the fastest login times. This allows us to use multiple features and create a set of unique patterns for every PIN digit.

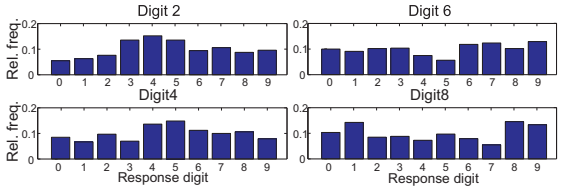


Fig. 1. Relative frequency with which a given response digit appears within $\ell = 4$ fastest response digits, for PIN digits: 2, 4, 6 and 8.

In this paper we present an extensive analysis of Mod10 scheme against side-channel timing attacks verified via user studies in our own implementation of Mod10. Please note that challenges in Mod10 are broadcast over a secret channel, therefore, in our attacker model we assume that the user sees the next challenge immediately after he responds to the current one. In this way, the adversary is able to infer when the user started working on each new challenge and thus obtain the timing information.

We implemented the Mod10 scheme as a web application and collected test statistics from 58 users, where each user was asked to login at least 30 times. An overall login success rate (out of 2491 login attempts) was about 85%, while the average login time was 10.39 seconds.

It is known that arithmetic operations with small-valued numbers (additions with challenges from the set $\{0, 1, 2\}$) score faster response times [19], [20], [21], [22]. These observations can be seen in Fig. 1, where we plot the relative frequencies for PIN digits 2, 4, 6 and 8 that is generated in such a way that for the fixed PIN digit d_i we count how many users (with PIN digit(s) d_i) have a given response digit within their ℓ "fastest" response digits (in our case ℓ = 4).

This cognitive asymmetry (variations in the user cognitive load) in Mod10 scheme inspired us to run a side-channel timing attack. On the high level, the timing attack is based on the observation of two attributes: a vector $\mathbf{r} = (r_1, r_2, \ldots, r_\ell)$ representing the $\ell \leq 10$ fastest response digits, and a scalar t_r representing the fastest response time, both for the unknown/sought class (PIN digit). Our decision to use fastest response digits and response times as a filter for our timing attack was based on the intuition that users cannot provide answers faster than their cognitive capabilities allow them (as opposite to slow answers). The timing attack presents a classification problem in which the observed data is assigned to one of the predefined classes (PIN digits). These decisions are based on two classifiers and the approach based on the *naive Bayesian classifier*.

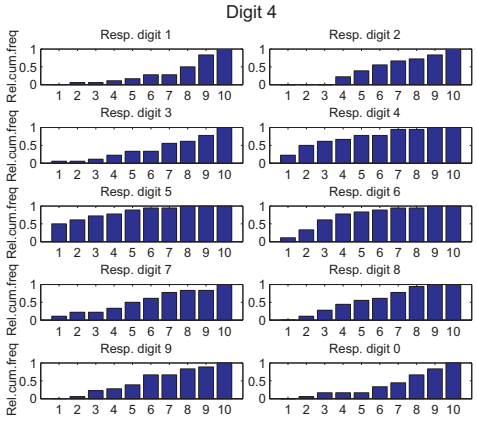


Fig. 2. Relative cumulative frequency with which a given response digit appears within ℓ fastest response digits ($\ell \in \{1, \ldots, 10\}$), for PIN digit 4.

A. Classification-based Timing Attack on Mod10

In our classification we used a continuous output from each classifier that was normalized to the interval [0 1] (posterior probability). These values (for each PIN digit) were combined using an algebraic combination rule [23]. We used a sum decision rule that combines posterior probability from each classifier, because it achieves the best classification results. We first describe the implementation of each classifier.

Algorithm 1 Estimating the class conditional probability $P(r_m|d_i)$ from the training set

1) Initialize: $\mathbf{C}_j^i = \mathbf{0}$,$\forall r_j^i$, $\forall d_i$,
2) Repeat $\forall d_i$: if r_j^i in ℓ fastest, then $C_{j,m}^i = C_{j,m}^i + 1$, $\forall m \in \{\ell, \ldots, 10\}$,
3) Calculate: $P(r_m|d_i) \backsimeq C_{j,\ell}^i / C_{j,10}^i$

1) Classifier Trained on Fastest Response Digits: The first classifier exploits the cognitive asymmetry of "easy additions" with small-valued challenges (e.g., 0, 1 and 2) that achieve fast response times. To learn these distributions of response digits r_j^i ($r_j^i \in \{0, \ldots, 9\}$) for every class value (PIN digit d_i), the classifier is first trained on the login results from 57 out of 58 users (*leave-one-out crossvalidation* technique). For every PIN digit d_i ($d_i \in \{0, \ldots, 9\}$) we repeat the following steps, as described by Algorithm 1: for every response digit r_j^i ($r_j^i \in \{0, \ldots, 9\}$) create a vector $\mathbf{C}_j^i$ of 10 counters ($\mathbf{C}_j^i = \mathbf{0}$). After that, for all users from the training set whose PIN comprises digit d_i, and for all such digits d_i for the given user repeat the following steps: first, rank the response digits r_j^i (for the given PIN digit d_i) according to the fastest response times. If the response digit r_j^i falls within ℓ fastest response digits, increment the counters $C_{j,\ell}^i$ to $C_{j,10}^i$ by one. As a result we obtain, for every PIN digit d_i, the cumulative frequency $C_{j,\ell}^i$ that indicates the number of times a given response digit r_j^i falls within ℓ fastest response digits. Dividing $C_{j,\ell}^i$ by $C_{j,10}^i$ we get the corresponding relative cumulative frequency. Fig. 2 shows the relative cumulative frequency of all response digits for PIN digit 4. Note that the response digits $4, 5$ and 6 are characterized by higher relative cumulative frequencies at small values of ℓ. This indicates that those response digits generally have faster response times for the corresponding PIN digit [24] (*easy additions* with $0, 1$ and 2).

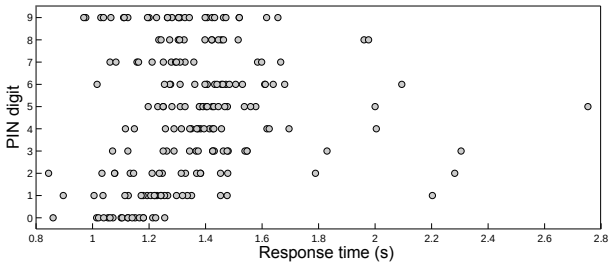


Fig. 3. The calculation results indicate that additions with small single integers achieve fastest response times (PIN digits 0,1 and 2).

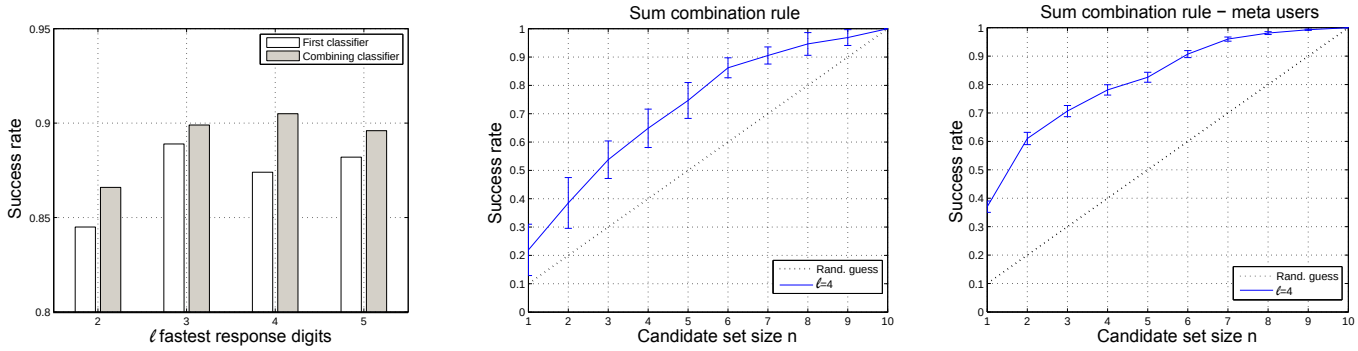
Our first classifier uses the resulting frequency distributions to perform the classification task of the feature vector $\mathbf{r} = (r_1, r_2, \ldots, r_\ell)$ of $\ell \leq 10$ fastest response digits (for the given PIN digit position). Given the feature vector $\mathbf{r}$, our classifier first estimates the posterior probability $P(d_i|\mathbf{r})$ for all PIN digits d_i and then selects into the final *candidate set* $n \leq 10$ digits d_i that have the highest posterior probabilities. We use the classical naive Bayes technique/assumption where we assume that conditioned on a class d_i, the feature vector elements $(r_1, r_2, \ldots, r_\ell)$ are mutually independent. Then we can express $P(d_i|\mathbf{r})$ as follows:

$$P(d_i|\mathbf{r}) = \frac{\prod_{m=1}^{\ell} P(r_m|d_i) \cdot P(d_i)}{\sum_{l=1}^{10} \prod_{m=1}^{\ell} P(r_m|d_l) \cdot P(d_l)}, \qquad (1)$$

with $P(d_i) = 1/10$ for all the digits d_i. As described in Algorithm 1, the class conditional probabilities $P(r_m|d_i)$ are estimated from the relative cumulative frequency distributions (obtained in the training phase).

2) Classification Based on Fastest Response Times: In our second (Bayesian) classifier we use only the fastest response time (t_r) among all the response digits. The second classifier exploits the cognitive asymmetry in Mod10 method by using the phenomenon called *problem-size effect* in simple arithmetic operations [22]. Basically, direct mental mapping technique that is characteristic for small-valued additions (e.g., $2 + 3$) is faster than the procedural techniques of addition that use large-valued additions (e.g., $7 + 4$) [19]. The results of fastest response times for all PIN digits in Fig. 3 show that additions with small one-digit integers achieve faster response times. Note that users with PIN digit 0 on average achieve faster response times (the response equals to the challenge).

