SEPIOR

Practical Threshold Cryptography for Cloud and Cryptocurrencies

Jakob I. Pagter

From sugar beets to Threshold Cryptography

2009

Damgård et al., Financial Crypto ´09: "Multi-Party Computation Goes Live"

2008

PARTISIA

Multiple commercial applications of MPC

- Sugar Beet auction
- Off-exchange matching (Tora)
- Privacy-friendly demographic profiling (Insights Network)
- …

2015

- "Secure Dropbox"

↓

- "KMaaS"
- EU funding 2016-17

2018

Crypto Wallet Security
Cloud Data Security
Blockchain Data Security
SEPIOR Threshold Cryptography

- Series A in 2018
- VCTRADE (SBI)
- Cloud, blockchain and cryptocurrencies

SBI BITS

The landscape for Threshold Cryptography (TC)

Incumbents

- HSM – Hardware Security Modules
  - E.g. Thales, Utimaco
- TEE – Trusted Execution Environments
  - E.g. SGX
- CSP offered KMS
  - E.g. AWS KMS

Building blocks

- MPC protocols
  - AES
  - ECDSA
  - General
- MPC frameworks for general MPC
  - libscapi
  - EMP toolkit
  - …

Sepior KMaaS (Key Management as-a-Service)

Core contributions

- Example architecture for TC-based KMaaS
- TC-friendly stream-cipher
- Experiences with commercialization

Design philosophy

- Easy integration [Developer]
- Simple administration [Security Officer]
- Transparent [End-user]

```
SepiorServicesClientConfiguration sepiorConfig = SepiorUtils.getConfigurationFromFile(Paths.get(sepiorServicesClientConfigurationFile));
SepiorServicesClient sepiorClient = SepiorUtils.getSepiorServicesClient(sepiorConfig);
AWSCredentials awsCredentials = new PropertiesFileCredentialsProvider(amazonS3ConfigurationFile).getCredentials();

/* Create Sepior S3 encryption object with the given configuration and get an AWS client. */
s3Enc = new SepiorS3Encryption(sepiorClient, awsCredentials);

private static void uploadUsingAmazonSDK(String s3Bucket, String s3Key, String filename) throws SepiorServiceException, SepiorUserException {
  AmazonS3 awsClient = s3Enc.getAmazonS3();
  PutObjectRequest put = s3Enc.getPutObjectRequest(s3Bucket, s3Key, new File(filename));
  awsClient.putObject(put);
}

private static void downloadUsingAmazonSDK(String s3Bucket, String s3Key, String filename) throws IOException {
  AmazonS3 awsClient = s3Enc.getAmazonS3();
  GetObjectRequest get = new GetObjectRequest(s3Bucket, s3Key);
  S3Object s3Object = awsClient.getObject(get);
  try (InputStream in = s3Object.getObjectContent()) {
    Files.copy(in, Paths.get(filename));
  }
}
```

Entire AWS S3 integration

Challenges in cloud encryption

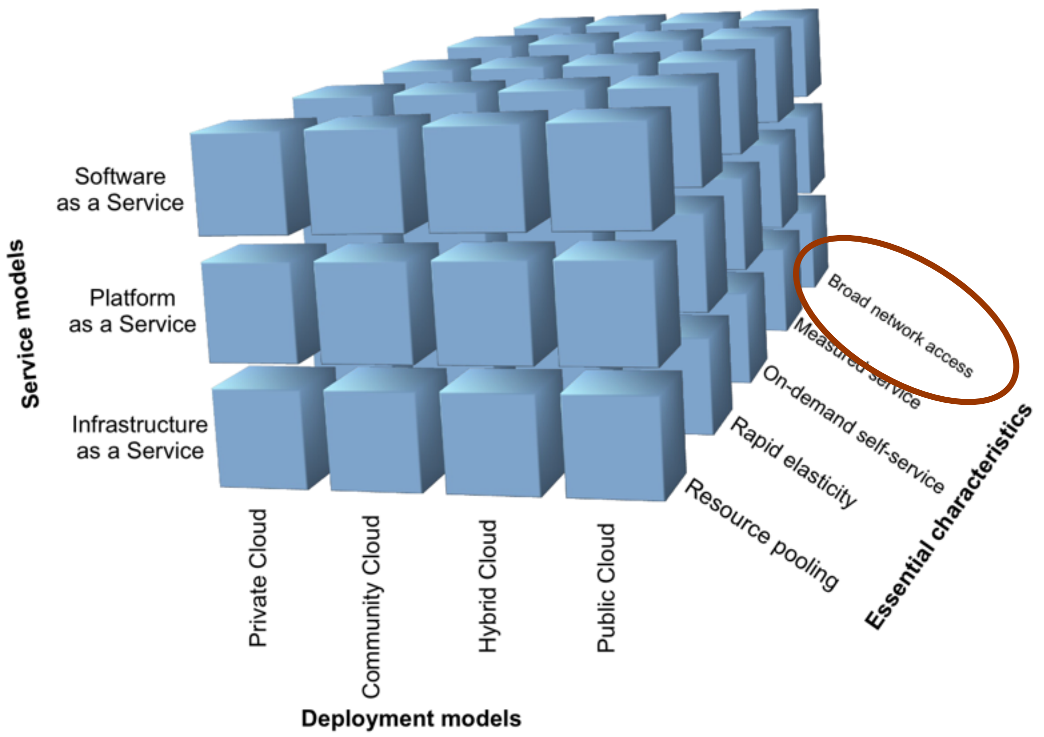
Security perspective (*-YOK)

