YPAP: The Yoking-proofs Based Authentication Protocol for Wearable Devices in Wireless Communications

Hong Liu, Rui Yu, Yueliang Wan
Run Technologies Co., Ltd. Beijing
Beijing 100192, China
Email: liuhongler@ieee.org; yurui@bjrun.com; yueliang@bjrun.com

***Abstract*—Along with the development of wireless communications, wearable devices are becoming popular for monitoring user data to provide intelligent service support. It makes that the wearable devices confront severe security issues compared with the traditional short-range communications. Due to limitations of computation capabilities and communication resources, it brings more challenges to design security schemes for the wearable devices. In this work, a yoking-proofs based authentication protocol (YPAP) is proposed for the wearable devices during secure wireless communications. In the YPAP, lightweight cryptographic operators are applied to realize authentication between a smart phone and two wearable devices, and yoking-proofs are established for the remote cloud server to perform simultaneous identification during a session. Meanwhile, Rubin logic based security formal analysis is performed to prove that the YPAP has theoretical design correctness. It indicates that the proposed protocol is flexible for ubiquitous wearable device applications.**

***Keywords*-Authentication protocol, wearable device, yoking-proof, security, Rubin logic.**

I. Introduction

Along with the development of wireless communications, wearable devices are becoming popular for an individual to provide intelligent service support. The wearable devices are mainly based on short-range wireless communication technologies (e.g., Bluetooth, WiFi, and near field communication (NFC)) to realize data perception. Currently, the wearable devices are still in the infancy, and confront several open issues due to the limitations of computation capabilities and communication resources [1], [2]. Considering the wearable devices being attached with a user's sensitive data (e.g., body signs, tracking, and preferences), it brings increasing security and privacy challenges via the open communication channels. It is noteworthy for designing security mechanism to address the security and privacy issues during the wireless communications.

Researches have been worked to strengthen security properties for the wearable devices in body area networks (BAN) and intelligent medical care applications [3]–[5]. Thereinto, user privacy and data trustworthiness are established for the mobile wearable devices, and secure communication could be achieved via wireless intra-body communication to support multiple wearable devices. Recently, the wearable devices are arranged into cloud environments, in which two or multiple wearable devices communicate among themselves along with establishing interactions with the remote cloud server. It is necessary to propose authentication schemes for the wearable devices to achieve security protection [6].

In this work, the authors identify a unique security issue, and present a lightweight authentication protocol to realize both secure and simultaneous identification for the wearable devices. Thereinto, the yoking-proof is applied for designing the authentication protocol. The concept of yoking-proof is first proposed in the radio frequency identification (RFID) applications [7]. Thereafter, several yoking-proofs or grouping-proofs based protocols are designed to realize that two or multiple tags are simultaneously scanned within a reader's interrogation range during a session [8], [9]. In these schemes, simultaneous existences of two or more tags are regarded as a pair or a group to be verified by a reader (or a database). In fact, such interactive mode is similar to scenarios of wearable device applications, in which two or more wearable devices establish authentication by a smart phone (or a cloud server). Here, we focus on both secure authentication and simultaneous identification for the wearable devices, and a yoking-proofs based authentication protocol (YPAP) is designed for the wireless communications, and the main contributions are as follows:

- Establishing yoking-proofs by involving two associated wearable devices into one session, which realizes that a cloud server simultaneously verifies the validity of the two wearable devices.
- Adopting lightweight cryptographic operators for authentication, in which wearable devices need not perform pseudo-random number generation operations, and only the bitwise logical operator and hash related functions are applied to ensure the data confidentiality and integrity.
- Applying random partition and dynamic update mechanisms into the authentication. The pre-shared secret is divided into two dynamic partial fields for self-refreshing. The timestamp based pseudo-random flags are applied for quick check with efficiency consideration.

978-1-4673-7211-4/15 $31.00 © 2015 IEEE

DOI 10.1109/UIC-ATC-ScalCom-CBDCom-IoP.2015.298

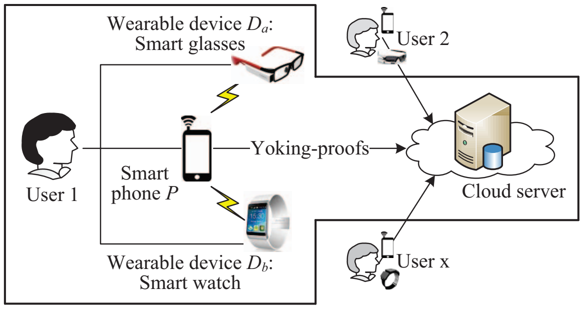

Figure 1. The system model.

