Algorithmic Architectures for End-to-End Security of Systems: Challenges

RK Shyamasundar

IIT Bombay

rkss@cse.iitb.ac.in

Organization

- General: Privacy, Public, Secret, ..
- Leakage of Information and Differential Privacy.
- Landscape of Map Reduce
    - Various attempts
- A Novel Approach
    - RWFM approach
    - End2End Preservation of Privacy in Hadoop: SecHadoop
- Summary

Public, Secret, Private

- Secret:
  - Secrets are things that are not meant to be shared.
  - Shared secrets are meant to remain within a well defined group,
  - Shared secrets are not meant to be shared outside that group.
- Example: Password
  - Website cannot afford to keep a password; just check the crypto-hash

Public information

- Public information is anything that is meant to be publicly known. There is no risk to anyone's privacy from you coming to know this.
- In public key cryptography
  - Public key
  - Private key (actually SECRET!)

Private information

- Your name is private.
    - It is not public information.
    - In closed environments such as a corporate office or a conference, your name is meant to be shared within that group, which is why you have to wear a name tag.
- An email or WhatsApp message you send to someone is private (similar to letters).
    - Other people cannot see it, unless you or the recipient choose to share it
    - Email and WhatsApp forwards are commonplace. They are still private.
        - There is no way to tell which piece of fake news is circulating around India on WhatsApp, because it’s private. You can only see what you send and receive.
        - Unless it somehow gets published in the media or on a public website, at which point it becomes public.

Where do we place Biometrics

- Our biometrics are also private information.
  - They are not secrets. You leave a copy of your fingerprints on almost everything you touch. Your iris biometrics can be extracted from a high resolution picture of your face, which even a modern smartphone is capable of. Unless you spend your life wearing gloves and shades, there is no hope of your biometrics being secret. They are available to the people you encounter in daily life, just like your name is.
- **Unlike your name**, other people have no use for your biometrics and don’t pay attention to them, so we may be fooled into thinking they are secrets. They are not.
- Biometrics are not public either. There is no public database from which biometrics can be freely downloaded

What is privacy

- Privacy is about the responsible maintenance of private information. This responsibility is hard to define, which is why laws are necessary.

Private versus secret

Yiur Aadhar Number

- **Private,**
- **Neither secret nor public**

- Biometric Issues:
- Biometric matching gives probabilistic, not deterministic answers. That means the scanner will score your match on a scale of 0% to 100%. It cannot give a straightforward ‘yes’ or ‘no’ answer.
- Most Biometrics have been broken

Biometric Issues

- Biometric matching gives probabilistic, not deterministic answers. That means the scanner will score your match on a scale of 0% to 100%. It cannot give a straightforward ‘yes’ or ‘no’ answer.
- Most Biometrics have been broken

Authority versus authentication

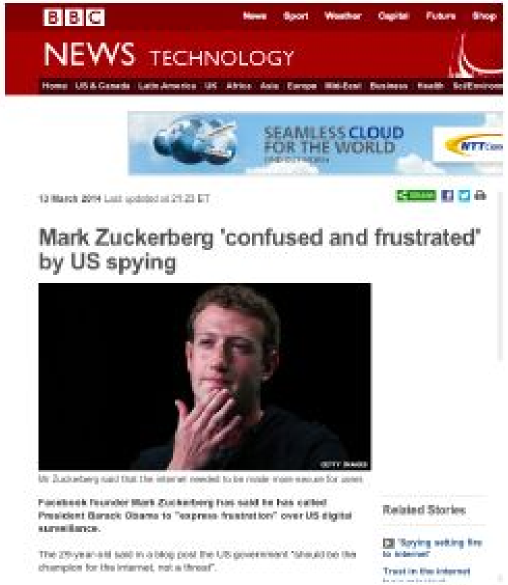
Biometric at say Immigration:
When you arrive at a foreign destination (or a foreigner arrives in India), the immigration official at the counter decides whether to let you in. The fingerprint scanner on the desk informs this official. The official is the authority, not the scanner or some remote server.

Biometric at Aadhaar: Implicit assumption that the official at the bank or mobile company cannot be trusted to certify your identity. The authority is with the Scanner/Server

Differential Privacy

- From Catuscia P

Leakage of information / privacy threats



BBC
News Sport Weather Capital Future Shop
NEWS TECHNOLOGY
Home US & Canada Latin America UK Africa Asia Europe Mid-East Business Health Sci/Environment

13 March 2014 Last updated at 21:23 ET

Mark Zuckerberg 'confused and frustrated' by US spying

Mr Zuckerberg said that the internet needed to be made more secure for users

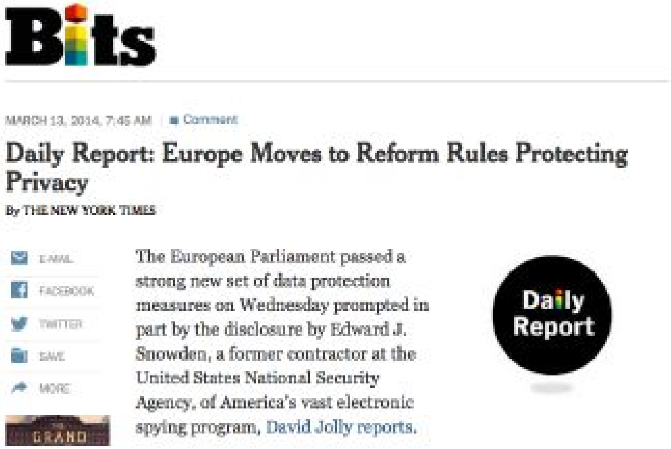
Facebook founder Mark Zuckerberg has said he has called President Barack Obama to "express frustration" over US digital surveillance.

The 29-year-old said in a blog post the US government "should be the champion for the internet, not a threat".

Related Stories

'Spying setting fire to internet'

Trust in the internet

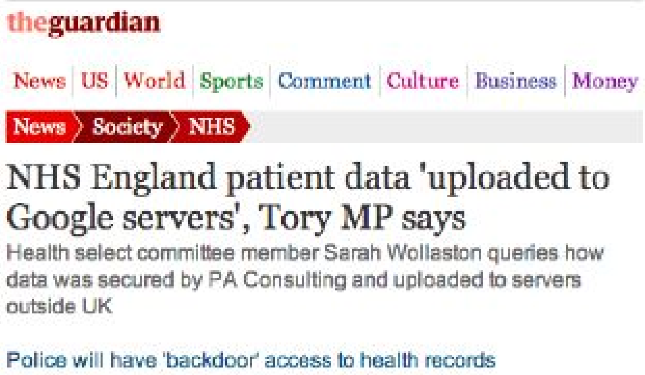


MARCH 13, 2014, 7:45 AM | Comment

Daily Report: Europe Moves to Reform Rules Protecting Privacy

By THE NEW YORK TIMES

The European Parliament passed a strong new set of data protection measures on Wednesday prompted in part by the disclosure by Edward J. Snowden, a former contractor at the United States National Security Agency, of America's vast electronic spying program, David Jolly reports.

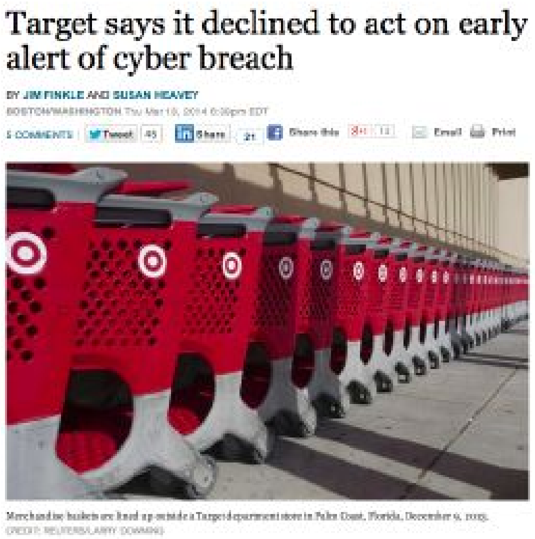


theguardian
News | US | World | Sports | Comment | Culture | Business | Money
News › Society › NHS

NHS England patient data 'uploaded to Google servers', Tory MP says

Health select committee member Sarah Wollaston queries how data was secured by PA Consulting and uploaded to servers outside UK

Police will have 'backdoor' access to health records



Target says it declined to act on early alert of cyber breach

BY JIM FINKLE AND SUSAN HEAVEY

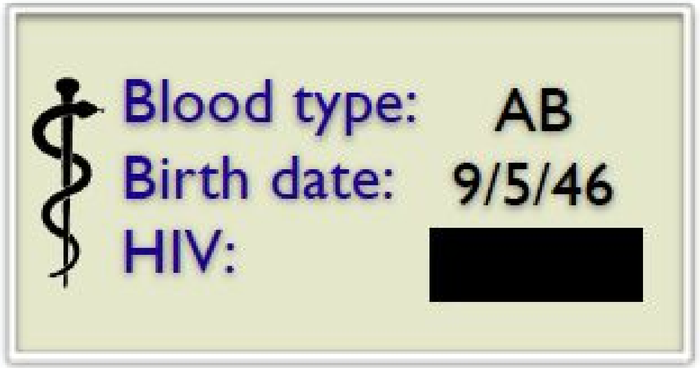
BOSTON/WASHINGTON Thu Mar 13, 2014 6:30pm EDT

5 COMMENTS | Tweet 45 | Share 21 | Share this | 8+1 13 | Email | Print

Merchandise baskets are lined up outside a Target department store in Palm Coast, Florida, December 9, 2013.
CREDIT: REUTERS/LARRY DOWNING

Protection of sensitive information

- Protecting the confidentiality of sensitive information is a fundamental issue in computer security



Blood type: AB

Birth date: 9/5/46

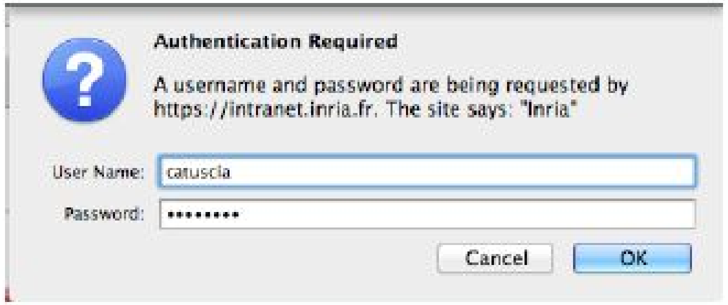
HIV: [redacted]

- Access control and encryption are not sufficient! Systems could leak secret information through the correlation with public information (observable).

- The notion of "observable" is subtle and crucial.
  - It depends on the power of the adversary
  - It may be combined from different sources

Leakage through correlated observables

Password checking

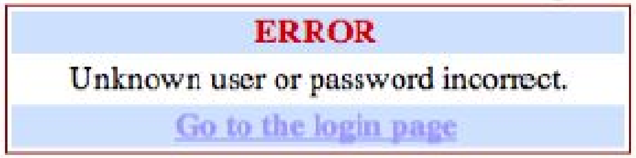


Authentication Required

A username and password are being requested by https://intranet.inria.fr. The site says: "Inria"

User Name: catuscia

Password: •••••••••

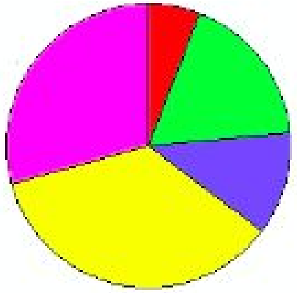


ERROR

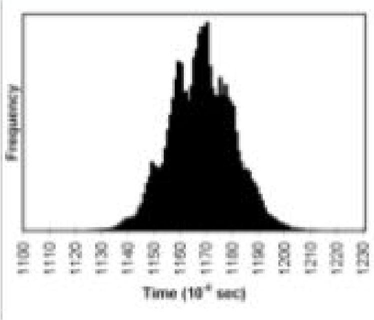
Unknown user or password incorrect.

Go to the login page

Election tabulation



Timings of decryptions



Reasoning about information leakage: Quantitative approaches

- It is usually impossible to prevent leakage completely. Hence we have to reason about the **amount** of leakage. This is usually related to the probability that the adversary discovers the secret

- Many methods to protect information use randomization to obfuscate the link between secret and observable. Hence the correlation itself may have a probabilistic nature.

Various notions of leakage

- The choice of an appropriate measure of leakage depends on many factors
- In particular, we need to choose whether to consider the **worst case**, or the **average** leak: individuals are usually interested in the first, while companies may prefer the second.

Differential Privacy

- Differential privacy [Dwork et al.,2006] is a notion of privacy originated from the area of **Statistical Databases**

- **The problem:** we want to use databases to get statistical information (aka aggregated information), but without violating the privacy of the people in the database

The problem

- Statistical queries should not reveal private information, but it is not so easy to prevent such privacy breach.

- Example: in a medical database, we may want to ask queries that help to figure the correlation between a disease and the age, but we want to keep private the info whether a certain person has the disease.

name	age	disease
Alice	30	no
Bob	30	no
Don	40	yes
Ellie	50	no
Frank	50	yes

Query:
What is the youngest age of a person with the disease?

Answer:
40

Problem:
The adversary may know that Don is the only person in the database with age 40

The problem

- Statistical queries should not reveal private information, but it is not so easy to prevent such privacy breach.

- Example: in a medical database, we may want to ask queries that help to figure the correlation between a disease and the age, but we want to keep private the info whether a certain person has the disease.

