# *"Cloud Computing Security & Privacy Survey"*

CS 848 – Class Project Presentation

*Mar 29th, 2010*

# *Outline*

## *Security*

Information security in cloud

*Atif Khan*

Accountability in cloud computing

*Somayyeh Zangooei*

## *Privacy*

Privacy from identification

*Kimiisa Oshikoji*

Privacy management in cloud computing

*Jason Ho*

Designing privacy-aware clouds

*Daniel Isaacs*

# *Introduction*

- Security
  - Technology, provides assurance
    - confidentiality
    - integrity, authenticity
- Privacy
  - Right, provides control
    - anonymity
    - primary & secondary use

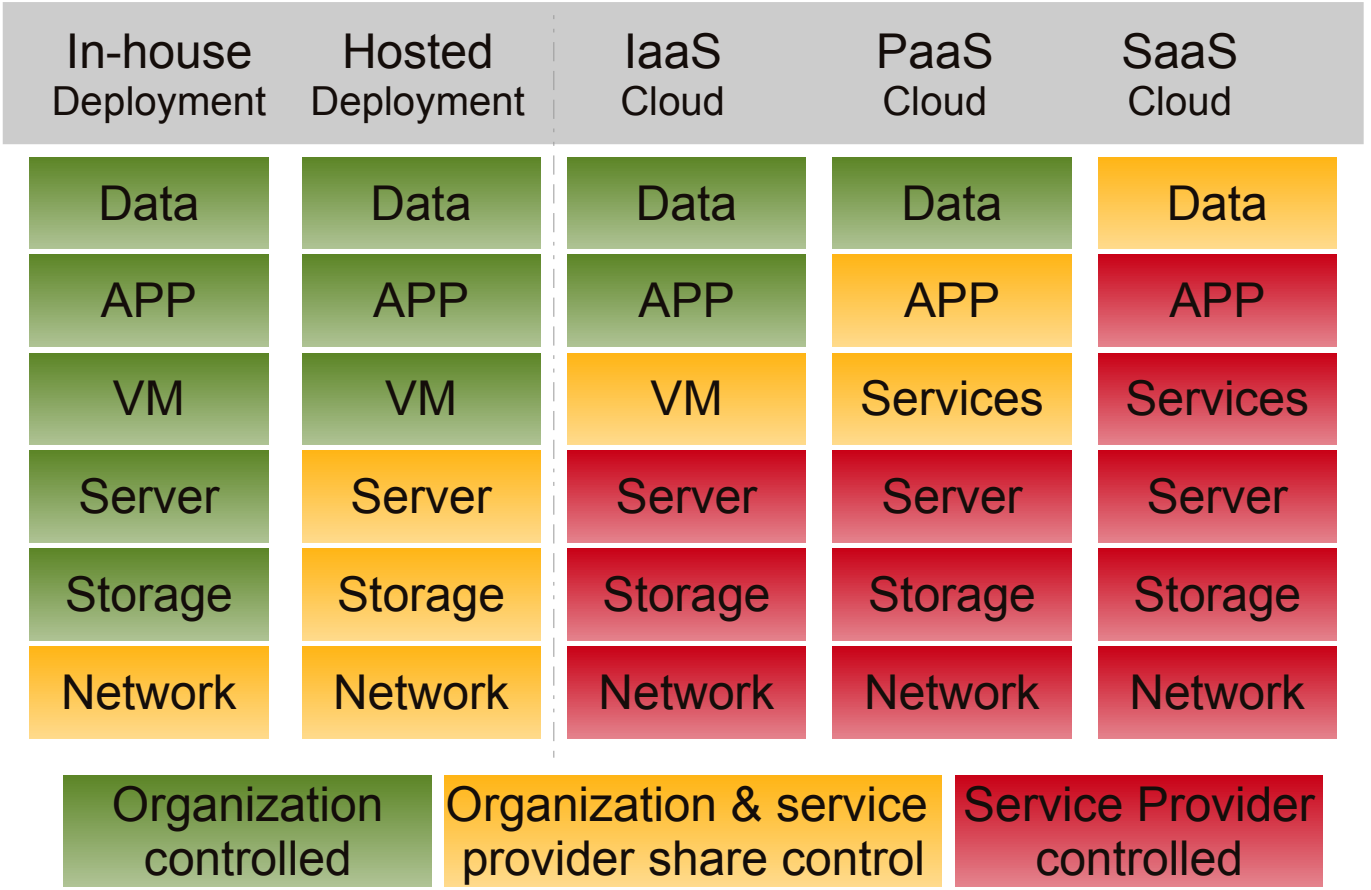
- Cloud Services
  - IaaS (*infrastructure as a service*)