Table I
NOTATIONS

Notation	Description
PID_{D_x}	The pseudo-random identifier of D_x.
F_{D_x}, F_P	The pseudo-random flag of D_x and P, which act as an identify label with timestamp.
r_0, r_1, r_2	The pseudo-random numbers generated by P.
k_P	The authentication key shared by P.
k_{D_x}	The secret keys owned by D_x.
S	The l-bit length secret shared for random partition.
$H_{k_*}(x)$	The keyed hash message authentication code (HMAC) function.
$f(x)$	The defined function involving the parameter x.

The remainder of this paper is organized as follows. Section II presents related works. Section III introduces the detailed protocol descriptions, and the Rubin logic based security formal analysis is performed in Section IV. Finally, Section V draws a conclusion.

II. Related Works

In former studies, there is less work on authentication of the wearable devices. The typical works are as follows.

Kim et al. [5] established an intra-body communication channel for the wearable devices. Intra-body communication transfers data via the human body, and supports multiple wearable devices to achieve secure data transmission. A prototype being wearable on the wrist, is assigned with an integrated processor to control the intra-body communication module. Instead of using radio transmission, the module uses the human body as communication channel to maximize the security of transmitted signals.

Diez et al. [6] focused on the self-authenticable wearable devices to propose a point-to-point authentication protocol, which enables secure mutual authentication between a wearable device and other entity such as another wearable device, a personal device (mobile phone), a remote server, or a user's application. Meanwhile, the related technologies such as near field communication, smart cards, point-to-point protocol, extensible authentication protocol, and imprinting are introduced for the wearable devices. Different security levels (i.e., low, intermediate, and high) oriented scenarios are described according to sensitivity of information handled by the wearable devices.

Towards the yoking-proofs based authentication protocols, the main works focus on the RFID applications.

Chien et al. [8] proposed a tree-based yoking-proof protocol, which designs a binary tree to arrange tags to reduce the computational cost from $O(N)$ to $O(1)$. In the scheme, the tags are assigned to the leaves of the tree structure, and the protocol addresses the updated paths to identify the tags. It brings another open issue for the yoking-proofs protocols since the verifier is off-line and the synchronization simultaneously involves multiple tags and the server.

Liu et al. [9] proposed a grouping-proofs-based authentication protocol (GUPA) to address the security issue for multiple readers and tags simultaneous identification in distributed RFID systems. In the GUPA, distributed authentication mode with independent subgrouping proofs is adopted to enhance hierarchical protection; an asymmetric denial scheme is applied to grant fault-tolerance capabilities against an illegal reader or tag; and a sequence-based odd-even alternation group subscript is presented to define a function for secret updating. It indicates that the GUPA is efficient for resource-constrained distributed RFID systems.

III. YPAP: The Proposed Authentication Protocol

A. System Initialization

In the system model, a user owns a smart phone P, and two wearable devices (i.e., smart glasses D_a, and smart watch D_b), as shown in Figure 1. The user's smart phone can connect the remote cloud server for requiring advanced service support along with other users. Each wearable device owns its pseudo-random identifier PID_{D_*} and secret key k_{D_*}. All the entities have the corresponding pseudo-random flags F_*, and a pre-shared secret S. The notations are introduced in Table I.

B. Protocol Descriptions

Figure 2 shows the proposed YPAP, in which a phone P and two wearable devices D_a and D_b establish interactions.

1) Challenge-Response Between P and D_a: The phone P generates a pseudo-random number r_0, and extracts its timestamp embedded pseudo-random flag F_P. P transmits the cascaded messages $r_0 \| F_P$ to the wearable device D_a as a query to initiate a new session. Upon receiving the challenge from P, D_a performs quick search to determine the correctness of F_P. If there is non-matching flag or the flag with wrong timestamp, P will be regarded as an illegal phone and the protocol will terminate. Otherwise, D_a extracts its pseudo-random identifier PID_{D_a} and its own