Randomness

Storage of key

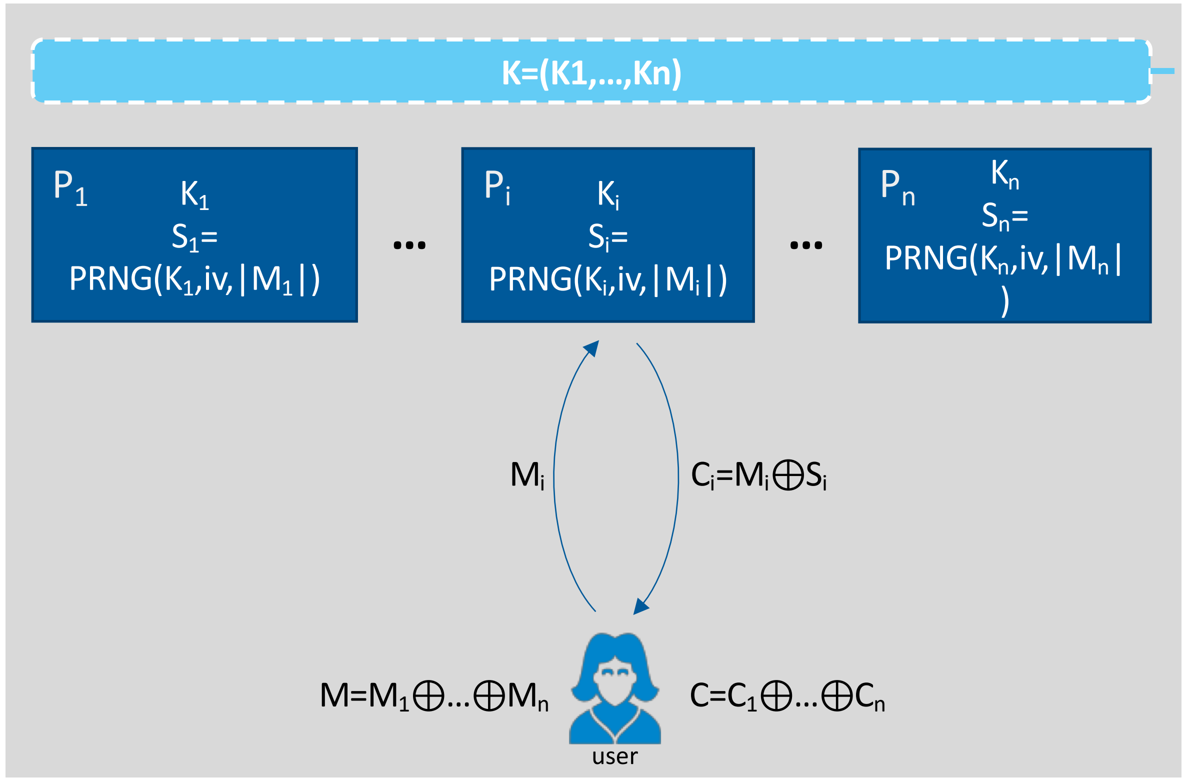
Usage of key

Business perspective



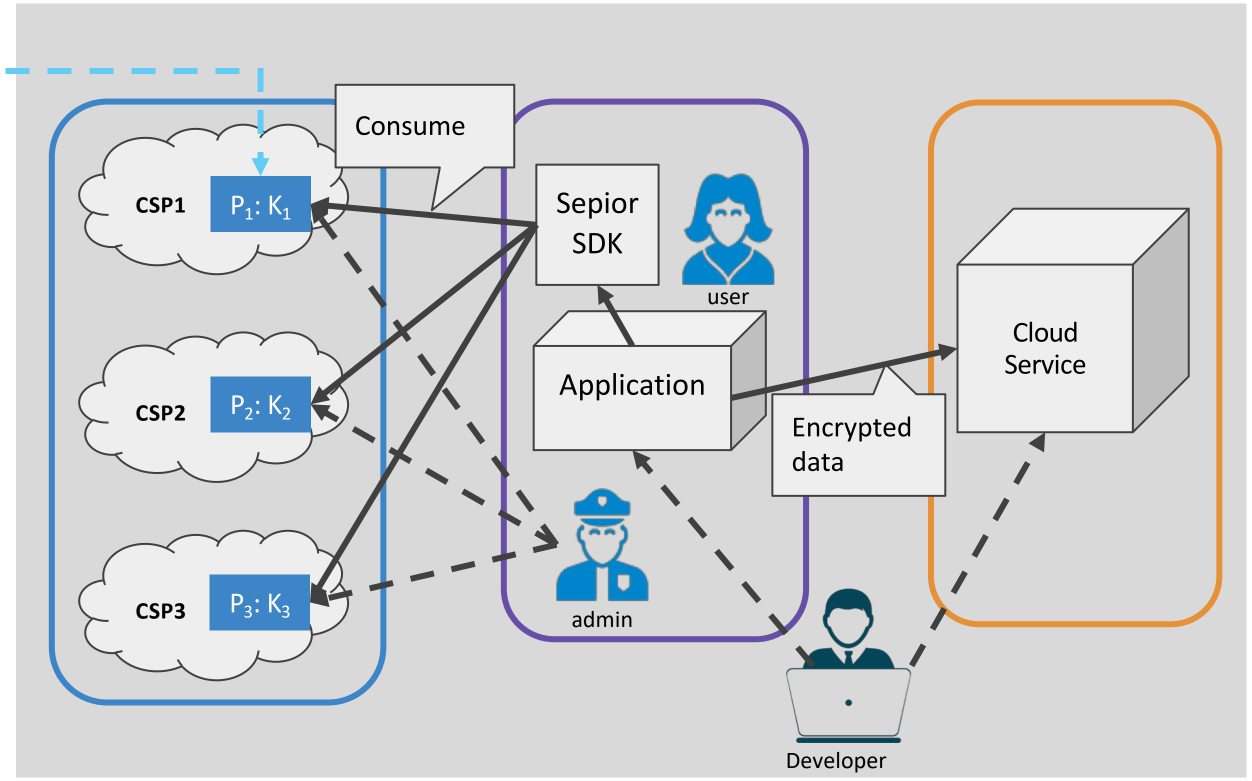
TC-based KMaaS

TC-friendly stream cipher



K=(K1,…,Kn)
P_1 K_1 S_1= PRNG(K_1,iv,|M_1|)
…
P_i K_i S_i= PRNG(K_i,iv,|M_i|)
…
P_n K_n S_n= PRNG(K_n,iv,|M_n|)
M_i
$C_i=M_i \oplus S_i$
$M=M_1 \oplus … \oplus M_n$
