CHALLENGES AND BEST PRACTICES IN KBA SCHEMES

Prasoon Dwivedi
Software Engineer
RSA, The Security Division of EMC
Prasoon.Dwivedi@emc.com

Geoffrey Thomas
Principal Engineer
RSA, The Security Division of EMC
Geoffrey.Thomas@emc.com

Table of Contents

Disclaimer: The views, processes or methodologies published in this article are those of the authors. They do not necessarily reflect EMC Corporation's views, processes or methodologies.

Introduction

Computer system is a magnificent tool for realizing the dreams of humanity. However, improperly securing this vulnerable tool is certain to give nightmares to its users. Securing a computer system has always been a contest of wits: the adversary is always in search of holes, and the designer always tries to patch those holes. Pervasive computing has made it crucial to give security utmost importance while building these systems, which work on a variety of devices, locations, and forms. Secure and usable systems are a requirement of modern computing.

Harmony between usability and security is critical to a successful computing scheme. There is a delicate balance between the two and one is often compromised for the need of other. The same holds true for the three pillars of security; confidentiality, integrity, and availability. Knowledge-Based Authentication (KBA) is a common authentication method for identity verification, account recovery, and risk-based identification.

With the ever-increasing footprint of users on the internet it is imperative to implement robust authentication systems. KBA is an authentication technique used to prove the identity of an individual based upon the common knowledge shared between an individual and a service provider. This article explores the challenges in KBA and proposes best practices to implement a user-centric, secure authentication system. "What's your mother's maiden name?", "What's your favorite color?", “What’s the name of your first pet”? These are some of the security questions that KBA presents you with when you do not remember your password. These challenges serve as an alternate means of authentication for account recovery. The secret question can verify your identity so you can choose another password or have the system e-mail or SMS the means to recover the password. As a person is less likely to forget his first pet's name, security questions are considered great from a customer usability aspect.

KBA can be great from usability point of view. However, if implemented incorrectly, KBA becomes a terrible choice for systems, which require high level of identity proofing.

One may remember the incident [4] from 2008 that exposed the weakness in the static KBA system where unauthorized access was gained to the e-mail account of former Alaska Governor Sarah Palin. The account’s password could be reset using shared secret questions, including "Where did you meet your spouse?” along with the date of birth and zip code. For such public figure this information was easily available on the internet.

This article explores the challenges and best practices involved in implementing a secure, usable KBA scheme keeping the interests of developers, testers, architects, and end users in focus. Only by understanding the issues involved in a complex KBA system can one make correct decisions while designing, implementing, and using such systems.

Understanding Authentication

Before we start exploring KBA systems it is important to understand what authentication means, the factors that determine its resilience, and why it is of prime importance to have a secure authentication system for an overall secure system.

Authentication is any process with which you verify that someone or something is who or what they claim they are. In computer systems, it is the process of establishing confidence in the attribute or data presented by an entity to prove its identity before an information system.

The ways in which someone is authenticated falls into three categories.

1. **Knowledge:** Something the user *knows*. Passwords, PIN and responses to challenges are some of the examples of this factor of authentication.
2. **Ownership:** Something the user *has*. Identification card, a device, a hardware or software token installed on a trusted mobile device are some of the examples of this factor of authentication.
3. **Inheritance:** Something the user *does*. Fingerprint, retina pattern, DNA sequence, voice pattern, and other biometric identifiers are examples of this factor of authentication.

Depending on the degree of identity proofing needed, one or a combination of two and more factors are used for an authentication system. If a combination of two or more authentication factors are used, the authentication falls under the category of multifactor authentication. For example, in a secure system the user may be required to enter the combination of PIN, which he knows (knowledge), and token code generated by an authenticator (RSA SecurID token) which he has (ownership) for successful log in. It is recommended[1] to use at least two authentication factors when designing and implementing an authentication system.

Knowledge-Based Authentication

A Knowledge-Based Authentication system, as is clear from its name, is a method of authentication which uses the knowledge(secret) a user shares with the information system for identity verification. KBA schemes are often used along with multifactor authentication and for password retrieval, as they are considered great from a usability point of view.

KBA is broadly classified into two categories:

1. Static KBA
2. Dynamic KBA

Static KBA

Static KBA – also referred to as Shared Secret Authentication – is a common authentication method used for identity verification, risk-based authentication, and account recovery in case a user forgets his password. These are generally answers to demographic questions which a service provider collects during sign-up process.

Dynamic KBA

Dynamic Knowledge-Based Authentication also employs secret questions for identity verification but unlike static KBA the questions are generated on the fly and are based upon user activity, records, logs, and information garnered from public records. Generally, these questions are not stored in the database making it more difficult for an adversary to answer these questions.