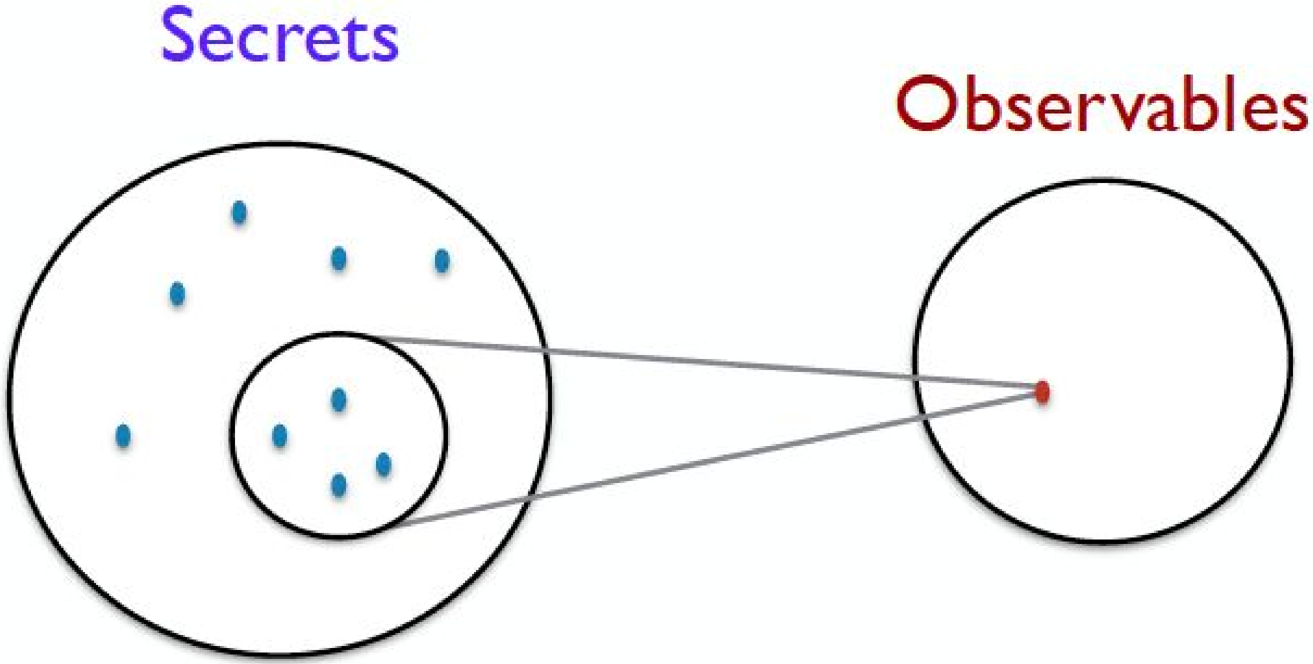
name	age	disease
Alice	30	no
Bob	30	no
Carl	40	no
Don	40	yes
Ellie	50	no
Frank	50	yes

k-anonymity: the answer always partition the space in groups of at least k elements

Alice	Bob
Carl	Don
Ellie	Frank

Correlation: Many-to-one

- Principle: Ensure that there are **many** secret values that correspond to **one** observable
- This is the general principle of most deterministic approaches to protection of confidential information (group anonymity, k-anonymity, ℓ-anonymity, cloacking, etc.)



The problem

Unfortunately, the many-to-one approach is not robust under **composition**:

name	age	disease
Alice	30	no
Bob	30	no
Carl	40	no
Don	40	yes
Ellie	50	no
Frank	50	yes

Alice	Bob
Carl	Don
Ellie	Frank

The problem of composition

Consider the query:

What is the minimal weight of a person with the disease?

Answer: 100

name	weight	disease
Alice	60	no
Bob	90	no
Carl	90	no
Don	100	yes
Ellie	60	no
Frank	100	yes

Alice	Bob
Carl	Don
Ellie	Frank

The problem of composition

Combine with the two queries:

minimal weight and the minimal age of a person with the disease

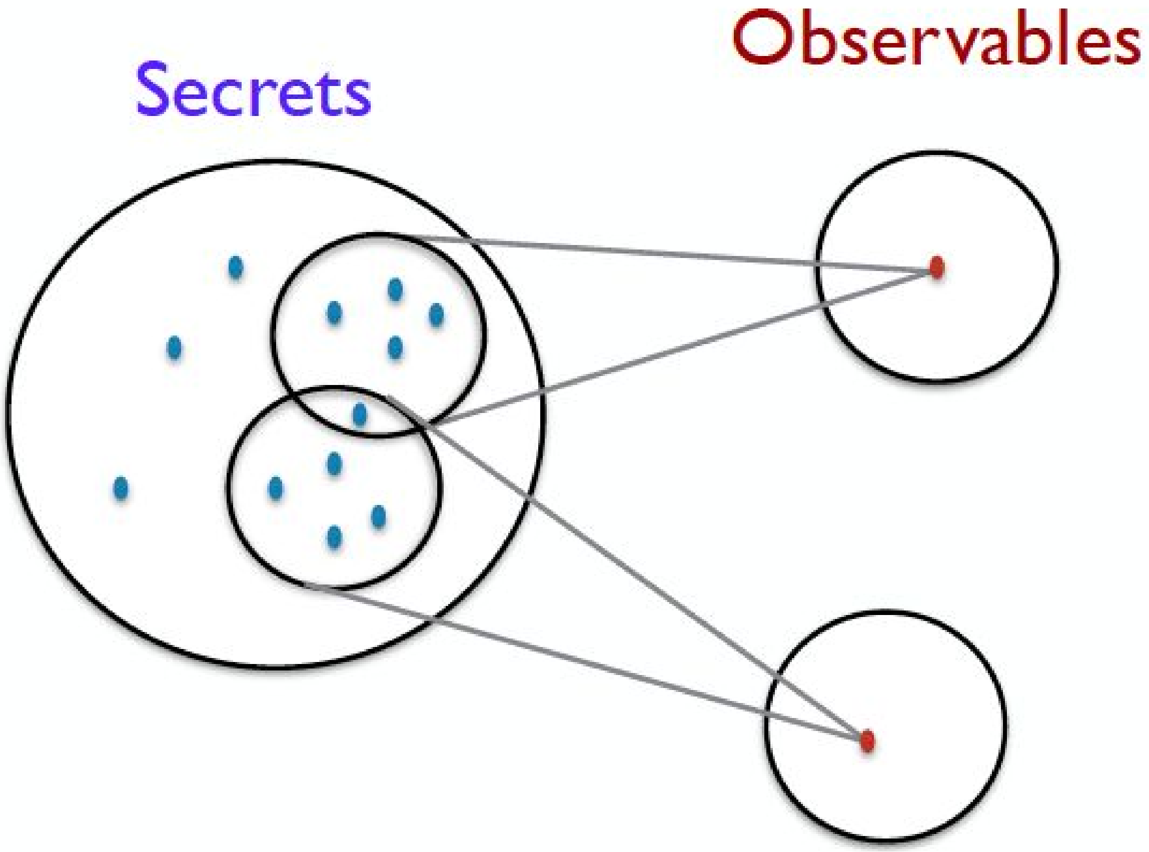
Answers: 40, 100

name	weight	disease
Alice	60	no
Bob	90	no
Carl	90	no
Don	100	yes
Ellie	60	no
Frank	100	yes

name	age	disease
Alice	30	no
Bob	30	no
Carl	40	no
Don	40	yes
Ellie	50	no
Frank	50	yes

Alice	Bob
Carl	Don
Ellie	Frank

This is a general problem of the deterministic approaches (based on the principle of many-to-one): the combination of observations determines smaller and smaller intersections on the domain of the secrets, and eventually result in singletones



Composition attacks

Composition attacks are real!
For instance, in a recent paper, Narayanan et Smatikov showed that by combining the information of two popular social network (Twitten and Flickr) they were able to de-anonymize a large percentage of the users (about 80%) and retrieve their private information with only a small probability of error (12%).

De-anonymizing Social Networks, Arvind Narayanan and Vitaly Shmatikov. Security & Privacy '09.

Solution

Introduce some probabilistic noise on the answer, so that the answers of minimal age and minimal weight can be given also by other people with different age and weight

name	weight	disease
Alice	60	no
Bob	90	no
Carl	90	no
Don	100	yes
Ellie	60	no
Frank	100	yes

name	age	disease
Alice	30	no
Bob	30	no
Carl	40	no
Don	40	yes
Ellie	50	no
Frank	50	yes

Alice	Bob
Carl	Don
Ellie	Frank

Noisy answers

minimal age:

40 with probability 1/2

30 with probability 1/4

50 with probability 1/4

name	age	disease
Alice	30	no
Bob	30	no
Carl	40	no
Don	40	yes
Ellie	50	no
Frank	50	yes

Alice	Bob
Carl	Don
Ellie	Frank

Noisy answers

minimal weight:

100	with prob. 4/7
90	with prob. 2/7
60	with prob. 1/7

name	weight	disease
Alice	60	no
Bob	90	no
Carl	90	no
Don	100	yes
Ellie	60	no
Frank	100	yes

Alice	Bob
Carl	Don
Ellie	Frank

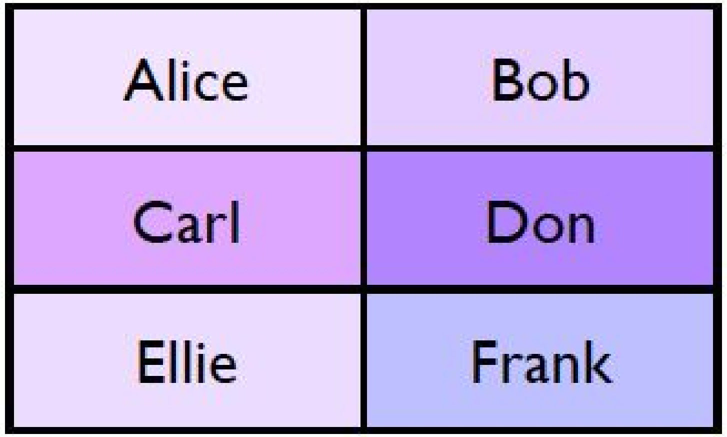
Noisy answers

Combination of the answers

The adversary cannot tell for sure whether a certain person has the disease

name	age	disease
Alice	30	no
Bob	30	no
Carl	40	no
Don	40	yes
Ellie	50	no
Frank	50	yes

name	weight	disease
Alice	60	no
Bob	90	no
Carl	90	no
Don	100	yes
Ellie	60	no
Frank	100	yes



Alice	Bob
Carl	Don
Ellie	Frank

Differential Privacy

- There have been various attempts to formalize the notion of privacy, but the most successful one is the notion of Differential Privacy, recently introduced by Dwork

- **Differential Privacy** [Dwork 2006]: a randomized function $\mathcal{K}$ provides **ε-** differential privacy if for all databases x, x' **which are adjacent** (i.e., differ for only one individual), and for all $z \in Z$, we have

$$\frac{p(K = z | X = x)}{p(K = z | X = x')} \leq e^{\epsilon}$$

- The idea is that the likelihoods of x and x' are not too far apart, for every S

- Differential privacy is robust with respect to composition of queries

- The definition of differential privacy is independent from the prior (but this does not mean that the prior doesn't help in breaching privacy!)

Differential Privacy: alternative characterization

- Perhaps the notion of differential privacy is easier to understand under the following equivalent characterization.
- In the following, Xi is the random variable representing the value of the individual i, and X≠i is the random variable representing the value of all the other individuals in the database

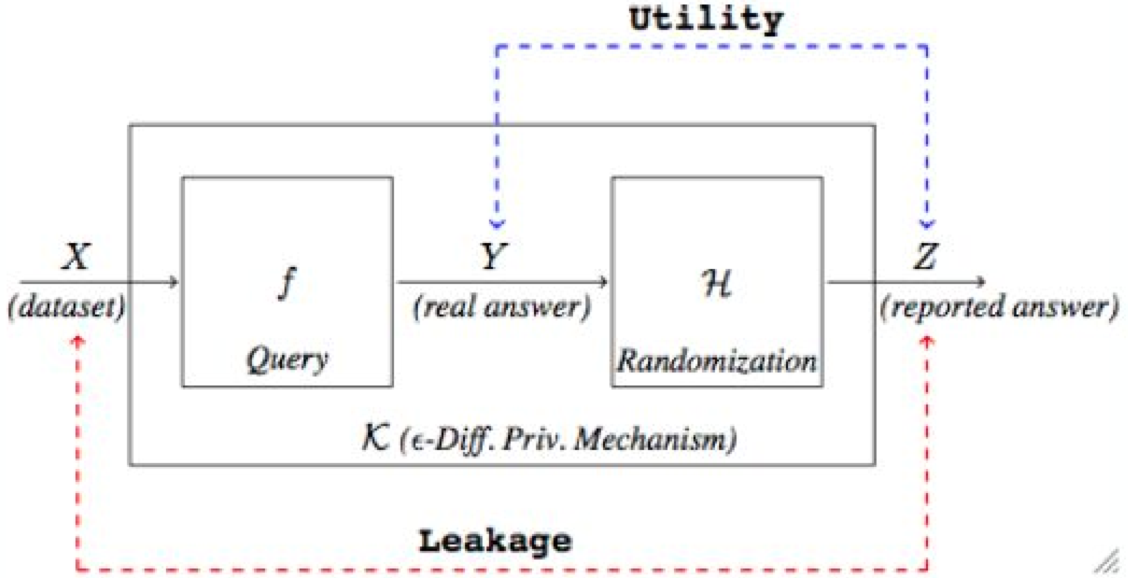
- **Differential Privacy, alternative characterization:** a randomized function $\mathcal{K}$ provides ε-differential privacy if and only if:

$$\text{for all } x \in \mathcal{X}, z \in \mathcal{Z}, p_i(\cdot)$$

$$\frac{1}{e^{\epsilon}} \leq \frac{p(X_i = x_i | X_{\neq i} = x_{\neq i})}{p(X_i = x_i | X_{\neq i} = x_{\neq i} \wedge K = z)} \leq e^{\epsilon}$$

Privacy and Utility

- The two main criteria by which we judge a randomized mechanism:
  - **Privacy:** how good is the protection against leakage of private information
  - **Utility:** how useful is the reported answer
- Clearly there is a trade-off between privacy and utility, but they are not the exact opposites: privacy is about the individual data, while utility is about the aggregate data.



