FASSKEY

Authentication for Everyone

Our Goals

- Secure & Simple Authentication and Identification for any purpose
- Simple and private Mobile payment and shopping
- Especially for the un-banked and developing countries

Username/Password

- So many breaches
- So many bad servers
- Poor/reused passwords
- Bigger website=bigger target
- Mobile servers: same problems
- Hopeless

myspace	359,420,698	MySpace accounts
in	164,611,595	LinkedIn accounts
	152,445,165	Adobe accounts
tumblr.	65,469,298	tumblr accounts
Fling.com	40,767,652	Fling accounts
	30,811,934	Ashley Madison accounts
mate1	27,393,015	Mate1.com accounts
	13,545,468	000webhost accounts
	13,186,088	R2Games accounts
gamigo	8,243,604	Gamigo accounts
Heroes of Newerth	8,089,103	Heroes of Newerth accounts
lifeboat	7,089,395	Lifeboat accounts
	5,915,013	Nexus Mods accounts
vtech	4,833,678	VTech accounts
@mail.ru	4,821,262	mail.ru Dump accounts
	4,789,599	Bitcoin Security Forum Gmail Dump accounts

Source: https://haveibeenpwned.com/PwnedWebsites

The 50 Most Used Passwords

1. 123456
2. password
3. 12345678
4. qwerty
5. 123456789
6. 12345
7. 1234
8. 111111
9. 1234567
10. dragon
11. 123123
12. baseball
13. abc123
14. football
15. monkey
16. letmein
17. shadow
18. master
19. 696969
20. michael
21. mustang
22. 666666
23. qwertyuiop
24. 123321
25. 1234...890
26. p*s*y
27. superman
28. 270
29. 654321
30. 1qaz2wsx
31. 7777777
32. f*cky*u
33. qazwsx
34. jordan
35. jennifer
36. 123qwe
37. 121212
38. killer
39. trustno1
40. hunter
41. harley
42. zxcvbnm
43. asdfgh
44. buster
45. andrew
46. batman
47. soccer
48. tigger
49. charlie
50. robert

Source: https://wpengine.com/unmasked/

	Position	Password	Entropy	Crack Time
SB Projects	Agent to the stars	clydek	12.2	0.2 seconds
twitter	Former Senior	s3ash311 (seashell)	15.6	2.5 seconds
TIME	Director	indiana	9.8	0.04 seconds
McDonald's (i'm lovin' it)	Senior Director	wuxi6969	21.9	5 minutes
Google	Software Engineer	muffins	12.3	0.2 seconds
IBM	Senior Manager	123456	1.0	0 seconds
CNBC	On-air Correspondent	ballet	12.2	0.24 seconds
Al Jazeera	Journalist	raven639	19.4	34 seconds
PayPal	Senior Engineer	ez1422	23.6	12 minutes
abc	Digital Reporter	blah11	14.7	1.3 seconds
sage	Manager	b0j1k0	27.8	5 hours
The New York Times	Editor	kellymisty	14.2	0.9 seconds
BuzzFeed	Editor	shagwell	16.8	5.8 seconds
Department of State	Division Chief	lincoln	10.9	0.09 seconds
WIRED	Editor	dgneo	22.0	5 minutes
WSJ	Editor	sassy1	12.0	0.2 seconds
NIKE	Global Director	[firstname]	9.1	0.02 seconds
Microsoft	Program Manager	123[yearofbirth]	10.4	0.07 seconds