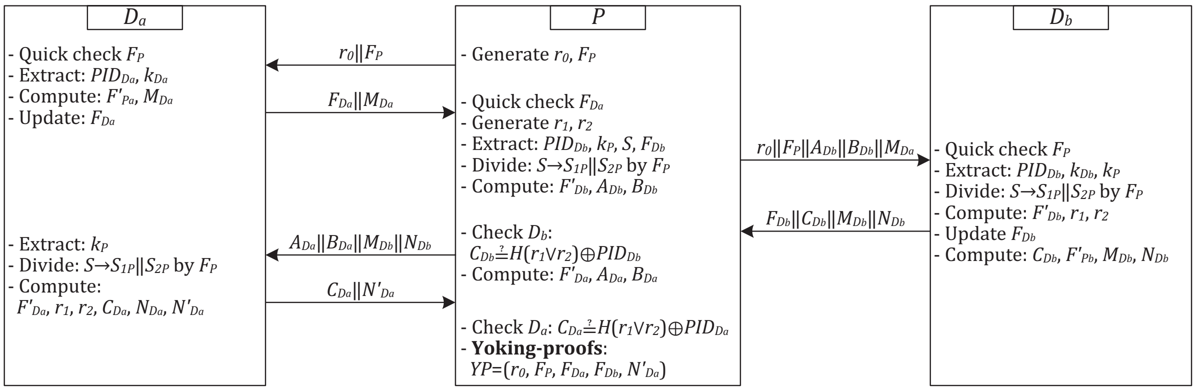

Figure 2. The proposed yoking-proofs based authentication protocol (YPAP)

secret key k_{D_a}. D_a computes F'_{P_a} and M_{D_a}.

$$F'_{P_a} = H_{k_{D_a}}(F_P \oplus r_0),$$
$$M_{D_a} = PID_{D_a} \oplus F'_{P_a}.$$

Afterward, D_a updates its pseudo-random flag F_{D_a}, and transmits the messages $F_{D_a}\|M_{D_a}$ to P as a response.

2) P Establishing Interactions Between D_a and D_b: Upon P receiving $F_{D_a}\|M_{D_a}$, P performs search to determine the correctness of F_{D_a}. If there is nonmatching flag or the flag with wrong timestamp, D_a will be regarded as an illegal entity. Otherwise, the protocol continues. P generates two pseudo-random numbers r_1 and r_2, and extracts a set of values $\{PID_{D_b}, k_P, S, F_{D_b}\}$. Thereinto, PID_{D_b} is the other wearable device D_b's pseudorandom identifier; k_P is an authentication key; S is a l-bit length secret; F_{D_b} is a locally stored pseudo-random flag which is transmitted by D_b in a former session. P first divides the pre-shared secret S into S_{1P} and S_{2P} by F_P. The partitioning method is as follows, 1) perform modulo operation on F_P by l to obtain $d = F_P \pmod{l}$; 2) mark the first d bit as a delimiter which divides S into two partial fields S_{1P} and S_{2P}. During the random partition, underflow should be considered, and zero is padded to the higher order bits. Thereafter, P computes the values F'_{D_b}, A_{D_b}, and B_{D_b}.

$$F'_{D_b} = H_{k_P}(F_{D_b} \oplus r_0),$$
$$A_{D_b} = PID_{D_b} \oplus (S_{1P} - r_1),$$
$$B_{D_b} = F'_{D_b} \oplus (S_{2P} + r_2).$$

P transmits $r_0\|F_P\|A_{D_b}\|B_{D_b}\|M_{D_a}$ to D_b as an access challenge. D_b performs the similar operations as D_a, including quick check on F_P and extracting $\{PID_{D_b}, k_{D_b}, k_P\}$. D_b divides the pre-shared secret S to obtain S_{1P} and S_{2P} according to the same random partition approach. Afterward, D_b re-computes F'_{D_b}, and derives r_1 and r_2.

$$F'_{D_b} = H_{k_P}(F_{D_b} \oplus r_0),$$
$$r_1 = S_{1P} - A_{D_b} \oplus PID_{D_b},$$
$$r_2 = B_{D_b} \oplus F'_{D_b} - S_{2P}.$$

3) P Performing Authentication on D_b: When P and D_b establish interactions, D_b updates its pseudo-random flag F_{D_b}, and computes C_{D_b}, F'_{P_b}, M_{D_b}, and N_{D_b} for further authentication.

$$C_{D_b} = H(r_1 \vee r_2) \oplus PID_{D_b},$$
$$F'_{P_b} = H_{k_{D_b}}(F_P \oplus r_0),$$
$$M_{D_b} = PID_{D_b} \oplus F'_{P_b},$$
$$N_{D_b} = f(M_{D_a}) = M_{D_a} \vee Rot(PID_{D_b} \vee r_0, k_P).$$

D_b transmits the messages $F_{D_b}\|C_{D_b}\|M_{D_b}\|N_{D_b}$ to P for identity authentication. Upon receiving the messages, P recomputes C_{D_b} by the locally generated random numbers r_1 and r_2, and compares the computed C_{D_b} with the received C_{D_b} to verify the validity of D_b. If the two values are not identical, D_b will be regarded as an illegal entity and the protocol will terminate. Otherwise, the protocol continues.

