Stealth Databases: Ensuring User-Controlled Queries in Untrusted Cloud Environments

Josef Spillner, Martin Beck, Alexander Schill, Thomas Michael Bohnert
<josef.spillner@zhaw.ch>

Service Prototyping Lab (blog.zhaw.ch/icclab)
Zurich University of Applied Sciences, Switzerland

December 9, 2015 | 8th IEEE/ACM UCC, Limassol, Cyprus

Intro: Data Management in the Cloud

«We did this 7 years ago.» (cloud databases/data management, DBLP)

«We did this 40 years ago.» (networked databases)

Unsolved challenges:

- Maintaining availability and confidentiality of data
- User-controlled multi-criteria optimisation
- Applicability to cluster, cloud and streaming environments

Data Service Quality Concerns

↑ Availability
~99.9999% available.

↑ Reliability
100% available, 100% correct.

↑ Confidentiality
not readable for third parties

↑ Scalability, Resilience, ...

↓ Volume, Price, ...

Intrusion

Risks

Dropbox Sync Error Causes Data Loss For Customers

Loss

Dropbox is experiencing issues.

Downtime

Leak

ZDNet

Hot Topics | Reviews | Downloads | Newsletters | White Pa

UK Edition | UK News | IT Budgeting | IT Jobs | 3D Printing | Hybrid Cloud

READ THIS: Apple Pay ready for lift-off and Google 'trying t

Topic: Security

Follow via:

AWS users fret over downtime ahead of Amazon's massive EC2 reboot

Bankruptcy

cloudsafe

We are saying Good Bye...

Dear CloudSafe User,

...]

Nevertheless, we did not convince enough users to pay for our services. The price per storage

important buying criterium. Ultimately you need to scale quickly to cover your operational costs

New pricing and chan

On November 19, 2013 th

Infinite Storage offering would increase in price from US$99/year to US$999/year. The move sparked an intense reaction from users at the company's forum, even though existing users were grandfathered into the original pricing plan [17] Reaction from Bloggers was particularly critical.

Price Jump

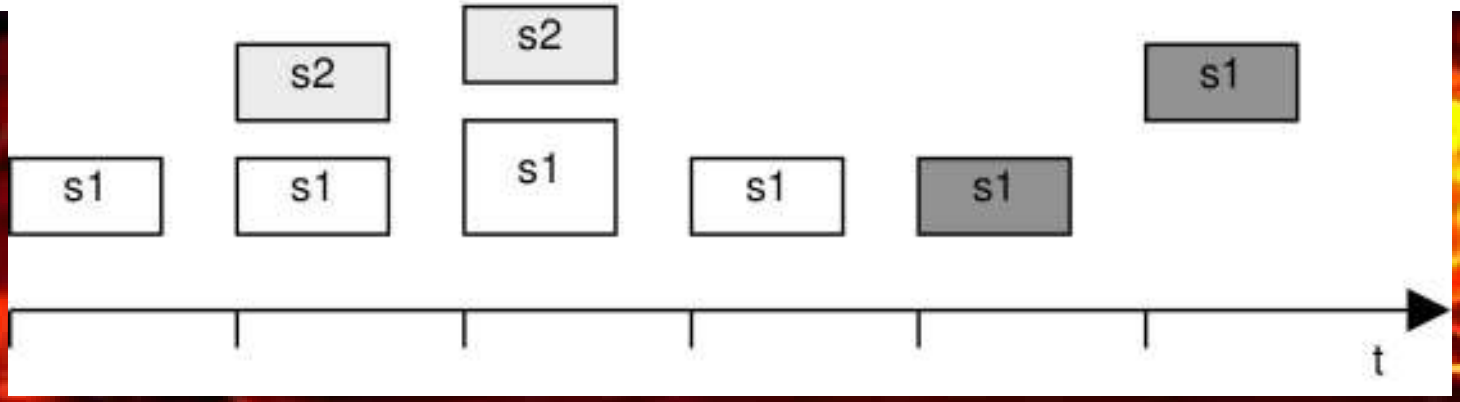
SPLAB

Stealth Idea

Stealth layer: Coverable cloud service evolution

initial | addition | growth | decline & vanish | price change | location change

s: cloud services (PaaS, IaaS)

