

Patient Controlled Encryption: Ensuring Privacy of Electronic Medical Records

Melissa Chase (MSR)

Joint work with Josh Benaloh, Kristin Lauter,
and Eric Horvitz

Medical Records

- Traditionally, health providers kept paper files
 - Transferring data very cumbersome
 - Visiting a new doctor requires paperwork
 - Emergency care often cannot access record



Electronic Medical Records

Movement to:

- Digitize records
- Make accessible to network of providers



Patients' records will be accessible to any provider who treats them

Advantages

- Better care
- Reduce costs

President Obama: “all medical records computerized ... within 5 years”

ARR Act: \$19 billion

Privacy concerns

- Also dangerous



– Much easier to steal digital records



– Much easier to attack remotely accessible system



– Large system is very vulnerable to abuse

ARR Act: Specific objectives:

- Secure communications
- “Ensure appropriate authorization”
- Encryption

Privacy Concerns

- Why are we concerned about privacy?
 - Want patients to be honest
 - Discrimination
 - Insurance
 - Employment
 - Social stigma (friends/coworkers)
 - Medical Identity Theft

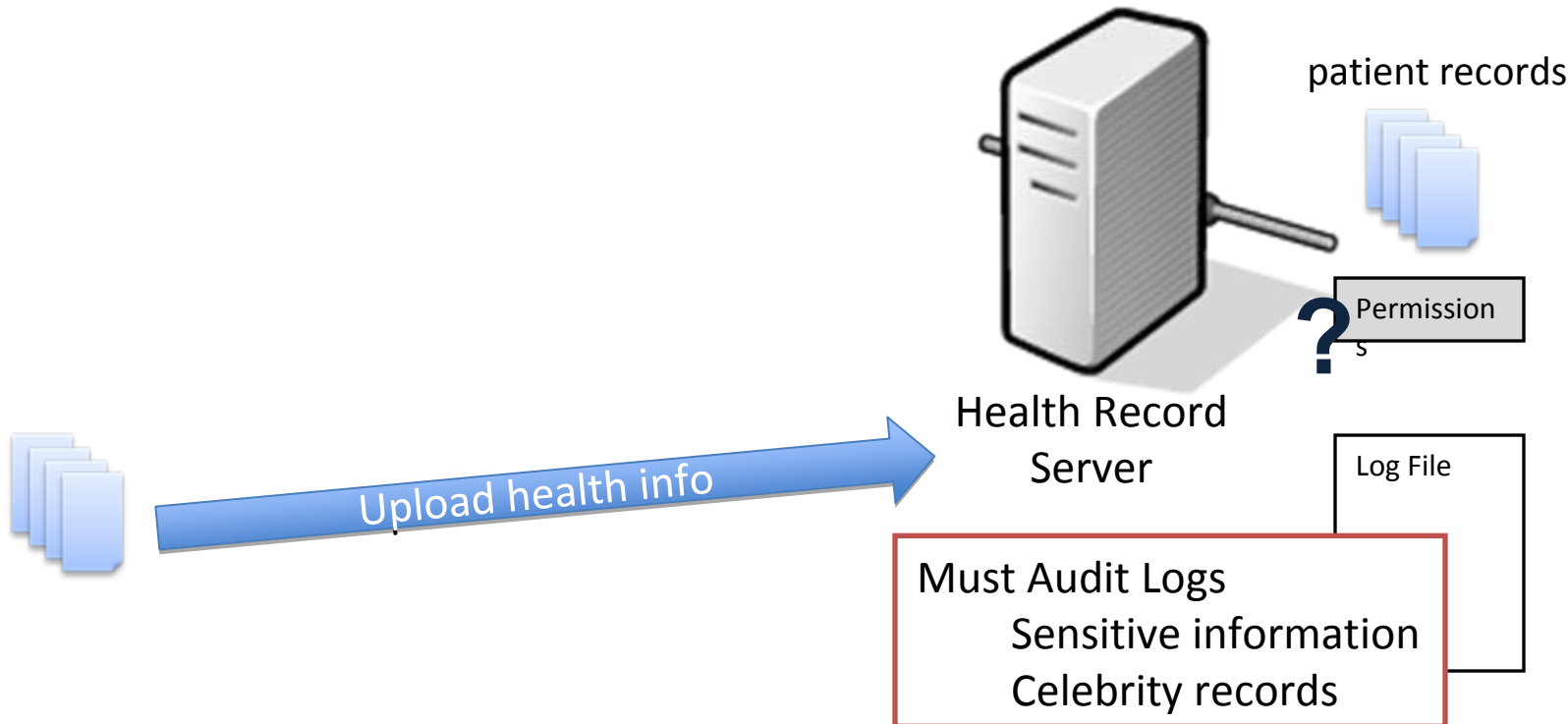
Standard Approach to Security: Provider Managed + Access Control



Patient Bob



Doctor Alice





Privacy Concerns

- Wide access
 - All or nothing permissions
 - Even more in large network scenario
 - Roughly **150 people** have access to a patient's record in a hospitalization



Patient Controlled Record

- grant access only to *appropriate part of record*
- allow patient to identify providers who treat him

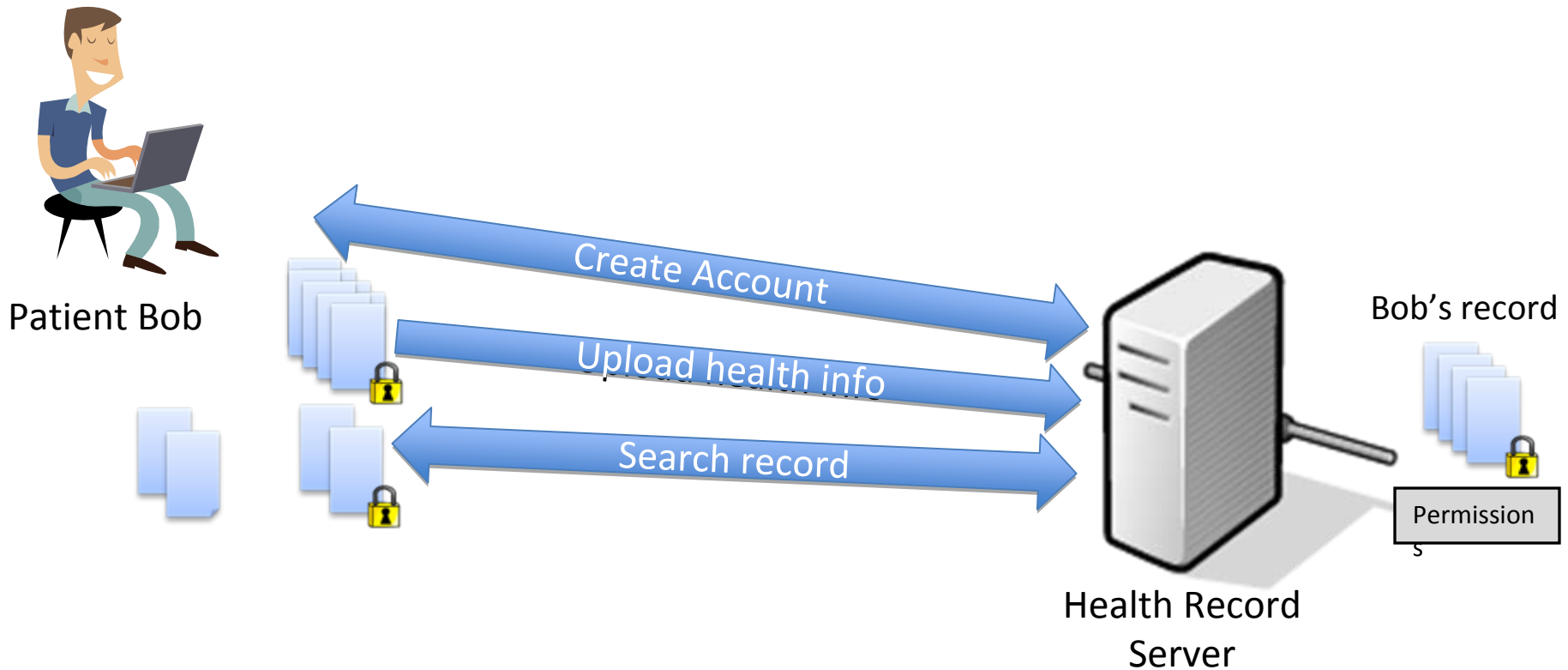


Privacy Concerns

- Wide access
 - All or nothing permissions
 - Even more in large network scenario
 - Roughly **150 people** have access to a patient's record in a hospitalization
- Access control
 - Theft
 - Attack
 - Patient must trust owner/administrator of data
 - for physical and electronic security
 - For privacy (Insider attacks)



New Approach: Encryption



Using Cryptography

- Who holds the key?

- Server 

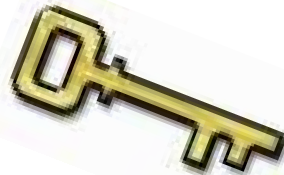
- Key can be stolen/compromised along with data

- Third party ?

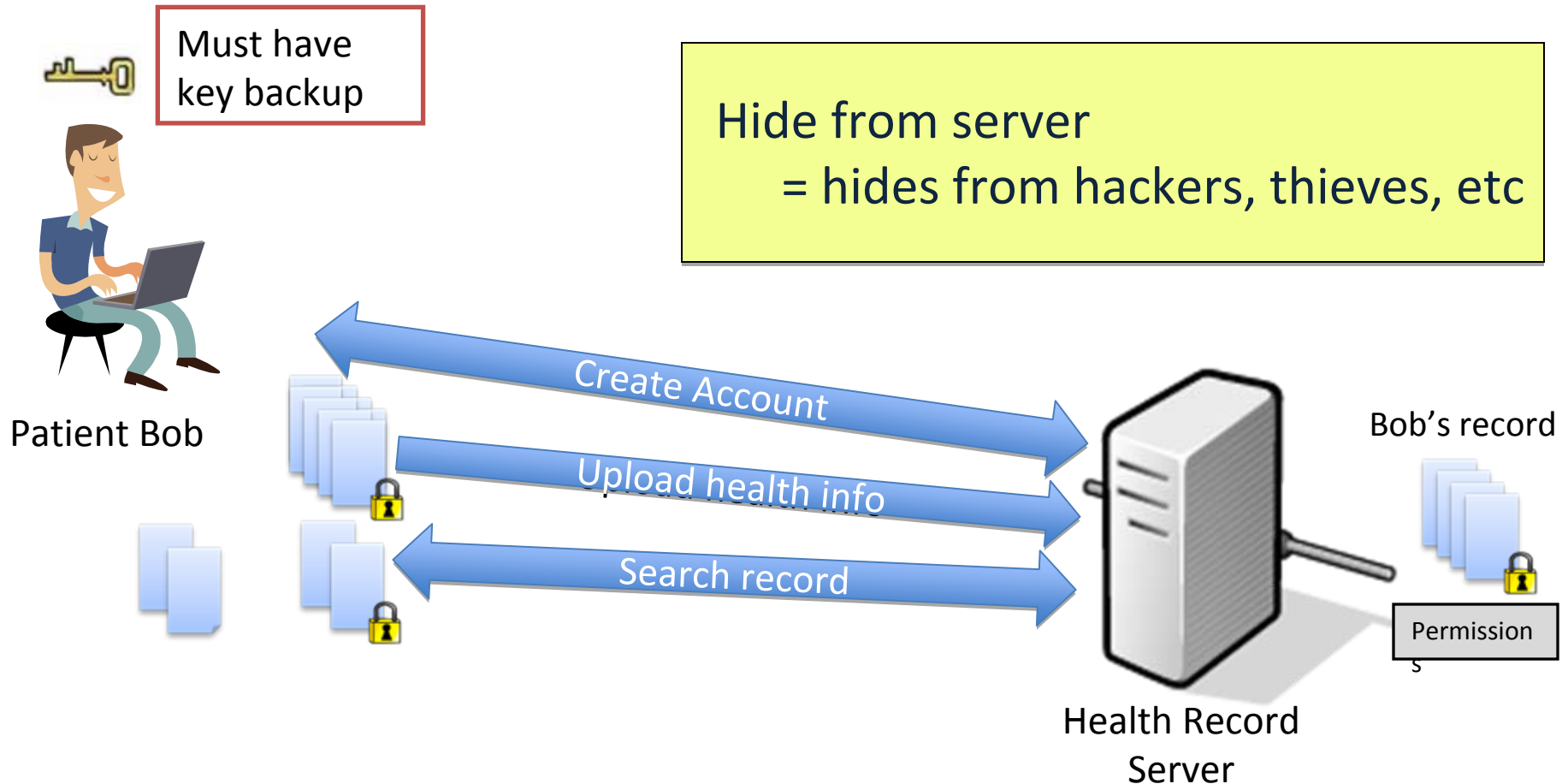
- Somewhat more secure
 - How to maintain functionality?