4) P Performing Authentication on D_a and the Yoking-Proofs Establishment: After the verification on D_b, P continues to compute F'_{D_a}, A_{D_a}, and B_{D_a}.

$$F'_{D_a} = H_{k_P}(F_{D_a} \oplus r_0),$$
$$A_{D_a} = PID_{D_a} \oplus (S_{1P} - r_1),$$
$$B_{D_a} = F'_{D_a} \oplus (S_{2P} + r_2).$$

P transmits the messages $A_{D_a}\|B_{D_a}\|N_{D_b}\|M_{D_b}$ to D_a for authentication. Upon receiving the messages, D_a D_a extracts an authentication key k_P, and performs random partition operation on S to obtain S_{1P} and S_{2P}. D_a also recomputes F'_{D_a}, derives $\{r_1, r_2\}$, and computes the following values.

$$F'_{D_a} = H_{k_P}(F_{D_a} \oplus r_0),$$
$$r_1 = S_{1P} - A_{D_a} \oplus PID_{D_a},$$
$$r_2 = B_{D_a} \oplus F'_{D_a} - S_{2P},$$
$$C_{D_a} = H(r_1 \vee r_2) \oplus PID_{D_a},$$
$$N_{D_a} = f(M_{D_b}) = M_{D_b} \vee Rot(PID_{D_a} \vee r_0, k_P),$$
$$N'_{D_a} = f(N_{D_a}, N_{D_b}) = N_{D_b} \oplus N_{D_a}.$$

Table II
SYMBOL NOTATIONS

Notation	Description
$POSS(E)$	Possession set that contains the security data known or possessed by E.
$BEL(E)$	Belief set that contains beliefs held by E.
$X contains Y$	Y is a submessage of X, i.e. X=$x_1 \cdot Y \cdot x_2$.
$X := f(X)$	Assignment symbol, X is replaced by the function value $f(X)$.
$X from E$	X is marked as being received from E.
$LINK(r)$	A challenge and a response are linked by a random number r. $LINK(r)$ is added to the belief set of an entity who generates r, and allows only one subsequently received message to contain r. After r is received by another entity, $LINK(r)$ will be removed.
Send(Q, X)	E sends X to Q.
Receive(Q, X)	E receives X from Q.
Generate-nonce(r)	E generates a nonce r to link a challenge and a response. $LINK(r)$ is removed from $BEL(E)$ upon E receiving the response.
Generate-secret(s)	E generates a secret s. $Observers(S)$ and $POSS(E)$ are updated.
Check-freshness(X)	E checks the freshness of X
Update(X)	The update function is used to maintain the observers of X.
Concat$(x_1, ..., x_n)$	X is constructed by submessages $x_1, ..., x_n$.
Split(X)	X is split break into submessages $x_1, ..., x_n$.

Thereinto, N'_{D_a} realizes to combine N_{D_a} and N_{D_b} into a whole, in which $\{M_{D_a}, M_{D_b}\}$, $\{PID_{D_a}, PID_{D_b}\}$, $\{k_{D_a}, k_{D_b}\}$, and $\{r_0, F_P\}$ are involved as parameters. Afterwards, D_a transmits $C_{D_a} \| N'_{D_a}$ to P for authentication.

P re-computes C_{D_a} by the locally derived $\{r_1, r_2\}$, and compares the re-computed C_{D_a} with the received C_{D_a} to verify the validity of D_a. If the two values are not identical, D_a will be regarded as an illegal entity. Till now, P has completes the authentication on $\{D_a, D_b\}$, and the yoking-proofs $YP = (r_0, F_P, F_{D_a}, F_{D_b}, N'_{D_a})$ will be established for the cloud sever to verify the validity of $\{D_a, D_b\}$.

IV. RUBIN LOGIC BASED FORMAL SECURITY ANALYSIS

Rubin logic [10] is applied for formal security analysis. The protocol should be reasonable by achieving the expectant security goals based on the logical knowledge and belief sets, actions, and inference rules. Table II defines the sets and actions for an entity E.

Rubin logic based formal analysis involves the following steps: 1) declaration of the initial specification of the global sets and local sets; 2) declaration of behavior list of principals; 3) verification by logical rules and formulas. The formalization of the protocol refers to specifying the protocol in the language of which provides rigorous rules of evaluation so that even subtle defects can be uncovered.

