A Survey of Template Protection Schemes and What One Might do With Them

Walter Scheirer & Terrance Boult

"Biometrics: Practical Issues in Privacy and Security"

IJCB 2011

Security Basics

Template Protection as a Solution

- Protect the Privacy and Security of the Biometric Features
- Revoke and re-issue biometric templates like a password or credit card #
- Match in an encoded space
- Prevent linking across databases (solve the biometric dilemma)
- Prevent the doppelganger attack (multi-factors)

"Getting this right has been much more challenging than we first thought." – Fabian Monrose

Lots of stuff out there!

- Biometric Encryption
- Non-invertible Transforms
- BioHashing
- Robust Hashing
- Fuzzy Vaults
- Fuzzy Commitment
- Fuzzy Extractors
- Revocable Biotokens
- Hybrid Combinations

How do they work?

How well do they work?

How secure are they?

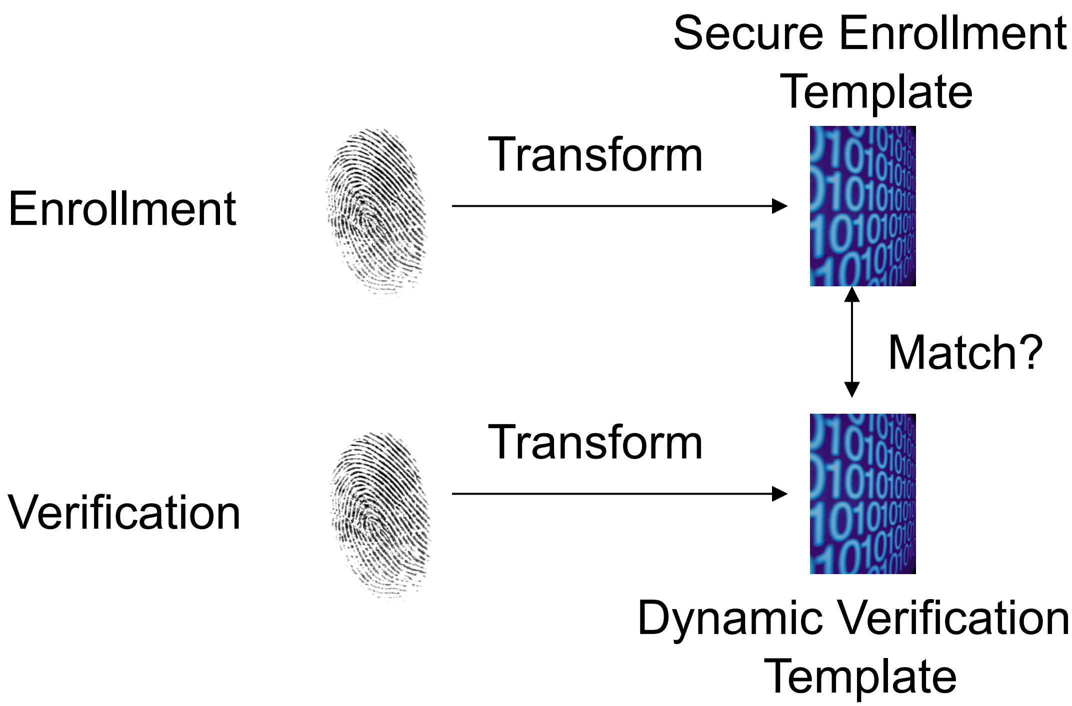
General Categories*

- Straight feature protection
- Key-generating
- Key-binding

*A. Jain, K. Nandakumar and A. Nagar, "Biometric Template Security", in EURASIP Journal on Advances in Signal Processing, Special Issue on Biometrics, 2008

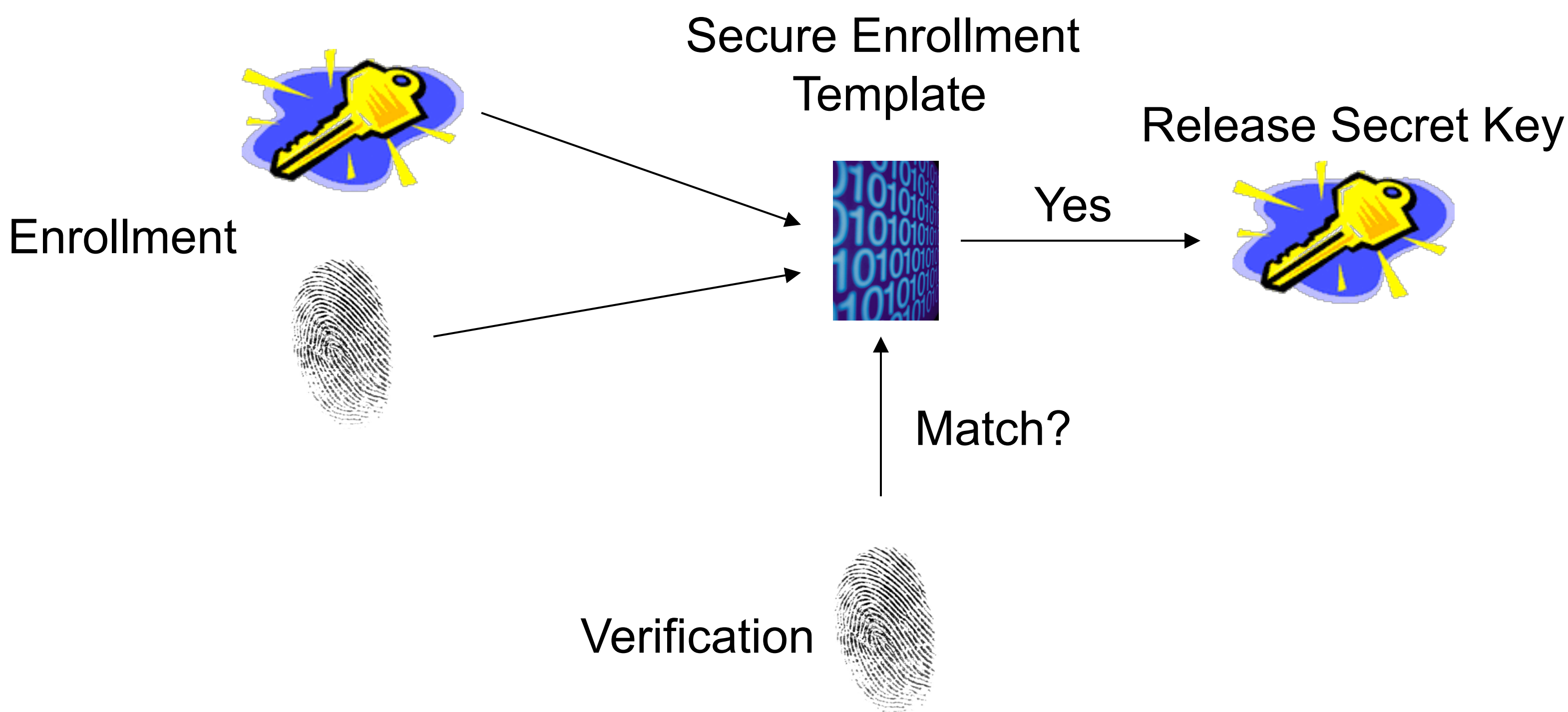
Straight Feature Protection

- Simply protect the original biometric features using some transformation that allows matching in encoded space



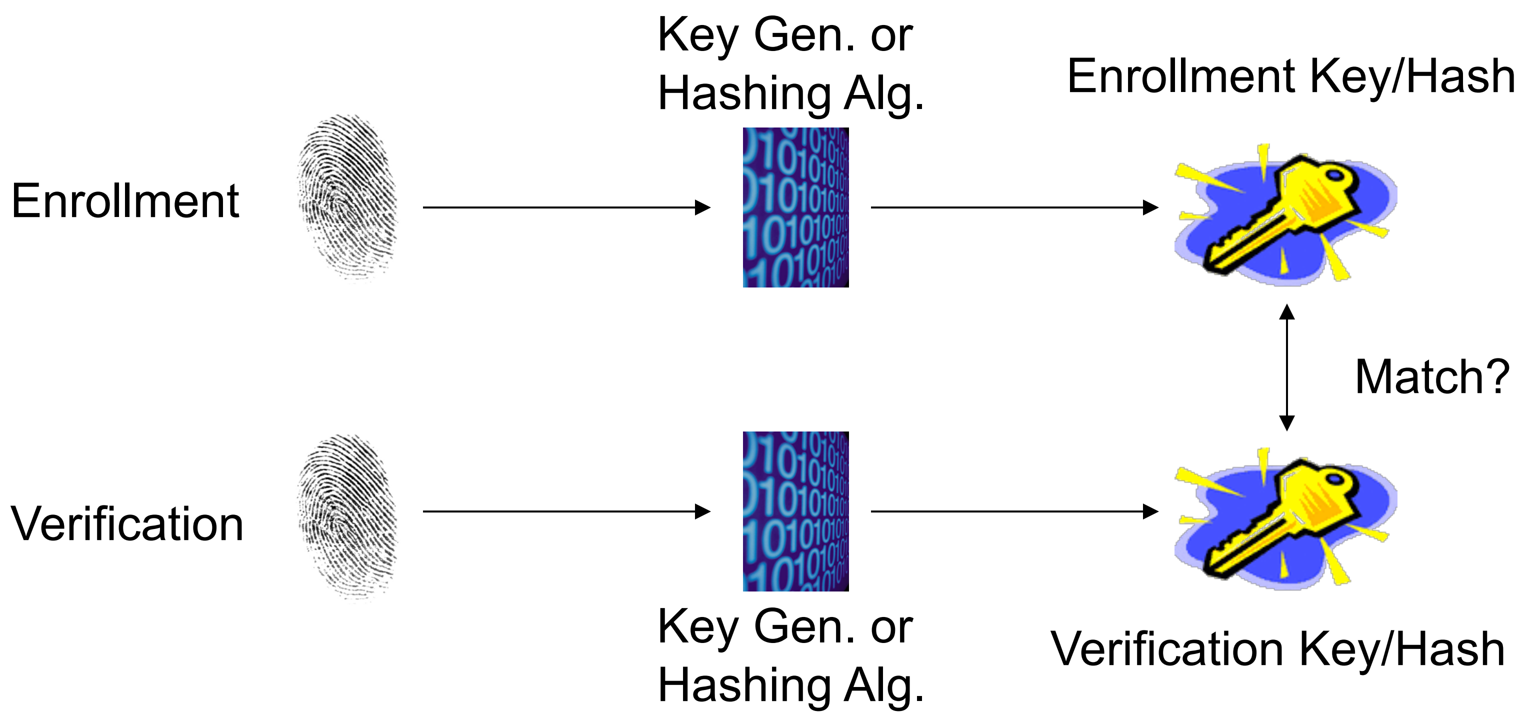
Key-binding

- Biometric cryptosystem that binds key data with the biometric data



Key-generating

- Biometric cryptosystem that derives a key from the biometric data



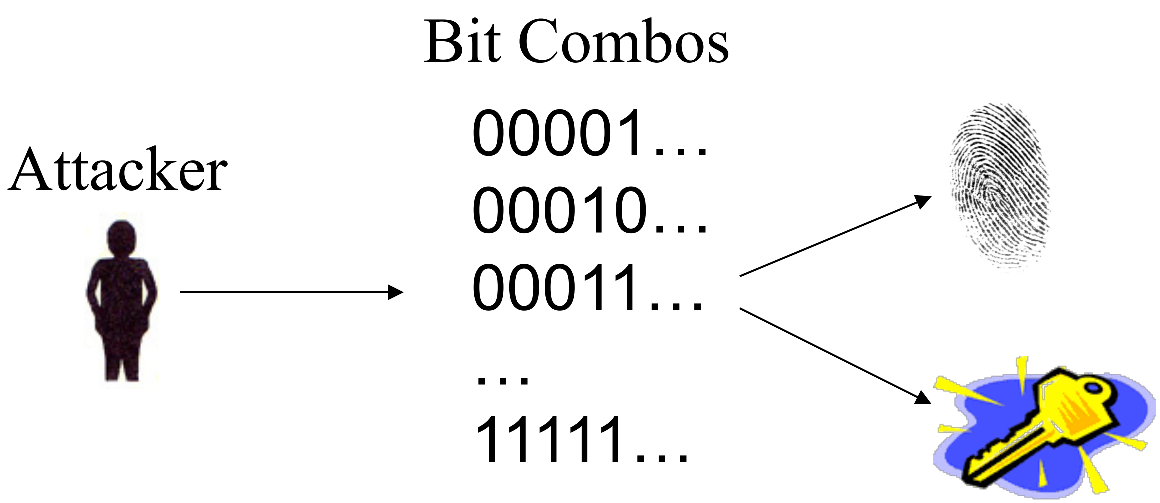
Attacks Against Secure Template Protection Technologies

