Cryptography: Building The Future

Prof Bill Buchanan OBE

http://asecuritysite.com

Twitter: billatnapier

Tokenization

BLOCKPASS IDENTITY LAB

World-leading Collaboration between
Blockpass IDN and Edinburgh Napier University

Audit Compliance

Citizen rights to access their own data

Detect
Respond
Investigate

Incident Response

Encryption

Pseudo-anonymity

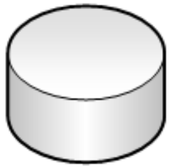
Surrogate identifiers

Personally Identifiable Information (PII)

PAN – Primary Account Number

ID=543 611 041

Name: Bobby Smith
Address: 10 Eve Row
Date of Birth: 5/5/55



Surrogate mapping table

Real	Surrogate
ID=543 611 041	ID=741 534 011
ID=533 841 943	ID= 666 001 845

Transactions

Surrogate Identifier

ID=741 534 011

ID	Transaction
741 534 001	Pay 666 001 845 $10
532 550 423	Pay 741 534 011 $190

Tokenization with currency

Normal currency

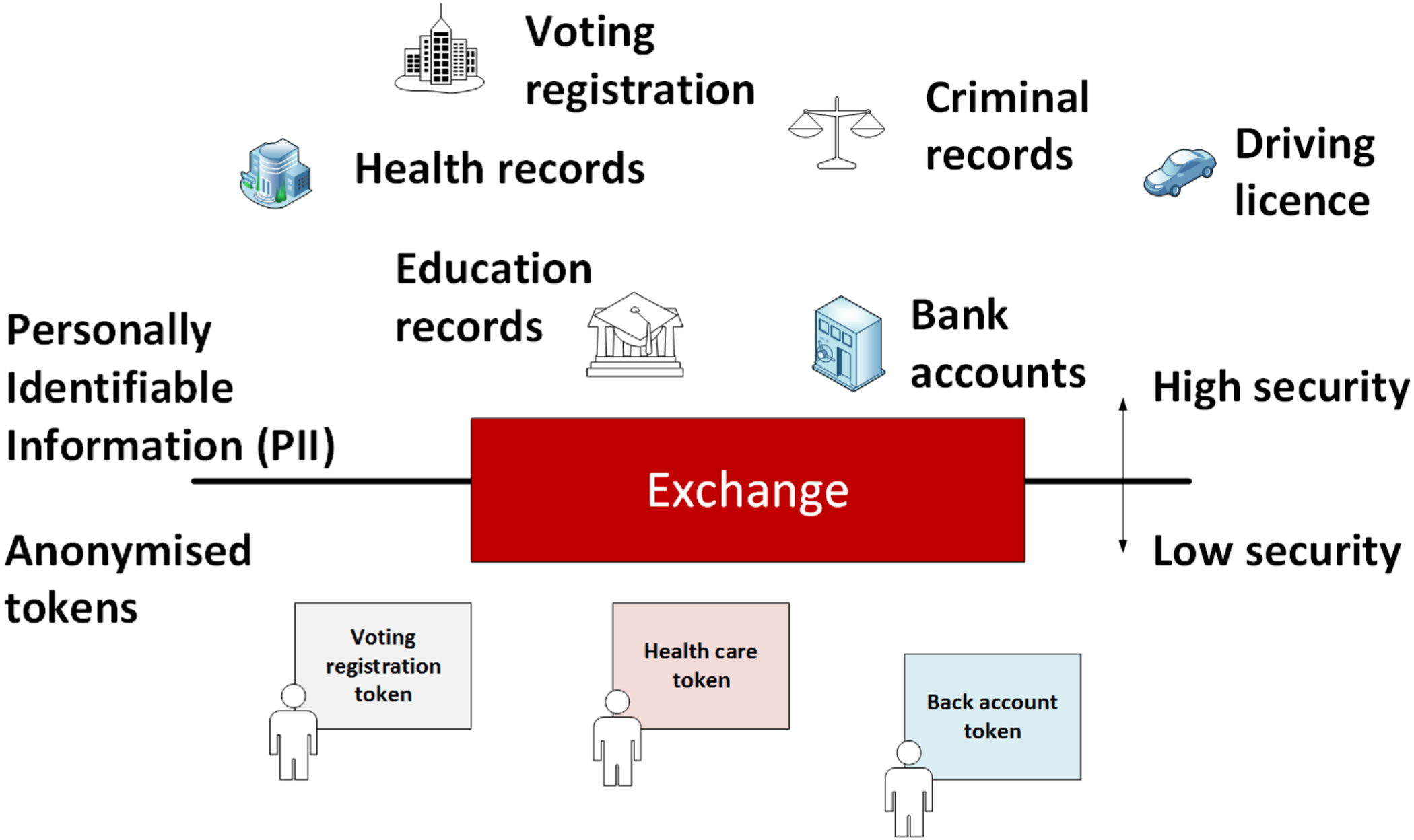
High security

Exchange

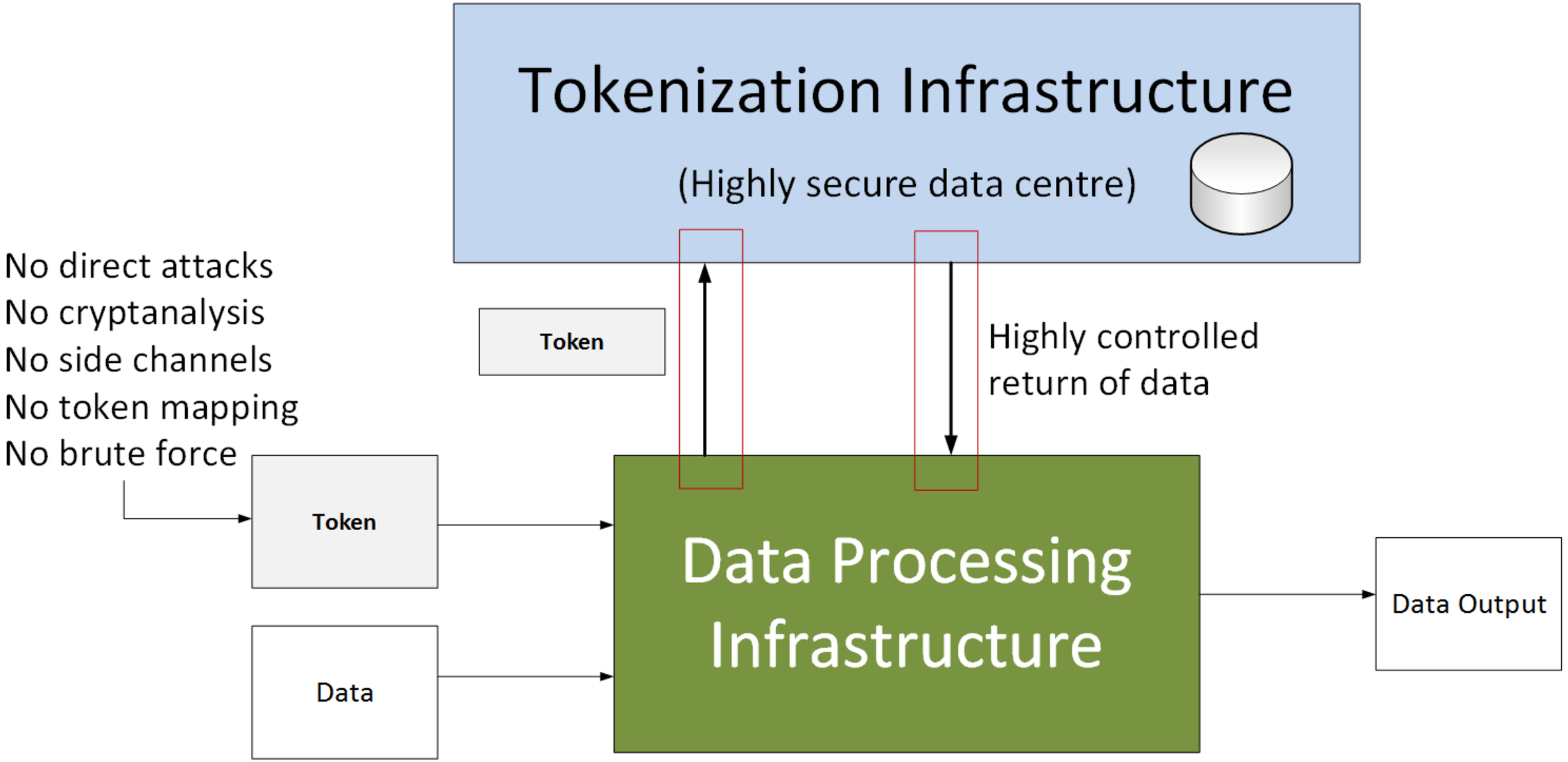
Tokens

Low security

Tokenization with data



Tokenization with data



Visa Best Practice for Tokenization

Tokenization Infrastructure

- Visa Best Practices

PAN - Primary Account Number
Payment Card Industry Data Security Standards (PCI DSS).

128-bit AES
2048 bit RSA.
224 bit ECC.
224 bit SHA

No direct attacks
No cryptanalysis
No side channels
No token mapping
No brute force

Recovery of PAN not possible

One-way function

Strong encryption

Token Generation

Token Mapping

- PAN -> Token ID mapping.
- Refund of transaction without ever revealing PAN.