Hacking real people's passwords

Source: https://wpengine.com/unmasked/

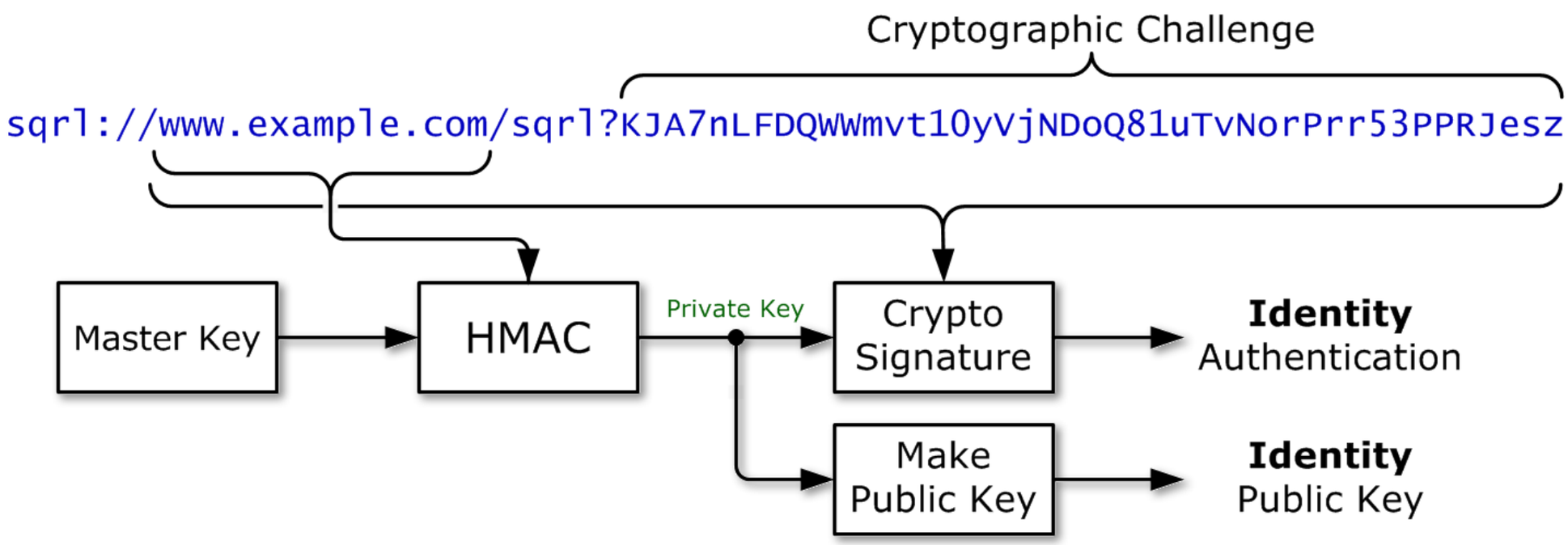
FASSKEY Method

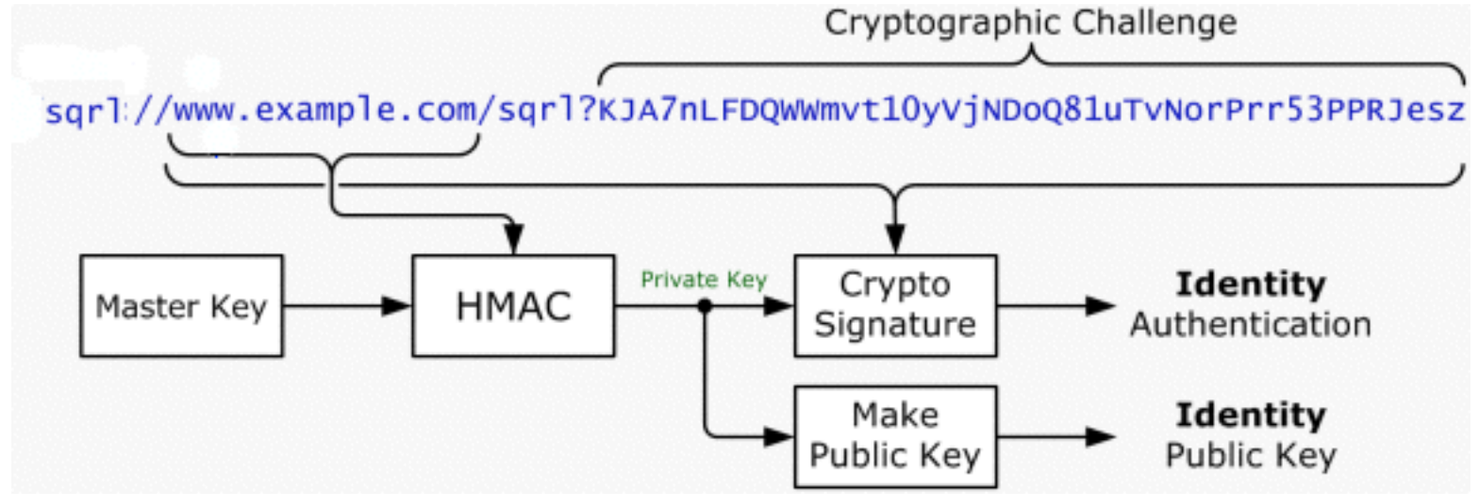
- Based on Steve Gibson's public domain SQRL (https://www.grc.com/sqrl)
- User generates a single, random 256 bit EC Master Key
- A web site's domain name is used to key a hash which produces a private key. The matching public key is created and registered with the website as the user's identity token for that site.
- To authenticate the user at logon, the website presents a per-logon "nonce" (qlink). The user signs the qlink using the site-specific private key and returns both the site-specific public key and the "nonce" signed by the matching private key.
- All messages are encrypted, authenticated and signed.

