Unus pro omnibus

Multi-Client Searchable Encryption via Access Control

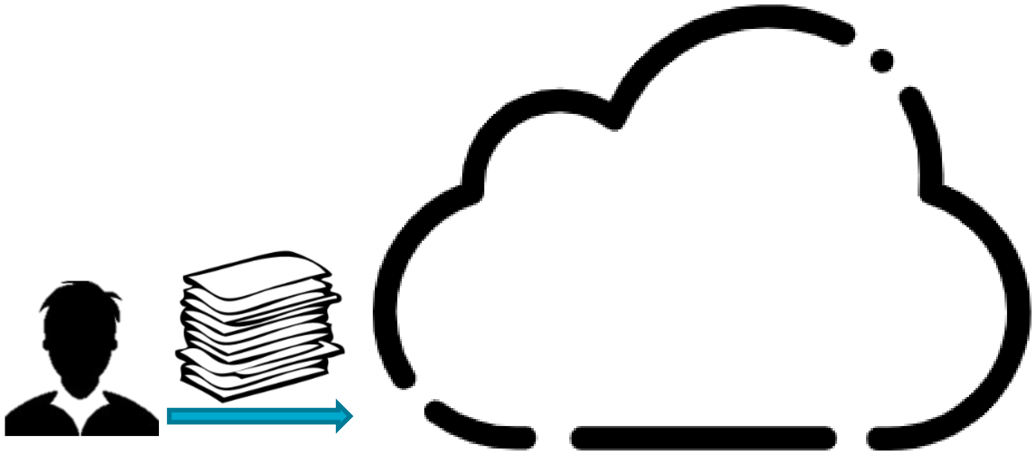
Jiafan Wang
Data61, CSIRO
Commonwealth Scientific and Industrial Research Organisation

Sherman S. M. Chow
Info. Engg., CUHK
Information Engineering
Chinese University of Hong Kong

NDSS SYMPOSIUM/2024
Internet Society
#NDSSSymposium2024

(Secure) Mobile Cloud Applications

- Store ever-growing data from individuals and companies
  - relieve the need for local storage/software



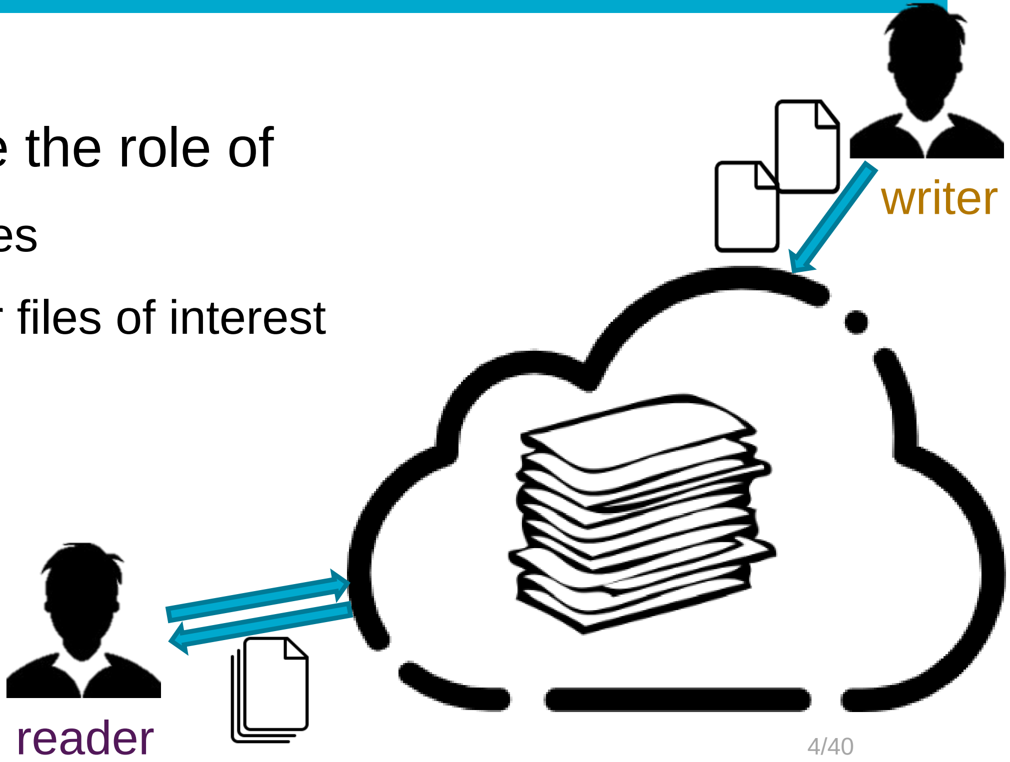
(Secure) Mobile Cloud Applications

- Store ever-growing data from individuals and companies
  - relieve the need for local storage/software
- Manifest as collaborative platforms
- Catalyse novel contributive applications
  - sensor network for crowdsourcing
  - machine learning
  - collective intelligence

from Wikipedia
Google Workspace: 120M education clients by 2021
Microsoft SharePoint: 200M clients by 2020