Key Management

Credit Card Data Vault

One-time use

Multi-use

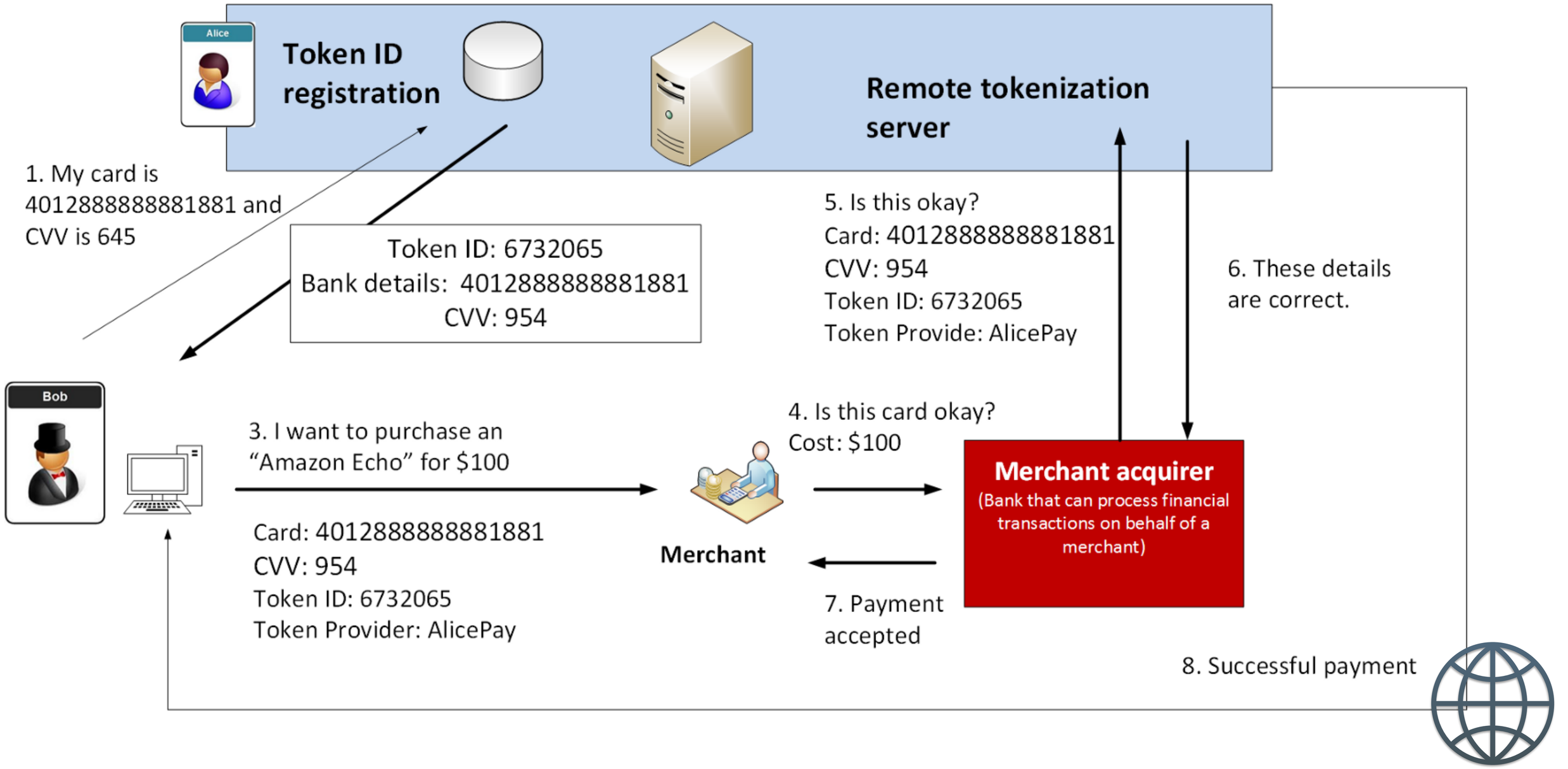
All link to PAN

Token ID: Hash(PAN+Salt) or Encryption of PAN using unique token key
No tracking by other business units.
If salt used: At least 64 bits

