

**Fourth Bertinoro PhD. Summer School on
Security of Wireless Networks
July 6th –July 10th 2009, Forli-Cesna, Italy**

Professor Radha Poovendran

EE Department, University of Washington, Seattle, WA

&

Professor Dawn Song

EECS Department, University of California, Berkeley, CA

Summer School Objectives

- **Exposure to current research topics that are cross-cutting wireless networking and security**
- **Provide multi-faceted view from cryptography, networking and network-security**
- **Cover one or two topics in depth that form the theme of the workshop**
- **Encourage research activities and collaborations based on the workshop**

Professor Radha Poovendran

- **Networking Framework that forms the basis of the lectures**
- **Monday-Control Channel Jamming with Node Capture (with and without back channels; with and without prior known bounds on the # of nodes to be exposed; includes collusion/insider attacks)**
- **Tuesday-Modeling and mitigating jamming(in general throughput reduction attacks) on wireless networks- a network flow and convex optimization framework**
- **Wednesday— I will not lecture on Wednesday**
- **Thursday—Understanding source anonymity in sensor networks (give impossibility result first and then proceed with practical approaches); RFID search.**
- **Friday—Network vulnerability metrics for the first part; networking coding result; and information theoretic notion of keying; key establishment based on channel reciprocity. (topics here will be chosen based on time availability)**

Professor Dawn Song

- **Applied Cryptography for Privacy in Wireless Applications**
 - Searches over Encrypted Data; Private stream search (M)
 - Computation over Encrypted Data (Tu)
- **Defending against Malicious Code in Mobile Computing**
 - Techniques and Tools for in-depth Malware Analysis (W & Th)

Summer School Lecture Schedules

Time	Monday	Tuesday	Wednesday	Thursday	Friday
9:30-11:00	DS	DS	DS	DS	RP
11:30-1:00	RP	RP	GS	RP	RP
15:00-16:00	DS				

Background Assessment

- **Which year are you in?**
- **Have you taken undergrad & grad classes in**
 - Security?
 - Cryptography?
 - Program analysis?
 - Networking?
 - Statistics?
- **Have you done research in security?**

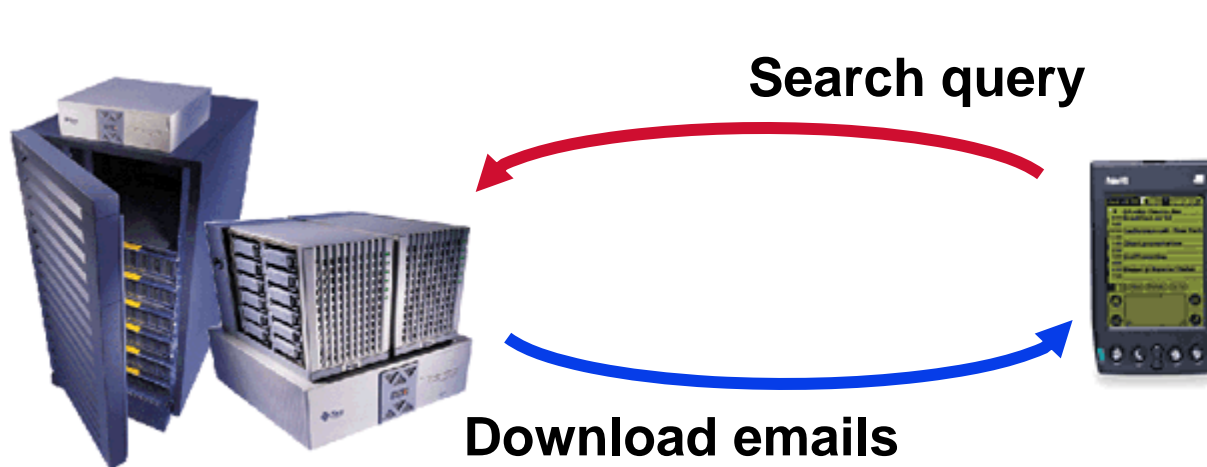
Part I: Applied Cryptography for Privacy in Wireless Applications

Overview

- **Privacy is important in information age**
- **Many mobile devices are thin**
 - How to have servers help mobile devices and preserve users' privacy at the same time?
 - How to enable private applications in community of mobile devices?
- **Example techniques & applications**
 - Searching on encrypted data
 - » Keyword search (equality test)
 - » Predicate encryption & multi-dimensional range query
 - Private stream search
 - » Techniques
 - » Application in analysis-resilient malware
 - Computation over encrypted data
 - » Private set operations
 - » Fully homomorphic encryption

Motivation

- **Why searches on encrypted data?**
 - Searching on encrypted e-mails on mail servers
 - Searching on encrypted files on file servers
 - Searching on encrypted databases
- **Why is this hard?**
 - Perform computations on encrypted data is often hard
 - Usual tradeoffs: security and functionality



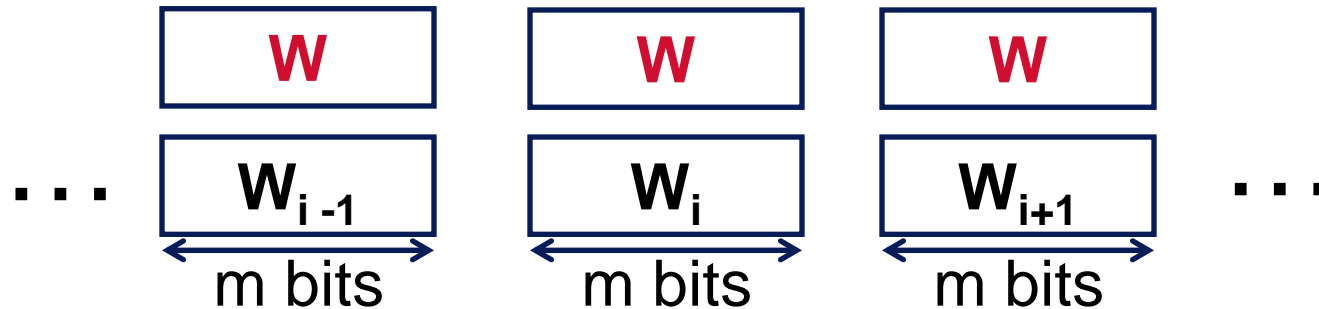
Outline

- **Searching on encrypted data**
 - **Keyword search (equality test) [SongWagnerPerrig]**
 - Multi-dimensional range query
- **Private stream search**
 - Techniques
 - Application in analysis-resilient malware
- **Computation over encrypted data**
 - Private set operations
 - Fully homomorphic encryption

Sequential Scan and Straw Man Example

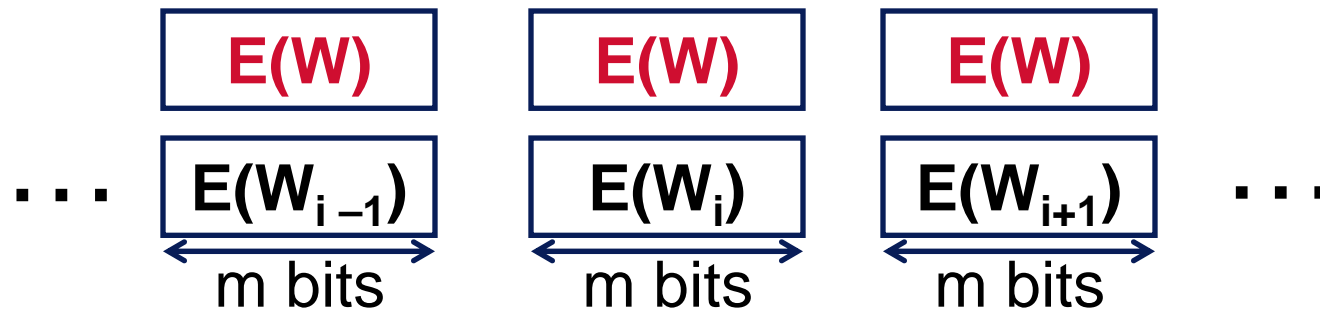
- Search by sequential scan:

Search for W



- Naïve approach:

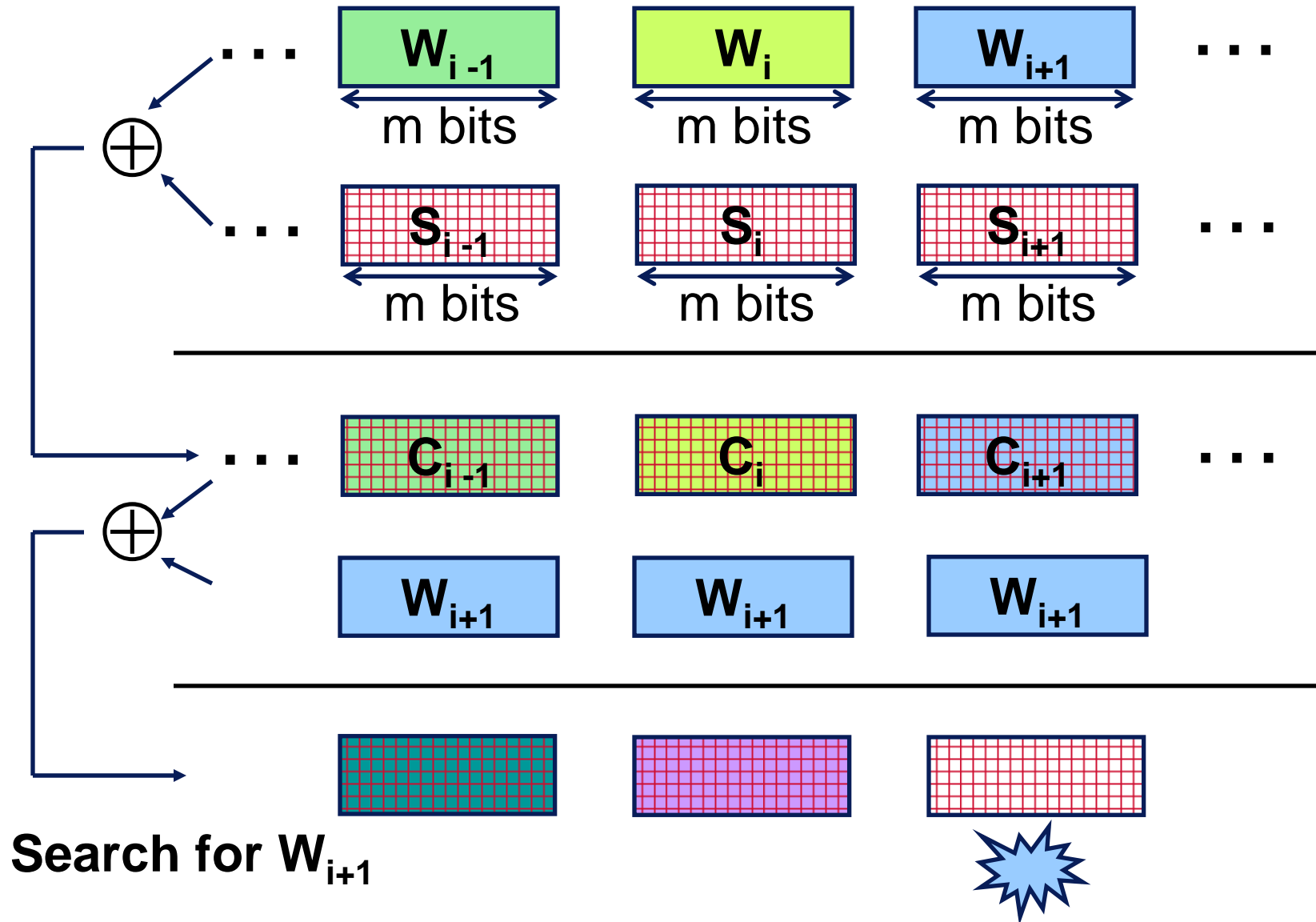
Search for W



Desired Properties

- **Provable security**
 - **Provable secrecy:**
encryption scheme is provable secure
 - **Controlled search:**
server cannot search for arbitrary word
 - **Query isolation:**
search for one word does not leak information about other different words
 - **Hidden queries:**
does not reveal the search words
- **Efficiency**
 - **Low computation overhead**
 - **Low space and communication overhead**
 - **Low management overhead**

The Key Idea



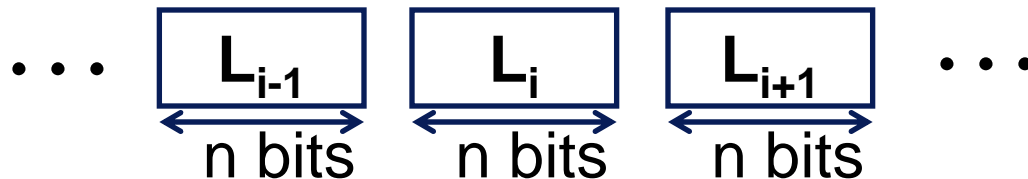
Setup and Notations

- **Document:** sequence of fixed length words



- **Pseudorandom Generator G and seed:**

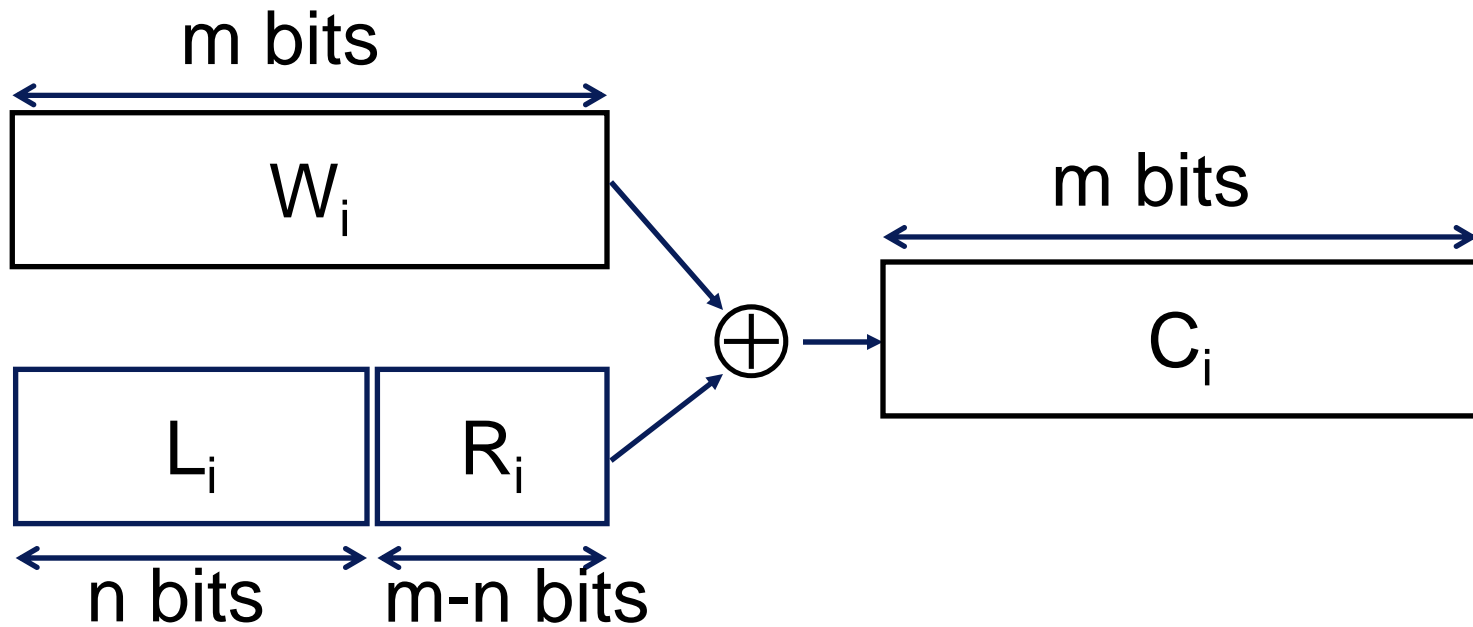
$$L \leftarrow G(\text{seed}), \quad L_i \leftarrow G_i(\text{seed})$$



