Secure Group Messaging

Alex Washburn

CSc-85030

Overview

01

What is it?

03

How does it work?

02

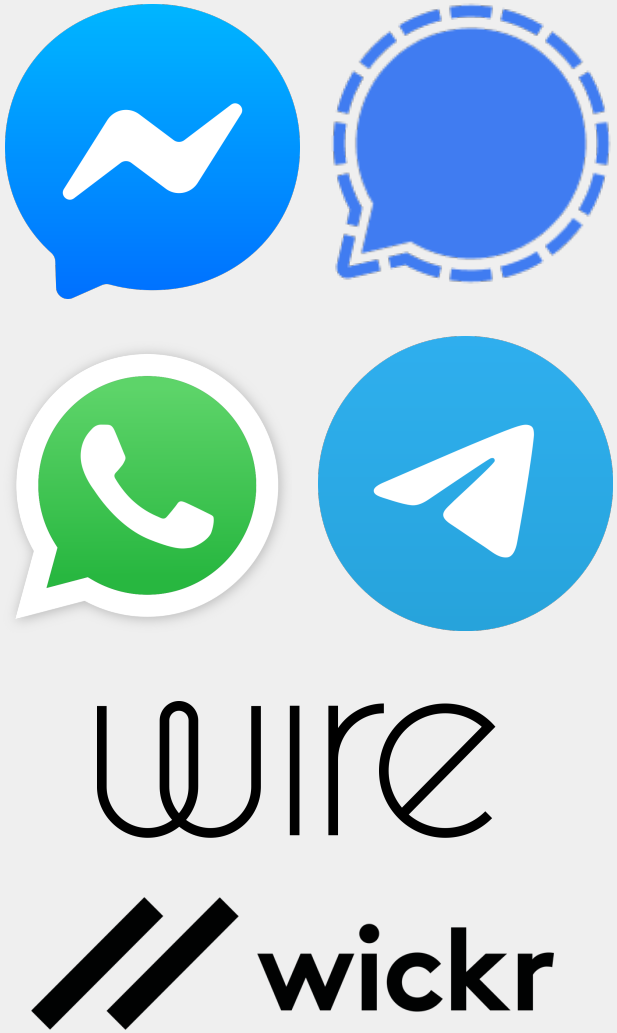
Why use it?

04

Where is it?

01

What is it?



Message Layer Security

- Internet Engineering Task Force
- Clear security guarantees
- Efficient and scalable

Security

End-to-end Encryption	Message • Authentication • Confidentiality • Integrity	Memebership Authentication
Forward Secrecy	Post-compromise Security	Deniablity

Asynchronicity

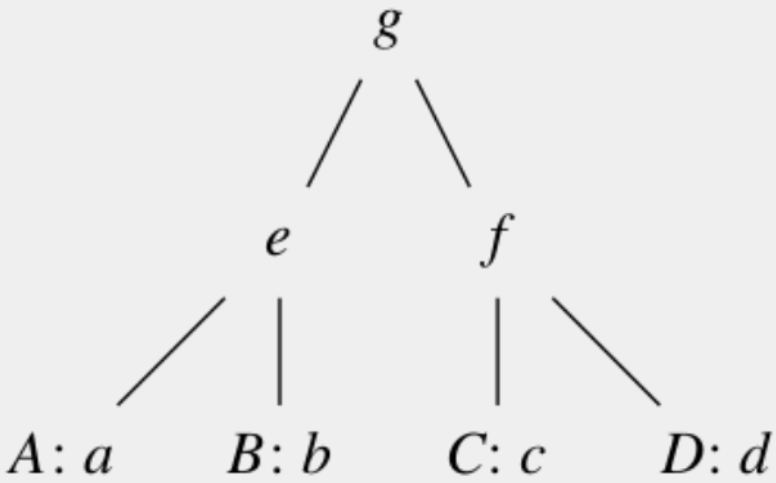
- Long-lived
- Online / offline members
- Message ordering

