HITACHI
Inspire the Next

Securing stored data in the cloud

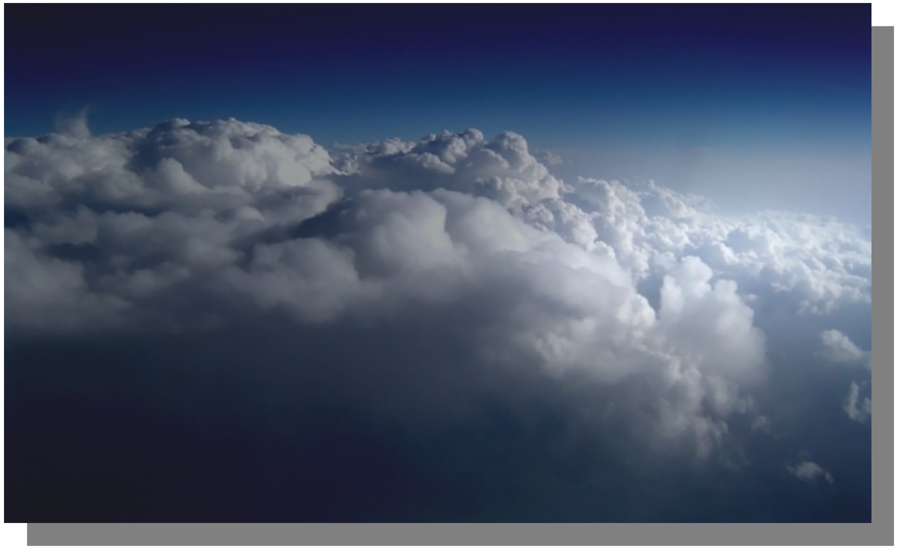
Jan 30th, 2009

Cyril Guyot
Hitachi GST Research,
Storage Architecture Group

Hitachi Global Storage Technologies

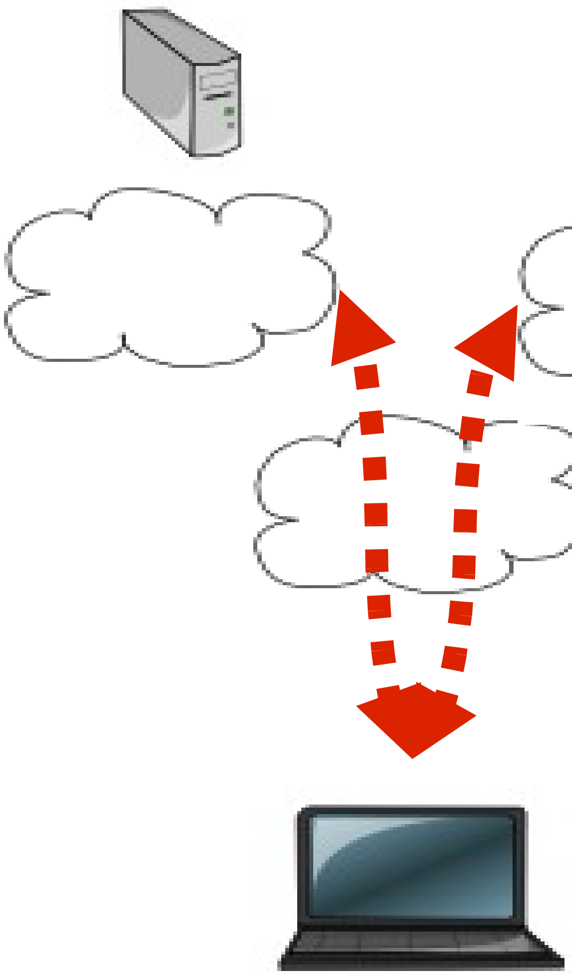
Talk outline

- **Cloud storage: definitions?**
- **Today's security in cloud storage**
  - Existing solutions
- **Future cloud storage security challenges**
  - ...and solutions!