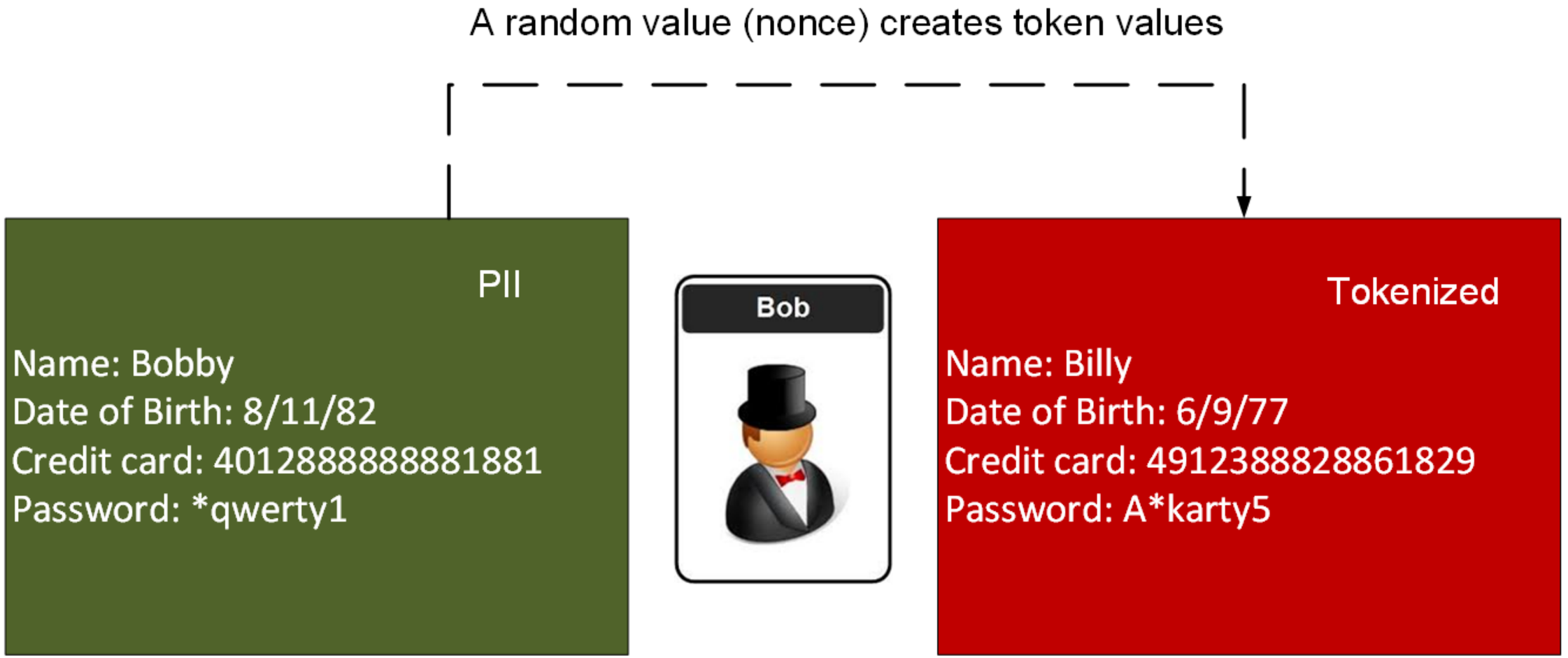
- Network Segmentation
- Strong Authorization.
- 24x7 Monitoring.
- Token Distinguishers. Clear demarcation between token IDs and credit card IDs.
- Logging of transactions must be protected.