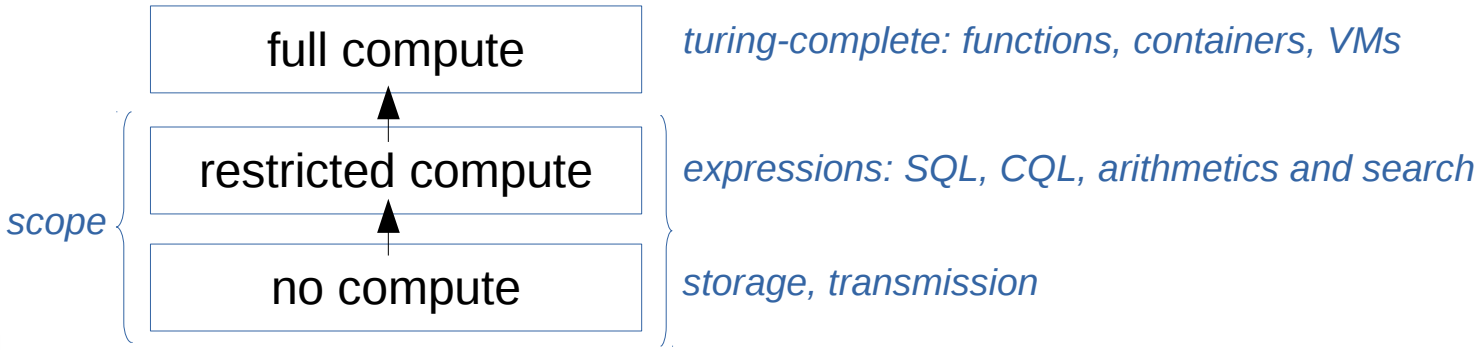


Stealth Concepts Overview

Stealth Computing: Well-protected processing of data in the cloud [NetSys'15, BlackSeaCom'15]

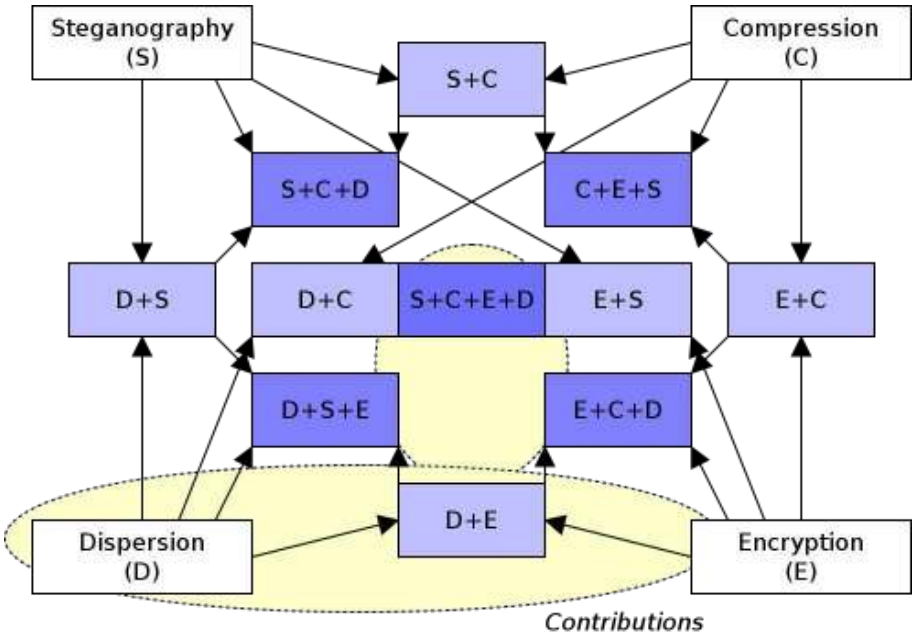
Concepts
- Data coding and distribution
- Data processing
- User preferences and quality constraints

Approach: operation-aware data multi-coding, distribution and processing
- in one pipeline for data records and data streams over multiple independent services evolving over time