- Basic Brute Force
- Correlation Attack*
- Known Key Attack*
- Substitution Attacks*
- Decodability Attack
- Doppelganger Attack
- Hill Climbing

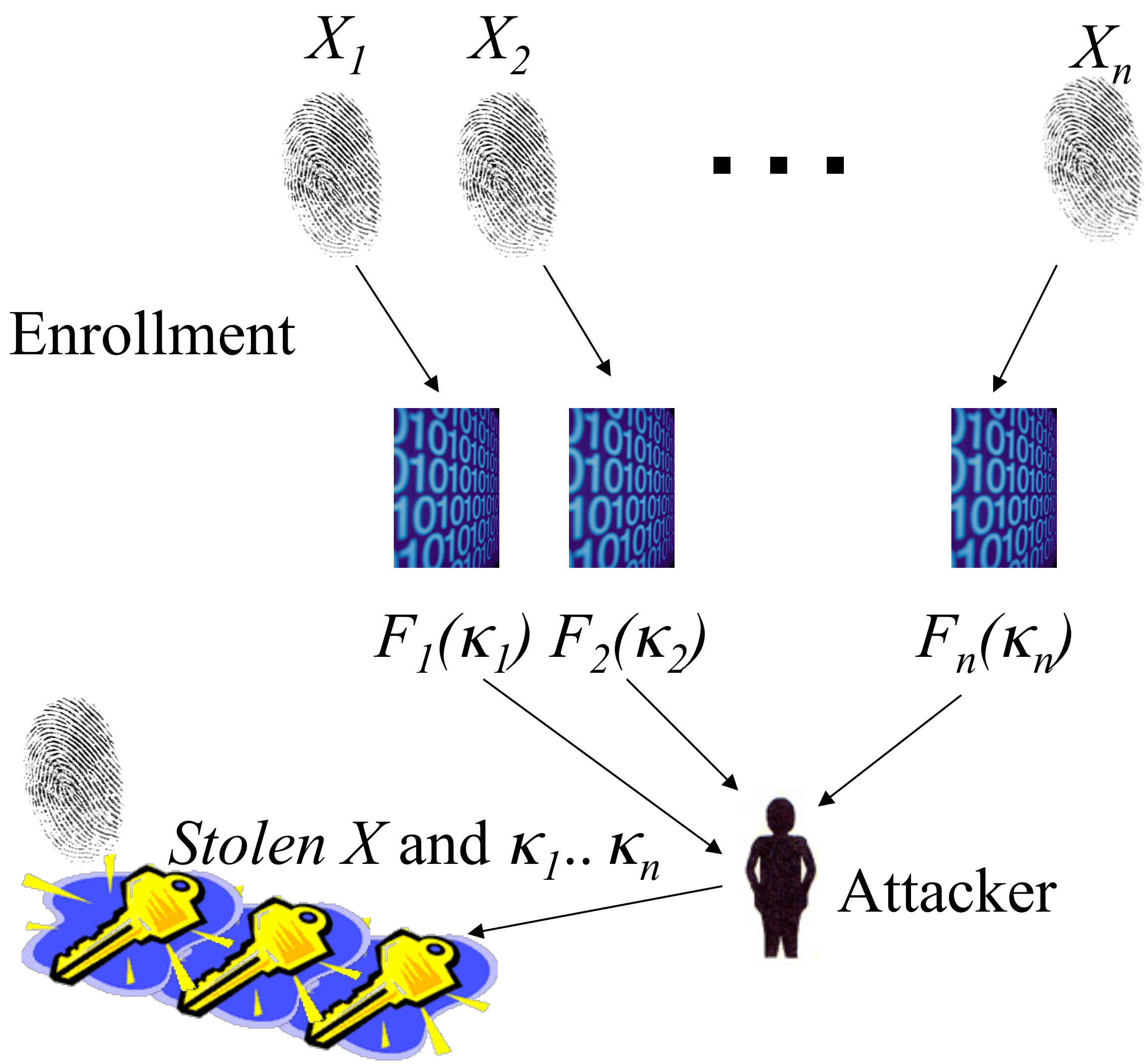
*W. Scheirer and T. Boult, "Cracking Fuzzy Vaults and Biometric Encryption," in Proc. of the 2007 Biometrics Symposium

Basic Brute Force

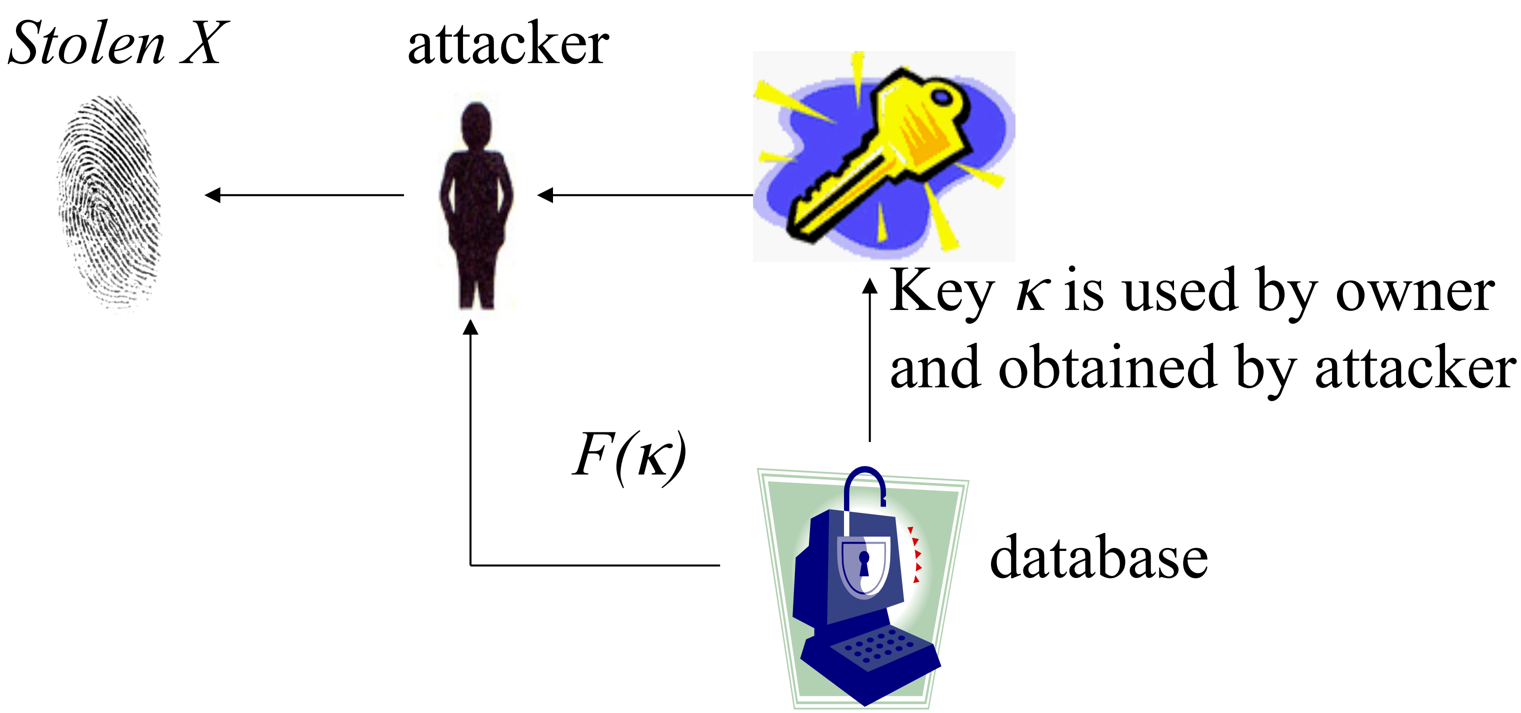
- Attacker tries every possible bit combination till they guess the correct original feature data or key
  - Need a way to test each bit combo



Correlation Attack

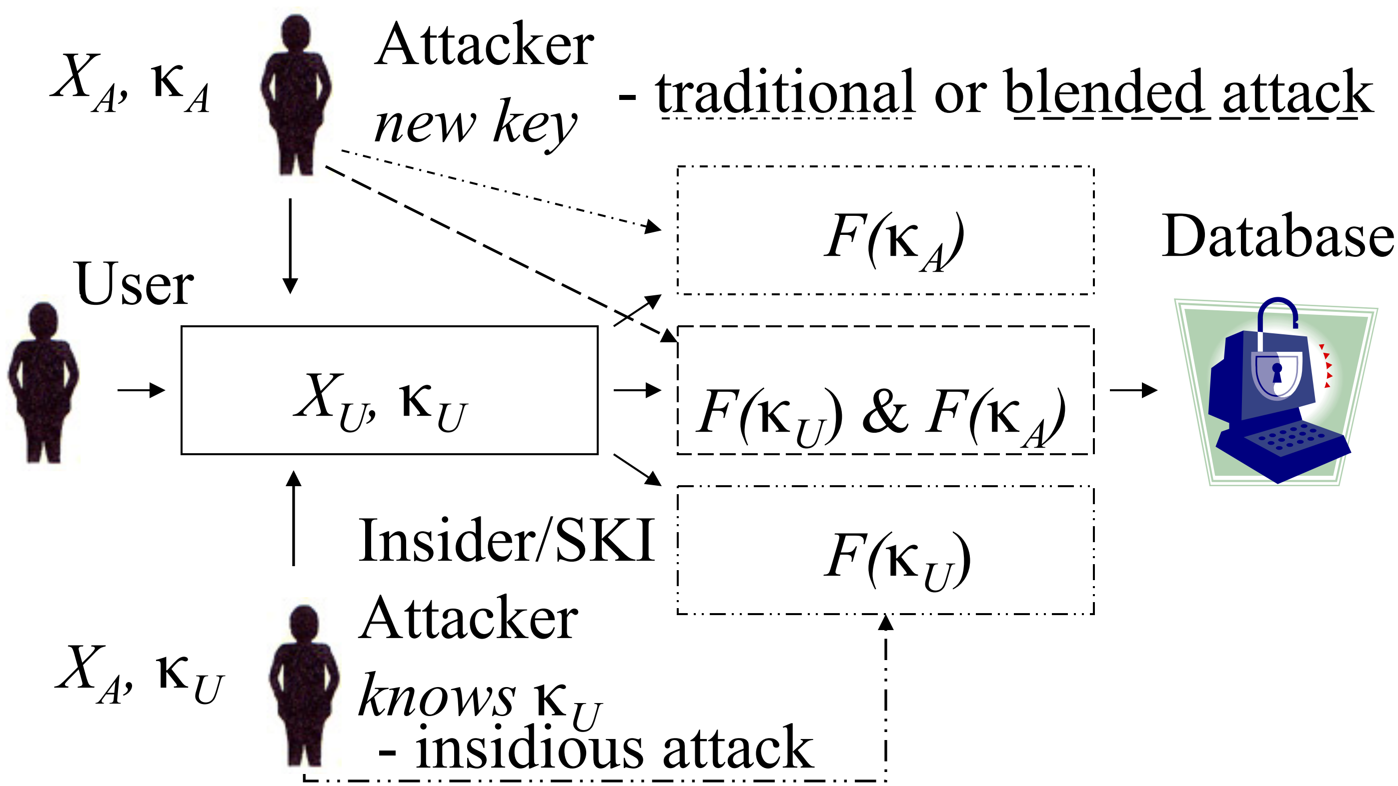


Known Key Attack



Substitution Attacks

"How difficult will it be to break into a folder containing biometric signatures and replace them with an attacker's biometric signature so that the attacker can get in with his/her own signature easily?*"