Applications of KBA

1. **Self-Service Password Reset**: It is common for a user to forget his credentials or lockout his account while trying to authenticate using invalid attributes. In such cases, a user is required to prove his identity using alternate means of authentication. Calling the help desk is an option but this process is slow and requires big investments. KBA systems are of great use in such cases. To prove identity, the user is challenged with a set of n questions, for which the user sets answers during the initial sign-up process. For successful authentication, the user is required to answer m (m<=n) challenges correctly.
2. **Identity Proofing:** Electronic identity proofing is defined as the process of establishing confidence in the identity of a user presented before a user system. By using KBA for identity proofing, the individual to be authenticated proves that he or she knows or possesses some secret information shared between them and the information system. E-Authentication Guidance for Federal Agencies, [OMB 04-04] defines four levels of authentication, Levels 1 to 4, in terms of the consequences of the authentication errors and misuse of credentials. Level 1 is the lowest assurance and Level 4 is the highest. We shall discuss these levels in the later section [When to use KBA?] of this article.

   Standalone KBA is implemented in systems, which require low level identity proofing where a user rarely logs into his system and does not want to maintain a password for authentication.
3. **Risk-Based Authentication**: A risk-based authentication system dynamically calculates the risk score by taking into account the threat associated with an activity and the user agent performing that activity. Depending upon the risk score, it classifies the activity and challenges the end user with the corresponding level of authentication. As the level of risk score increases, the authentication process becomes more comprehensive and restrictive. KBA schemes alone or in combination with other authentication factors serve as the levels for authentication.

Why are KBA schemes enticing?

KBA schemes are popular for identity proofing for a number of reasons. We will evaluate KBA based on three factors:

1. Usability
2. Security and
3. Cost of Operation

1. **Usability:** We emphasized earlier that KBA schemes are indeed great from an end user usability aspect. They are easy to implement and end users are familiar with them, making them ideal for self-service consoles. Following are the parameters for evaluating the usability of a system and we shall examine where KBA fits in:
   - **a.** *Speed:* Speed determines how fast a task can be accomplished. KBA schemes are ideal for fast and real-time knowledge-based identity-proofing solutions. Based on an organizations need, an in-person or self-service based support structure is created with KBA as the base. Self-service is the preferred KBA solution implementation as it requires much less time to serve an end user.
   - **b.** *Efficiency:* This is a factor that determines how many mistakes are committed before accomplishing a task. The questions selected have secrets associated with them which are easy for the end user to remember. Thus, when challenged with these questions for identity proofing an end user makes minimal mistakes before completing the challenge successfully.
   - **c.** *Learnability: How easy is it to learn a new system?* This defines learnability. KBA systems use the simple concept of challenging the user with a set of questions. From an end user point of view this is something which he has been doing all his life making KBA easy to learn and remember
   - **d.** *Memorability:* Once learned, the ease of using the system is determined by the parameter of memorability. Like learnability this parameter of usability is also satisfied by KBA to a great extent. Though there are some points which need to be taken care of during the design phase. We discuss them later in this article.
   - **e.** *User Preference: What do users like?* Seamless, easy, and real-time identity proofing makes KBA an ideal choice.

2. **Security:** A well-designed and implemented KBA scheme is sufficient to meet the authentication requirements of a security system. A fine line between usability and security needs to be maintained for a successful KBA system. Neither usability nor security should be compromised for the need of the other. In a well-implemented KBA system the security question are easy, and their answers (secrets) are easy for the user to remember while at the same time are hard for an adversary to guess. The secrets do not change (or do not change very often) making them easy for the user to recollect even after a long time. A well-selected set of challenges and secrets makes the system more resilient from attacks.

3. **Cost of Operation:** The United States Federal Trade Commission (FTC) has estimated that almost ten million Americans discovered they were the victims of identity theft, with a total cost to businesses and consumers over $50 billion[3]. To mitigate the risks, FDIC (Federal Deposit Insurance Corporation) recommends[3] upgrading password-based single-factor user authentication systems to multi-factor authentication. KBA schemes are comparatively easier to implement and are cost-effective for the organizations as well. Adding a new layer of authentication by introducing an ownership factor (e.g. hardware token) over a pre-existing single-factor username-password based authentication scheme will require cost involved in procuring, distributing, and maintaining tokens, Fobs, or devices needed for identity validation. Similarly, to add a new layer of inheritance factor (e.g. fingerprint), installation and maintenance of biometric identification devices and peripherals is needed. However, these investments are not needed to set up a KBA scheme, saving organizations large sums in investment.