In Fig. 3 we can see that multiple classes (PIN digits) have overlapping distributions of response times, and accordingly, the observed fastest response time t_r can be attributed to multiple basic classes (PIN digits). Similarly to [25], instead of assigning an observed feature t_r to a specific class d_i, it may be assigned to the meta-class C_j that comprises of multiple basic classes. Our classifier uses two meta-classes C_f and C_s,



(a) (b) (c)

Fig. 4. (a) Combining classifier achieves better results than the first classifier. (b) Effectiveness of the sum combination rule that combines the results of two classifiers. Dotted line presents guessing attack. (c) Effectiveness of the sum combination rule that aggregates the results of m=3 users into a single meta-user.

TABLE I. EFFECTIVENESS OF THE SECOND CLASSIFIER TRAINED ON THE FASTEST RESPONSE TIMES.

PIN digit	0	1	2	3	4	5	6	7	8	9
Meta-class	C_f	C_f	C_f	C_s	C_s	C_s	C_s	C_f	C_s	C_f
Probability of correct assignment	0.78	0.62	0.48	0.53	0.61	0.66	0.62	0.52	0.64	0.58

where C_f is comprised of "fast response digits" $\{0, 1, 2, 7, 9\}$, while C_s is comprised of "slow response digits" $\{3, 4, 5, 6, 8\}$. We first calculate the posterior probabilities $P(C_j|t_r)$ for $j \in \{f, s\}$ using Bayes' theorem, i.e.:

$$P(C_j|t_r) = \frac{P(t_r|C_j)P(C_j)}{P(t_r)}, \tag{2}$$

where $P(C_f) = P(C_s) = 1/2$ for the prior probabilities. Concerning the class-conditional probabilities $P(t_r|C_j)$ we use Flexible Bayes learning algorithm for estimating continuous distributions in Bayesian classifier [26] where conditional probabilities are estimated using the *kernel estimation* with Gaussian kernel. By assuming that every digit d_i within the meta-class C_j is uniformly distributed [27], the posterior probabilities $P(d_i|t_r)$ for every class within the subset are obtained using the following rule:

$$\text{if} \quad d_i \in C_j \quad \text{then} \quad P(d_i|t_r) = \frac{P(C_j|t_r)}{|C_j|}. \tag{3}$$

3) Classifier Combination Using a Sum Decision Rule: We use the sum decision rule to combine the output posterior probabilities from each classifier, because it achieves the best classification results (compared to min/max/product rule) [23]. The posterior probabilities can be combined as follows [23]:

$$P(d_i|\mathbf{r}, t_r) \approx const. + P(d_i|\mathbf{r}) + P(d_i|t_r). \tag{4}$$

As before, using the resulting posterior probabilities $P(d_i|\mathbf{r}, t_r)$ we can rank the classes d_i and select the ones with the highest rank into the final candidate set.

4) Effectiveness of the Classifiers: We can measure the effectiveness of a given classifier by estimating the probability that the unknown PIN digit will fall into the output candidate set and by comparing it against a random guessing strategy. Please note that a random guessing strategy will be successful with probability at most $n/10$ for a candidate set of size n. For $\ell = 4$ and $n = 4$ the first classifier achieves 65% better results than pure random guessing. Moreover, for $n = 7$ the classifier holds a sought PIN digit with probability 0.88 (Fig. 4(a)).

The effectiveness of the second classifier is summarized in Table I. It shows the probability that the PIN digit d_i will be assigned to the correct meta-class. As expected, digits with the fastest response times within the meta-class C_f (e.g., 0, 1 in Fig. 3) and digits with the slowest response times within C_s (e.g., 5, 6 and 8 in Fig. 3) have higher probability to be classified to the correct meta-class.

After combining the results from each classifier using the sum combination rule, we can see a slightly better performance in the classification correctness as shown in Fig. 4(b). It shows the probability that a tested unknown digit d_i will fall in a candidate set of size n. For example, for the same parameters as before ($\ell = 4$ and $n = 4$) our combining classifier performs 72% better than random guessing. Generally, the combining classifier performs better than the first classifier for almost all candidate set sizes n; in Fig. 4(a) we compare the combining against the first classifier for a candidate set size of $n = 7$.

We next estimate the amount of information about an unknown PIN that the attacker can extract using our combining classifier. Let us denote with n^* a candidate set size that holds unknown PIN digit with the probability above 90%. Then, $\log_2(10/n^*)$ is the approximate number of extracted bits per PIN digit. As shown in Fig. 4(a) the combining classifier for a candidate set of the size $n = 7$ and $\ell = 4$ fastest response digits will comprise the sought unknown PIN digit with probability slightly over 90%. Therefore, the attacker can extract approximately $\log_2(10/7) \approx 0.5$ bits per PIN digit. Accordingly, the attacker can extract approximately 2 bits of information about an unknown 4 digit PIN, i.e., the PIN entropy is reduced from $4 \log_2 10 \approx 13.3$ bits to 11.3 bits.

By observing more login sessions (but still a polynomial number) we expect even more significant reduction in the PIN entropy. To prove our assumption we aggregated the results from $m = 3$ randomly selected users who had the same PIN

digit and created a metauser with 90 successful logins (on average). Using the same classification technique, we were able to reduce a candidate set size to $n^* = 6$ digits, as shown in Fig. 4(c). Therefore, the attacker can extract approximately 0.73 bits per PIN digit by observing more login sessions.

We assumed that the maximum amount of information the attacker can learn about the PIN digit was based on the results obtained by applying our combining classifier. We will next estimate the conditional entropy for the given response digit r_i and response time t_r:

$$H(D|R = r_i, T_r = t_r) = -\sum_{j=1}^{10} P(d_j|r_i, t_r) \log_2 P(d_j|r_i, t_r). \quad (5)$$

The probability $P(d_j|r_i, t_r)$ can be calculated as follows:

$$P(d_j|r_i, t_r) = \frac{p(t_r|d_j, r_i)}{\sum_{l=1}^{10} p(t_r|d_l, r_i)}. \quad (6)$$

We estimate the probability $p(t_r|d_j, r_i)$ using the data collected in our tests, by assuming that they have Gaussian distribution. To estimate the expected conditional entropy $H(D|R, t_r)$ over all possible response digits R we do the following:

$$H(D|R, T_r) = \frac{1}{100} \sum_{i=1}^{10} \sum_{j=1}^{10} \int_{t_r=0.8}^{3.5} p(t_r|r_i, d_j) \cdot H(D|r_i, t_r) dt_r. \quad (7)$$

By evaluating the Eq. (7), we finally obtain $H(D|R, T_r) \approx 2$ bits; i.e. the attacker can extract approximately 2 bits of information about the unknown PIN digit.

IV. Timing Attack on the Hopper-Blum (HB) Authentication Protocol

Hopper-Blum (HB) protocol became popular as one of the first human authentication protocols that was provably secure against passive observation attacks [12]. The HB protocol implements k-out-of-n paradigm, where the user (U) and the end system (S) share a secret binary vector $\mathbf{s} \in \{0, 1\}^n$. The Hamming weight of the vector $\mathbf{s}$ is k, that is, the secret vector $\mathbf{s}$ has k non-zero valued components. The HB authentication proceeds as follows:

Algorithm 2 The Hopper-Blum protocol [12]

1) S sets $t := 1$
2) Repeat m times:
   a) S generates a random challenge $\mathbf{c}_t \in_U \{0, 1\}^n$.
   b) With prob. $1-\eta$, U responds with $r_t := \mathbf{c}_t \cdot \mathbf{s}$, otherwise U responds with $r_t := 1 - \mathbf{c}_t \cdot \mathbf{s}$.
   c) if $r_t = \mathbf{c}_t \cdot \mathbf{s}$, S increments t.
3) if $t > (1 - \eta)$m, S accepts U.

During a single challenge-response round t, the user has to compute the binary inner product $\mathbf{c}_t \cdot \mathbf{s}$. To accomplish this, the user recalls the positions of k non-zero valued components of $\mathbf{s}$ and counts the number of ones appearing in the binary challenge $\mathbf{c}_t$ at those k positions; i.e., he/she calculates a Hamming weight $\mathbf{s} \oplus \mathbf{c}_t$. The user then responds with the parity bit r_t of $\mathbf{s} \oplus \mathbf{c}_t$, under the restriction that once in a while (i.e., with the fixed probability η) the user intentionally responds with an incorrect parity bit. The main intuition that lead us to investigate cognitive-asymmetry side channels in the HB protocol is the following: a user will take more time to respond to a challenge when he needs to add more nonzero bits together to compute the response.

Let us denote the Hamming weight of $\mathbf{s} \oplus \mathbf{c}_t$ as the counting load ℓ ($0 \leq \ell \leq k$). The counting load ℓ thus represents the number of non-zero valued components appearing at same indexes in both the secret vector $\mathbf{s}$ and the challenge $\mathbf{c}_t$. This design choice in the HB protocol inevitably leads to variations in user response times at each challenge-response round t. More specifically, the user response latency will be highly correlated with the counting load ℓ, thus potentially leaking information about the secret vector $\mathbf{s}$ through the *timing channel*.

In order to understand the extent of this vulnerability, we designed a probabilistic algorithm through which the attacker can gradually learn the user's secret $\mathbf{s}$, after observing a polynomial number of rounds. Before giving details of our attacking algorithm we first present a generative probabilistic model for a human running the HB protocol.

A. Modeling a Human Running the HB Protocol

In this section we describe a general and realistic model of a user running the HB protocol on which we build our attack. Cognitive-asymmetry side channel in our model is based on the fact that users enter their responses immediately after calculating them and the fact that they can not execute mental operations faster then their abilities allow them. This, for different challenges $\mathbf{c}_t$, results in different user's response times RT that can be observed by a passive attacker.

We first present a probabilistic model for the user's response time RT_ℓ associated with the fixed counting load ℓ. For this purpose, we extend the model first described in the paper by Yan et al. [11]:

$$RT_\ell = \overbrace{(0.3964 + 0.0383 \cdot \phi \cdot \gamma \cdot k) \cdot k + \alpha_0}^{\delta\text{: a fixed delay}} + D_\ell. \quad (8)$$

In this model, the expression $(0.3964 + 0.0383 \cdot \phi \cdot \gamma \cdot k)$ is a formula for the reaction time of cued recall obtained from the experimental results [28], [29], where ϕ is the ratio of cued recall compared to a single item recognition ($\phi = 1.969$ in [28]), while γ is the additional penalty if subjects are required to simultaneously recall the position of an item ($\gamma = 1.317$ in [29]), k is the size of the user's password, while $\alpha_0 = 0.738$ is the average reaction time for modulo 2 reduction operations reported in experiments [30]. Please note that in Eq. (8) δ represents a fixed delay, while $D_\ell > 0$ denotes a random delay associated with the counting load ℓ.