*Avinash Kadam, MIEL e-Security, "The Memory Game," Information Week, July 29th, 2011

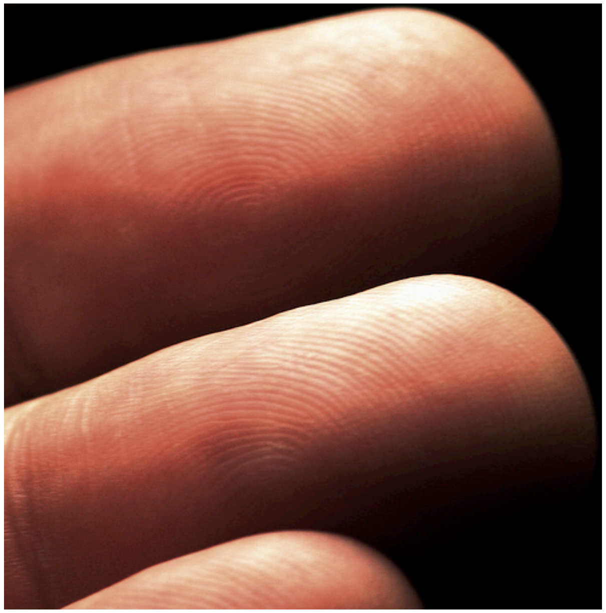
Decodability Attack

- Exploit available information to link across databases*
- Assume a template W contains helper data H and biometric data X
  - $W_1 = H_1 \oplus X_1;\ W_2 = H_2 \oplus X_2$
- If $W_1 \oplus W_2$ is decodable, the two templates are probably derived from the same person

*F. Carter and A. Stoianov, "Implications of Biometric Encryption on Wide Spread Use of Biometrics," EBF Biometric Encryption Seminar, June 2008.

The Doppelganger Threat

- If the FAR is 1 in *X*, then an attacker can try more than *X* different prints
- Lots of public data available!
  - Fingerprint: NIST DB 14, NIST DB 29, FVC 2002, FVC 2004 …
  - Face: MBGC, FRGC, FVT, FERET …
  - Think of this as a biometric dictionary attack



Information Theoretical Security Analysis vs. Practical Matching Security

- A disconnect exists between information theoretical security models and matching accuracy
  - Both are important!
- Information leakage is bounded by matching accuracy
  - If a false match to a template releases the correct key, the system leaks 100% of the key information
    - ECC often overcorrects, which drives up the FAR

Hill Climbing

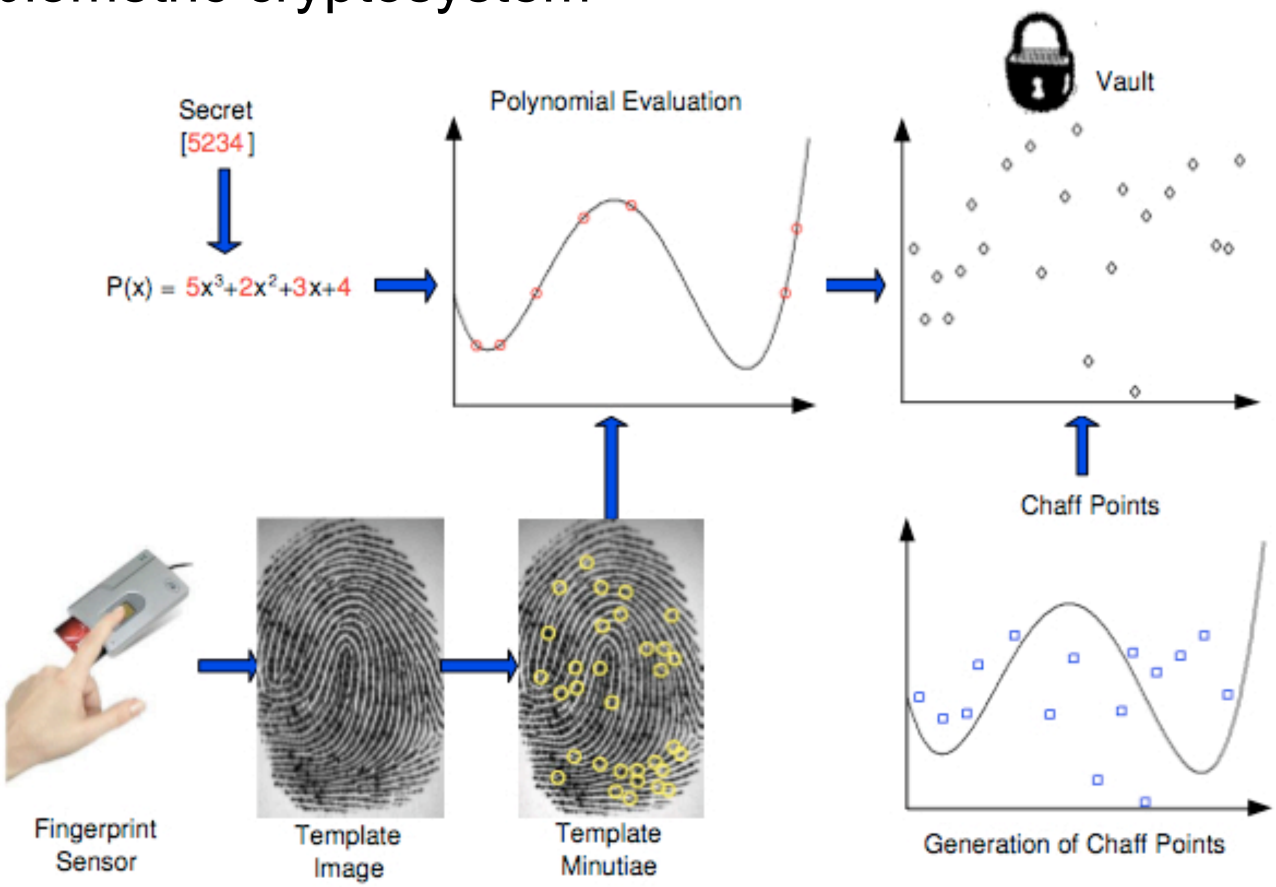
- Requires less than brute-force effort to recover an embedded secret
- Provides an estimate of the enrollment image

In an iterative fashion, modifications are made to the input, and those that increase the match score are retained.

Prevalent Template Protection Schemes

Fuzzy Vaults

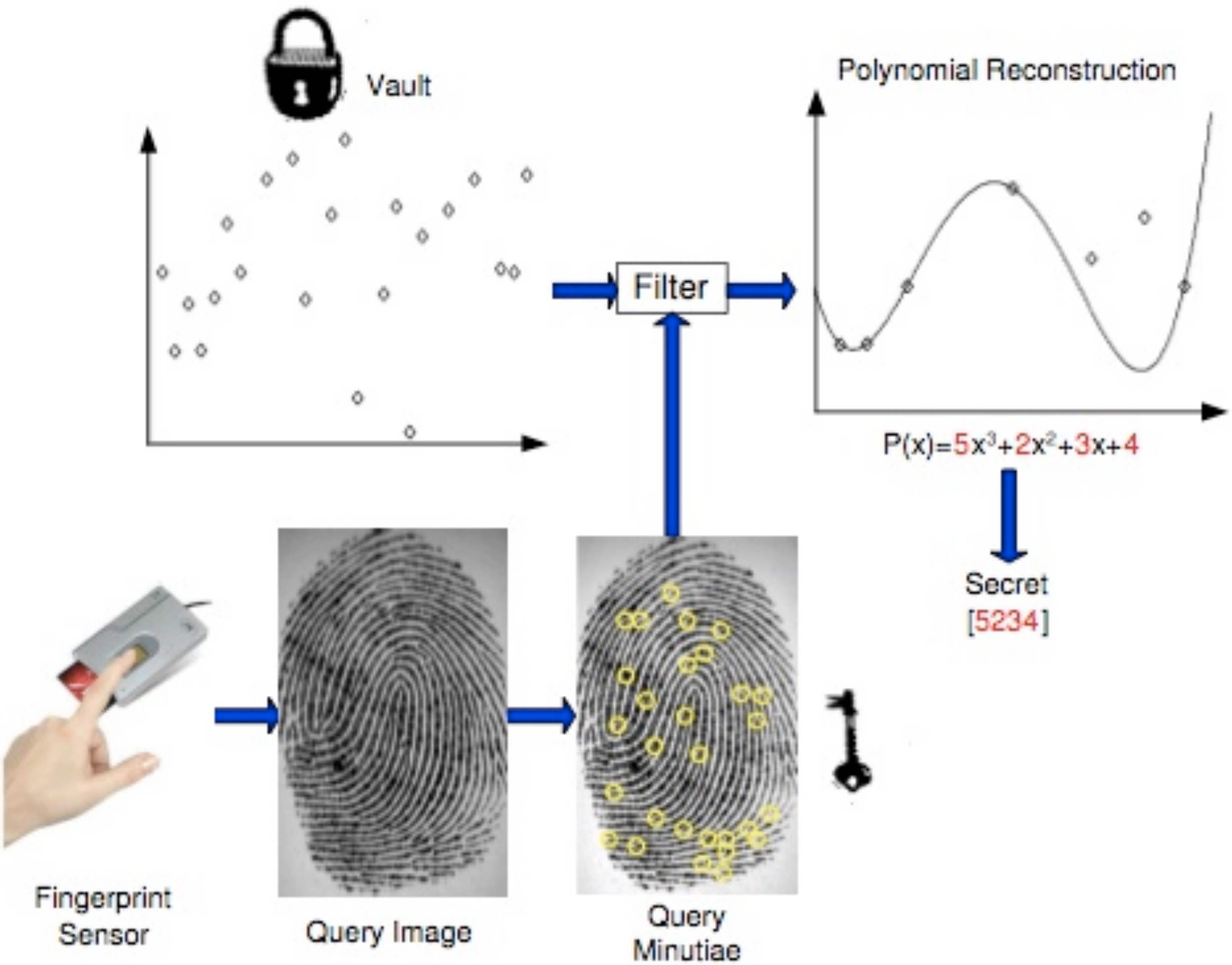
- Not specific to biometric data, but typically applied to minutiae based fingerprint matchers as a key binding biometric cryptosystem



Encoding

* A. Juels and M. Sudan, “A Fuzzy Vault Scheme,” IEEE International Symposium on Information Theory, 2002.

Fuzzy Vaults



Decoding

Performance Numbers

	112 Bits		128 Bits		160 Bits	
	GAR	FAR	GAR	FAR	GAR	FAR
F.P. Fuzzy Vaults[1]	89	0.13	89	0.01	84	0
F.P. FV, Mosaic with 2 Queries[1]	96	0.24	95	0.04	89	0
Password Vault[2]	88	?	86	?	79	?

1. K. Nandakumar, A. K. Jain and S. Pankanti, "Fingerprint-based Fuzzy Vault: Implementation and Performance", In IEEE TIFS, vol. 2, no. 4, 2007

2. K. Nandakumar, A. Nagar and A. K. Jain, "Hardening Fingerprint Fuzzy Vault Using Password", in Proc. of ICB 2007

Fuzzy Vaults: Security Problems

- Chaff Point Identification[1]
- Improved Brute Force Attack[2]
- Correlation Attack, Known Key Attack, Correlation Attacks
- Hill Climbing
  - May be theoretically possible
    - Security proof assumes data held in the vault is random; not the case with biometrics
    - Chaff is placed carefully so as not to conflict with legitimate points; strays from randomness assumption

1. W. Chang, R. Shen, and F. W. Teo, "Finding the Original Point Set Hidden Among Chaff," in Proc. of the ACM Symposium on Information, Computer And Communications Security, 2006.

