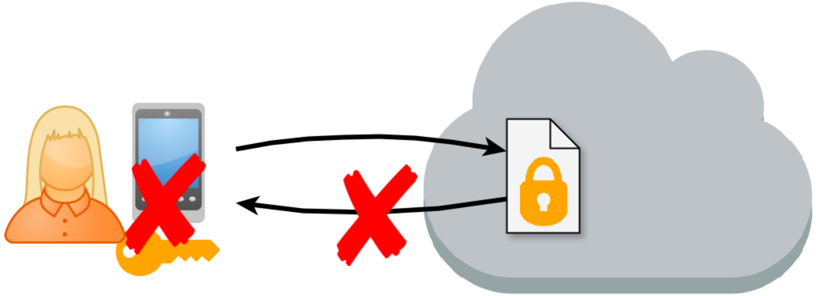
Cloud Data Sharing and Device-Loss Recovery with Hardware-Bound Keys

Felix Hörandner
Graz University of Technology
Graz, Austria

Franco Nieddu
Graz University of Technology
Graz, Austria

December 19, 2019

Motivation



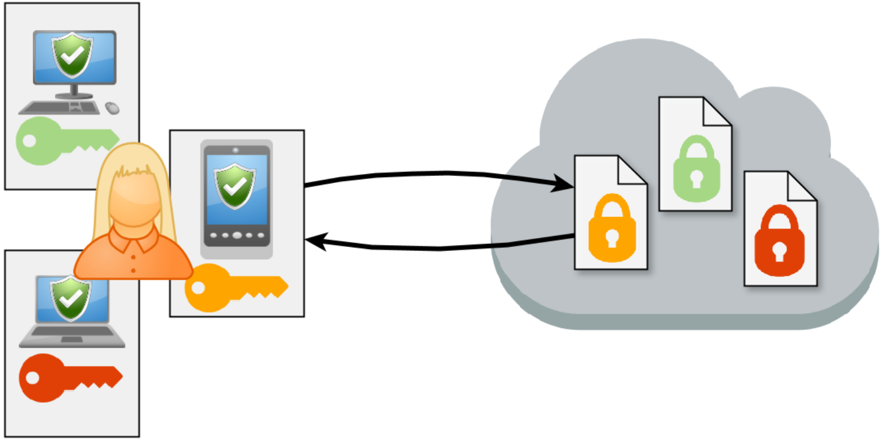
What if device is lost or stolen?

- Can't access data without key
- ➢ Need to recover from key loss

Traditional approaches

- Backup on flash drive?
- Sheet with QR code?
- Password-encrypted key at cloud storage?
- Secret Sharing?

Motivation

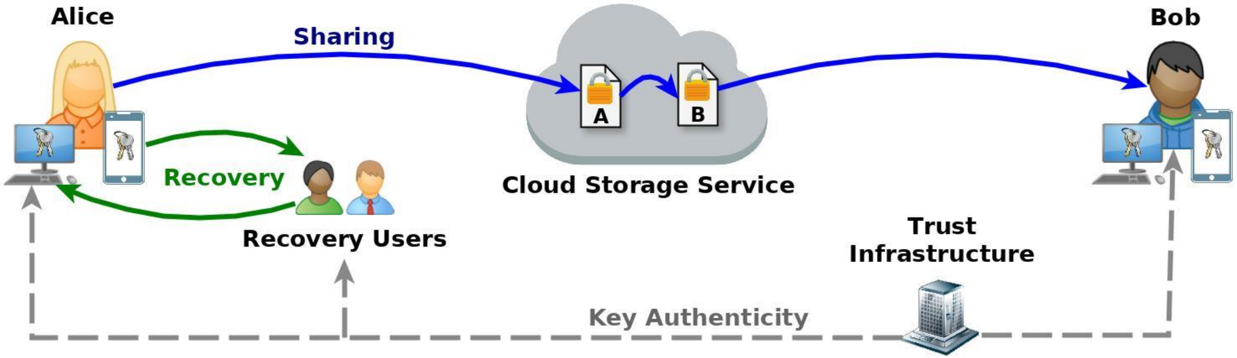


- Multiple devices per user
  - Shared Key?
  - Individual Keys?
  - Keys bound to device?
- Challenges
  - Full functionality on each device
  - Recovery with hardware-bound keys
  - Recovery if only one device

Ambition and System Model

Data Sharing in the Cloud with:

Multiple Devices per User + Hardware-bound Keys + Recovery after Loss of Device/Key