Let $f_{D_\ell}(t)$ be the probability density function (*pdf*) of D_ℓ, i.e., $D_\ell \sim f_{D_\ell}(t)$; $f_{D_\ell}(t)$ can take any form appropriate for modeling human reaction times (e.g., ex-Gaussian [31], ex-Wald, Weibull, etc.). We parametrize $f_{D_\ell}(t)$ as follows:

$$f_{D_\ell}(t) = f(t|d(\ell), \mathbf{p}), \text{ with support } t > 0 \quad (9)$$

where $d(\ell)$ is the mean of $f_{D_\ell}(t)$, and $\mathbf{p}$ represents all other relevant density parameters (i.e. variance, shape). We assume

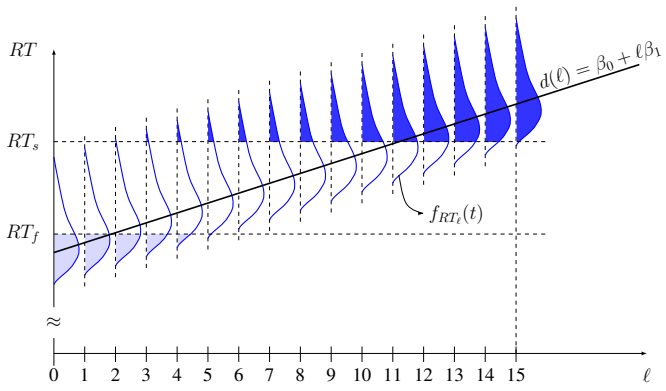


Fig. 5. An example of a human model where for all $f_{RT_\ell}(t)$ we have the same density parameters $\mathbf{p}$ and where the mean is defined by $d(\ell) = \beta_0 + \ell\beta_1$ (with $\beta_i > 0$). The expected value $E[RT] = \delta + \beta_0 + \beta_1 \frac{k}{2}$ as by [11].

$d(\ell)$ to be an arbitrary increasing positive function of ℓ that is strictly increasing for at least one counting load ℓ. This constraint on $d(\ell)$ reflects a reasonable assumption that a higher value of the counting load ℓ will likely result in a higher latency in the observable user's response time RT. Using $f_{D_\ell}(t)$, we can easily derive the distribution of RT_ℓ:

$$f_{RT_\ell}(t) = f(t|\delta + d(\ell), \mathbf{p}), \text{ with support } t > 0 \qquad (10)$$

Recall, in the HB protocol, the user's counting load ℓ depends on a random n-bit challenge $\mathbf{c}_t$. Therefore, the observable user's response time RT can be characterized by the following mixture distribution:

$$f_{RT}(t) = \sum_{\ell=0}^{k} \pi_\ell f_{RT_\ell}(t), \text{ where } \pi_\ell = \frac{\binom{k}{\ell}}{2^k} \text{ and } \sum_{\ell=0}^{k} \pi_\ell = 1. \quad (11)$$

The expected value of RT is $E[RT] = \sum_{\ell=0}^{k} \pi_\ell E[RT_\ell]$. Fig. 5 illustrates an example of one possible model of a human running the HB protocol; here we assume $d(\ell) := \beta_0 + \ell\beta_1$ (with $\beta_i > 0$).

B. Probabilistic Attacking Algorithm

In this section we describe a weight-updating algorithm (Algorithm 3) used in our cognitive-asymmetry based timing attack. For a given secret vector $\mathbf{s} \in \{0,1\}^n$ and a challenge vector $\mathbf{c}_t \in_U \{0,1\}^n$ let us define a set $\{\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_T\}$ of vectors $\mathbf{w}_t \in \{0,1\}^n$, $t \in \{1, 2, \ldots, T\}$, t being a protocol round. A vector $\mathbf{w}_t$ consists of n binary weights $w_{i,t}$ associated with the individual components of $\mathbf{s}$, in round t. Let us define two thresholds RT_f and RT_s, where $RT_f \leq RT_s$. Finally, for the observed user's response time RT in round t, let $F_t := \{RT < RT_f\}$ be a *fast event*, and $S_t := \{RT \geq RT_s\}$ a *slow event*.

Cognitive asymmetry in HB protocol is manifested as different latencies in user response times. Weight-updating algorithm (Algorithm 3) can be interpreted as follows. If, in a given challenge-response round t, the user enters his/her response r_t with a low latency (the fast event F_t takes place), the algorithm will set to 1 those weights $w_{i,t}$ of $\mathbf{s}$ that correspond to the positions of zero valued components in $\mathbf{c}_t$. Likewise, if a high latency is detected (the slow event S_t occurs), the algorithm will set to 1 those weights $w_{i,t}$ of $\mathbf{s}$ that correspond to the positions of non-zero valued components in $\mathbf{c}_t$. By repeating the above process over multiple rounds, we calculate the cumulative weight $W_i = \sum_{t=1}^{T} w_{i,t}$, for each component i of the secret vector $\mathbf{s}$. As we show in the sequel, the weights $w_{i,t}$ associated with the non-zero valued components of the secret vector $\mathbf{s}$ take value 1 more often on average compared to the weights of the zero valued components, resulting in the higher cumulative weights W_i. This allows us to recognize the positions of the k non-zero valued components of the secret vector $\mathbf{s}$, after observing a polynomial number of protocol rounds - see Section IV-C for a detailed analysis.

Algorithm 3 Weight-updating algorithm

1) Initialize: $\{\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_T\} = \{\mathbf{0}, \ldots, \mathbf{0}\}$
2) Repeat: $\forall t$: $w_{i,t} = \begin{cases} c_{i,t}, & \text{if } S_t \\ (1 - c_{i,t}), & \text{if } F_t \\ 0, & \text{otherwise} \end{cases}, \forall i$
3) Output: $W_i = \sum_{t=1}^{T} w_{i,t}, \ \forall i$

C. Analysis of the Weight-updating Algorithm

Before proceeding with the analysis of the Algorithm 3, we make some remarks concerning notation. From now on, all variables associated with non-zero valued components of the secret vector $\mathbf{s}$ will be marked with the sign "$\sim$". Accordingly, $\widetilde{w}_{i,t}$ represents the binary weight associated with the component s_i of $\mathbf{s}$, where $s_i = 1$, while $w_{j,t}$ represents the binary weight of the jth component of $\mathbf{s}$, where $s_j = 0$. Similarly, $\widetilde{W}_i$ and W_j are associated with the non-zero and zero valued components of the secret vector $\mathbf{s}$, respectively.

Theorem 1. *Weights $\widetilde{w}_{i,t}$ and $w_{j,t}$ are Bernoulli random variables with success probabilities $\widetilde{p} := P(\widetilde{w}_{i,t} = 1)$ and $p := P(w_{j,t} = 1)$, respectively. Under some mild assumptions (given in the proof below), the following holds: $\widetilde{p} > p$.*

Proof: The first part is trivial: by Algorithm 3, the weights $\widetilde{w}_{i,t}$ and $w_{j,t}$ are 0-1 random variables. Next we prove that $\widetilde{p} > p$, under two mild and realistic assumptions. In this direction, we use $F_\ell(RT)$ to denote the probability that the user response time satisfies $RT \leq RT_f$, given the counting load ℓ (light-shaded areas in Fig. 5); i.e., $F_\ell(RT) := \int_0^{RT_f} f_{D_\ell}(t)\,\mathrm{d}t$. In a similar manner, we define $S_\ell(RT) := \int_{RT_s}^{+\infty} f_{D_\ell}(t)\,\mathrm{d}t$ (dark-shaded areas in Fig. 5). By Algorithm 3 we have:

$$\begin{aligned} p &= P(w_{j,t}{=}1) = P(c_{j,t}{=}0, F_t) + P(c_{j,t}{=}1, S_t) \\ &= 2^{-1} \cdot \Big(P(F_t|c_{j,t}{=}0) + P(S_t|c_{j,t}{=}1)\Big), \end{aligned} \qquad (12)$$

$$\widetilde{p} = 2^{-1} \cdot \Big(P(F_t|\widetilde{c}_{i,t}{=}0) + P(S_t|\widetilde{c}_{i,t}{=}1)\Big). \qquad (13)$$

From the definitions of F_t and S_t, and the distribution of RT

(11), we can easily obtain:

$$p = 2^{-k-1} \cdot \left(\sum_{\ell=0}^{k} \binom{k}{\ell} \cdot F_\ell(RT) + \sum_{\ell=0}^{k} \binom{k}{\ell} \cdot S_\ell(RT) \right), \quad (14)$$

$$\widetilde{p} = 2^{-k-1} \cdot \left(\sum_{\ell=0}^{k-1} \binom{k-1}{\ell} \cdot F_\ell(RT) + \sum_{\ell=0}^{k-1} \binom{k-1}{\ell} \cdot S_{\ell+1}(RT) \right). \quad (15)$$

To prove that $\widetilde{p} > p$, using (14) and (15), it suffice to show that:

$$\sum_{\ell=0}^{k-1} \Big(F_\ell(RT) - F_{\ell+1}(RT) \Big) \cdot \tbinom{k-1}{\ell} > 0, \quad (16)$$

$$\sum_{\ell=0}^{k-1} \Big(S_{\ell+1}(RT) - S_\ell(RT) \Big) \cdot \tbinom{k-1}{\ell} > 0. \quad (17)$$

It is a matter of a straightforward calculation to show that both conditions (16) and (17) hold under the following *mild and realistic assumptions* about the human behavior (Fig. 5):

(i) $F_{\ell+1}(RT) \leq F_\ell(RT)$ and for at least one ℓ we have $F_{\ell+1}(RT) < F_\ell(RT)$,
(ii) $S_{\ell+1}(RT) \geq S_\ell(RT)$ and for at least one ℓ we have $S_{\ell+1}(RT) > S_\ell(RT)$.