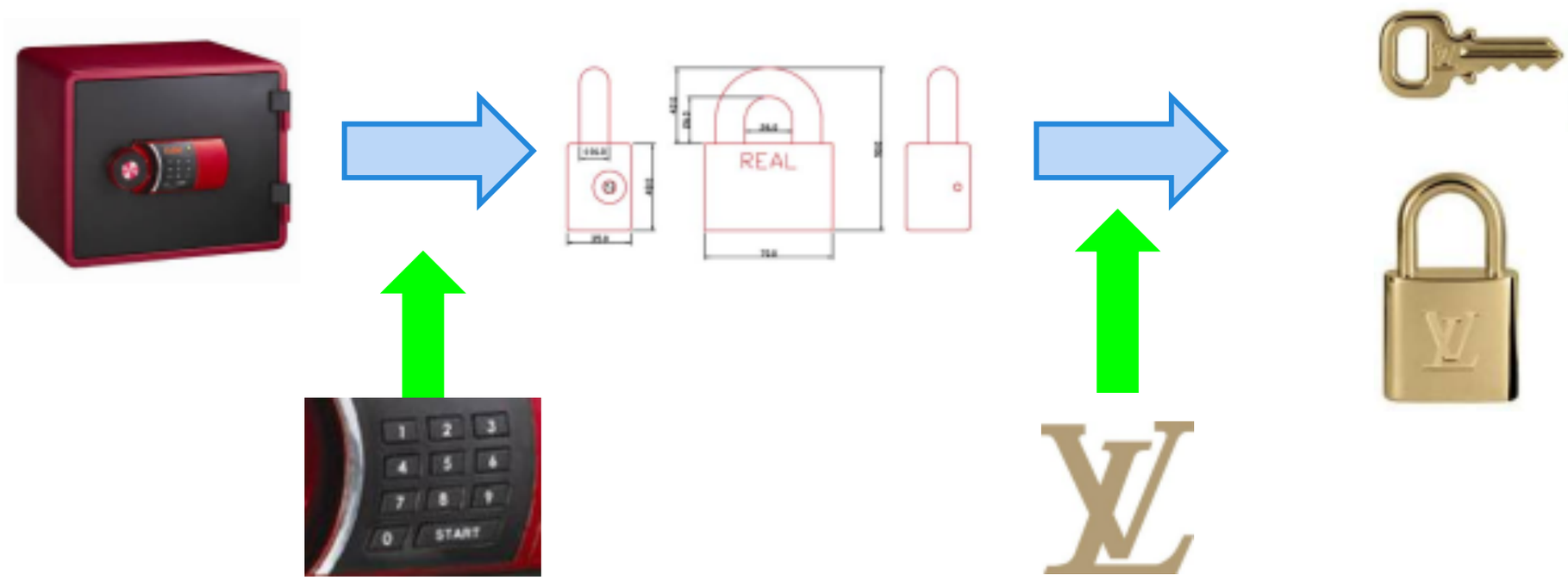
qlink

- Domain for key generation
- 32 byte nut includes: timestamp, site & operator IDs, nonce, MAC
- protocol type
- 32 byte server ephemeral public key

qrl://demo.fasskey.com/squal_auth_login.php?
nut=zBPmr7UN1rO-kMCWFAfAqEtS5ibAqOYq59n7Cxtlse0
&protocol=SQT-AUTH
&epk=ka46kc_XlCevG34NAYwizRqq9JRJ58QGphQ-5o24Hds