- Strong encryption
- PCI Compliance.

Token Mapping

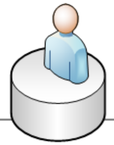


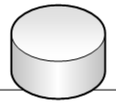
Token Mapping



Pseudonymization

"pseudonymization" – a process rendering data neither anonymous nor directly identifying. Pseudonymization is the separation of data from direct identifiers so that linkage to an identity is not possible without additional information that is held separately.





Name	CHI
Fred Bloggs	0603614357

CHI	Age	Gender
0603614357	55	Male

Address

Juniper Green

Patient ID Lookup

Community Health Index

This number is normally formed using the patient's date of birth (as DDMMYY), followed by four digits: two digits randomly generated, the third digit identifying gender at birth (odd for men, even for women) and a check digit.

Email

w.buchanan@napier.ac.uk

Patient Record

10-digit CHI number, and which is the basis of the identity of health records).

Patient's date of birth (DDMMYY), and then two random digits and then two digits for their gender at birth (odd for male, and even for female). At the end we have a check digit.

Thus the CHI number of a male born on 5 Feb 2016 can be: 0502160510. This number should NEVER be revealed on the database, but we need something that looks like it. In this way FPE can replace the actual CHI number.

Ledger approach

Medication Record

0603614357

2mg Penicillin

XXXXXXXXXX

XXXXXXXXXX

XXXXXXXXXX

0603614357

4mg Aspirin

H/S

0603614357

Hash with Salt

5473b74302

2mg Penicillin

XXXXXXXXXX

XXXXXXXXXX

XXXXXXXXXX

7143540765

4mg Aspirin

Name

CHI

Fred Bloggs

0603614357

Salt

Patient ID Lookup

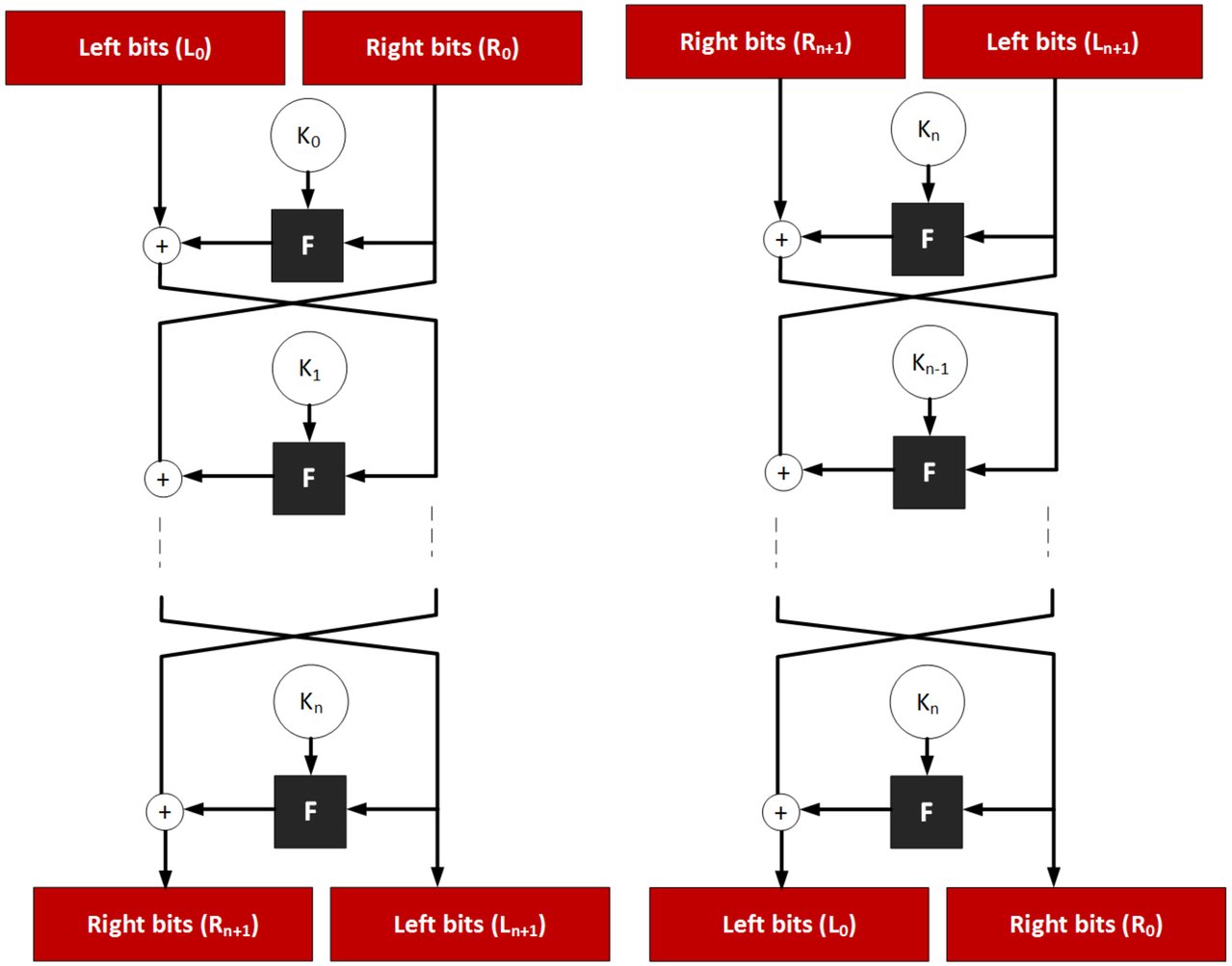
Fred Bloggs

0603614357

5473b74302

7143540765

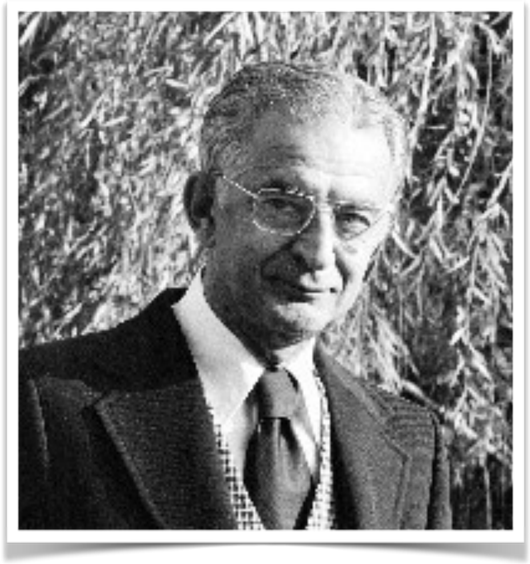
Feistel structure



$$L_{i+1} = R_i$$

$$R_{i+1} = L_i \oplus F(R_i, K_i)$$

The Feistel cipher applies a symmetric key infrastructure and was named after Horst Feistel. His work at IBM led to the creation of the Lucifer and DES ciphers.