Collecting & Utilizing Data with the Cloud

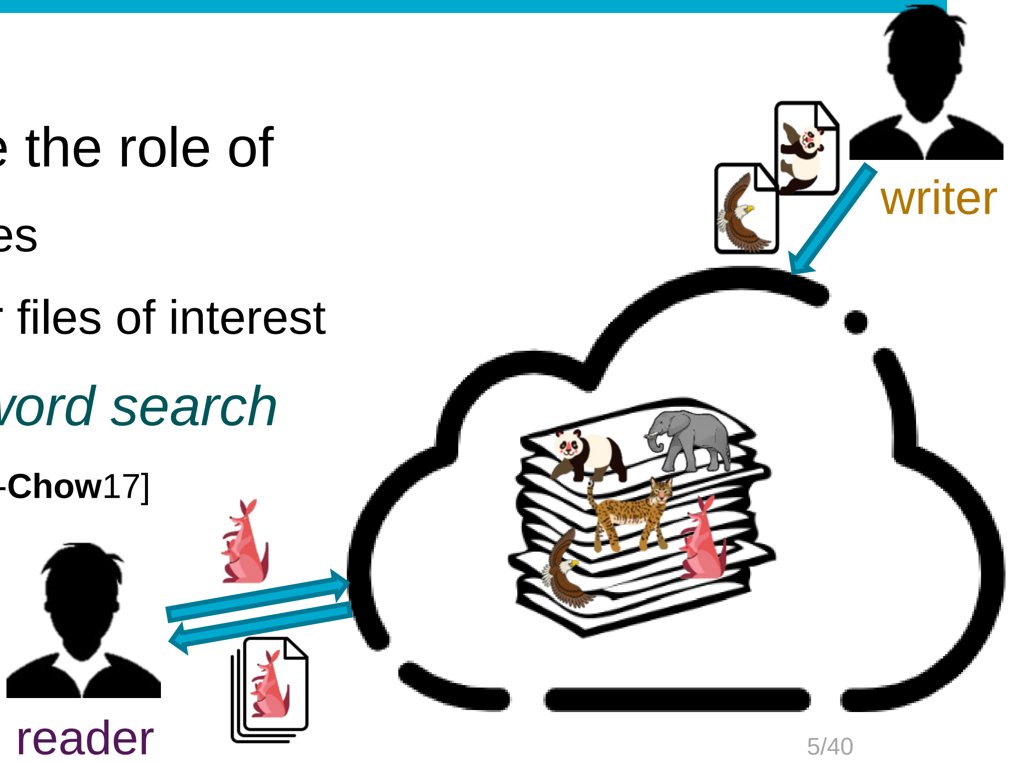
- Each of **many clients** can take the role of
  - *writer* who contributes/updates files
  - *reader* who retrieves/searches for files of interest



Collecting & Utilizing Data with the Cloud

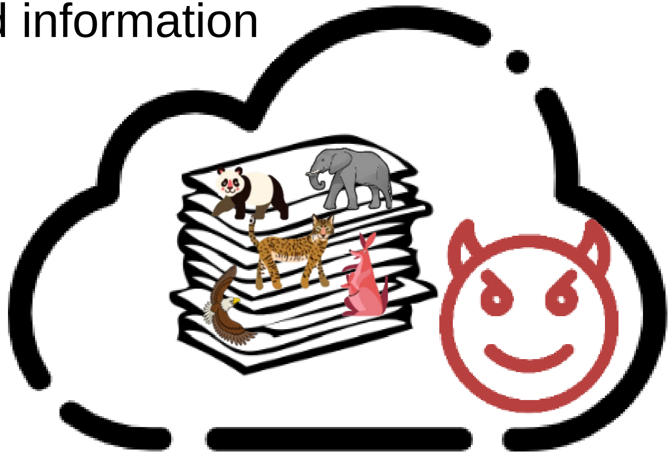
- Each of **many clients** can take the role of
  - *writer* who contributes/updates files
  - *reader* who retrieves/searches for files of interest
- We focus on *fundamental keyword search*
  - cover general query-response [Lai-**Chow**17]
  - each file has a set of keywords
  - "files of interest"

  = those contain a specific keyword



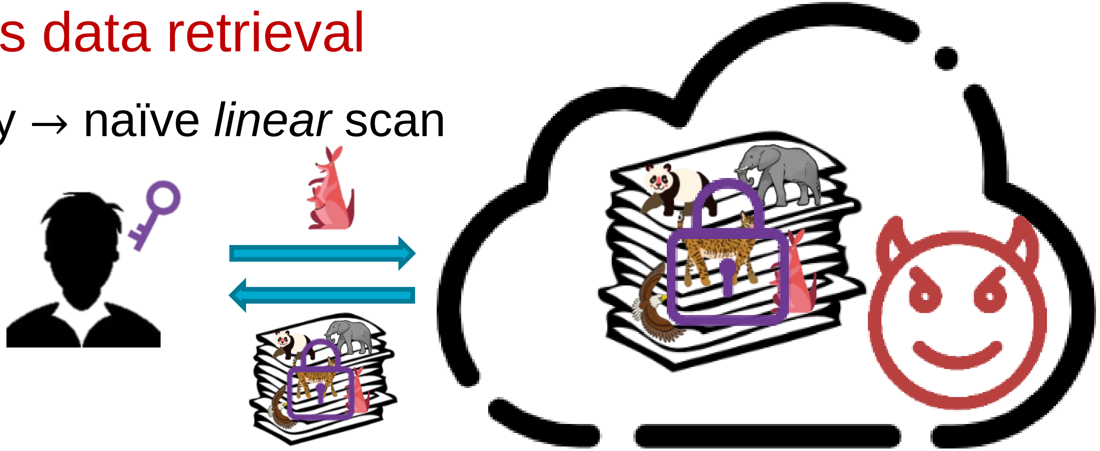
Secure Data Outsourcing

- Cloud servers are untrustworthy
- Outsourced data are sensitive
  - e.g., data breach of medical records or credit card information



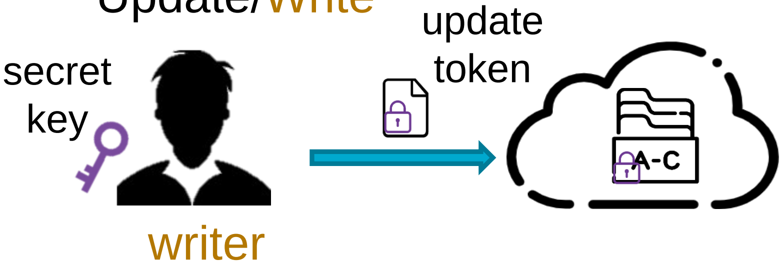
Secure Data Outsourcing

- Cloud servers are untrustworthy
- Outsourced data are sensitive
- Standard encryption hinders data retrieval
  - download all → decrypt locally → naïve *linear* scan
  - no multi-client support