Scalability

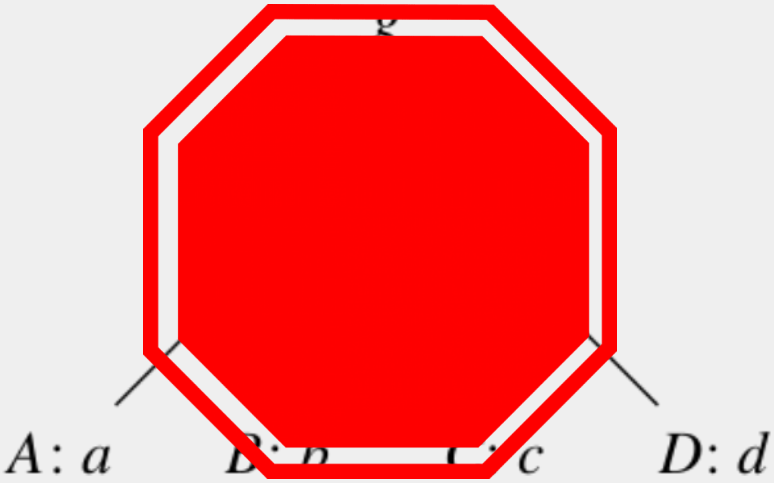
- Up to 50,000 members

TreeKEM

State of the art MLS protocol



~~IncKEM~~



Wait!

Lets review...

Abstraction Layers

SGM

General use case

MLS

Set of security and efficiency guarantees

TreeKEM

Specific protocol

02

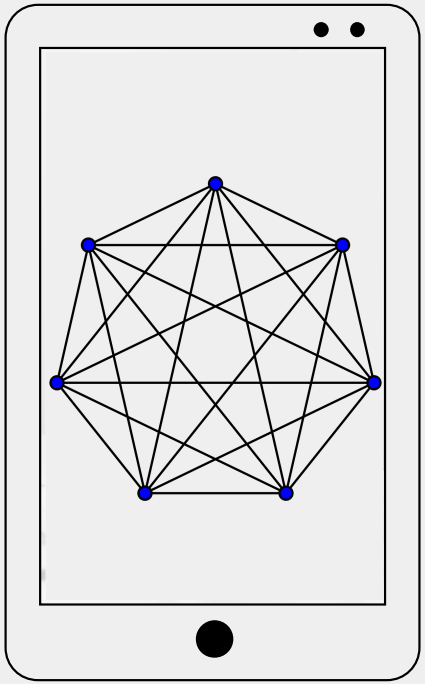
Why use it?