  - PaaS (*platform as a service*)



  - SaaS (*software as a service*)

# *Control Boundaries*




[1] **Visualizing the Boundaries of Control in the Cloud. Dec 2009.**
http://kscottmorrison.com/2009/12/01/visualizing-the-boundaries-of-control-in-the-cloud/

# *Information Security in Cloud*

Presentation by Atif Khan

# *Information Security Concerns*

- Confidentiality-*"safe from prying eyes"*
  - communication, persistence
- Authenticity-"*data is from a known source*"
- Integrity-*"data has not been tampered with"*
  - provenance (computation)
  - persistence
- Non-repudiation-*"assurance against deniability"*

# *Information Security Concerns*

- Access control - *"access & modification by privileged users"*
  - individual vs. group access
  - multi-tenancy (PaaS, SaaS)
- Long term security
  - change in authentication/authorization
  - proof of possession
  - confidentiality
    - crypto systems do not provide long term guarantees
  - intersection attacks

# *Security Enhancing Techniques*

- Encryption
  - Symmetric encryption (*data*)
  - Public key cryptography (*identity, authentication*)
    - secret private key, published public key
  - Hash / Message Authentication Code (*integrity*)
  - Digital Signatures (*authentication, non-repudiation*)
  - TLS/SSL (*communication*)

# *Security Enhancing Techniques*

- Encryption
  - Homomorphic encryption [2]
    - allow for arbitrary computing over encrypted data
      - if $E(p) = c$ then $D(2c) = 2p$ (multiplication operation)
      - allows for data processing without decryption
    - promising but *not practical* so far [3]
  - Key management challenges
    - increase as the access control granularity increases

[2] Gentry, C. 2009. Fully homomorphic encryption using ideal lattices. In Proceedings of the 41st Annual ACM Symposium on theory of Computing (Bethesda, MD, USA, May 31 - June 02, 2009). STOC '09. ACM, New York, NY, 169-178.

[3] Bruce Schneier. Schneier on Security. http://www.schneier.com/blog/archives/2009/07/homomorphic_enc.html

# *Security Enhancing Techniques*

- Secure query & search
  - PIR/SPIR
    - *"allows a user to retrieve an item from the server without revealing the item to the database"*[4]
    - requires more work

[4] Chor, B., Kushilevitz, E., Goldreich, O., and Sudan, M. 1998. Private information retrieval. J. ACM 45, 6 (Nov. 1998), 965-981.

# *Security Enhancing Techniques*

- Secure query & search
  - Encrypted data search
    - matching with encrypted keywords
      - meta-data driven
      - single party query
    - secure anonymous database search (SADS)[5]
      - multi party queries
    - not easy, may require trusted third parties

[5] Raykova, M., Vo, B., Bellovin, S. M., and Malkin, T. 2009. Secure anonymous database search. In Proceedings of the 2009 ACM Workshop on Cloud Computing Security (Chicago, Illinois, USA, November 13 - 13, 2009). CCSW '09. ACM, New York, NY, 115-126.