- Patient 

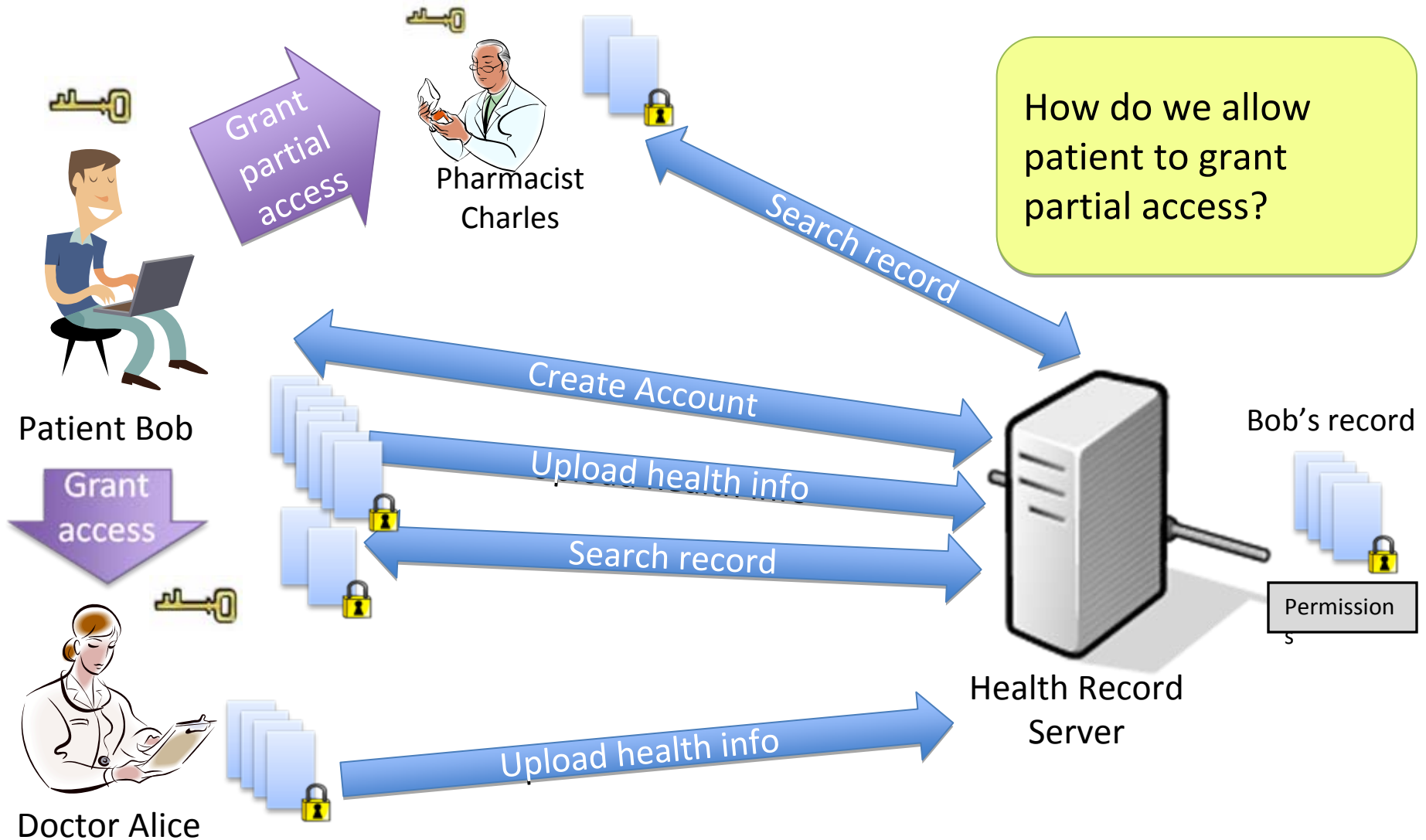
- What we will look at



Using Cryptography



Using Cryptography



How to grant partial access?

- Two approaches
 - Hierarchical record sharing:
 - Patient Controlled Encryption [BCHL09]
 - Assumes hierarchical health record
 - Based on standard, efficient primitives (hash functions, block ciphers)
 - Also consider how to incorporate searchability
 - Re-encryption based sharing:
 - Functional Re-Encryption [CCV12]
 - Easier to revoke users / add revocation date
 - Easier key management
 - More complex constructions
 - Based on bilinear pairing

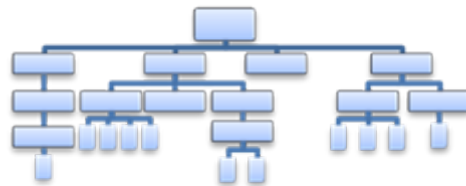
Hierarchical Record Sharing

Joint work with Eric Horvitz,
Kristin Lauter,
and Josh Benaloh

Hierarchical Record Sharing

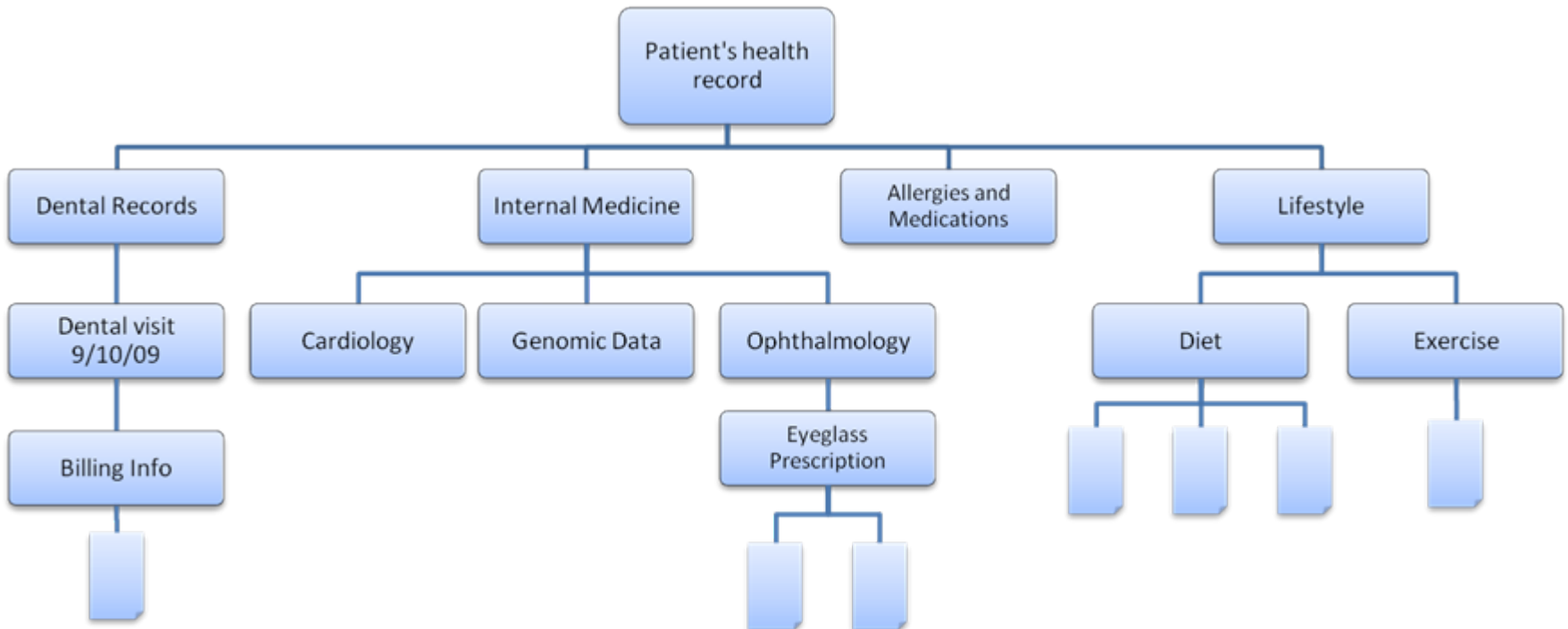


- Assumption on types of delegation
 - Arrange the record in a hierarchy
 - Allowable delegations: rights to a category