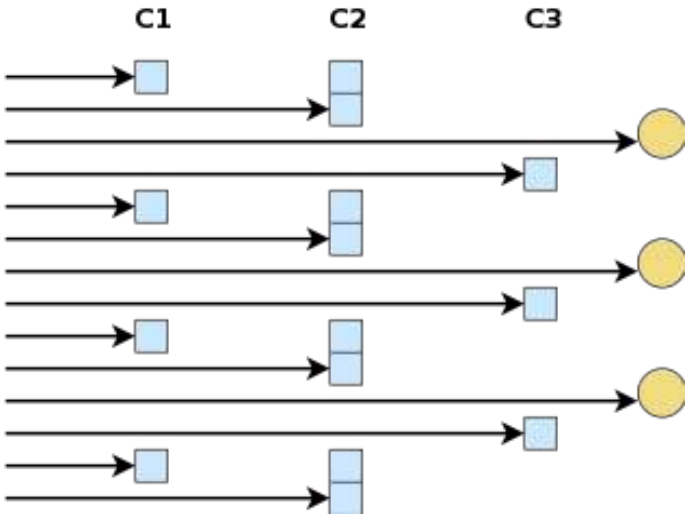
Concept: Multi-Coded Distributed Data

Coding



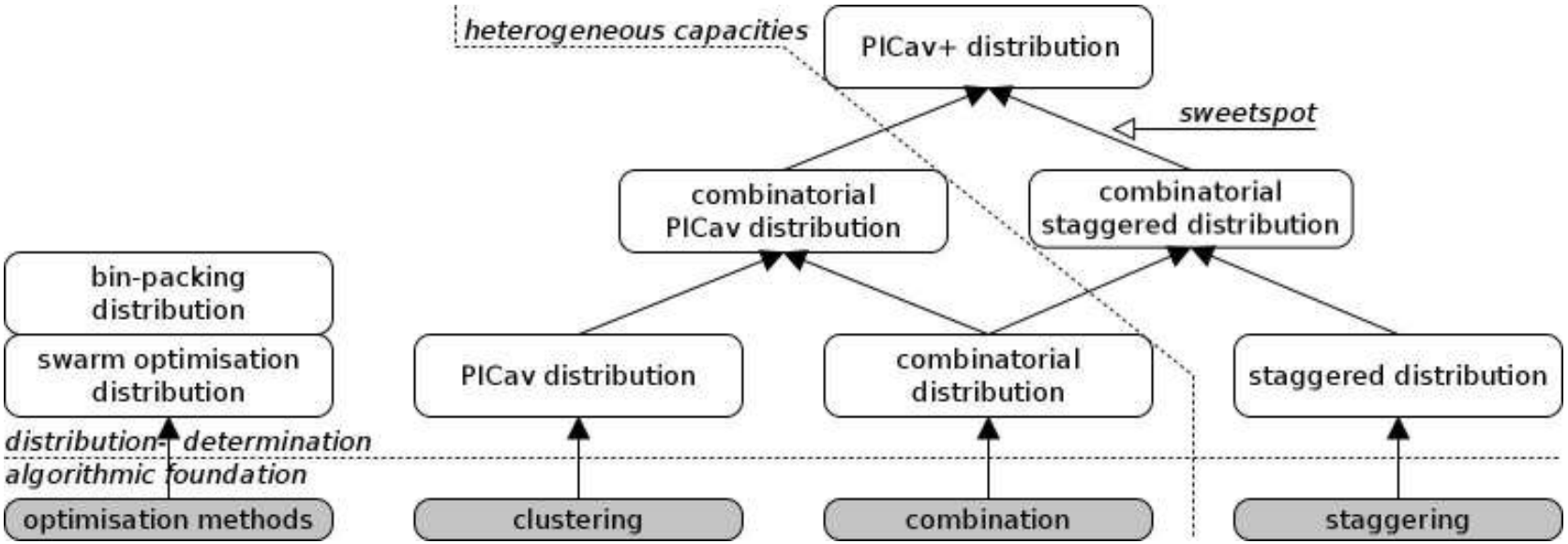
Goals:
availability
→ executable
dispersion
confidentiality/privacy
→ executable
encryption
capacity
→ executable
compression

Executable Multi-Coding
= «Stealth Data»