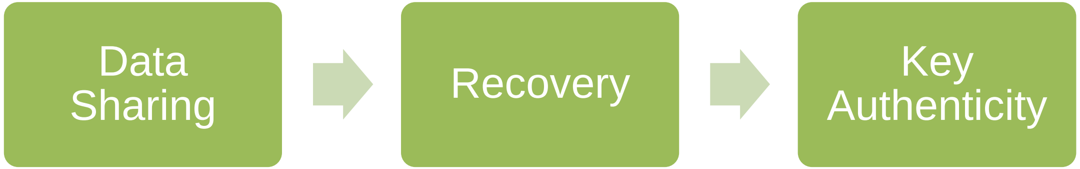


Our Contribution

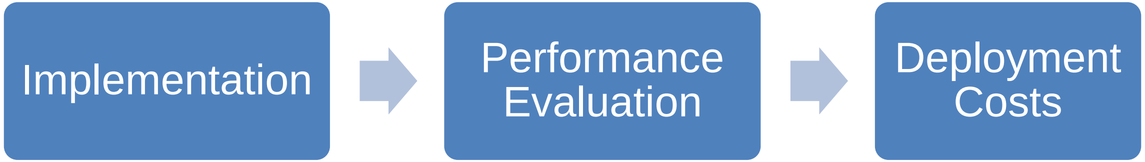
Data Sharing in the Cloud with:

Multiple Devices per User + Hardware-bound Keys + Recovery after Loss of Device/Key

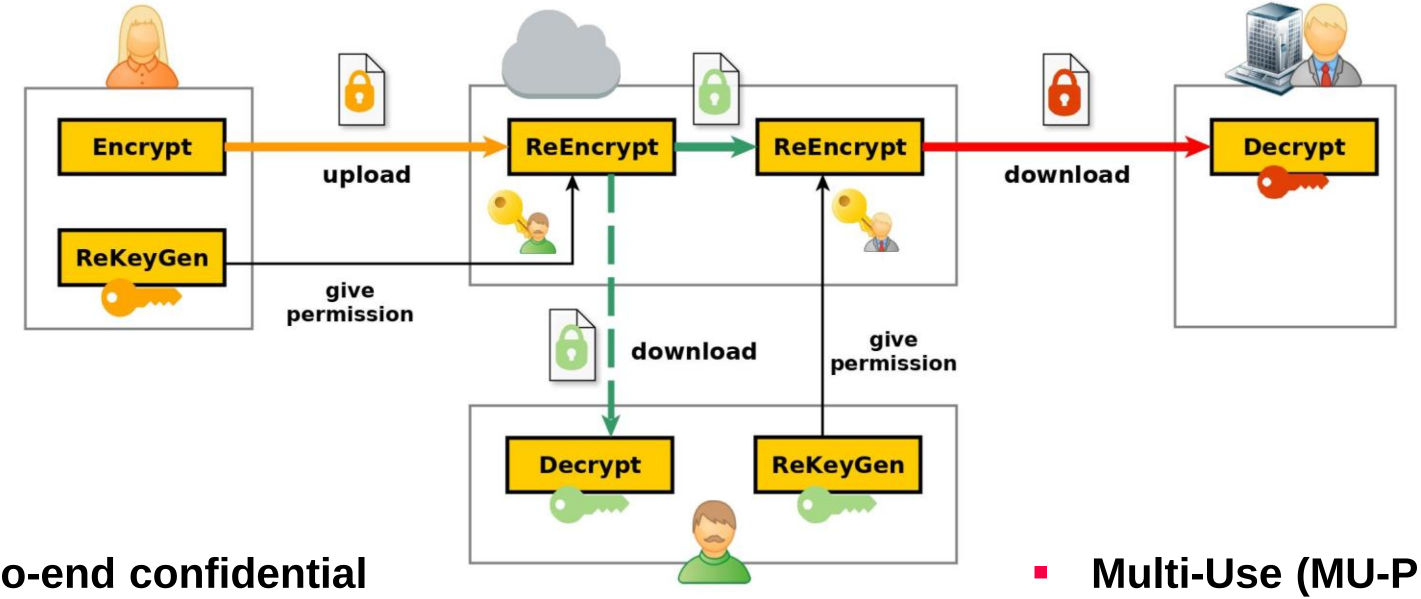
Concept:



Implementation and Evaluation:



Background: Proxy Re-Encryption (PRE) [AFGH06]