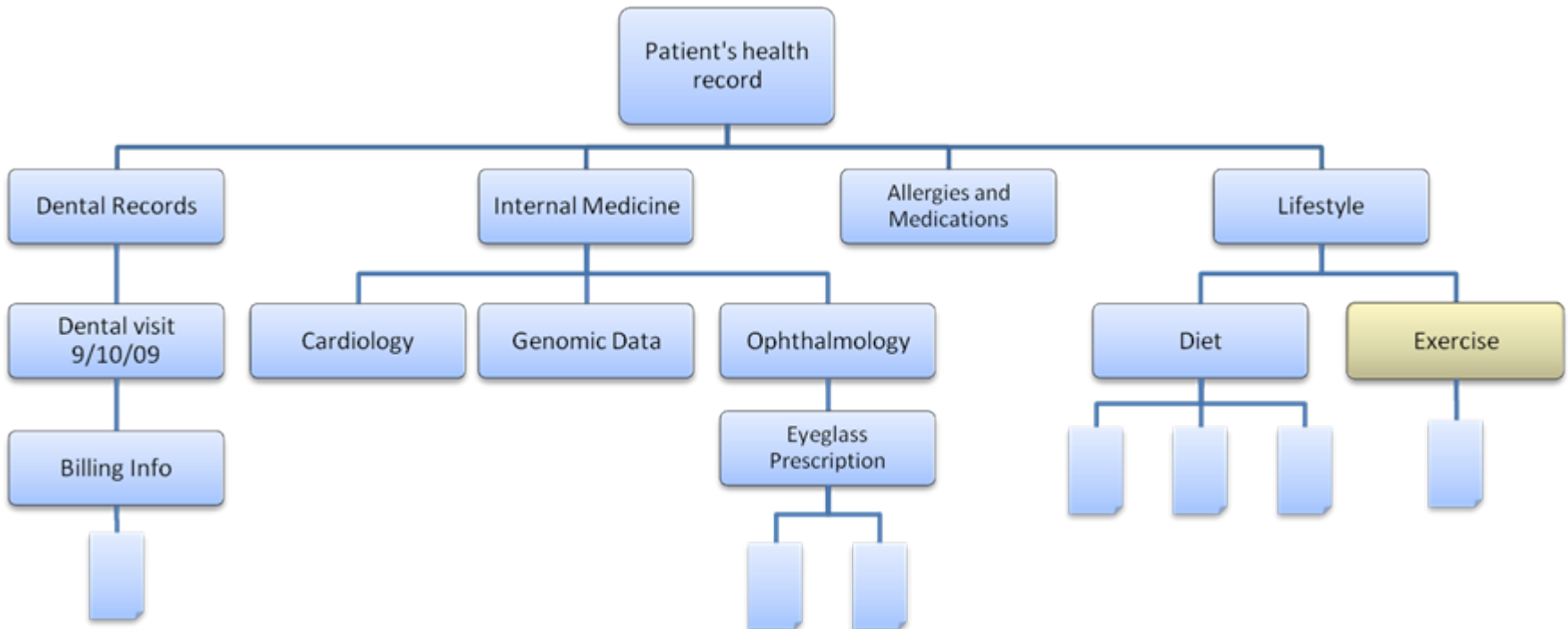
Hierarchical Health Records

- Ex:
 - Give Doctor access to entire record



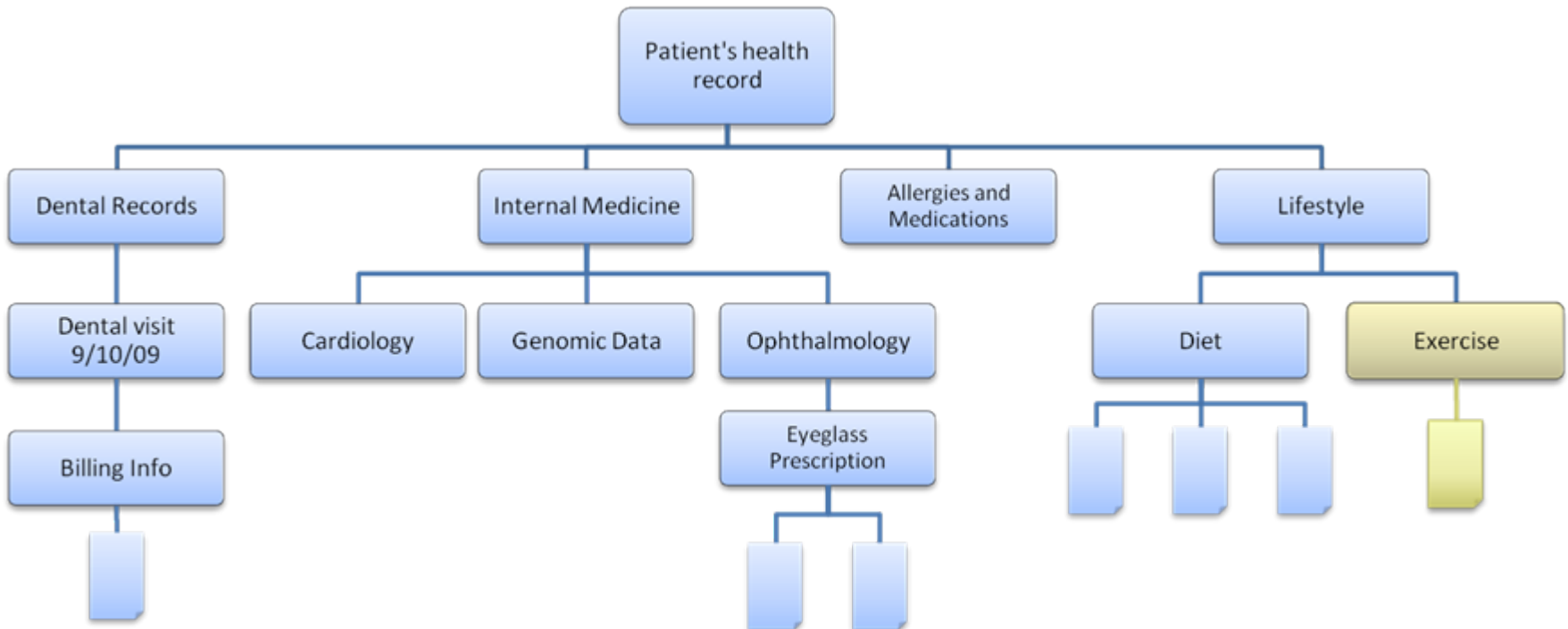
Hierarchical Health Records

- Ex:
 - Give Doctor access to entire record
 - Give Exercise info to cousin



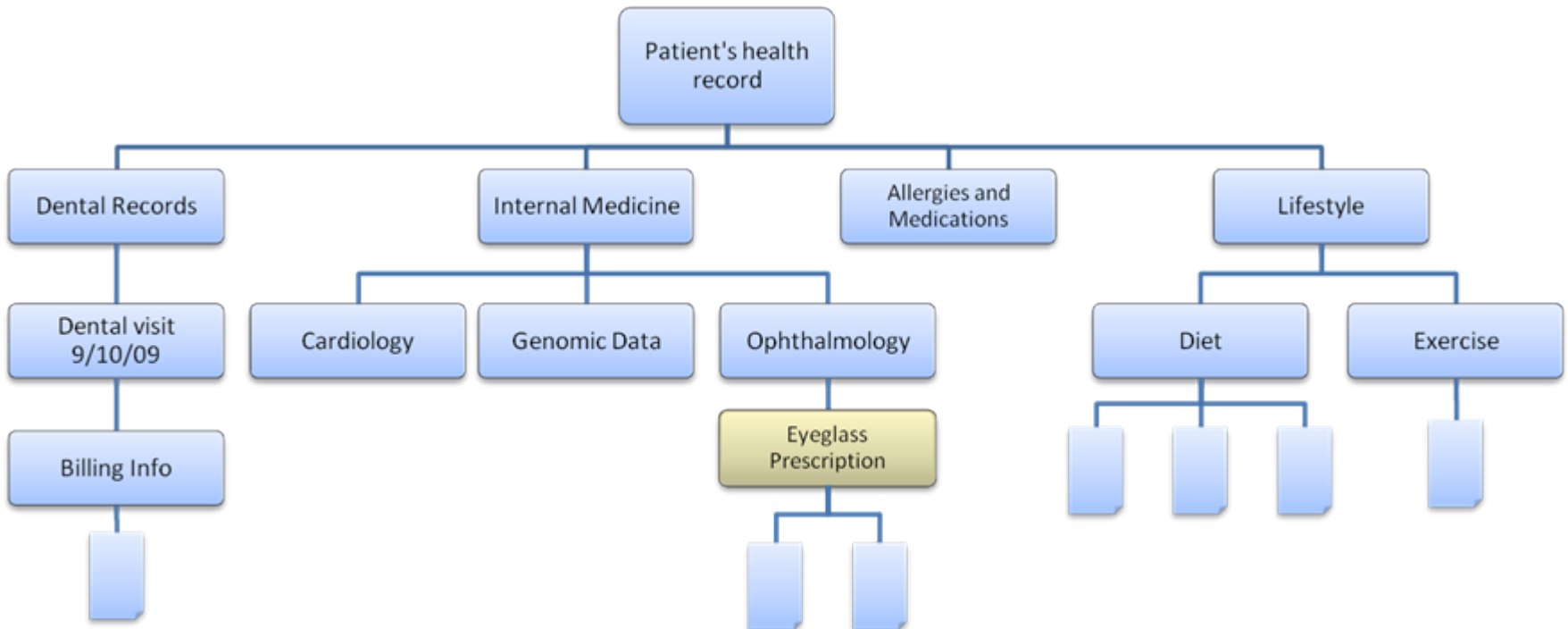
Hierarchical Health Records

- Ex:
 - Give Doctor access to entire record
 - Give Exercise info to cousin



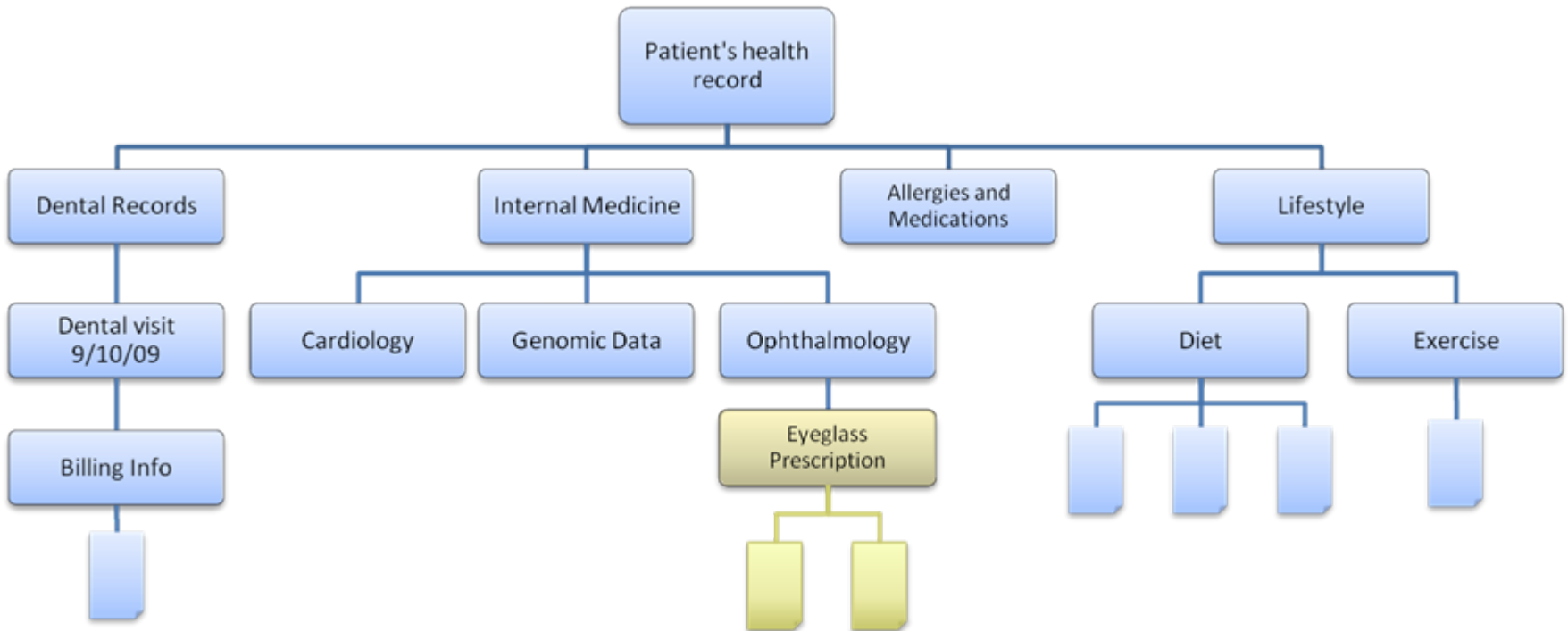
Hierarchical Health Records

- Ex:
 - Give Doctor access to entire record
 - Give Exercise info to cousin
 - Give Eyeglass Prescription to Retailer



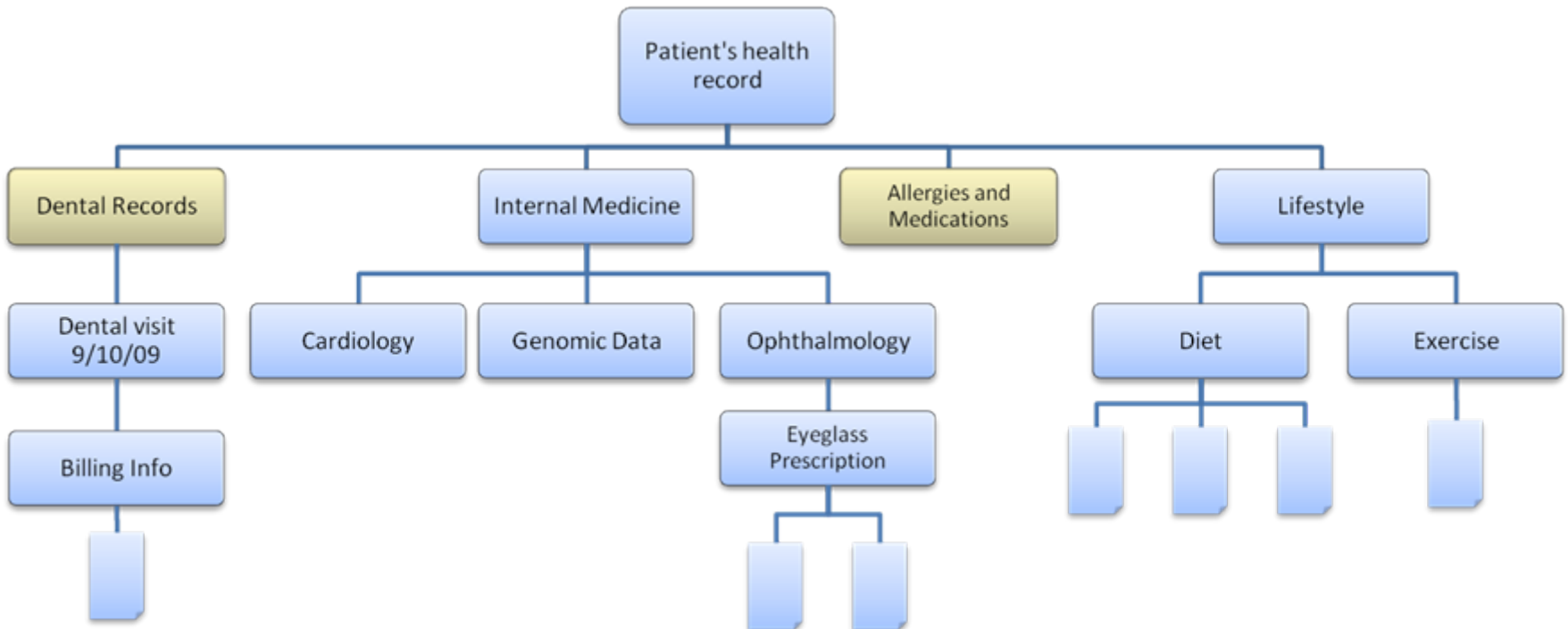
Hierarchical Health Records

- Ex:
 - Give Doctor access to entire record
 - Give Exercise info to cousin
 - Give Eyeglass Prescription to Retailer