Cryptographic Challenge
sqrl://www.example.com/sqrl?KJA7nLFDQWWmvt10yVjNDoQ81uTvNorPrr53PPRJesz
Master Key
HMAC
Private Key
Crypto Signature
Identity Authentication
Make Public Key
Identity Public Key



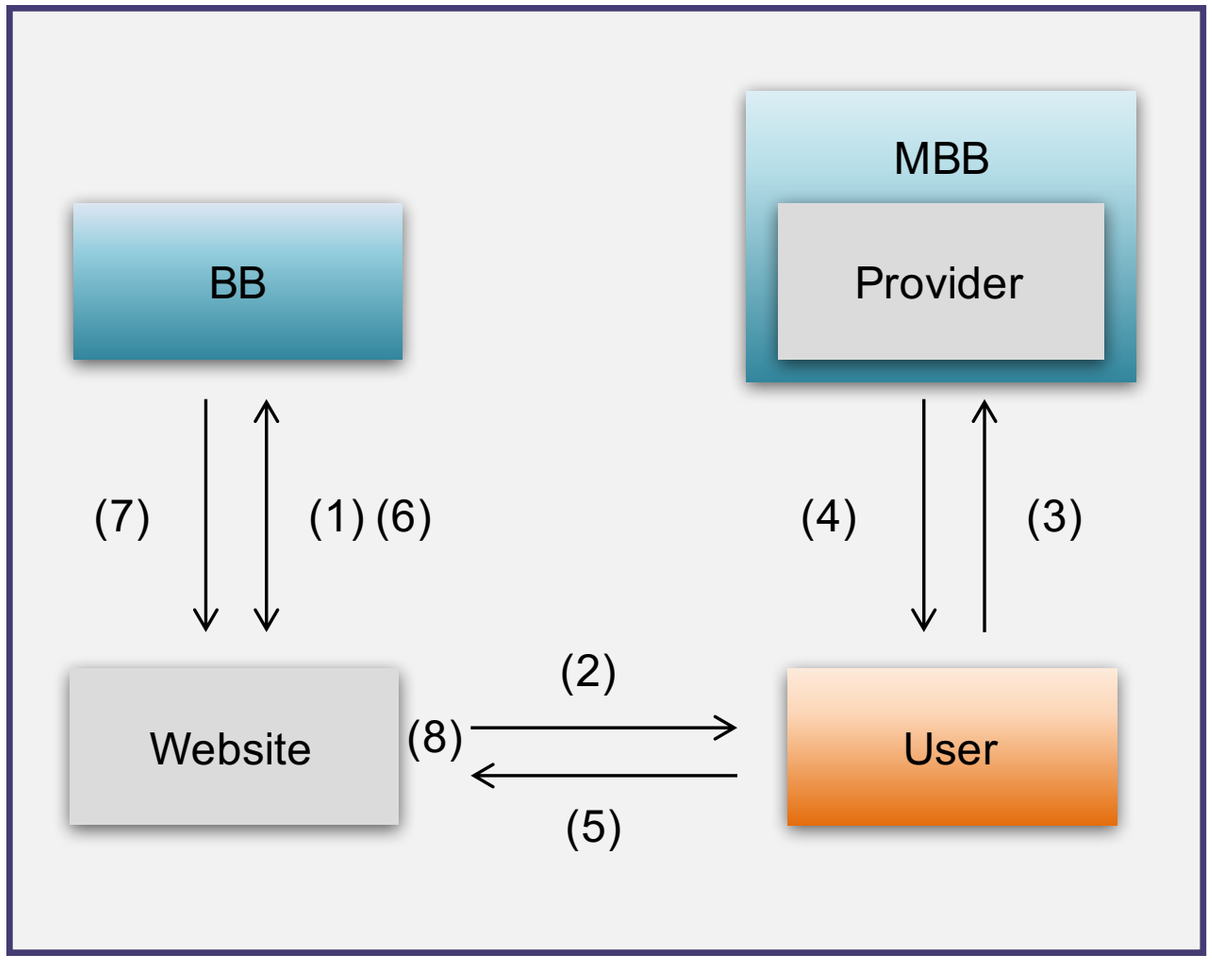


Creating per-site key pairs

This means:

- Users are per-site pseudonymous.
  - They present a unique but fixed identity to every site.
  - Inter-site tracking is eliminated.
- Since identities are deterministically synthesized from the site's domain name, so no per-site data needs to be stored.
- Users give web servers no secrets to keep. Web servers receive an identity token that is only meaningful for that site.
  - Disclosure of the public key is not damaging
  - No cross site correlation or tracking
- The use of a nonce using a simple challenge/response mechanism prevents reuse / replay attacks.