Concept: Multi-Coded Distributed Data

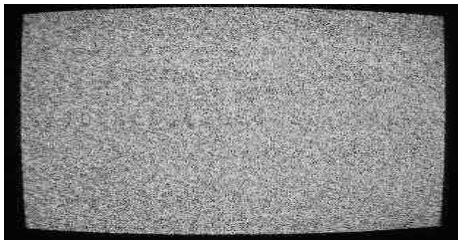
Distribution: find optimal placement according to desired availability with minimum redundancy overhead



Placement algorithm PICav+: Improved Precise, Iterative, Complement-based availability calculation [UCC'14]

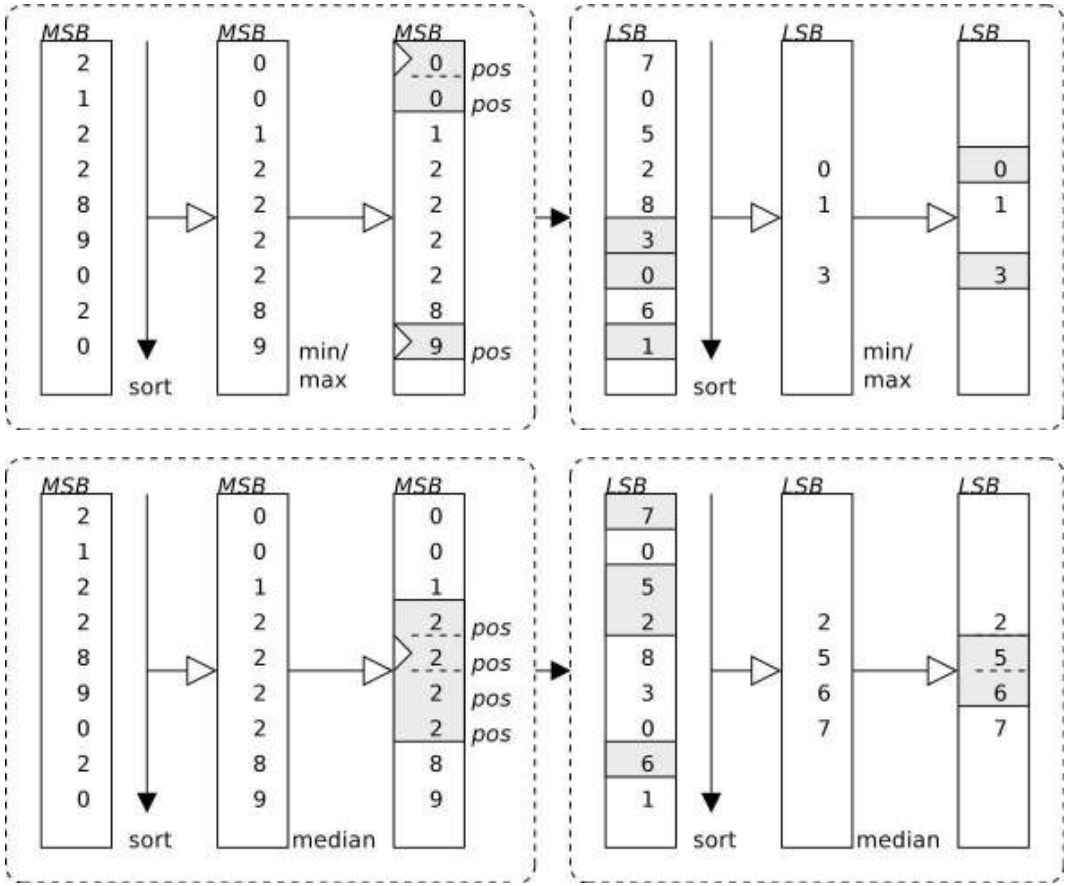
$$\mathcal{C}_n \subset \mathcal{C}_h \in \mathfrak{P}(\mathcal{C}) = \{\{T_1\}, \{T_1,T_2\}, \ldots, \{T_1, \ldots, T_n\}\}$$

Concept: Stealth Data Processing



Local processing in each location

- resource provider view: random data blocks
- application view: map-carry-reduce access to full results



