Secure Authentication on the Internet of Things

Phillip H. Griffin
Griffin Information Security
Raleigh, North Carolina, United States
phil@phillipgriffin.com

***Abstract*—This paper describes biometric-based cryptographic techniques for providing confidential communications and strong, mutual and multifactor authentication on the Internet of Things. The described security techniques support the goals of universal access when users are allowed to select from multiple choice alternatives to authenticate their identities. By using a Biometric Authenticated Key Exchange (BAKE) protocol, user credentials are protected against phishing and Man-in-the-Middle attacks. Forward secrecy is achieved using a Diffie-Hellman key establishment scheme with fresh random values each time the BAKE protocol is operated. Confidentiality is achieved using lightweight cryptographic algorithms that are well suited for implementation in resource constrained environments, those limited by processing speed, limited memory and power availability. Lightweight cryptography can offer strong confidentiality solutions that are practical to implement in Internet of Things systems, where efficient execution, and small memory requirements and code size are required.**

Keywords— Authentication; Biometrics; Cryptography; Internet of Things

I. Introduction

The term Internet of Things (IoT) describes the "ongoing evolution of the Internet into a network of smart objects" [1]. These objects are internet-connected devices that can "communicate with each other and with centralized resources" [1]. Devices in the IoT can be very small in size, "as long as the underlying processor is large enough to support the TCP/IP protocol" using Internet Protocol version 6 (IPv6) [2].

Limited device size and computing capabilities present challenges to using strong cryptography when users of smart objects must authenticate their identities to gain access to systems and to communicate securely. Authentication methods that rely on the operation of a Public Key Infrastructure (PKI) can require too many computation, memory size, and bandwidth resources to support IoT implementations. Biometric authentication coupled with cryptographic techniques for symmetric key establishment can provide lightweight alternatives to PKI-based methods.

The development of lightweight cryptographic algorithms arose from the need to secure devices in the resource constrained IoT. The phrase *lightweight cryptography* describes the "cryptographic primitives, schemes and protocols tailored to extremely constrained environments" [1]. In this description, the term 'lightweight' does not mean "weak cryptography", but cryptography that is efficient when measured in terms of its "execution time, runtime memory (i.e. RAM) requirements, and binary code size" [1].

Small, low power devices such as Radio Frequency Identification (RFID) tags, "sensors in wireless sensor networks" (WSN) or more generally, in "small internet-enabled appliances" make up a large part of the growing IoT landscape [3]. RFID technology has become "a key enabler of modern supply-chain management and industrial logistics" [1]. WSN have been widely adopted in applications from "home automation", to "environmental surveillance and traffic control to medical monitoring" [1]. When physical objects associated with biometric reference templates have been preregistered with relying parties, they can be paired with networked biometric sensors and used for mutual and multifactor authentication in access control systems [4].

II. Biometric Security Techniques

A. Biometric Identity Authentication

Identity authentication is a security control used to manage risk of unauthorized access to IoT devices and information systems. Biometrics-based techniques can be used to implement access control systems to provide "*something-you-are* options that support ease of use and Universal Access (UA)" in identity authentication systems [5]. Biometric technology can provide inclusive identity authentication to persons with diverse abilities (e.g., to see, to hear, or to speak) when they are presented with a set of biometric type alternatives and allowed to select one that is convenient for them to use.

When biometric identity authentication can be coupled with possession objects and user knowledge, to eliminate the need for costly and cumbersome PKI-based security controls, such as Transport Layer Security (TLS). When a biometric authentication factor is paired with other factors, it is possible to provide both low cost, effective, easy to use mutual and multifactor authentication systems. These systems can support universal access, and leverage BAKE and its underlying Password Authenticated Key Exchange (PAKE) and Diffie-Hellman Key Exchange techniques to establish a channel for confidential communications [5].

It's hard to imagine requiring most users to properly configure and consistently manage the security of IoT products that rely on PKI-based protocols for secure authentication and confidential communications. PKI protocols such as TLS challenge even experienced information technology professionals. These security protocols are ill-suited for many

home users or the elderly or infirm populations in ambient assisted living environments. Instructing users to configure TLS and to "avoid collision attacks by being sure not to use a 64-bit block size block cipher algorithm such as Blowfish or 3DES in CBC mode" are likely to be well beyond the technical capabilities of many users.