2. P. Mihailescu, "The Fuzzy Vault for Fingerprints is Vulnerable to Brute Force Attack," 2007.

Fuzzy Vaults: Correlation Attack

- Without a matching sample, the polynomial reconstruction problem is infeasible to solve
- What if we have *two or more* BFV instances?
  - Take the intersection of the abscissa (x) values for the BFV instances
  - The result is the original template data
  - Some chaff points are likely to match - but the error correcting code is designed for this possibility

Fuzzy Vaults: Known Key Attack

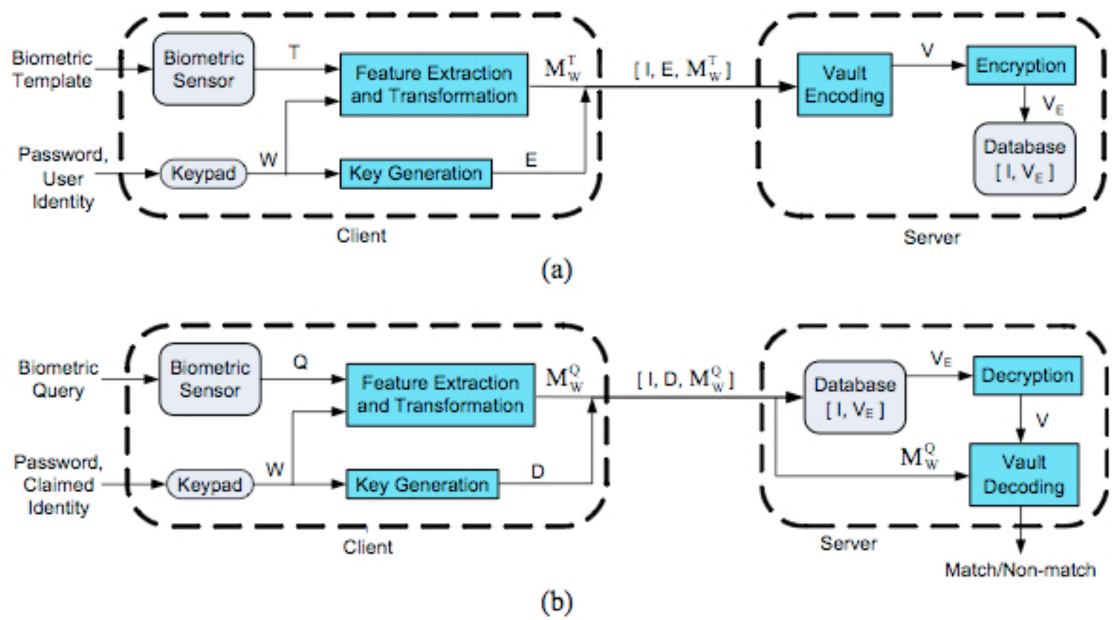
- From κ, the polynomial p is directly reconstructed
- R may be directly enumerated to separate the template data, in the form $(A, p(A))$, from the chaff
- Again, the error correcting code will help us if some chaff matches

Fuzzy Vaults: Substitution Attacks

- Most of the vault is chaff. Matching uses only a small fraction of real data hidden in it.
- Overwrite chaff lines with attacker's template data, encoding X_A and κ_A
- Resulting template has both the user's and attacker's data.
- Insidious attack - attacker encodes their data with the user's key κ_U

Response To Vulnerabilities in Fuzzy Vaults

- Password Hardened Fuzzy Vault*



Enrollment

Authentication

*Karthik Nandakumar, Abhishek Nagar and Anil K. Jain, “Hardening Fuzzy Vault Using Pasword”, in Proc. of ICB 2007 (and image credit)

Securics® © 2011

Securics®: The science of security™

Response to Vulnerabilities in Fuzzy Vaults

- Fuzzy Commitment to "encrypt" polynomial evalutions[1]
- Carefully chosen chaff[2]
- Incorporate local ridge information of minutiae (also incorporates a password)[3]
- Distance preserving hash functions[4]

1. A. Nagar et al. "Securing Fingerprint Template: Fuzzy Vault with Minutiae Descriptors," ICPR 2008

2. S. Lee et al. "Secure Fuzzy Fingerprint Vault Against Correlation Attack," IEICE Electronics Express, Vol. 6, No. 18, 2009.

3. P. Li et al. "Security-Enhanced Fuzzy Fingerprint Vault Based on Minutiae's Local Ridge Information," ICB, 2009.

4. C. Orencik et al. "Securing Fuzzy Vault Schemes Through Biometric Hashing," Turk. J. Elec. Eng. & Comp. Sci., Vol. 18, No. 4, 2010.

UCCS

Fuzzy Commitment

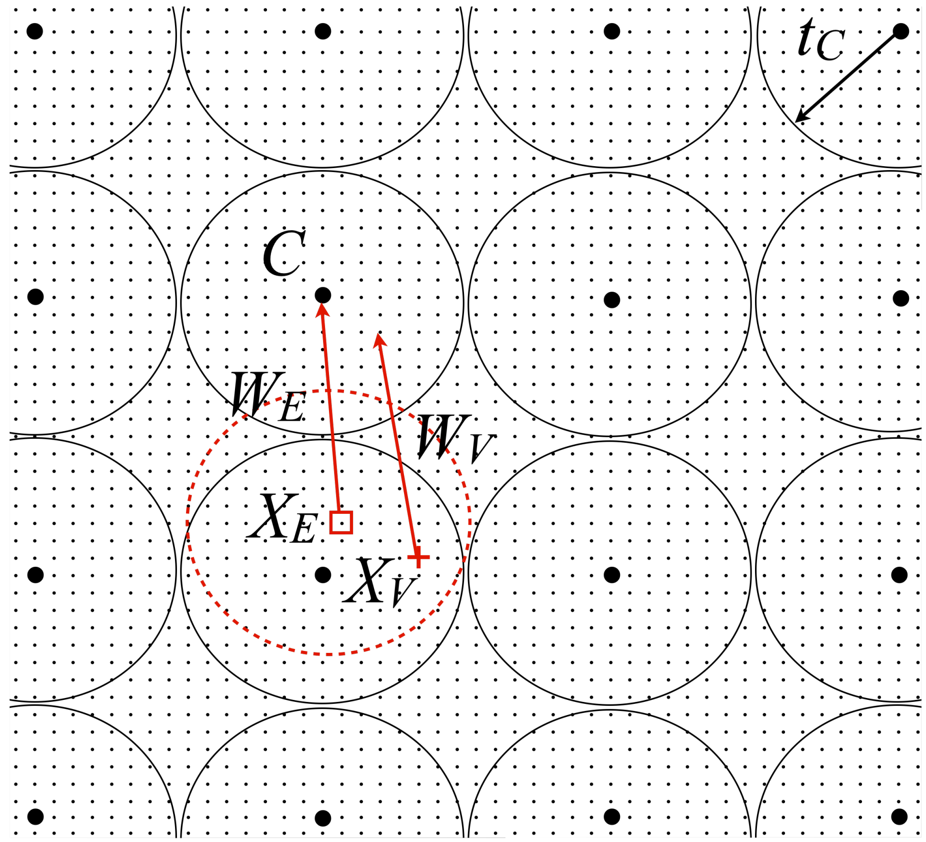
- Another well known key binding approach*
- Enrollment
  - Commit a codeword C (acts as the key) of an error correcting code using a fixed length biometric feature vector X as a witness
  - Store a hash h of C as "helper data"
  - Fuzzy Commitment: $X \oplus C,\ h(C)$

*A. Juels and M. Wattenberg, "A Fuzzy Commitment Scheme," 6th ACM Conf. on Computer and Communication Security, 1999.

Fuzzy Commitment

- Verification
  - User presents a biometric, producing feature vector X'
  - X' is then used to unlock the codeword
    - $(X \oplus C) \oplus X' = C' = C \oplus e$
    - Hamming distance d_H indicates the number of errors corrupting C
      - $\epsilon = d_H(X, X') = ||e||$
    - An ECC Decoder can correct errors, yielding an extracted candidate key K
    - A successful match occurs when $h(K) = h(C)$

Illustration of Fuzzy Commitment

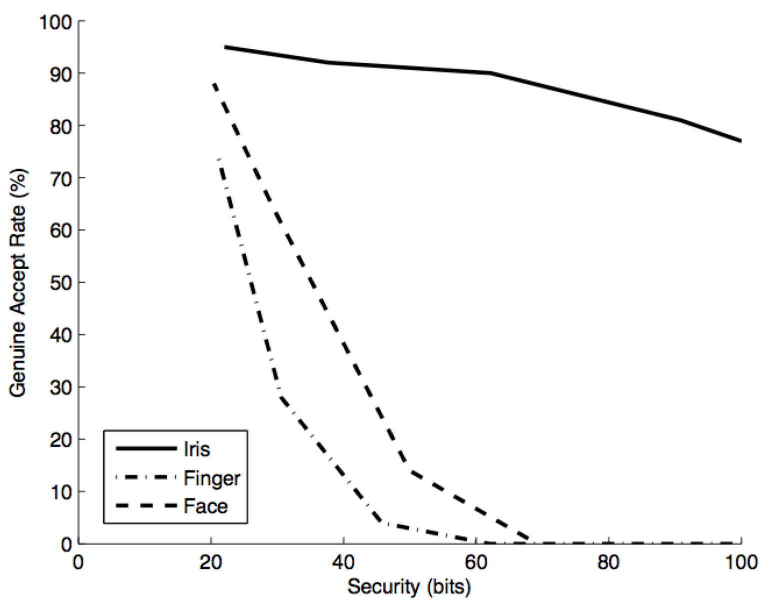


Grid of small dots: word space $\{0,1\}^{n_c}$

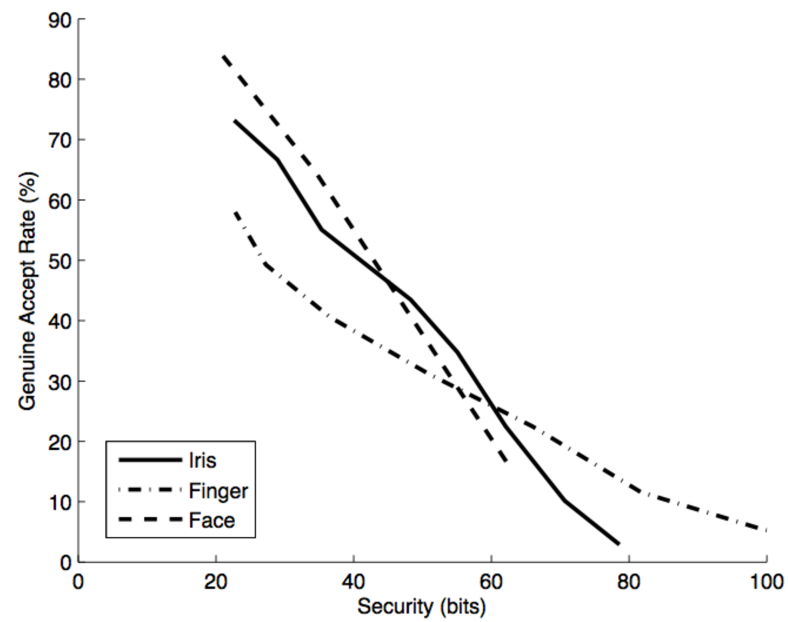
Bigger dots: codewords from C with the error correcting capability of the circles with radius t_c

Image adapted from: Kelkboom et al. "Preventing the Decodability Attack Based Cross-Matching in a Fuzzy Commitment Scheme," T-IFS, March 2011.

Performance Numbers



CASIA Ver-1, FVC 2002 DB2, XM2VTS



WVU Multimodal

	FVC/CASIA/XM2VTS	WVU
Iris	37%	91%
Face	30%	2%
Finger	33%	12%

Comparison of GAR at 53 bits of security

Images and Results: A. Nagar, "Biometric Template Security," Thesis Proposal, Michigan State University, 2001.