$C=C_1 \oplus … \oplus C_n$
user

Architecture



Experiences

Positives

- 👍 People ”get it”
- 👍 Ease-of-use
- 👍 Price

Negatives

- 👎 Lack of standards
- 👎 Lack of certification
- 👎 First-mover reluctance

Sepior ThresholdSig

Core contributions

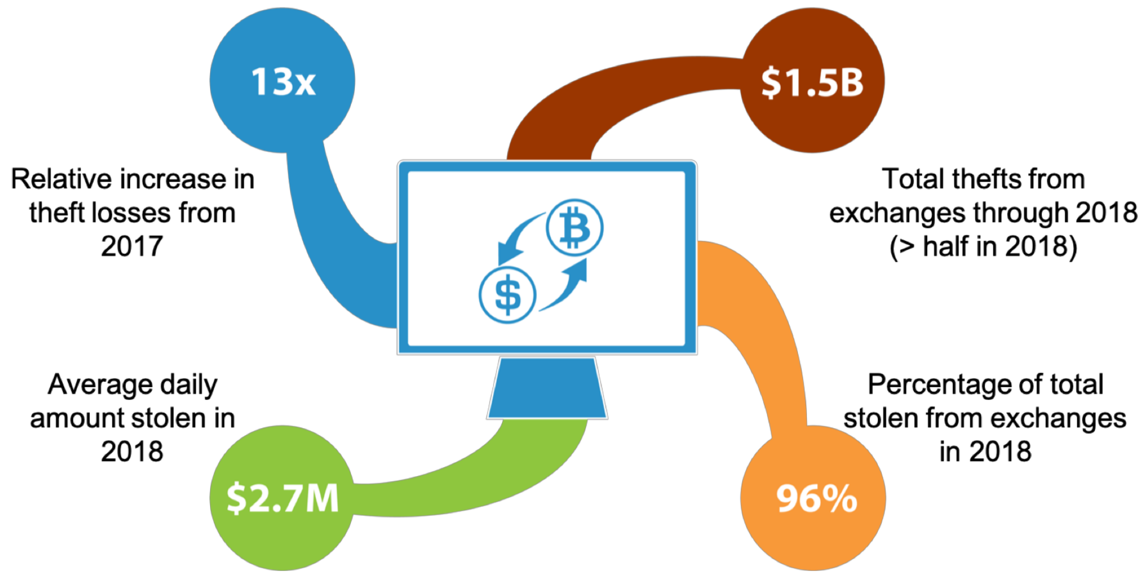
- ECDSA protocols in TC-setting
- Architectures for deployment
- Additional relevant features