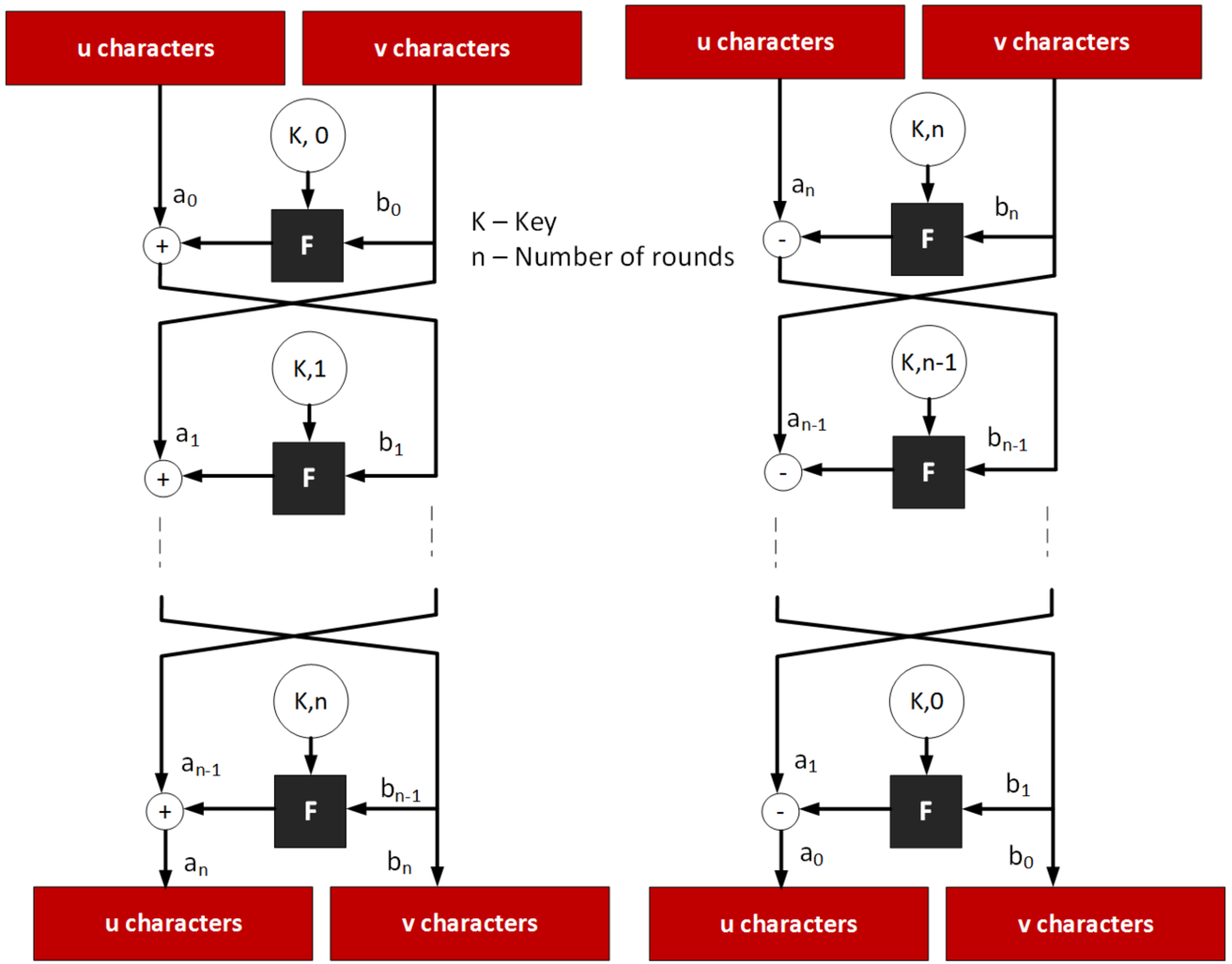


$$F(x, k) = (i \times k)^x \pmod{2^{32} - 1}$$

$$L_i = R_{i-1}$$

$$R_i = L_{i-1} \oplus F(R_{i-1}, k_i)$$

FFX



NIST have thus defined a standard known as SP 800–38G

Format-preserving, Feistel-based encryption

For FF1 we have 10 rounds and for FF3 we have eight rounds

Radix - number of characters in the output

```
h = hmac.new(self.key, key +
struct.pack('I', i),
self.digestmod)

c = self.add(radix, a,
self.round(radix, i, b))
a, b = b, c
```

```
c = self.sub(radix, a,
self.round(radix, i, b))
a, b = b, c
```