Though TLS protocol implementations could avoid some of the more difficult dialogs by relying on preset, limited configuration options, each IoT implementation "may choose to support a different subset of cipher suites" [6]. Therefore, for two implementations to interoperate they must both support "protocol agility" and be able to "negotiate a common ciphersuite"[6]. This ability of a protocol to be agile also allows IoT implementations to "transition from old cryptographic algorithms to new ones" [6].

B. *Telebiometric Authentication Objects*

The term telebiometric authentication objects (TAO) describes “tagged physical objects functionally coupled with biometric sensors and connected to a telecommunications network” [4]. TAO can be used to provide “strong, low cost mutual and multi-factor authentication on the Internet of Things (IoT)” that support the goals of universal access in access control systems that can be convenient for people to use [5].

These objects "do not require users to carry individually assigned security tokens, remember complex passwords, or possess and manage cryptographic keys and public key certificates" [4]. Users can be assigned objects in their environments such as door locks, appliances, and medical devices. Persons that are enrolled in a biometric system can be "associated with a set of physical objects" by mapping their biometric reference template identifiers to a set of one or more unique RFID tags [4].

Pre-registered tagged object/biometric associations can be used to provide *something-you-have* and *something-you-are* factors during identity authentication. Some biometric sensor data can contain both *something-you-are* and *something-you-know* factors. This dual factor attribute of biometrics, the ability to verify a physical identity and also collect user knowledge, can simplify the authentication process for the user while providing them the security benefits of strong, multifactor authentication.

For example, a microphone can serve as a biometric sensor that collects humans as they speak. Using a single sensor, both biometric matching data and user knowledge (i.e., in the form of a spoken password) can be collected from a single source. These factors can be combined in an access control system to enable multi-factor user authentication.

C. *Substitution Strings*

A *something-you-know* authentication factor can be extracted from observations of a sequence of gestures collected by an image-based biometric authentication system [7]. Both biometric matching and knowledge matching can be performed on elements of this data. User provided gestures based on American Sign Language (ASL) hand signs are traditionally used to represent the single alphabetical characters indicated in Figure 1. Using this single character representation, a sequence of hand signs provided by a user can be registered as their password [7]. A user's password characters can be extracted from their presented hand signs and used as inputs to operate an Authenticated Key Exchange (AKE) protocol [7].

Fig 1. American Sign Language [8]

The knowledge information extracted from user gesture observations need not represent, in a direct or traditional way, the character string that is input to an AKE protocol. A substitution string that serves as a proxy for the user's expressed characters can be used instead, if the substitution string and gesture mapping are known to both client and server. So, a simple, easy to remember and use sequence of hand sign gestures can be associated with complex sets of characters, as illustrated by the example simple gesture password "BAKE" in Table 1.

TABLE I. EXAMPLE HAND SIGN SUBSTITUTION STRING MAPPING

Hand Sign	Substitution String
	`R'W]$Pq57]mbTk`
	`#QsWK}um<~k3D%`
	`hLNSaCF#<`A!U2`
	`whl={H04<"%A;U`

In this example password, each of the hand signs have been mapped to a string of characters far too complex for a user to

easily recall, and too long to present to an access control system quickly. By representing the sign language signs of a user password with substitution strings, short sequences of signs result in a strong password input to the AKE protocol. Single and short gesture sequences can be critical for ensuring secure access to IoT devices and information systems in Ambient Assisted Living (AAL) environments.

To provide the benefits of universal access to infirm and elderly individuals who may reside in medical or AAL environments, access controls that support convenient and easy to use identity authentication are necessary. These same user interface characteristics are also desirable in other environments. These environments include the financial services where institutions strive to improve customer convenience while at the same time safeguarding access to restricted systems, and in applications in which response time is critical, such as air traffic control or defense systems.

D. Changing User Passwords

Password expiration is a widely practiced security control whose intended purpose is to force the user to change their password in order to revoke "access to an account by an attacker who has captured the account's password" [9]. While legitimate users may be inconvenienced at having to choose and remember a new password, recent research indicates that attackers are not. They can guess a new user password based on prior patterns chosen by the user, so that the result is that "the effectiveness of expiration in meeting its intended goal is weak" [9].