Cloud storage definition

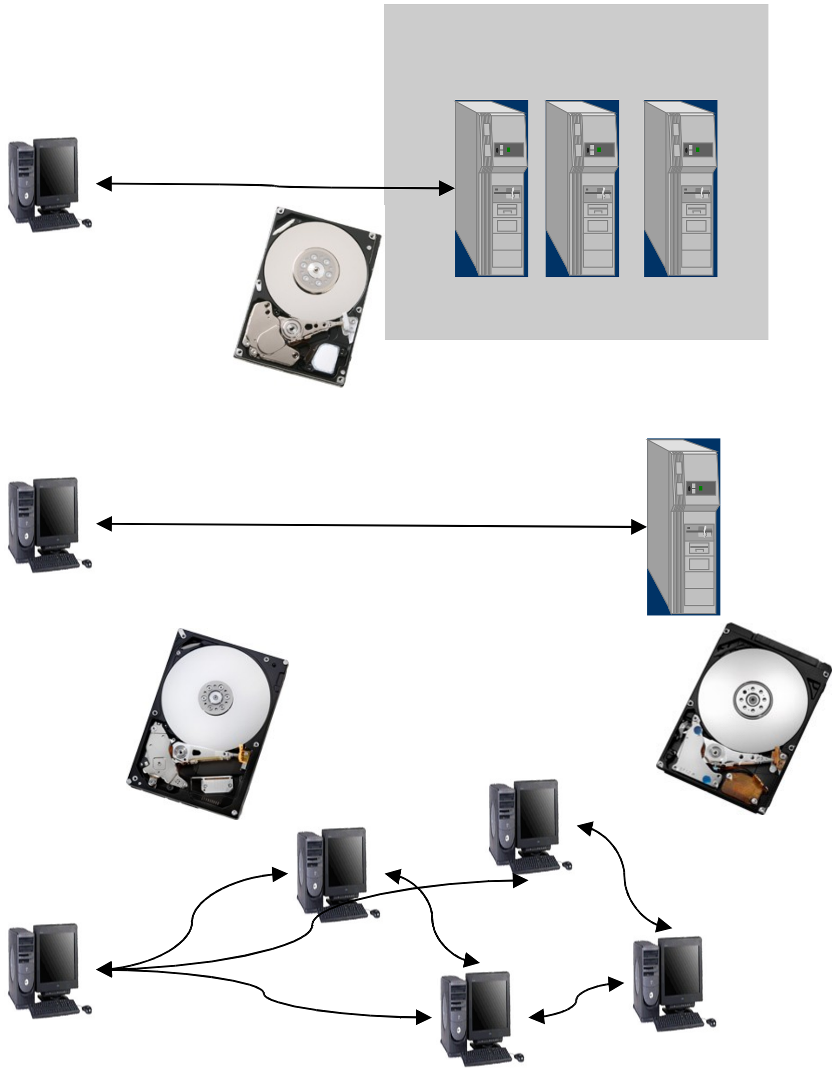
- **Standard definition**
  - Distributed storage across an unknown/ untrusted blob of communication
  - Key properties:
    - Accessed via the Internet
    - Simple to use
      - Reuses existing UIs
- **Why?**
  - Ease of access
  - Safety of data
    - Data redundancy
    - Geographical redundancy



Typical architectures

- **Back-end storage**
  - Raid arrays in large datacenters
  - Smaller servers
  - Distributed P2P storage

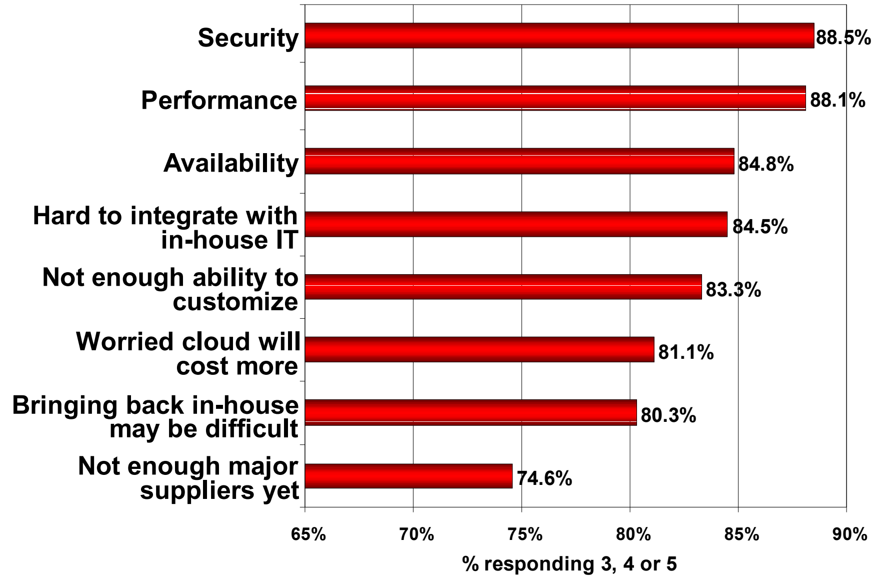
- **Type of storage devices**
  - Enterprise class drives for large datacenters
  - Nearline SATA drives for smaller server farms
  - Standard laptop/desktop drives for P2P



Cloud: does security matter?

HITACHI
Inspire the Next

- **According to responders to a 2008 IDC report, security is the most significant challenge facing Cloud applications (computing/storage)!**



- **Legal requirements for storage security are becoming more and more common!**

- **But multiple roles and varying topologies render the security threats more difficult to analyze...**