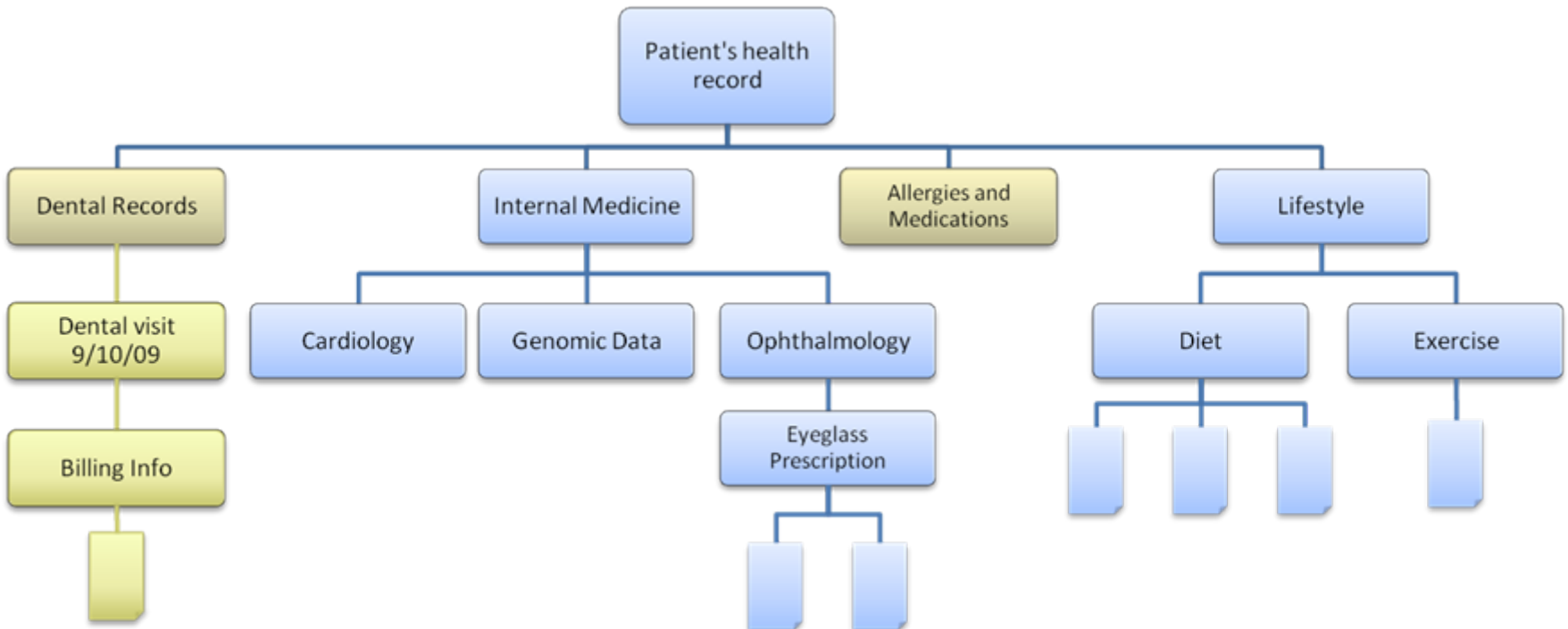
Hierarchical Health Records

- Ex:
 - Give Doctor access to entire record
 - Give Exercise info to cousin
 - Give Eyeglass Prescription to Retailer
 - Give Dental Information and Allergies and Medications to Dentist



Hierarchical Health Records

- Ex:
 - Give Doctor access to entire record
 - Give Exercise info to cousin
 - Give Eyeglass Prescription to Retailer
 - Give Dental Information and Allergies and Medications to Dentist

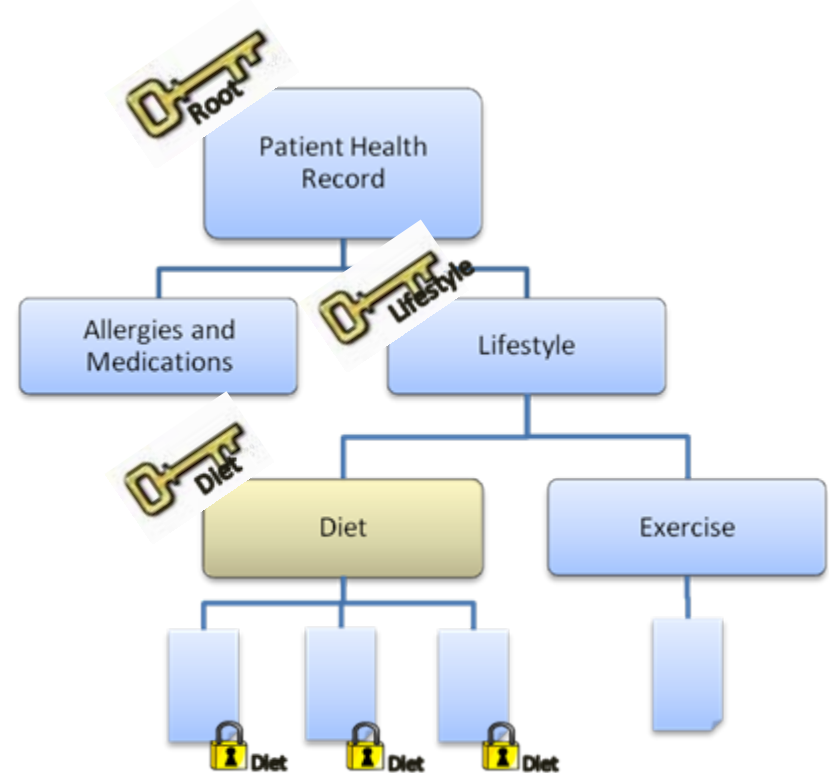
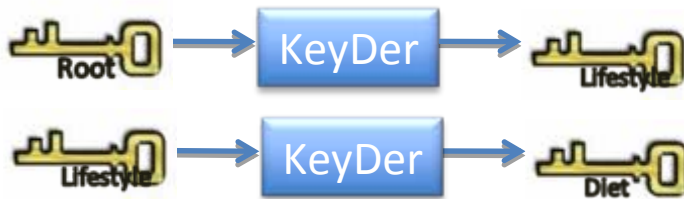


Hierarchical Encryption

Symmetric Key [AT83, S88, ...]

Public Key: HIBE[GS02, ...]

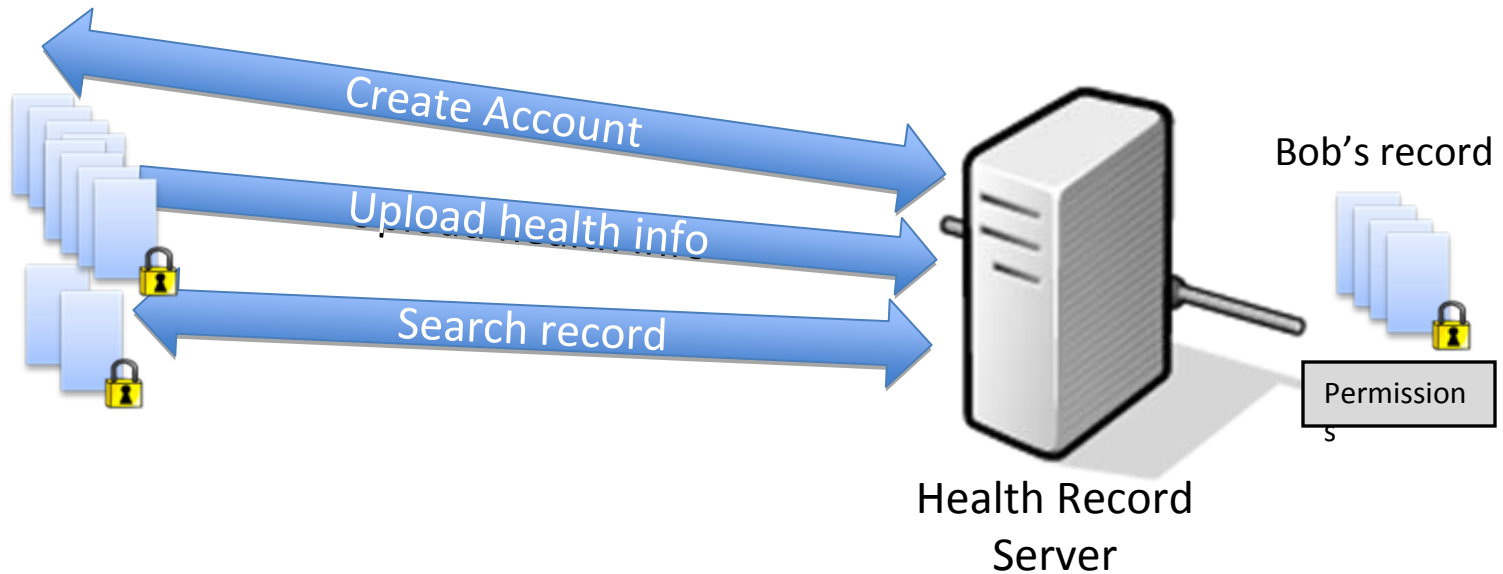
- Security
 - only have access if an appropriate key was given