Authentication Protocol Flow



V (1). Website gets qlink from BB.
S (2). User receives qlink from Website.
T (3). User sends qlink to Provider(MBB) for checking.
T (4). User receives attestation key from Provider(MBB), proving Website.
T (5). User sends signed, TLE response to website.
V (6). Website forwards response to BB.
V (7). BB approves User and gives User ID.
S (8) Logged in, web page updated.

V – VPN connection
T – TLE protocol
S – OptionalSSL / TLS connection

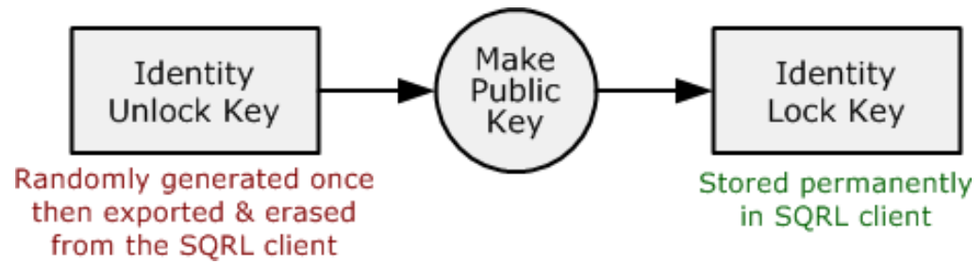
Problems

- User must remember one strong password, which never leaves his device
- Website cannot “recover” or “reset” user’s authentication
- If master key is compromised…

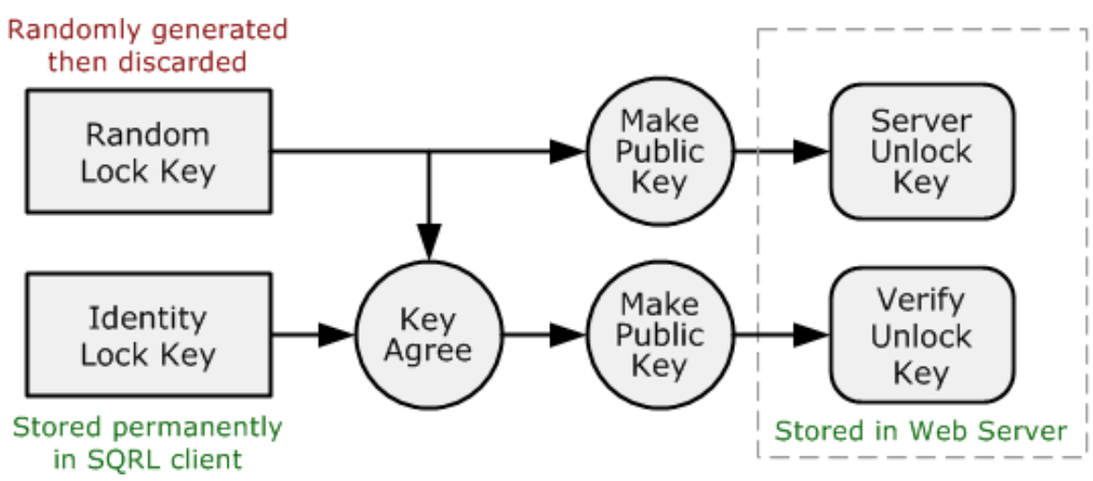
Lock/Unlock/Rekey

- We use a special key pair for these operations
- The user may lock any or all sites from his client
- Unlock or Rekey can be done only with special “unlock key”, which is never stored in the client.
- Unlock key is generated along with master key, then stored offline, via printout and operator backup
- The unlock key is an EC secret key, the matching public key is called the “lock key”.