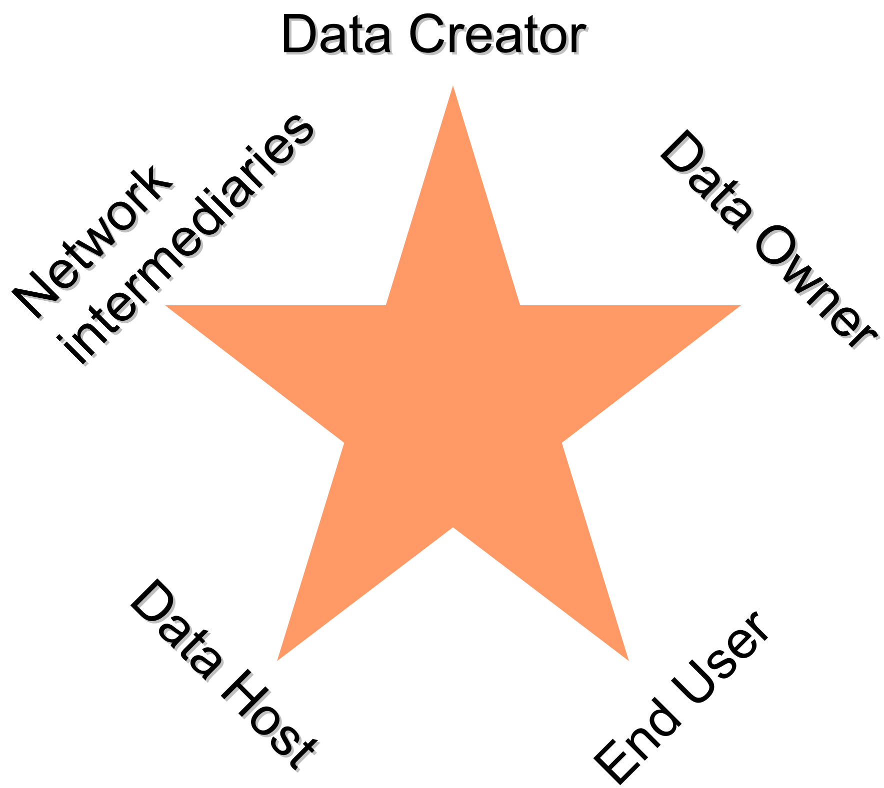
Cloud security requirements

- **Legal requirements**
  - Health care
    - HIPAA requires data encryption when data flows across open networks
      - Cloud is a perfect example of open network
    - Unauthorized disclosure of patient information carries high penalties
      - Up to $250,000, 10 years in prison...
  - Banking
    - Basel II requires that sensitive data transiting over public networks be encrypted

- **Market requirements**
  - Private data
    - Emails
    - Pictures
    - Documents
      - Collaborative editing
  - Fast erasure/repurposing

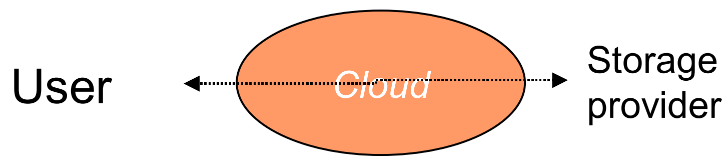
Cloud storage: roles

- **Data creator**
  - Typically data owner unless copyrights relinquished to another owner
- **Data owner**
- **End user**
  - Entity who uses the cloud to gain access to the data
- **Data host**
  - Entity member of the cloud who stores the data.
- **Network intermediaries**
  - Multiplicity of entities between all the previously described ones
    - Typically have no access to the data itself