- **Pseudorandom Function F and K :**

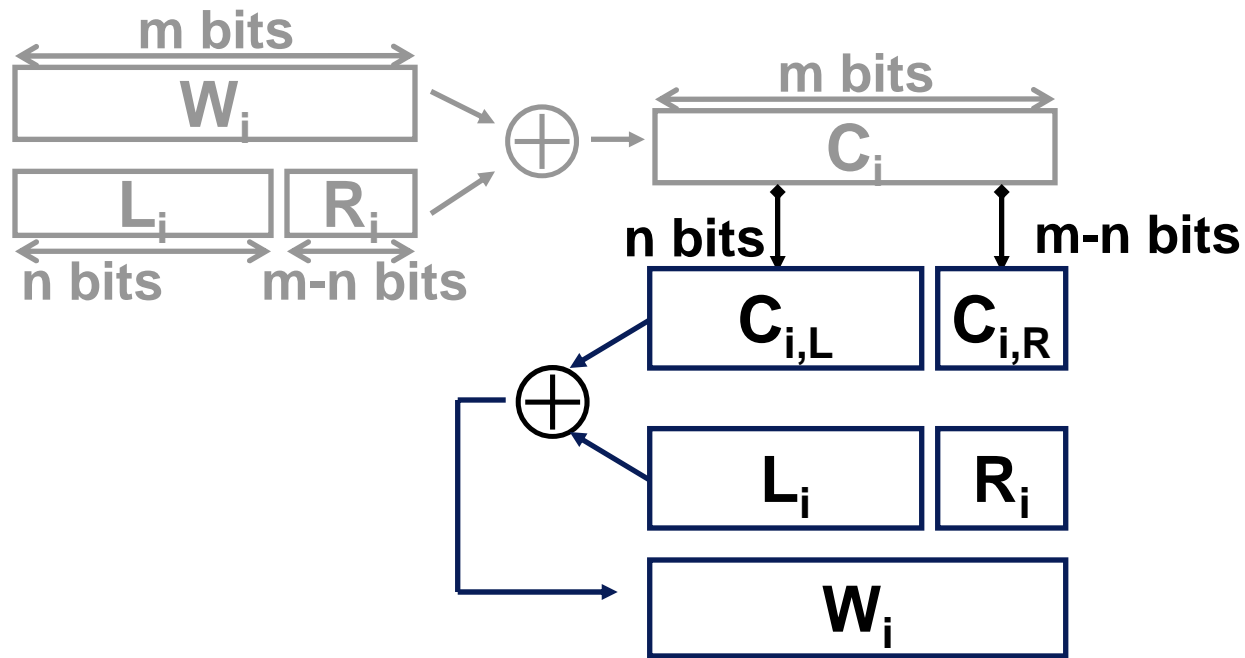
F_K maps n bits to $m-n$ bits

Basic Scheme (Encryption)



$$L_i \leftarrow G_i(\text{seed}), \quad R_i \leftarrow F_K(L_i)$$

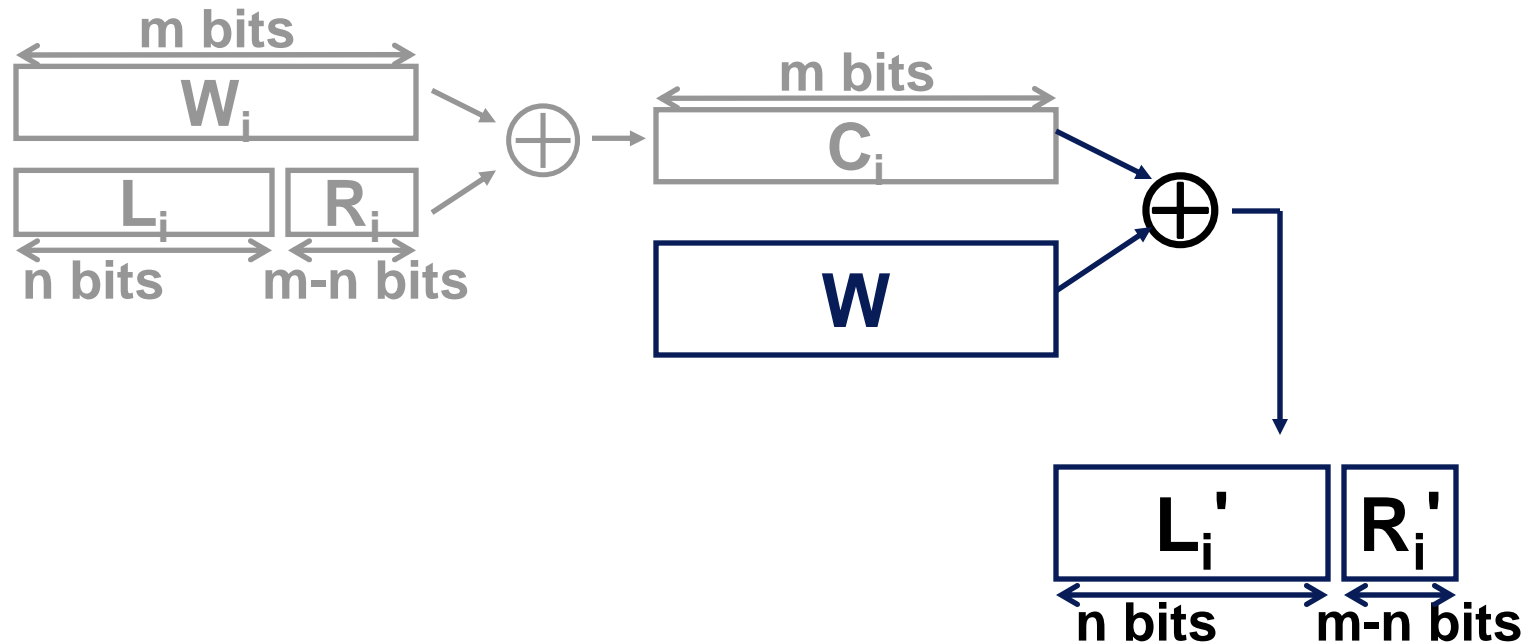
Basic Scheme (Decryption)



$$L_i \leftarrow G_i(\text{seed}), \quad R_i \leftarrow F_K(L_i)$$

Basic Scheme (Searches)

Search for word W , give server W and K



Check: $R_i' = F_K(L_i')$?

Yes \Rightarrow match,

(false positive rate = $1 / 2^{m-n}$)