These assumptions hold if (i) $d(\ell)$ (Eq. (9)) is an increasing positive function that is strictly increasing for at least one ℓ, and (ii) $f_{D_\ell}(t)$ (Eq. (9)) takes any form appropriate for modeling human reaction times (e.g., ex-Gaussian, etc.). ■

Corollary 1. *The weights* $\widetilde{W}_i = \sum_{t=1}^{T} \widetilde{w}_{i,t}$ *and* $W_j = \sum_{t=1}^{T} w_{j,t}$, *output by Algorithm 3, have binomial distributions, i.e.* $\widetilde{W}_i \sim B(T, \widetilde{p})$ *and* $W_j \sim B(T, p)$.

Indeed, because challenges $\mathbf{c}_t$ are randomly generated in each round t, $\widetilde{w}_{i,t}$, $t \in \{1, 2, \ldots, T\}$, are mutually independent Bernoulli random variables. Similar observation holds for $w_{j,t}$. Combining this result and the fact that $\widetilde{p} > p$ (by Theorem 1) we have the following result on the *correctness* of Algorithm 3:

Corollary 2. *For a sufficiently large* T, $E[\widetilde{W}_i] > E[W_j]$.

Corollary 2 shows that after observing a sufficient number of challenge-response rounds T, we will be able to distinguish zero from non-zero valued components of the secret vector $\mathbf{s}$.

Let us examine some other properties of the random variables $\widetilde{w}_{i,t}$ and $w_{j,t}$. In a single challenge-response round t, $\widetilde{w}_{i,t}$ and $w_{j,t}$ are not mutually independent (in general). However, those variables are defined on the same probability space, so it makes sense to analyze them jointly as bivariate Bernoulli random variables.

Lemma 1. *The correlation coefficients* $\rho_{\widetilde{w}_{i,t}, w_{j,t}}$ *and* $\rho_{w_{j,t}, w_{k,t}}$ *of the bivariate Bernoulli random variables* $(\widetilde{w}_{i,t}, w_{j,t})$ *and* $(w_{j,t}, w_{k,t})$, *respectively, are given by:*

$$\rho_{\widetilde{w}_{i,t}, w_{j,t}} = \frac{\widetilde{p}(1/2 - p)}{\sqrt{\widetilde{p}(1-\widetilde{p})p(1-p)}}, \quad \rho_{w_{j,t}, w_{k,t}} = \frac{1/2 - p}{1 - p} \quad (18)$$

We provide the proof in the Appendix. In general, we have $\rho_{\widetilde{w}_{i,t}, w_{j,t}} \neq 0$ and $\rho_{w_{j,t}, w_{k,t}} \neq 0$, except for $p = 1/2$.

Theorem 2. *The condition* $RT_f = RT_s$ *implies* $p = 1/2$ *and the independence of Bernoulli random variables* $\widetilde{w}_{i,t}$ *and* $w_{j,t}$, *as well as the independence of the Bernoulli random variables* $w_{j,t}$ *and* $w_{k,t}$. *Moreover, to maximize the difference* $(\widetilde{p} - p)$ *it is sufficient to consider the thresholds satisfying* $RT_f = RT_s$.

We provide the proof of Theorem 2 in the Appendix. As shown in [32], a sequence of independent bivariate Bernoulli random variables $(\widetilde{w}_{i,1}, w_{j,1})$, $(\widetilde{w}_{i,2}, w_{j,2})$,...,$(\widetilde{w}_{i,T}, w_{j,T})$ gives rise to the bivariate binomial random variable $(\widetilde{W}_i, W_j)$ with the correlation coefficient $\rho_{\widetilde{w}_{i,t}, w_{j,t}}$. Likewise, the bivariate binomial random variable (W_j, W_k) has the correlation coefficient $\rho_{w_{j,t}, w_{k,t}}$.

Theorem 3. [32] *When* $T \to \infty$, *the distribution of the bivariate binomial random variable* $(\widetilde{W}_i, W_j)$ *converges to the bivariate normal distribution with the correlation coefficient* $\rho_{\widetilde{W}_i, W_j} = \rho_{\widetilde{w}_{i,t}, w_{j,t}}$; *i.e.,* $\widetilde{W}_i$ *and* W_j *are jointly normal. The same holds for the bivariate binomial random variable* (W_j, W_k), *with* $\rho_{W_j, W_k} = \rho_{w_{j,t}, w_{k,t}}$.

Corollary 3. *For* $RT_f = RT_s$ *(and assuming* $T \to \infty$*), the jointly normal random variables* $\widetilde{W}_i$ *and* W_j, *as well as the jointly normal random variables* W_j *and* W_k, *are independent.*

Proof: Setting $RT_f = RT_s$ implies $p = 1/2$ by Theorem 2. Therefore, $\rho_{\widetilde{W}_i, W_j} = 0$ and $\rho_{W_j, W_k} = 0$ by Theorem 3 and Lemma 1. We conclude the proof by observing that two jointly normal random variables that are uncorrelated, are also independent [32]. ■

1) Estimating the Required Number of Rounds T: So far, we know that by setting $RT_f = RT_s$ and when T is large, the random variables $\widetilde{W}_i$ and W_j, as well as W_j and W_k, are normally distributed and mutually independent (by Theorem 3 and Corollary 3), and we know that their expected difference is maximized (by Theorem 2). By using these results that arise from the cognitive asymmetry in the HB protocol, we can now estimate the number of rounds T required to discover all non-zero valued components of the secret vector $\mathbf{s}$ with high probability. In other words, we want to determine the number of rounds T so the following holds:

$$P\left(\min_{1 \leq i \leq k} \widetilde{W}_i > \max_{1 \leq j \leq n-k} W_j \right) \approx 1. \quad (19)$$

Using the normal approximation for $\widetilde{W}_i$ and W_j, we have:

$$\widetilde{W}_i \sim \mathcal{N}(\mu_{\widetilde{W}_i}, \sigma^2_{\widetilde{W}_i}), \quad \mu_{\widetilde{W}_i} = T\widetilde{p}, \quad \sigma^2_{\widetilde{W}_i} = T\widetilde{p}(1-\widetilde{p}) \quad (20)$$

$$W_j \sim \mathcal{N}(\mu_{W_j}, \sigma^2_{W_j}), \quad \mu_{W_j} = Tp, \quad \sigma^2_{W_j} = Tp(1-p) \quad (21)$$

Let us define W_{crit} such that $P(W_j < W_{crit}) \approx 1$. From this and the fact that the random variables $\widetilde{W}_i$ and W_j are independent (for $RT_f = RT_s$) the following holds:

$$\begin{aligned} &P\left(\min_{1 \leq i \leq k} \widetilde{W}_i > \max_{1 \leq j \leq n-k} W_j \right) \approx \\ &\approx P\left(\widetilde{W}_1 > W_{crit}, ..., \widetilde{W}_k > W_{crit} \right) \geq 1 - kP\left(\widetilde{W}_i \leq W_{crit} \right). \end{aligned} \quad (22)$$

By combining (19) and (22) we obtain:

$$P\left(\widetilde{W}_i \leq W_{crit} \right) = \varepsilon / k \ll 1, \quad (23)$$

where ε is a negligible probability. By standardizing $\widetilde{W}_i = \widetilde{z}\sigma_{\widetilde{W}_i} + \mu_{\widetilde{W}_i}$ and $W_{crit} = z_{crit}\sigma_{W_j} + \mu_{W_j}$ we get:

$$P\left(\widetilde{W}_i \leq W_{crit}\right) = P\Big(\widetilde{z} \leq \frac{z_{crit}\sigma_{W_j} + \mu_{W_j} - \mu_{\widetilde{W}_i}}{\sigma_{\widetilde{W}_i}}\Big) = \varepsilon/k. \quad (24)$$

The *standard score* z_{crit} is obtained from the condition $P(W_j < W_{crit}) = P(z < z_{crit}) \approx 1$. Clearly, z_{crit} must be sufficiently larger than 0. Knowing that $RT_f = RT_s$ implies $p = 0.5$ (Theorem 2), and using (20) and (21), Eq. (24) can be re-written as:

$$P\left(\widetilde{W}_i \leq W_{crit}\right) = P\Big(\widetilde{z} \leq \underbrace{\frac{0.5z_{crit} - \sqrt{T}\,(\widetilde{p} - 0.5)}{\sqrt{\widetilde{p}(1-\widetilde{p})}}}_{\widetilde{z}_{\varepsilon/k}}\Big) = \varepsilon/k. \quad (25)$$

Observe from Eq. (25) how the standard score $\widetilde{z}_{\varepsilon/k}$ (and implicitly the number of challenge-response rounds T) relates to the targeted probability for $P\big(\widetilde{W}_i \leq W_{crit}\big)$, i.e., ε/k. We can see that in order to have $\varepsilon/k \ll 1$ (as by (23)), $\widetilde{z}_{\varepsilon/k}$ must be sufficiently smaller than 0. Moreover, using Eq. (25), we can obtain the following explicit expression for the number of challenge-response rounds T required to learn all the non-zero valued components of the secret vector $\mathbf{s}$, with high probability[2]:

$$T = \left(\frac{0.5z_{crit} - \widetilde{z}_{\varepsilon/k}\sqrt{\widetilde{p}(1-\widetilde{p})}}{\widetilde{p} - 0.5}\right)^2. \quad (26)$$

For practical values of n and k, z_{crit} can be bounded above by a small postive constant and $\widetilde{z}_{\varepsilon/k}$ can be bounded below by a small negative constant. Combined with Eq. (26), this leads to the following approximation for T:

$$T = \mathcal{O}\left(\frac{1}{(\widetilde{p} - 0.5)^2}\right). \quad (27)$$

In what follows, we evaluate and verify our theoretical results through numerical analysis.

Example: In the following example, our goal is to estimate the number of rounds T required to discover all the k non-zero valued components of $\mathbf{s} \in \{0,1\}^n$, where $n = 200$ and $k = 15$ [12]. We use the probabilistic generative model introduced in Section IV-A to model a user running the HB protocol, i.e., his/her response times. We make the following assumptions about the user model: the *pdf* of the user response time RT_ℓ is given by (10) and is parametrized by a linear positive function $d(\ell) = \beta_0 + \ell\beta_1$ (Fig. 5), where β_0 and β_1 account for counting calculation ($\beta_0 = \beta_1 = 0.738$ s [11]). The density parameters in $\mathbf{p}$ (e.g., the shape of the distribution, variance) are estimated from the study of the Mod10 method with 58 real users (please check Section III for more details[3]).