When to use KBA

Based on consequences of authentication errors or misuse of credentials, National Institute of Standards and Technology (NIST) has defined four authentication levels [2]. As the consequences become serious the level increases, with Level-4 at the highest. We briefly visit these levels below:

Level 1: This level does not require identity proofing though it provides methods to verify that the same user who claims the resource has access to it. A number of authentication methods can be use at this level, i.e. a simple password-based challenge. In addition, the authentication methods employed at Levels 2, 3, and 4 can also be used at this level. Passwords in plain text are never transmitted over the network and long term shared secrets may be revealed to the verifier. For password- or PIN-based authentication systems, this level requires the probability of success of an online password guessing attack by an attacker who has no prior knowledge of the password, but knows the user name of the target, shall not exceed 2-10 (1 in 1024)

Level 2: Single factor 'over the network' authentication is introduced at Level 2. Users need to prove their identity through a secure authentication protocol for successful authentication. Long-term shared secrets are never revealed unless operated by a Credential Service Provider (CSP). This level of authentication prevents eavesdropping, replay, and online-guessing attacks. For password- or PIN-based authentication systems, this level requires the probability of success of an on-line password guessing attack by an attacker who has no a prior knowledge of the password, but knows the user name of the target, shall not exceed 2-14 (1 in 16384)

Level 3: This level introduces multi-factor authentication over remote network. In addition to identity proofing, this level also requires verification of identity material and information. Claimants need to prove over the cryptographic channel that they control the token and unlock it using a password and use the unlocked token to prove his identity. The tokens are classified into three categories: Hard Token, Soft Token, and One Time Password (OTP).

Level 4: This level defines the highest level of 'over the network' authentication. Similar to Level 3, this level is based upon possession of a key through cryptographic channels except that only hard tokens validated at FIPS 140 Level 2 or higher are accepted at this level.

Organizations must evaluate the risks associated with their deployment and authentication needs. Additionally, detailed study of consequences when an authentication system fails or is misused by an adversary must be performed. An appropriate level of authentication which confirms their authentication and compliance requirements can then be selected.

From our study of NIST-defined authentication levels it is evident that a standalone KBA scheme will be able to completely meet the authentication guidelines of Level 1 and Level 2 only. As Level 3 requires multi-factor authentication, a KBA scheme in conjugation with other authentication methods can be used, i.e. sending the challenge question to a trusted device. KBA cannot be used in Level 4 authentication as it is mandatory to use a hardware token at this level.

Authentication Levels and Knowledge-Based Authentication

Authentication Level	Can KBA be used?
Level 1	Yes
Level 2	Yes
Level 3	Yes (if used along with other authentication mechanisms)
Level 4	No (as it requires hardware token)

Challenges in KBA

Standalone KBA schemes for authentication are no longer considered secure for identity proofing. The advent of social media and exponential development in social engineering attacks in the last few years has made KBA much more vulnerable. There are a multitude of reasons that make KBA an inappropriate choice for systems which require a high degree of identity proofing. Below are a prominent few discussed in brief.

- Searchable personal data
- Privacy challenges
- The domino effect
- Usability compromise

The concept of KBA is based upon shared secrets between the user and the authentication service. Historically, the challenge questions in KBA are based upon the demographic data, answers to which are easy for the user to remember. Social networking has made it very easy for an adversary to mine out someone’s KBA secrets. “What’s your mother’s maiden name?”, “What is your childhood’s friend’s name?” or “In which city were you born? These are among the most common questions found in the list of KBA challenges. The secrets to these challenges are not that hard to uncover if the adversary visits the user’s Facebook, Google+, or any other social networking profile. Often a simple Google search will return the result for these questions. People seem to share their world on Facebook, Twitter, LinkedIn, etc. The Internet and the vast measure of searchable personal data it stores has largely eliminated KBA as an effective means of identity validation.

Static KBA is considered great from a usability point of view but from the security aspect, it is terrible. Though better than nothing there are a number of problems associated with static KBA, which make it a bad choice for systems which require high levels of identity proofing.

Password reset has become the most prevalent way for adversaries to hijack user accounts. The hackers frequently target services that employ static KBA for user authentication and account recovery by harvesting publicly available information about a user and by social engineering attacks.

A number of people close to the user may know answers to these ‘secret’ questions. The secret question set is usually limited and the answers are generally demographic data concerning the user. This poses a great danger, as this information can be available on the Internet for an adversary to use.

One may argue that some KBA challenge questions are more robust than others, a statement which is true. Response to a challenge like *“What is your childhood crush name?”* is harder to guess or to mine from searchable social data. Introduction of more such strong questions in the challenge set mitigates the risk of social engineering attacks, but they encroach into user’s personal space and are considered bad for user’s privacy.