Controlled Searches and Query Isolation

- **Controlled searches on words**

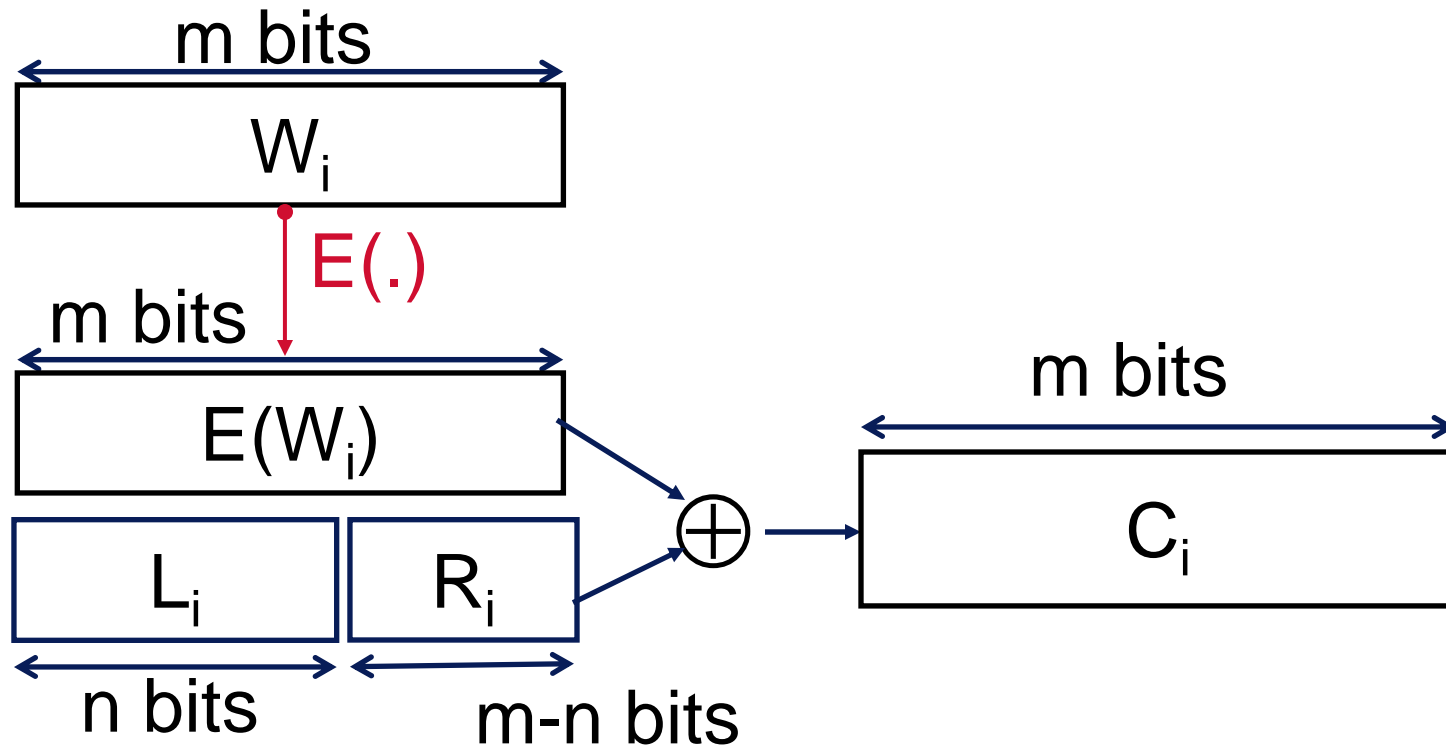
Instead of $R_i \leftarrow F_K (L_i)$,

$R_i \leftarrow F_{K_i} (L_i)$,

where $K_i = F'_K (W_i)$

- **Enhancements (in paper) :**
 - Check for a word in a single chapter/section only
 - Check only for “word occurs at least once” in document
 - Check only for “word occurs at least N times” in document

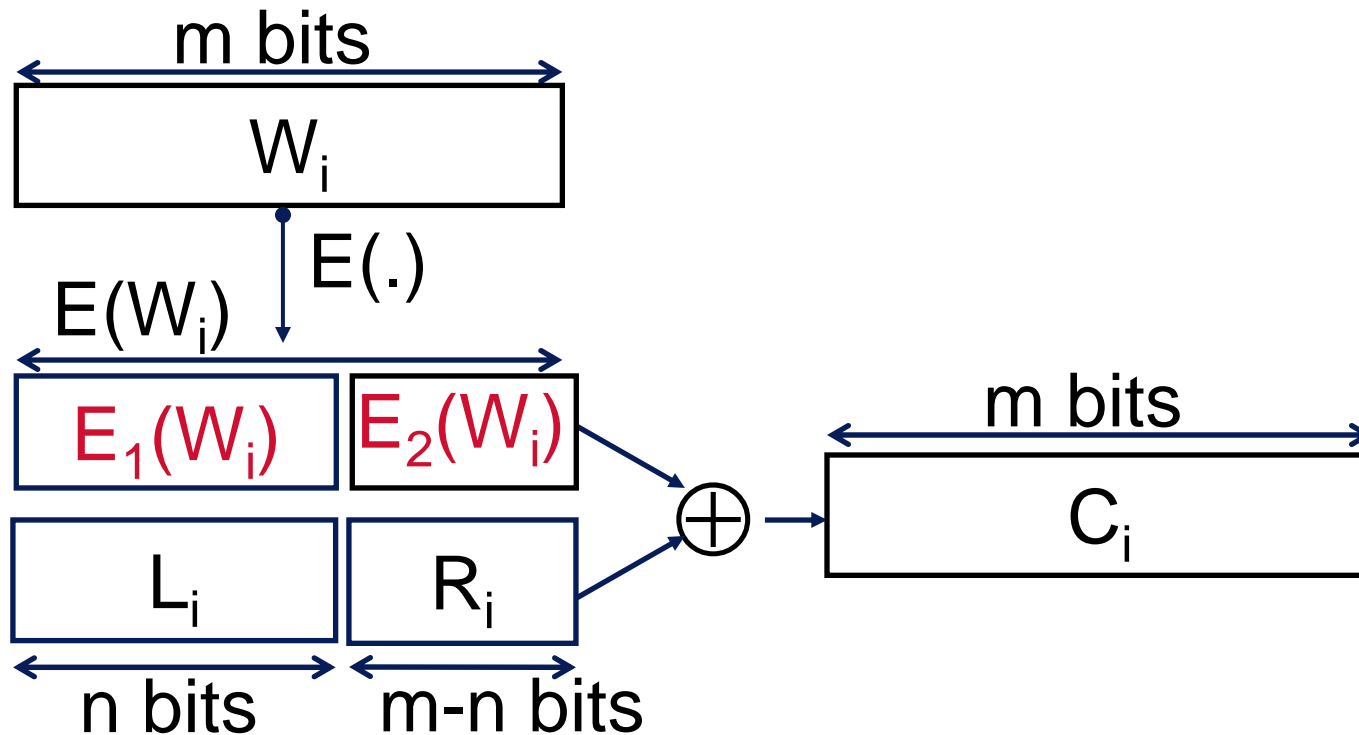
Hidden Queries



$$L_i \leftarrow G_i(\text{seed}), \quad R_i \leftarrow F_{K_i}(L_i)$$

$$\text{where } K_i = F'_K(E(W_i))$$

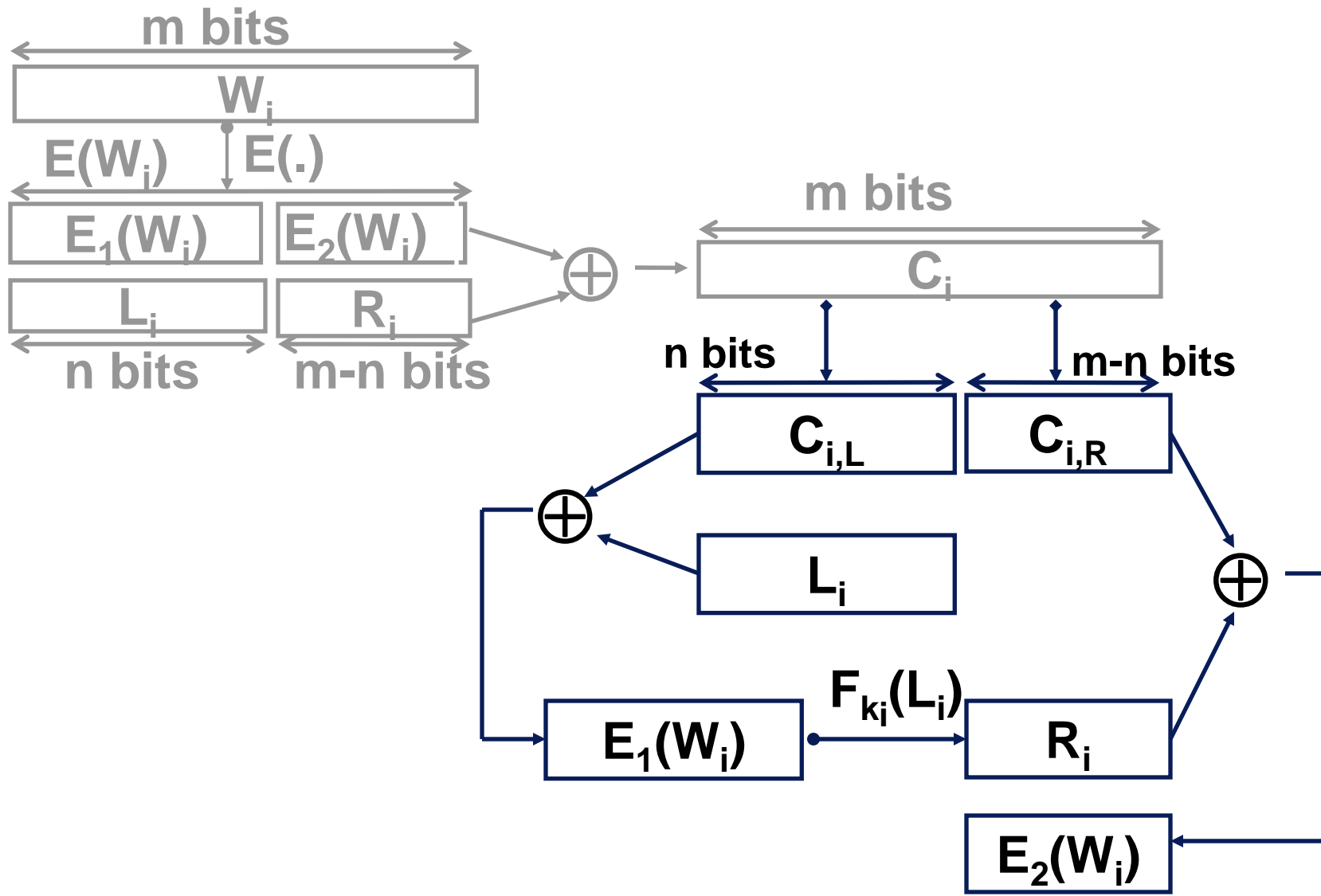
Final Scheme (Encryption)



$$L_i \leftarrow G_i(\text{seed}), \quad R_i \leftarrow F_{K_i}(L_i)$$

$$\text{where } K_i = F'_K(E_1(W_i))$$

Final Scheme (Decryption)