Honey Encryption

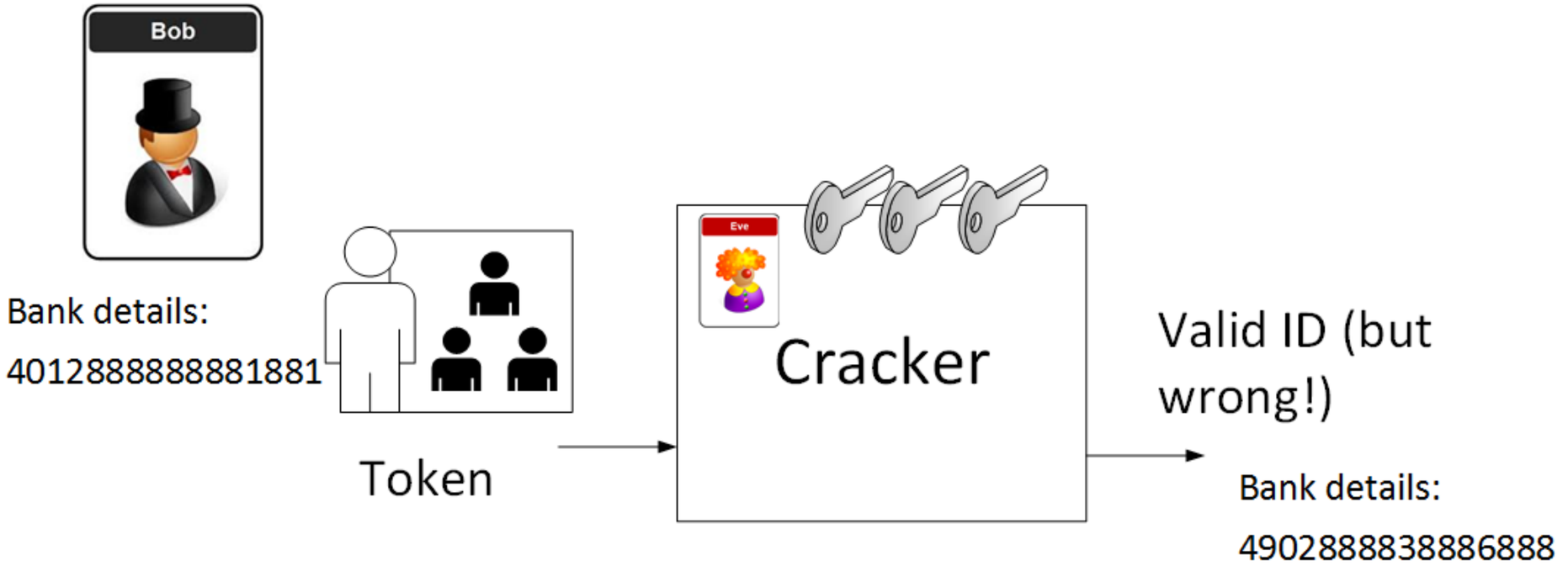


Table 1

'prefix':	[numRandom,	cardLength,	probWeight]:
'604646':	[0, 16, 1],		
'519293':	[0, 16, 1],		
'519290':	[0, 16, 1],		
'479293':	[0, 16, 1],		
'435744':	[0, 16, 1],		
'421323':	[0, 16, 1],		
'377441':	[0, 15, 1] …		
}			

Cryptography: Building The Future

Prof Bill Buchanan OBE

http://asecuritysite.com

Twitter: billatnapier

Tokenization

BLOCKPASS IDENTITY LAB

World-leading Collaboration between
Blockpass IDN and Edinburgh Napier University