Privacy and utility

- There may be other differences between privacy and utility, depending on the application domain: one may be worst-case and the other average-case, one may take into account the prior information and the other not, etc.

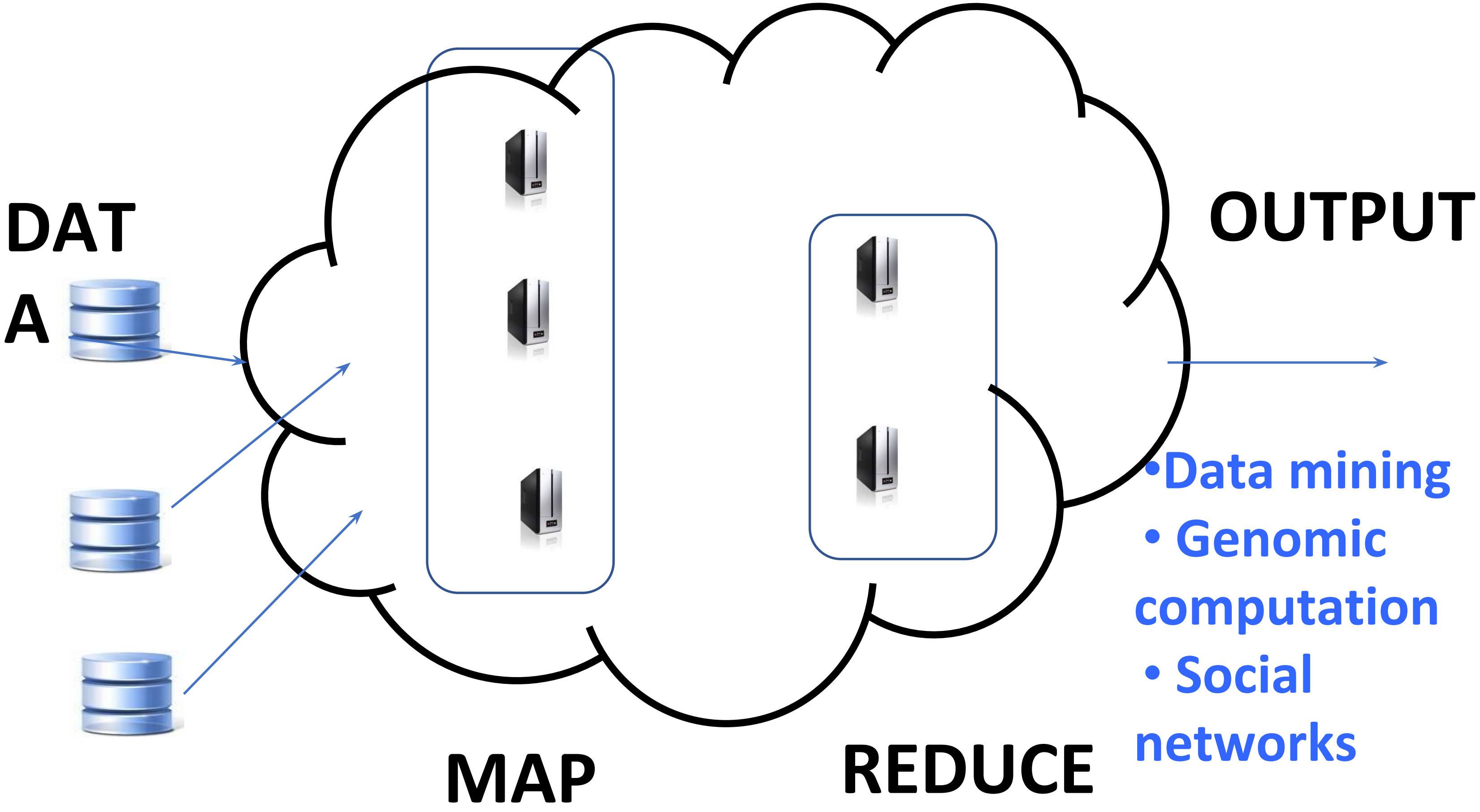
- The construction of mechanisms that optimize the trade-off between privacy and utility is an active field of research

SecHadoop: End-to-End Privacy Preserving Hadoop

Motivation

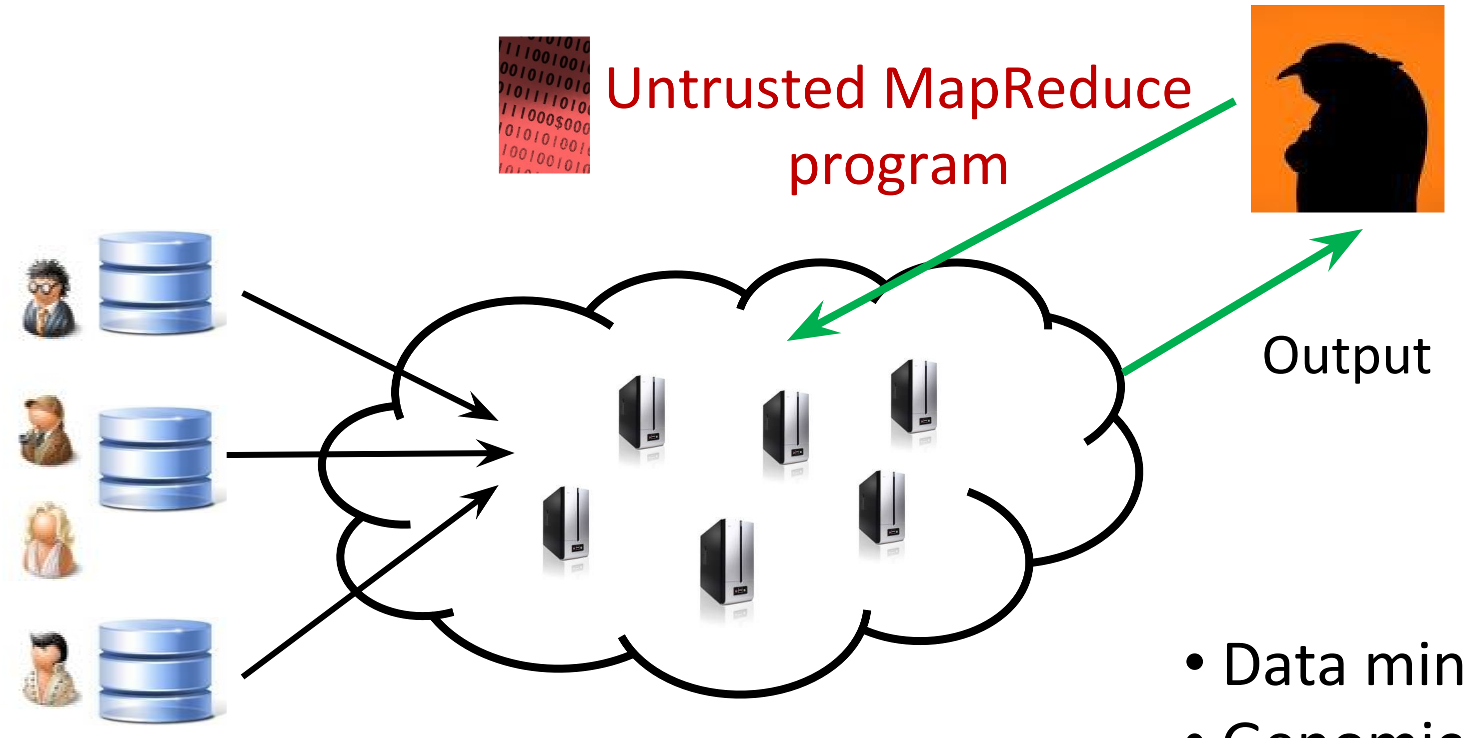
- Hadoop: Extensive scalable computations on massive data.
    - Achieves through Map-Reduce framework for computation and HDFS for distributed storage of the data.
- Privacy concerns arise due to
    - data divisions and intermediate data creations that is taking place while the computations are being carried out.
- User intervention in job execution in the form of Malware (or learning) in Hadoop can lead to privacy breaches.
    - Naturally requires a robust **decentralized information flow control**
- Secure end-to-end data flow in a Hadoop in a decentralized way preserving the original data privacy invariant
- Use such a scheme to protect against
    - Learning systems,
    - Desensitization of data without any leak,
    - Realize Provenance for Neurological data

MAP REDUCE



- Data mining
- Genomic computation
- Social networks

Impact of Untrusted Program



Information leak?

Untrusted MapReduce program

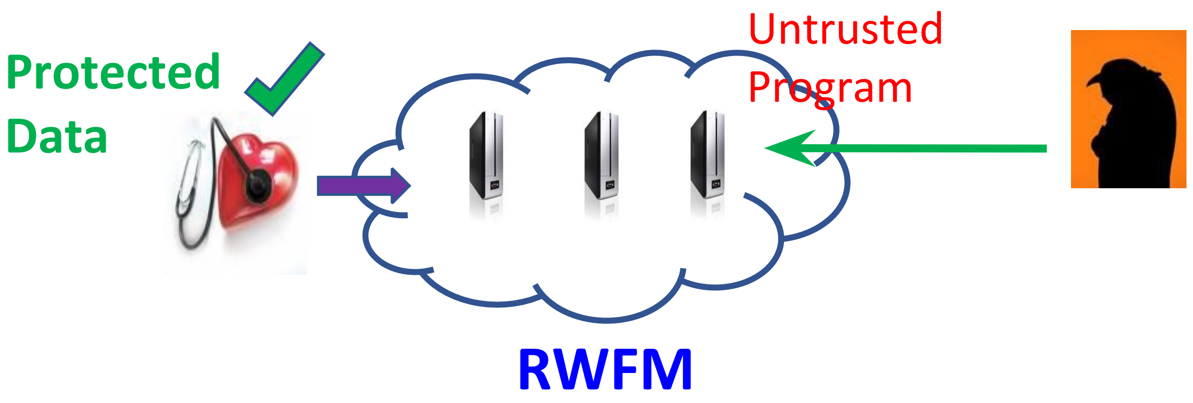
Output

Health Data

- Data mining
- Genomic computation
- Social networks

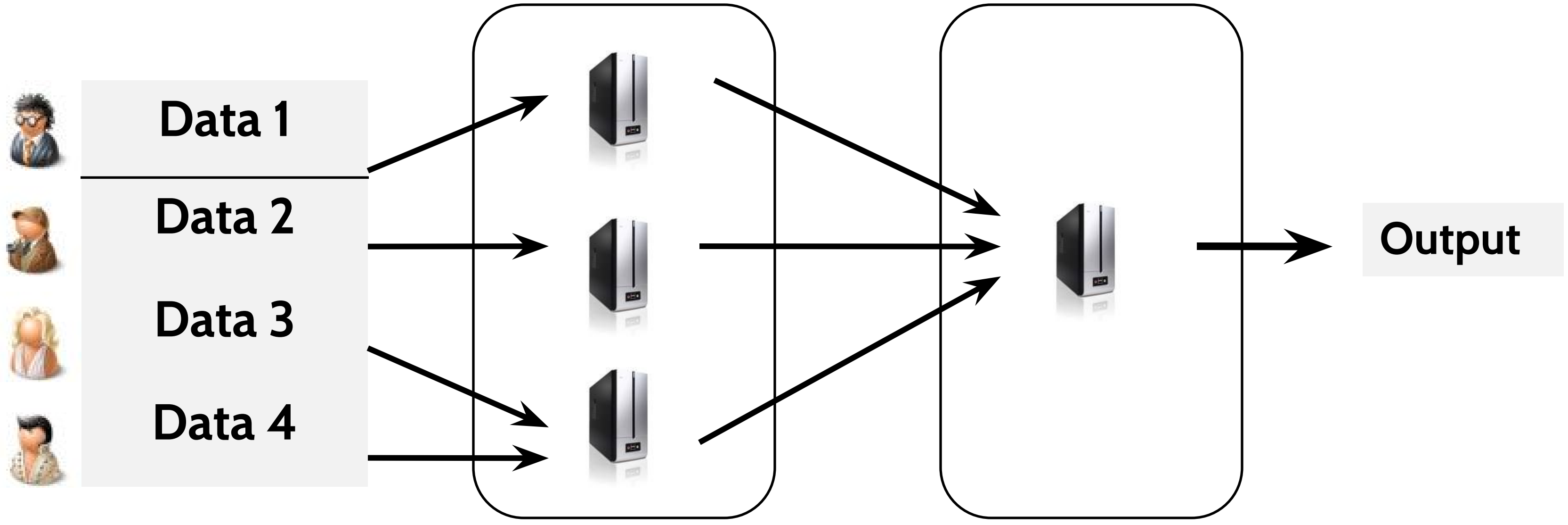
Realizing Privacy

Framework for privacy-preserving MapReduce computations with untrusted code.