Advanced Search Queries

- **Building blocks for advanced search queries:**
 - W_1 and W_2 ,
 - W_1 near W_2 ,
 - W_1 immediately precedes W_2
- **Supports variable length words:**
 - Same provable security
 - Similar efficiency

Summary

- **Provable security**
 - Provable secrecy
 - Controlled search
 - Query isolation
 - Hidden queries
- **Simple and efficient**
 - $O(n)$ stream cipher and block cipher operations per search
 - Almost no space and communication overhead
 - Easy to add documents
 - Convenient key management :
user needs only one master key
- **Embedding information in pseudorandom bit streams**

Student Forum

- **We want to hear about your research too 😊**
- **Voluntary (but encouraged 😊)**
- **Thu morning**
- **10 min each**
 - 8 min presentation
 - 2 min Q&A and feedback
- **Structure**
 - What is the problem?
 - Why is it important (motivation)?
 - What is the approach (overview)?
 - Comparison to related work

Public-key based Search on Encrypted Data

- **Based on pairings and identity-based encryption**
 - Boneh, Crescenzo, Ostrovsky, Persiano, [Eurocrypt 2004]

Outline

- **Searching on encrypted data**
 - Keyword search (equality test) [SWP]
 - **Multi-dimensional range query and predicate encryption**
- **Private stream search**
 - Techniques
 - Application in analysis-resilient malware
- **Computation over encrypted data**
 - Private set operations
 - Fully homomorphic encryption

Motivating example

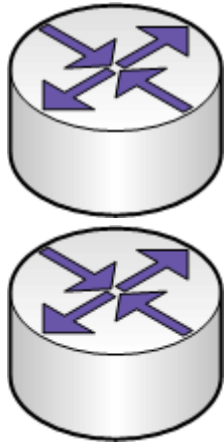
- **Network worms**
 - Malicious program
 - Worm characteristic, e.g.,
port = 1434 for SQL slammer
- **Collecting network audit logs**
 - Study origin, dynamics of worms
- **Privacy concerns**

Typical network audit log

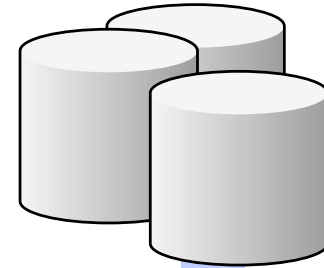
Src IP	Dest IP	Time	Src Port	Payload
1.1.1.1	1.1.1.2	Jan 1, 3:22	80	xYdcaYi
2.2.2.1	2.2.2.2	Jan 2, 4:22	90	czUEhc
3.3.3.3	3.3.3.2	Jan 3, 5:22	100	caeYD
4.4.4.1	4.4.4.2	Jan 4, 6:18	3389	caefU
...

Network Audit Logs

ISPs



Research center



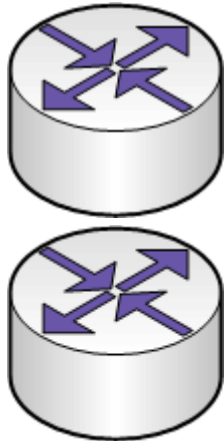
$(\text{port} = 1434) \wedge (\text{ip} \in 128.1.*.*)$



