Accelerometer Based Activity Recognition Human Identification through Image Evaluation access control to ubiquitous hospital information and services

Adapa venkata subbarao[#1], Knvssk Rajesh[#2]

[1,2#] (Kakinada Institute of Engineering and Technology, JNTU Kakinada, A.P, India)

Abstract:

Activity recognition is becoming an important research area, and finding its way to many application domains ranging from daily life services to industrial zones. Sensing hardware and learning algorithm are two important components in activity recognition. For sensing devices, we prefer to use accelerometers due to low cost and low power requirement. Our implementation not only outperforms the original method in terms of computation complexity at least 10 times faster in our experiments based systems. The protocols proposed in literature are based on some underlying difficult mathematical problem, which are tuned so as to make them easily computable by humans. As a result these protocols are easily broken when desired to be efficiently executable. We present a Human Identification Protocol based on the ability of humans to efficiently process an image given a secret predicate. It is a challenge-response protocol in which a subset of images presented satisfies a secret predicate shared by the challenger and the user.

Corresponding Author: Knvssk Rajesh (Associate Professor, M.tech CSE Department)

1 Introduction

The problem of constructing human identification protocols is an important one in the cryptographic community. That is to say, how can a human *H* authenticate to a remote server *C*, without using any computational aid? To make matters worse, the communication link between *H* and *C* is controlled by an adversary who can either passively listen to their conversation or actively interfere at will. Under such conditions, it is desirable that this adversary should not be able to masquerade as *H* even after observing a number of authentication sessions. Nowadays, activity recognition is an increasingly important research area [1]. Modern life style tends to involve in more sedentary jobs, while there are growing evidences in the relationship between common health problems such as diabetes, cardiovascular, osteoporosis and the level of physical activity. Therefore, simple monitoring will not protect anybody from any disease but may help to assess and then alter the life style, which in turn could result in health benefits [2]. In addition, activity recognition has been considered to be a potential factor in improving convenience as well as productivity at work place; for example, in smart hospitals in aircraft maintenance or in a workshop [3] Also, such activity recognition systems can be used to predict abnormal behaviors such as falling down for emergency response in health-care systems. There are various approaches using video and deployed sensors, many researchers, however, have used

accelerometers in their research work due to low cost, low power requirement, portability, and versatile characteristics. Therefore, we also use accelerometer-based input for our activity recognition system. With respect to recognition methods, sliding window approach is commonly used in accelerometer based activity recognition [4], However, in most cases, a sliding window cannot cover one complete activity, since the duration of different activities usually varies significantly and the start time of an activity in a continuous stream is unknown in advance. Thus, sliding window approach may produce fragments of activities making it difficult to obtain comprehensive models to satisfy the performance requirements of a continuous activity recognition system. One feasible solution for this problem is to take into account the duration of activities so that some short-length fragments can be eliminated. In addition to the problem associated with sliding windows, the interdependency among activities is another issue, which should be considered when detecting activities in a continuous data stream. Nevertheless, to the best of our knowledge [5], none of the existing activity recognition models is able to handle all the afore mentioned problems, especially, for large-scale active2.

2 An Informal Description of the Protocol

The central idea proposed in this paper is informally as follows: How can we combine the notion of CAPTCHA (creating a challenge-response that is not susceptible to bots) and secure user authentication that is not vulnerable to shoulder surfing or sniffing? To accomplish this feat.

we propose the following protocol Setup. User and the server agree upon a secret that is a composite of the following:

1. A simple question *Q* (which we will call a predicate) with only a binary answer, such as “Does the picture contain a woman?”

2. A set of distinct random numbers *a*1, *a*2, . . . , *ar*; all between 1 and *n*.

Server to user. A list of *n* pictures that are uniformly distributed with respect to the question *Q* above. User to server. *n*-bit binary string such that for pictures numbered *a*1, a2,. . .,ar the corresponding bits are answers to the secret question *Q* and for all the other positions the bits are random.70 H. Jameel et al. The server accepts if the answer string is correct at the designated places. We can do the online step repeatedly to amplify security. In the full version of the protocol in , we permute the *ai*’s to make it harder for the adversary to extract any information from the answer string. The probability of guessing the correct permutation would be far less than that of guessing a correct random ordering of numbers. Security is based on the fact that (i) a bot or a computer program does not know the relationship between the pictures, and
(ii) a human watching the proceedings would not know which bits are significant, which in turn will make it hard to guess the question being answered.