Performance Numbers

- 3-layer coding scheme[1]: ERR of 6.5% for 1032 bit key on FVC2000 DB2
- Multibiometric Fusion[2]:

	FVC/CASIA/XM2VTS	WVU
AND Rule	27%	89%
"Multibiometric Cryptosystem"	75%	99%

Comparison of GAR at 53 bits of security

- Bringer et al. 2008[3] for 2028 bit keys:
  - ICE: FRR 5.62%, FAR < 10^{-5}
  - CASIA: FRR 6.65%, FAR 0%
  - FVC 2000: FRR 2.73%, FAR 5.53%

1. X. Shao et al., "A 3-layer Coding Scheme for Biometry Template Protection Based on Spectral Minutiae", ICASSP, 2011.

2. A Nagar et al., "Technical Report: Multibiometric Cryptosystem", MSU Tech. Report, 2011.

3. J. Bringer et al., "Theoretical and Practical Boundaries of Binary Secure Sketches", IEEE T-IFS, 2011.

Fuzzy Commitment: Security Problem

- Decodability Attack*
  - Codewords: C_1, C_2
  - Biometric Data: X_1, X_2
  - $W_1 = C_1 \oplus X_1; W_2 = C_2 \oplus X_2$
  - $W_1 \oplus W_2 = (C_1 \oplus C_2) \oplus (X_1 \oplus X_2) = C_3 \oplus (X_1 \oplus X_2)$
  - If $(X_1 \oplus X_2)$ is small, the result of the XOR will be close to another codeword (decodes)

*F. Carter and A. Stoianov, “Implications of Biometric Encryption on Wide Spread Use of Biometrics,” EBF Biometric Encryption Seminar, June 2008.

Response to Vulnerabilities in Fuzzy Commitment*

- Incorporate random bit permutation process
- Prior to the XOR operation of the biometric data X with the code word C, randomize X with a bit permutation matrix M_r
- The new template: $W = C \oplus M_r X$
- M_r is not considered a secret

*Kelkboom et al. “Preventing the Decodability Attack Based Cross-Matching in a Fuzzy Commitment Scheme,” T-IFS, March 2011.

Fuzzy Extractors

- Key generating biometric cryptosystem*
- Attractive proposition, but difficult due to intra-user variability
- Goal: Extract a uniformly random string R from its input w in a noise-tolerant way
  - If the input changes to some w', but remains close, the string R can still be reproduced exactly

*Dodis et al., "Fuzzy Extractors: How to Generate Strong Keys from Biometrics and Other Noisy Data," EUROPCRYPT, 2004.

Secure Sketch*

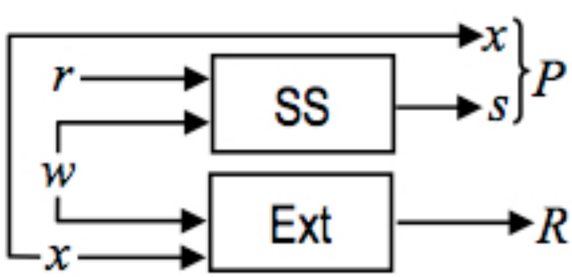
- “Helper Data” for Fuzzy Extractors
- A *secure sketch* produces public information about its input w that does not reveal w, and yet allows exact recovery of w given another value that is close to w.

*Y. Dodis, L. Reyzin and A. Smith, Fuzzy Extractors,” In Security with Noisy Data: Private Biometrics, Secure Key Storage and Anti-Counterfeiting, P. Tuyls, B. Skoric and T. Kevenaar, Eds., Springer-Verlag, 2007.

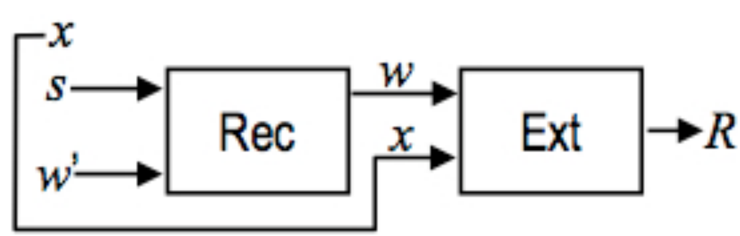
Fuzzy Extractors

- A secure sketch SS producing a string s bound with a random number x forms the basis of the helper string *P*
- Recovery procedure allows matching with a “close” string w'
- Extractor returns a string R, *the key*, when approximate input matching is successful
- P assists in the reproduction of R

Sketching Procedure



Recovery Procedure



r is some randomness

Security Analysis: Fuzzy Extractors

- Security analysis of the fuzzy extractor scheme made in terms of the *min-entropy*
- An adversary's best strategy is to guess the most likely value
  - Predictability of a random variable
  - Min-entropy is the "worst case" entropy
- Information theoretical balance between stability and suitable randomness

*Analysis is not made with consideration to FAR/GAR!

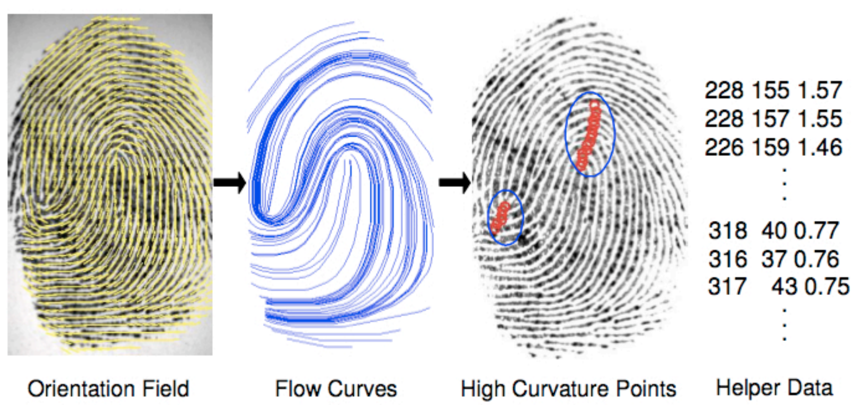
Practical Concerns

- At the present, fuzzy extractors exist in the realm of theory
- Fuzzy extractors may suffer from practical constraints during error-prone data collection; difficulty for key generation*
  - Unclear whether known constructions can correct the errors typically generated by humans
  - Require biometric inputs with high min-entropy, but haven't discussed feature selection

*Ballard, S. Kamara and M. Reiter, "The Practical Subtleties of Biometric Key Generation", in Proc. of the USENIX Security Symposium, 2008.

What's so difficult about all of these "fuzzy" techniques?

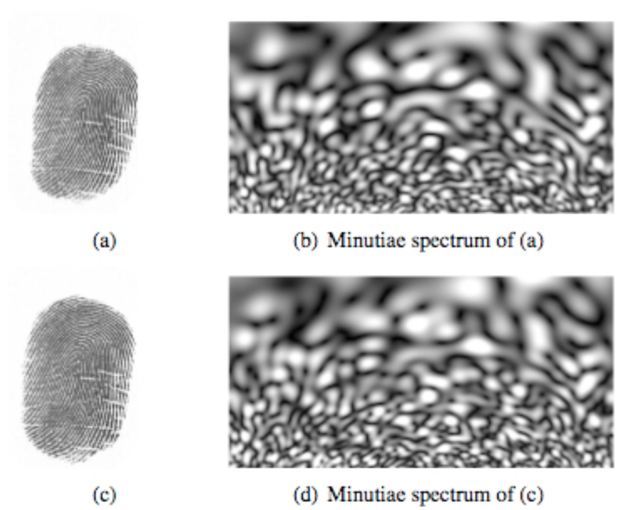
- In essence, if biometric features are not aligned properly, these schemes fail to work
- Solution* for fingerprint fuzzy vaults: helper data
  – accurately aligns the template and query minutiae, but does not reveal any information about the minutiae points - larger templates



*K. Nandakumar and A. Jain, "Fingerprint-based Fuzzy Vault: Implementation and Performance", IEEE TIFS, Dec. 2007 (and image credit)

What's so difficult about all of these "fuzzy" techniques?

- Fuzzy Commitment requires a fixed length feature vector representation of a biometric modality
  - Minutiae-based representation will not work



One approach*: Fourier-Mellin Transform; invariant to translation, scaling and rotations become translations

*H. Xu, R. Veldhuis, T. Kevenaar, A. Akkermans and A. Bazen, "Spectral Minutiae: A Fixed-length Representation of a Minutiae Set", in Proc. of the IEEE Computer Society Workshop in Biometrics, 2008.

Revocable Biotokens

- We want two different things:
  - Robust distance/matching
  - Security/Revocability

→Break data into two parts:

Stable and *Unstable*

5ft (stable)
2in unstable

6ft (stable)
1in unstable

- Stable part is encrypted/hashed to provide security/privacy and revocability - straight feature protection
- Two parts together provide robust distance measure, which we can prove will not decrease accuracy

Revocable Biotokens*

- Assume a biometric produces a value v that is transformed via scaling and translation
  - $v' = (v - t) * s$
- Split v' into stable component q and residual component r
- For user j, leave the residual un-encoded (base scheme)
  - $r_j(v')$
- Encrypt q with public key P
  - $w_{j,1}(v', P)$

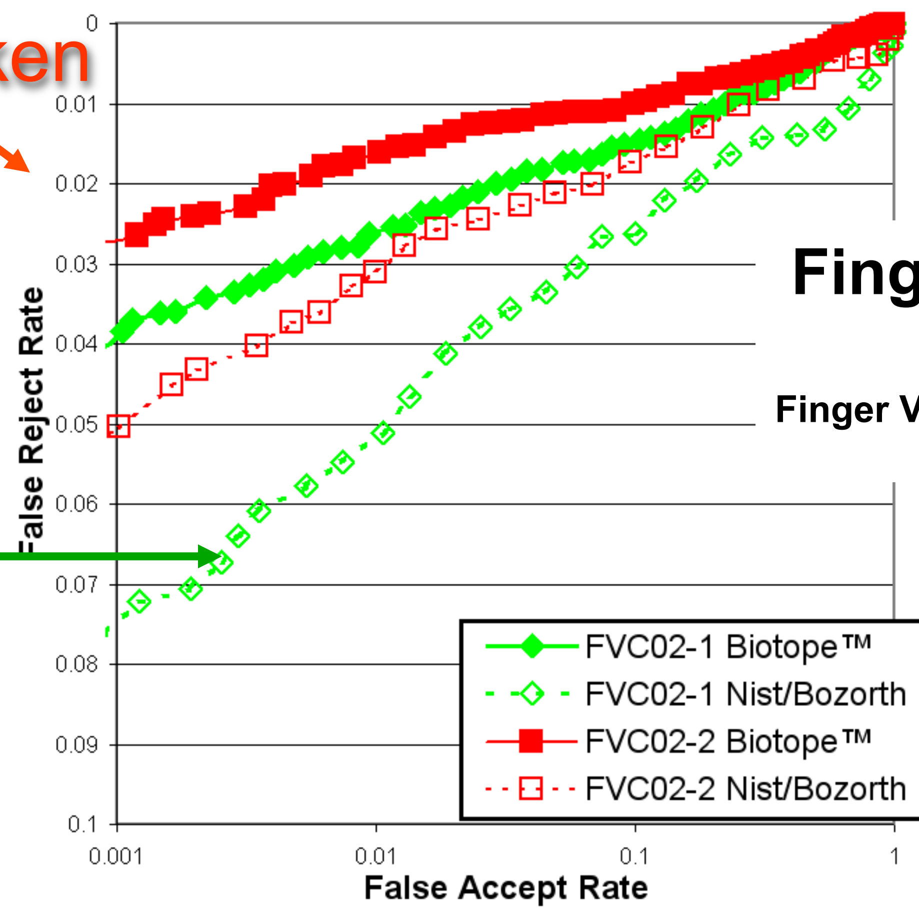
Brute Force Attack to revert biotoken back to original features: 2^{108} for insider, 2^{120} without access to all keys/data

*T. Boult, W. Scheirer and R. Woodworth, "Revocable Fingerprint Biotokens: Accuracy and Security Analysis," CVPR 2007.