Design philosophy

- Flexible components
- Simple deployment
- “Forget you have a hammer”

Challenges in cryptocurrency (wallet) security

Lose key = lose money 😲



MultiSig not enough

- Privacy
- Transaction cost
- Flexibility

See e.g. Gennaro et al. (eprint.iacr.org/2016/013.pdf).

Or,

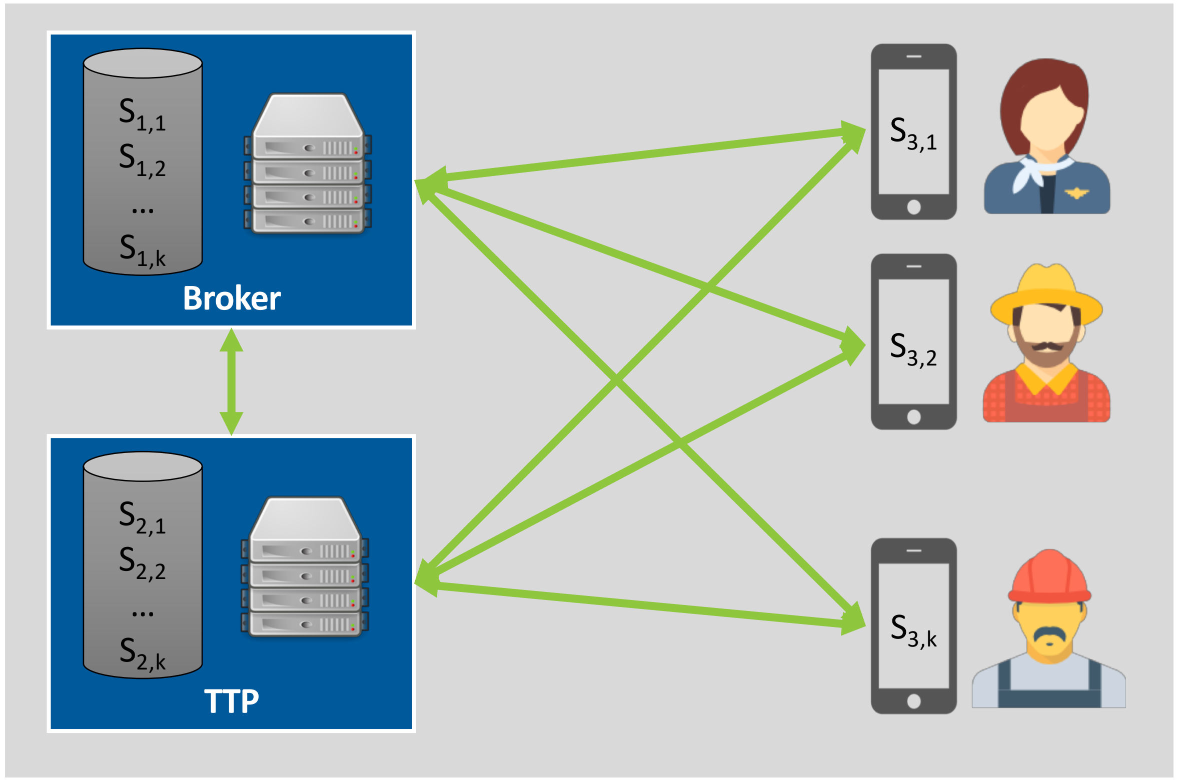
- Onchain vs. off-chain

TC-based ECDSA solution

Cipher(s)

- (t/n)-ECDSA ciphers
  - Honest and dishonest
  - Active security
  - Based on security of ECDSA
- 1500 signatures/second on with each party running one server
- Allow abort
  - Manual intervention if need be
- Use preprocessing
  - (time above includes time for preprocessing)

Architecture



Experiences

Positives

- 👍 Good alignment with decentral trust model
- 👍 Much untapped potential

Negatives

- 👎 Many cryptocurrencies use ciphers which are not TC-friendly
- 👎 General lack of awareness of TC potential

Summary

- Two commercial use case
- Obstacles from being new technology
- Vast potential

Thank You!