A. Specification

1) Global Set: Principal set $\{P, D_a, D_b\}$ contains the entities involved in the YPAP. P acts as an initiator to query the wearable devices D_* (i.e., D_a, D_b). Each secret $\{S, k_P, k_{D_*}, F_P, F_{D_*}, PID_{D_*}\}$ has an observer set.

$Observers(S, k_P)$=$\{P, D_*\}$,
$Observers(F_P, F_{D_*}, PID_{D_*})$=$\{P\}$,
$Observers(k_{D_*}, F_{D_*}, PID_{D_*})$=$\{D_*\}$.

2) Local Set: Suppose that all entities believe in the freshness of the pre-shared secrets. The initial local sets are defined for P, D_a, and D_b.

- **For the entity P:**
  $POSS(P)$=$\{S, k_P\}$
  $BEL(P)$=$\{\sharp(S), \sharp(k_P)\}$
  $BL(P)$=
  - Generate-nonce(r_0) $P.1$
    Generate-secret(F_P) $P.2$
    Send$(D_a, \{r_0, F_P\})$ $P.3$
    Update$(\{r_0, F_P\})$ $P.4$
    Receive$(D_a, \{F_{D_a}, M_{D_a}\})$ $P.5$
    Check-freshness(F_{D_a}) $P.6$
    Generate-nonce(r_1, r_2) $P.7$
    Split(S) $P.8$
    Generate-secret(S_{1P}, S_{2P}) $P.9$
    Send$(D_b, \{r_0, F_P,$ Concat(PID_{D_b}, S_{1P}, r_1), Concat(Encrypt$(k_P, \{F_{D_b}, r_0\})$, S_{2P}, $r_2)$, $M_{D_a}\})$ $P.10$
    Update$(\{A_{D_b}, B_{D_b}\})$ $P.11$
    Receive$(D_b, \{F_{D_b}, C_{D_b}, M_{D_b}, N_{D_b}\})$ $P.12$
    Check-freshness(C_{D_b}) $P.13$
    Send$(D_a$, {Concat(PID_{D_a}, S_{1P}, r_1), Concat(Encrypt$(k_P, \{F_{D_a}, r_0\})$, S_{2P}, $r_2)$, $N_{D_b}\})$ $P.14$
    Update$(\{A_{D_a}, B_{D_a}\})$ $P.15$
    Receive$(D_a, \{C_{D_a}, N'_{D_a}\})$ $P.16$
    Check-freshness(C_{D_a}) $P.17$
- **For the entity D_a:**
  $POSS(D_a)$=$\{S, k_P, k_{D_a}, PID_{D_a}\}$
  $BEL(D_a)$=$\{\sharp(S), \sharp(k_P), \sharp(k_{D_a}), \sharp(PID_{D_a})\}$
  $BL(D_a)$=
  Receive$(P, \{r_0, F_P\})$ $D_a.1$
  Check-freshness(F_P) $D_a.2$
  Generate-secret(F_{D_a}) $D_a.3$
  Send$(P$, $\{F_{D_a}$, Concat$(PID_{D_a}$, Encrypt$(k_{D_a}, \{F_P, r_0\})))\})$ $D_a.4$
  Update(F_{D_a}, M_{D_a}) $D_a.5$
  Receive$(P, \{A_{D_a}, B_{D_a}, M_{D_b}, N_{D_b}\})$ $D_a.6$
  Split(S) $D_a.7$
  Generate-secret(S_{1P}, S_{2P}) $D_a.8$
  Send$(P$, {Concat(r_1, r_2, PID_{D_a}), Concat$(PID_{D_a}, r_0, k_P, M_{D_b}, N_{D_b})\})$ $D_a.9$
  Update$(\{C_{D_a}, N'_{D_a}\})$ $D_a.10$
- **For the entity D_b:**
  $POSS(D_b)$=$\{S, k_P, k_{D_b}, PID_{D_b}\}$
  $BEL(D_b)$=$\{\sharp(S), \sharp(k_P), \sharp(k_{D_b}), \sharp(PID_{D_b})\}$
  $BL(D_b)$=
  Receive$(P, \{r_0, F_P, A_{D_b}, B_{D_b}, M_{D_a}\})$ $D_b.1$
  Check-freshness(F_P) $D_b.2$
  Split(S) $D_b.3$

Generate-secret(S_{1P}, S_{2P}) $D_b.4$

Generate-secret(F_{D_b}) $D_b.5$

Send(P, $\{F_{D_b}$, Concat(r_1, r_2, PID_{D_b}),
Concat(PID_{D_b}, Encrypt(k_{D_b}, $\{F_P$, $r_0\}$)),
Concat(M_{D_a}, PID_{D_b}, r_0, k_P)$\}$) $D_b.6$