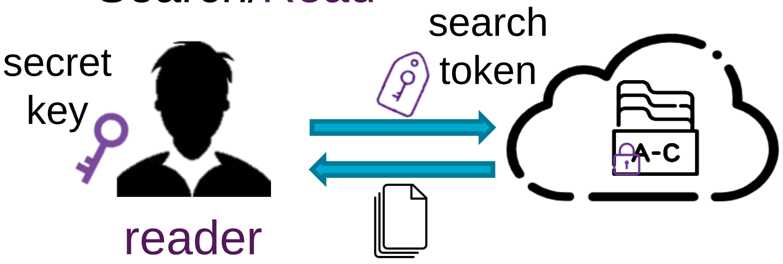


Searching Encrypted Data in the Cloud

- Searchable Symmetric Encryption (SSE) [SWP00, CGKO06, KPR12]
  - Update/Write



  - Search/Read



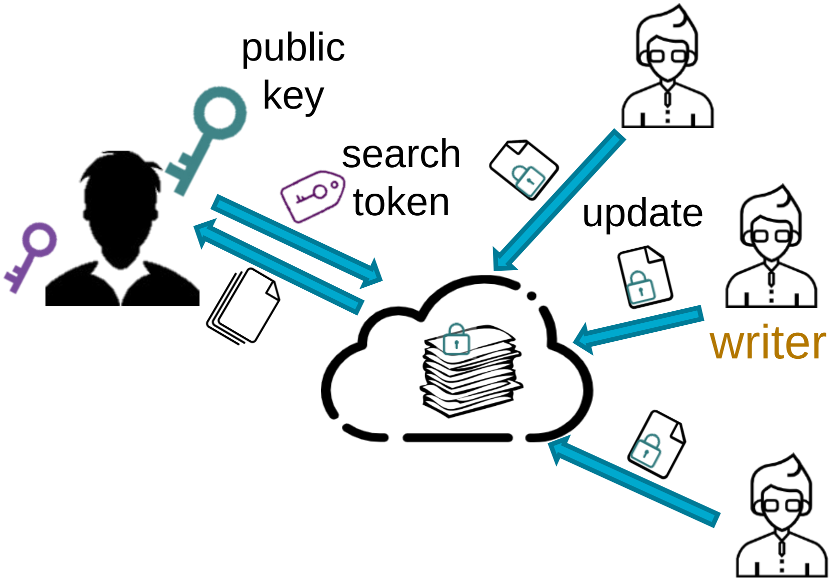
✓ Sublinear search
- index data for optimal search time

✓ Forward privacy [Bost16, Lai-**Chow**17]
- updates can't be searched by old search tokens
- upon any related search, "update" update tokens
- update "changed" so old search token won't work

✘ No multi-client support
- writer = reader = secret-key owner
- symmetric setting hinders multi-writer

Searching Encrypted Data in the Cloud

- Public-key Encryption with Keyword Search (PEKS) [BDOP04]



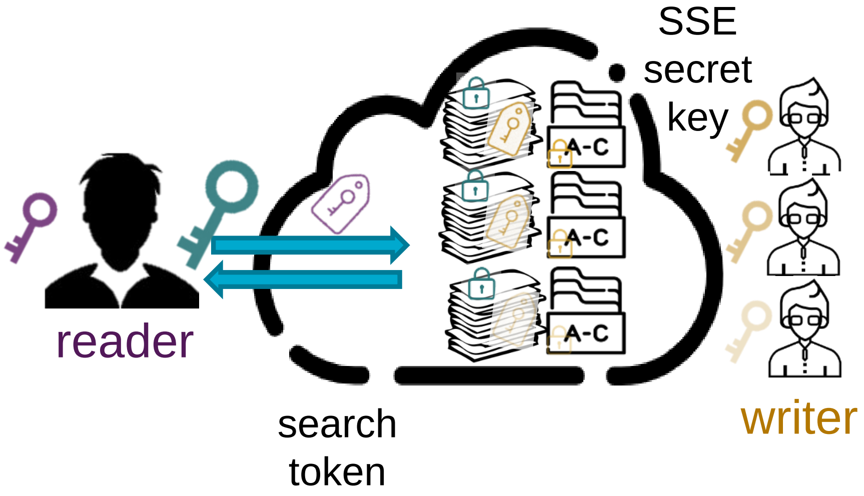
✓ Multiple writers for a reader
- no secret-key distribution
- no synchronous communication

✘ Often require linear testing
- challenging to jointly index with no shared secret/coordination

✘ Hard to be forward private
- writers don't know search state from reader
- nor when to "change" update tokens

Hybrid Searchable Encryption [**Wang-Chow22**]

- 1st non-trivial attempt *towards* the best of SSE [SWP00] & PEKS [BDOP04]



✓Multiple writers for a single reader
- *"All for One"* (*"Omnes pro uno"*)
- access writers' SSE tokens via PEKS

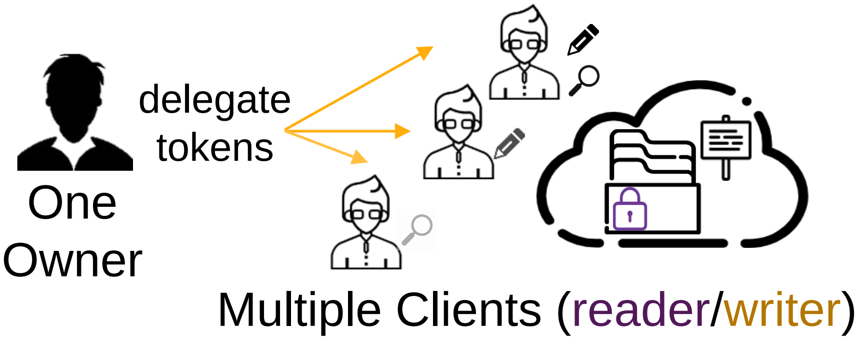
✓Sublinear search
- PEKS.Search for tokens + SSE.Search for files
- PEKS from *ID-Coupling Key-Aggregate Enc.*

✓Forward privacy (per epoch)
- synchronize writers with a global clock
- writers rebuild (per epoch) for fast search

"One for All": Delegatable Searchable Encryption

SSE	PEKS	HSE
✓ Sublinear/optimal search	✓ Multi-writer support	✓ Sublinear search (not optimal ✗)
✓ Forward privacy	✗ Linear Search	✓ Multi-writer support
✗ No multi-client support	✗ No forward privacy	✓ Confined forward privacy (epoch + rebuild ✗)

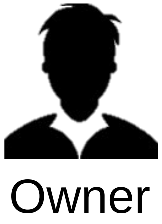
DSE: A New Notion Advancing Searchable Encryption



- ✓ Standard Forward Privacy
- ✓ Search is Optimal Asymptotically
- ✓ Multi-Writer Multi-Reader Support
- ✓ Construction allows Extensibilities