wire

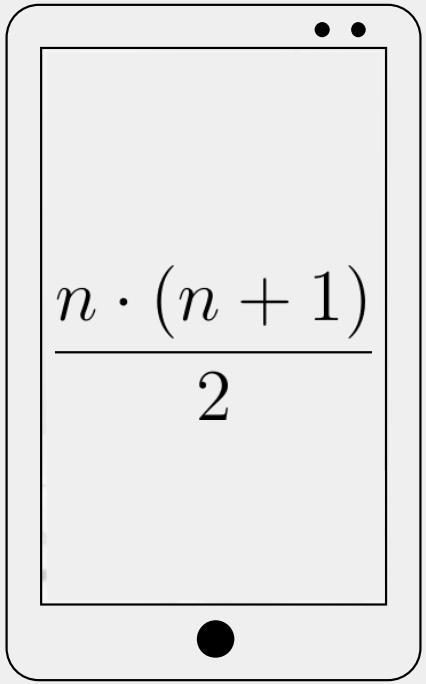
wickr

Naive Protocol

Pairwise secure messaging channels



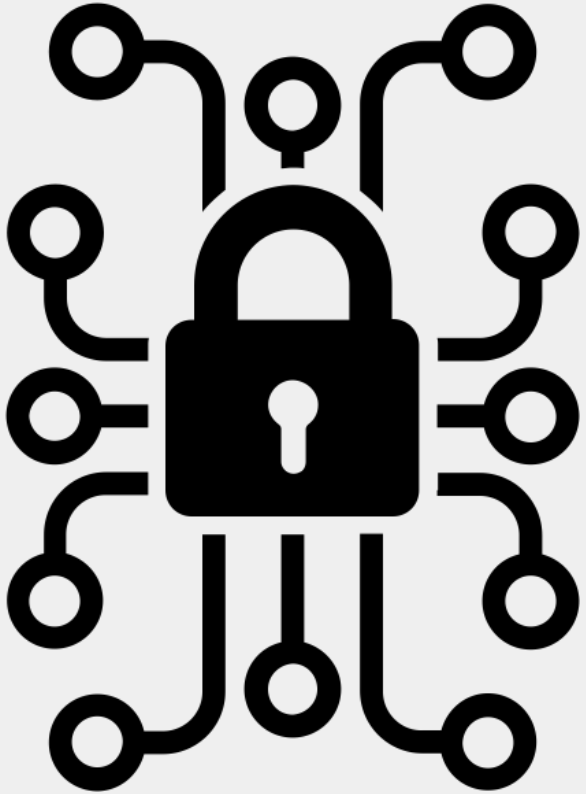
Naive Protocol



- 50K members → 1 billion channels
- Not scalable
- Not MLS

How does it work?

03



Continuous Group Key Agreement

Init

$$ID \to \Gamma$$

Outputs an initial state γ for the id

Create

$$\Gamma \times \overrightarrow{ID} \to (\Gamma, W)$$

Form a group from alist of ids and a γ

Update

$$\Gamma \to (\Gamma, T)$$

Start a new epoch, transitioning from γ to γ’

Add

$$\Gamma \times ID \to (\Gamma, W, T)$$

Add a new group member

Remove

$$\Gamma \times ID \to (\Gamma, T)$$

Remove an existing group member

Process

$$\Gamma \times T \to (\Gamma, I)$$

Consume control message to update crypto material

Continuous Group Key Agreement

Correctness

All group members output the same update secret I in update epochs.

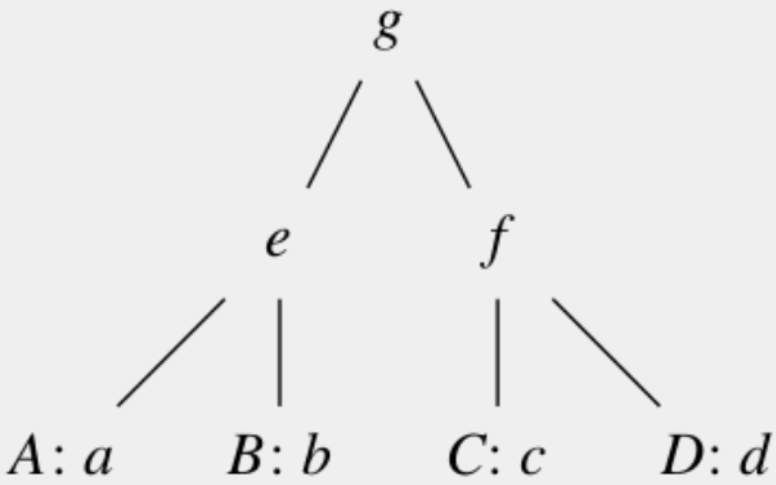
Forward Secrecy

If the state of any group member is leaked, previous update secrets remain hidden from the attacker.

Post-compromise

After every group member whose state was leaked performs an update, update secrets are unknown to the attacker.