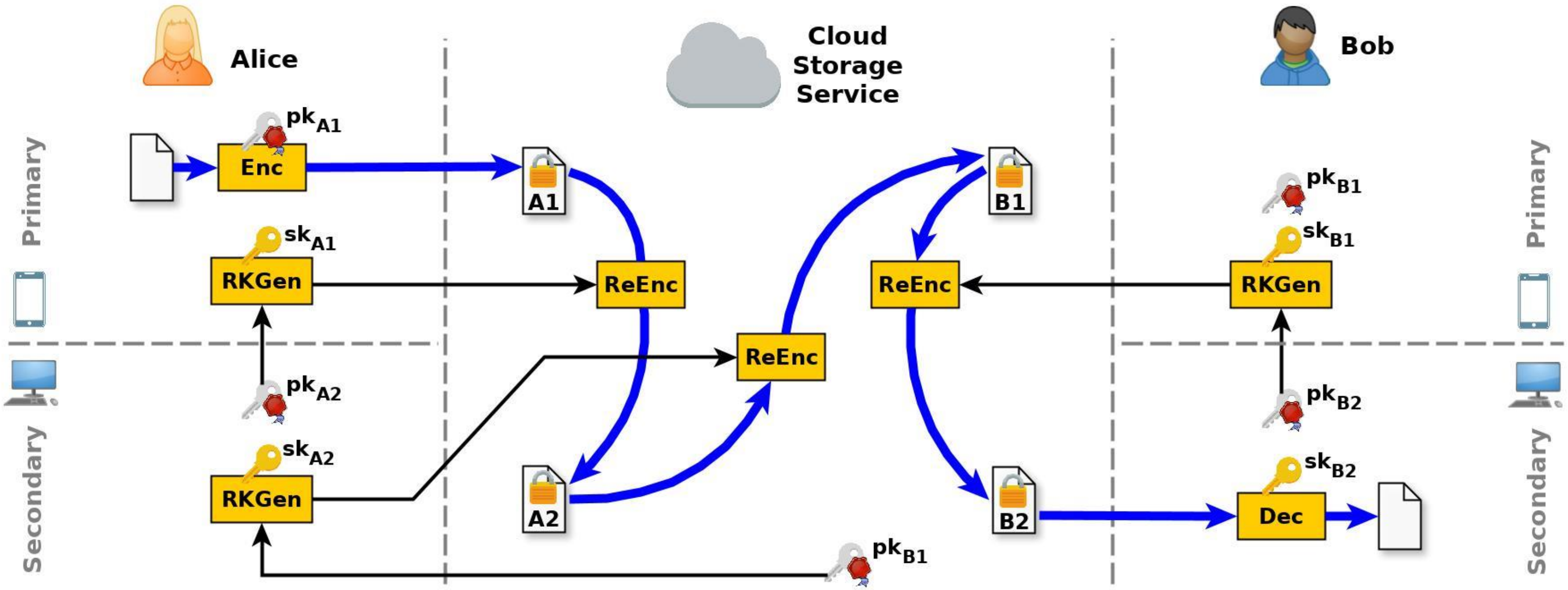
- **End-to-end confidential**
- User: **no need to fully trust proxy**
- **Control**: through re-encryption key
- No duplicate data

- **Multi-Use (MU-PRE):** [CL14] Re-Encrypt multiple times

[AFGH06] Ateniese G., Fu K., Green M., Hohenberger S.: ACM Trans. Inf. Syst. Secur. 2006 Improved proxy re-encryption schemes with applications to secure distributed storage.

[CL14] Cai Y. and Liu X.: A Multi-Use CCA-Secure Proxy Re-Encryption Scheme. IEEE DASC 2014