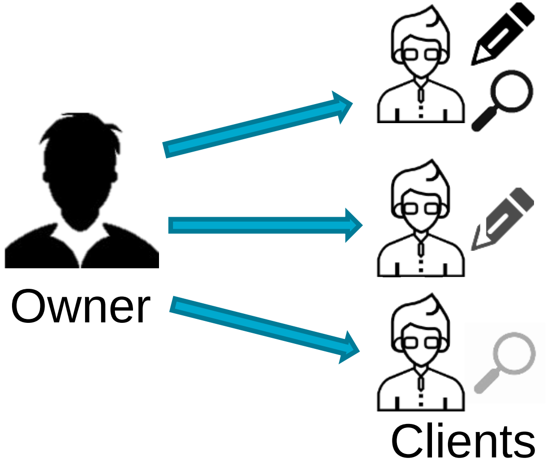
DSE: Access Control on Joint Index

- A data owner initializes the global state and empty database



DSE: Access Control on Joint Index

- A data owner initializes the global state and empty database
- Simple *one-time* (token) delegation (*per client*) by the data owner
  - grant *keyword-specific* updating and/or searching rights

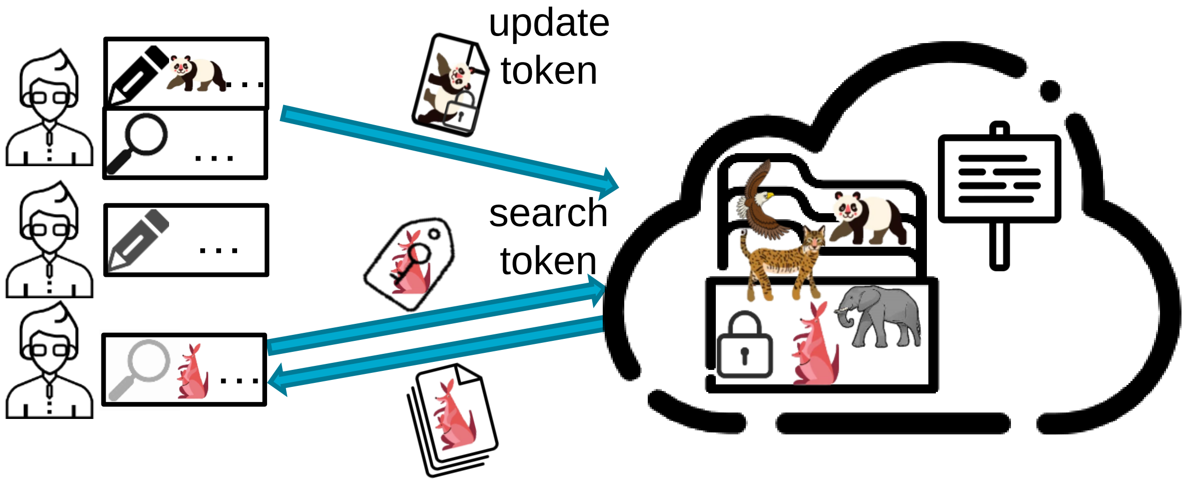


DSE: Access Control on Joint Index

- A client becomes a writer and/or reader of specific keywords

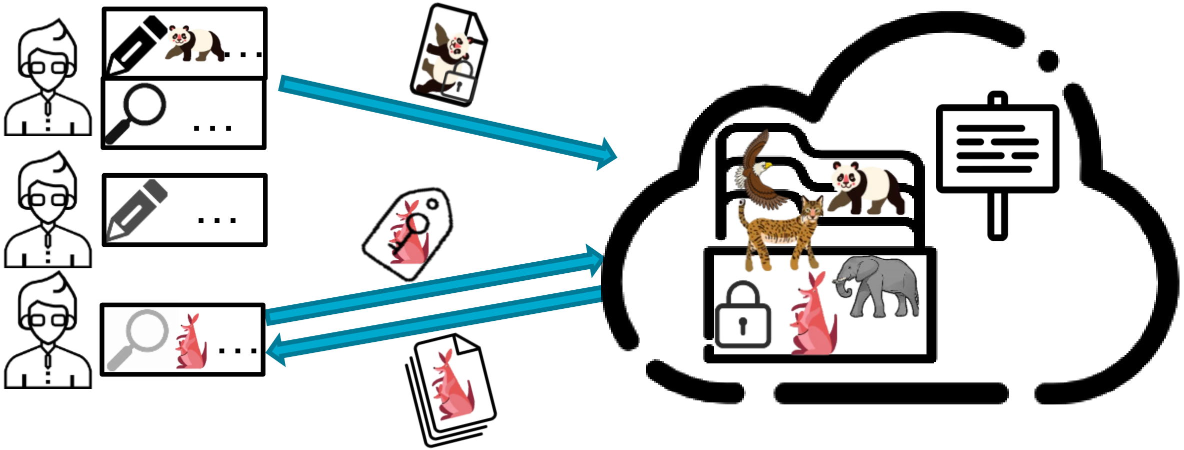
DSE: Access Control on Joint Index

- A client becomes a writer and/or reader of specific keywords
  - make updates and/or searches on these keywords
  - without the help of the data owner



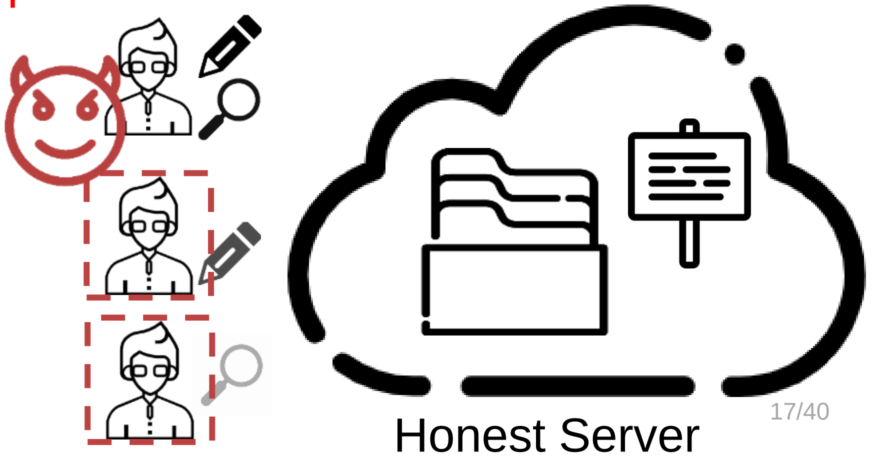
DSE: Access Control on Joint Index

- A client becomes a writer and/or reader of specific keywords
  - make updates and/or searches on these keywords
  - without the help of the data owner
- Delegation makes *joint indices* feasible, boosting searches