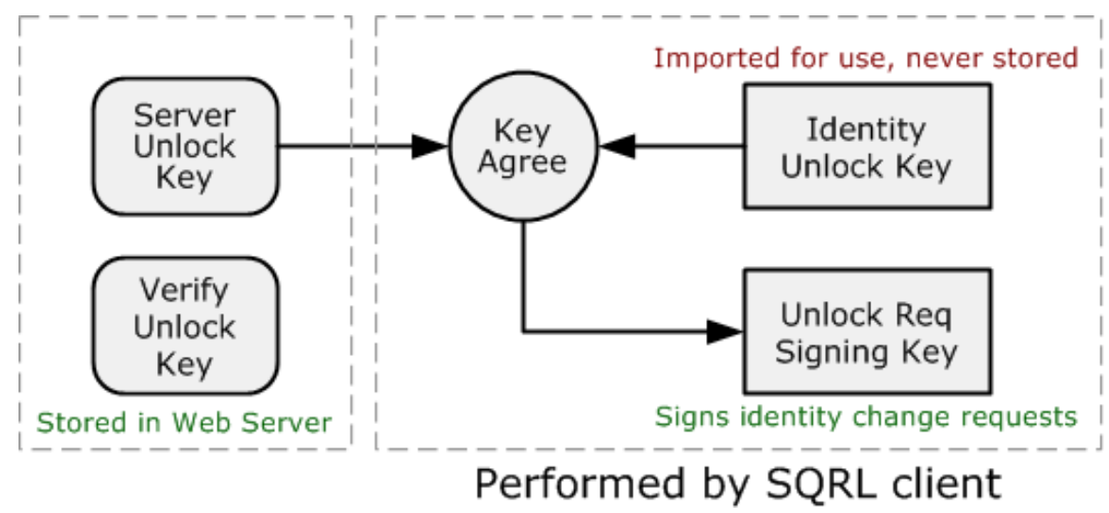
Performed during identity creation



Performed to lock identity



Performed to unlock identity



We use Diffie-Hellman Key Agreement:

Given public key one (PubKey1) and secret key one (SecKey1)
and public key two (PubKey2) and secret key two (SecKey2), then:

```
DHKA(PubKey1,SecKey2)  =  DHKA(PubKey2,SecKey1)
```

Our lock key, IdentityLock, is stored in the client. The unlock key, IdentityUnlock, is offline.

```
IdentityLock := MakePublic(IdentityUnlock)
```

Generate a random unlock key for this server.

```
ServerUnlock := MakePublic(RandomLock)
```

