Precise Biometrics
whitepaper

Match-on-Card for Java Cards

Magnus Pettersson / Michael Harris, Precise Biometrics

9th November 2002

1. Introduction

Biometric verification has the advantage of ensuring that only the correct physical user can gain access to certain information or physical locations. The biometric identity can never be borrowed, and it is up to the security administrator of a system to decide who is to be granted or denied access.

The biometric process can be divided into two functions - *enrolment* and *verification*. For enrolment, unique features are extracted from the biometric data and stored in a *reference fingerprint template*. During verification, data is extracted from the live raw image data to be compared with the previously stored reference template.

As the reference template secure the user's digital identity, it is of high importance that this data be stored securely. From a user's perspective, it may also be of high importance that the personal data integrity is being safeguarded and the acceptance of having one's biometric data stored on a server might cause security trust problems. When deploying biometrics in an enterprise network, a server solution may also introduce limitations in terms of scalability.

The solution to these privacy, security and scalability challenges is to perform biometric verification inside the smart card, so that the storing and verifying of identity is accomplished directly on the card.

2. Match-on-Card

The ideal way to verify that a fingerprint presented to a fingerprint reader, actually matches the template stored from an earlier enrolment session, is to do the matching on the smart card, using the embedded smart card processor. By using Match-on-Card you gain three distinct advantages:

Privacy The template never leaves the smart card [4]

Security Two-factor authentication

Scalability Matching performed in the card – unlimited scalability

Open Standards Resulting in full adaptability & low cost readers

Integration into Public Key Infrastructure (PKI) PKI unlocked in card – all infrastructure unchanged

The software realization of Match-on-Card is Precise BioMatch™

3. Java Card Forum

The major smart card manufacturers started the Java Card Forum[1] (JCF) with the primary purpose to promote Java Cards as the preferred platform for multi- application smart card solutions. The primary activity for the JCF is the development and recommendation of a standardized specification to the existing Java Card API.

The Java Card Forum Biometry API defines the industry standard for Java Card implementations of Match-on-Card.

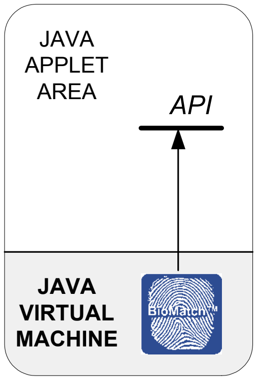
4. Precise BioMatch™ on ALL Java cards

Precise BioMatch™ runs on all types of Java cards, providing identical external interface characteristics and biometric matching functionality.

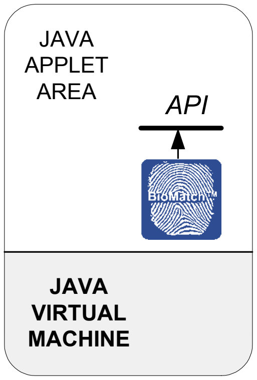
Major smart card vendors currently provide Precise BioMatch™ as an integrated function within their *Java Card Operating System*, thereby saving application memory space and improving performance in terms of biometric verification speed.

For Java cards from vendors that have not yet integrated Precise BioMatch™, a Java applet is available for easy downloading.

The biometric matching performance and interface are identical for all Java cards, regardless if the card has native support for Precise BioMatch™, or if the Precise BioMatch™ Java applet is used.



Precise BioMatch™ integrated by card vendor



Precise BioMatch™ downloaded as an applet

Precise BioMatch™ supports the Java Card Forum Biometry API and will run on all Java cards. Application code using the JCF-API version 2.2 biometry specification enables transparent integration of Java Applet or Java Card native OS implementations. This effectively eliminates vendor lock-in by providing a framework that accepts isolated applets or native OS code modules across varying Java Card smart card platforms.

5. Match-on-Card Characteristics

5.1. Match-on-Card vs. Match-on-PC Performance

The Match-on-Card technology is far more secure than matching on PC or server[2], as the fingerprint never leaves the secure environment of the smart card and no biometric data ever has to be transmitted over an open network.

The Match-on-Card procedure can be divided into two separate operations.

Pre-processing - the fingerprint image is enhanced and the reference areas to be matched are located. These operations require greater processing power, but do not utilize the reference template, allowing the operation to be safely done outside of the smart card.