TreeKEM



CGKA

Protocol defines operations for each of the six algorithms

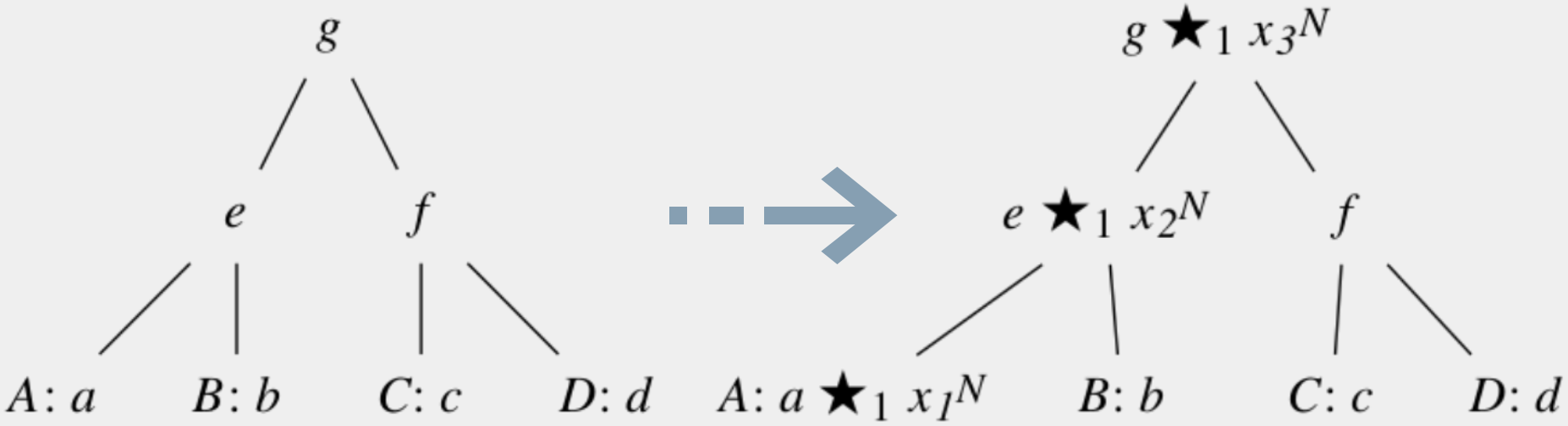
Tree-based

Facilatates efficient updates

MLS

Provides (most) required security guarantees

TreeKEM



TreeKEM

User V_0 initiates an update

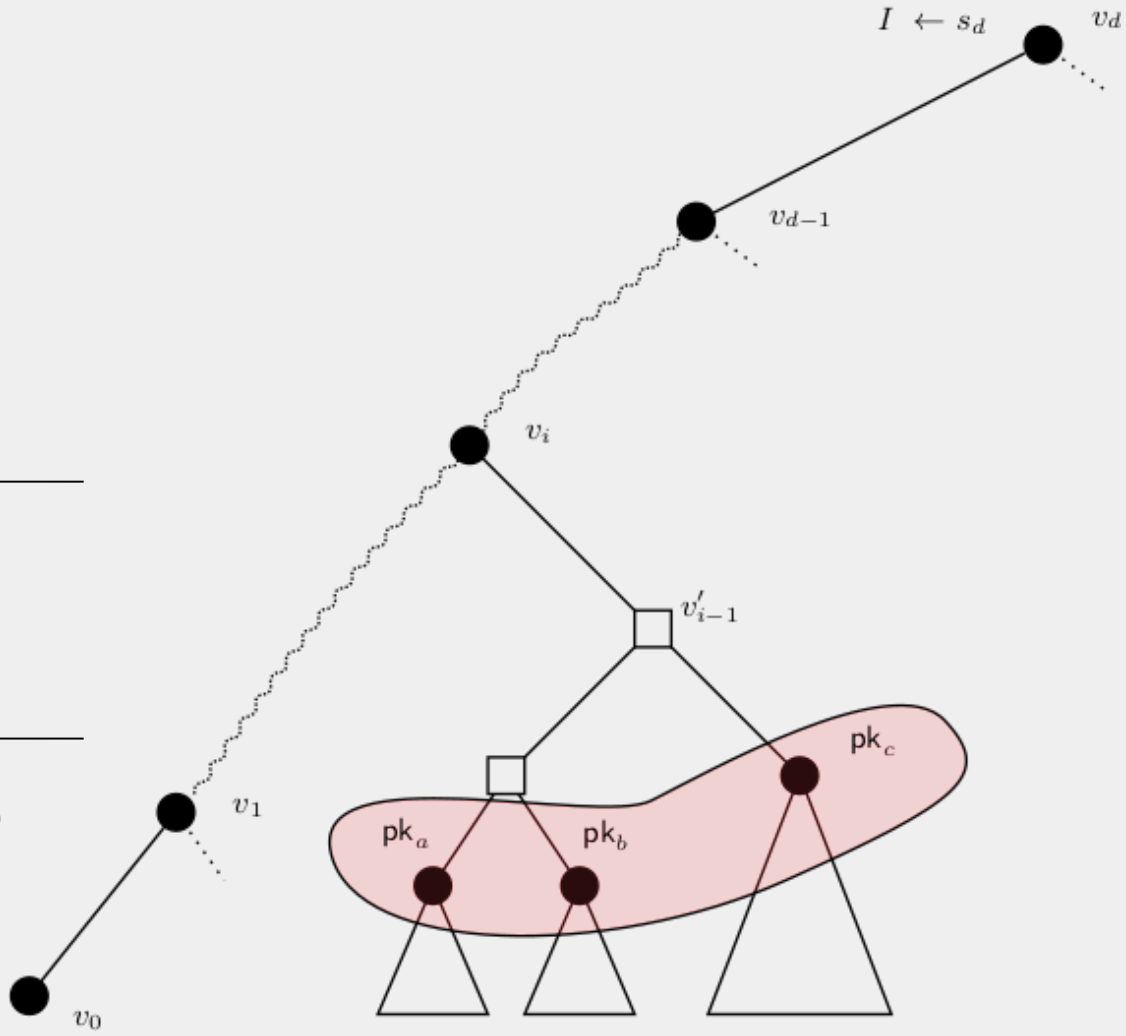
Direct path to root is updated