As a first step in our analysis, we need to estimate the probability $\widetilde{p}$ (recall, $p = 1/2$). For this purpose, we used the above probabilistic model and generated 1000 traces, each trace comprising 500 challenge-response rounds. Fig. 6 shows one such trace. As a threshold for $RT_f = RT_s$ (Theorem 2) we used the average of the generated response times of 34.5 seconds (which is close to the expected user reaction time of 33.4 seconds from the model in [11]). Although the average of the response times might not always be the optimal strategy for the threshold $RT_f = RT_s$, it will still provide good results (as we show later on in this section). Due to lack of space, we do not provide the proof for this statement. From each trace we randomly extracted the normally distributed variable $\widetilde{W}_i$ (Eq. (20)). Since variables $\widetilde{W}_i$ were independently drawn from each trace, we were able to estimate the mean $\mu_{\widetilde{W}_i}$, and finally $\widetilde{p} = 0.5929$.

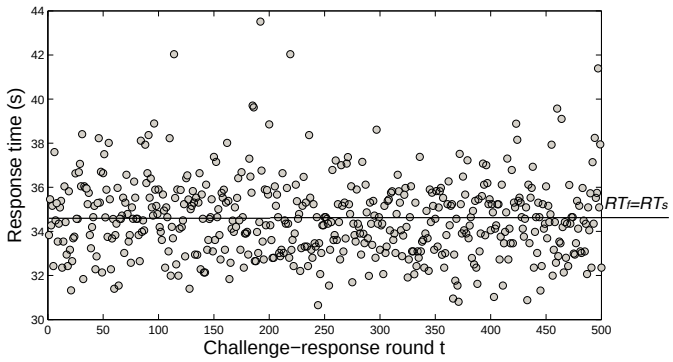


Fig. 6. A trace of response times RT generated by the proposed generative model of the human behavior. The average of the collected response times was selected as a threshold $RT_f = RT_s$.

By plugging in the value of $\widetilde{p} = 0.5929$ into Eq. (26) we can estimate the number of challenge-response rounds T required to discover all non-zero valued components of $\mathbf{s}$. By setting the value of z_{crit} relatively high (≈ 3.1) and $\widetilde{z}_{\varepsilon/k}$ relatively low (≈ -3.1) the algorithm will discover all non-zero valued components with probability $P(\min \widetilde{W}_i > \max W_j) \approx 1$ after observing approximately $T \approx 1100$ rounds. Furthermore, 1100 rounds represents approximately 158 logins (each login consists of 7 rounds), meaning that the HB protocol can be broken in a polynomial time. In Table II we also estimate the number of rounds required to discover all non-zero valued components of $\mathbf{s}$ as we change the slope β_1 of the linear positive function $d(\ell)$ (where $d(\ell) = \beta_o + \ell \cdot \beta_1$). Please note, even for a small slope $\beta_1 = 0.05$ we still satisfy the condition $\widetilde{p} > p$ (Theorem 1). However, since $\widetilde{p} = 0.518$ is very close to $p = 0.5$ (Theorem 2), the attacker will now have to observe a larger, but still a polynomial number of challenge-response rounds required to discover all non-zero valued components of $\mathbf{s}$. Indeed, for a smaller slope β_1 we will have a larger overlap between probability density functions f_{D_ℓ} ($f_{D_\ell} \sim D_\ell$) that will lead to a decreased discriminatory potential. Our choice to use $\beta_0 = \beta_1 = 0.738$ in our model to estimate the number of rounds T required to discover all non-zero valued components of secret $\mathbf{s}$ was inspired by experimental data in [30] that states that user on average achieves 0.738 seconds to solve zero-and-one problems, such as counting (specific for HB method).

Fig. 7 shows the probability $P(\min \widetilde{W}_i > \max W_j)$ given

[2]Please note that this implies z_{crit} and $\widetilde{z}_{\varepsilon/k}$ to be sufficiently larger and smaller than 0, respectively.

[3]Only operations involving 0 or 1 as an operand or answer from Mod10 study are used in this example [11].

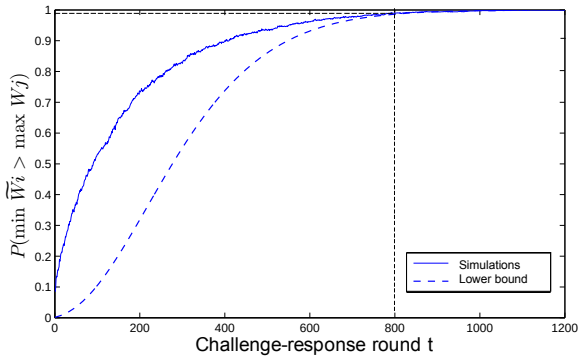


Fig. 7. The probability where attacker learns all non-zero valued components of the secret vector **s**.

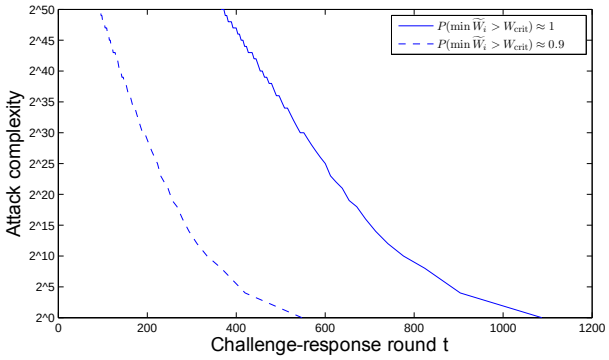


Fig. 8. Attack complexity of the brute-force attack in which the attacker discovers all non-zero valued components of **s** with high probability.

TABLE II. THE NUMBER OF CHALLENGE-RESPONSE ROUNDS THE ATTACKER HAS TO OBSERVE TO RECOVER SECRET **S** AS A FUNCTION OF THE SLOPE β_1 BEING THE PART OF $d(\ell)$.

	β_1	**0.738**	0.5	0.3	0.2	0.1	0.05
Basic attack	$\widetilde{p} - p$	**0.093**	0.088	0.073	0.06	0.032	0.018
	Login sessions	**157**	176	254	375	1331	4283
Optimized attack	Login sessions $P(\min \widetilde{W}_i > W_c) \approx 1$	**92**	102	148	219	781	2517
	Login sessions $P(\min \widetilde{W}_i > W_c) \approx 0.9$	**13**	16	22	32	113	361

by Eq. (22) at which the algorithm distinguishes all non-zero components of **s** in relation to the number of collected challenge-response rounds based on the cognitive asymmetry in the HB protocol. To confirm the validity of our model we also plot this probability ($P(\min \widetilde{W}_i > \max W_j)$) obtained as a success rate from $N = 100$ traces (simulations) with 1200 challenge-response rounds. We can see an inaccuracy of simulations with Eq. (22) for small values of observed challenge-response rounds. This inaccuracy follows from a sum-bound (Eq. (22)) that actually presents a lower-bound. Please note in Fig. 7 that lower bound becomes accurate for large challenge-response rounds T ($T > 800$) where ε is negligible (Eq. (23)).

2) Speeding up the Attack (reducing the number of rounds T): As we can see from Eq. (26), for practical values of n and k, we still require a somewhat large number of observed rounds T necessary to learn all non-zero valued components of the secret vector **s** with high probability. Please note that the computational complexity of the original weight updating algorithm is essentially zero. As we show, by slightly modifying the weight updating algorithm we can significantly reduce the observed number of challenge-response rounds T at the cost of the increased computational complexity.

After a sufficient number of rounds T, our (modified) algorithm outputs $M = k + K$ components that holds k actual non-zero valued components (with high probability) along with K extra zero-valued components. To distinguish all k non-zero components (from $M = k + K$ candidates), every candidate for the secret vector **s** is tested on the collected T challenge-response pairs. Please note, the computational complexity of such attack is $\binom{M}{k}$. The candidate whose responses match the collected ones with highest success rate presents the targeted secret vector **s**. Since users intentionally respond with an incorrect parity bit with probability η (the HB protocol description in Algorithm 2) the best candidate has matches with rate around $1 - \eta$.

Essentially, our goal is to find a sufficient number of rounds T that holds all k non-zero valued components of secret vector **s** within M non-zero valued component candidates. From Eq. (26) it is obvious that the number of required rounds T can be reduced by either influencing the values of z_{crit} or $\widetilde{z}_{\varepsilon/k}$. Recall from Eq. (24), to estimate the number of rounds T required to discover all non-zero valued components of secret vector **s** with high probability, the variable $\widetilde{z}_{\varepsilon/k}$ must be sufficiently small. Therefore, we can decrease the number of rounds T by decreasing z_{crit}. Since z_{crit} is obtained by standardizing W_{crit}, smaller z_{crit} implies smaller W_{crit}. Furthermore, $(n - k)$ variables W_j are mutually independent (Corollary 3) and identically distributed from the same normal distribution (Eq. (21)). According to the law of large numbers, for large $(n - k)$ and a fixed K, $\frac{K}{n-k}$ approximately corresponds to the fraction of the total number of independent variables W_j larger or equal than some chosen value W_{crit}:

$$P\left(W_j \geq W_{crit}\right) \approx \frac{K}{n-k} \tag{28}$$

Therefore, a larger number of extra components K reduces the value of W_{crit} and z_{crit}, leading to a smaller number of rounds T (Eq. (26)). In the following examples we will estimate the computational complexity of the modified weight updating algorithm as well as the number of rounds T.

Example: Fig. 8 illustrates the relation between the computational (attack) complexity of modified weight updating algorithm and the number of observed challenge-response rounds T (for different probabilities $P(\min \widetilde{W}_i > W_{crit})$). For $P(\min \widetilde{W}_i > W_{crit}) \approx 1$, $K = 56$ ($z_{crit} = 0.5166$), $\widetilde{p} = 0.5929$ and $p = 0.5$ (Theorem 2) the computational complexity of the algorithm will will be $\binom{k+K}{k} \approx 2^{50}$ (solvable in polynomial time [33]). However, the algorithm

will extract all k non-zero valued components after observing $T \approx 380$ rounds.