3 Definitions

We start with a set of definitions formalizing the notions of Identification Protocols and the new concepts introduced in this paper. We first define the notion of an AI problem solver which is modified from the definition of an AI problem in [6].

Definition 1:

An AI problem solver is a function f : S → R, where S is a set of AI problem instances and R ∈ {0, 1} is the answer alphabet. A family of AI problem solvers is a map F : Keys(F)×S → R. Here Keys(F) denotes the set of all AI problem solvers from S to R. Namely, if k ∈ Keys(F) then Fk : S → R is an AI problem solver.

Notice that we specifically define a family of AI problem solvers instead of just a single one. Such a family will allow us to distribute different secrets, namely $k \in$ Keys(F), to different users for authentication. The concept is similar to a function family.

Definition 2:

An AI problem solver f is said to be (δ, τ)-solved if there exists a program A, running in time at most τ, on an input s
R←S, such that
Pr

_
s
$R \leftarrow S : A(s) = f(s)$

_
$\geq \delta$
F is said to be (δ, τ)-hard if no current program is a (δ, τ)solution to f, and the AI community agrees that it is hard to find such a solution.

Definition 3:

A family of AI problem solvers is said to be (δ, τ)-hard if for all keys k ∈ Keys(F), Fk is (δk, τk)-hard with δk ≤ δ and τk ≤ τ.

Definition 4:

We say that a function family, F : Keys(F)×S → R is (λ (r) , τ)- resilient against key recovery, if for all H running in time at most τ, we have:
Pr[*k*
R←Keys(F); b1b2 . . . br
R←{0, 1}r ;
s1s2 . . . sr ← S| Fk (s1) Fk (s2) . . . Fk (sr) = b1b2 . . . br;
k_ ← H (s1s2 . . . sr): k = k_] < λ(r)

Human Identification Through Image Evaluation Using Secret Predicates 71Notice that *H* is not shown the value of the function *Fk* at each of the '*r*' inputs. *H* only knows that the answer to each input belongs to the range *R*. It has to guess the correlation between the different inputs. Of course, the inputs must have an internal structure in order for the above definition to make sense.

We do not elaborate this correlation between the inputs here as it will become clear when we describe the security of our protocol against human adversaries, instantiated with a suitable choice of the function family We restate the definitions of identification protocols and human executable protocols

Definition 5:

An Identification Protocol is a pair of probabilistic interactive programs (H,C) with shared auxiliary input z, such that the following conditions hold:

– For all auxiliary inputs z, Pr [_H (z), C (z)_ = accept] > 0.9
– For each pair x = y, Pr [_H (x) , C (y)_ = accept] < 0.1
When _H,C_ = accept, we say that H authenticates to C. The transcript of C contains challenges c and that of H comprises responses r to these challenges.

Definition 6:

An Identification Protocol (H,C) is (α, β, t)-human executable if at least a (1 − α) portion of the human population can perform the calculations H unaided and without errors in at most t seconds with probability greater than (1 − β).

Algorithm 1: Forward algorithm for calculating Z_X

```
Forward
    for t = 1 To T do
        for y = 1 To StateNum do
            α[y][t] = 0
            γ[y][t] = 0
            λ[y][t] = 0
            for d = 1 To D do
                if t − d + 1 > 0 then
                    γ[y][t] += λ[y][t − d] e^{G(y,t−d+1,t)}
                    γ[y][t] += e^{G_IA(IA,1,t−d)+G(y,t−d+1,t)}
                else
                    Break
            if t > 1 then
                α[y][t] = α[y][t − 1] e^{G_IA(IA,t,t)} + γ[y][t]
            else
                α[y][t] = γ[y][t]
            for y' = 1 To StateNum do
                λ[y][t] = λ[y][t] + α[y'][t] e^{w^{Tr}(y',y)}
    Z_X = e^{G_IA(IA,1,T)}
    for y = 1 To StateNum do
        Z_X = Z_X + α(y, T)
end
```