Users of subtree V_i process

Propogate changes from V_i to leaf

Process maintains root info

Users know root information again after process algorithm finishes



TreeKEM

User V_0 initiates an update

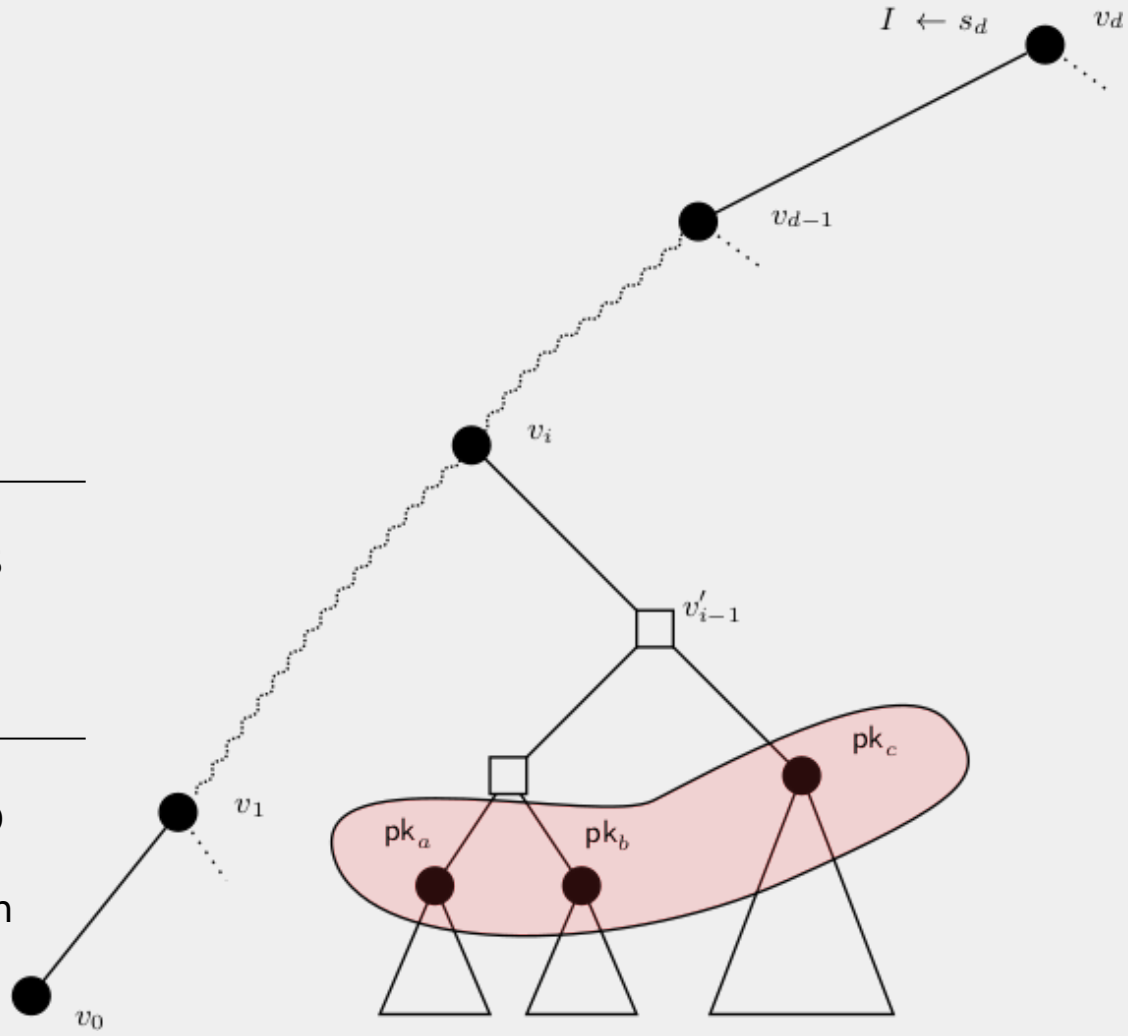
$\mathcal{O}(\log n)$ Control messages

Users of subtree V_i process

$\mathcal{O}(\log n)$ Changes from V_i