Please note that the required number of challenge-response rounds can be further reduced by slightly increasing the value of $\widetilde{z}_{\varepsilon/k}$ (while still maintaining the large probability of success). For example, with probability $P(\min \widetilde{W}_i > W_{crit}) \approx 0.9$ and $K = 56$ (the complexity of the attack is 2^{50}), the algorithm will be able to determine the targeted secret vector $\mathbf{s}$ after $T = 95$ rounds. If a single login session contains $m = 7$ rounds, the attacker will have to observe 13 login sessions! In Table II we also estimate the number of login sessions required to discover all non-zero valued components of $\mathbf{s}$ as we change the slope β_1 of the linear positive function $d(\ell)$. As expected, a smaller $\widetilde{p}$ will require a larger number of login sessions for the attacker.

V. Usability Implications of Strengthening Mod10 and HB schemes

We have shown that both Mod10 and HB, as cognitive authentication schemes, are vulnerable to cognitive-asymmetry side channels, or more precisely to passive (side-channel) timing attacks. In the following, we argue that the only way to strengthen these schemes is to perturb the usability (e.g. increase the overall login time and/or increase PIN/password size).

A. Entering a Response After a Predefined Delay

One way to strengthen the security of Mod10 and HB schemes against the passive side-channel timing attack is to prevent users to enter the response immediately after calculating it. In this way, the attacker will not be able to extract sufficient information about the user's secret from the recorded response times.

For example, for Mod10 scheme, if we assume a Gaussian probability distribution, the majority of response times are not greater than 3.5 seconds. By preventing the users to enter the response in less than 3.5 seconds, we eliminate the side-channel attack based on the variations in the users' cognitive load (response times). However, the overall authentication time will increase from the initial 10.4 to approximately 15 seconds (4×3.5 + waiting cost [sec]). Similarly, for HB scheme we can see in Fig. 6 that the majority of response times are not greater than 40 seconds. By preventing the users to enter the response in less than 40 seconds, this means that the overall authentication time increases from the initial 241.5 to approximately 280 seconds (40×7 [sec]).

Altogether, it is possible to strengthen the security of both cognitive authentication schemes by preventing users to enter the response faster than the predefined delay. Nevertheless, we have shown that this solution necessarily increases overall authentication time; thus a higher security comes at some form of the decreased usability.

B. Increasing the PIN Size and Related Usability Cost

In this section we investigate how a slightly longer secret k affects the security and usability of Mod10 and HB schemes. In the case of Mod10 protocol users would have to remember a larger PIN (also denoted with k), while in HB protocol users would have to remember a larger number k of non-zero valued components of secret vector $\mathbf{s}$.

By increasing the number of non-zero valued components k, the HB protocol still remains insecure against the timing attack. This can be clearly seen from Eq. (27), where the time complexity of the attack is independent of the practical values of k. A larger k can slightly increase the number of login rounds the attacker has to observe, but cannot prevent the attacker from learning all k components (expressed through the standardized variable $z_{\varepsilon/k}$ in Eq. (26)).

For Mod10 scheme, please note that the minimum PIN size, denoted with k, has to satisfy the following expression:

$$k = \left\lceil \frac{4 \cdot \log_2 10}{H(D|\mathbf{r}, t_r)} \right\rceil, \tag{29}$$

where $H(D|\mathbf{r}, t_r)$ is the conditional entropy of the random PIN digit D given the vectors of observed response digits $\mathbf{r}$ and response times t_r. The reduction in entropy due to the time-based side-channel has to be compensated by extending the 4-digit PIN size. Recall, by evaluating Eq. (7) numerically, we were able to reduce the entropy by approximately 2 bits per PIN digit. So, according to Eq. (29), the new PIN size should be at least $k = 7$ digits long.

In order to estimate the associated usability cost for the Mod10 scheme, we use the computational model proposed in [11]. The expected value of response time for a single PIN digit is given by the following equation:

$$E[RT] = (0.3694 + 0.0383 \cdot \varphi \cdot k) + \alpha_1 + \alpha_0 \tag{30}$$

where $(0.3694 + 0.0383 \cdot \varphi \cdot k)$ is the reaction time by the user to recall k PIN digits, $\alpha_0 = 0.738$ and $\alpha_1 = 0.773$ are the average reaction times for modulo 10 reduction operations and small additions, respectively, while $\varphi = 1.969$ is the ratio of cued recall compared to single item recognition [11]. From Eq. (30) we obtain $E[RT] = 2.182$ seconds (what is consistent with the average user reaction time of 2.098 seconds for a single digit from our test). Using this model, we can see that by increasing the PIN size k from the initial 4 to 7 digits, the expected value of the overall authentication time increases from initial 8.72 (4×2.182) to 16.85 (7×2.408 [sec]) seconds.

We have shown that longer PIN strengthens the security of Mod10 scheme against timing attacks. However, a higher security comes at non-negligible usability cost; i.e., an increase in both the PIN size and the overall authentication time.

C. Discussion

In the previous section we have shown that the only way to strengthen the security of HB and Mod10 schemes against the timing attack is to increase the overall authentication time. In the case of the HB scheme we have shown that the security can only be strengthened by preventing the users to enter the response before the predefined delay expires, what inevitably increases the overall authentication time (the usability cost). The increase of the overall authentication time is also inevitable when strengthening the Mod10 scheme, either

by increasing the PIN size or by introducing a predefined delay. Since usability score of an authentication scheme is correlated with short authentication times, the designers of new cognitive authentication schemes should not neglect the timing side channel attack.

VI. Related Work

There is a body of research focused on designing secure PIN-entry schemes in face of the threat posed by observation attacks [34], [35], [36]. Some research design their solutions secure against a short-term memory attackers, using the fact that the human short-term memory has a limited capacity. In these solutions, the user is requested to give answers to a set of challenges during a login procedure. However, the authentication scheme is designed in a way that the user can easily respond to the questions, while the cognitive capacity exceeds the attacker (human) memory.

Bianchi et al. proposed a nonvisual unimodal schemes, which uses hidden audio and vibration challenges for user authentication [37]. In another work by Bianch et al. Spinlock, Colorlock and Timelock schemes that achieve faster times than Spinlock [38], [39]. However, all three schemes have partial leakage of information in the observation attack.

Other solutions assume the existence of stronger attacker that can record the complete login session and try to recover the user's secret PIN/password [14], [12], [7], [13], [40], [41]. However, all these scheme are not usable in practice since they all take large authentication time.

Designing a scheme secure against even a simple passive attack in a model where the attacker can observe both challenges and responses appears to be challenging [7], [42], [11]. In Cognitive authentication scheme (CAS) [7], a user mentally computes a path formed by his portfolio images, and gives an answer based on that (mentally) computed path. CAS scheme is vulnerable to SAT solver attacks [42] and an attack based on probabilistic decision tree [11]. However, the speed of such attacks can be improved in combination with a timing attack. This comes from the fact that not all decision paths are of the same length, what leads to faster or slower user response times. By measuring this time, the attacker can eliminate some of the possible decision paths what can speed up the whole process of finding the user's password.

To speed up the login process while keeping the solution safe against observation attacks, some solutions rely on the presence of secondary-based (unobservable) channels. Kuber and Yu [43] and Sasamoto et. al. [5] use a tactile channel as a secure hidden challenge channel.

In VibraPass authentication system user receives hidden challenges via his mobile phone [44] (a vibration telling the user to enter true/false response). Hidden challenges are used to avoid possible manipulations by the attacker. The authors mentioned confused waiting as a potential timing attack.

In the Undercover solution the user simultaneously receives a visual challenge and a hidden tactile challenge via a protected channel and authenticates by answering correctly to several challenges. One of authors of Undercover, Hasegawa et. al. proposed two alternative designs to Undercover [4], one of which uses an audio channel as the carrier of the hidden challenges. However, the proposed solution is prone to intersection attacks [9]. Unfortunately, Undercover is also prone to intersection attacks as independently demonstrated in [9] and [11]. This problem can be easily mitigated if challenges are fixed instead of being randomized [9]. Unfortunately, Undercover is not secure in a very strong attacker model where attacker records user's response time [9]. The attack is based on design flaws and exploits human users' nonuniform behavior on how users respond to different challenges.

VII. Conclusions

In this paper we have shown how it is possible to exploit detectable variations in the user's cognitive load when authenticating using cognitive authentication schemes. We demonstrated this by exploiting vulnerability of two well-known cognitive authentication schemes (Hopper-Blum (HB) and Mod10 methods) to cognitive-asymmetry side-channel timing attacks. The novelty of these attacks is that they are not based on the asymmetry of neither physical nor virtual user interfaces of a certain method. With this, we have shown the existence of a strong tradeoff between security and usability in cognitive authentication methods, where security can only be achieved at a cost of a perturbed usability (e.g. longer authentication time and/or longer PINs/passwords).

An important conclusion from the results presented in this paper is that researchers designing and evaluating cognitive authentication schemes should exercise caution when claiming superiority of their proposals in terms of usability with respect to existing proposals. High usability scores are often correlated with short authentication times. However, potential vulnerabilities of those authentication methods to timing attacks could easily render the claimed "short" authentication times not so short in the end.

A general conclusion that results from the study of HB and Mod10 methods (as representatives of a whole range of cognitive authentication schemes) can be summarized in the following guideline: *When designing new cognitive authentication methods designers should pay particular attention to potential asymmetry in both the cognitive load (cognitive-asymmetry side-channels) and the physical interface of different elements of their methods.*

For future work we plan to investigate the influence of other factors (such as memory factor) on the effectiveness of the side-channel timing attacks on cognitive authentication schemes by conducting a long term study with users. Also, as HB protocol found its application in secure RFID systems [16], we plan to investigate the security of such RFID systems against the timing attack in detail.

References

[1] E. Grosse and M. Upadhyay, "Authentication at Scale," *IEEE Security and Privacy*, vol. 11, pp. 15–22, 2013.

[2] F. Tari, A. A. Ozok, and S. H. Holden, "A comparison of perceived and real shoulder-surfing risks between alphanumeric and graphical passwords," in *Proc. Symposium on Usable Privacy and Security*, ser. SOUPS. ACM, 2006, pp. 56–66.