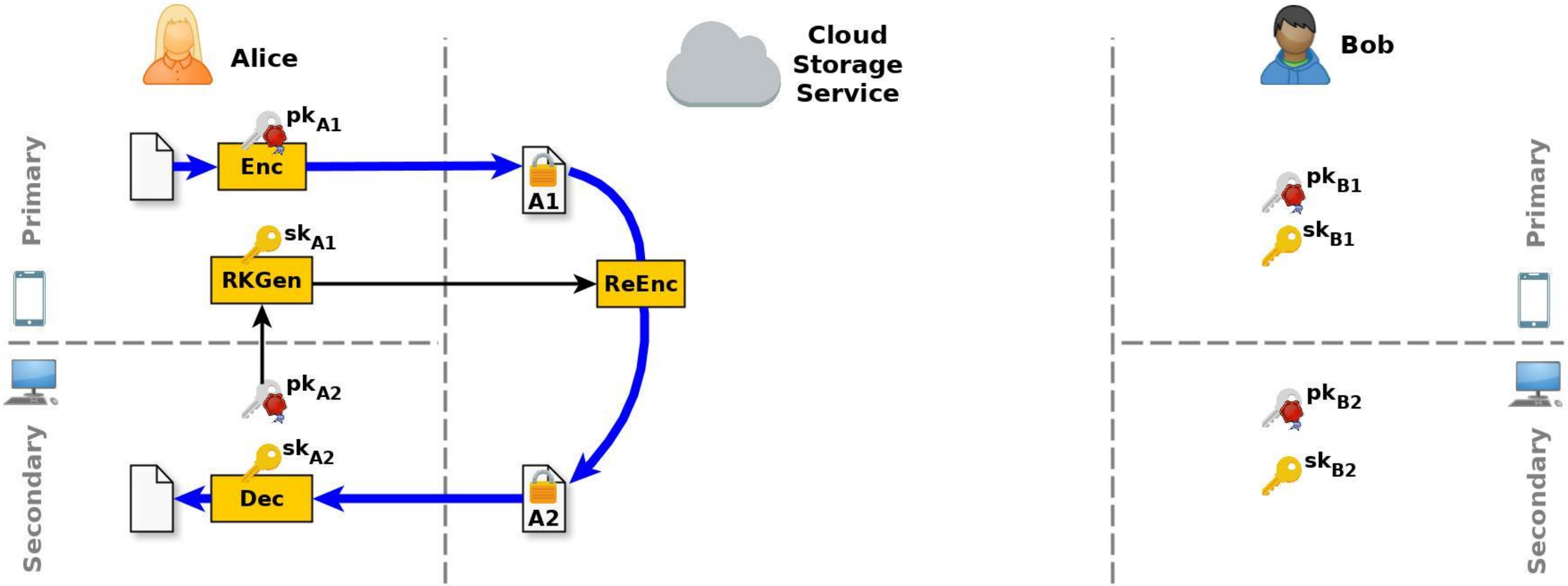
Data Sharing



- **Encrypt**: always for primary
- **Share**: always to primary

➢ Upload, access and sharing: with **any device**

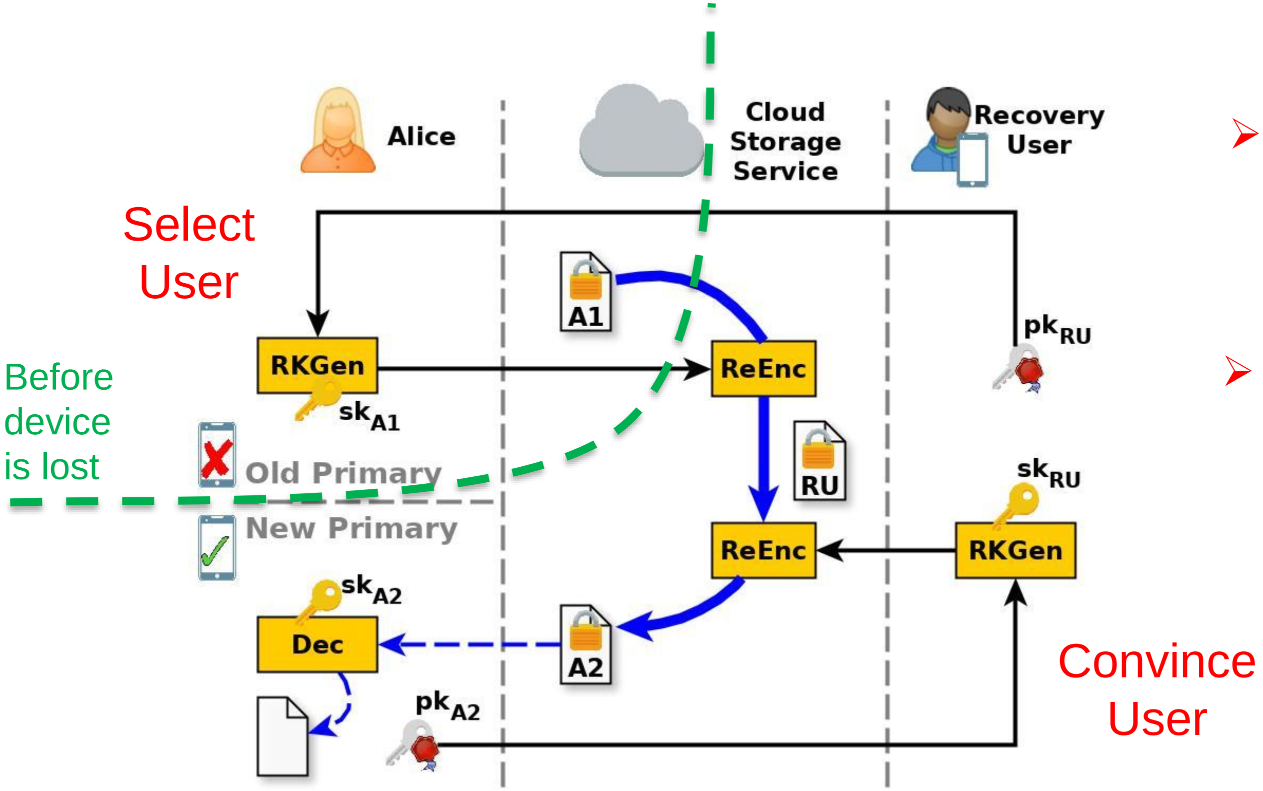
Recovery: with Secondary Device



Secondary breaks: add new secondary
Primary breaks: secondary becomes new primary