Revocable Biotoken Performance*

Finger Biotoken

Base "NIST" Bozorth



Finger Biotoken on

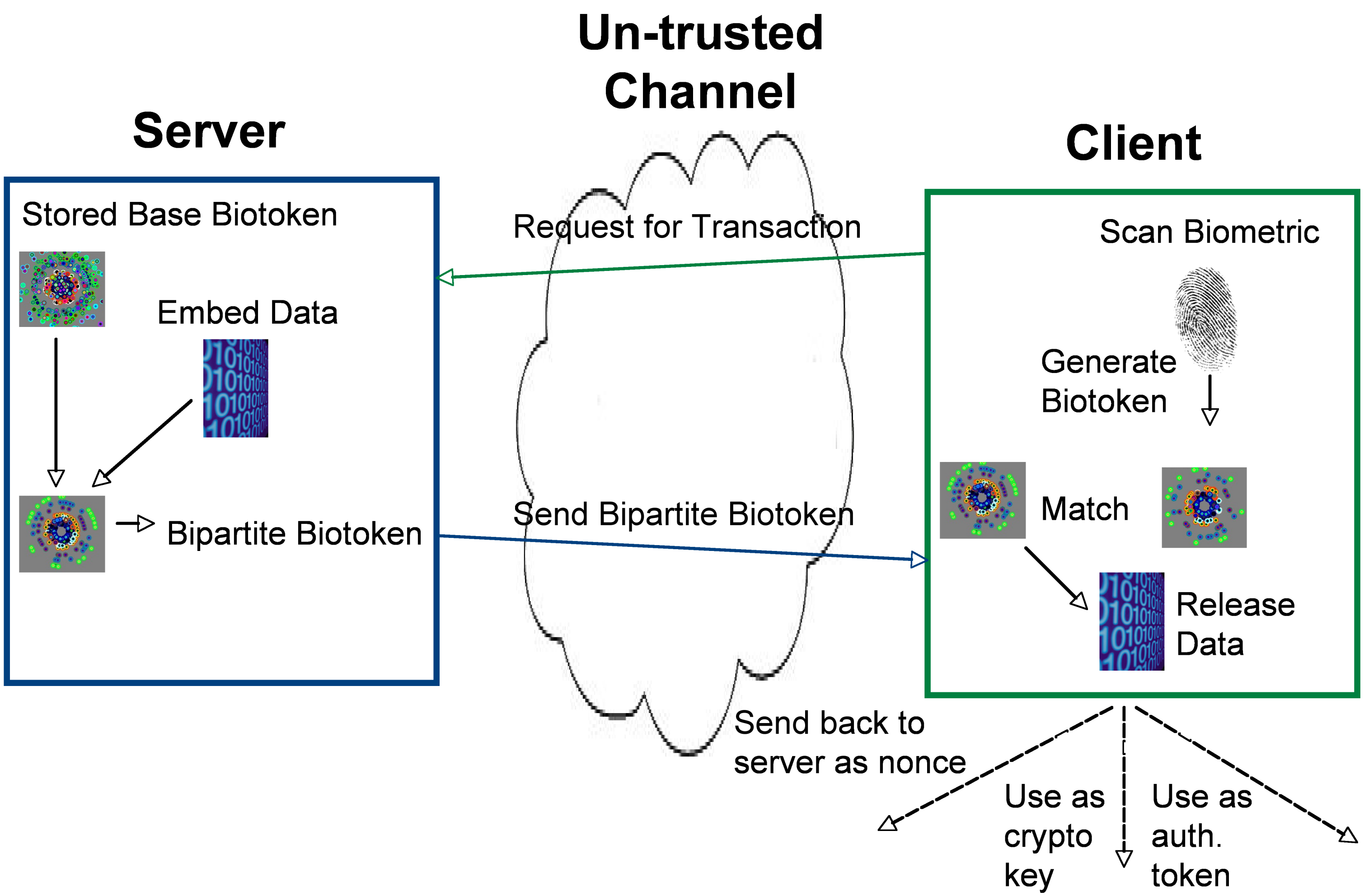
Finger Verification Challenge datasets

*T. Boult, W. Scheirer and R. Woodworth, "Secure Revocable Finger Biotokens." In Proc. of IEEE CVPR 2007, Minneapolis, MN

Nesting Property

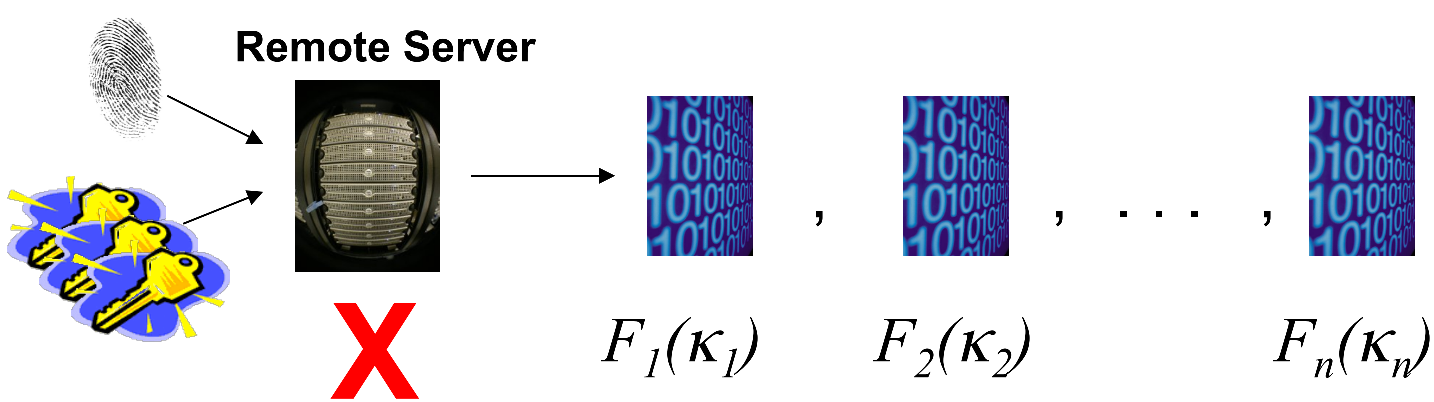
- w_j is re-encoded using a transformation function T
  - 1st encoding: $w_{j,1}(v', P)$
  - 2nd encoding: $w_{j,2}(w_{j,1}, T_2)$
  - nth encoding: $w_{j,n}(w_{j,n-1}, T_n)$
- The nesting process is formally invertible via the keys, but cryptographically secure

The Goal: Transactions



Does "nesting" apply to other secure template technologies?

- Fuzzy Vaults have already been "cracked," but…
- Any nesting of a fuzzy vault (with or without passwords) would have to be able to identify and then modify the data and the embedded key, which means the nesting system effectively knows the "secrets" and hence can compromise the security and privacy protection of the data.



Does "nesting" apply to other secure template technologies?

- Fuzzy commitment is reasonably secure
  - But its base formulation does *not* possess a nesting property
- The feature data X is always needed when changing keys

$$W_1 = C_1 \oplus X;$$
$$W_2 = C_2 \oplus X;$$
$$\ldots$$
$$W_n = C_n \oplus X$$

Does "nesting" apply to other secure template technologies?

- Fuzzy extractors* theoretically provide secure template protection.
  - But they do *not* possess a nesting property

Lemma 5.1*

Suppose we compose an $(m, \tilde{m}, t)$-secure sketch, (SS, REC) for a space M and a universal hash function $EXT : M \rightarrow \{0,1\}^l$ as follows: In Gen, choose a random i and let $P = (SS(w), i)$ and $R = Ext(w; i)$; let $Rep(w', (s, i)) = Ext(Rec(w', s), i)$. The result is an (m, l, t, ε)-fuzzy extractor with $l = \tilde{m} + 2 - 2\log(1/\varepsilon)$.

One needs the original biometric data w and a random i to create a new instance of a fuzzy extractor!

*Y. Dodis, L. Reyzin and A. Smith, "Fuzzy Extractors." in *Security with Noisy Data*, Springer-Verlag, 2007

Bipartite Biotokens

– Let B be a revocable biotoken. A bipartite biotoken* B_p is a transformation $bb_{j,k}$ of user j's k^{th} instance of B. Any bipartite biotoken $B_{p,k}$ can match any revocable biotoken B_k for the same user.

– $bb_{j,k}$ must allow the embedding of some data d into B_p

  - $bb_{j,k}(w_{j,k}, T_k, d)$

– If $B_{p,k}$ and B_k match, d is released

* W. Scheirer and T. Boult, "Bipartite Biotokens: Definition, Implementation, and Analysis," ICB 2009.

Experimental Results

FVC02 DB 2	112 Bits GAR	112 Bits FAR	128 Bits GAR	128 Bits FAR	160 Bits GAR	160 Bits FAR
F.P. Fuzzy Vaults[1]	89	0.13	89	0.01	84	0
F.P. FV, Mosaic with 2 Queries[1]	96	0.24	95	0.04	89	0
Password Vault[2]	88	?	86	?	79	?
Bipartite Biotokens	97	0	97	0	97	0

Comparison with Fuzzy Faults

FVC02 DB #	192 Bits GAR	192 Bits ECC	256 Bits GAR	256 Bits ECC	512 Bits GAR	512 Bits ECC	1024 Bits GAR	1024 Bits ECC
1	97	5	94	2	95	5	77	10
2	97	2	97	2	92	6	82	9

Larger Key Sizes

1. K. Nandakumar, A. K. Jain and S. Pankanti, “Fingerprint-based Fuzzy Vault: Implementation and Performance”, In IEEE TIFS, vol. 2, no. 4, 2007

2. K. Nandakumar, A. Nagar and A. K. Jain, “Hardening Fingerprint Fuzzy Vault Using Password”, in Proc. of ICB 2007

The **Big** Test

- The Doppleganger Attack
  - If the FAR is 1 in X, then an attacker can try more than X attempts
- A very large impostor test
  - Mixed combinations of FVC 2002, FVC 2004, NIST DB 14 and NIST DB 29
  - 6 bytes of ECC, 128 bit, 256 bit, and 512 bit keys, 8000 byte probe/gallery biotokens

Zero False Accepts from processing over 1 Billion impostor trials to date!

Protocols and Applications

Security in the "Cloud"

- The model isn't new: an updated version of timesharing from the 1960s…
  - Many popular services have always been in the cloud
    - **Gmail** →
    - **Facebook** →
    - **Paypal** →
    - **Dropbox** →

- What is different from the PC model: the trust boundary shifts one step further away

Risks of the Cloud

"If you entrust your data to others, they can let you down or outright betray you*."

- Misplaced, stolen or sold data
- Less privacy protection in practice and under the law
- Vendor defines how much control a user has over their own data

*J. Zittrain, "Lost in the Cloud," New York Times, 2009.

Risks of the Cloud

- Is it dangerous to move biometric data to the cloud?
  – Maybe Not
- The key issue*: another level of trust
  – When a computer is on your network, you control the security mechanisms
  – There should be some facility for the owner to protect and control their data

*B. Schneier, “Cloud Computing,” Schneier on Security, 2009.

Biometric Solution?