Matching - This operation requires the reference template and must be performed in the secure environment of the smart card. Through the optimised design of the BioMatch algorithm, the matching can be done quickly, on the smart card chip without degrading matching accuracy or performance.

In terms of accuracy (FAR/FRR[1]) the matching performance is identical, whether processed on the smart card or in the PC. Similarly, matching speed is equivalent although the lower overhead (e.g. no network traffic) associated with Match-on-Card often yields faster overall results.

[1] FAR - False Acceptance Rate. FRR - False Rejection Rate.

5.2. Minimum memory usage

The Match-on-Card algorithm requires minimal code space. When implemented as a Java Card applet, the Match-on-Card process typically uses around 1200-bytes of on-card application memory (1-KB applet space, 200-Bytes runtime processing). Conversely, when the Match-on-Card algorithm is implemented natively in the card ROM, application memory is not required for the algorithm and only negligible (approx. 200 bytes) memory is used for runtime processing. In either case, Java Applet or Native ROM implementation, an additional 500-bytes of memory space is necessary for each template saved to the card.

The small size of the algorithm storage and dynamic runtime allocation along with the typical 500-btye template size, enables the flexibility for most smart cards to support multiple biometrics. This valuable feature offers the implementation capability for higher-security (multiple-matching) or alternate finger matching. Alternate finger matching can be useful for damaged fingers or when the implementation requires tying fingers to specific applications.

5.3. Scalability without limitations

In comparison to a server based system, there is no limitation in the number of possible users when utilizing Match-on-Card. All fingerprint verification is performed locally on the smart card without any need for network resources or server processing. Match-on-Card, in effect, creates a highly scaleable, distributed, and transportable database with each biometric asset maintained in it's own secure smart card environment.

5.4. Privacy and Integrity

With Match-On-Card, the fingerprint template is stored within the card, unavailable to external applications and the outside world. In addition, the matching decision is securely authenticated internally by the smart card itself.

Match-On-Card must not be confused with merely storing the biometric template on a smart card and performing the match decision outside the card on a server or a client PC. Such a solution does not add any security or controlled access to the information stored on the card.

Precise Match-on-Card enables PKI identification within the network for increased security while maintaining the biometric privacy and integrity of the end-user.

5.5. Compatible with all on-card applications

The Precise Match-on-Card for Java Cards does not interfere with other card applications. A Match-on-Card applet will work side-by-side with any other smart card application such as loyalty, banking or identification programs.

Additionally, other applications on the card, have the capability to take advantage of the on card biometrics through the sharable interfaces defined by the Java Card Forum - thereby easily adding biometric security to their independent functionality.

5.6. Smart Card and Fingerprint reader independence

The Precise Match-on-Card process does not require any special capabilities of the biometric or Smart Card reader. A combination reader (Fingerprint + Smart Card) might be used as well as different heterogeneous brands of independent fingerprint readers and smart card readers. For example, a biometric-only reader can be used along with a separate generic PC/SC smart card reader or a combination Fingerprint and smart card reader (e.g. SCM) can be utilized.

Precise Biometrics provides this flexibility by staunchly supporting and adhering to the diverse smart card and biometric standards. Additionally, Precise Biometrics offers Biometric Service Providers (BSP) solutions that can work transparently with other certified biometric vendor's products.

6. Conclusion

The Precise Match-on-Card for Java cards has several important advantages:

Match-on-Card
Ensures highest security and privacy
Minimizes infrastructure investments
Provides off-line operation
Compatible with all on-card applications
Promotes integrity and scalability

Precise BioMatch™
Compatible with installed card base
Small code and template sizes
High speed in combination with highest accuracy
Integration with Card Operating System (COS) Security Management
Open sharable interfaces on card
Smart Card and biometric reader independent

Standards Compliancy
ISO/IEC 7816-11
ISO/IEC 14443-A
NISTIR 6529 (CBEFF)
NISTIR 6887 (GSA Interop. Spec.)
FIPS 140-2
Java Card Forum v2.2
Open Card Framework
BioAPI enrolment

7. References

7.1. WWW

[1] Java Card Forum, JCF - http://www.javacardforum.org/

7.2. Precise Biometrics whitepapers

[2] How secure is your biometric solution - http://www.precisebiometrics.com

[3] Precise BioMatch™ - http://www.precisebiometrics.com

[4] Ensuring integrity with fingerprint verification http://www.precisebiometrics.com

[5] The Match-on-Card technology - http://www.precisebiometrics.com