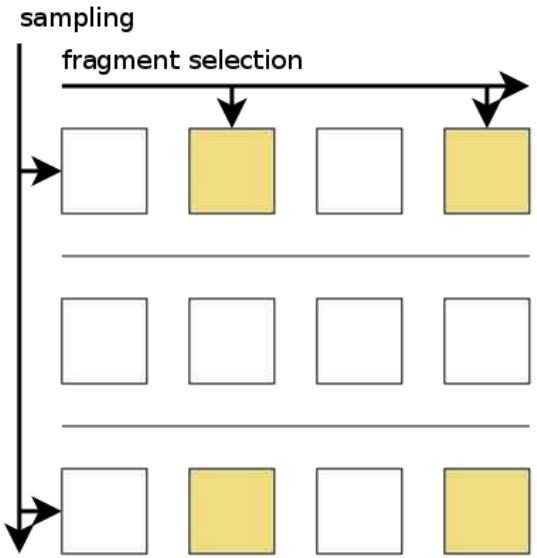
Concept: Type-Ops-Dependent Coding

Strings
- exact string search
- redundant fragments: fuzzy string search
- fragment selection: heuristics

Integer numbers
- exact arithmetics
- sampling: approximate arithmetics
- fragment selection: heuristics

Floating point numbers
- fragment selection: varying-precision processing

Multimedia
- fragment selection: interpolated processing



Concept: Query Optimisation

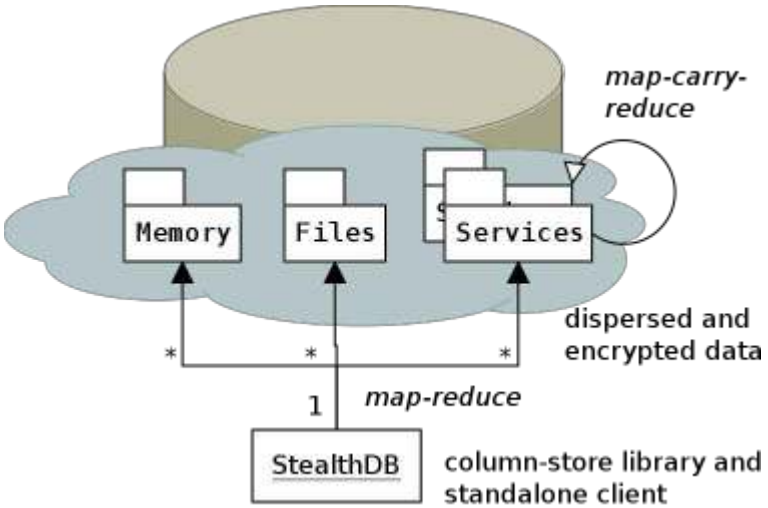
Syntax: `SELECT ... OPTIMIZE FOR <goal(s)>`

Preferences

- performance
  - apply fragment selection & sampling
- precision
  - request all fragments (even beyond default)
- reliability
  - query fully replicated values & compare
- energy efficiency
  - upon sorting: apply sweetspot CPU frequency
- power
  - upon sorting: apply lowest CPU frequency

StealthDB System Overview

Resources
- local storage (RAM, files) + compute
- remote storage+compute services



(Processable) dispersion
- replication, hashring, erasure, bitsplit

(Processable) encryption
- homomorphic, order-preserving, searchable, fuzzy

Preferences
- optimises queries for performance, reliability, energy efficiency, precision

Features
- data records and streams
- per-column distribution and migration control
- map-carry-reduce operators

SPLAB

StealthDB Architecture

Coding: Dispersion
- erasure | bitsplitting | hashring replication

Coding: Encryption
- homomorphic + order-preserving + searchable + diffuse

Features:
- datasets & streams
- per-column distribution
- migration control
- map-carry-reduce operations
- user requirements optimisation
- dynamic deployment of processing code