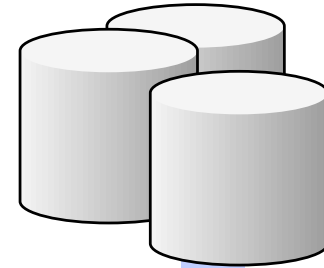
Auditor

Network Audit Logs

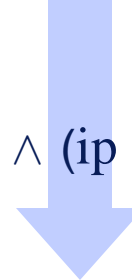
ISPs



Research center



$(\text{port} = 1434) \wedge (\text{ip} \in 128.1.*.*)$

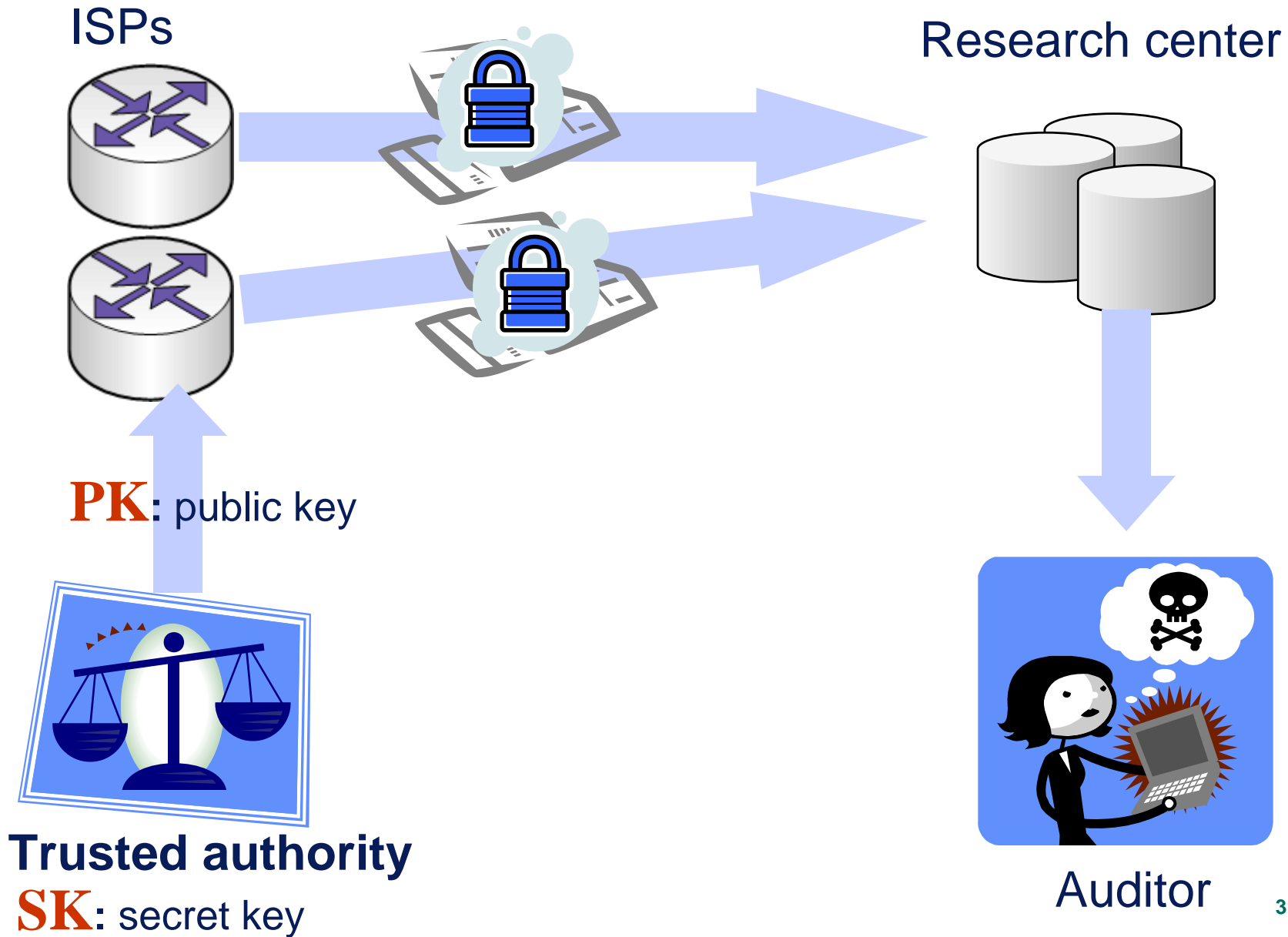


ISPs care about privacy

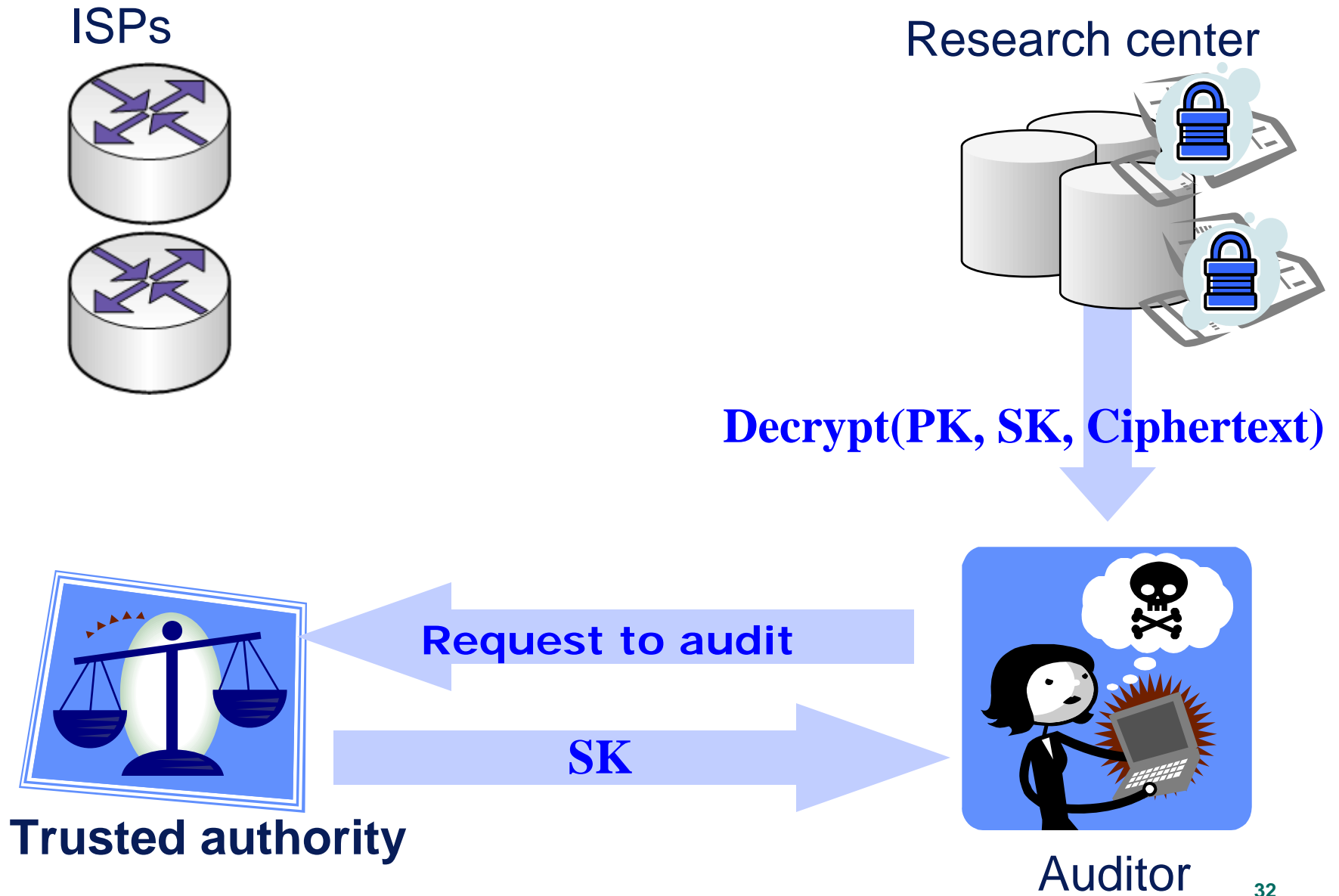


Auditor

A naïve solution



A naïve solution



The privacy perspective

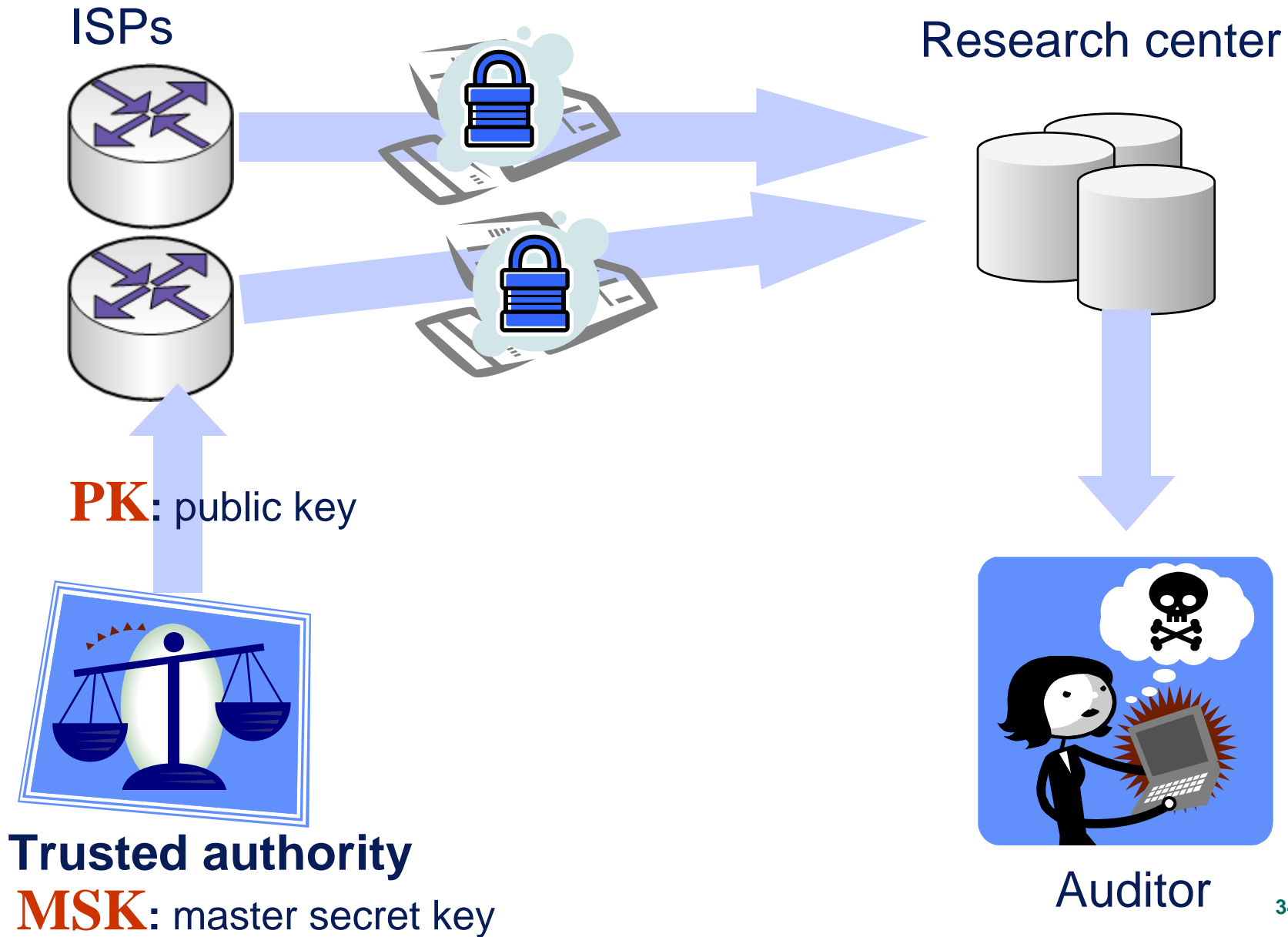
Naive solution:

- Auditor is able to decrypt everything

Ideal solution:

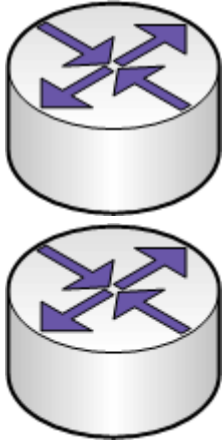
- Auditor should be able to decrypt only suspicious flows
- Benign users' flows still remain secret

Predicate Encryption

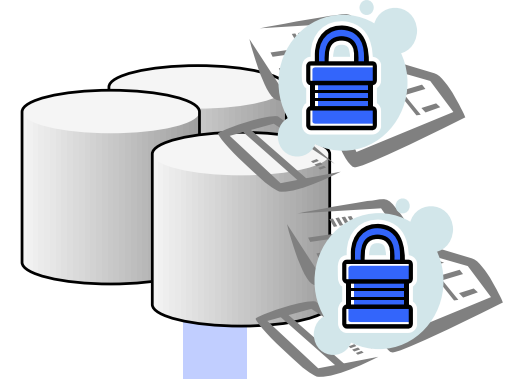


Predicate Encryption

ISPs



Research center



Decrypt(PK, TK, Ciphertext)



Trusted authority

$(\text{port} = 1434) \wedge$
 $(\text{ip} \in 128.2.*.*)$

token TK

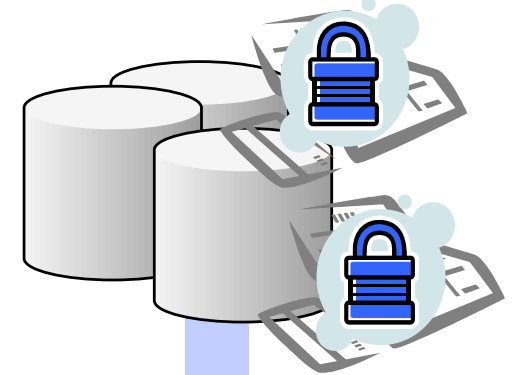


Auditor

Predicate Encryption

- **TK** is a partial decryption key
- Allows auditor to decrypt entries satisfying attack characteristic
- All other entries remain secret

Research center



Decrypt(PK, TK, Ciphertext)



Trusted authority

$(\text{port} = 1434) \wedge$
 $(\text{ip} \in 128.2.*.*)$

token TK



Auditor

Recap: Predicate Encryption

- **Traditional Encryption**
 - all-or-nothing decryption
- **Predicate Encryption**
 - A token allows one to learn partial information
 - Controlled release of information

Predicate encryption: Definition

$X = (\text{IP}, \text{port}, \text{pkt_len})$

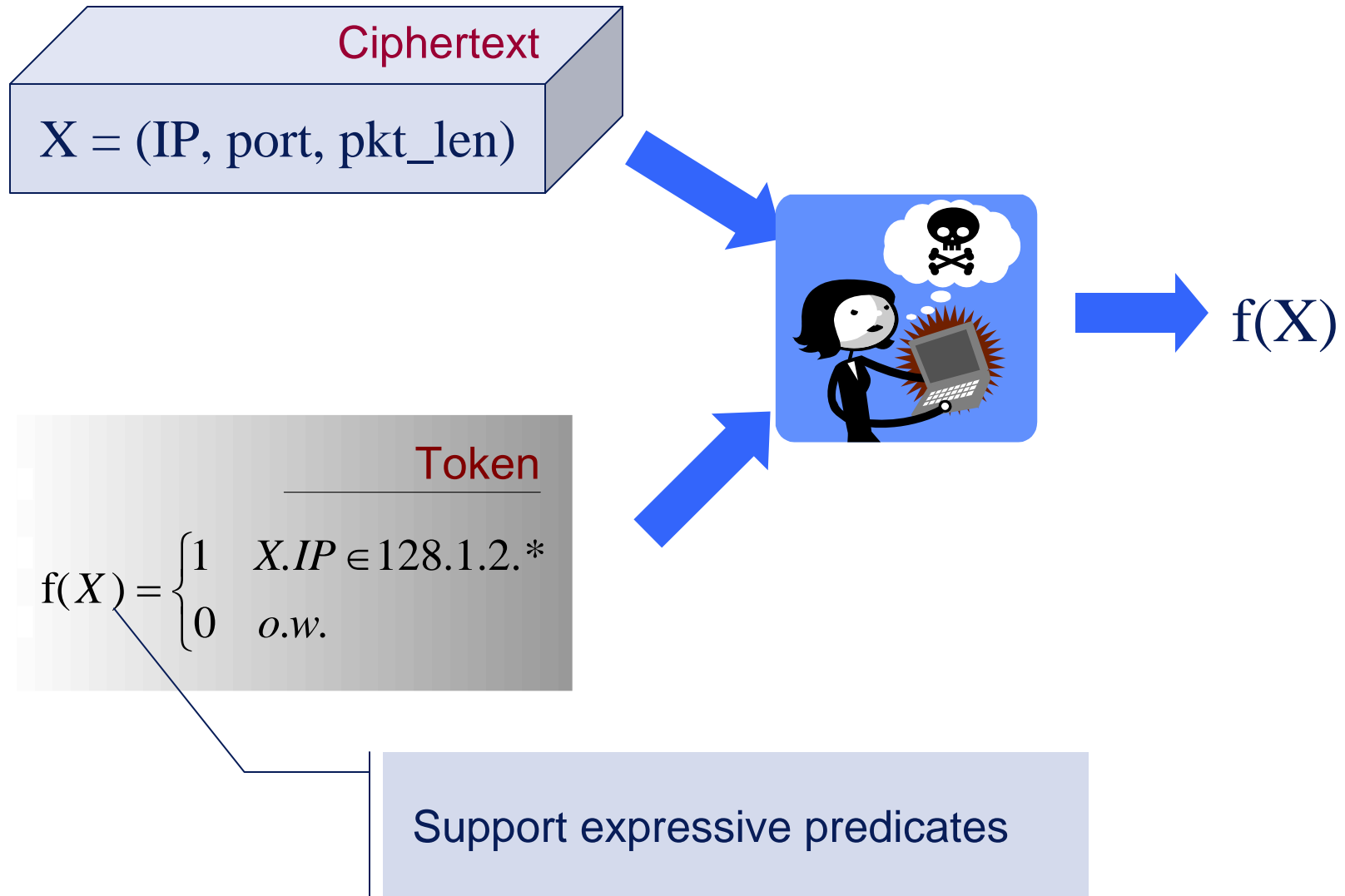
Ciphertext
 $X_1 = (1.2.3.4, 56, 78)$

Ciphertext
 $X_2 = (3.2.3.5, 91, 78)$

...

Ciphertext
 $X_m = (6.7.8.8, 11, 23)$

Predicate encryption: Definition



Predicate encryption: Prior Work

- **Equality test:**

- Goldreich, Ostrovsky, [JACM 1996]
- Song, Wagner, Perrig, [S&P 2000]
- Boneh, Crescenzo, Ostrovsky, Persiano, [Eurocrypt 2004]

$$f_a(X) = \begin{cases} 1 & X = a \\ 0 & o.w. \end{cases}$$

Multi-dimensional Range Query

- **Multi-dimensional range queries:** $X = (x_1, x_2, \dots, x_n)$

$$f_{a_1, a_2, b_1, b_2}(X) = \begin{cases} 1 & (x_1 \in [a_1, a_2]) \wedge (x_3 \in [b_1, b_2]) \\ 0 & o.w. \end{cases}$$