- **With secondary device:** Simple recovery

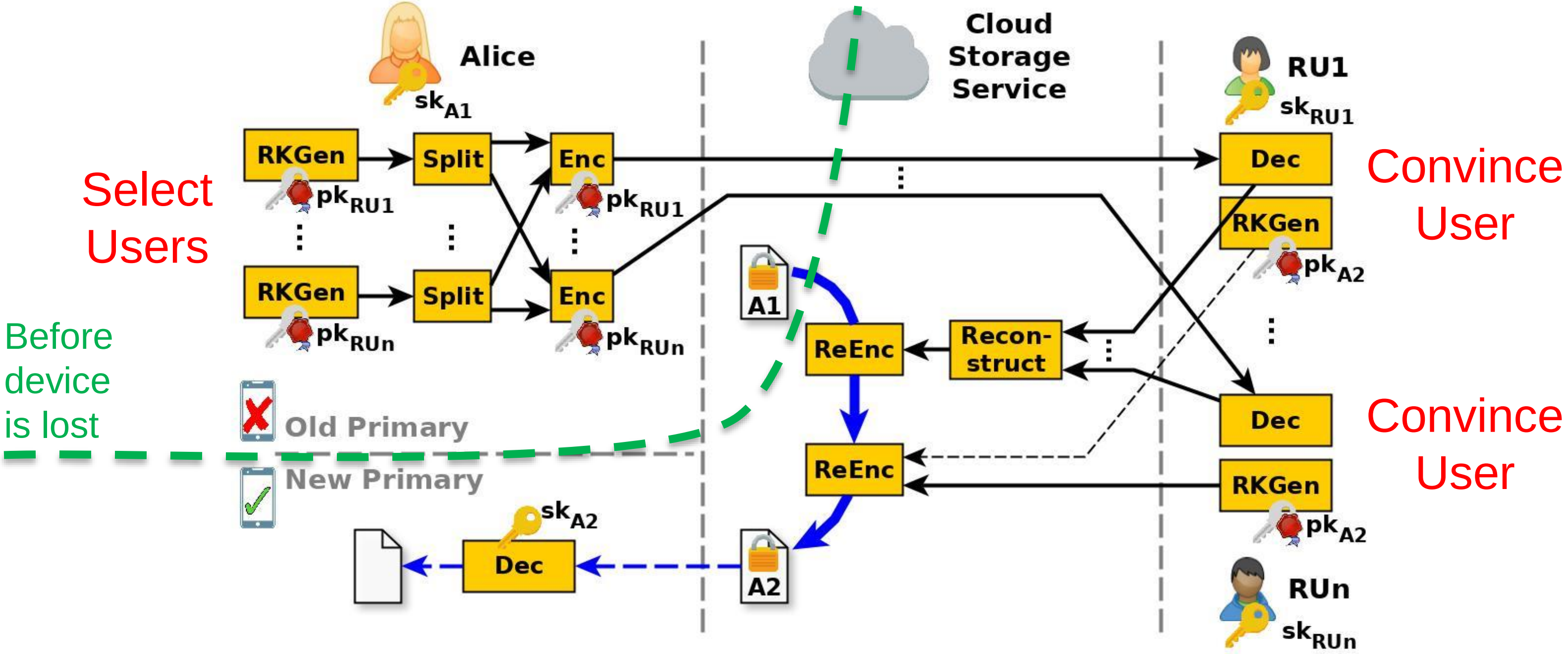
Recovery: without Secondary Device



- ➢ **Rely on Recover User:** Authenticates new device
- ➢ **Split trust:** Recovery user & cloud sharing service

Recovery: with Multiple Recovery Users

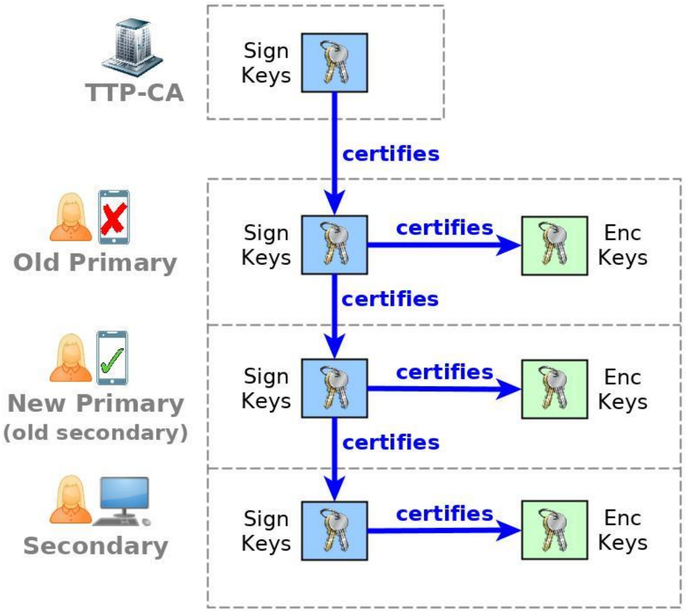
Don’t want to rely on one recovery user? Still available and willing?