# *Security Enhancing Techniques*

- Remote data checking
  - Client preprocessing
    - data in chunks along with MAC for each chunk
    - server stores *data chunk + MAC* combinations
    - forward error correction
      - long term recoverability

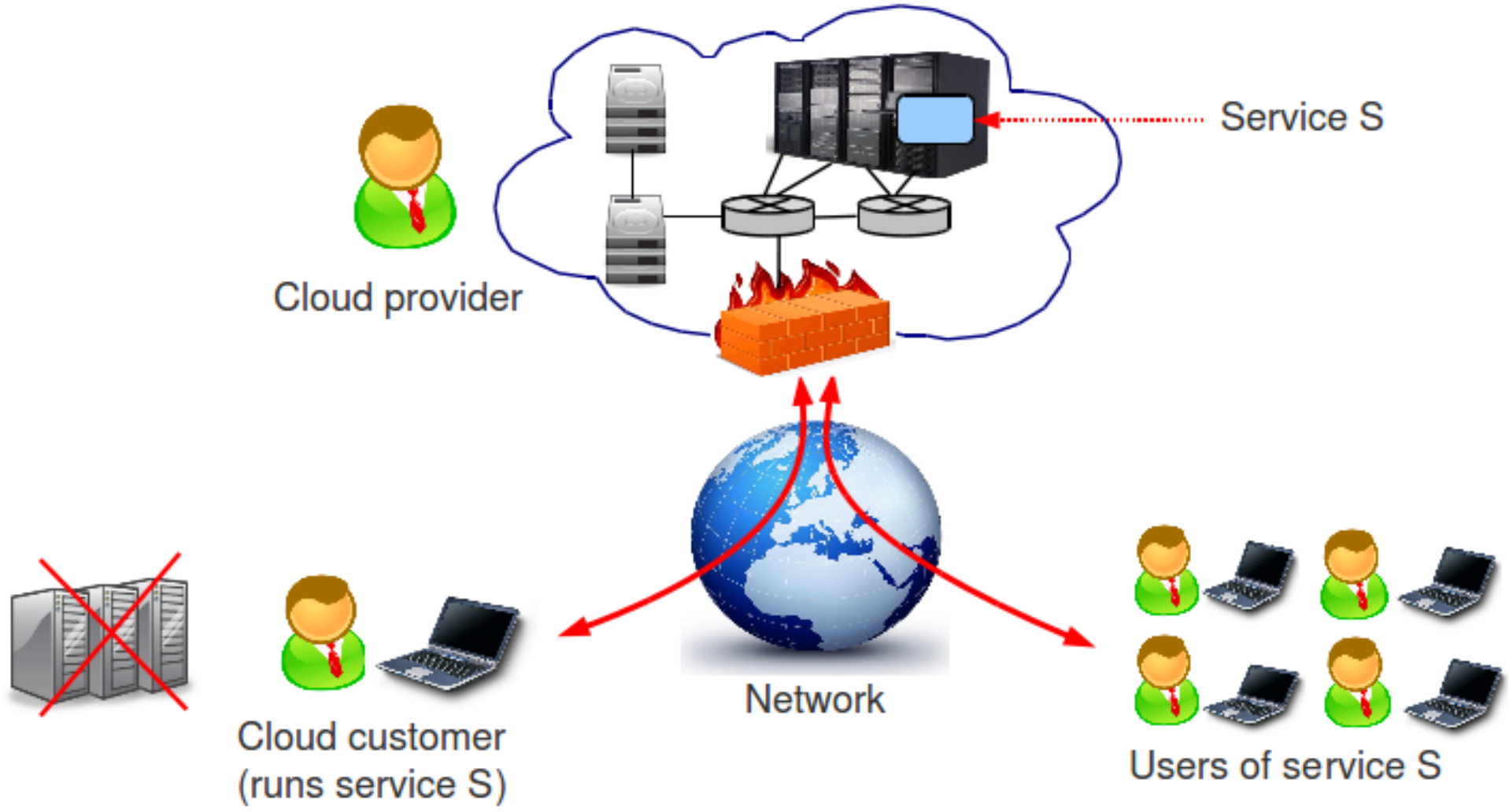
# *Data Remanence*

- Data Remanence
  - "*Residual representation of data after purge*"
  - How to purge data in cloud?
    - risk at all levels (SaaS, PaaS, and IaaS)
  - Secure deletion
    - encrypt the data in the cloud
    - data deletion = key destruction