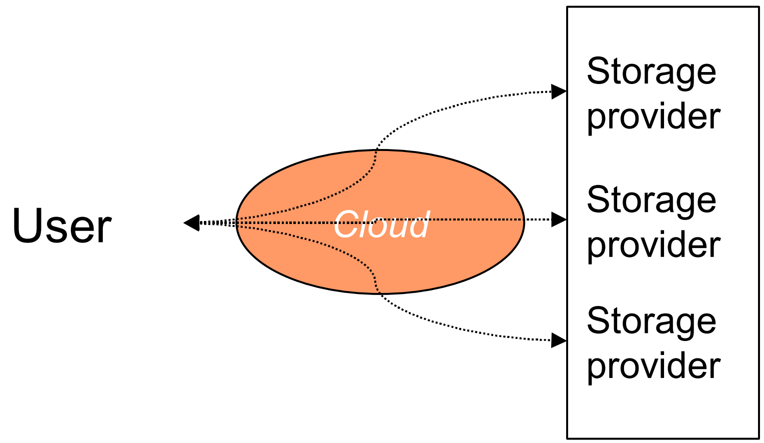


Cloud security: multiple storage topologies

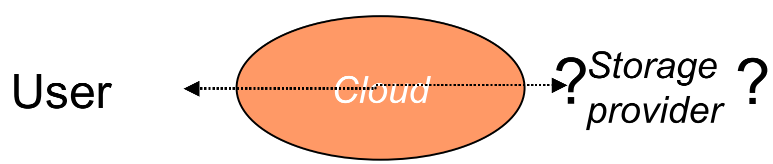
- **User to single provider**



- **User to multiple providers**



- **User to untrusted providers**



Securing the cloud: network vs. storage security

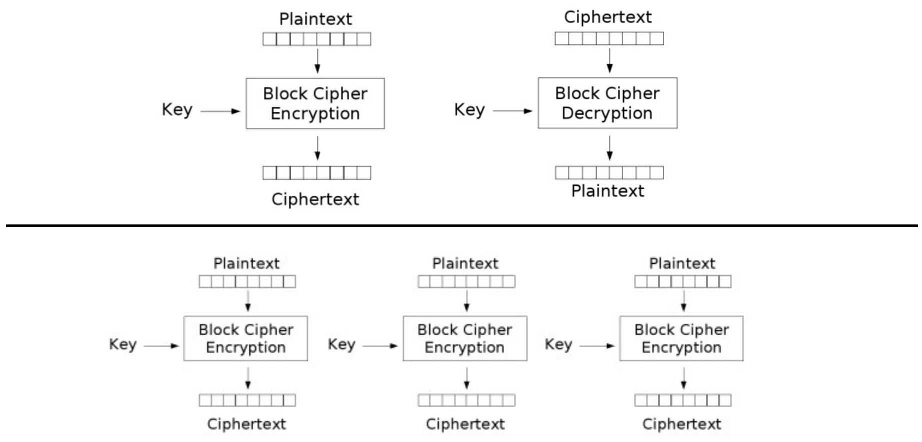
HITACHI
Inspire the Next

- **Network**
- **Threat model / Use cases**
  - Eavesdropper

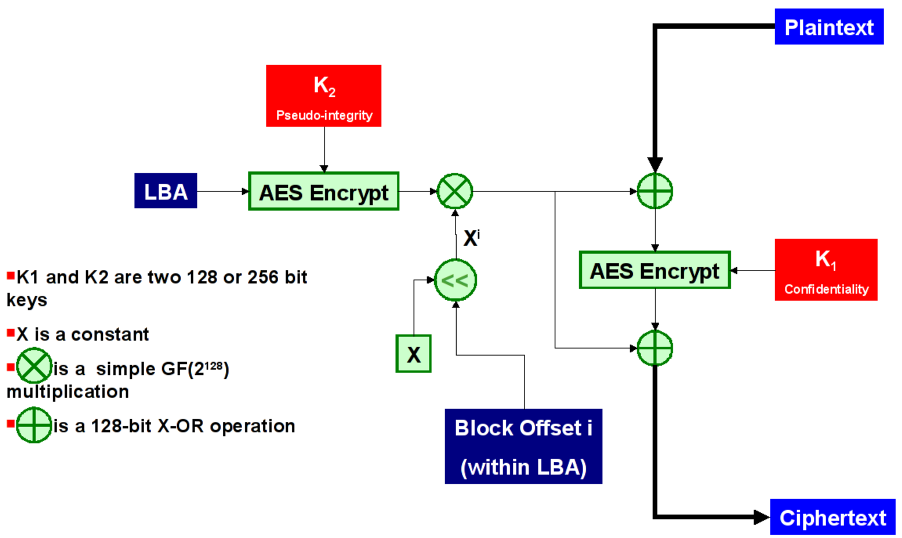
- **Storage**
- **Threat model / Use cases**
  - Stolen storage device/drive
  - Secure disposal of storage devices
  - Fast re-purposing of storage devices

Storage security: main principles

- **Block ciphers:**
  - Pseudo-random permutations defined for small block sizes (128 bits for AES)
- **How to encrypt more than one block?**
  - Electronic Code Book mode (ECB): blocks are encrypted independently
  - **Tweakable modes (XTS, LRW): blocks are encrypted with non-guessable position specific information**
- **Security criteria**
  - Confidentiality
    - Ability to hide plaintext information
  - Pseudo-integrity
    - Ability to detect modified ciphertext, except rollbacks to a previously valid state
    - Full integrity is not achievable for non-expanding modes...



Electronic Codebook (ECB) mode encryption



Cloud storage security

- **In what way is securing data in the cloud different from securing traditional data?**
  - Not that different!
    - Data needs to be stored protected
      - Confidentiality and integrity matter
    - Access control needs to be enforced

    Standard architectures developed for storage security can be used in the cloud:
      - **TCG Opal**
  - Differences:
    - Path to storage in the cloud is untrusted
      - Security of data in transit needs to be considered
    - Entity in charge of storing data in the cloud is typically different from the entity who owns the data

    Searching/data mining of the encrypted data might typically not be achievable...

Existing solutions for Cloud Storage security

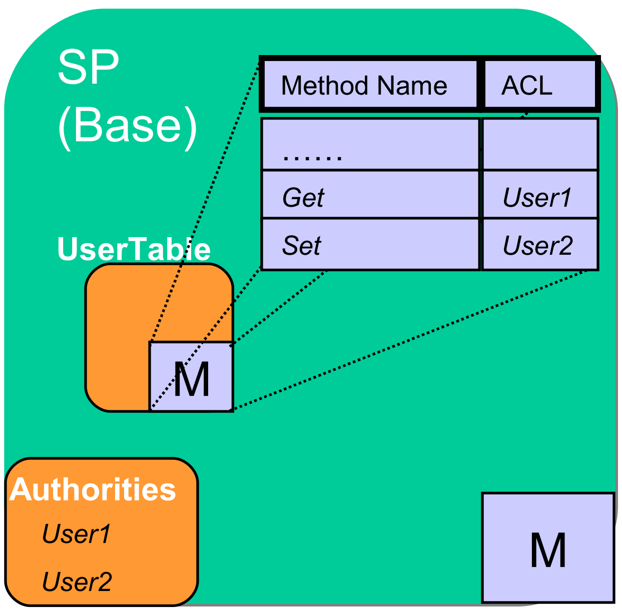
Hitachi Global Storage Technologies

TCG Storage Workgroup

HITACHI
Inspire the Next

Storage Work Group specifications are intended to provide a comprehensive command architecture for putting selected features of storage devices under policy-driven access control.