- **Threshold of Recovery Users:**
  Trade-off availability vs. confidentiality

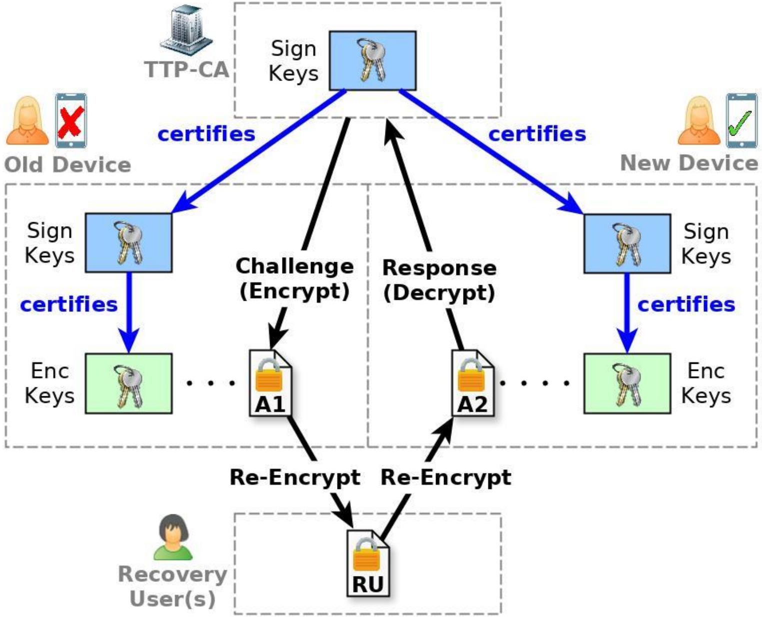
Key Authenticity: Right key to use for Encrypt and ReKeyGen?

With Secondary Device:



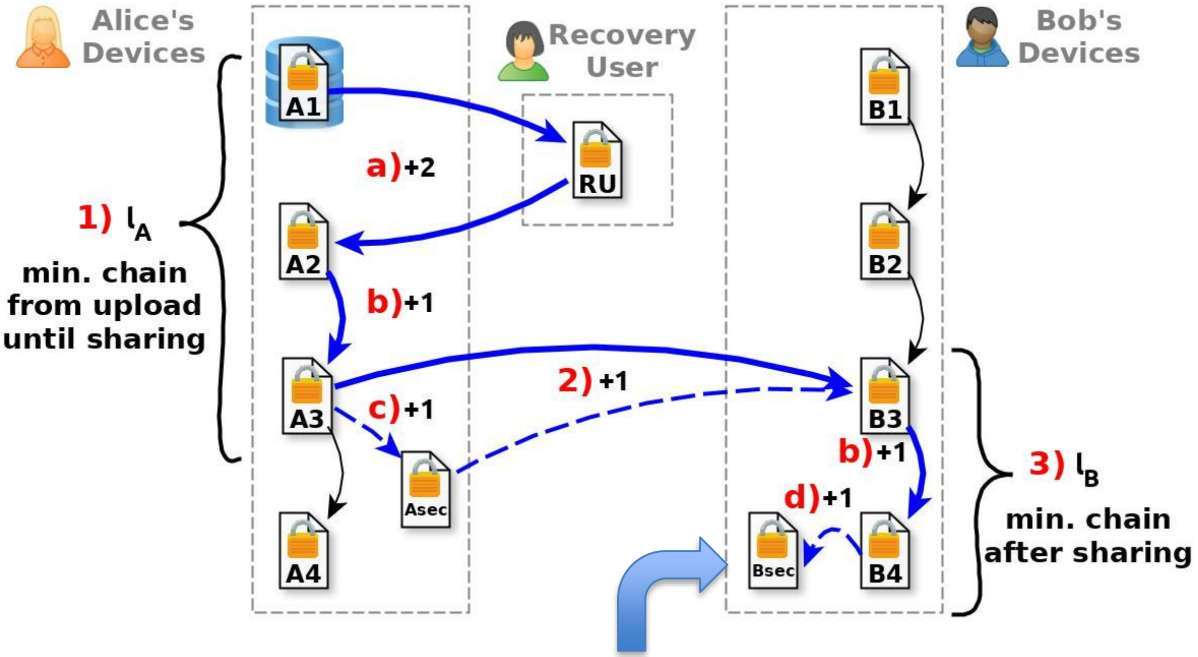
- Build certificate chain

Without Secondary Device:



- Authentication of new device:
  via decryption rights given by rec. users

Data Sharing after Recovery



Access here

1) Chain at data owner
2) Sharing
3) Chain at receiver

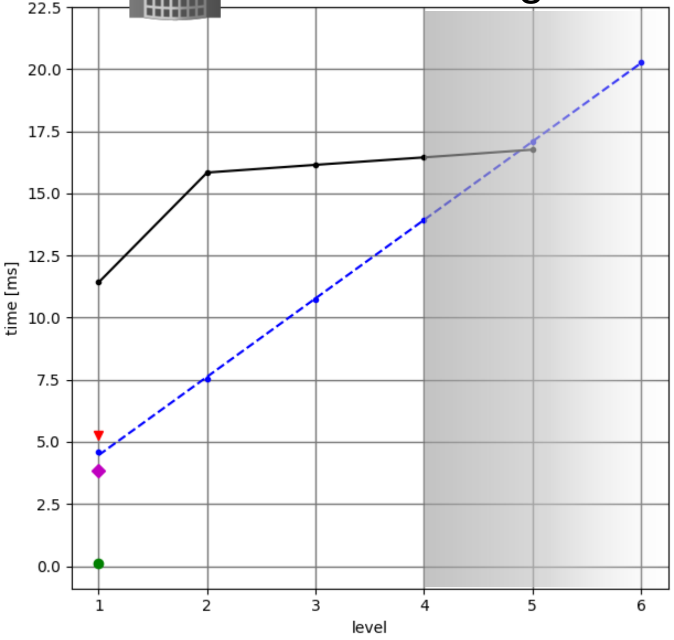
a) Recovery via other user
b) Recovery via sec. device
c) Sharing with sec. device
d) Access from sec. device