Update($\{F_{D_b}$, C_{D_b}, M_{D_b}, $N_{D_b}\}$) $D_b.7$

In the YPAP, an initial action in $BL(P)$ is marked with "•". Applying inference rules, the next action is marked with "∘" to show that it has been successfully executed, then "•" is moved to the next action. The control flow shows how the analysis proceeds sequentially through the behavior list of P, D_a, and D_b. The actions Send(.) and Update(.) are bound together, and the analysis moves to the next Receive(.) of the principal specified in the previous Send(.) after each Update(.).

B. Logical Analysis

The logic analysis is based on the initial specification, and related inference rules provided by Rubin logic [10]. The first four actions $P.1$-$P.4$ in $BL(P)$ are executed resulting in new elements added to the sets $POSS(P)$ and $BEL(P)$. The update action $P.4$ causes $Observers(r_0, F_P) = P$, i.e. $\{r_0$, $F_P\}$ are known by P.

- **For the entity P:**
  $POSS(P)$=$\{S$, k_P, r_0, $F_P\}$
  $BEL(P)$=$\{\sharp(S)$, $\sharp(k_P)$, $\sharp(F_P)$, $LINK(r_0)\}$
  $BL(P)$=
  ......
  ∘ Send(D_a, $\{r_0$, $F_P\}$)
  ∘ Update($\{r_0$, $F_P\}$)

Upon $P.4$ being executed, the next actions turn to D_a's behavior list. The actions $D_a.1$-$D_a.5$ are executed, and the updated local set of D_a is as follows.

- **For the entity D_a:**
  $POSS(D_a)$=$\{S$, k_P, k_{D_a}, PID_{D_a}, F_{D_a}, M_{D_a}, $\{r_0, F_P\}fromP\}$
  $BEL(D_a)$=$\{\sharp(S)$, $\sharp(k_P)$, $\sharp(k_{D_a})$, $\sharp(PID_{D_a})$, $\sharp(F_{D_a})$, $\sharp(F_P)\}$
  $BL(D_a)$=
  ......
  ∘ Send(P, $\{F_{D_a}$, Concat(PID_{D_a}, Encrypt(k_{D_a}, $\{F_P$, $r_0\}$))$\}$)
  ∘ Update(F_{D_a}, M_{D_a})

Upon $D_a.5$ being executed, the next actions turn to P's behavior list. The actions $P.5$-$P.11$ are executed, and the updated local set of P is as follows.

- **For the entity P:**
  $POSS(P)$=$\{S$, S_{1P}, S_{2P}, k_P, r_0, r_1, r_2, F_P, $\{F_{D_a}, M_{D_a}\}fromD_a$, $\{A_{D_b}$, $B_{D_b}\}\}$
  $BEL(P)$=$\{\sharp(S)$, $\sharp(k_P)$, $\sharp(F_P)$, $\sharp(F_{D_a})$, $LINK(r_1)$, $LINK(r_2)$, $LINK(S_{1P})$, $LINK(S_{2P})\}$
  $BL(P)$=
  ......
  ∘ Send(D_b, $\{r_0$, F_P, Concat(PID_{D_b}, S_{1P}, r_1), Concat(Encrypt(k_P, $\{F_{D_b}$, $r_0\}$), S_{2P}, r_2), $M_{D_a}\}$)
  ∘ Update($\{A_{D_b}$, $B_{D_b}\}$)

Upon $P.11$ being executed, the next actions turn to D_b's behavior list. The actions $D_b.1$-$D_b.7$ are executed, and the updated local set of D_b is as follows.

- **For the entity D_b:**
  $POSS(D_b)$=$\{S$, k_P, k_{D_b}, PID_{D_b}, F_{D_b}, C_{D_b}, M_{D_b}, N_{D_b}, $\{r_0, F_P, A_{D_b}, B_{D_b}, M_{D_a}\}fromP$,
  $BEL(D_b)$=$\{\sharp(S)$, $\sharp(k_P)$, $\sharp(k_{D_b})$, $\sharp(PID_{D_b})$, $\sharp(F_{D_b})$, $\sharp(F_P)$, $LINK(S_{1P})$, $LINK(S_{2P})\}$
  $BL(D_b)$=
  ......
  ∘ Send(P, $\{F_{D_b}$, Concat(r_1, r_2, PID_{D_b}), Concat(PID_{D_b}, Encrypt(k_{D_b}, $\{F_P$, $r_0\}$)), Concat(M_{D_a}, PID_{D_b}, r_0, k_P)$\}$)
  ∘ Update($\{F_{D_b}$, C_{D_b}, M_{D_b}, $N_{D_b}\}$)

Upon $D_b.7$ being executed, the next actions turn to P's behavior list. The actions $P.12$-$P.15$ are executed, and the updated local set of P is as follows.