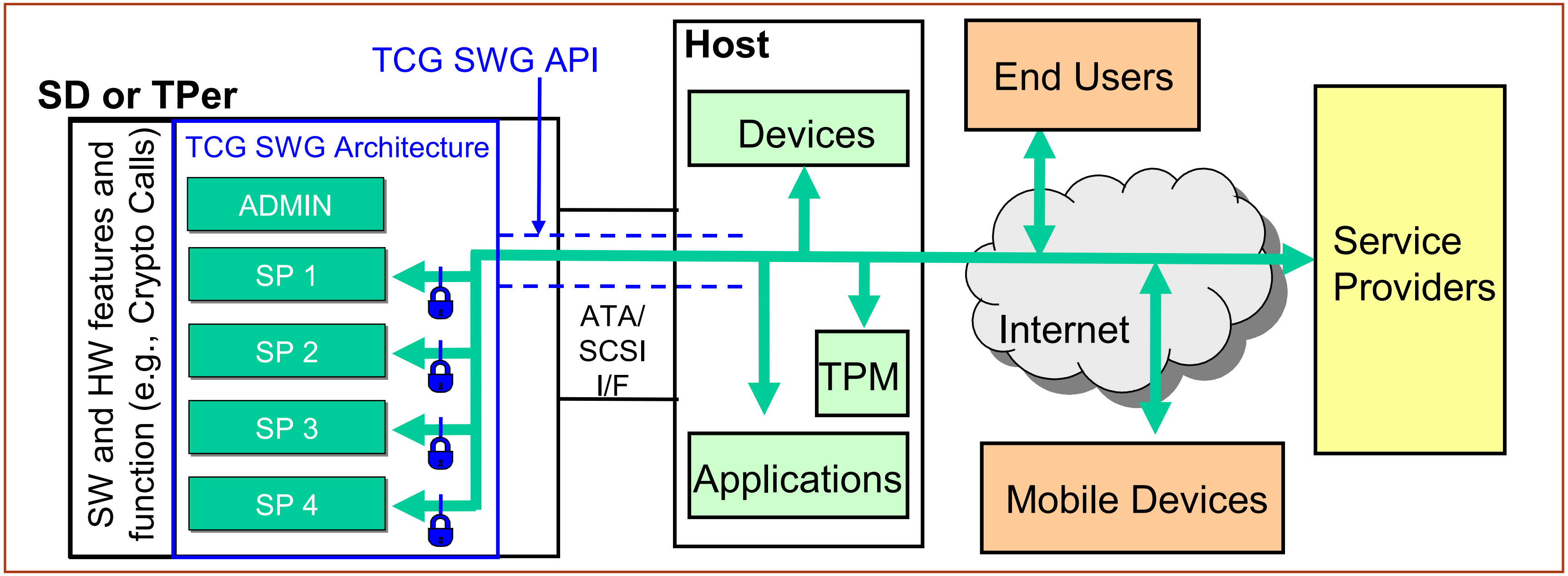
- Features will be packaged into individual functionality containers called "Security Providers" or SPs.



- Each SP is a "sand box" exclusively controlled by its owner. SP functionality is a combination of pre-defined functionality sets called SP Templates
  - Base
  - Admin
  - Crypto
  - Log
  - Clock
  - Locking
- SPs are a collection of tables and methods that control the persistent trust state of the Storage Device (SD).
  - Method invocation occurs under access control.
  - The SP has a list of authorities and their respective credentials for access control.