- ➢ Length of re-encryption chain grows
- ➢ Performance optimizations

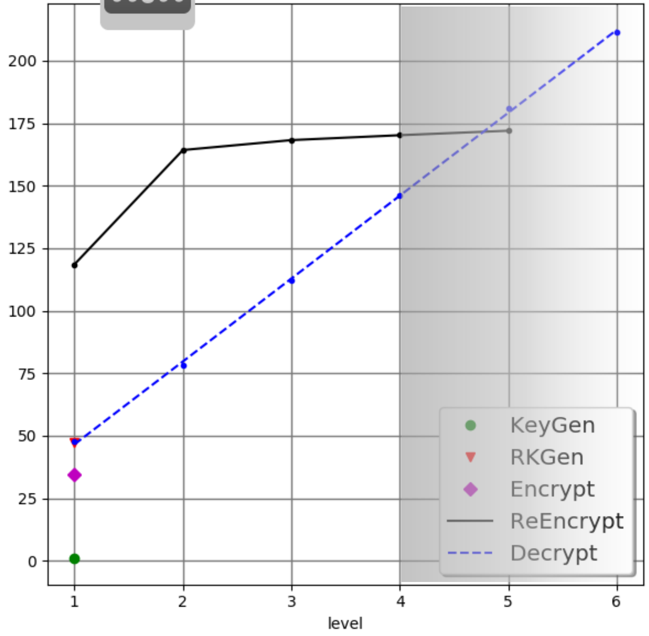
Implementation and Evaluation

MU-PRE scheme [CL14], RELIC toolkit, 128bit security, sharing AES keys

Cloud Server:
AWS c5.xlarge



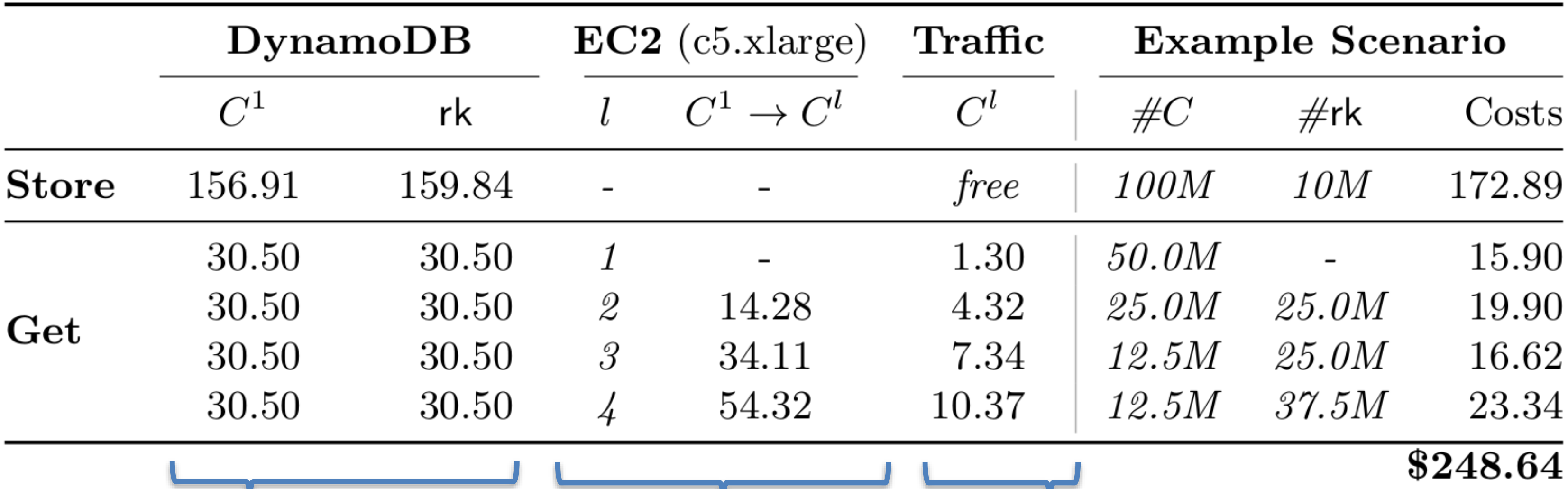
Mobile phone:
OnePlus 6T



[CL14] Cai Y. and Liu X.: A Multi-Use CCA-Secure Proxy Re-Encryption Scheme. IEEE DASC 2014

Deployment Costs (on Amazon Web Services)

Additional costs to employ our cryptography, for 100M items in $



	DynamoDB		EC2 (c5.xlarge)		Traffic	Example Scenario		
	C^1	rk	l	$C^1 \rightarrow C^l$	C^l	$\#C$	#rk	Costs
Store	156.91	159.84	-	-	*free*	*100M*	*10M*	172.89
Get	30.50	30.50	*1*	-	1.30	*50.0M*	-	15.90
	30.50	30.50	*2*	14.28	4.32	*25.0M*	*25.0M*	19.90
	30.50	30.50	*3*	34.11	7.34	*12.5M*	*25.0M*	16.62
	30.50	30.50	*4*	54.32	10.37	*12.5M*	*37.5M*	23.34
								$248.64

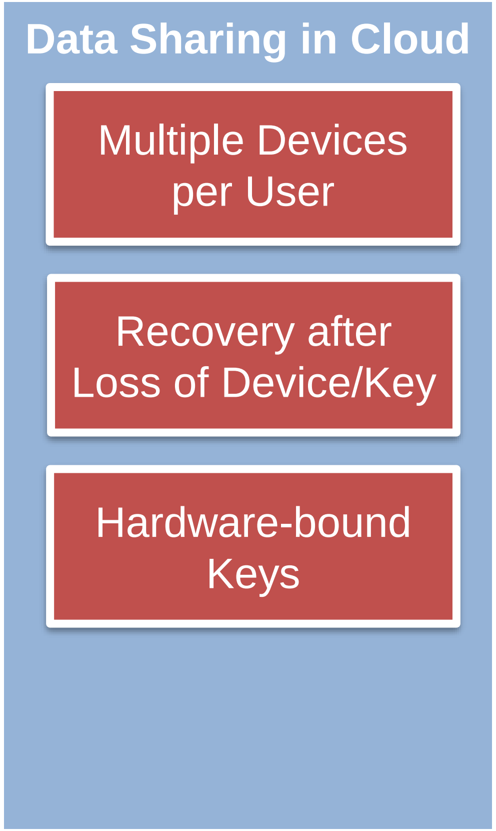
Store:
$1.525/1M requests
$0.306/1GB-month
Get:
$0.305/1kB/1M requests

Get:
$0.194/h
2.15 scaling factor

Get:
$0.09/1GB

- For 100M up- and downloads

Summary: Key Messages



Data Sharing in Cloud

Multiple Devices per User

Recovery after Loss of Device/Key

Hardware-bound Keys

- **Data Sharing in the Cloud**
  - Support for multiple devices per user
  - Key authenticity
- **Recovery**
  - For multi-device users: simple
  - For single-device users: supported by relying on recovery users
  - Threshold of recovery users to choose trade-off: availability vs. collusion
- **Hardware-bound keys**
  - No need to export keys: can be bound to the hardware
  - Improved key security
  - If stolen: no re-keying in user's domain
- **Performance**
  - Evaluation shows practical efficiency
  - Guideline for deployment costs

Thank you! Any Questions?