Background: MapReduce

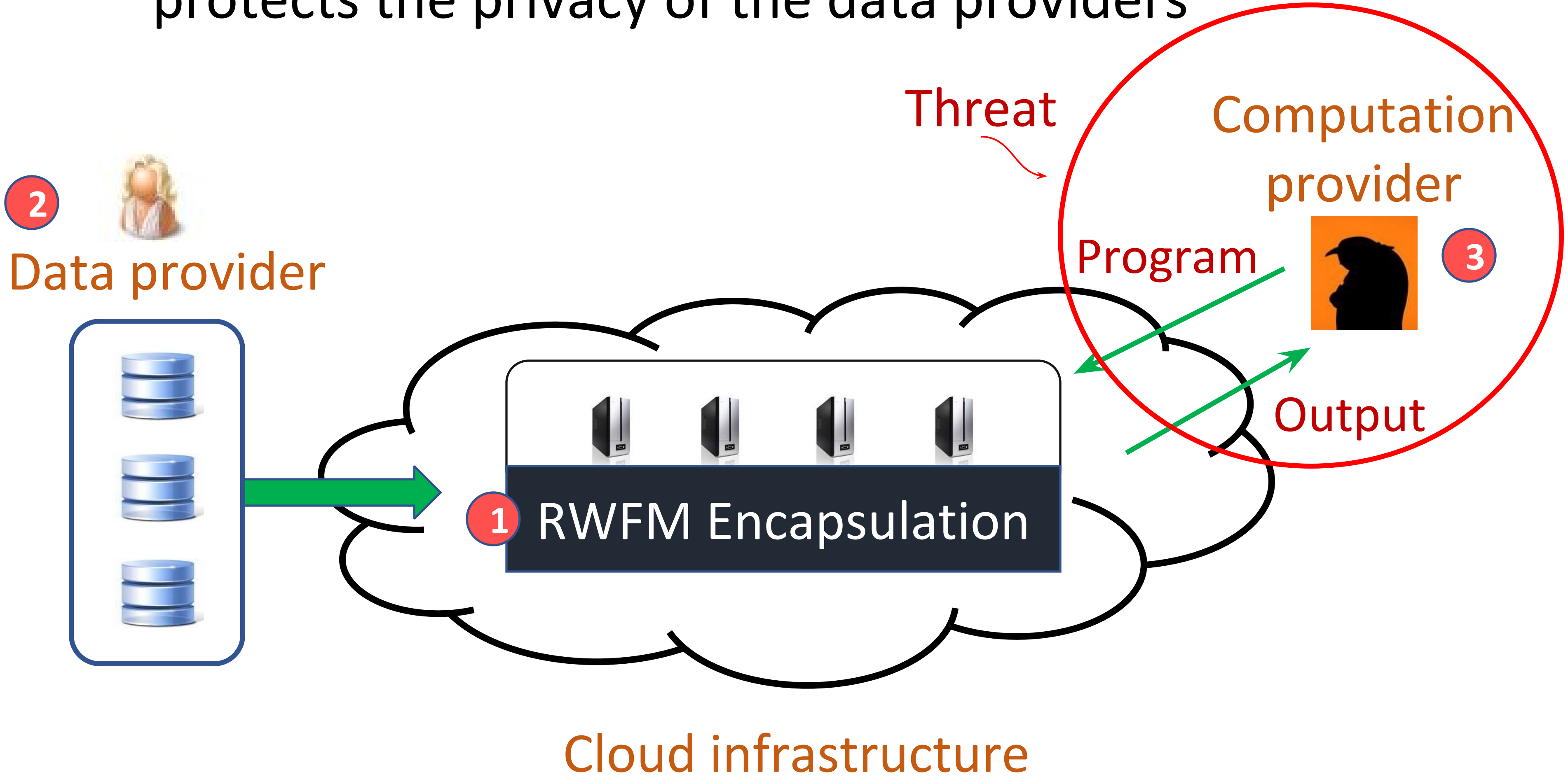
$$\text{map}(k_1,v_1) \rightarrow \text{list}(k_2,v_2)$$
$$\text{reduce}(k_2, \text{list}(v_2)) \rightarrow \text{list}(v_2)$$



Map phase Reduce phase

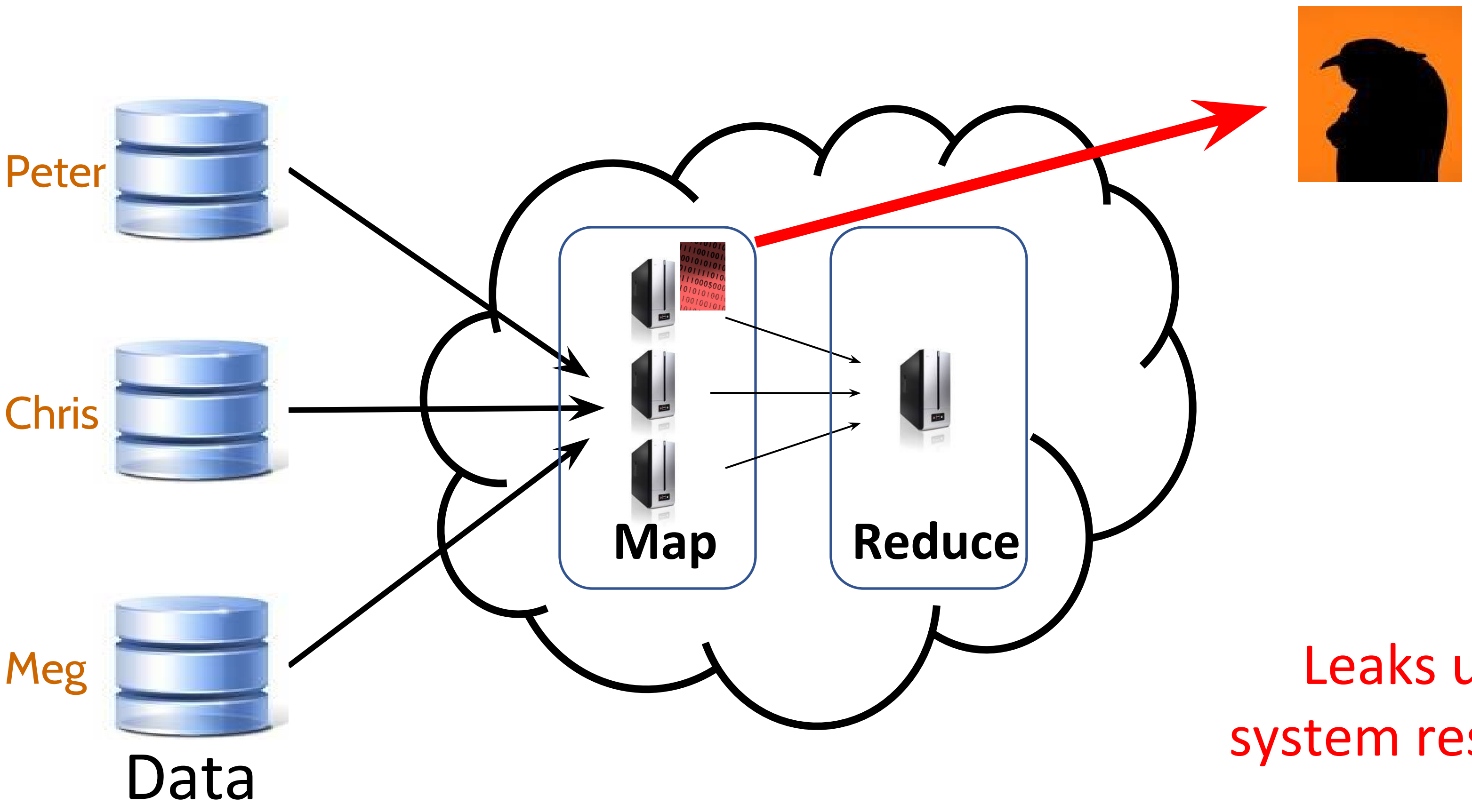
Threat model

- RWFM encapsulation monitors the computation, and protects the privacy of the data providers



Challenge 1: Untrusted mapper

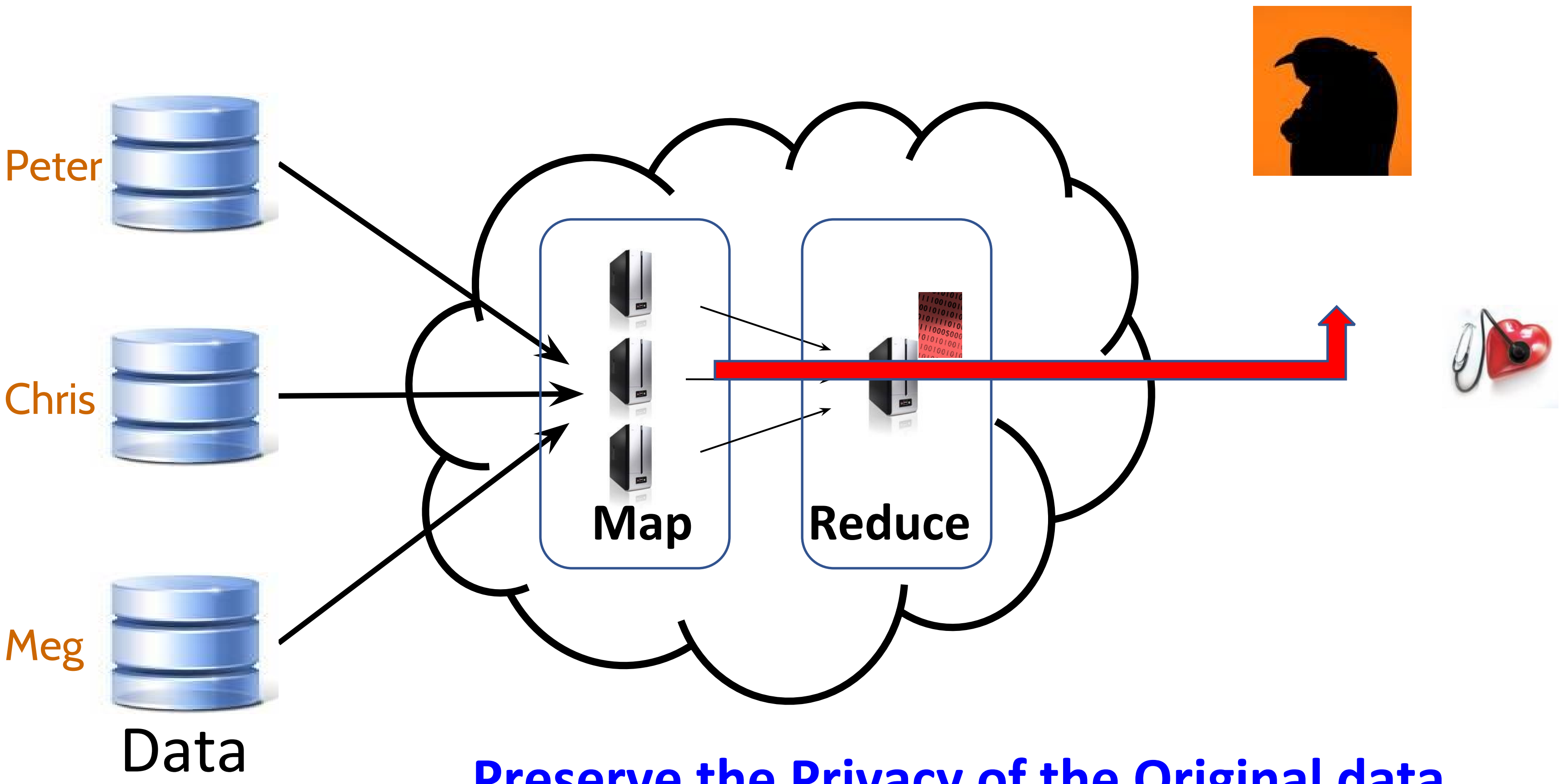
- Untrusted mapper code copies data, sends it over the network



Leaks using system resources

Challenge 2: Untrusted Reducer

- Output of the computation is also an information channel



Preserve the Privacy of the Original data

Approach 1: Airavat

Airavat*

- Airavat = SELinux (fixed set of syntactic labels) + "trusted" reducer + diff. privacy (add noise) -
- "reduce" provided by the user – difficult to trust
- SELinux ≠ full power of DIFC
- Differential policy will be difficult for dynamic evolving data
- RWFM = crisp combination of MAC (IFC) + DAC
    - Fine-grained labels preserve the privacy including that of the intermediate results without trust assumptions