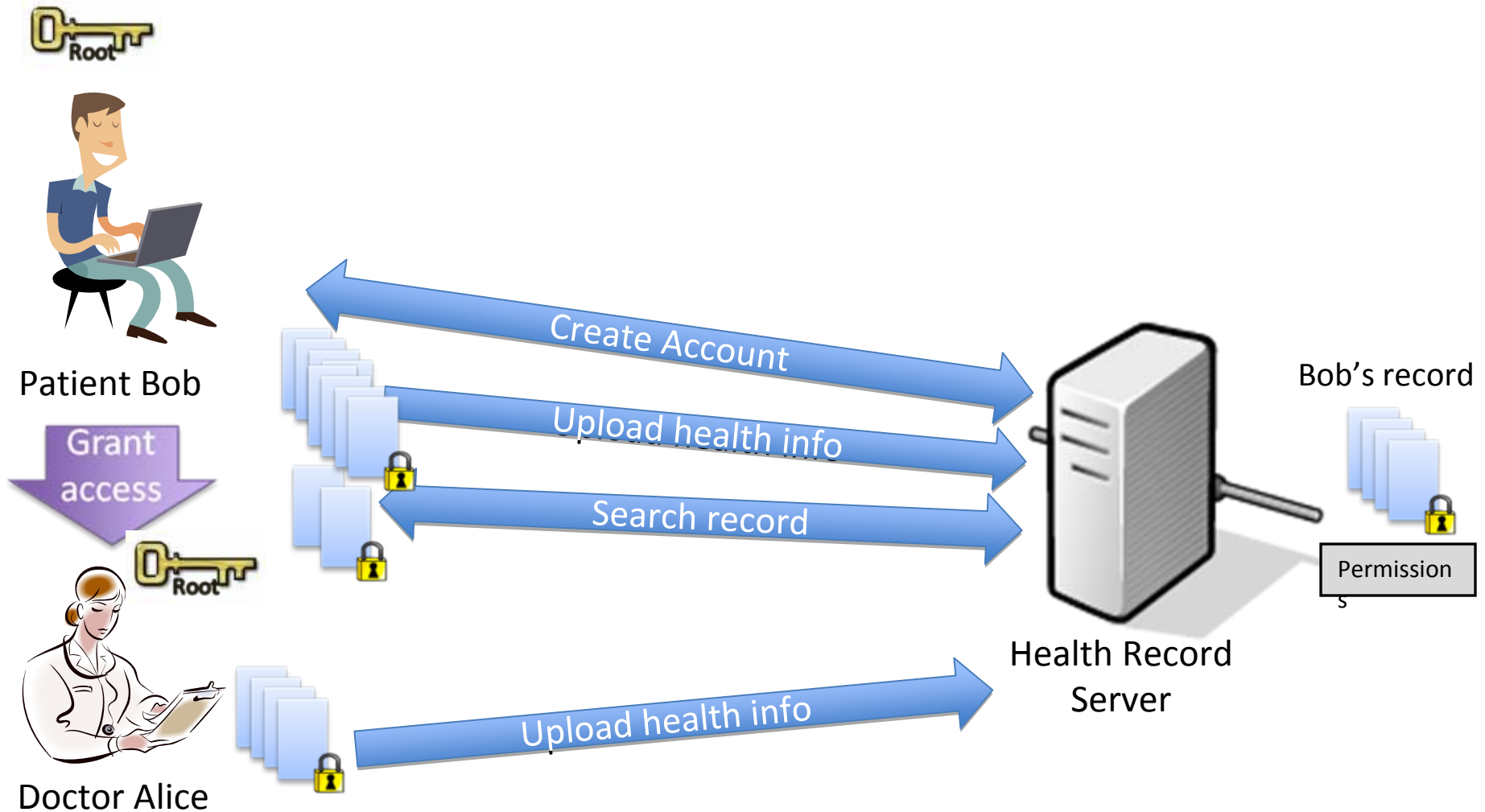
Patient Controlled Encryption



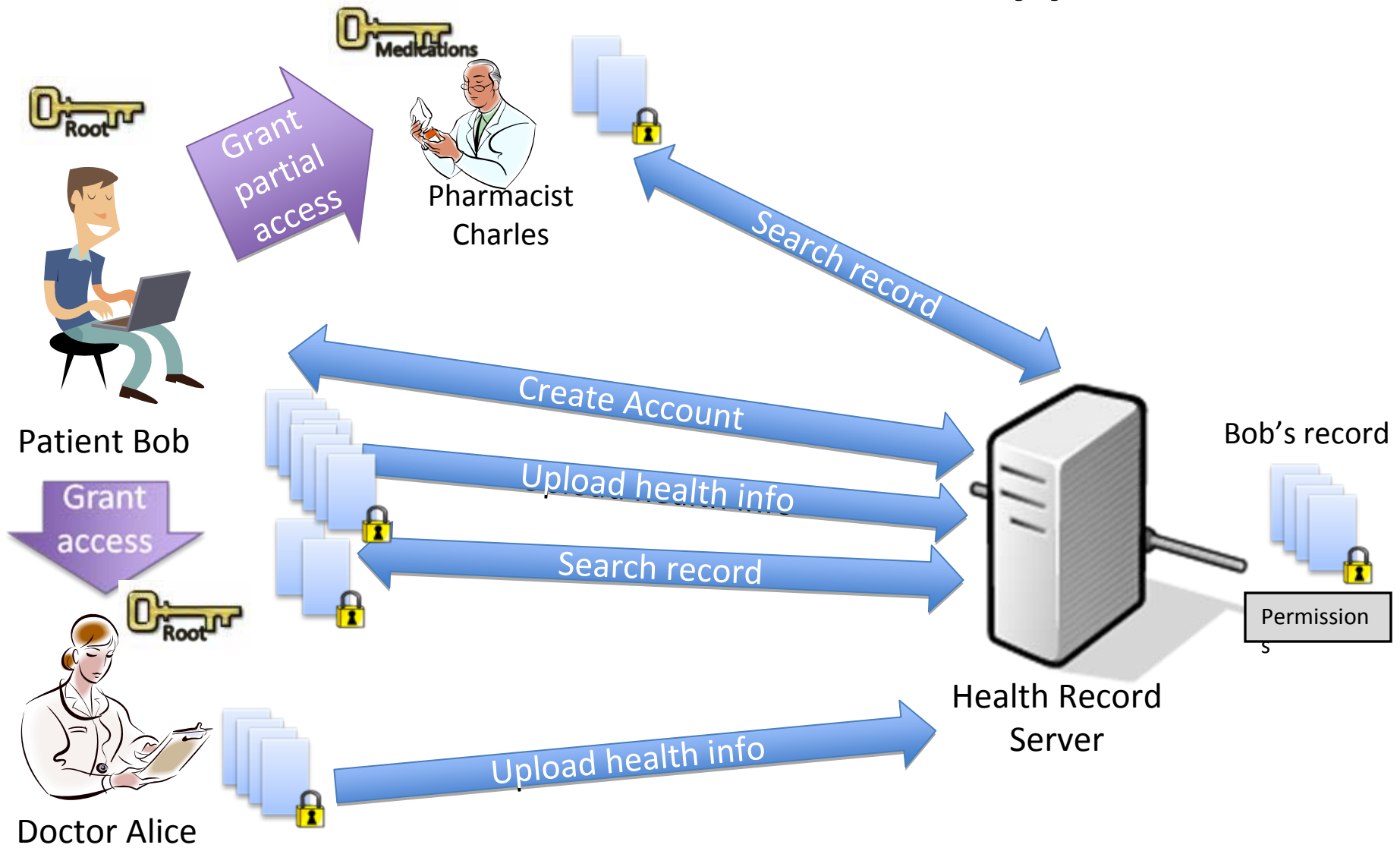
Patient Bob



Patient Controlled Encryption



Patient Controlled Encryption



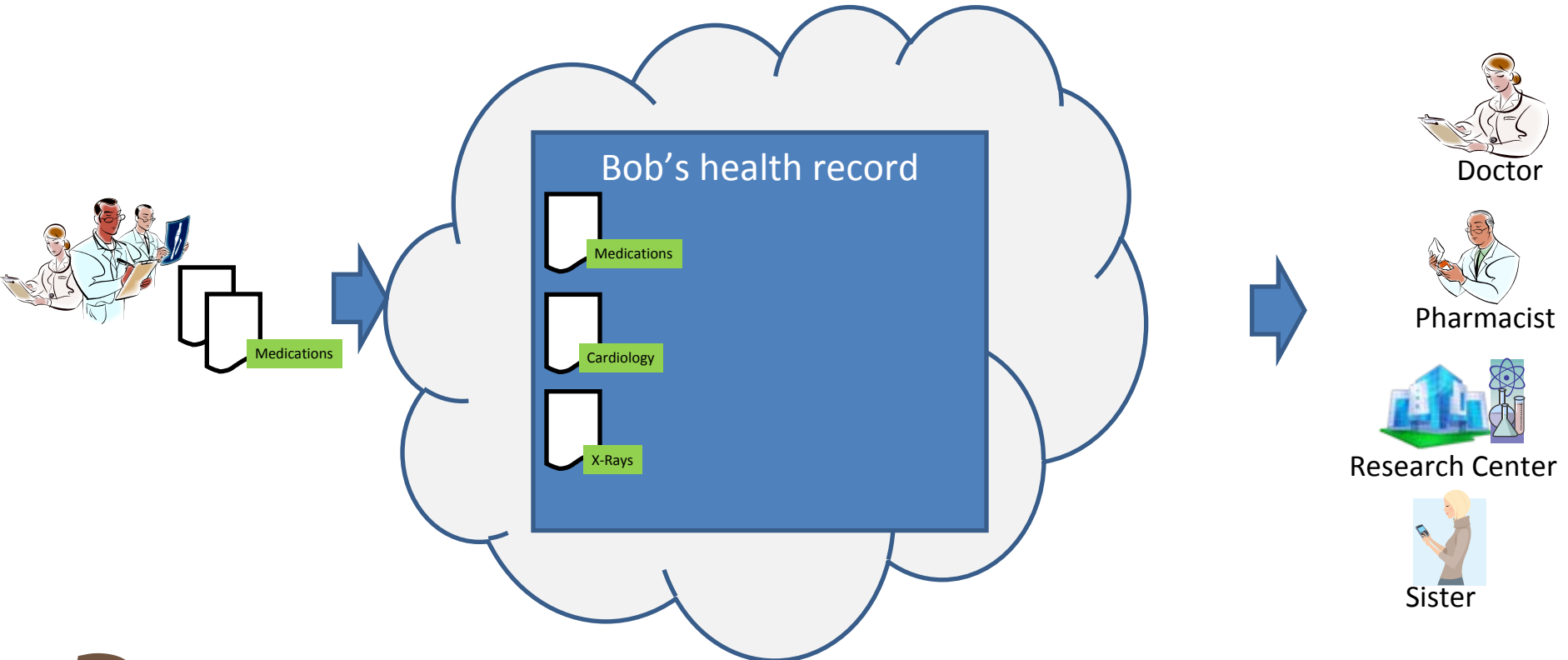
Patient Controlled Encryption

- All health data is stored encrypted
 - Only patient knows the key
 - Secure against theft, compromise, or insiders
- Patient can securely give partial access
- Also consider how to:
 - Search efficiently without leaking information
 - Hide hierarchy structure and category names
 - Hide identity of users accessing the record

Re-Encryption based Sharing

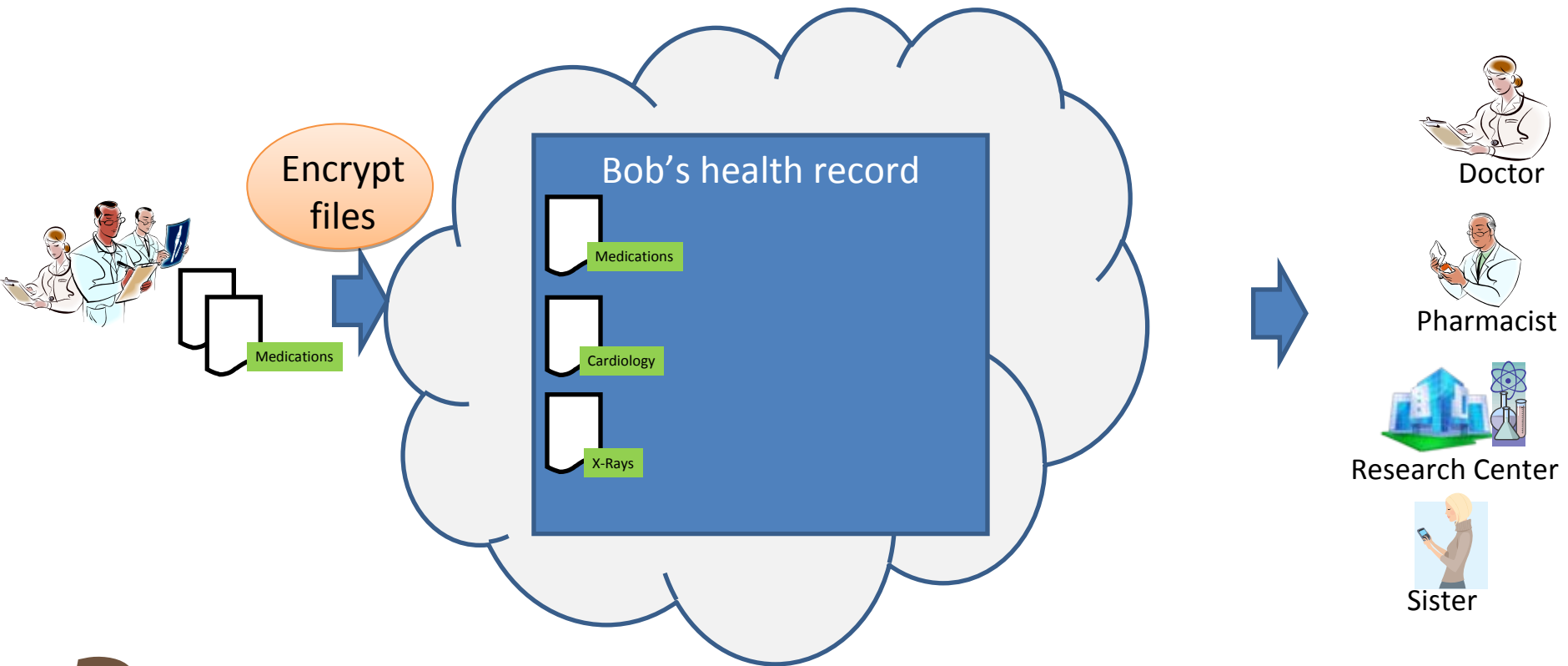
Joint work with Nishanth Chandran
and Vinod Vaikuntanathan

An example: Cloud storage for patient health records



- Access Policy**
- Doctor: everything
 - Pharmacist: medications
 - Research: everything except mental health
 - Sister: medications and emergency info

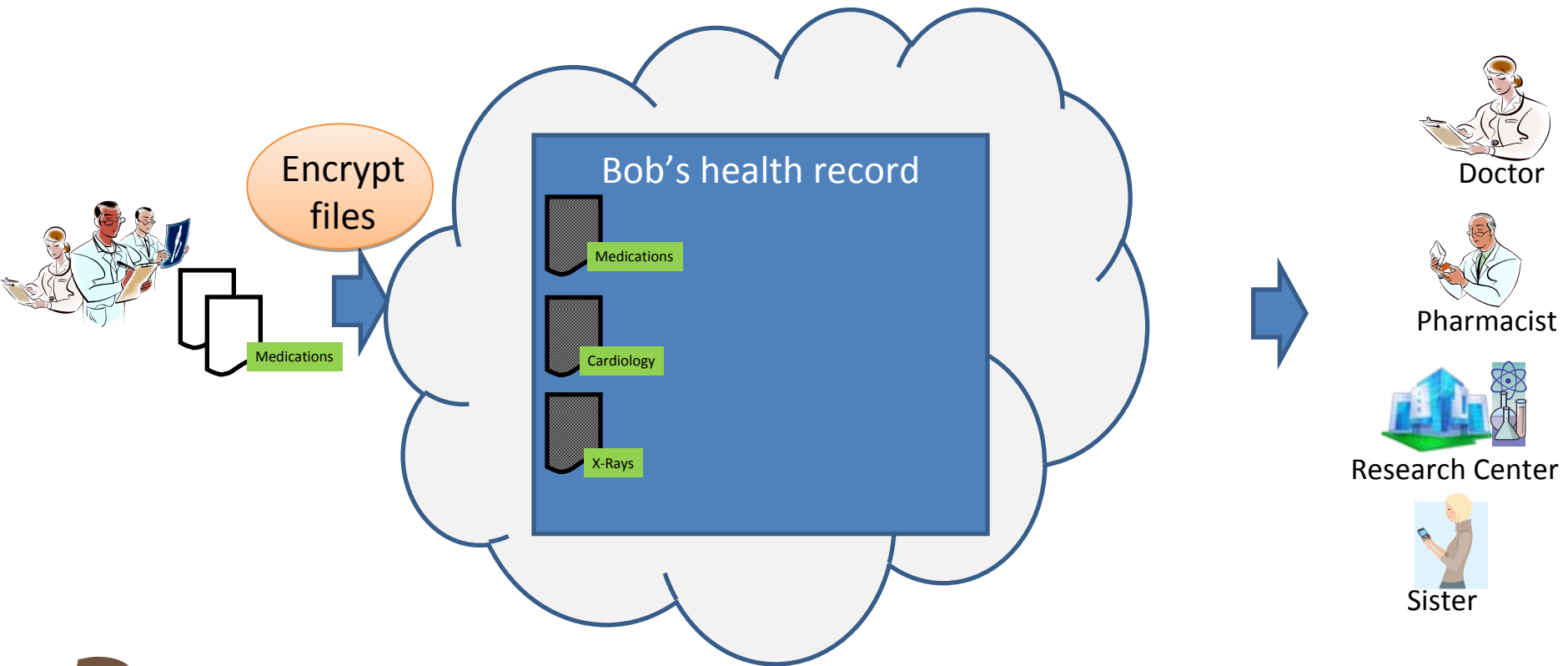
Encrypted Cloud Storage



Access Policy

- Doctor: everything
- Pharmacist: medications
- Research: everything except mental health
- Sister: medications and emergency info