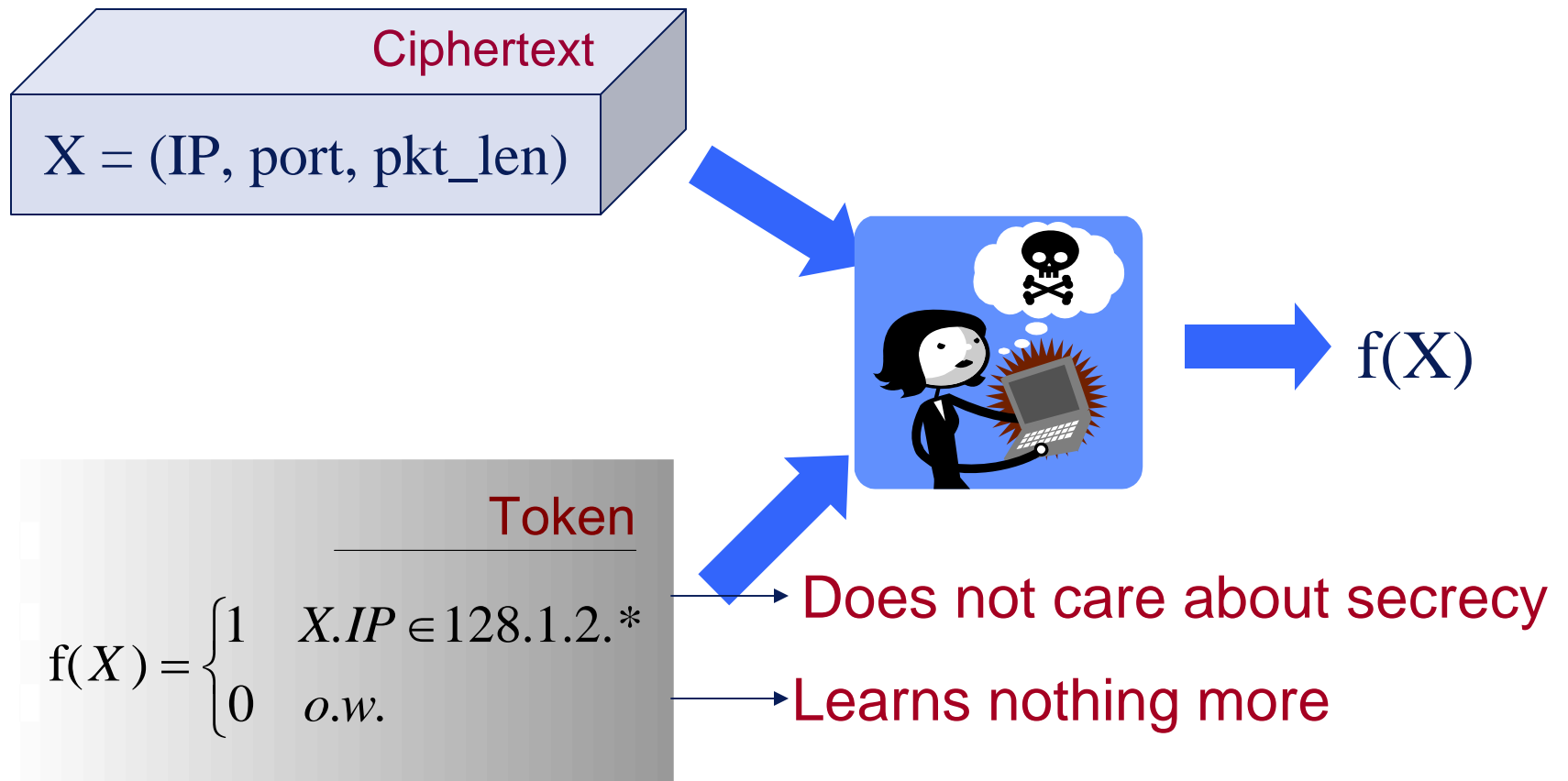
- **Core technique: conjunctive queries**

$(IP \in 128.2.*.*) \wedge (\text{port} \in [1000, 2000])$

$(IP \in 128.2.*.*) \wedge (\text{port} = 1434)$

$$f_{a,b}(X) = \begin{cases} 1 & (x_1 = a) \wedge (x_3 = b) \\ 0 & o.w. \end{cases}$$

Match-revealing security



a.k.a. one-sided security

Multi-dimensional Range Query

- **Plaintext:** $\mathbf{X} = (\text{IP}, \text{port}, \text{pkt_len})$

- **Queries:**

$(\text{IP} \in 128.2.*.*) \wedge (\text{port} \in [1000, 2000])$

$(\text{IP} \in 128.2.*.*) \wedge (\text{port} = 1434)$

- **Consider match-revealing security**
 - If \mathbf{X} satisfies predicate, then auditor actually would like to decrypt entire entry
 - Otherwise, preserve secrecy of encrypted point \mathbf{X}

Multi-dimensional range query

Scheme	PK. size	Enc. Time per entry	Ciphertext Size per entry	TK. Size	Dec. Time per entry
AIBE 05	$O(1)$	$O(1)$	$O(1)$	$O(T^D)$	$O(T^D)$
[BW06]	$O(D \cdot T)$	$O(D \cdot T)$	$O(D \cdot T)$	$O(D)$	$O(D)$
Our Scheme	$O(D \cdot \log T)$	$O(D \cdot \log T)$	$O(D \cdot \log T)$	$O(D \cdot \log T)$	$O((\log T)^D)$

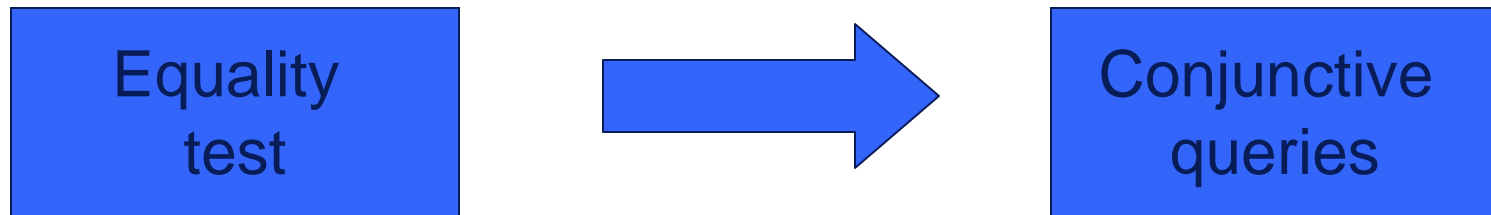
[BW06]: Boneh and Waters, TCC 2007: “Conjunctive, Subset, and Range Queries on Encrypted Data”, match concealing

Our scheme: S&P 2007

T: # different values for each field

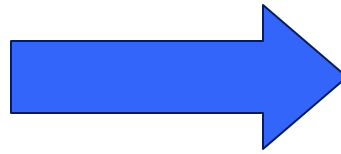
D: # fields

Scheme for Conjunctive Equality Test

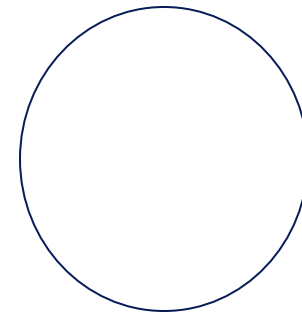


Naïve solution

Equality
test



Conjunctive
queries



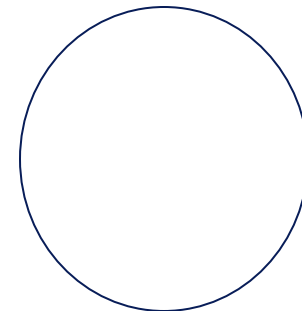
$(IP = 1.2.3.4) \wedge (port = 1434)$

Naïve solution



 (IP = 1.2.3.4)

 (port = 1434)



$(\text{IP} = 1.2.3.4) \wedge (\text{port} = 1434)$

Security requirement

- **Given a token for**

(IP = 1.2.3.4) \wedge (port = 1434)

- **One should not be able to learn individual clauses:**

(IP = 1.2.3.4)

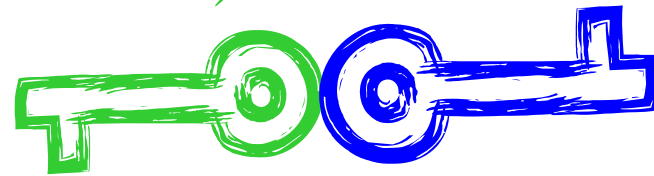
(port = 1434)

Idea for a fix

- Go to store, buy some industrial glue:



(IP = 1.2.3.4)



(port = 1434)

Our construction [SBCSP]

- **D**: number of fields in an entry
- 5 relevant performance measures: all $O(D)$
 - Public key size
 - Ciphertext size (per entry)
 - Encryption time (per entry)
 - Token size
 - Decryption time (per entry)
- **Security**: reduced to hard problems in certain mathematical groups (pairings)

Summary

- **Searching on encrypted data is an important primitive**
- **Techniques for keyword search (equality test)**
- **Generalization---predicate encryption**
- **Techniques for multi-dimensional range query**
- **Open problems**
 - **more efficient match-concealing multi-dimensional range query**
 - **Other predicate encryption classes**