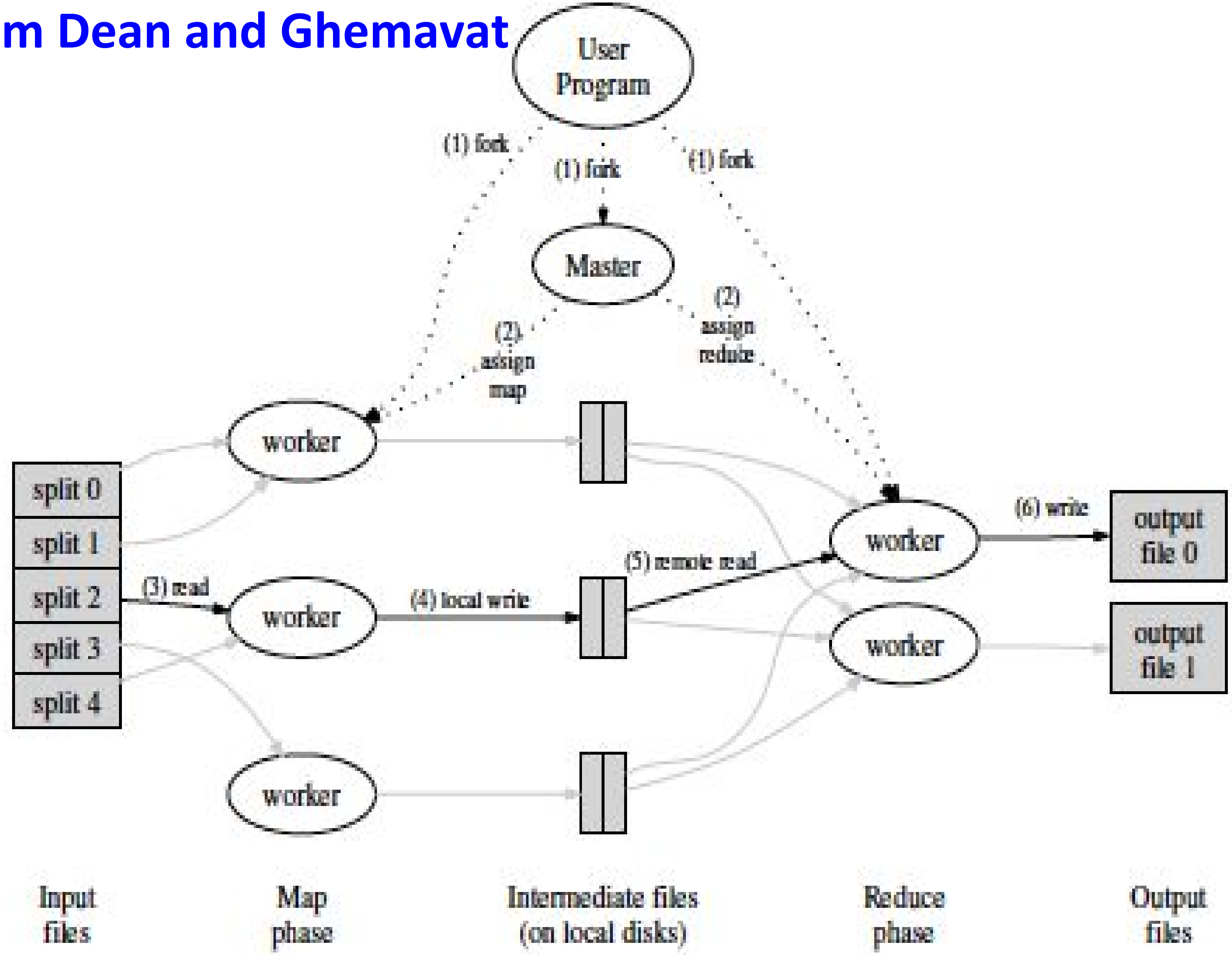
I. Roy, S. T. V. Setty, A. Kilzer, V. Shmatikov, and E. Witchel. **Airavat: Security and privacy for mapreduce**. In *7th USENIX NSDI, 2010*, pages 297–312.

MLS MapReduce*

- MLS MapReduce = SELinux (fixed set of syntactic labels) + different HDFS name nodes (appropriately linked) for different labels
- Rigid data storage structure, inefficient solution
- RWFM labels more fine-grained – new lattice points generated as appropriate, particularly useful when combining information at different security levels

T. D. Nguyen, M. A. Gondree, J. Khosalim, and C. E. Irvine. **Towards a cross-domain mapreduce framework**. In IEEE *MILCOM, 2013,* pages 1436–1441.

From Dean and Ghemavat



Status of Decentralized Information Flow Model

Decentralized Label Model

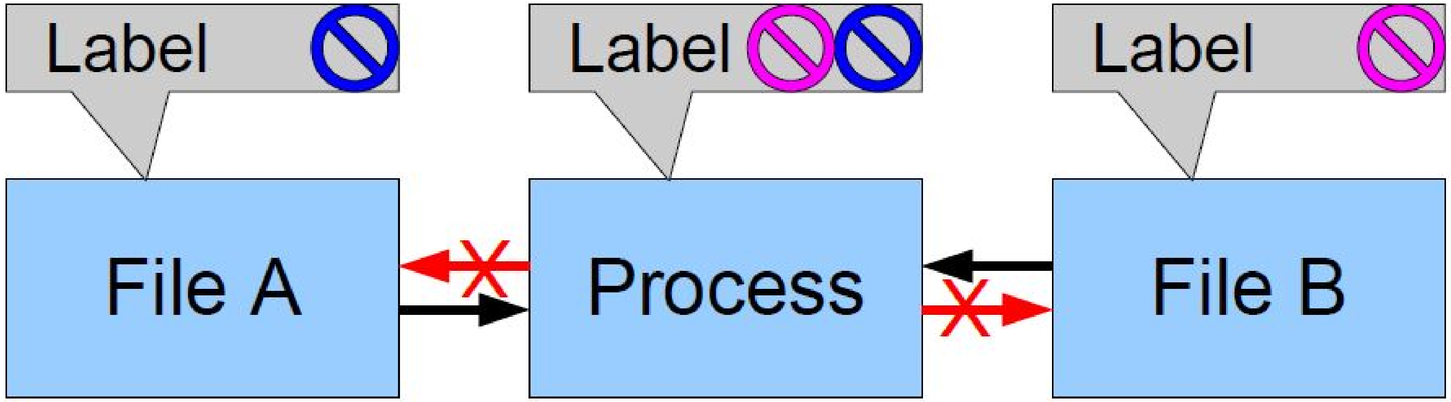
Myers and Liskov (2000)

- First model after **the seminal Lattice Information Flow model of Dorothy Denning (the lattice defines the flow)**
- Addresses the weaknesses of earlier approaches to the protection of confidentiality in a system containing untrusted code or users, even in situations of mutual distrust
- Allows users to control the flow of their information without imposing the rigid constraints of a traditional MLS
- Defines a set of rules that programs must follow in order to avoid leaks of private information
- Protects confidentiality for users and groups rather than for a monolithic organization
- Introduces a richer notion of declassification
  - in the earlier models it was done by a trusted subject; in this model principals can declassify their own data

Labels control information flow

🟦 Color is category of data (e.g. my files)

🛇 Blue data can flow only to other blue objects



Issues of State-of-the-art (a)

- 1985 Trusted Computer Systems Evaluation Criteria (Orange Book)
  - defines the security of a computer system by how well it **implements flow control** and how good its assurance is
- Despite huge efforts, systems developed had several drawbacks:
  - large TCB, slow, not easy to use, and very limited functionality

Issues of State-of-the-art (b)

- 2000 Myers & Liskov (DLM)
  - First Decentralized Label Model after 25 years (Myers and Liskov) – Cf. B Lampson
  - only readers for protecting confidentiality and only writers for protecting integrity
  - Issues: *for a proper tracking of any information flow property, it is important to control both reading and writing by subjects*

Issues of State-of-the-art (c)

- HiStar, Flume and Laminar systems
  - based on the product of Confidentiality and Integrity
  - **Issues**: *confidentiality and integrity are not orthogonal properties and issues of treating Declassification as a DAC*
  - Fred Schneider, in his book# chapter, clearly brings out the perils of combining confidentiality and integrity policies in this manner

# yet to be published,
available at http://www.cs.cornell.edu/fbs/publications/chptr.MAC.pdf

Issues of State-of-the-art (d)

- 2012 Mitchell et al. (DC labels)
  - not easy to derive consistent DC labels for modelling a given requirement
  - Flaw: *support for downgrading (discretionary control) is orthogonal to the IFC, thus, defeating the purpose of the mandatory controls*
- *New Robust decentralized Information Flow control model – RWFM (2016,2017) – Readers Writers Flow Model*

RWFM BASICS

RWFM Model

NV Narendra Kumar and RKs 2016, 2017

Readers-Writers Labels

- Security requirements of practical applications are often stated / easily understood in terms of who can read / write information
- Observations:
  - information readable by s_1 and $s_{2,}$ can-flow-to information readable only by s_1
  - information writable only by $s_{1,}$ can-flow-to information writable by s_1 and s_2
- Readers and writers can be used as labels!!

RWFM Label Format

- (owner/authority, readers, writers)
  - First component is a single subject denoting
    - *owner* in case of an object label
    - *authority* in case of a subject label
  - Second component is a set of subjects denoting
    - permissible readers in case of an object label
    - subjects who can read all the objects that this subject can read in case of a subject label
  - Third component is a set of subjects denoting
    - permissible writers in case of an object label
    - subjects who can write all the objects that this subject can write in case of a subject label

State of an Information System

- State of an information system is defined as the set of subjects and objects in the system together with their labels. Initial state
    - Objects and their labels as required for application
    - Each subject s starts with label (s,*,φ)
- Whenever a subject tries to perform an operation on an object, it may lead to a state change and will have to be permitted only if deemed safe
    - Read
    - Write
    - Create
    - Downgrade
    - Relabel

State Transitions in RWFM

- Subject s with label (s_1,R_1,W_1) requests ***read*** access to an object
  - If $s_1 \in R_2$ the
    - relabel s to $(s_1,R_1 \cap R_2,W_1 \cup W_2$ AL ccess
  - Else
    - DENY access
- <u>POSSIBLE</u> state change (label of s may change)

s can

s has accessed information accessible only by

s is influenced by both W_1 and W_2

State Transitions in RWFM

- Subject s wit s to an object
  - If $s_1 \in W_2$ and R_2 and $\in W_2$ then
    - ALLOW access
  - Else
    - DENY access
- <u>NO</u> state change

$\supseteq$

s can write o

all subjects can access

all subjects that have influenced the current information of s can also influence o

State Transitions in DWFM

s, and all subjects that have influenced the current information of s have influenced o

accessed by s so far, ca

- Subject s with label (s,R,W) reque *ation* of an object o
  - create an object o and label it (s,R,W ∪ {s})

- DEFINITE state change (a new object is added to the system)

State Transitions in RWFM

- Subject s t o with la bel (s_3,R_3,W

subjects that could not access o but can access its downgraded version must have influenced information in o

all the subjects s o can access its downgra on also

subjects that can access o can also access

  - If $s_1 \in R_2$ and $s_1=s_2=s_3$ and $W_1=W_2=W_3$ and $R_1=R_2$ and $R_3 \supseteq R_2$ and $R_3-R_2 \subseteq W_2$ then
    - ALLOW
  - Else
    - DENY
- POSSIBLE state change (label of o may change)

State Transitions in RWFM

s, and all subjects that influenced the current information of s have influenced the relabelling

- Subject s with la t o
  w

all subjects that can access the relabelled object, could have accessed all the information that s has accessed so far, and the original object

- If $s_1 \in R_2$ and s_1 d $W_2 \subseteq W_1$ and W_3 $W_1 \cup \{s\}$ and R_2 R_1 R_3 then
  - ALLOW
- Else $\supseteq$ $\supseteq$
  - DENY

- POSSIBLE state change (label of o may change)

Downgrading (Declassifying)

- For practical applications, adding readers (downgrading) to the result of a computation is essential for use by relevant parties
- Downgrading rules
  - only the owner of information may downgrade it
  - if a single source is responsible for the information, then readers that can be added is unrestricted
  - if multiple sources influenced the information, then only those who influenced it may be added as readers

RWFM permits intuitive specifications with simple access checks

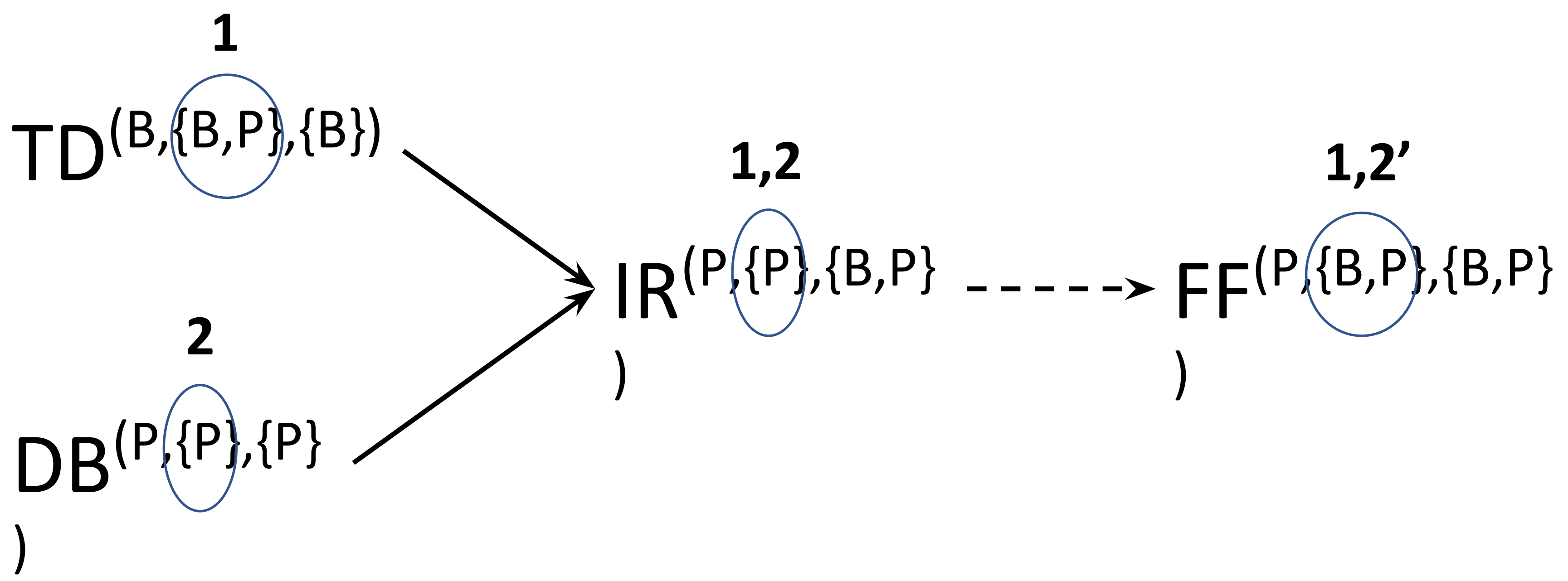
- The above proposition simplifies the access check to $s \in R(o)$ for subject s to read object o and $s \in W(o)$ for subject s to write object o.