Process maintains root info

$\mathcal{O}(\log n)$ Time for new epoch



TreeKEM

User V_0 initiates an update

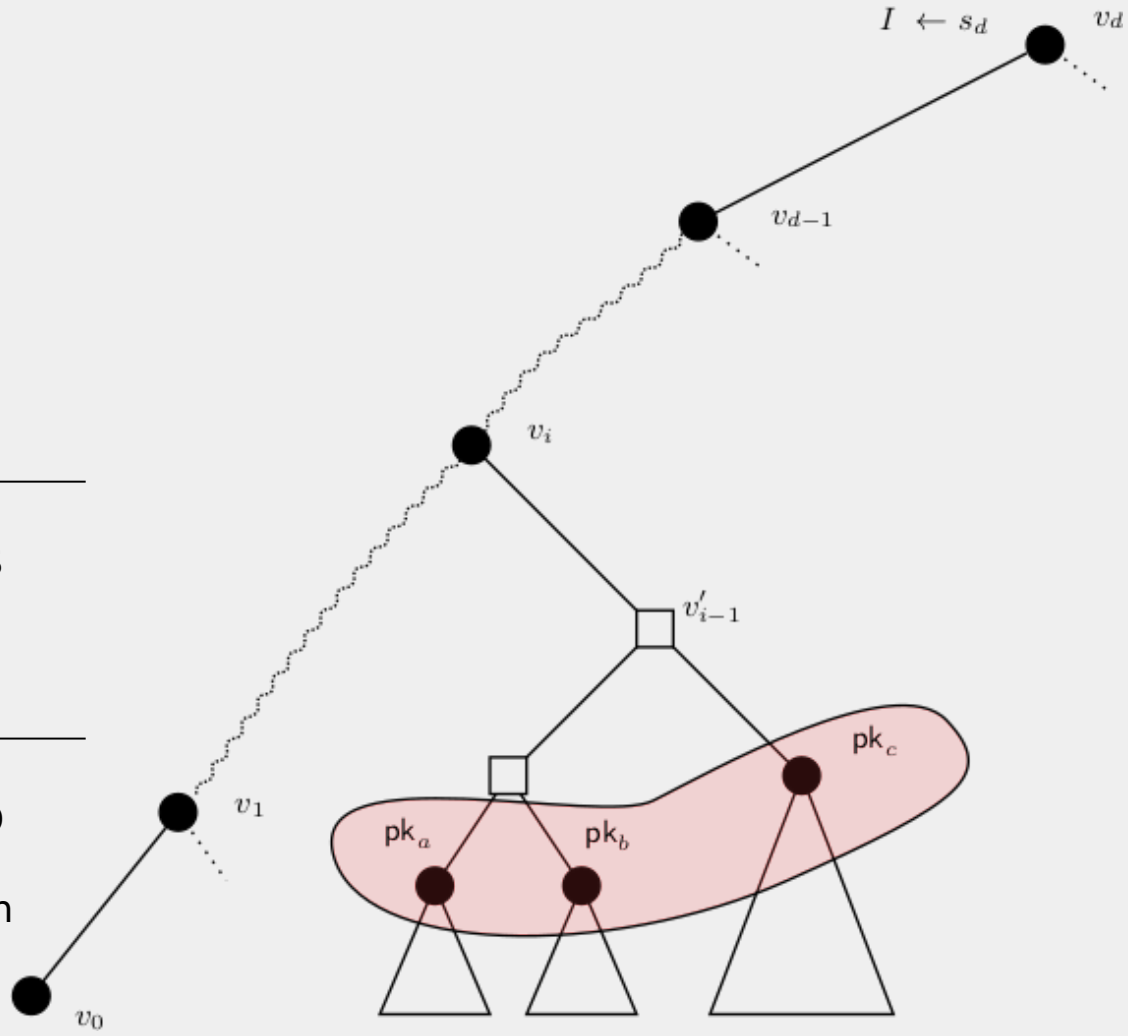
$\mathcal{O}(\log n)$ Control messages

Users of subtree V_i process

$\mathcal{O}(\log n)$ Changes from V_i

Process maintains root info

$\mathcal{O}(\log n)$ Time for new epoch



Where is it?