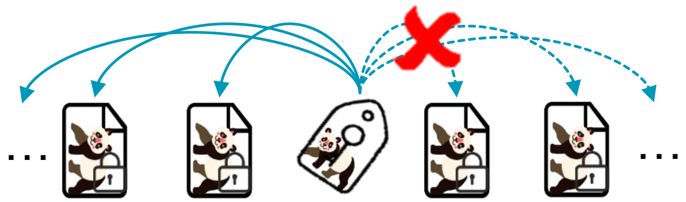
DSE Threat Models

- Server and some clients are possibly corrupted and collusive
- Maintain data privacy and integrity for the trusted data owner
- Clients with searching/updating rights pose *an “orthogonal” threat*
  - leakage to an honest server + corrupted clients could be *less*



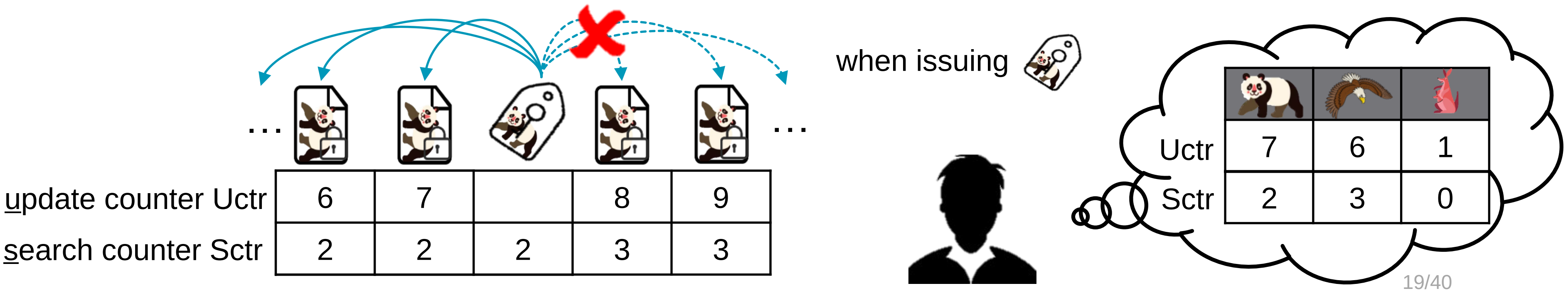
DSE Security

- Forward privacy: search token can't work for *any* future update
  - given the updated keyword *not delegated to any corrupt client*
  - any update *hides anything* w.r.t. non-corrupt keywords



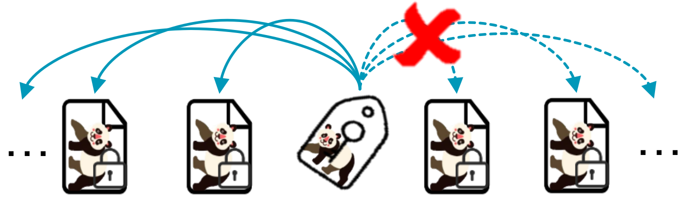
DSE Security

- Forward privacy: search token can't work for *any* future update
  - given the updated keyword *not delegated to any corrupt client*
  - any update *hides anything* w.r.t. non-corrupt keywords
  - a usual trick in SSE: *state maintenance (per keyword)* on the client side



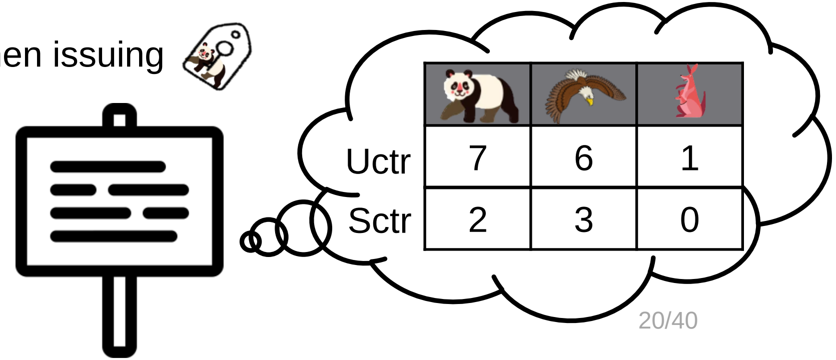
DSE Security

- Forward privacy: search token can't work for *any* future update
  - given the updated keyword *not delegated to any corrupt client*
  - any update *hides anything* w.r.t. non-corrupt keywords
  - a usual trick in SSE: *state maintenance (per keyword)* encrypted on the server



update counter Uctr	6	7		8	9
search counter Sctr	2	2	2	3	3

when issuing



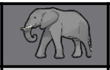
DSE Security

- Forward privacy: search token can't work for *any* future update
  - given the updated keyword *not delegated to any corrupt client*
  - any update *hides anything* w.r.t. non-corrupt keywords

- Integrity: malicious clients can't tamper with the database
  - new concern in DSE to make clients behave themselves
  - ensure correct modification of the global state
  - later searches by honest clients can locate others' updates in the right places

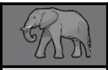
Shiftable Multi-Recipient Encryption (SME)

- A tailor-made *homomorphic* encryption for *a set of messages*
  - *shiftable*: homomorphic shifting can be publicly computed

			
8	4	2	7
9	**5**	2	7

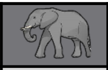
Shiftable Multi-Recipient Encryption (SME)

- A tailor-made homomorphic encryption for a set of messages
  - shiftable: homomorphic shifting can be publicly computed
  - *expandable*: more recipients can be included (with new secret keys)

				
8	4	2	7	
9	**5**	2	7	
9	5	2	7	**0**

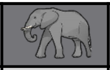
Shiftable Multi-Recipient Encryption (SME)

- A tailor-made homomorphic encryption for a set of messages
  - shiftable: homomorphic shifting can be publicly computed
  - expandable: more recipients can be included (with new secret keys)
  - *shift non-committing*: ciphertexts can be simulated without knowing the shifted messages and the shifted offsets

				
?	?	?	?	
?	?	?	?	
?	?	?	?	0

Shiftable Multi-Recipient Encryption (SME)