TCG Storage Work Group Solution Overview

HITACHI
Inspire the Next

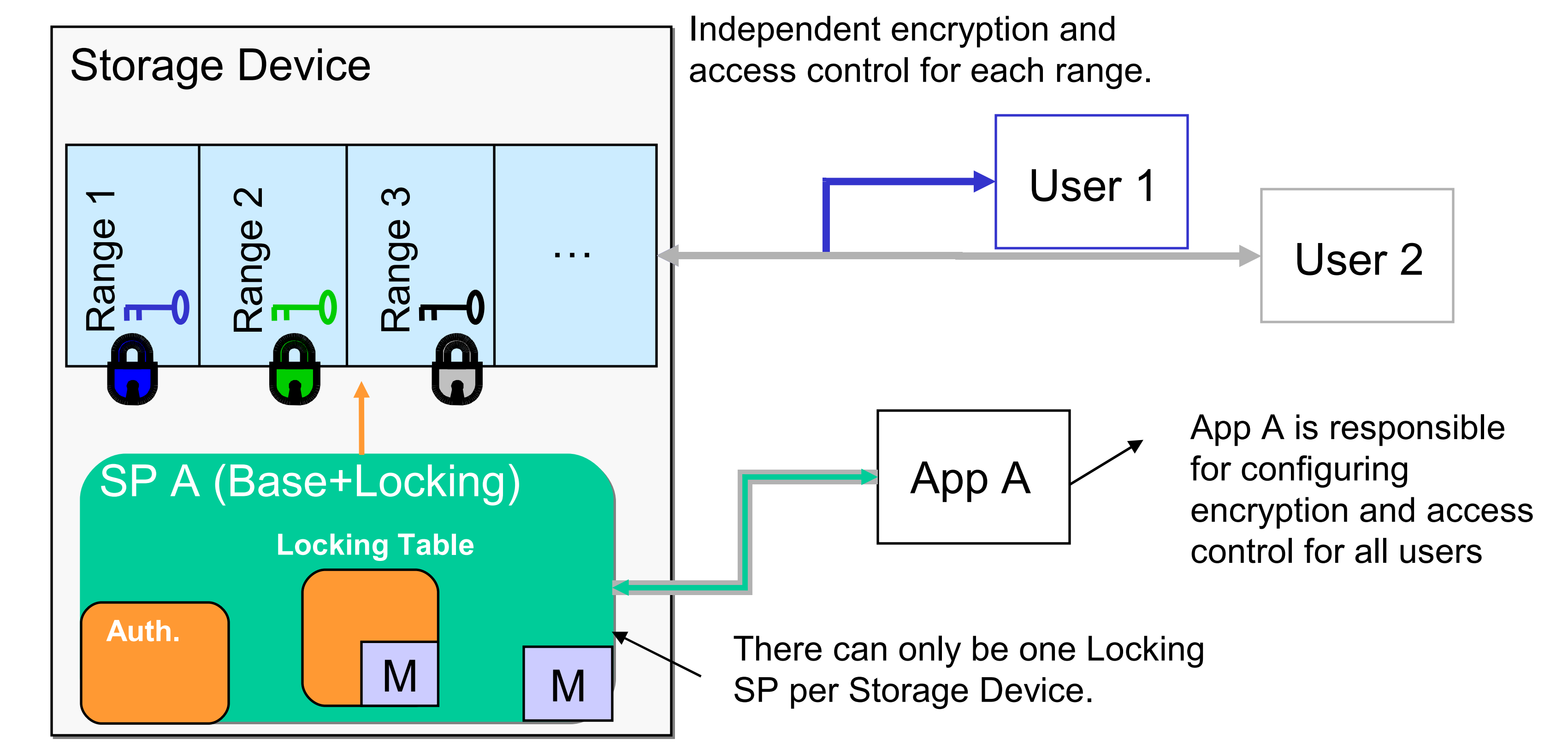


- The host platform, applications, devices, local end users, and remote users/ service providers can gain exclusive control of selected features of the storage device. This allows them to simultaneously and independently extend their trust boundary into the storage device or trusted peripheral (TPer)