4. A User Friendly Implementation

For human users the parameters $L = 10$, $l = 5$ and $m = 4$ can be chosen. Note that a random guess attack can only succeed with a probability $2-20$ in this case, which is more than the security of a 4-digit pin number. The user is given a hidden permutation string say: 0098030502. When the user inputs his ID, he is brought to a page containing 10 pictures in a 2X5 grid at the bottom of which is a text box. The user answers by randomly picking '0' or '1' in place of the '0's' in the permutation string and answering the pictures in the specified order corresponding to the digits other than the '0's'. So, for example, a possible answer would be 1011001101, where the underlined digits denote the actual answers and the rest are random bits. The user would input the string 1011001101 and will go to the next series of 10 pictures if this answer is correct at the specified positions. The procedure continues until 4 steps and the user is accepted once all the 4 steps result in a success [6]. The choice of using 10 pictures in a challenge seems appropriate, as they can easily be displayed on the A single picture would take around 5 seconds for a human verify whether it satisfies the predicate or not. This would take around 100 seconds to execute the protocol. This amount of time seems reasonable if we use the protocol only under certain circumstances such as when the user is trying to log on through an insecure public terminal..

5. Limitations and Discussion

One might ask the question that how long the protocol should be run to keep a desired security level. It is evident that based on our assumption, the protocol can be safely executed for a number of times if we only consider adversarial programs. How about human adversaries? We know an inherent weakness in us humans; the more the work load, the less efficient we are. So if a human adversary is given a collection of, say 2000, pictures and is asked to find the hidden predicate, then he might not be able to examine all these pictures. In order to make the task harder for a human adversary, we can have two or more predicates connected through a truth clause. The user then checks whether the picture satisfies the clause and answers accordingly. This will result in an increase in execution time, but guessing the secret predicates will be much harder. Another question is regarding the possible usage of our protocol. We insist that our protocol should be used only in situations when a user is away from the luxury and security of his personal computer or office environment. The user might want to use our system when using a public terminal to log in for emails while on a business trip. However, when he is back in his office or home, he can use the normal password based system to log in to his computer. So, we can safely use our system for a small number of authentications before the secret predicate can be refreshed and a new hidden permutation can be used. The fact that the secret predicate plus the hidden permutation is not a load on a human's memory the hidden permutation being the size of a normal telephone number makes this switch very practical and reusable. Consequently, we can use this system, for say 20 or 30 authentications before renewing the secret. A modified version of the protocol, secure against general active adversaries is also desirable; without making it infeasible or increasing the number of rounds. The most important limitation is the selection of the predicates and selecting appropriate pictures that satisfy these predicates. This can be done by a dedicated group from the service providers. We do not know however, if this task can be performed by a computer or not [7]. Automatically generated predicates and pictures might prove helpful and will increase the practicality of our scheme.

6. Conclusion

We have presented a novel implementation of semi-Markov Conditional Random Fields and fast algorithms for gradient calculation. The solution not only is able to make use of the interdependency and the duration of activities to increase the accuracy, but also takes a practical amount of time for parameter estimation. Although the algorithm produced a low accuracy in some particular case, overall, we have shown that our approach obtains better results When compared to others. The problem of making secure human identification protocols in which a human authenticates to a remote server has resulted in many efficient authentication protocols over the years. These protocols try to make things easy for humans by presenting them a challenge based on some mathematical problem which is easy to compute but difficult for an adversary to crack. However, the efficiency of these protocols lies in the user friendly representation.

References:

1. Ling Bao and Stephen S. Intille. Activity recognition from user annotated acceleration data. In *Proceedings of the 2nd International Conference on Pervasive Computing*, volume 3001, pages 1–17, 2004.

2. Ulf Blanke and Bernt Schiele. Daily routine recognition through activity spotting. In *Proceedings of the 4th International Symposium on Location and Context-Awareness*, volume 5561, pages 192–206, 2009.

3. Stephen Boyd and Lieven Vandenberg he. *Convex Optimization*. Cambridge University Press, 2004.

4. Oliver Brdiczka, Jrme Maisonnasse, Patrick Reignier, and James L. Crowley. Detecting small group activities from multimodal observations. *Applied Intelligence*, 30:47–57, 2009.
5. Matsumoto, T., Imai, H.: Human Identification through Insecure Channel. Advances in Cryptology - EUROCRYPT 91, Lecture Notes in Computer Science, Springer-Verlag. **547** (1991) 409–421

6. Wang, C.H., Hwang, T., Tsai, J.J.: On the Matsumoto and Imai's Human Identification Scheme. Advances in Cryptology - EUROCRYPT 95, Lecture Notes in Computer Science, Springer-Verlag. **921** (1995) 382–392

7. Matsumoto, T.: Human-computer cryptography: An attempt. 3rd ACM Conference on Computer and Communications Security, ACM Press. (1996) 68–75