StealthDB Software

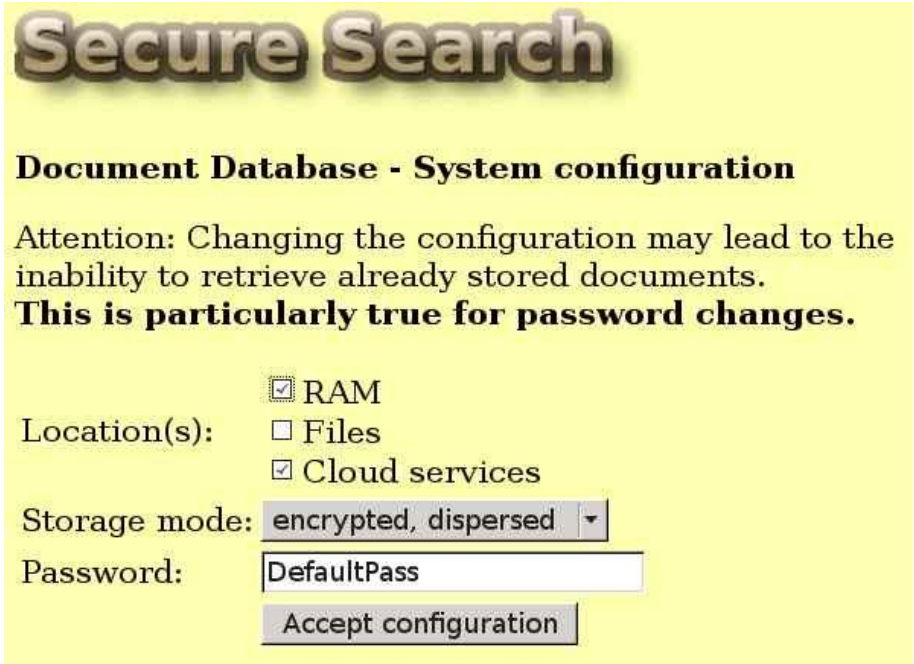
Laboratory approach: live demo - recomputable results / reproducibility

```
josef@rumba:/repos/space-universe/dispersedalgorithms/db$ ./stealthdb
~~ StealthDB ›master ›Wed May 20 16:14:37 2015 +0200 ~~
Type HELP; to get started.
Using database 'stealthdb'.
Storing all data and performing all procedures on ['mem://localhost'] with ['replication'].
>>> HELP;
StealthDB Quickhelp
 HELP [<topic>]
 SHOW DATABASES|TABLES
 CREATE TABLE <table> [(<column> <column-type>, ...)]
 DESCRIBE <table>
 DROP TABLE [IF EXISTS] <table>
 CREATE DATABASE <database>
 USE DATABASE <database>
 DROP DATABASE <database>
 [EXPLAIN ANALYZE] SELECT [DISTINCT] */<column>/<aggregate>(*/<column>)/<predicate>, ... [FROM <table>]
  [WHERE <column> LIKE/=/... <value>] [ORDER BY <column> [ASC|DESC]] [OPTIMIZE FOR <goal>] [FOREVER]
 INSERT INTO <table> (<column>, ...) VALUES (<value>, ...)
 DELETE FROM <table>
 USE CLOUDS <cloud> [AND <cloud>...][WITH <distribution>]
 ALTER TABLE <table> [ALTER COLUMN <column>] USE CLOUDS ...
 MODE <mode>
>>>
```

Functionality Evaluation: Stealth Apps

Embedding StealthDB
- low-level: Python methods
- high-level: SQL parser method
- transparent: as a network proxy/ service