[3] M. Backes, M. Dürmuth, and D. Unruh, "Compromising Reflections-or-How to Read LCD Monitors around the Corner," in *IEEE Symposium on Security and Privacy*, 2008, pp. 158–169.

[4] M. Hasegawa, N. Christin, and E. Hayashi, "New Directions in Multisensory Authentication," in *Proc. Int. Conf. Pervasive Computing (Pervasive)*, 2009.

[5] H. Sasamoto, N. Christin, and E. Hayashi, "Undercover: Authentication Usable in Front of Prying Eyes," in *Proc. Conf. Human Factors in Computing Systems*, ser. CHI '08, 2008.

[6] T. Perković, M. Čagalj, and N. Saxena, "Shoulder-Surfing Safe Login in a Partially Observable Attacker Model," in *Proc. Int. Conf. Financial Cryptography and Data Security*, ser. FC, 2010.

[7] D. Weinshall, "Cognitive Authentication Schemes Safe Against Spyware (Short Paper)," in *Proc. IEEE Symposium on Security and Privacy*, ser. SP, 2006, pp. 295–300.

[8] L. Zhuang, F. Zhou, and J. D. Tygar, "Keyboard Acoustic Emanations Revisited," in *CCS: Proc. ACM Conf. Computer and Communications Security*, 2005.

[9] T. Perkovic, S. Li, A. Mumtaz, S. A. Khayam, Y. Javed, and M. Cagalj, "Breaking Undercover: Exploiting Design Flaws and Nonuniform Human Behavior," in *SOUPS - Symposium On Usable Privacy and Security*, 2011, p. 15.

[10] G. T. Wilfong, "Method and Appartus for Secure PIN Entry," in *Lucent Technologies, Inc., Murray Hill, NJ, U. S. Patent*, 1999.

[11] Q. Yan, J. Han, Y. LI, and R. Deng, H., "On Limitations of Designing Usable Leakage-Resilient Password Systems: Attacks, Principles and Usability," in *Network & Distributed System Security Symposium (NDSS), Distinguisged Paper Award,*, 2012.

[12] N. Hopper and M. Blum, "Secure Human Identification Protocols," in *Proc. Int. Conf. on the Theory and Application of Cryptology and Information Security: Advances in Cryptology*, ser. ASIACRYPT, 2001.

[13] S. Li and H.-Y. Shum, "Secure Human-Computer Identification (Interface) Systems Against Peeping Attacks: SecHCI. Cryptology ePrint Archive, Report 2005/268," 2005.

[14] H. J. Asghar, J. Pieprzyk, and H. Wang, "A New Human Identification Protocol and Coppersmith's Baby-Step Giant-Step Algorithm." in *Applied Cryptography and Network Security*. Springer, 2010.

[15] M. Lei, Y. Xiao, S. V. Vrbsky, C.-C. Li, and L. Liu, "A virtual password scheme to protect passwords." in *Proceedings of IEEE International Conference on Communications, ICC*, 2008.

[16] S. Piramuthu, "HB and Related Lightweight Authentication Protocols for Secure RFID Tag/Reader Authentication," in *Proc. CollECTeR Europe Conference*, 2006.

[17] A. J. Aviv, B. Sapp, M. Blaze, and J. M. Smith, "Practicality of Accelerometer Side Channels on Smartphones," in *Proc. Annual Computer Security Applications Conf.*, ser. ACSAC, 2012.

[18] Q. Yan, J. Han, Y. Li, J. Zhou, and R. H. Deng, "Designing leakage-resilient password entry on touchscreen mobile devices," in *Proc. of the ACM Symposium on Information, Computer and Communications Security*, ser. ASIACCS, 2013.

[19] N. Zbrodoff, "Why is 9+ 7 Harder Than 2+ 3? Strength and Interference as Explanations of the Problem-size Effect," *Memory & Cognition*, vol. 23, no. November 1994, pp. 689–700, 1995.

[20] R. S. Siegler, "The Perils of Averaging Data Over Strategies: An Example From Children's Addition," *J. Experimental Psychology-general*, vol. 116, pp. 250–264, 1987.

[21] M. H. Ashcraft and J. Battaglia, "Cognitive Arithmetic: Evidence for Retrieval and Decision Processes in Mental Addition," *J. Experimental Psychology: Human Learning & Memory*, vol. 4, pp. 527–538, 1978.

[22] D. C. Geary, "The Problem-size Effect in Mental Addition: Developmental and Cross-national Trends," *Mathematical Cognition*, vol. 2, pp. 63–94, 1996.

[23] J. Kittler, M. Hatef, R. P. W. Duin, and J. Matas, "On Combining Classifiers," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 3, pp. 226–239, 1998.

[24] D. C. Geary, K. F. Widaman, and T. D. Little, "Cognitive Addition and Multiplication: Evidence for a Single Memory Network." *Memory & Cognition*, vol. 14, no. 6, pp. 478–87, Nov. 1986.

[25] G. Tsoumakas and I. Katakis, "Multi Label Classification: An Overview," *Int. J. Data Warehousing and Mining*, vol. 3, no. 3, pp. 1–13, 2007.

[26] G. H. John and P. Langley, "Estimating Continuous Distributions in Bayesian Classifiers," in *Proc. Conf. Uncertainty in Artificial Intelligence*, ser. UAI, 1995, pp. 338–345.

[27] R. Jin and Z. Ghahramani, "Learning with Multiple Labels," *Advances in Neural Information Processing Systems*, pp. 897–904, 2002.

[28] P. A. Nobel and R. M. Shiffrin, "Retrieval Processes in Recognition and Cued Recall," *Journal of Experimental Psychology: Learning, Memory, and Cognition*, pp. 384–413, 2001.

[29] J. M. L. Corbin, "Effect of a Simple Experimental Control: The Recall Constraint in Sternberg's Memory Scanning Task," *European Journal of Cognitive Psychology*, vol. 5, no. 20, pp. 913–935, 2008.

[30] J. I. Campbell and Q. Xue, "Cognitive Arithmetic Across Cultures," *Journal of Experimental Psychology*, vol. 130, no. 2, pp. 299–315, 2001.

[31] R. Whelan, "Effective analysis of reaction time data," *The Psychological Record*, vol. 58, pp. 475–482, 2008.

[32] A. Marshall and I. Olkin, "A Family of Bivariate Distributions Generated by the Bivariate Bernoulli Distribution," *American Statistical Association*, vol. 80, no. 390, pp. 332–338, 1985.

[33] M. Curtin, *Brute Force: Cracking the Data Encryption Standard*. Springer-Verlag New York, Inc., 2005.

[34] V. Roth, K. Richter, and R. Freidinger, "A PIN-entry Method Resilient Against Shoulder Surfing," in *Proc. ACM Conf. Computer and Communications Security*, ser. CCS. ACM, 2004, pp. 236–245.

[35] M.-K. Lee, "Security Notions and Advanced Method for Human Shoulder-Surfing Resistant PIN-Entry," *IEEE Trans. Inf. Forensics and Security*, vol. 9, no. 4, 2014.

[36] A. D. Luca, K. Hertzschuch, and H. Hussmann, "ColorPIN: securing PIN entry through indirect input." in *CHI*. ACM, 2010.

[37] A. Bianchi, I. Oakley, and D. S. Kwon, "The Secure Haptic Keypad: A Tactile Password System," in *Proc. SIGCHI Conf. Human Factors in Computing Systems*, ser. CHI. ACM, 2010, pp. 1089–1092.

[38] A. Bianchi, I. Oakley, and D.-S. Kwon, "Counting Clicks and Beeps: Exploring Numerosity Based Haptic and Audio PIN Entry." *Interacting with Computers*, vol. 24, no. 5, pp. 409–422, 2012.

[39] ——, "Spinlock: A Single-Cue Haptic and Audio PIN Input Technique for Authentication." in *Haptic and Audio Interaction Design*, vol. 6851. Springer, 2011, pp. 81–90.

[40] S. Wiedenbeck, J. Waters, L. Sobrado, and J.-C. Birget, "Design and Evaluation of a Shoulder-surfing Resistant Graphical Password Scheme," in *Proc. Working Conf. Advanced Visual Interfaces*, ser. AVI '06. ACM, 2006, pp. 177–184.

[41] H. Zhao and X. Li, "S3PAS: A Scalable Shoulder-Surfing Resistant Textual-Graphical Password Authentication Scheme," in *Proc. Int. Conf. Advanced Information Networking and Applications Workshops - Volume 02*, ser. AINAW, 2007.

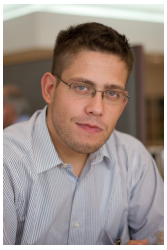
[42] P. Golle and D. Wagner, "Cryptanalysis of a Cognitive Authentication Scheme (Extended Abstract)," in *Proc. IEEE Symposium on Security and Privacy*, ser. S&P, 2007, pp. 66–70.

[43] R. Kuber and W. Yu, "Authentication Using Tactile Feedback," in *Interactive Experiences, HCI, London, UK*, 2006.

[44] A. De Luca, E. von Zezschwitz, and H. Hussmann, "VibraPass: Secure Authentication Based on Shared Lies," in *Proc. SIGCHI Conf. Human Factors in Computing Systems*, ser. CHI. ACM, 2009, pp. 913–916.

Mario Čagalj is an Associate Professor in the Department of Electronics at Faculty of Electrical Engineering, Mechanical Engineering, and Naval Architecture (FESB), University of Split, Croatia. He received the Dipl.Ing degree in computer science and electrical engineering from the University of Split, Croatia, in 1998, and the PhD degree in communication systems from the Ecole Polytechnique Federale de Lausanne (EPFL) in 2006. In 2000 and 2001, he completed the Predoctoral School in Communication Systems, EPFL. From 2001 to 2006, he was a research assistant in the Laboratory for Computer Communications and Applications (LCA) at EPFL. In December 2006, Mario Čagalj was elected Assistant Professor and in September 2010 he was promoted to Associate Professor at the University of Split, Croatia. His research interests include the design and analysis of security protocols for wireless networks, applied cryptography, applications of game theory to wireless (and wired) networks, and the design of energy-efficient communication protocols for wireless networks.