- A tailor-made homomorphic encryption for a set of messages
  - shiftable: homomorphic shifting can be publicly computed
  - expandable: more recipients can be included (with new secret keys)
  - *shift non-committing*: ciphertexts can be simulated without knowing the shifted messages and the shifted offsets
  - can “explain” (previously) simulated ciphertexts when the key is revealed

				
?	?	?	?	
?	?	?	?	
?	?	?	?	0

Shiftable Multi-Recipient Encryption (SME)

- A tailor-made homomorphic encryption for a set of messages
  - shiftable: homomorphic shifting can be publicly computed
  - expandable: more recipients can be included (with new secret keys)
  - shift non-committing: ciphertexts can be simulated without knowing the shifted messages and the shifted offsets

?	?	?	?	
?	?	?	?	
?	?	?	?	0

- Efficient SME construction
  - from randomness-reusing ElGamal encryption
  - *non-committing* proven under the generic group model
  - synchronize updates via the global state for *standard forward privacy*

DSE-F: Foundation with Forward Privacy

- Data owner sets up the DSE system with a master key
  - an SME instance, with each ctxt slot corresponding to one keyword
    - SME ctxt to encrypt *update counters* of each keyword
  - multiple IBE instances, each corresponding to one keyword
    - (Anonymous Identity-Based Encryption implies PEKS)
    - to encrypt documents for each keyword, using its *search counter* as identity
  - pseudorandom function PRF to locate addresses for tuples

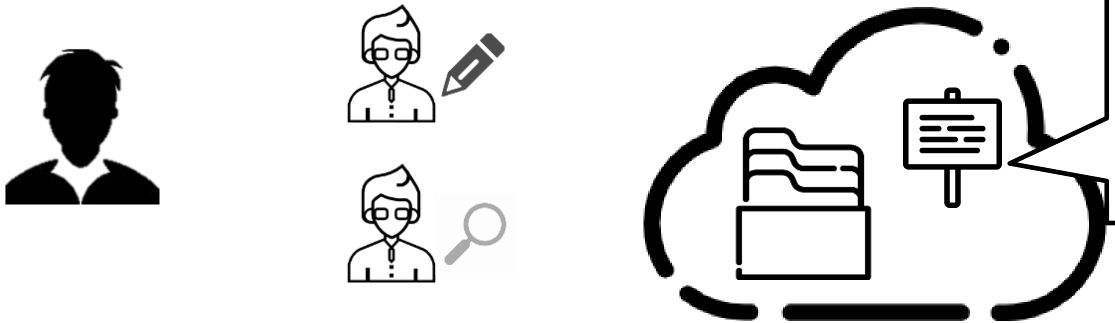
DSE-F: Foundation with Forward Privacy

- Global state (Uctr: update counters; Sctr: search counters)
  - public keys + SME-encrypted Uctr + Sctr (in plain, & “pseudonym” of keyword)

Uctr	0	0	0
Sctr	0	0	0

DSE-F: Foundation with Forward Privacy

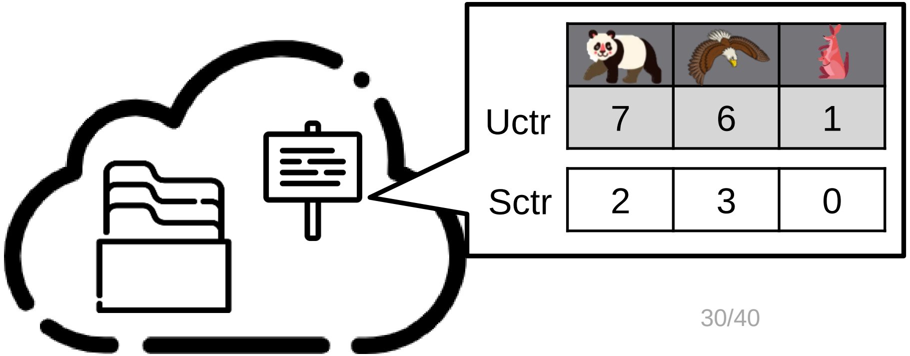
- Global state (Uctr: update counters; Sctr: search counters)
  - public keys + SME-encrypted Uctr + Sctr (in plaintext, pseudonym)
- Data owner provides keyword-specific keys to clients
  - : SME secret key + PRF secret key + IBE secret key
  - : SME secret key + PRF secret key (PRF operates over Index Key (IK))



Uctr	0	0	0
Sctr	0	0	0

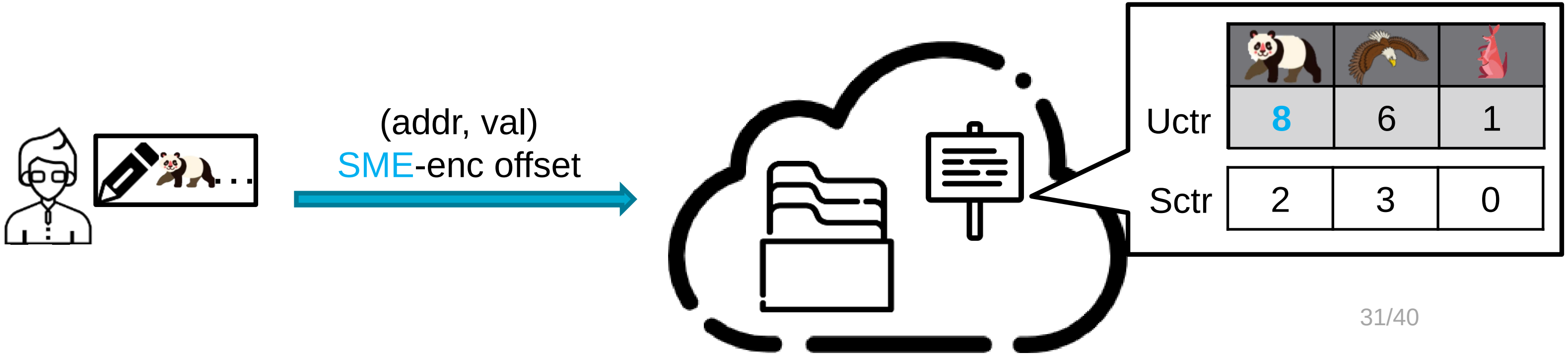
DSE-F: Foundation with Forward Privacy

- To insert a new tuple (, Fid), a writer outputs
  - addr as PRF(IK, Uctr[ ] + 1), with IK = PRF(Sctr[ ]), Uctr from SME
  - val as IBE(Fid, Sctr[ ]) using ’s IBE instance
- The writer also outputs SME-encrypted offset