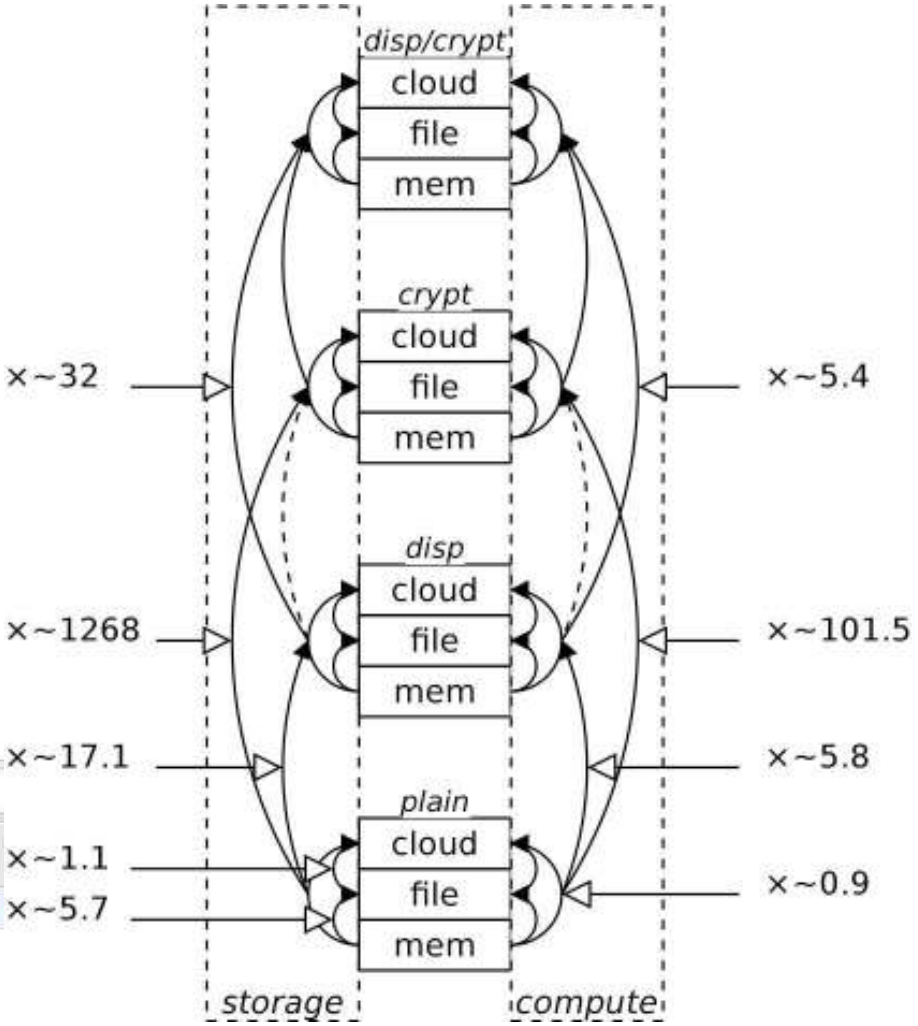
So far
- 3 stealth web applications
- 1 stealth IoT streaming prototype



SPLAB

Performance Evaluation

Diverse configurations - Raspberry Pi cluster, Amazon EC2, localhost...



Note: quantitative considerations to be generalised at some point

Summary & Future Work

Achievements

- tiny but powerful data processing prototype for mixed setups
  - clouds, clusters (RPC), in-memory, files, ...
- user-friendly query optimisations to ensure SLAs can be met
- novel design paradigm for inherent quality in cloud applications

Future work

- effort/benefit comparison with other reliability and security techniques
  e.g. cloud-native applications
- marketplaces full of stealthy applications!

Thanks to ad-hoc collaboration partners

References

[Spi15h] Josef Spillner:
Investigations of the risk minimisation technique Stealth Computing for distributed data-processing software with user-controllable guaranteed properties.
Habilitation treatise, Technische Universität Dresden, expected publication in December 2015.

[Spi15] Josef Spillner:
Secure Distributed Data Stream Analytics in Stealth Applications. (Demo)
3rd IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom), Constanţa, Romania, May 2015.

[SMS15] Josef Spillner, Lorenzo Miori, Julian Sanin:
Stealth Apps for Secure Personal Data Analytics in the Cloud. (Demo)
2nd International Conference on Networked Systems (NetSys), Cottbus, Germany, March 2015.

[Spi14] Josef Spillner:
Project Report: Dispersed Data Processing Services for Third-Party Applications.
HPI Future SOC Lab project, University of Potsdam technical report, September 2014.

[SS14b] Josef Spillner, Alexander Schill:
Algorithms for Dispersed Processing.
1st International Workshop on Advances in Cloud Computing Legislation, Accountability, Security and Privacy (CLASP), London, UK, December 2014.

[SS14a] Josef Spillner, Alexander Schill:
Towards Dispersed Cloud Computing.
2nd IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom), Chișinău, Moldova, May 2014.