04

TreeKEM Shortcomings

Forward Secrecy Broken	Post-compromise Security Weak
Message size Large	Deniablity None

TreeKEM Revised (early 2020)

Forward Secrecy ~~Broken~~ Fixed!	Post-compromise Security Weak
Message size Large	Deniablity None

Alwen et al. *Security analysis and improvements for the IETF MLS standard for group messaging*. In Annual International Cryptology Conference, pages 248–277. Springer, 2020.

TreeKEM Surveyed (late 2020)

Forward Secrecy	Post-compromise Security
~~Broken~~ Fixed!	Weak
Message size	**Deniablity**
~~Large~~ Compressed!	None

Katsumata et al. *Scalable ciphertext compression techniques for post-quantum KEMs and their applications*. Internationa Conference Theory & Application of Cryptology and Information Security, p. 289–320. Springer, 2020.

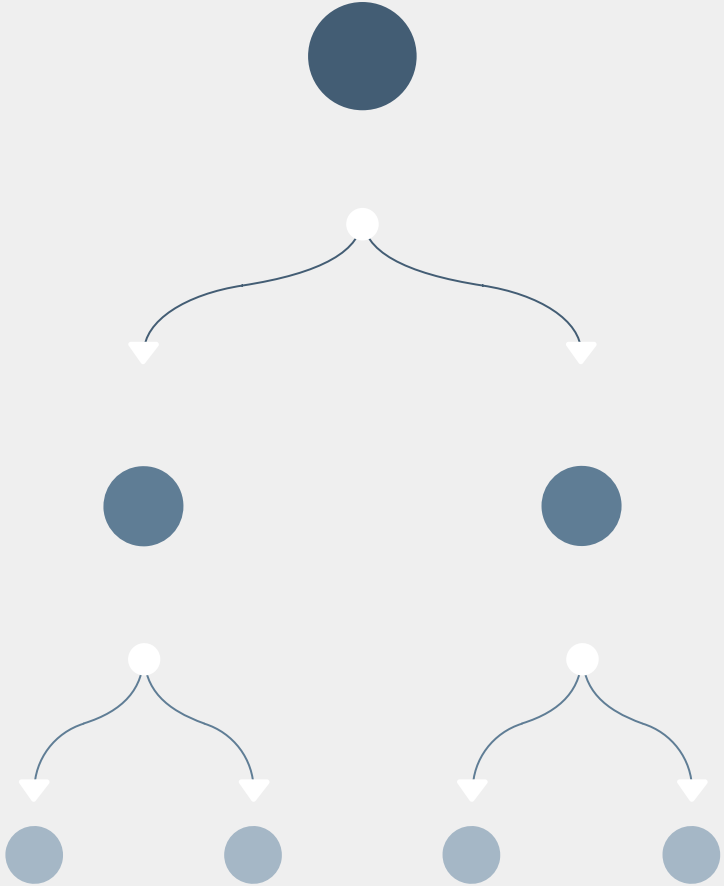
Tainted TreeKEM (late 2021)

Forward Secrecy	Post-compromise Security
~~Broken~~ Fixed!	~~Weak~~ Strengthened!
Message size	**Deniablity**
~~Large~~ Compressed!	None

Klein et al. *Keep the dirt: Tainted treekem, adaptively and actively secure continuous group key agreement*. In 2021 IEEE Symposium on Security and Privacy (SP), pages 268–284. IEEE, 2021.

Conclusion

- SGM actively researched
- Ample use cases for MLS
- TreeKEM iteratively improved
- Multiple open problems
- Coming soon?



Thank You

Do you have any questions?

Alex Washburn
academia@recursion.ninja
www.recursion.ninja