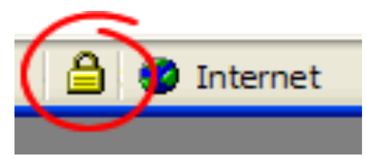
- By adding a second factor, we can mitigate the inherent trust problem with the cloud model
- What about Biometrics?
  - Improved non-repudiation
  - Strong verification for actors in a transaction
  - Strong verification for PKI-like functionality
    - certificate authority establishment, and general certificate issue

Address the trouble with Biometrics using Template Protection

Biocryptographic Key Infrastructure

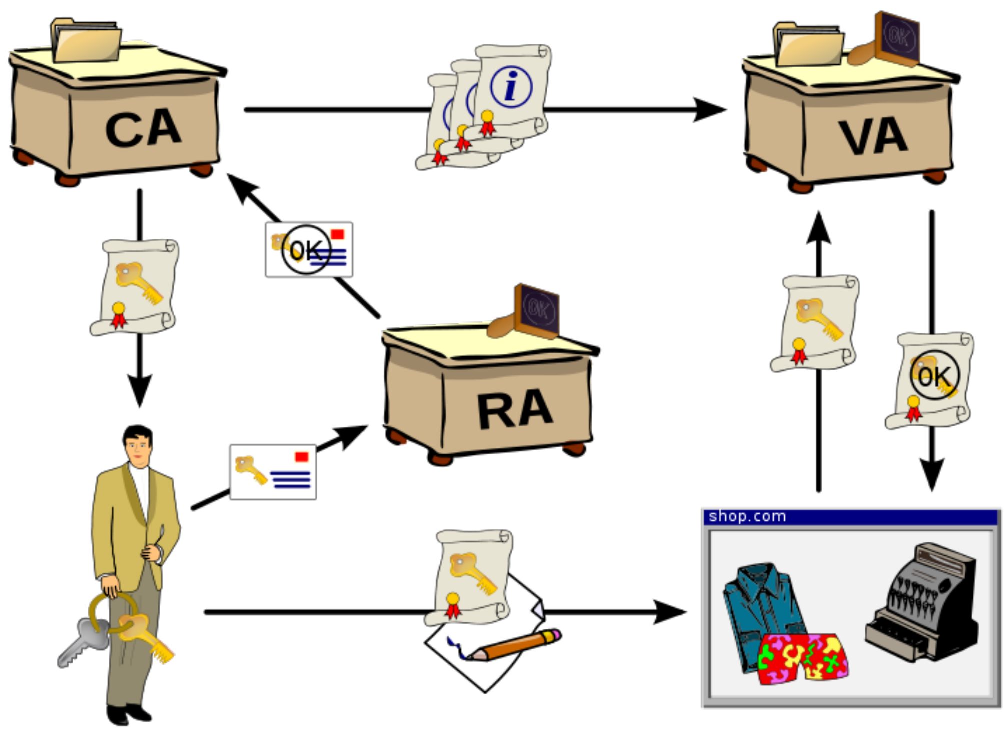
- Solution to both traditional and biometric data management in the cloud
- Analogous to PKI, but incorporates biometrics and template protection to establish identity beyond certificates

Public Key Infrastructure enables asymmetric secure machine-to-machine communication, but it does not solve Identity Issues. We need asymmetric verification.



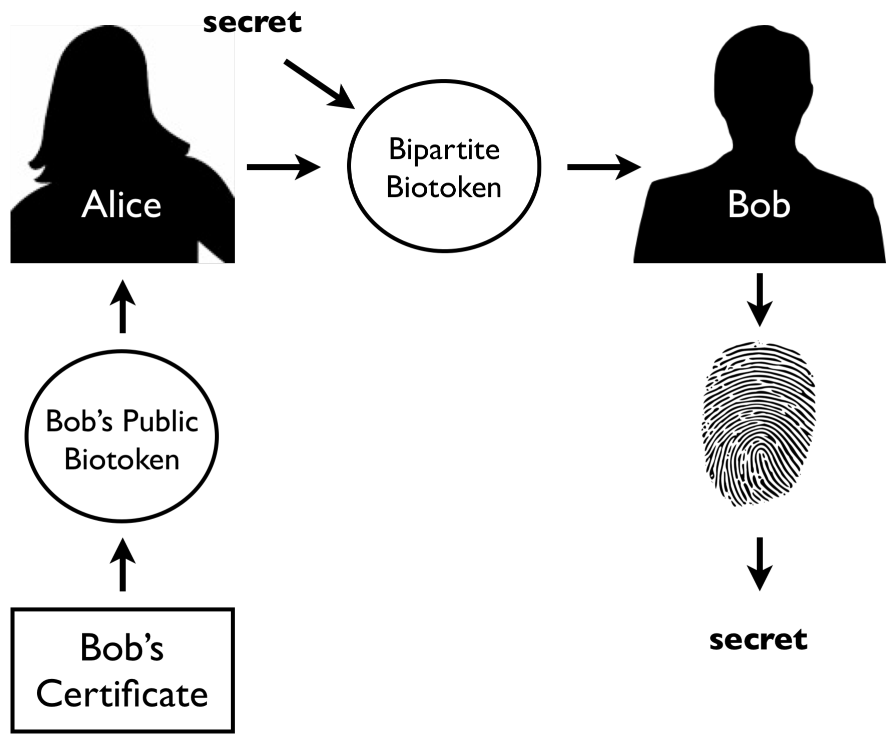
Public Key Infrastructure

- PKI is the infrastructure for handling the complete management of digital certificates (x.509 compliant)
  - Certificates contain trusted information: a public key



Benefit of a BKI

- Ability to store public biotokens in digital certificates
  - Any entity in the infrastructure can send secret data that only the owner of the biotoken can unlock



Requirements for a Biocryptographic Key Infrastructure

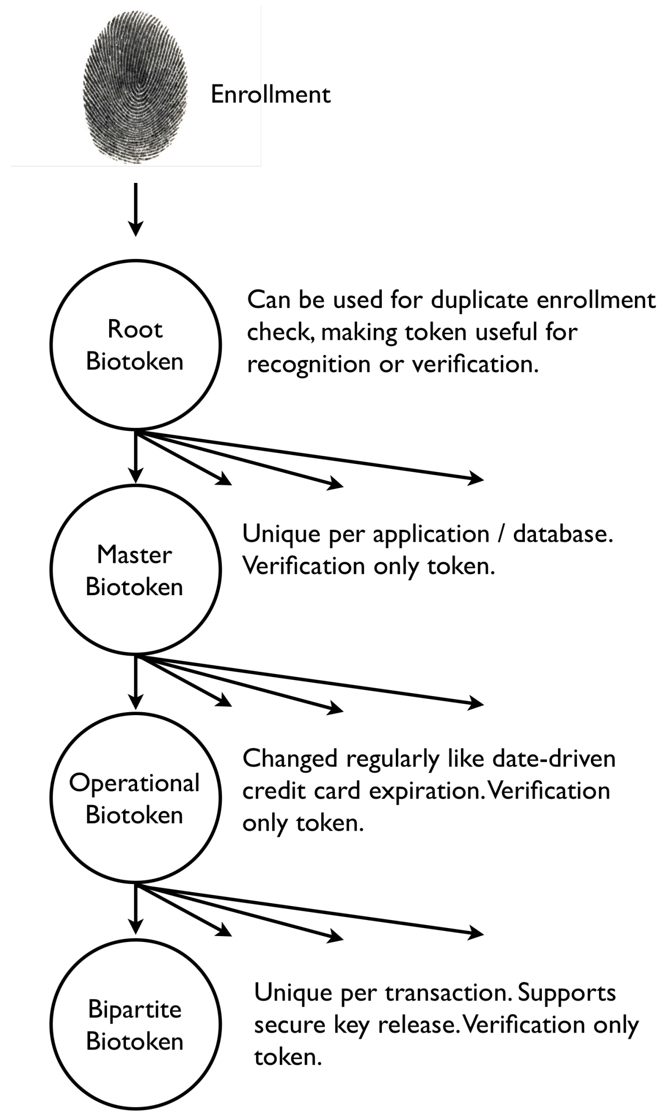
1. Cryptographically strong protection of the underlying biometric features
2. Ability to revoke and re-issue templates
3. Nested re-encoding, allowing a hierarchy of templates to be generated from a single base template
4. Support for public templates
5. Key-binding capability without the need of intervention by the person associated with the template

Can a BKI be supported by other technologies besides revocable biotokens?

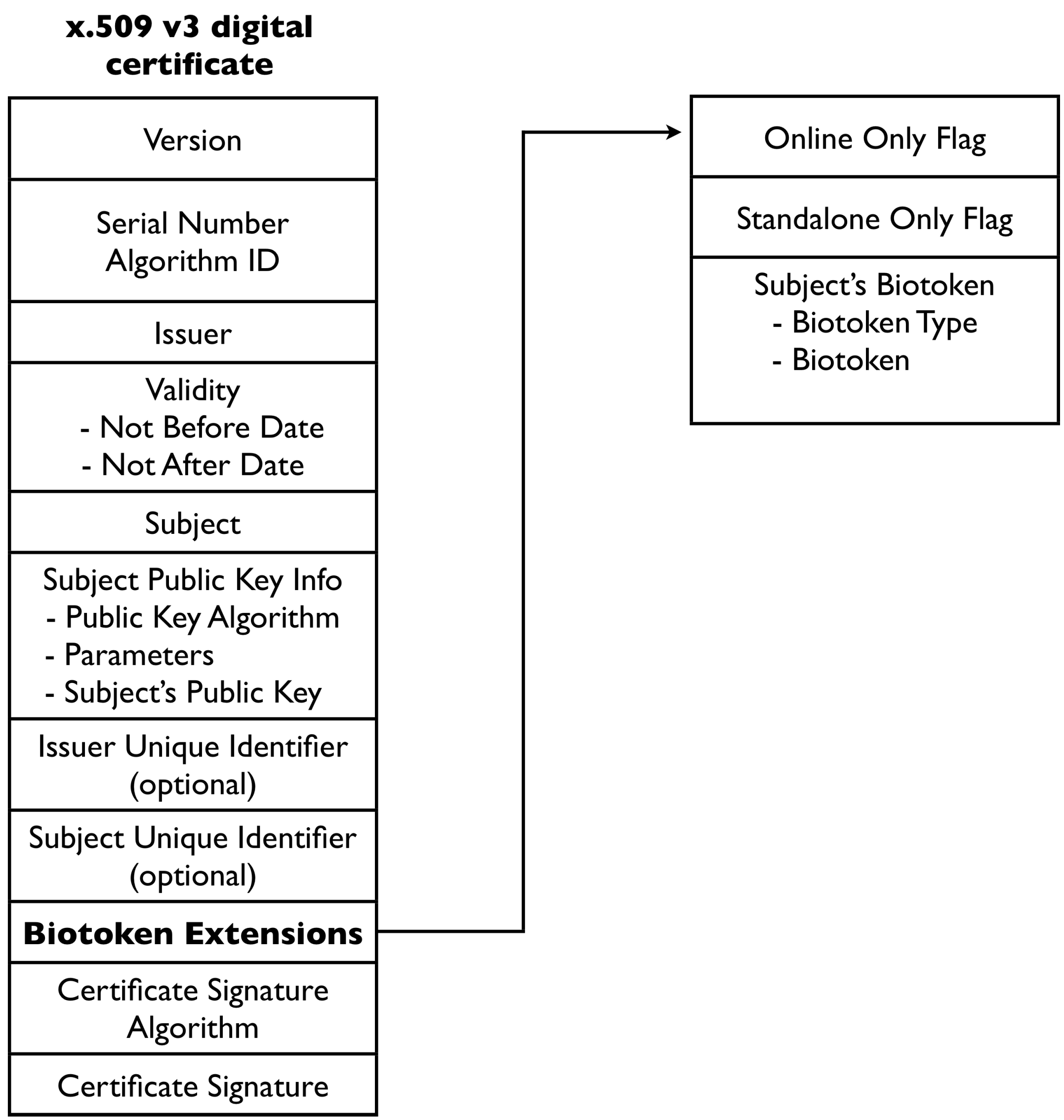
- Fuzzy Extractors support key transfer[1], but not unique transactions
- Kanade et al.[2] proposed a scheme for key-binding without re-enrollment
  - secret key + error correction $\Theta_{ps} \oplus$ shuffled biometric data $\Theta_{canc} = \Theta_{lock}$
  - Vulnerable to the SKI Attack: If an attacker knows Θ_{ps}, then $\Theta_{ps} \oplus \Theta_{lock} = \Theta_{canc}$

1. X. Boyen et al. "Securics Remote Authentication Using Biometric Data," EUROCRYPT, 2005.
2. S. Kanade et al. "Generating and Sharing Biometrics Based Session Keys for Secure Cryptographic Applications," IEEE BTAS, 2010