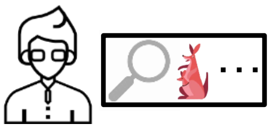
DSE-F: Foundation with Forward Privacy

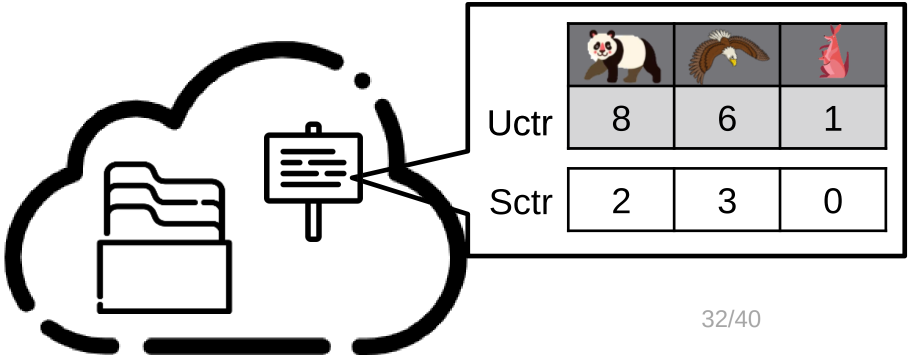
- To insert a new tuple (, Fid), a writer outputs
  - addr as PRF(IK, Uctr[ ] + 1), with IK = PRF(Sctr[ ]), Uctr from SME
  - val as IBE(Fid, Sctr[ ]) using 's IBE instance
- The writer also outputs SME-encrypted offset
- The server stores val at addr and shifts SME ctxt



DSE-F: Foundation with Forward Privacy

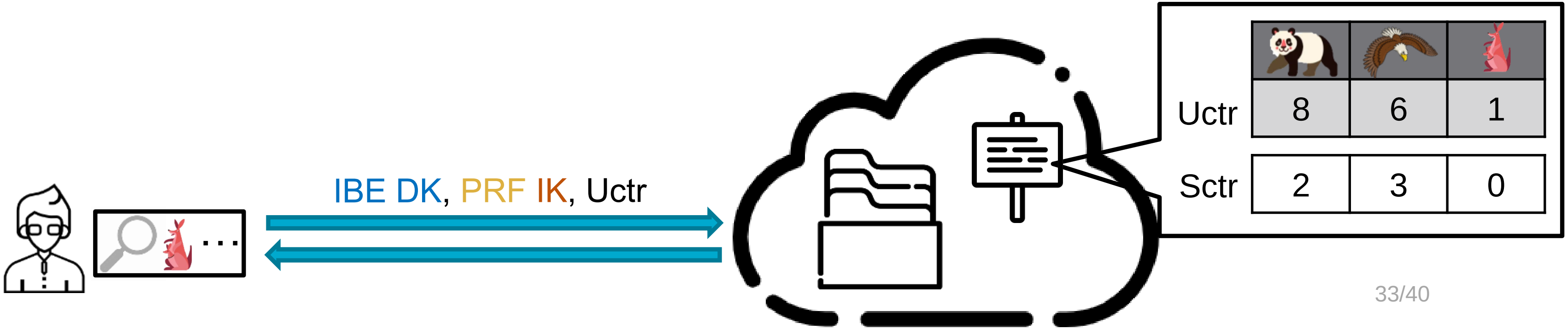
- To search for , a reader outputs
  - Decryption key DK for 's IBE instance, w.r.t. Sctr[ ]
  - Index key IK for PRF as IK = PRF(Sctr[ ])
  - Uctr[ ], by decrypting the corresponding slot of SME ctxt



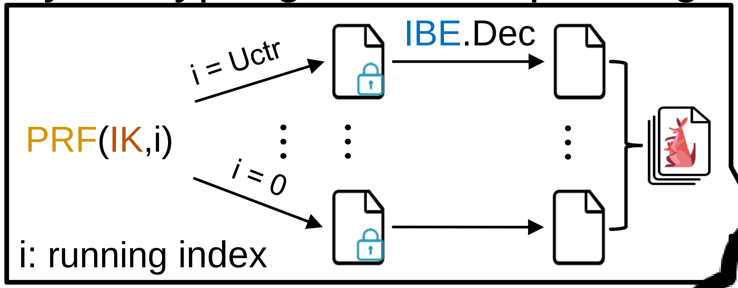


DSE-F: Foundation with Forward Privacy

- To search for , a reader outputs
  - Decryption key DK for ’s IBE instance, w.r.t. Sctr[]
  - Index key IK for PRF as IK = PRF(Sctr[])
  - Uctr[], by decrypting the corresponding slot of SME ctxt
- The server retrieves entries with IK, decrypts them with DK



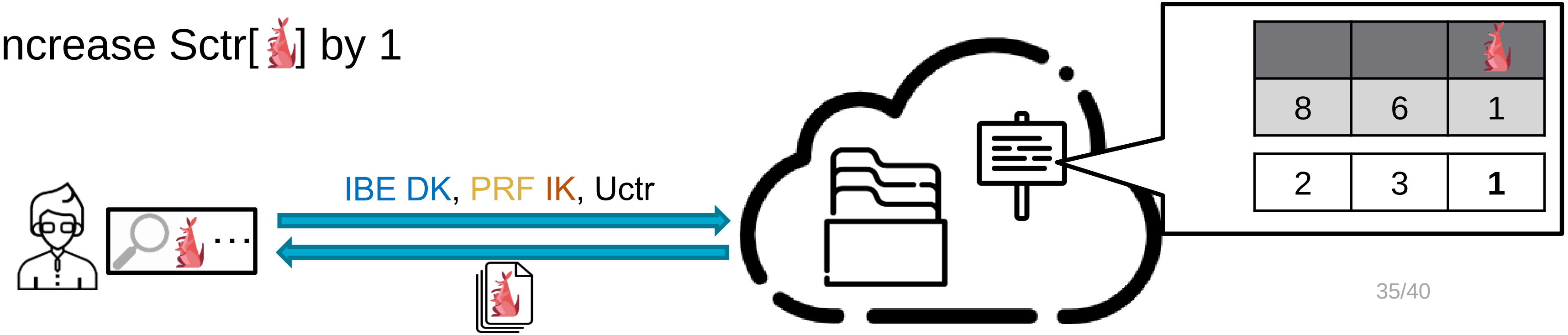
DSE-F: Foundation with Forward Privacy

- To search for , a reader outputs
  - Decryption key DK for 's IBE instance, w.r.t. Sctr[]
  - Index key IK for PRF as IK = PRF(Sctr[])
  - Uctr[], by decrypting the corresponding slot of SME ctxt
- Server 