Opal SSC Components – Locking and data encryption

HITACHI
Inspire the Next

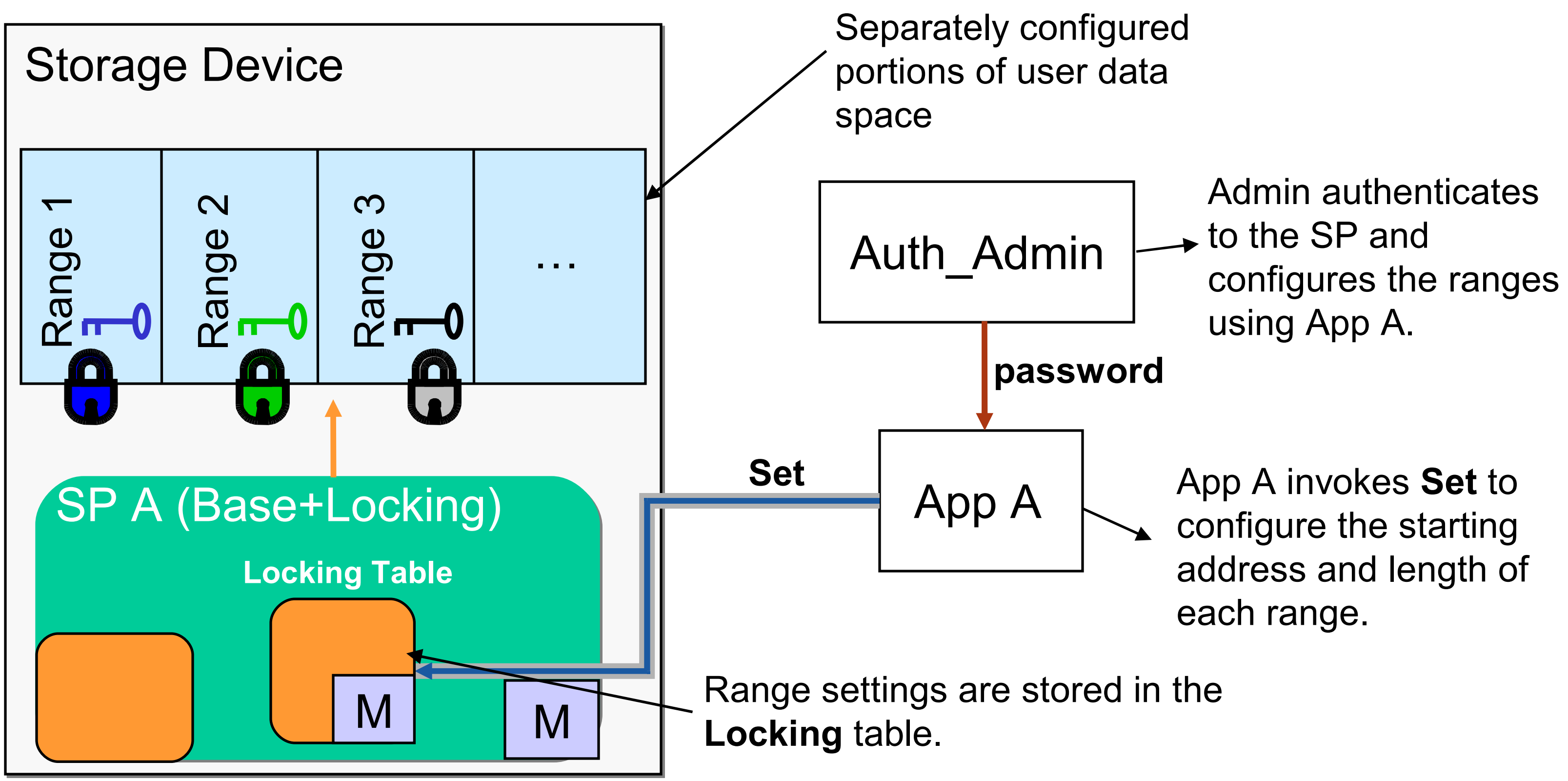
The storage device can have only one SP with Locking capability. When it is present, the storage device will be able to encrypt all the user data. Furthermore, access control to user data can be configured. The storage device will support a certain number of independent ranges.