Example-1
WebTax

- Bob provides his tax-data to a professional tax preparer, who computes Bob's final tax form using a private database of rules for minimizing the tax payable and returns the final form to Bob
- Security requirements
  1. Bob requires that his tax-data remains confidential
  2. Preparer requires that his private database remains confidential

Example-1

WebTax



1

$TD^{(B,\{B,P\},\{B\})}$

2

$DB^{(P,\{P\},\{P\}}$
)

1,2

$IR^{(P,\{P\},\{B,P\}}$
)

1,2’

$FF^{(P,\{B,P\},\{B,P\}}$
)

TD	Tax-data	IR	Intermediate results
DB	Database of tax optimization rules	FF	Final tax form
⟶	Flows-to	- -➤	Downgraded-to

Example-1
WebTax

	DLM	DC	RWFM
TD	{B: B}	(B, B)	(B, {B,P}, {B})
DB	{P: P}	(P, P)	(P, {P}, {P})
IR	{B: B; P: P}	(B $\wedge$ P, B $\vee$ P)	(P, {P}, {B,P})
FF	{B: B}	(B, B $\vee$ P)	(P, {B,P}, {B,P})

- DLM label format: policies separated by ‘;’, where each policy is of the form ‘owner: readers’
- DC label format: ‘readers, writers’, where readers control confidentiality, writers control integrity
- RWFM label format: ‘owner, readers, writers’

DLM, DC and RWFM Comparison

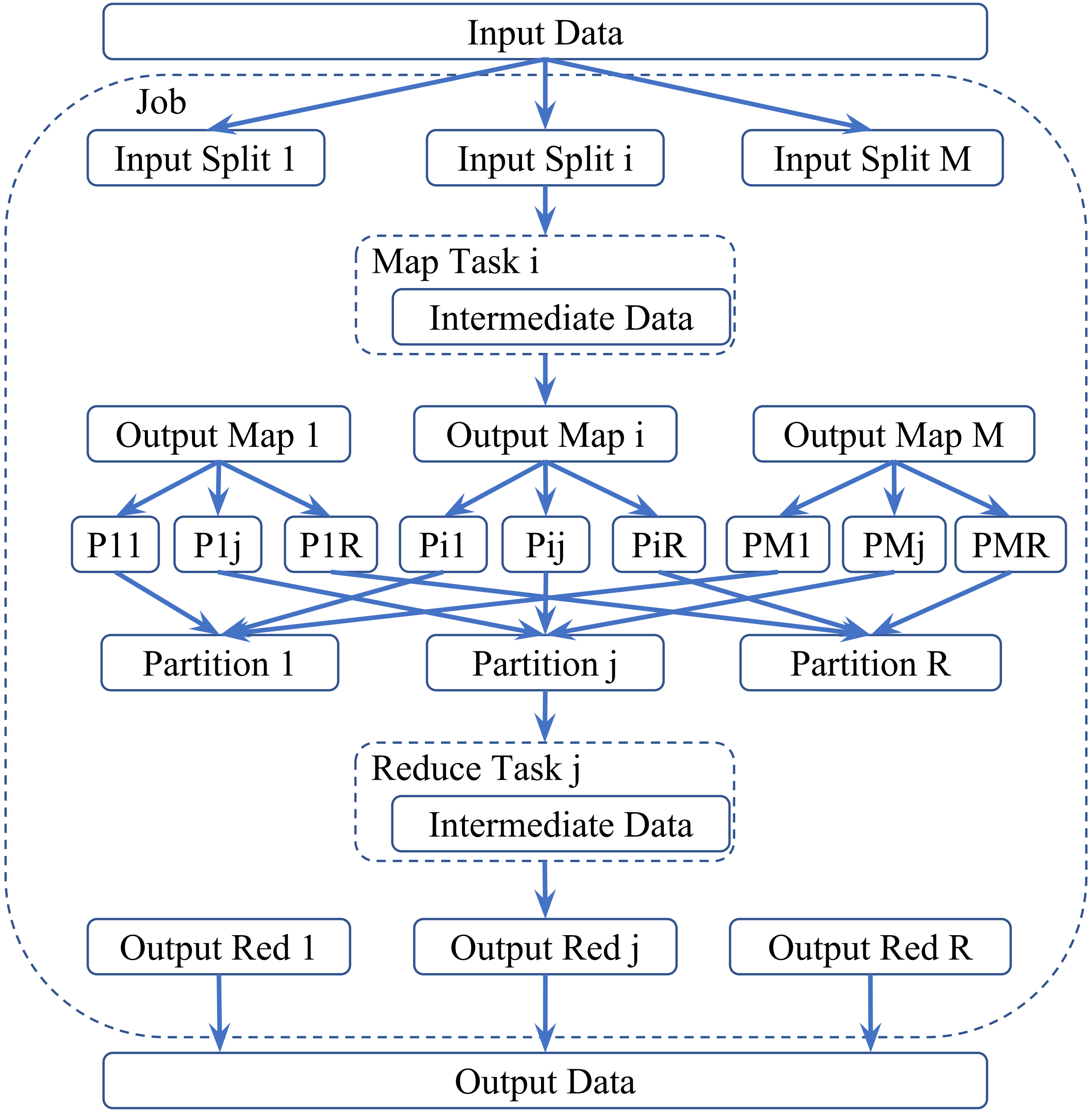
	DLM	DC	RWFM
Confidentiality	only Readers	only Readers	Readers and Writers
Integrity	only Writers	only Writers	Readers and Writers
Downgrading (DAC)	Purely discretionary	Purely discretionary	Consistent with IFC (MAC)
Ownership	Explicit	Implicit	Explicit
Authority	Orthogonal to the label	Orthogonal to the label	Explicit in the label

DLM, DC and RWFM Comparison

	DLM	DC	RWFM
Principal hierarchy and Delegation	Orthogonal to the label	Orthogonal to the label	Embedded in the label
Bi-directional flow	Difficult	Difficult	Simple and Accurate
Ease of use	Moderate	Moderate	Easy
Label size	Moderate to Large	Large	Small
No. of labels	Large	Large	Small (as required by the application)

Labelling Map Reduce Framework

Flow of a MapReduce Job



Flow of a MapReduce Job

1. The job tracker splits input data, and creates and assigns map tasks
2. Map tasks execute on slave nodes to produce intermediate results
3. Job tracker partitions (shuffles and sorts) the intermediate results and assigns reduce tasks
4. Reduce tasks execute on slave nodes to produce final results
5. Job tracker aggregates the final results and produces the output for the user

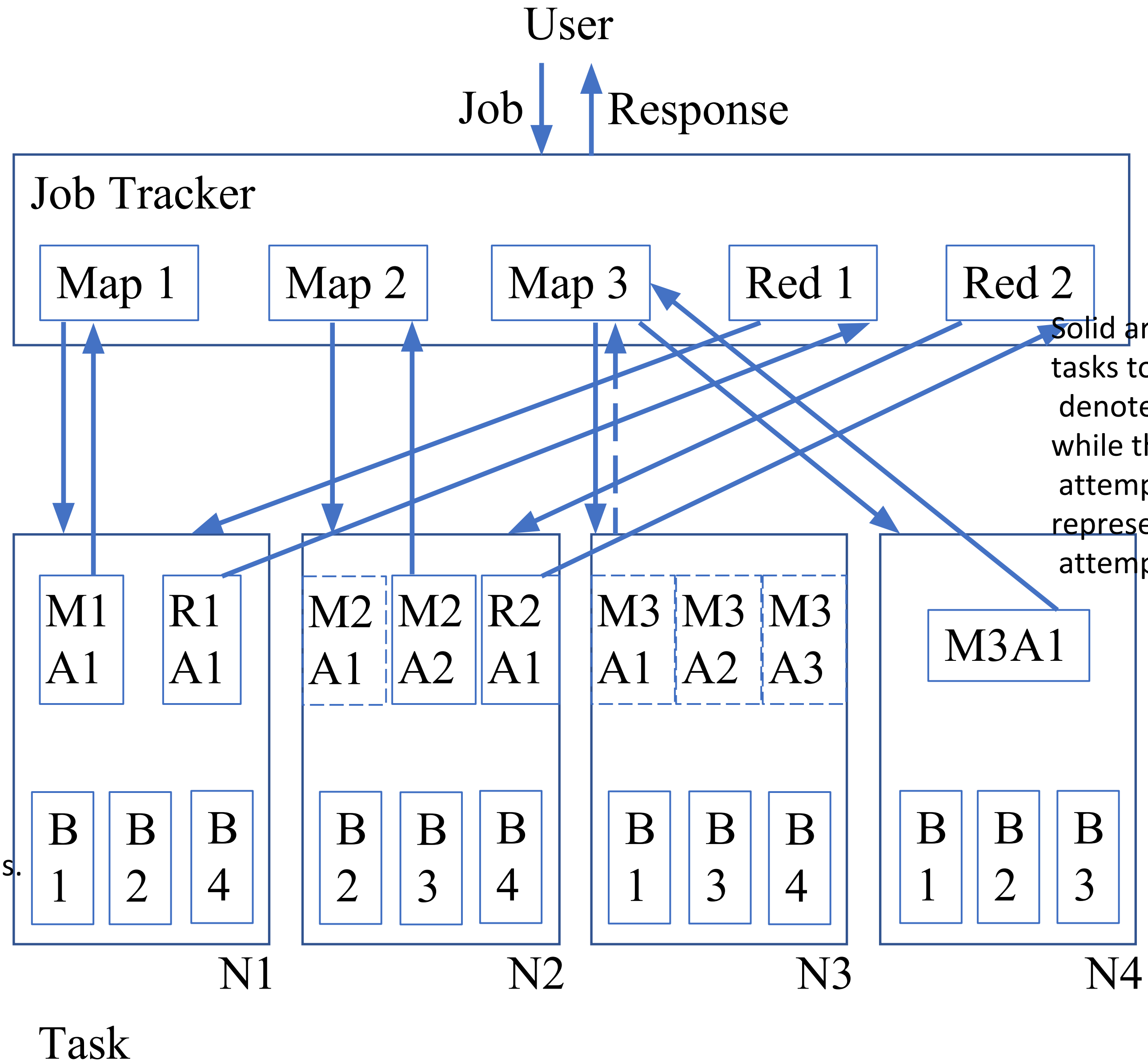
Example Configuration of MapReduce

Dotted arrows from nodes to tasks represent failure of execution of the task on the node– happens due to data corruption

Boxes with dashes outlines represent failed attempts . E.g.,, attempt 1 of map 2 (M2A1) fails on N2, which then creates M2A2

which succeeds.

Shaded boxes represent data stored on the nodes. E.g., blocks 1, 2 and 4 are stored on node N1.