Encrypted Cloud Storage

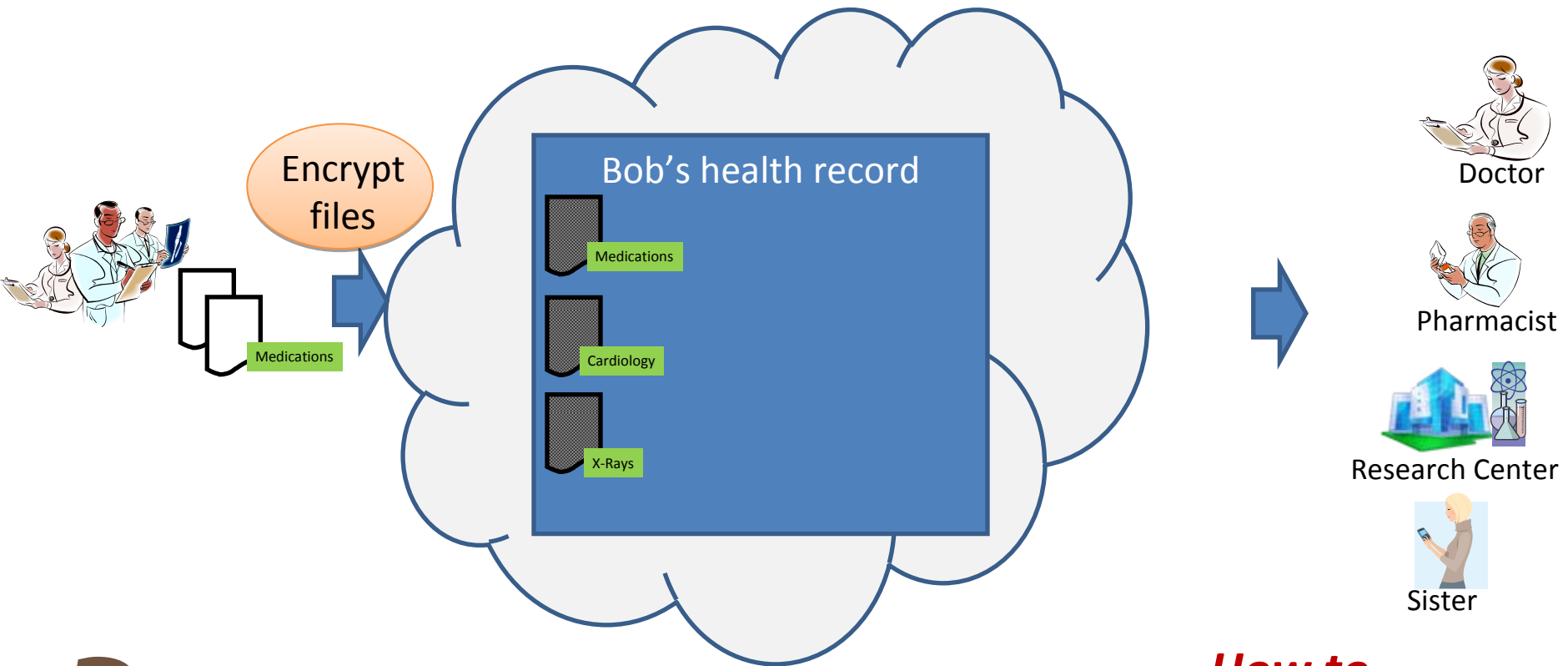


Access Policy

- Doctor: everything
- Pharmacist: medications
- Research: everything except mental health
- Sister: medications and emergency info



Encrypted Cloud Storage



Access Policy

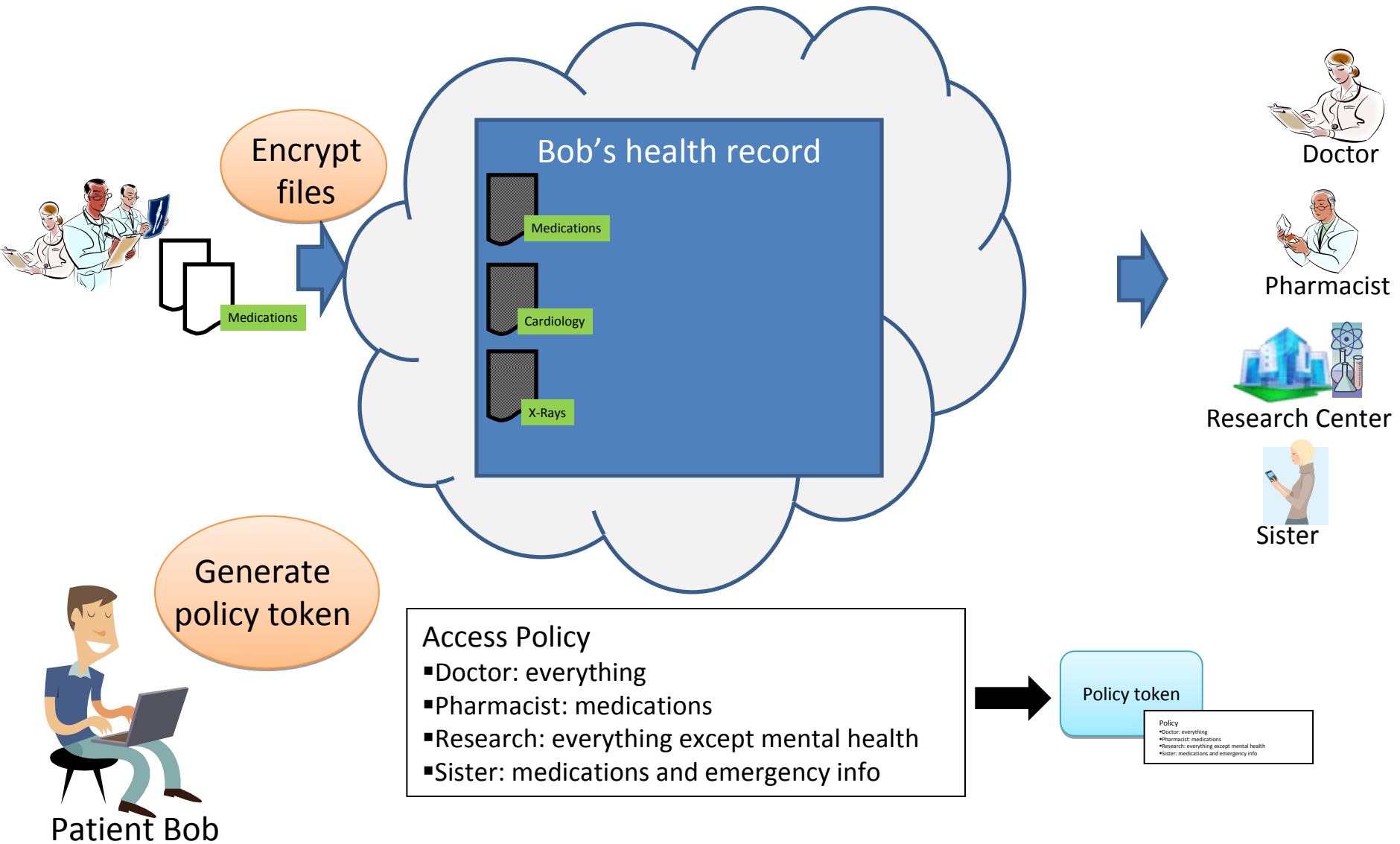
- Doctor: everything
- Pharmacist: medications
- Research: everything except mental health
- Sister: medications and emergency info

***How to
implement
access policy??***

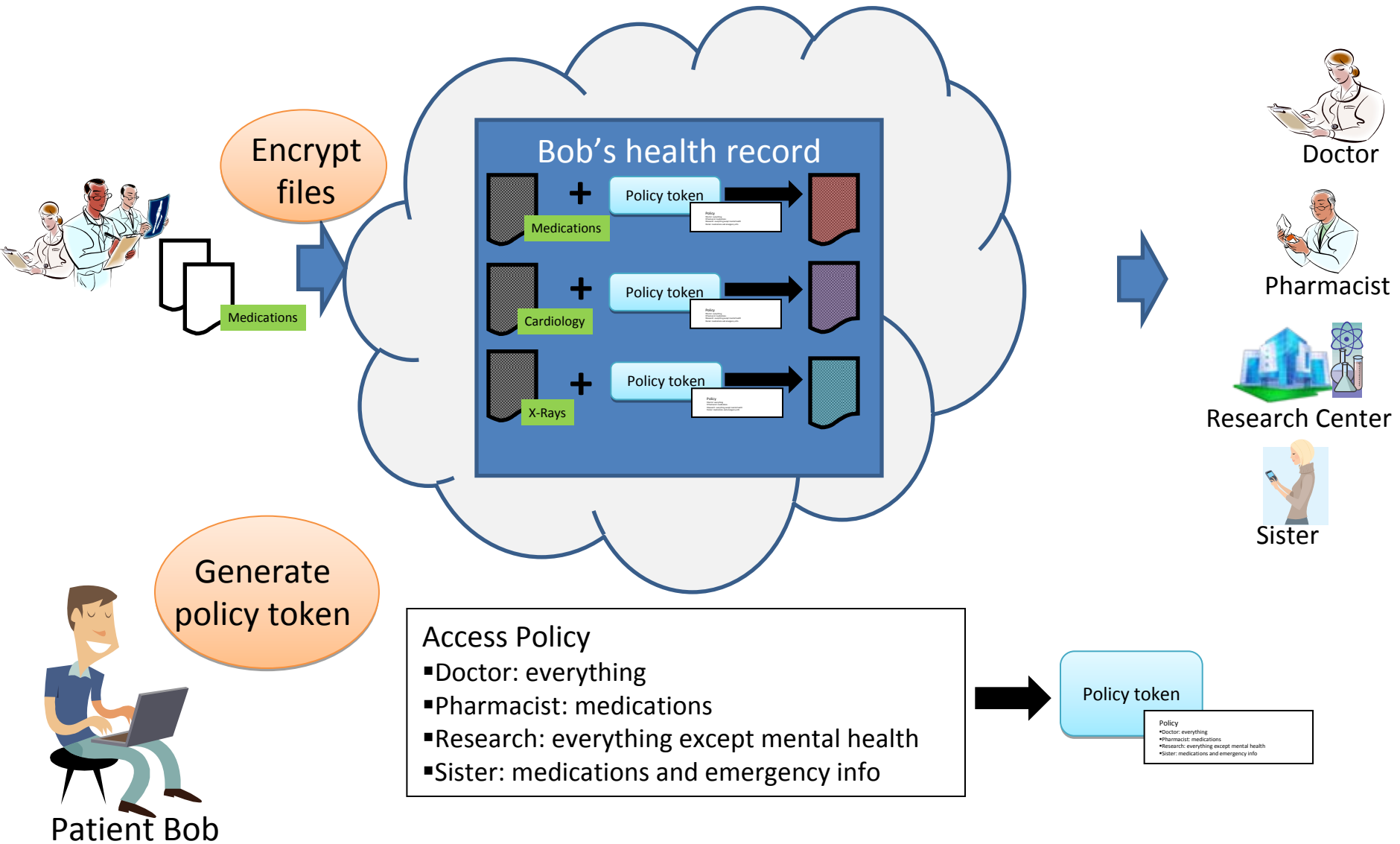
Access control for encrypted data

- ***How can we implement access control when data is encrypted?***
- Our goal:
 - Allow server to perform access control *on encrypted data*
 - Server will:
 - take files encrypted for Bob
 - transform them into files encrypted for appropriate recipient
 - *without decrypting anything.*
 - **Server cannot decrypt!**