# *Accountability in Cloud Computing*

Presentation by Somayyeh Zangooei

# *Cloud Computing*



# *Split Administrative Domain*

- Cloud customer loses control over his computation and data
- What if something goes wrong?
  - Example: LinkUp
- Management responsibilities are split
- Who should address the problem?
  - Provider: does not understand details of computation
  - Customer: has only remote access to cloud and thus limited information

# *Handling Problems*

- Who is responsible?
- Customer's perspective:
  - If something is wrong, how will I know? (detection)
  - How can I tell if it's my fault or the cloud's fault?
  - If it's the cloud's fault, how can I convince the provider?

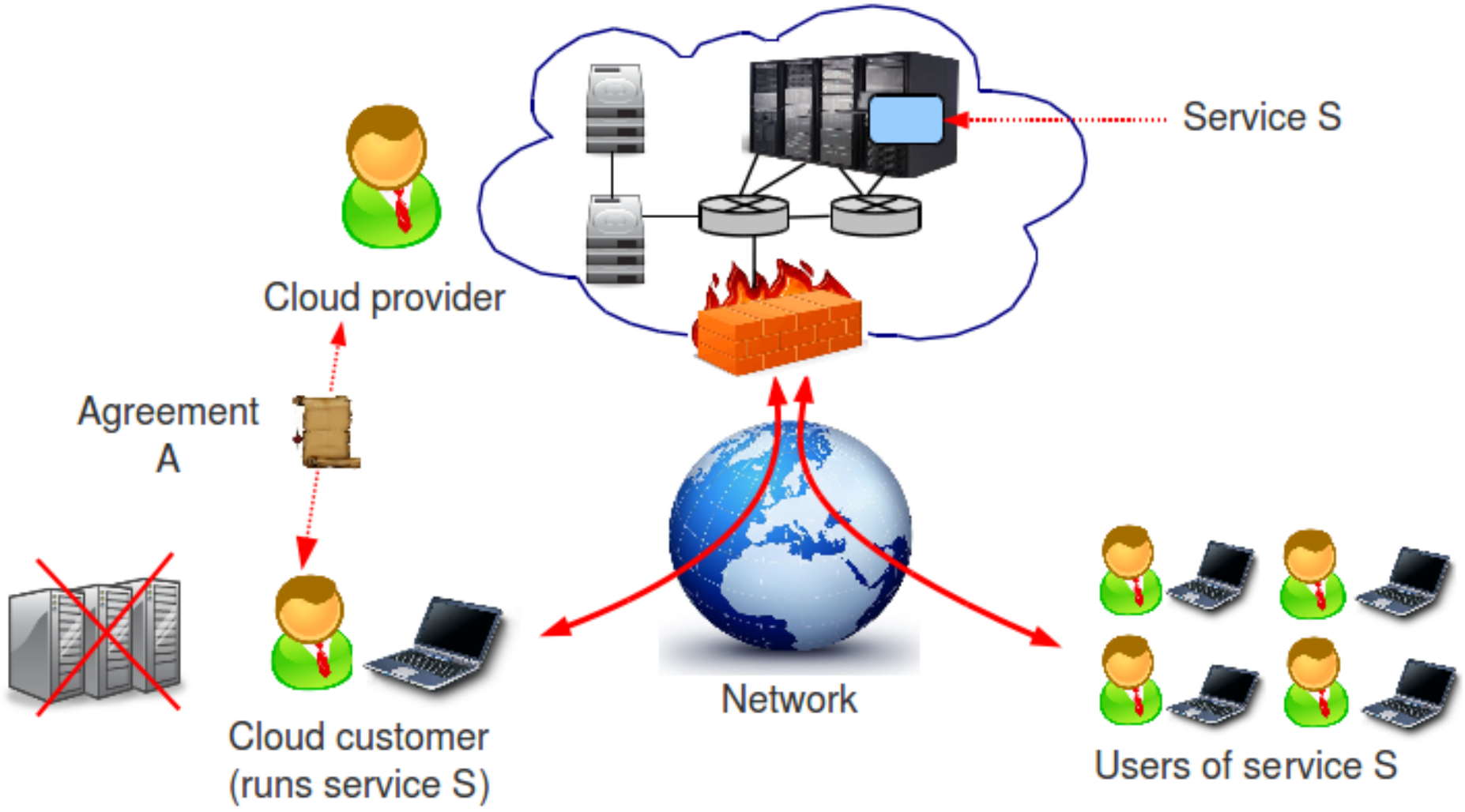
# *Handling Problems*

- Who is responsible?
- Provider's perspective:
  - If something is wrong, how will I know? (detection)
  - How can I tell if it's my fault or the customer's fault?
  - If it's the customer's fault, how can I convince the customer?

# *Accountable Clouds*

- A cloud is accountable if
  - Faults can be reliably detected
  - Each fault can be linked to one party (customer or provider)

# *Cloud Computing*



# *Audit*

- Customer wants to run service S on the cloud
- Agreement A: How the cloud should run S
- Customer can call an Audit primitive
- Audit (A,S,t1,t2): Checks whether the cloud has fulfilled A during the interval [$t_1$..$t_2$] for service S

# *Accountable Clouds*

- Properties of accountable clouds
  - Completeness: If the agreement is violated, Audit will report this violation
    No false negative
  - Accuracy: If the agreement is not violated, Audit will not report a violation
    No false positive
  - Verifiability: Audit produces evidence that would convince a disinterested third party

# *Tamper-Evident Log*

- A possible approach for accountability:
  - Cloud records its actions in a tamper-evident log
  - Cloud customer and provider can audit the log and check for faults
  - Use log to construct evidence that a fault does (not) exist

# *Benefits of Accountable Clouds*

- Customer's incentives