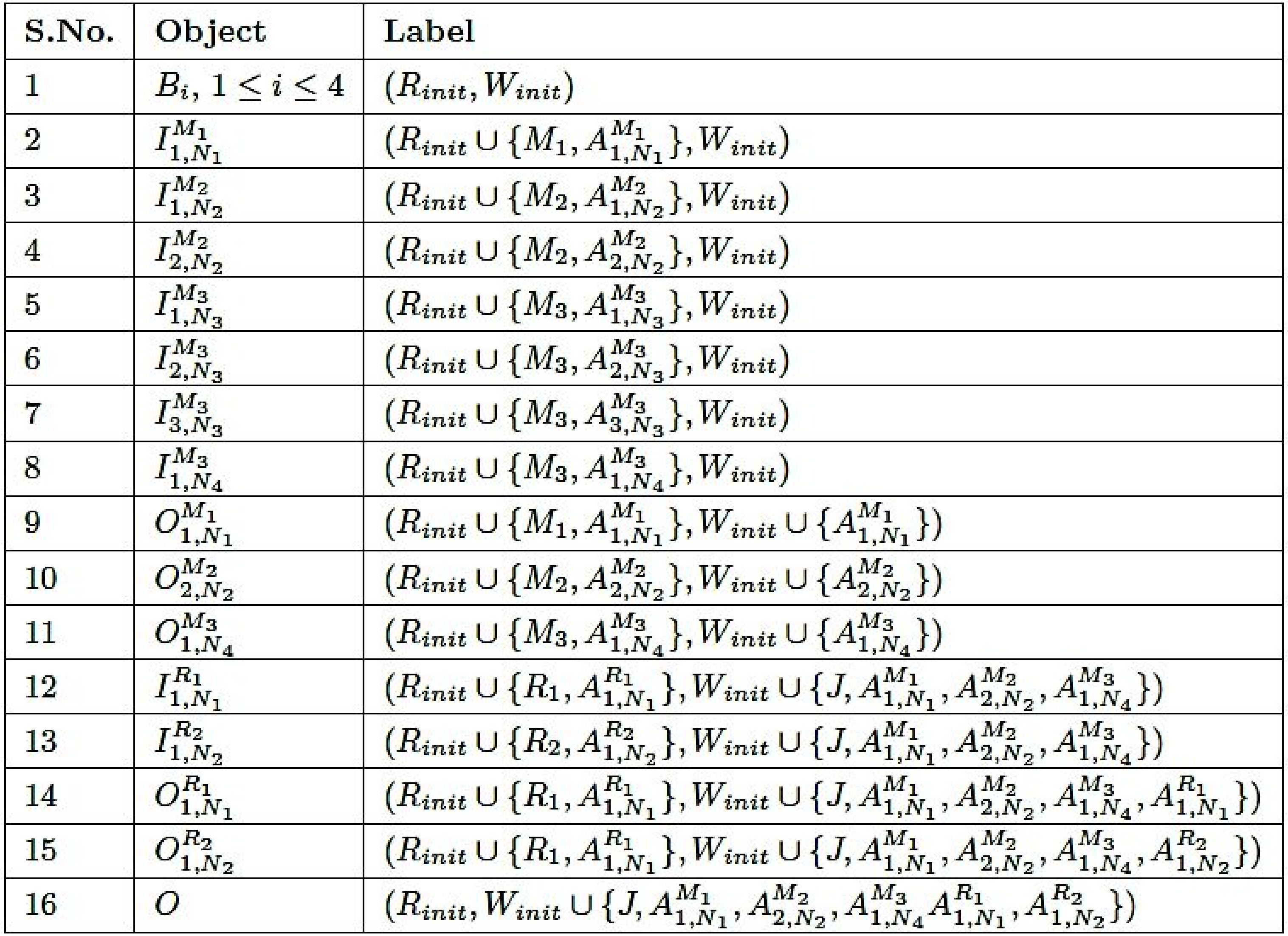


Solid arrows from tasks to nodes denote assignments, while those from attempts to tasks represent successful attempts.

Notation

1. Shaded boxes represent data stored on the nodes. For example, blocks 1, 2 and 4 are stored on node N1.
2. Boxes with dashes outlines represent failed attempts. For example, attempt 1 of map 2 (M2A1) fails on N2, which then creates M2A2 which succeeds.
3. Solid arrows from tasks to nodes denote assignments, while those from attempts to tasks represent successful attempts.
4. Dotted arrows from nodes to tasks represent failure of execution of the task on the node. This happens – potentially due to data corruption – when a threshold number of attempts of a task on a node fail. For example, task map 3 fails on node 3.

Labels for Example Configuration



S.No.	Object	Label
1	$B_i, 1 \leq i \leq 4$	(R_{init}, W_{init})
2	$I^{M_1}_{1,N_1}$	$(R_{init} \cup \{M_1, A^{M_1}_{1,N_1}\}, W_{init})$
3	$I^{M_2}_{1,N_2}$	$(R_{init} \cup \{M_2, A^{M_2}_{1,N_2}\}, W_{init})$
4	$I^{M_2}_{2,N_2}$	$(R_{init} \cup \{M_2, A^{M_2}_{2,N_2}\}, W_{init})$
5	$I^{M_3}_{1,N_3}$	$(R_{init} \cup \{M_3, A^{M_3}_{1,N_3}\}, W_{init})$
6	$I^{M_3}_{2,N_3}$	$(R_{init} \cup \{M_3, A^{M_3}_{2,N_3}\}, W_{init})$
7	$I^{M_3}_{3,N_3}$	$(R_{init} \cup \{M_3, A^{M_3}_{3,N_3}\}, W_{init})$
8	$I^{M_3}_{1,N_4}$	$(R_{init} \cup \{M_3, A^{M_3}_{1,N_4}\}, W_{init})$
9	$O^{M_1}_{1,N_1}$	$(R_{init} \cup \{M_1, A^{M_1}_{1,N_1}\}, W_{init} \cup \{A^{M_1}_{1,N_1}\})$
10	$O^{M_2}_{2,N_2}$	$(R_{init} \cup \{M_2, A^{M_2}_{2,N_2}\}, W_{init} \cup \{A^{M_2}_{2,N_2}\})$
11	$O^{M_3}_{1,N_4}$	$(R_{init} \cup \{M_3, A^{M_3}_{1,N_4}\}, W_{init} \cup \{A^{M_3}_{1,N_4}\})$
12	$I^{R_1}_{1,N_1}$	$(R_{init} \cup \{R_1, A^{R_1}_{1,N_1}\}, W_{init} \cup \{J, A^{M_1}_{1,N_1}, A^{M_2}_{2,N_2}, A^{M_3}_{1,N_4}\})$
13	$I^{R_2}_{1,N_2}$	$(R_{init} \cup \{R_2, A^{R_2}_{1,N_2}\}, W_{init} \cup \{J, A^{M_1}_{1,N_1}, A^{M_2}_{2,N_2}, A^{M_3}_{1,N_4}\})$
14	$O^{R_1}_{1,N_1}$	$(R_{init} \cup \{R_1, A^{R_1}_{1,N_1}\}, W_{init} \cup \{J, A^{M_1}_{1,N_1}, A^{M_2}_{2,N_2}, A^{M_3}_{1,N_4}, A^{R_1}_{1,N_1}\})$
15	$O^{R_2}_{1,N_2}$	$(R_{init} \cup \{R_1, A^{R_1}_{1,N_1}\}, W_{init} \cup \{J, A^{M_1}_{1,N_1}, A^{M_2}_{2,N_2}, A^{M_3}_{1,N_4}, A^{R_2}_{1,N_2}\})$
16	O	$(R_{init}, W_{init} \cup \{J, A^{M_1}_{1,N_1}, A^{M_2}_{2,N_2}, A^{M_3}_{1,N_4} A^{R_1}_{1,N_1}, A^{R_2}_{1,N_2}\})$

RKS and NV Kumar, 2016)

Security Properties Assured by the Labelling

- Privacy Invariance: security and privacy reqs on the inputs are maintained as an invariant throughout the computation including the intermediate data that is produced in the process

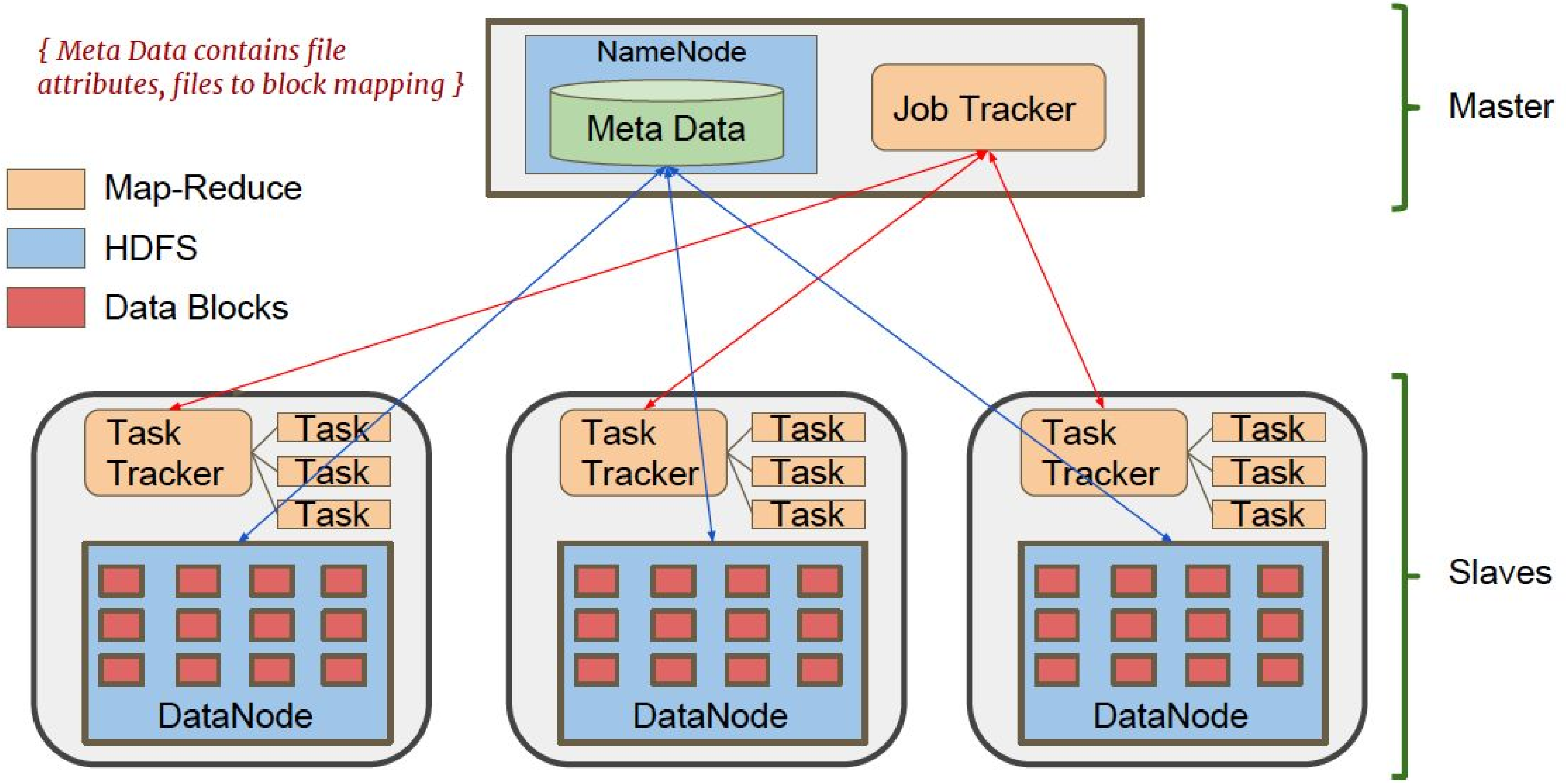
Security Properties Assured by the Labelling

- Protection from Malware: map and reduce are provided by the user and may be malicious, yet the attempt executing these tasks cannot access any data on the node other than the data provided as its input

Security Properties Assured by the Labelling

- Non-interference Free Execution: the attempts (could be of tasks of the same job or not) executing simultaneously on a give node are isolated due to labelling, and therefore cannot interfere with one another

Hadoop Architecture



Hadoop source is huge and complex which consists of 2.3 MLOC

Challenges in Implementation

CRUX

- Build an RWFM monitor to control the information flow
- Integrate DAC of HADOOP with the Information Flow labels of RWFM

Hadoop: Distributed Computing Infrastructure for Big Data Computations

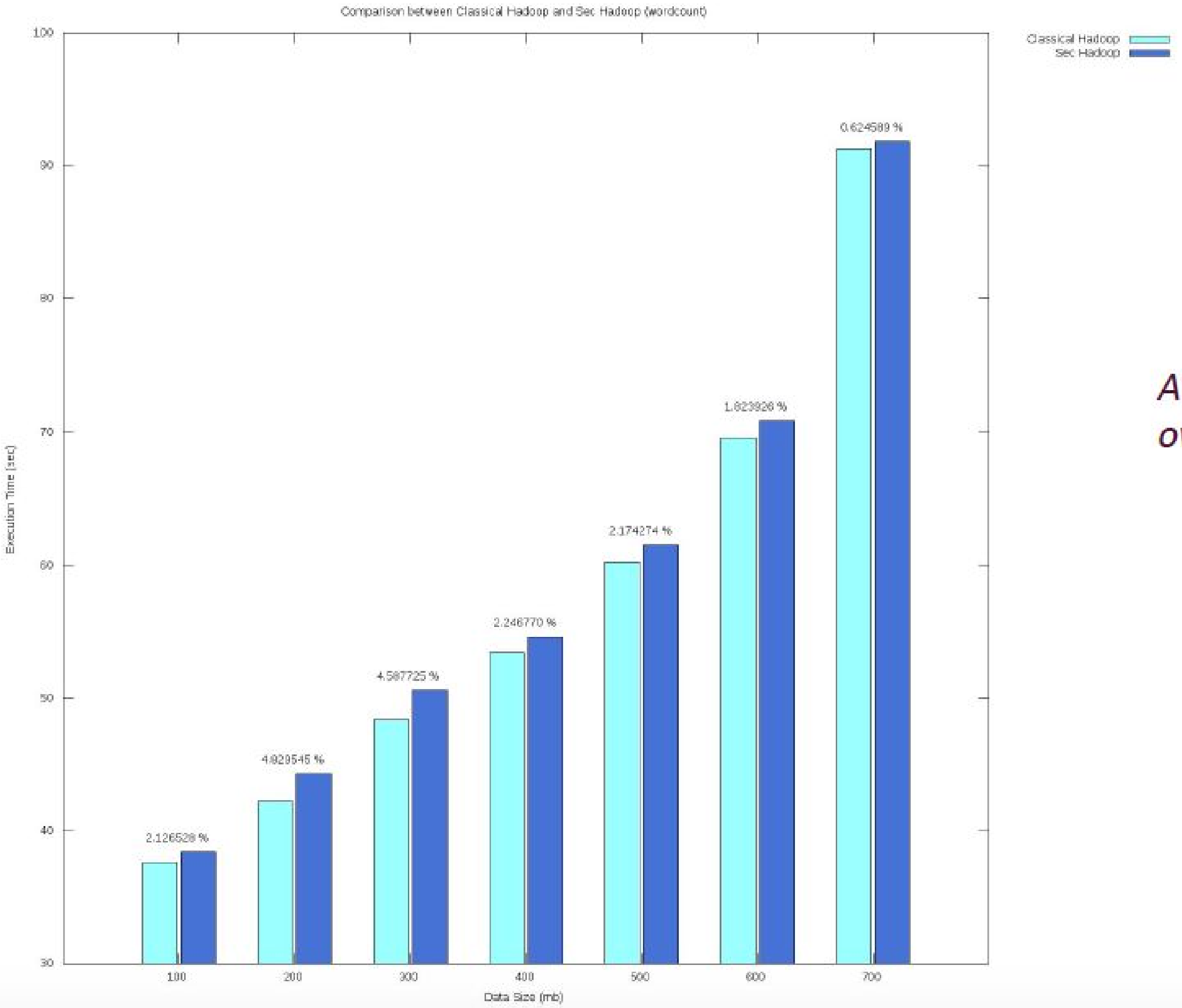
Hadoop Modules:

- Hadoop Distributed File System (HDFS)
  - Stores user data in files and provides redundancy for high availability.
- MapReduce Framework
  - Processes problems parallely on large data sets with large number of nodes.
  - Prefers locality of data, minimizes network congestion, increases overall throughput.
  - Advantages : Scalability, Fault Tolerance
- Identify possible points of leakage in the Hadoop System

Performance Results

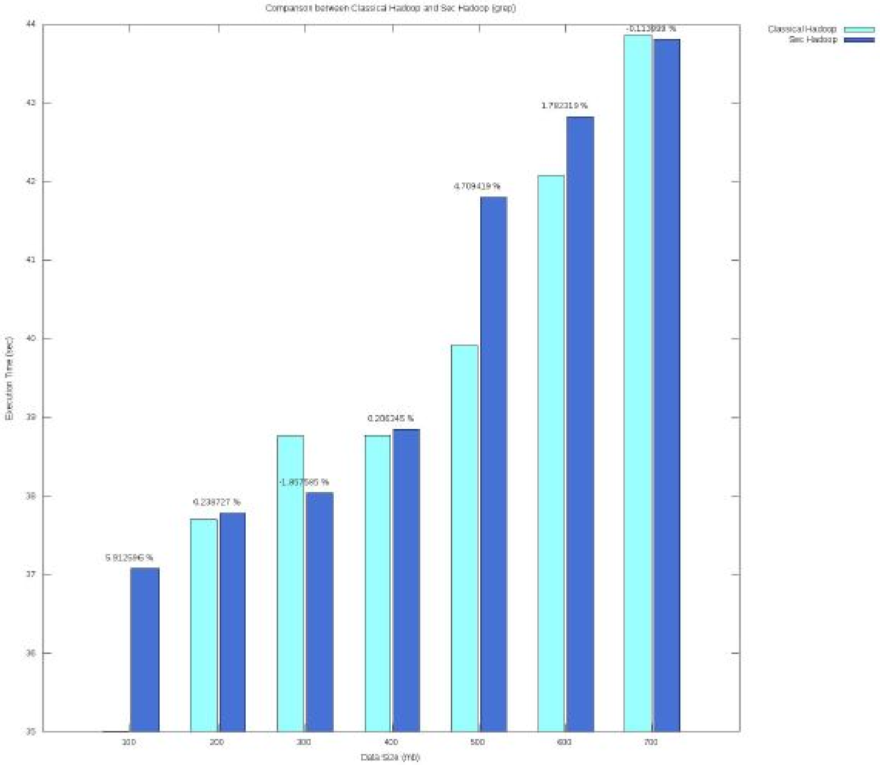
- Comparison between the performance of Classical Hadoop and SecHadoop by increasing input file size
- Performance overhead of SecHadoop is 2-5% more in comparison to Classical

Performance Comparison for WordCount Job



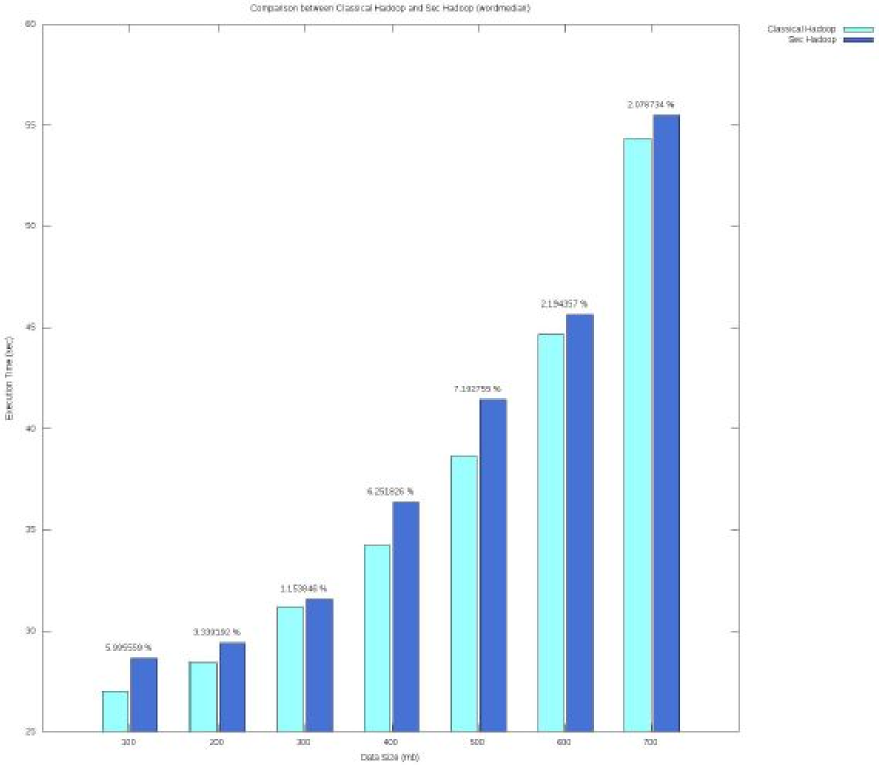
Average performance overhead : 2.62%

Performance Comparison for Grep Job



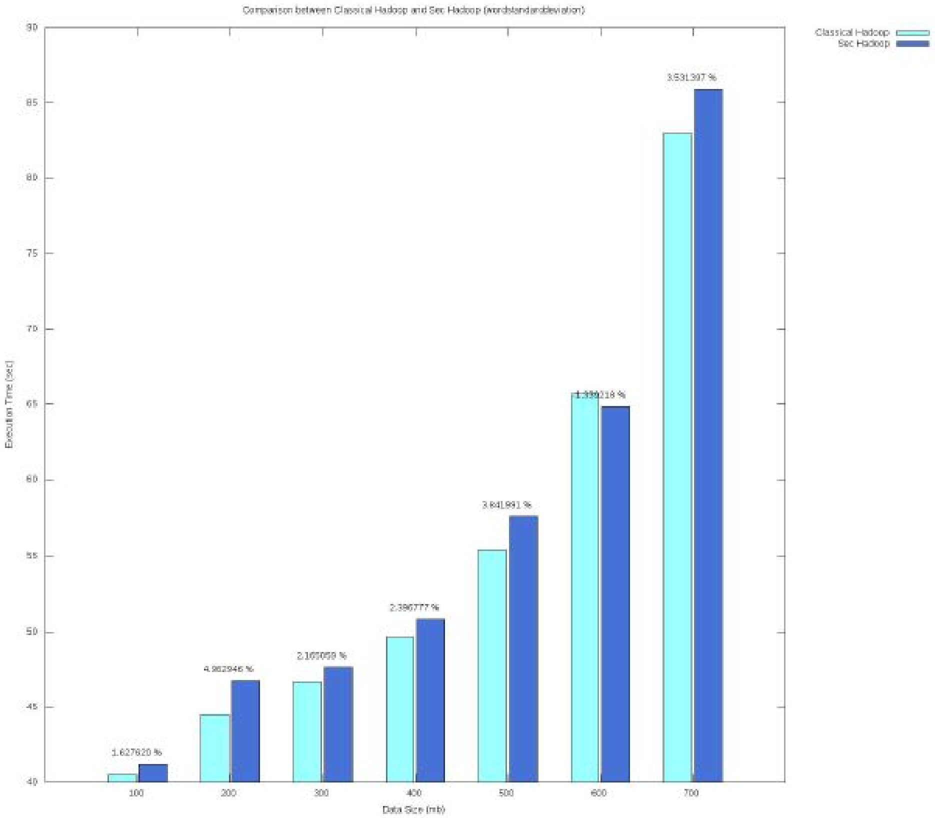
Average performance overhead : 1.55%

Performance Comparison for WordMedian Job



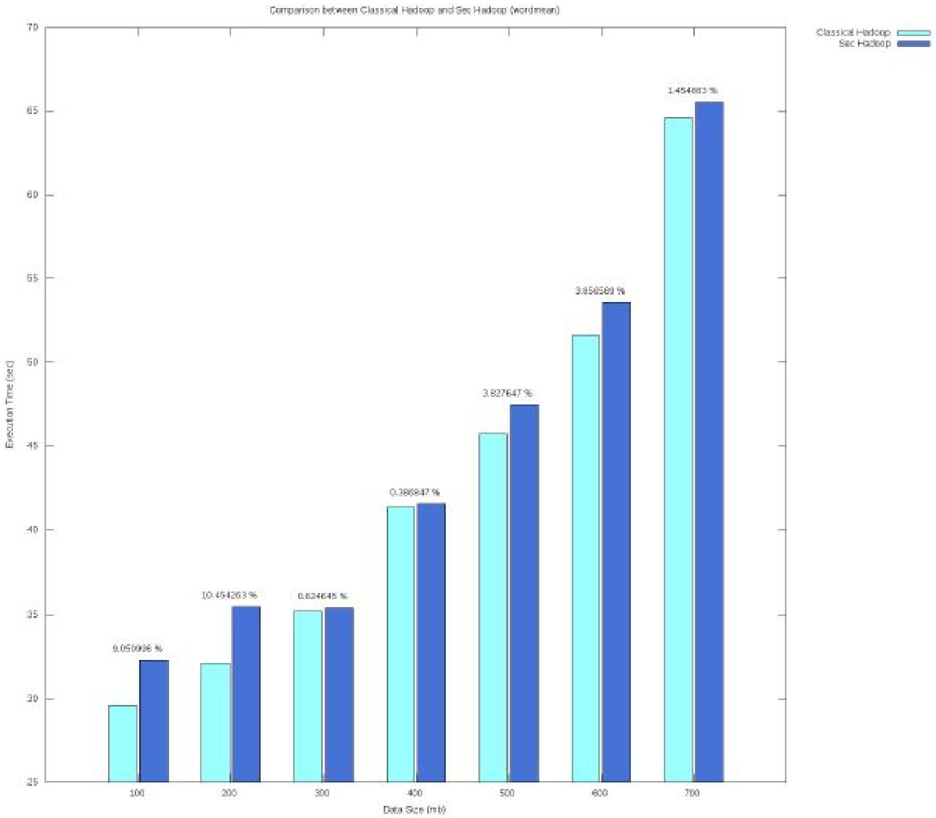
Average performance overhead : 4.02%

Performance Comparison for Word Standard deviation Job



Average performance overhead : 2.41%

Performance Comparison for WordMean Job



Average performance overhead : 4.23%

Differential Privacy

- No need to use Differential Privacy which depends on noise introduction (hence difficult for evolving data) and other issues of privacy violation

Ease of Use

- Labels of initial objects (data) need to be provided for specifying the security and privacy requirements
- Zero-changes to the programming model – jar files for map and reduce
- Zero-overhead in terms of system usage – job submission and configurations
- Negligible performance overhead

Summary

- Preserves Privacy end-to-end
- Applicable for merging databases (for desensitization)
- Very Little Overhead
- Avoids “noises” required in differential privacy and also applicable for dynamic data

Ongoing work for Medical Data Sharing

Medical wisdom: Realized through a large number of experiments by a multiple parties. In the creation of such datasets, two properties are vital:

1. Privacy: very important as the medical information of the patient needs to be kept private by the individual and can be used for the purpose treatment and possible to gather data (or warnings) for the community.
2. provenance. important for re-constructing intermediate results or new experiments from intermediate ones and ownerships (IPRs)
3. For time, we need to integrated Attribute access control as well.

Another Plus Point: Orange Book Standard

- Trusted Computer System Evaluation Criteria universally known as “the Orange Book”.
- B1 – Labeled Security Protection: the system must implement the Mandatory Access Control in which every subject and object of the system must maintain a security label, and every access to system resource (objects) by a subject must check for security labels and follow some defined rules.

Thank You