Analysis of Human-Computer interaction has revealed that users tend to select the same set of challenge-response for different services. On the surface, this seems innocuous and obvious. However, this is a horrifying fact from the security angle. A secure system is as secure as its weakest link. Compromise of a challenge-response set of a user in a service which employs KBA for identity proofing may lead to the compromise of other KBA-based services if the user has selected the same challenge and response set.

Unlike static KBA, dynamic KBA does not depend on fixed challenges. In dynamic KBA the challenges are ever changing and are generated by mining public records and logs. Dynamic KBA is an alternative to static KBA and largely addresses issues involved with static KBA. However, there are distinct complexities associated in implementing a dynamic KBA scheme.

The first problem linked with dynamic KBA is difficulty of implementation. Compared to static KBA, dynamic KBA is harder to implement. There is no standard reusable model available for dynamic KBA which fits the need of all the organizations. Generally, dynamic KBA is designed and implemented for an agency taking into account their authentication needs and data available for mining and challenge creation.

Moreover, many countries have strict privacy laws regarding use of public data in the commercial space making it difficult to use public data for dynamic KBA.

Best practices in KBA

Following is a list of common threats and possible mitigations which must be considered when developing a KBA system. The list presented here is brief and does not delve into the technical specification.

Selection of Challenges

- Have a large set of challenges
- Select the challenges which are applicable to users accessing the system
- Secrets to challenges must be easy to remember and hard to guess
- Secrets must not change over time
- Give user the ability to select the challenges from the challenge set
- Additionally, provide the admin/user the capability to add his own challenges
- Categorize the challenges
- Graphical/audio challenges can be employed

Secret Management

- Set up validation for secrets (e.g. minimum length)
- Do not store secrets in plain text. It is advised to use tokens or one-way hash.
- Do not have default values for secrets
- Store secrets on distributed locations
- Employ mechanisms for disaster recovery and high availability

Automated Attacks

- Authenticate the user with different challenges (from the set of challenge-response) every time KBA is used
- Order the challenges randomly
- It is recommended to have a mix of static and dynamic challenges for a KBA scheme
- Challenge set should be formed from different categories
- Use out-of-band authentication

Impact on Privacy

- KBA challenges should not dig deep into the user's private-personal space
- Solution architects and developers must be trained not to interfere with user's privacy while developing KBA systems

Usability

- KBA solution must be designed to be simple and intuitive
- Authenticate within an acceptable time
- Make it easy for users to learn and use the system

Authentication failures

- Limit number of failed attempts
- Notify the user by email or SMS when authentication-using KBA is used as an alternative
- Add exponential delays – with a max cap – for every failed attempt
- Set up account lockdown (it may lead to denial of service attack)
- Auditing

Identity Theft

- Even if a KBA system is compromised a user's real identity must not be revealed
- Store secrets in token or one-way hash

Conclusion

Authentication is of utmost importance for securing an information system. The number of authentication methods available is overwhelming making it difficult to select an appropriate deployment method. Depending upon the overall risk and level of authentication needed, one or a combination of two or more authentication methods can be employed. Standalone KBA is one of the simplest authentication methods often used for identity proofing and account retrieval. Static KBA has been historically used for account recovery and low-level authentication.

The growth of social engineering attacks and easy availability of personally identifiable information (PII) over the Internet may render KBA systems obsolete. Additionally, concern over user privacy has also hit KBA systems hard. Nonetheless, while KBA systems may not be enough for the authentication needs of a modern computing scheme, it cannot be totally discarded.

KBA still holds great importance for secure-usable systems with low level of authentication. Solutions are available that increase the overall resilience of a KBA system. Some of them include combining static and dynamic challenges, using KBA along with out-of-band authentication [8], and employing graphical and audio challenges.

Taking into consideration the organizational needs and associated risks, a strong KBA system can be set up to successfully meet authentication requirements.

References

1. Federal Financial Institutions Examination Council (2008) Authentication in an Internet Banking Environment
2. Recommendation of National Institute of Standards and Technology(2006) Electronic Authentication Guideline
3. Federal Deposit Insurance Corporation(2004) Putting an End to Account-Hijacking Identity Theft
4. The Telegraph Sarah Palin vs the hacker
5. OWASP Choosing and Using Security Questions Cheat Sheet
6. OWASP Forgot Password Cheat Sheet
7. Santosh Chokhani Knowledge-Based Authentication (KBA) Metrics
8. Out-of-Band Authentication

EMC believes the information in this publication is accurate as of its publication date. The information is subject to change without notice.

THE INFORMATION IN THIS PUBLICATION IS PROVIDED "AS IS." EMC CORPORATION MAKES NO RESPRESENTATIONS OR WARRANTIES OF ANY KIND WITH RESPECT TO THE INFORMATION IN THIS PUBLICATION, AND SPECIFICALLY DISCLAIMS IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.