  - Can detect violations
  - Can hold the provider responsible
- Provider's incentives
  - Attractive to prospective customers
  - Helps with handling angry support calls

# *Privacy from Identification*

Presentation by Kimiisa Oshikoji

# *Protect User Identifies*

- What can identify a user?
  - Name
  - Birth date
  - Home Address
  - Where you work
  - Information you are interested in
  - Where you are

# *Questions*

- Would it help to encrypt the data?
  - Who is responsible?
- Is the solution downloading the entire database?
- Could spreading out the data over multiple servers help?
- Who do we need to protect against?

# *PIR*

- Private Information Retrieval
  - Identity of the record being accessed is hidden
  - For single server database
  - For multiple server database

# *SPIR*

- Symmetric Private Information Retrieval
  - Oblivious transfer
  - User's knowledge is restricted to only what they request

# *ORAM*

- Oblivious RAM
  - Data is managed by the user
  - Server has no knowledge or control over data

# *Privacy Management in Cloud Computing*

Presentation by Jason Ho

# *Privacy Management*

- Privacy can be protected by means of:
  - encryption
  - privacy policy setup
- Third-party privacy manager

# *Encryption*

- Levels of encryption
  - No privacy
    - Unsensitive data
    - Cloud provider stores data without any form of encryption
  - Privacy with trusted cloud provider
    - Data is not encrypted before transferred to the cloud
    - Data is stored encrypted by a specific key provided by the cloud provider
    - The cloud provider is trusted to encrypt the data using its key.
  - Privacy with non-trusted cloud provider
    - Encryption outside of cloud provider by a data owner's key (on client end / trusted 3rd party)
    - Data cannot be accessed by the cloud provider

# *Encryption*

- Full encryption
  - Privacy is fully preserved
  - Private data stored in the cloud is entirely encrypted
- Partial encryption
  - Also called obfuscation
  - Portion of private data stored in the cloud is not encrypted
  - Need to set up policy on unencrypted data