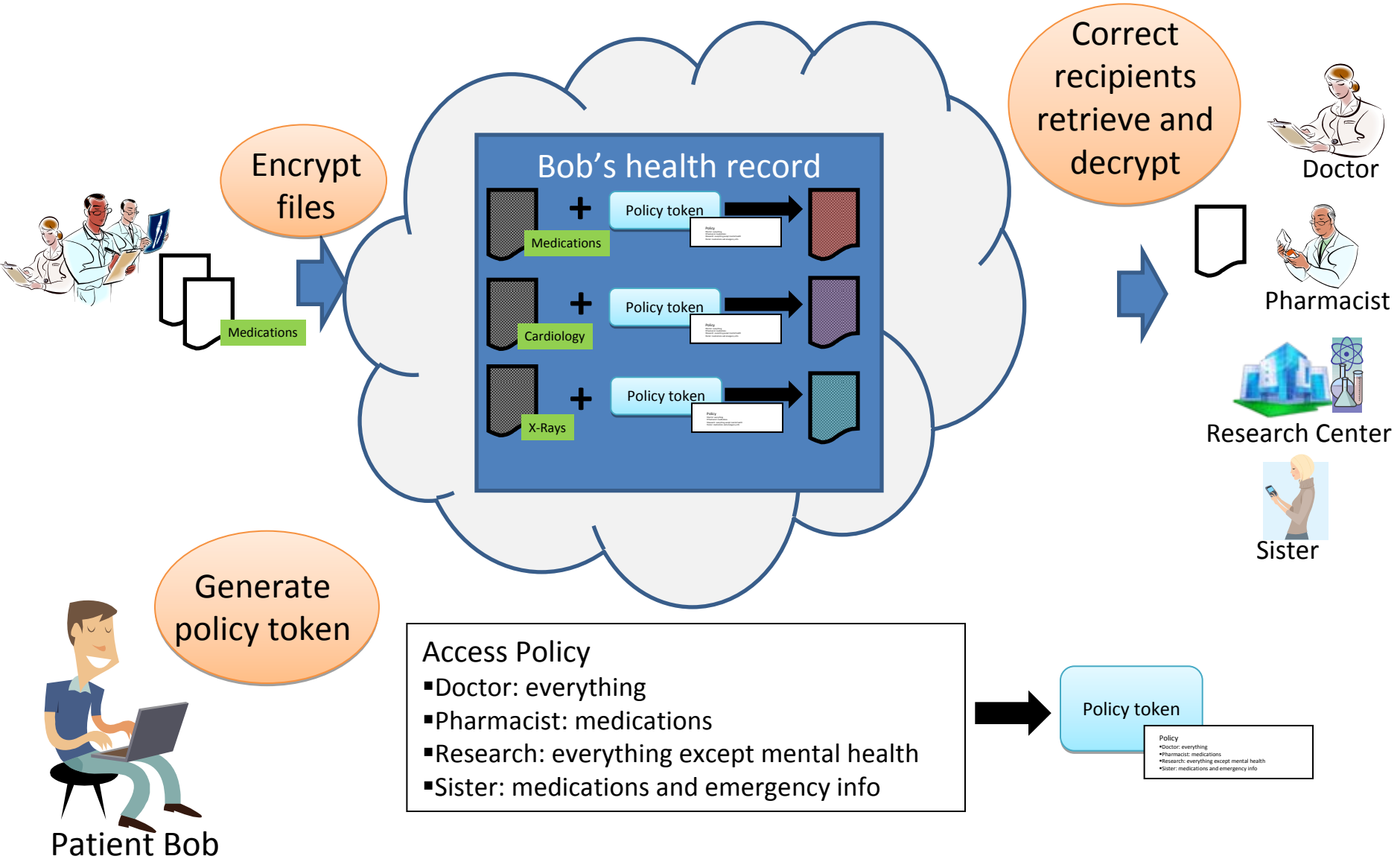
Access control for encrypted data: Our approach



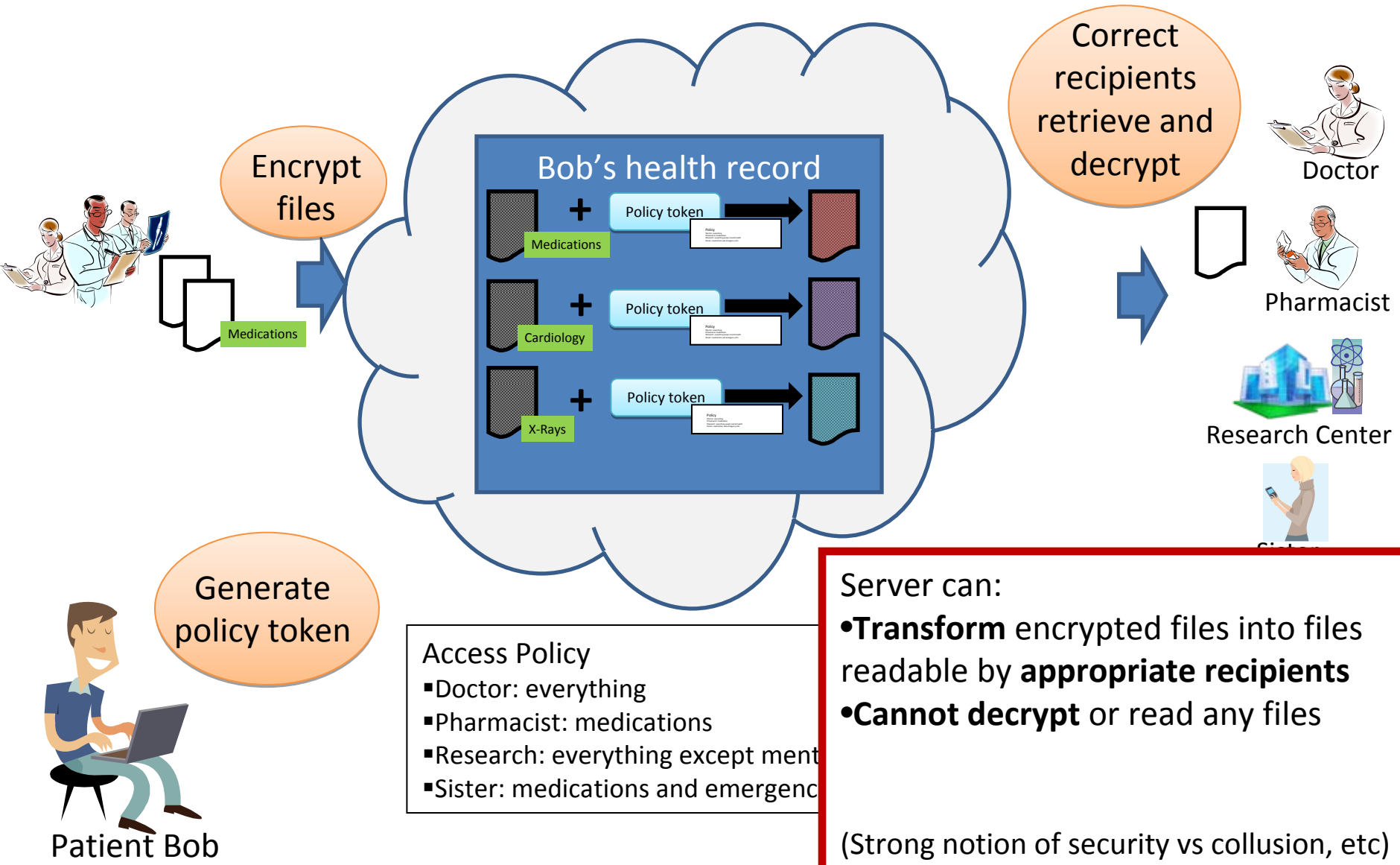
Access control for encrypted data: Our approach



Access control for encrypted data: Our approach



Access control for encrypted data: Our approach



- Access Policy
- Doctor: everything
 - Pharmacist: medications
 - Research: everything except ment
 - Sister: medications and emergenc

Server can:

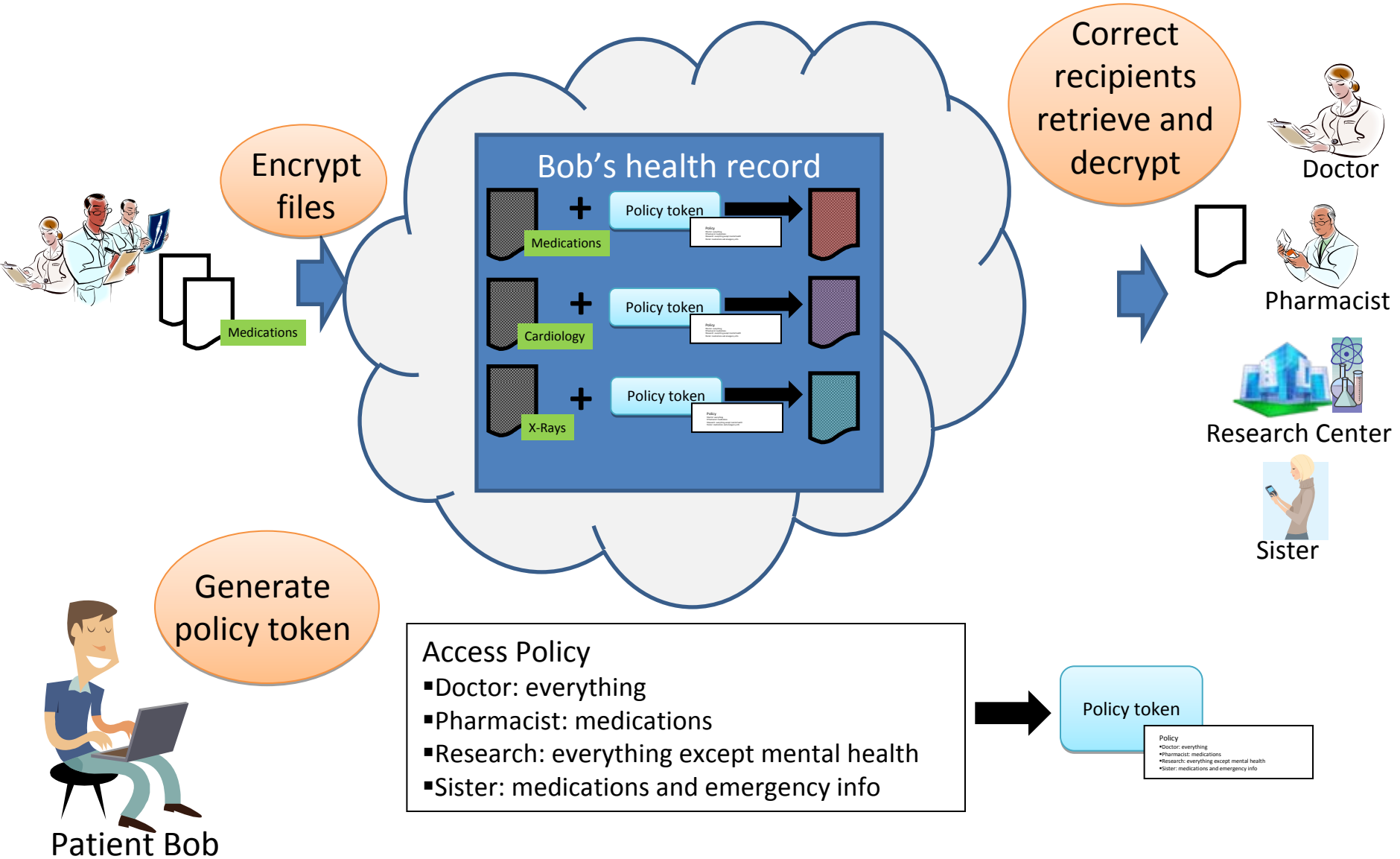
- **Transform** encrypted files into files readable by **appropriate recipients**
- **Cannot decrypt** or read any files

(Strong notion of security vs collusion, etc)