- **For the entity P:**
  $POSS(P)$=$\{S$, S_{1P}, S_{2P}, k_P, r_0, r_1, r_2, F_P, $\{F_{D_b}, C_{D_b}, M_{D_b}, N_{D_b}\}fromD_b$, $\{A_{D_a}$, $B_{D_a}\}\}$
  $BEL(P)$=$\{\sharp(S)$, $\sharp(k_P)$, $\sharp(F_P)$, $\sharp(F_{D_a})$, $\sharp(F_{D_b})$, $LINK(r_1)$, $LINK(r_2)$, $LINK(S_{1P})$, $LINK(S_{2P})\}$
  $BL(P)$=
  ......
  ∘ Send(D_a, $\{$Concat(PID_{D_a}, S_{1P}, r_1), Concat(Encrypt(k_P, $\{F_{D_a}$, $r_0\}$), S_{2P}, r_2), $N_{D_b}\}$)
  ∘ Update($\{A_{D_a}$, $B_{D_a}\}$)

It is obtained that $\{F_P\}fromP \in POSS(D_a/D_b)$, which means that F_P is marked as being received from P, and is possessed by D_a and D_b.

According to the message meaning rule:

$$\frac{\{X\}_k fromQ \in POSS(E), k \in POSS(E)}{BEL(E) := BEL(E) \cup \{X \in POSS(Q)\}}$$

It is obtained that $BEL(D_a/D_b) := BEL(D_a/D_b) \cup \{F_P \in POSS(P)\}$, which means that D_a and D_b believe that P possesses F_P. Thereinto, F_P is a plaintext without applying k. It is obtained that:

1) $(F_P \in POSS(P)) \in BEL(D_a/D_b)$: P possesses F_P, and D_a and D_b believe the fact;
2) $\sharp(F_P) \in BEL(D_a/D_b)$: D_a and D_b believe that F_P is fresh;
3) $F_P fromP \in POSS(D_a/D_b)$: F_P is from P, and is possessed by D_a and D_b.

According to the nonce verification rule:

$$\frac{(X \in POSS(E)) \in BEL(Q), \quad \sharp(X) \in BEL(E), X fromQ \in POSS(E)}{BEL(E) := BEL(E) \cup \{Q\ believes \sharp(X)\}}$$

It is obtained that $BEL(D_a/D_b)) := BEL(D_a/D_b)) \cup \{R\ believes \sharp(F_P)\}$, which means that D_a/D_b believes that P believes that F_P is fresh, and the fact is added into $BEL(D_a/D_b))$. Similarly, the nonce verification rule can be applied to obtain that P believes that D_a/D_b believes that F_{D_a}/F_{D_b} is fresh, and the fact is added into $BEL(P)$.

Till now, it is obtained that:

1) $\sharp(k_P) \in BEL(P)$: P believes that k_P is fresh;
2) $k_P \in POSS(P)$: P possesses k_P in $POSS(P)$;
3) $LINK(S_{1P}) \in BEL(P)$, $LINK(S_{2P}) \in BEL(P)$: $LINK(S_{1P})$ and $LINK(S_{2P})$ are in P's belief set $BEL(P)$, and they have not been used in former session. Hereafter $LINK(S_{1P})$ and $LINK(S_{2P})$ are removed from $BEL(P)$;
4) $\{F_{D_b}, C_{D_b}, M_{D_b}, N_{D_b}\} contains f(S_{1P}, S_{2P})\}$: $\{F_{D_b}, C_{D_b}, M_{D_b}, N_{D_b}\}$ contains the functions $f_1(S_{1P})$ and $f_2(S_{2P})$;
5) $\{F_{D_b}, C_{D_b}, M_{D_b}, N_{D_b}\} contains C_{D_b}$: C_{D_b} is the submessage of $\{F_{D_b}, C_{D_b}, M_{D_b}, N_{D_b}\}$;
6) $\{F_{D_b}, C_{D_b}, M_{D_b}, N_{D_b}\} from D_b \in POSS(P)$: $\{F_{D_b}, C_{D_b}, M_{D_b}, N_{D_b}\}$ is sent from D_b, and is possessed by P in $POSS(P)$.