# *Privacy Policy Setup*

- Allow data owner to set preference on her data in the cloud:
  - Data usage
  - User access control
  - Duration

# *3rd-Party Privacy Manager*

- Handles encryption and privacy policy
- Between clients and cloud provider
- Benefits
  - Transparency
  - Scalability
  - Vendor independency
- Further investigation
  - How to analyze the encrypted data

# *Designing Privacy-Aware Clouds*

Presentation by Daniel Isaacs

# *Guidelines For Design*

1. Minimize personal information sent to and stored in the cloud

- Analyze the minimal amount of information required from a customer in order for a cloud to operate.
- Cloud applications need to store only data which is planned to be used immediately.
- Storing data mechanisms can be lessen if there is less information to store in a cloud.

# *Guidelines For Design*

2. Protect personal information in the cloud

- Personal information has to be protected from any lost or theft created by intruders.
- Additionally, employees or third parties should only be give access to information they need to fulfill their business purpose.
- To ensure this, security safeguards can be used in order to prevent unauthorized access, copying, or modification of personal information.

# *Guidelines For Design*

3. Maximize user control

- Users or companies must be given access to control the data that is being stored about them.
- Giving control to users about their information generates trust.
- For example, users should be able to access a user interface to modify their personal information on the cloud at anytime.

# *Guidelines For Design*

4. Allow user choice

- Users must be presented with a choice whether they want to share their information or not.
- Designers can create opt in and opt out mechanism, to allow users to decide if they want to share their information or not.
- However, legal requirements for opt in and opt out mechanisms can vary between the different places a design may be used.

# *Guidelines For Design*

5. Specify and limit the purpose of data usage

- When the information is loaded into the cloud, it must be limited to the preferences and conditions set by a user or organization.
- Data usage has to be restricted only to the user's specified purpose.
- Cloud applications design should always validate the data usage against the allowed usage intentions.

# *Guidelines For Design*

6. Provide feedback

- Cloud applications should be user friendly and clearly indicate privacy functionality by using icons, providing tutorials, help documents, and visual metaphors.
- Applications need to be designed in a way that users are provided with feedback, allowing them to make knowledgeable decisions in terms of privacy.

# *Tradeoffs of Privacy-Aware Design*

- Solutions such as encryption, deprive cloud service providers the opportunity of merging identical data, which would reduce storage space.
- Additionally, encryption hinders the capability to index and process the data.

# *Privacy Designs*

1. A Client-Based Privacy Manager

- Goal is to reduce the risk of data leakage and the loss of privacy on sensitive data processed in a cloud.
- On the client side to help the user protect his privacy when accessing cloud services
- Nonetheless, the privacy manager requires the help from a server-side component for effective operation.

# *Privacy Designs*

## 1. A Client-Based Privacy Manager

- Design Features
  - Obfuscation
  - Preference setting
  - Data access
  - Feedback
  - Personae

# *Privacy Designs*

## 1. A Client-Based Privacy Manager

- Drawbacks
  - The privacy manager needs the full cooperation of the cloud service provider.
  - Cloud service providers that sell the user data to advertisers, may not be willing to allow users to preserve their privacy.

# *Privacy Designs*

## 2. A Virtual Private Data Repository

- Design a privacy-aware general mechanism to access data in cloud environment applications.
- The VPDR architecture is based on three components:
  - Virtual private disk (VPD)
  - Virtual network buffer (VNB)
  - Virtual cloud storage (VCS).

# *Privacy Designs*

## 2. A Virtual Private Data Repository

- Drawbacks:
  - The data could be deciphered with vast computing resources.
  - The VCS component complicates the process of deleting and migrating user data

# *Conclusion*

- Cloud offers a much weaker information security model, centred around encryption
- Accountability provides advantages for both cloud customer and cloud provider
- It is important that a cloud user's identity remain secure
- 3rd-party privacy manager gives data owner more control over her own data
- Privacy should be a fundamental design goal, and it should cover both the users and the service providers

# Thank you!