Toni Perković is currently employed as a young researcher (senior assistant) in the Department of Electronics at Faculty of Electrical Engineering, Mechanical Engineering, and Naval Architecture (FESB), University of Split, Croatia. He received the Dipl. Ing. degree in telecommunications and electrical engineering from the University of Split, Croatia, in 2007, and the PhD degree in Computer Science from the University of Split, Croatia, in 2013. His research interests include the location privacy, the usability, design and analysis of security protocols for wireless networks, the usability and design of the secure authentication protocols. He is a member of the IEEE.



Marin Bugarić holds the position of a young researcher (senior assistant) at Department of Electronics at Faculty of Electrical Engineering, Mechanical Engineering, and Naval Architecture (FESB), University of Split, Croatia. He received both Dipl. Ing. degree (ISCED 5A) and Ph.D. (ISCED 6) in Computer Science from the University of Split, Croatia, in 2007. and 2013, respectively. His research interests include early forest fire detection systems based on visual smoke detection, GIS applications in firefighting activities, augmented reality systems, forest fire risk indexes, as well as the usability and design of the secure authentication protocols and analysis of security protocols for wireless networks.

APPENDIX

A. Proof of Lemma 1

The correlation coefficient between $\widetilde{w}_{i,t}$ and $w_{j,t}$ is:

$$\rho_{\widetilde{w}_{i,t},w_{j,t}} := \frac{\mathrm{cov}\left(\widetilde{w}_{i,t}w_{j,t}\right)}{\sigma_{\widetilde{w}_{i,t}}\sigma_{w_{j,t}}}. \tag{31}$$

Because $\widetilde{w}_{i,t}$ and $w_{j,t}$ are Bernoulli r.v. with the success probability $\widetilde{p}$ and p, respectively, we have: $\sigma_{\widetilde{w}_{i,t}} = \sqrt{\widetilde{p}(1-\widetilde{p})}$ and $\sigma_{w_{j,t}} = \sqrt{p(1-p)}$. We next determine the numerator of $\rho_{\widetilde{w}_{i,t},w_{j,t}}$. Using $E\left[\widetilde{w}_{i,t}\right] = \widetilde{p}$ and $E\left[w_{j,t}\right] = p$, we have:

$$\begin{aligned}\mathrm{cov}\left(\widetilde{w}_{i,t}w_{j,t}\right) &= E\left[\widetilde{w}_{i,t}w_{j,t}\right] - \widetilde{p}p \\ &= P\left(\widetilde{w}_{i,t}{=}1, w_{j,t}{=}1\right) - \widetilde{p}p \qquad (32)\\ &= \big(P\left(w_{j,t}{=}1|\widetilde{w}_{i,t}{=}1\right) - p\big)\widetilde{p}. \qquad (33)\end{aligned}$$

Now, observe that the r.v. $w_{j,t}$ will take the value of 1, given $\widetilde{w}_{i,t} = 1$, only if the challenge bits $c_{j,t}$ and $\widetilde{c}_{i,t}$, in the round t challenge $\mathbf{c}_t$, are equal (by Algorithm 3). This and the fact that $\mathbf{c}_t$ is generated uniformly at random in the HB protocol, imply: $P\left(w_{j,t}{=}1|\widetilde{w}_{i,t}{=}1\right) = 1/2$. By combining this with (33) and plugging everything back into (31) we conclude the proof. We can similarly derive the expression for $\rho_{w_{j,t},w_{k,t}}$. ■

B. Proof of Theorem 2

Setting $RT_f = RT_s$ implies that $F_\ell(RT) = 1 - S_\ell(RT)$, and hence the expression (14) for p simplifies as follows:

$$\begin{aligned}p &= 2^{-k-1}\cdot\left(\sum_{\ell=0}^{k}\binom{k}{\ell}\cdot\Big(1 - S_\ell(RT)\Big) + \sum_{\ell=0}^{k}\binom{k}{\ell}\cdot S_\ell(RT)\right)\\ &= 2^{-k-1}\cdot\sum_{\ell=0}^{k}\binom{k}{\ell} = 2^{-k-1}\cdot 2^k = \frac{1}{2}.\end{aligned}$$

Next we show that $\widetilde{w}_{i,t}$ and $w_{j,t}$ are independent for $p = 1/2$. To prove the independence for $p = 1/2$, we will show that the following holds: $P\left(\widetilde{w}_{i,t}, w_{j,t}\right) = P\left(\widetilde{w}_{i,t}\right)\cdot P\left(w_{j,t}\right)$. Observe:

$$\begin{aligned}p = 1/2 \implies \rho_{\widetilde{w}_{i,t},w_{j,t}} = 0 \qquad &\text{(by Lemma 1)}\\ \rho_{\widetilde{w}_{i,t},w_{j,t}} = 0 \implies P\left(\widetilde{w}_{i,t}{=}1, w_{j,t}{=}1\right) = \widetilde{p}p \qquad &\text{(by (32))}\end{aligned}$$

This, with $\widetilde{p} := P\left(\widetilde{w}_{i,t}{=}1\right)$ and $p := P\left(w_{j,t}{=}1\right)$, gives:

$$P\left(\widetilde{w}_{i,t}{=}1, w_{j,t}{=}1\right) = P\left(\widetilde{w}_{i,t}{=}1\right)\cdot P\left(w_{j,t}{=}1\right). \tag{34}$$

Using (34), we can show that a similar equality holds for the three remaining combinations of $\widetilde{w}_{i,t}$ and $w_{j,t}$. Here we provide the proof just for the case $(\widetilde{w}_{i,t}, w_{j,t}) = (0,1)$.

$$\begin{aligned}P\left(\widetilde{w}_{i,t}{=}0, w_{j,t}{=}1\right) &= P\left(\widetilde{w}_{i,t}{=}0|w_{j,t}=1\right)\cdot P\left(w_{j,t}{=}1\right)\\ &= \big(1 - P\left(\widetilde{w}_{i,t}{=}1|w_{j,t}{=}1\right)\big)\cdot P\left(w_{j,t}{=}1\right)\\ &= P\left(w_{j,t}{=}1\right) - P\left(\widetilde{w}_{i,t}{=}1, w_{j,t}{=}1\right)\\ &\overset{(34)}{=} P\left(w_{j,t}{=}1\right)\cdot\big(1 - P\left(\widetilde{w}_{i,t}{=}1\right)\big)\\ &= P\left(w_{j,t}{=}1\right)\cdot P\left(\widetilde{w}_{i,t}{=}0\right).\end{aligned}$$

In the same fashion we can prove the independence of the random variables $w_{j,t}$ and $w_{k,t}$, for $p = 1/2$.

Finally, we prove that to maximize the difference $(\widetilde{p} - p)$ it is sufficient to consider only those response time thresholds that satisfy $RT_f = RT_s$. In this direction, we consider the following optimization problem:

$$\begin{aligned}&\text{maximize} \quad (\widetilde{p} - p)\\ &\text{subject to} \quad RT_f \le RT_s\ .\end{aligned} \tag{35}$$

Let Δp^* be the optimal value of the problem (35), i.e., $\Delta p^* = \sup\left\{(\widetilde{p} - p)\,|RT_f \le RT_s\right\}$. Using the expressions (14) and (15) for $\widetilde{p}$ and p, respectively, we can write the following:

$$\Delta p^* = \sup_{RT_f \le RT_s}\Big\{\sum\nolimits_S + \sum\nolimits_F\Big\}$$

where

$$\sum\nolimits_S := \sum_{\ell=0}^{k-1}\Big(S_{\ell+1}(RT) - S_\ell(RT)\Big)\cdot\binom{k-1}{\ell}$$

$$\sum\nolimits_F := \sum_{\ell=0}^{k-1}\Big(F_\ell(RT) - F_{\ell+1}(RT)\Big)\cdot\binom{k-1}{\ell}$$

In what follows, we use the sign * to mark the optimal values that different variables take at the optimal solution to the problem (35). Let us assume that the optimal value $\Delta p^* = \sum_S^* + \sum_F^*$ is such that the following holds $\sum_F^* < \sum_S^*$. Now, let us set RT_f so that $RT_f = RT_s^*$. Then, $F_\ell(RT) + S_\ell^*(RT) = 1$, that is, $F_\ell(RT) = 1 - S_\ell^*(RT)$, $\forall\ell$. Plugging this equality into the expression for $\sum_F$, we obtain:

$$\begin{aligned}\sum\nolimits_F &= \sum_{\ell=0}^{k-1}\Big(F_\ell(RT) - F_{\ell+1}(RT)\Big)\cdot\binom{k-1}{\ell}\\ &= \sum_{\ell=0}^{k-1}\Big(1 - S_\ell^*(RT) - 1 + S_{\ell+1}^*(RT)\Big)\cdot\binom{k-1}{\ell} = \sum\nolimits_S^*.\end{aligned}$$

So, there exists RT_f such that $RT_f \le RT_s^*$ and $\sum_F^* < \sum_S^* = \sum_F$, contradicting the optimality of Δp^*. Therefore, at the optimal solution, we must have $\sum_F^* \ge \sum_S^*$.

Let us now assume that the optimal value Δp^* is characterized by $\sum_F^* > \sum_S^*$. Setting RT_s so that $RT_s = RT_f^*$, implies $S_\ell(RT) = 1 - F_\ell^*(RT)$, $\forall\ell$. Plugging this equality into the expression for $\sum_S$, we obtain:

$$\begin{aligned}\sum\nolimits_S &= \sum_{\ell=0}^{k-1}\Big(S_\ell(RT) - S_{\ell+1}(RT)\Big)\cdot\binom{k-1}{\ell}\\ &= \sum_{\ell=0}^{k-1}\Big(1 - F_\ell^*(RT) - 1 + F_{\ell+1}^*(RT)\Big)\cdot\binom{k-1}{\ell} = \sum\nolimits_F^*.\end{aligned}$$

Similarly as before, there exists RT_s such that $RT_f^* \le RT_s$ and $\sum_S^* < \sum_F^* = \sum_S$, contradicting the optimality of Δp^*. Therefore, at the optimal solution, we must have $\sum_F^* \le \sum_S^*$. We conclude that at the optimal solution we must have $\sum_F^* = \sum_S^*$, i.e., $\Delta p^* = 2\sum_F^* = 2\sum_S^*$. More importantly, we showed constructively that Δp^* is attainable under the following condition $RT_f = RT_s$.