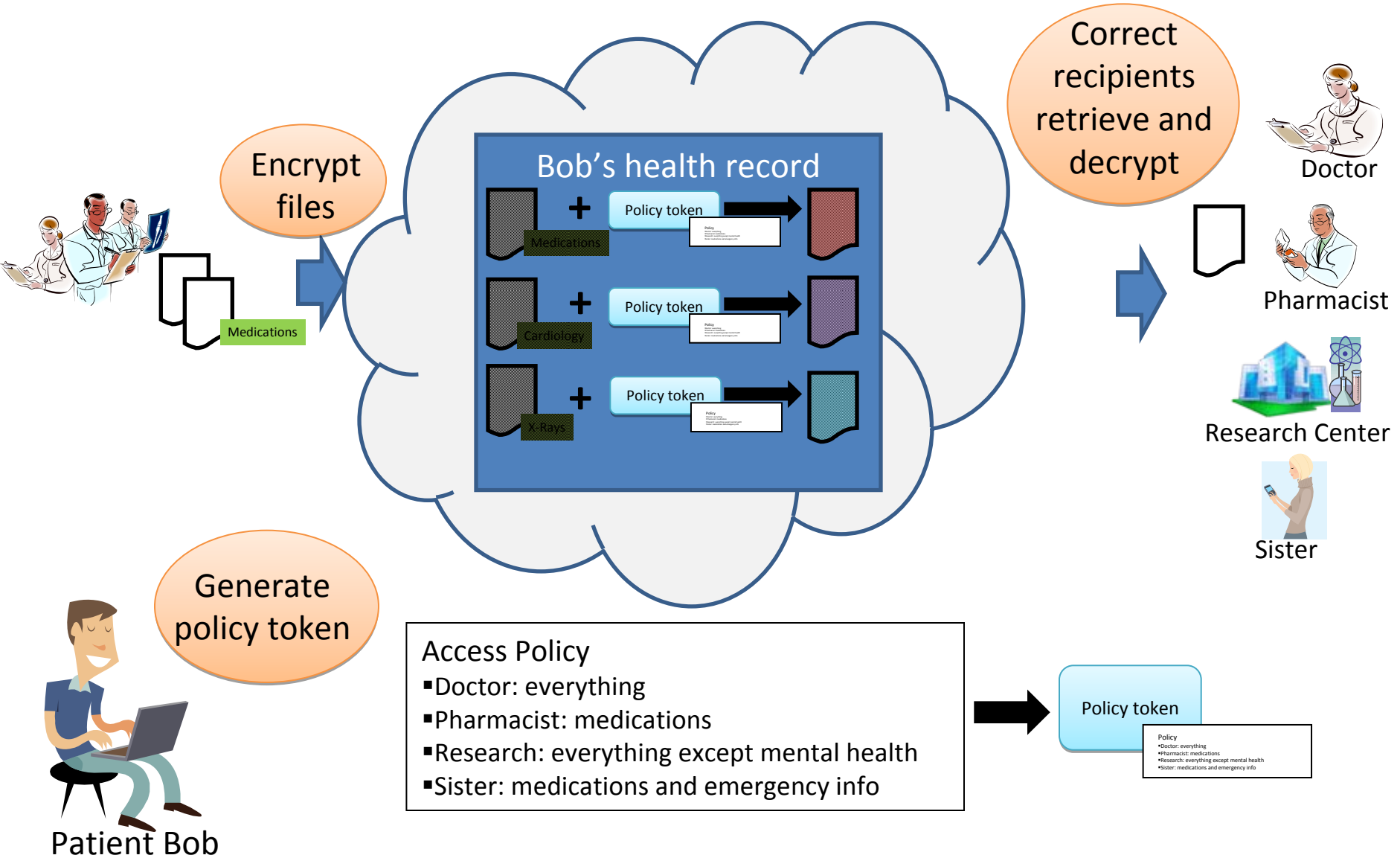
Private Access Control

- Want “private cloud”, where sensitive data is hidden even *from cloud operator*
- Previous scenario: server can implement policy but cannot decrypt any ciphertexts
- ***But sometimes the policy itself is private!***
 - E.g.:
 - Whether patient has chosen to participate in research project
 - Policy for mental health records
 - May want to give access to one family member without revealing that fact to others
- Question: ***Can we allow the same functionality, but without revealing the policy, even to the cloud provider?***

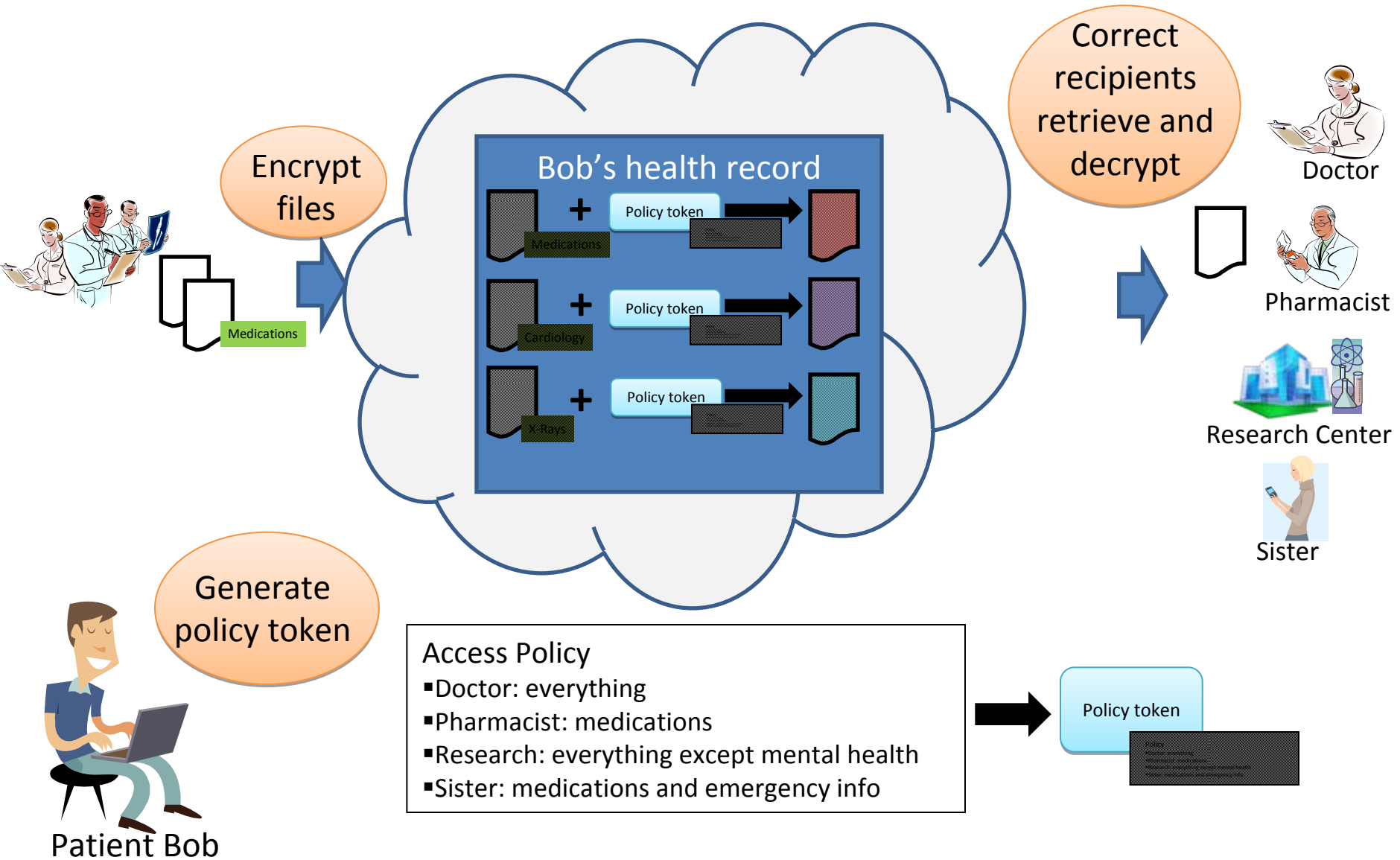
Access control for encrypted data: Our approach



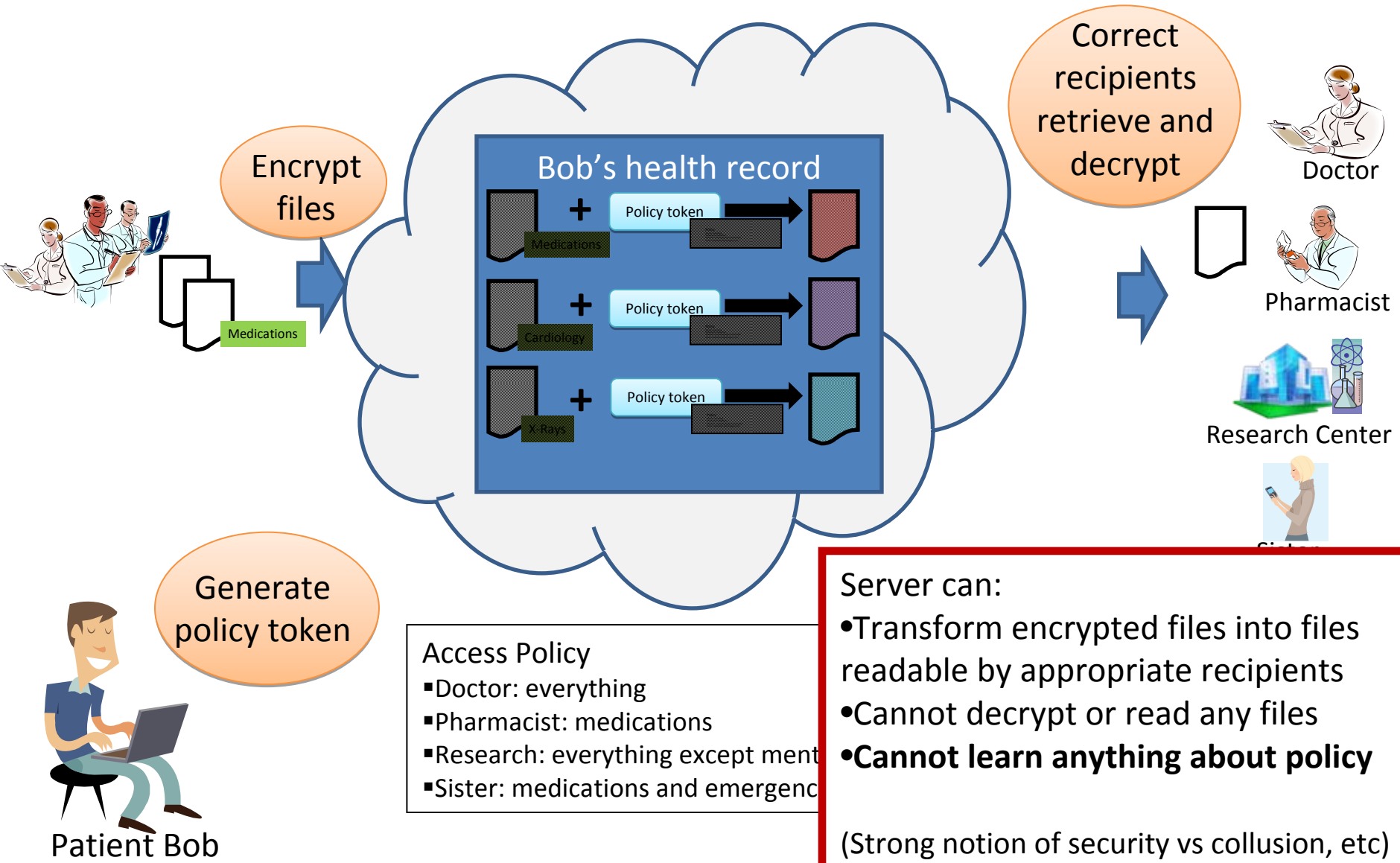
Access control for encrypted data: Our approach



Access control for encrypted data: Our approach



Access control for encrypted data: Our approach



Constructions

- Private access control on encrypted data
 - Hides policy and tags
 - For very simple policies
- Construction based on pairings
 - Relatively mild assumptions about pairing curves



- Can also achieve more efficient constructions where
 - Policy and tags are not hidden
 - Fairly general formulas
 - Similar to work on outsourcing ABE decryption [GHW 2011]

Advantages

- Private data is hidden from server:
 - Security against server compromise, theft, untrusted operators, etc
- *As secure as* patient downloading, decrypting, re-encrypting
 - Even if server colludes with recipients
- Patient only online to set/change policy
- Access control is invisible to recipients
 - Decryptor's efficiency independent of policy
 - Policy is hidden from recipients
 - Revocation is invisible for recipients

Conclusions and Future Work

Open Issues

- Additional privacy concerns
- Key backup 
- Identification
- Usability 

Conclusions

- Electronic Medical Records present risks to privacy
- Access control is not sufficient
- Encryption + Patient Control is the right approach
- 2 approaches for partial access when files are encrypted
 - Hierarchical sharing
 - Re-encryption based sharing

Questions