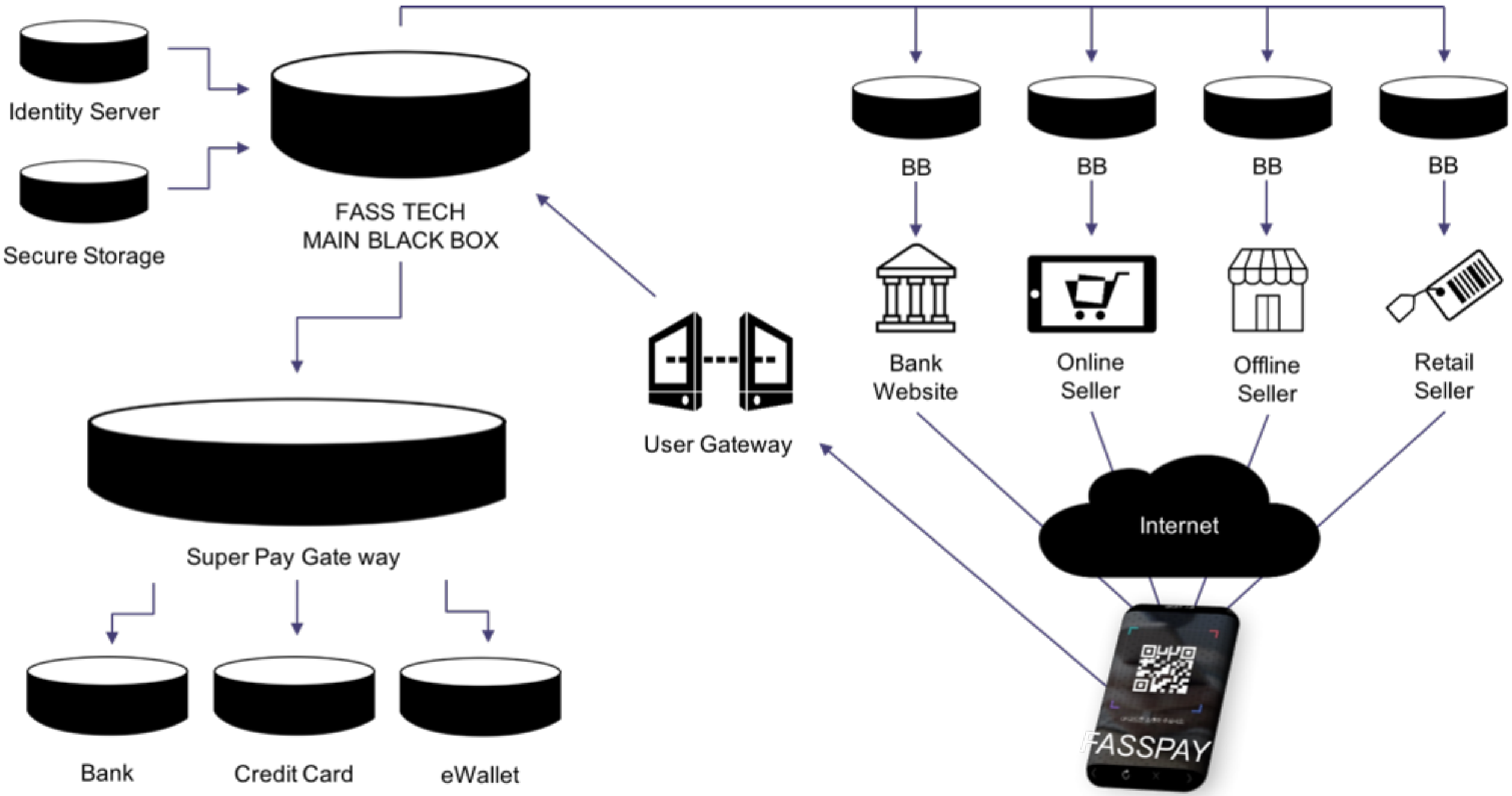
Use this to also create a VerifyUnlock key.

```
VerifyUnlock := MakePublic( DHKA(IdentityLock,RandomLock))
```

The server stores both ServerUnlock and VerifyUnlock. The client discards RandomLock.
To unlock or rekey the account, server sends ServerUnlock to the client. Client uses this, in combination with his IdentityUnlock key to generate the secret key:

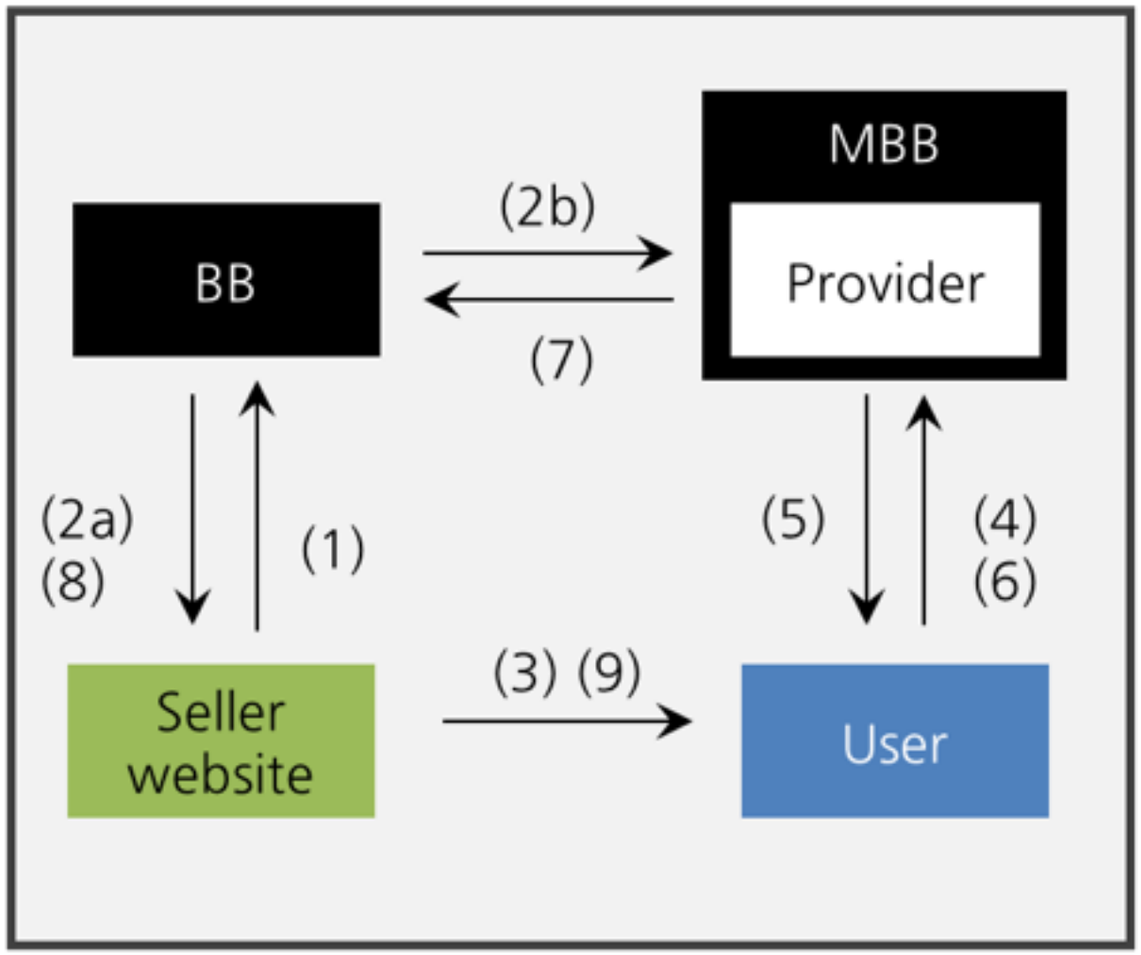
```
DHKA(IdentityLock,RandomLock) = DHKA(ServerUnlock,IdentityUnlock)
```

This is used to sign the unlock/rekey message to the server. At the server, the signature must match the associated public key, VerifyUnlock.



FASSKey System Architecture

Protocol flow for Shopping



V (1). Seller website gets qlink from BB.

V (2a). BB sends qlink to Seller website.

V (2b). BB registers qlink with MBB.

A (3). User gets qlink from Seller website.

T (4). User sends qlink to Provider(MBB), proving Seller website.

T (5). MBB confirms qlink and sends payment query to User.

T (6). User confirms purchase and payment method.

V (7). MBB sends purchase confirm to BB.

V (8). BB confirms purchase to Seller website.

A (9). Seller website updates to show paid.

V - VPN connection

T - TLE protocol

A - Visual QR Code or NFC

$$\left.\begin{aligned} k1 &= \text{DHKA}(\text{SEPK}, \text{SSSK}) \\ k2 &= \text{DHKA}(\text{SDPK}, \text{CESK}) \\ K &= \text{DHKA}(k1, k2) \end{aligned}\right\} \quad \text{Client Session Key}$$

- Client knows: Site Specific Secret Key (SSSK)
- From qlink learns: Server Ephemeral Public Key (SEPK)
- From MBB learns: Server Domain Public Key (SDPK)
- Creates: Client Ephemeral Secret Key (CESK)

- Server knows: Server Domain Secret Key (SDSK) and Server Ephemeral Secret Key (SESK)
- From Client learns: Client Ephemeral Public Key (CEPK) and user's Site Specific Public Key (SSPK)

$$\left.\begin{aligned} k1 &= \text{DHKA}(\text{SESK}, \text{SSPK}) \\ k2 &= \text{DHKA}(\text{SDSK}, \text{CEPK}) \\ K &= \text{DHKA}(k1, k2) \end{aligned}\right\} \quad \text{Server Session Key}$$

Payment methods

- Integrated eWallet system
- Shopping payment methods
- P2P payments
- Store POS sales
- Casual POS sales
- Banking
- ATM access

All payments methods can use pure eWallet system.
No bank account, no credit card necessary.

Questions?