decrypts them with DK

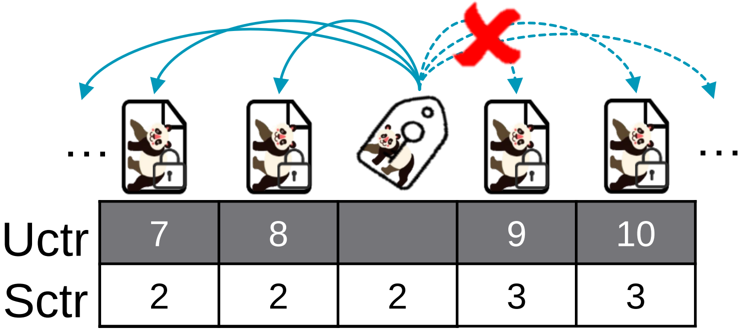
DSE-F: Foundation with Forward Privacy

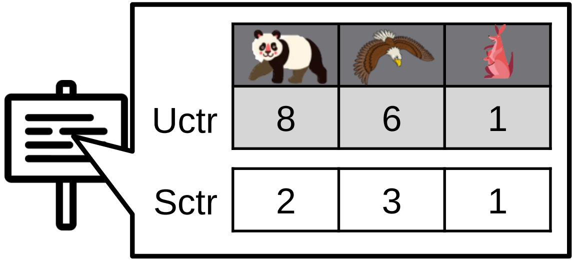
- To search for , a reader outputs
  - Decryption key DK for 's IBE instance, w.r.t. Sctr[ ]
  - Index key IK for PRF as IK = PRF(Sctr[ ])
  - Uctr[ ], by decrypting the corresponding slot of SME ctxt
- The server retrieves entries with IK, decrypts them with DK
  - Increase Sctr[ ] by 1



Security Analysis of DSE-F

- Update leaks nothing of non-corrupt keywords
- IBE.Enc and PRF IK are w.r.t *current* search & update counter
- Standard forward privacy



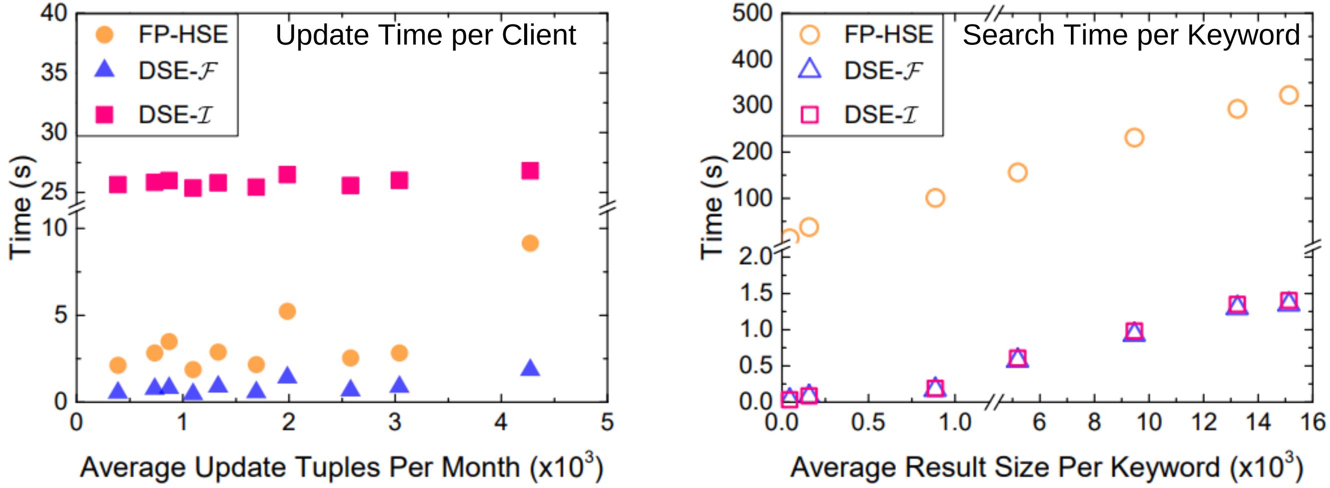


DSE-I: Instrument for Integrity

- Any search or update comes with a proof
  - using commitment [Pedersen91] and argument of knowledge [LMR19]
  - update: prove SME ctxt is correctly shifted
  - search: prove IBE DK and PRF IK are generated w.r.t correct Sctr

Experimental Evaluations

- Enron: 510K e-mails from 146 employees from 1999 to 2002
  - each client has an average of 460 keywords and 3493 emails
  - each client updates per month; regular searches per half year
  - amortize rebuild time (per half year) into HSE update time



Summary of Ideas

- *One-time* delegation to grant reading and/or writing ability
  - with respect to a keyword by the data owner
- Global state for synchronization needed for forward privacy
  - Maintained by *shiftable multi-recipient non-committing encryption* (SME)
  - *Forward privacy is now possible without client interactions*
- Jointly build and retrieve *one* index
  - Instead of adding a public-key layer over multiple indexes in HSE
  - *Multi-writer multi-reader with sublinear search*

Conclusion and Future Work

jiafan.wang@data61.csiro.au
sherman@ie.cuhk.edu.hk

- DSE-F and DSE-I
  - Adaptively secure
  - Forward-private
  - DSE-I achieves *integrity* (with overhead)

	SSE	PEKS	HSE	DSE
Sublinear Search	✓	×	✓	✓
Standard Forward Privacy	✓	×	×	✓
Multiple Clients	×	W	W	W/R

- Versatility and Extensibility
  - Extended defence: backward privacy, volume hiding, mitigating keyword guess
  - Optimized efficiency: fewer public-key operations via HSE-like hybrid technique
  - See the paper for more