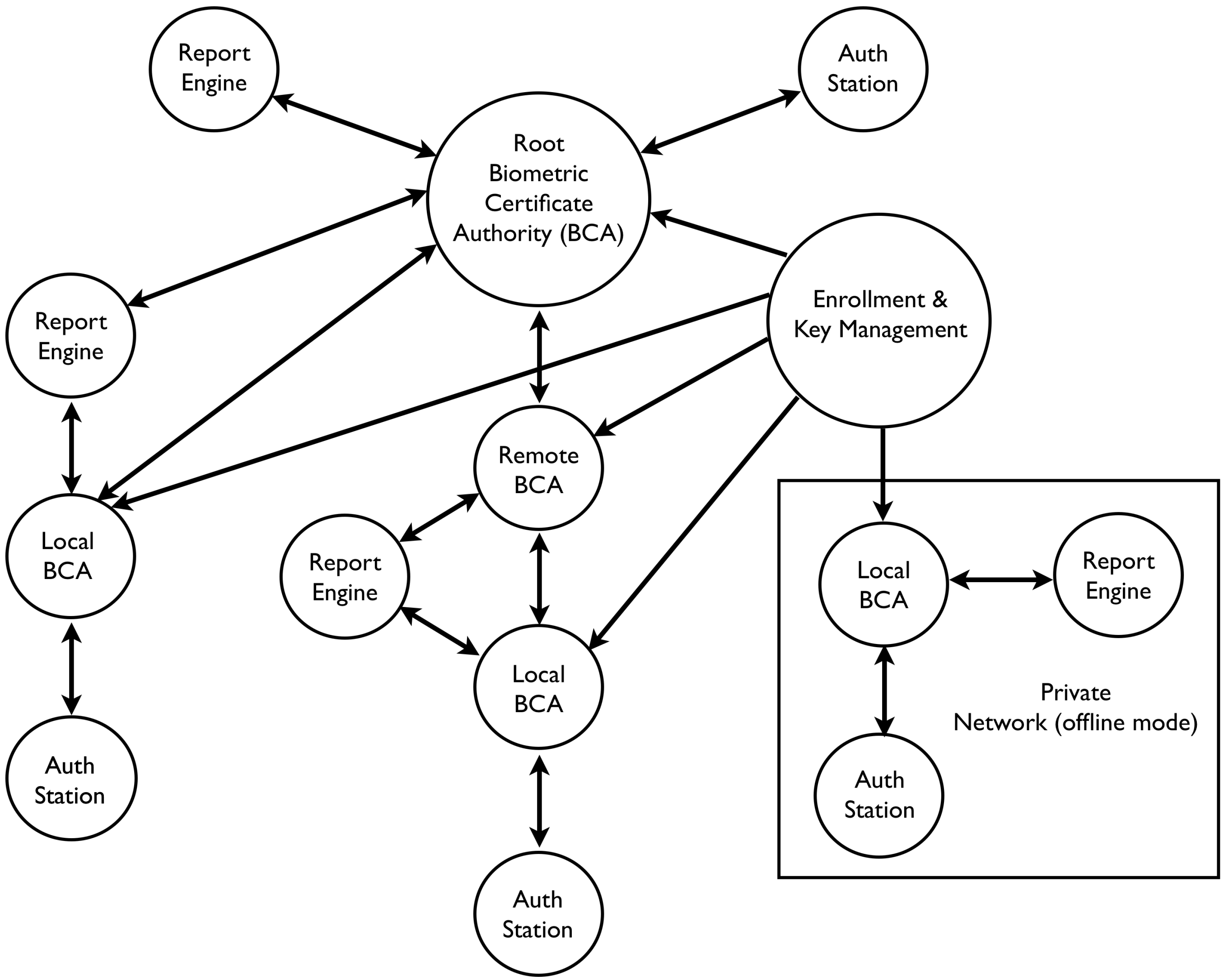
Biotoken Issue/Re-Issue Tree



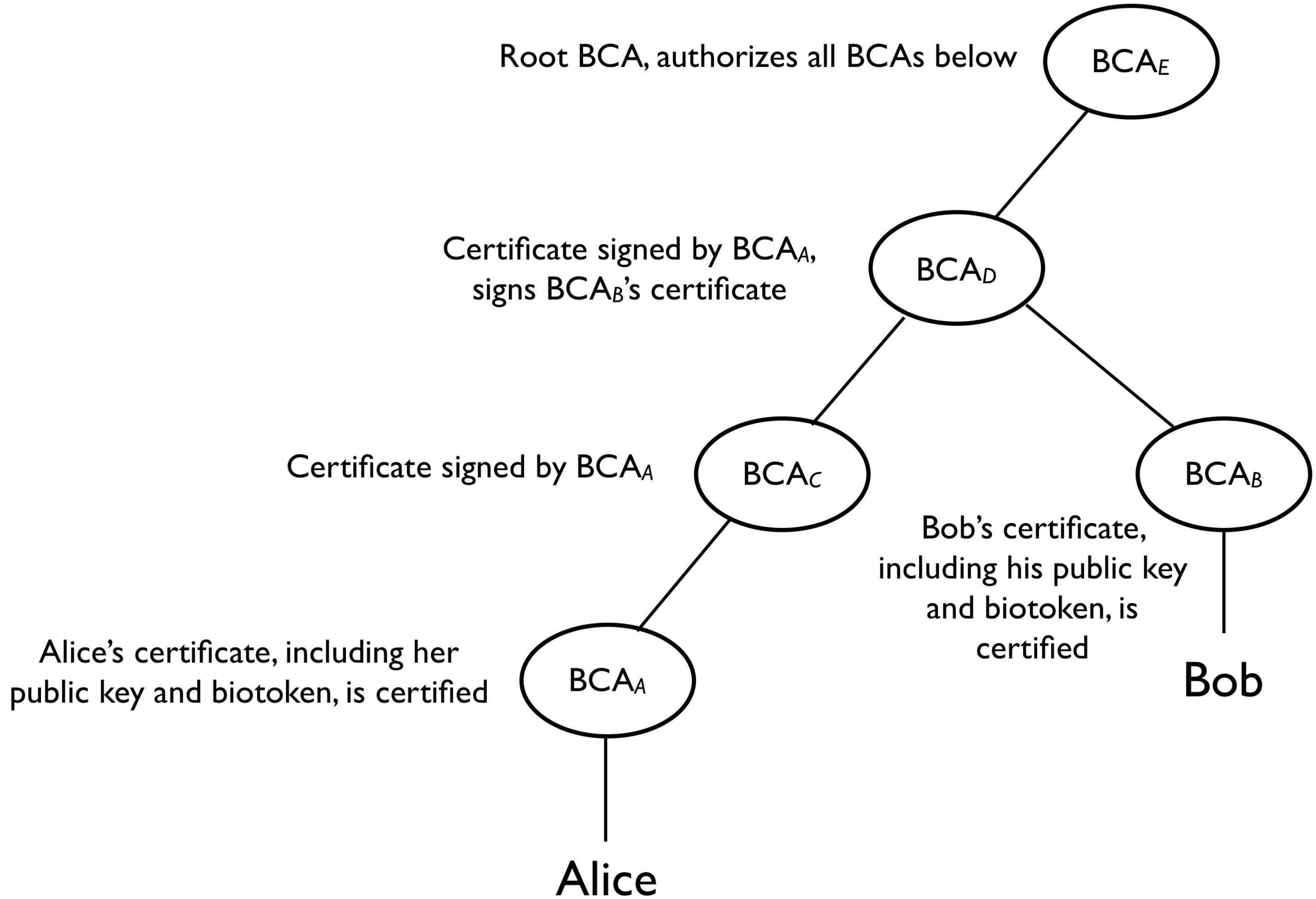
Digital Cert. Supporting Biotokens



A Biocryptographic Key Infrastructure

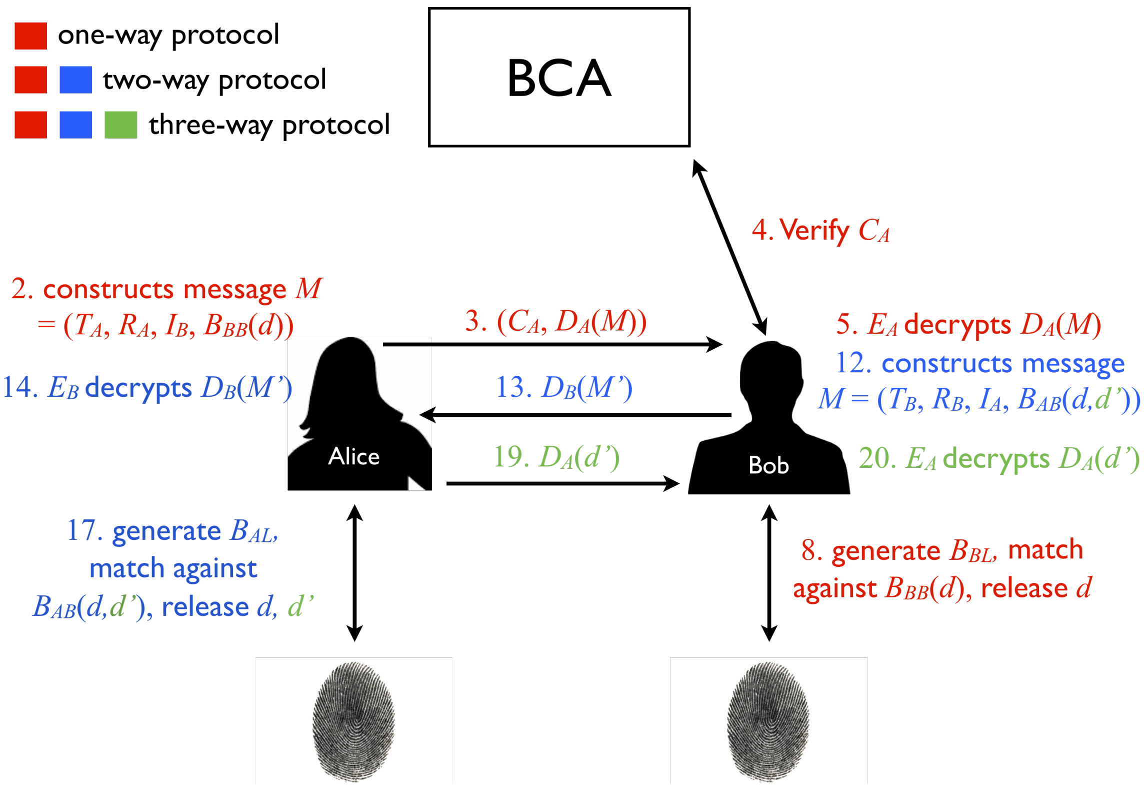


Certificate Retrieval Path



Three Authentication Protocols

- 1-Way protocol: establishes identity and trust of Receiver
- 2-Way protocol: assures send that Receiver is not impostor
- 3-Way protocol: validates both identities in the transaction



Certificate Revocation

- We must consider certificate *and* biometric re-issue
- Scenario 1: Manual re-issue
  - Certificate owner generates a new public-private key pair and a new biotoken
- Scenario 2: Automatic re-issue of biotoken
  - BCA retains transformation keys, reverts public biotoken to a lower level, issues new transformation keys and public biotoken
- Scenario 3: Automatic re-issue of key-pair
  - BCA issues new key-pair, transmits secret key to owner via bipartite biotoken

CRN Message

Certificate Re-issue Notification

Serial Number
New Serial Number
Biotoken Re-issued Flag
Key-pair Re-issued Flag
Biotoken and Key-pair Revoked Flag
*Keyring for Biotoken (Optional)
Biotoken Type (Optional)
Biotoken (Optional)
Signature

*Keyring is encrypted with the user's public key

New Applications

- Authenticate to the cloud
- Manage your own data in the cloud

And also:

- Thwart Man-in-the-Middle and Phishing attacks!
- Bio-Kerberos
- Bio-S/Key
- BKI-enabled LDAP
- Biometric Digital Signatures
- Mobile Biometrics

The BKI bring identity to crypto protocols

Commercial Solutions

- GenKey (http://www.priv-id.com)
  - Fuzzy Commitment (?)
- Securics (http://www.securics.com)
  - Revocable Biotokens

genkey

- Offering a host of privacy related software products
  - BioHASH SecureID SDK
  - BioHASH Match-on-Card SDK
  - Biometric ID Management System
- Established research group with strong publication record
  - Published work through Philips and the University of Twente
  - "Security with Noisy Data*" is even advertised on their site!

*P. Tuyls, B. Skoric, and T. Kevenaar (eds.), "Security with Noisy Data," Springer-Verlag, 2007.

UNIVERSITY OF COLORADO AT COLORADO SPRINGS

- Multiple products built around Revocable Biotokens and the BKI
- We've published the details as Securics, Inc. and the University of Colorado
- Have questions about our technology???
  - Please ask!

Questions?