According to the linkage rule:

$$\frac{\begin{array}{c}\sharp(k) \in BEL(E), k \in POSS(P), \\ LINK(r) \in BEL(E), X contains f(r), \\ X contains\ x_1, \{X\}_k from Q \in POSS(E)\end{array}}{BEL(E) := (BEL(E) - LINK(r)) \cup \{\sharp(x_1)\}}$$

It is obtained that $BEL(P) := (BEL(P) - LINK(S_{1P}) - LINK(S_{2P})) \cup \{\sharp(C_{D_b})\}$, which means that any submessage of a valid response is believed to be fresh by the receiver. Thus, P believes that the submessage C_{D_b} is fresh.

Upon P.15 being executed, the next actions turn to D_a's behavior list. The actions D_a.6-D_a.10 are performed to add $\{A_{D_a}, B_{D_a}, M_{D_b}, N_{D_b}\} from P$ into $POSS(D_a)$. D_a performs the similar operations as D_b to obtain $\{C_{D_a}, N'_{D_a}\}$, which is sent to P for authentication. The actions P.16 and P.17 are performed to check the freshness of C_{D_a}. The freshness of C_{D_a} can be proved according to the procedure of proving the freshness of C_{D_b}.

Hence, the YPAP is analyzed by Rubin logic, in which P, D_a, and D_b build beliefs during the authentication by checking the freshness of the exchanged messages. It indicates that the YPAP is logically correct and can ensure the nonexistence of obvious design defects.

V. CONCLUSION

In this work, a unique security issue is identified for the wearable devices during wireless communications, and a yoking-proofs based authentication protocol (YPAP) is proposed to achieve both secure authentication and simultaneous identification. The YPAP establishes yoking-proofs by involving two associated wearable devices into one session, and adopts lightweight cryptographic operators for authentication. The random partition, dynamic update and quick check mechanisms are jointly applied with security and efficiency considerations. Moreover, Rubin logic is applied to prove that the YPAP has theoretical design correctness. It indicates that the proposed protocol owns advantages for the resource-constrained wearable devices.

ACKNOWLEDGMENT

This work is supported by Science and Technology Foundation of Beijing (Grant No. Z141100002714003).

REFERENCES

[1] A. Pyattaev, K. Johnsson, S. Andreev, and Y. Koucheryavy, "Communication Challenges in High-density Deployments of Wearable Wireless Devices," *IEEE Wireless Communications*, vol. 22, no. 1, pp. 12-18, 2015.

[2] Y. Lin, Y. Lin, C. Chih, et al., "EasyConnect: A Management System for IoT Devices and Its Applications for Interactive Design and Art," *IEEE Internet of Things Journal*, 2015.

[3] D. He, S. Chan, and M. Guizani, "User Privacy and Data Trustworthiness in Mobile Crowd Sensing," *IEEE Wireless Communications*, vol. 22, no. 1, pp. 28-34, 2015.

[4] R. Kirkham, and C. Greenhalgh, "Social Access vs. Privacy in Wearable Computing: A Case Study of Autism," *IEEE Pervasive Computing*, vol. 14, no. 1, pp. 26-33, 2015.

[5] S. D. Kim, S. M. Lee, and S. E. Lee, "Secure Communication System for Wearable Devices Wireless Intra Body Communication," in *Proceedings of 2015 IEEE International Conference on Consumer Electronics (ICCE)*, pp. 381-382, 2015.

[6] F. P. Diez, D. S. Touceda, J. M. S. Camara, and S. Zeadally, "Toward Self-authenticable Wearable Devices," *IEEE Wireless Communications*, vol. 22, no. 1, pp. 36-43, 2015.

[7] A. Juels, "'Yoking-proofs' for RFID Tags," in *Proceedings of the Second IEEE Annual Conference on Pervasive Computing and Communications Workshops*, pp. 138-143, 2004.

[8] H. Y. Chien and S. B. Liu, "Tree-Based RFID Yoking Proof," in *Proceedings of the International Conference on Networks Security, Wireless Communications and Trusted Computing (NSWCTC 2009)*, pp. 550-553, 2009.

[9] H. Liu, H. Ning, Y. Zhang, D. He, Q. Xiong, and L. T. Yang, "Grouping-proofs Based Authentication Protocol for Distributed RFID Systems," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 7, pp. 1321-1330, 2013.

[10] A. D. Rubin and P. Honeyman, "Nonmonotonic Cryptographic Protocols," in *Proceedings of Computer Security Foundations Workshop VII*, pp. 100-116, 1994.