When substitutions strings are used as the shared secret needed to operate an authenticated key exchange (AKE) protocols, such as BAKE and PAKE, the user gains several benefits. They can keep using their simple, familiar, and easy to recall base passwords, as their more complex substitution strings are updated.

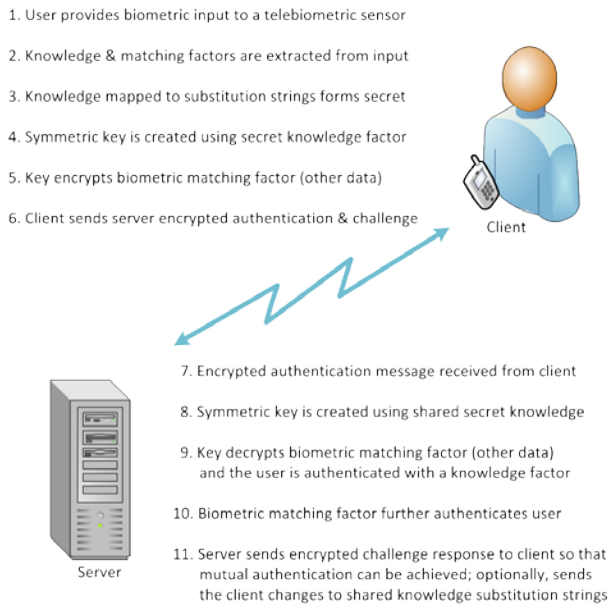
E. Biometric Authenticated Key Exchange

BAKE protocols extract knowledge information from biometric sensor data and use these extracted strings to operate Password Authenticated Key Exchange (PAKE) protocols as illustrated in Figure 2.

A Diffie-Hellman Key Exchange protocol is performed by the Client in step 4 and by the Server in step 8 using the password extracted from a user biometric sample in step 2. The substitution strings for the shared user password are refreshed by the Server in step 11. The Server must wait to update these strings for use in subsequent authentication attempt until the user acknowledges their receipt over the secure channel established at the completion of the BAKE protocol.

If the Client fails to acknowledge their receipt, the Server should assume that the protocol has been interrupted and that the substitution strings used in the current execution of the protocol remain unchanged. At most two sets of password substitution strings for each user must be maintained by the server at any one time. This allows the user to enjoy the benefits of one-time passwords without interrupted access and loss of server availability.

Fig 2. BAKE Protocol



III. LIGHTWEIGHT CRYPTOGRAPHY

A. Practical IoT Symmetric Algorithms

The Diffie-Hellman key exchange protocol allows communicating parties to establish a shared secret key on an unprotected network. Having established the key, subsequent communications over an insecure channel can be encrypted using a symmetric key cipher to ensure data confidentiality during transfer. Though the widely used Advanced Encryption System (AES) is a strong and efficient symmetric key cipher, it is considered '*to heavy*' for use in resource constrained IoT environments, and not a best choice alternative for many implementations.

Confidentiality is best achieved using lightweight cryptographic algorithms that are well suited for implementation in resource constrained environments, those limited by processing speed, limited memory and power availability. Lightweight cryptography can offer strong confidentiality solutions that are practical to implement in IoT systems, where efficient execution, and small memory requirements and code size are required.

B. Constraints

The environment in which IoT security is needed "includes extremely constrained platforms, with as little as 64 KiB ROM and 4 KiB RAM" [2]. These constraints make it difficult to implement strong cryptographic solutions. Though there has been some success in developing lightweight versions of AES, "the Advanced Encryption Standard", AES has "widely been deemed too complex and energy-hungry for the RFID environment" [10].

Securing these devices requires encryption algorithms that can achieve "high levels of security using only a small computing power" [3]. The lack of suitable, lightweight cryptographic primitives suitable for use in the IoT has

spawned much research from academia, government, and industry. These efforts have resulted in a number of new, lightweight "streamciphers, blockciphers, hashfunction and recently one-pass authenticated encryption" [3] proposals.

The 'weight' of a lightweight cryptographic primitive is roughly "the amount of resources necessary in both time and space for it to run" [3]. How the weight of a primitive is measured depends on context, on whether it targets a hardware or software solution. A primitive that is lightweight in one context may not be lightweight in the other. However, a common measure that is " relevant in both contexts is the power consumption" [3].

There are often tradeoffs to be made in achieving design goals, and some block cipher design choices may result in optimal performance only for specific platforms. Some block ciphers that target constrained devices have been specifically designed to perform well "on dedicated Application-Specific Integrated Circuits (ASICs)" [11]. Other block ciphers have been designed to optimize performance "on low-cost microcontrollers with limited flash, SRAM, and/or power availability", and those that do well on one platform may not perform well on the other [11].

C. Standarization

The 2012 edition of the ISO/IEC 29192-2 lightweight cryptography standard [12] defines two block cipher algorithms, CLEFIA and PRESENT, one 128-bit and the other 64-bit.

The CLEFIA algorithm is a "symmetric block cipher that can process data blocks of 128 bits" [12]. CLEFIA uses cryptographic key lengths of "128, 192, or 256 bits" [12]. The number of rounds and round keys that are required varies by key length, as illustrated in the following table:

TABLE II. CLEFIA CHARACTERISTICS

CLEFIA			
Block size (bits)	**Key length (bits)**	**Number of Rounds**	**Round Keys**
128	128	18	36
128	192	22	44
128	256	26	52

The PRESENT algorithm is a 31-round "symmetric block cipher that can process data blocks of 64 bits, using a key of length" of 80-bits for PRESENT-80, and 128-bits for PRESENT-128 [12]. Each processing round consists of a "sequence of simple transformations" [12].

The valid cipher key lengths for PRESENT can be represented using Abstract Syntax Notation One (ASN.1) as a value of type `PresentKeyLengths` defined as follows:

```
PresentKeyLengths ::= INTEGER {
      k80(80), k128(128) }
           ( k80 | k128 )
```

The fundamental structure of CLEFIA is based on a generalized Feistel network that is used in both the "data processing part and the key schedule " [12]. The valid key lengths for CLEFIA can be represented using ASN.1 as a value of type `ClefiaKeyLengths` defined as follows:

```
ClefiaKeyLengths ::= INTEGER {
k128(128), k192(192), k256(256) }
       ( k128 | k192 | k256 )
```

The 2016 revision of ISO/IEC 29192-2 adds two additional families of block ciphers, SIMON and SPECK. These are relatively new lightweight algorithm proposals only recently "put forth by researchers from the National Security Agency (NSA) of the USA" in June, 2013 [13]. Both are believed to be "efficient and secure" algorithms that can enable solutions that are "low-cost and easy to implement and deploy on multiple platforms" [13]. This makes them appealing for use in IoT applications "from mobile devices, through RFID tags to electronic locks" [13].

They are being rapidly adopted as international standards based largely on how they compare with their predecessors. Both SIMON and SPECK demonstrate "very competitive performance, small memory footprint" that beats "most existing lightweight ciphers in terms of efficiency and compactness" [13]. They have "very simple and elegant" designs that are "built on the ARX philosophy" [13], where "ARX stands for Addition/Rotation/XOR" [14]. ARX describes "a class of cryptographic algorithms based on the simple arithmetic operations: modular addition, bitwise rotation (and bitwise shift) and exclusive-OR", operations that have a long history of use in designs dating back to the 1980's [14].

These block cipher families have block sizes of "32, 48, 64, 96, and 128 bits, with up to three key sizes for each block size" [10]. The ISO/IEC 29192-2 revision includes all of these block sizes except for the smallest, as shown in the following table:

TABLE III. SIMON AND SPECK CHARACTERISTICS

Block size (bits)	**Key length (bits)**
48	96
64	96
64	128
96	96
96	144
128	128
128	192
128	256

SIMON and SPECK provide "lightweight ciphers which are secure, flexible and efficient across a wide range of applications", and both have been designed "specifically for

resource-constrained devices" [15]. These features make them ideal for deployment in home health care and assisted living environments [16[. Both algorithms use only "basic arithmetic operations such as modular addition, XOR, bitwise AND and bit rotation" [13]. Together they offer "great performance on hardware and software platforms", with the SIMON block cipher "designed towards hardware applications and SPECK for software applications" [15].

IV. Conclusions

Biometric-based access controls and authenticated key exchange protocols such as BAKE can help manage the security risk of unauthorized access to IoT devices and the information and communications technology systems to which they connect. When coupled with lightweight cryptography, BAKE and the underlying PAKE protocol can ensure the confidentiality of communications. These protocols offer strong, multi-factor user identity authentication.

BAKE and PAKE also provide the additional assurance that users gain through the use of mutual authentication, assurance that they are actually connected to the systems they intended to connect to - systems that they can trust. When fresh random values are chosen each time the underlying Diffie-Hellman Key Exchange protocol is operated, both BAKE and PAKE protocols can thwart phishing and man-in-the-middle attacks.

Lightweight symmetric key cryptography solutions such as SIMON and SPECK are suitable for use in the constrained environments of the IoT. Though lightweight, these algorithms are not weak. Both SIMON and SPECK can provide the cryptographic strength needed to protect sensitive user credentials during identity authentication, and during subsequent communications. When these keys are derived using strong, password substitution strings that can be refreshed within a PAKE protocol, user convenience of easy to remember passwords can be maintained while ensuring complex and changing inputs are provided to the underlying key exchange protocols. With the use of protected password substitution strings, users enjoy the security benefits of one-time-passwords, without being required to frequently change their actual passwords.

References

[1] D. Dinu, Y. Le Corre, D. Khovratovich, L. Perrin, J. Großschädl, and A. Biryukov. "Triathlon of lightweight block ciphers for the internet of things." *IACR Cryptology ePrint Archive* 2015 (2015): 209.

[2] G. Pereira, C. Puodzius, and P. SLM Barreto. "Shorter hash-based signatures." *Journal of Systems and Software* 116 (2016): 95-100.

[3] D. Dinu and L. Perin, Lightweight cryptography from CryptoLUX.

[4] P. Griffin, "Telebiometric authentication objects." *Procedia Computer Science* 36 (2014): 393-400.

[5] P. Griffin, "Biometric-based cybersecurity techniques." In *Advances in Human Factors in Cybersecurity*, pp. 43-53. Springer International Publishing, 2016.

[6] K. Bhargavan and G. Leurent. "On the practical (In-) security of 64-bit block ciphers: Collision attacks on HTTP over TLS and OpenVPN." In *ACM Conference on Computer and Communications Security.*

[7] P. Griffin, "Biometric knowledge extraction for multi-factor authentication and key exchange." *Procedia Computer Science* 61 (2015): 66-71.

[8] W. Vicars, *American Sign Language (ASL)*. (2011). Retrieved January 14, 2017, from http://www.lifeprint.com

[9] Y. Zhang, F. Monrose, and M. K. Reiter. "The security of modern password expiration: An algorithmic framework and empirical analysis." In *Proceedings of the 17th ACM conference on Computer and communications security*, pp. 176-186. ACM, 2010.

[10] M. Saarinen and D. Engels. "A do-it-all-cipher for rfid: Design requirements." *IACR Cryptology ePrint Archive* 2012 (2012): 317.

[11] B. Ray, S. Douglas, S. Jason, T. Stefan, W. Bryan, and W. Louis. (2013). *The SIMON and SPECK families of lightweight block ciphers*. Cryptology ePrint Archive, Report./404.

[12] International Organization for Standardization (ISO) / International Electrotechnical Commission (IEC). (2012). ISO/IEC 29192-2 Information technology -- Security techniques -- Lightweight cryptography -- Part 2: Block ciphers

[13] A. Biryukov, A. Roy, and V. Velichkov. (2014, March). Differential analysis of block ciphers SIMON and SPECK. In *Fast Software Encryption* (pp. 546-570). Springer Berlin Heidelberg.

[14] A. Biryukov, V. Velichkov, and Y. Le Corre. (2016). Automatic search for the best trails in arx: Application to block cipher speck. In *Fast Software Encryption–FSE*.

[15] S. Bhasin, T. Graba, J. Danger, and Z. Najm. (2014, May). A look into SIMON from a side-channel perspective. In *Hardware-Oriented Security and Trust (HOST), 2014 IEEE International Symposium on* (pp. 56-59). IEEE.

[16] P. Griffin. (2015, December). "Security for Ambient Assisted Living: Multi-Factor Authentication in the Internet of Things." 2015 IEEE Globecom Workshops. doi:10.1109/glocomw.2015.7413961.