Configuring Locking Ranges

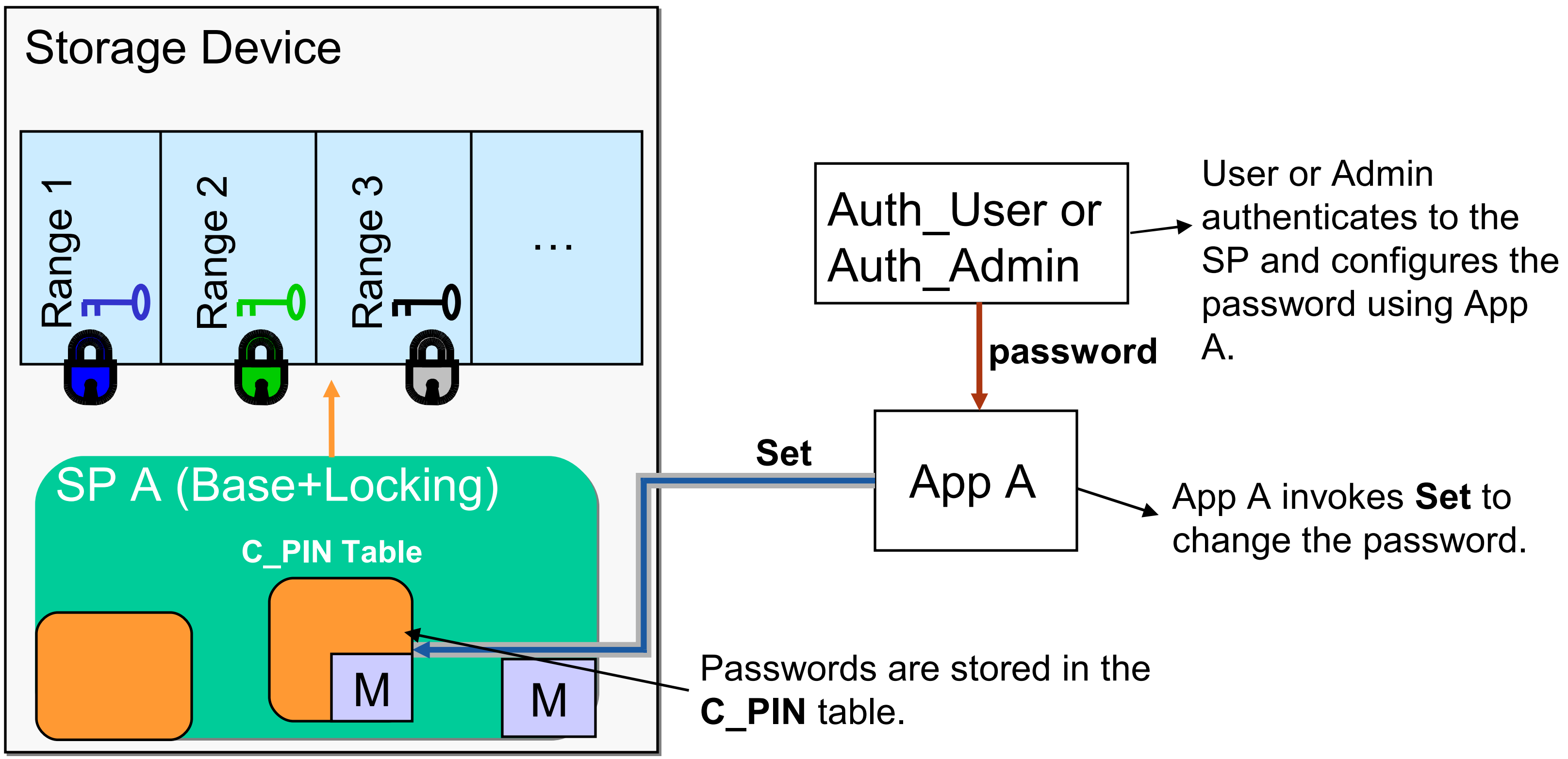
HITACHI
Inspire the Next

The Locking-enabled SP enables independent ranges of the user data space to be separately configured for read/write access control by an authorized and authenticated Admin.



Configuring User Passwords

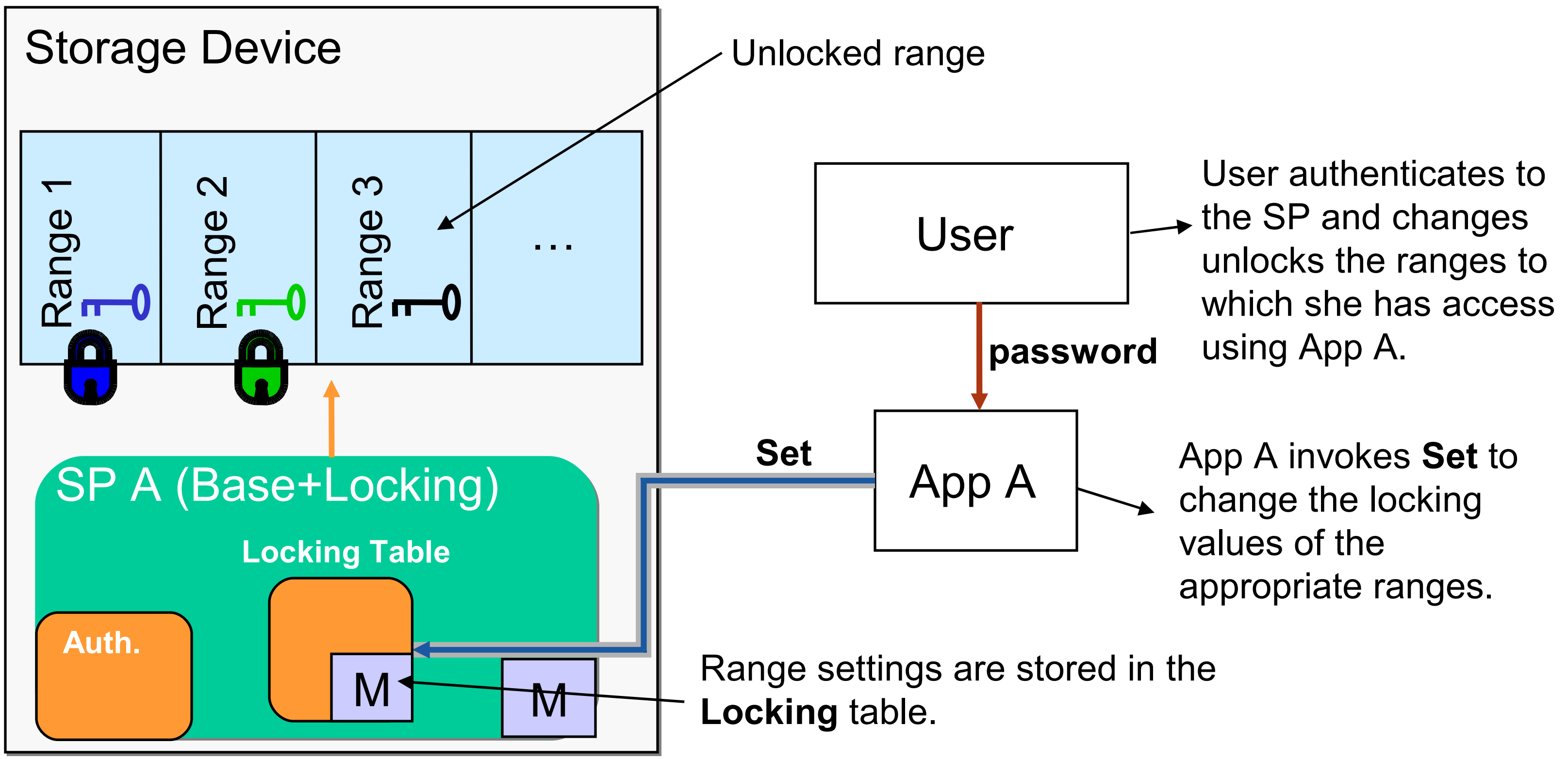
Each system user is assigned a separate password that is used for authentication to the Locking SP. Passwords can be set by the user of the password, or by the Admin.



Unlocking Ranges

HITACHI
Inspire the Next

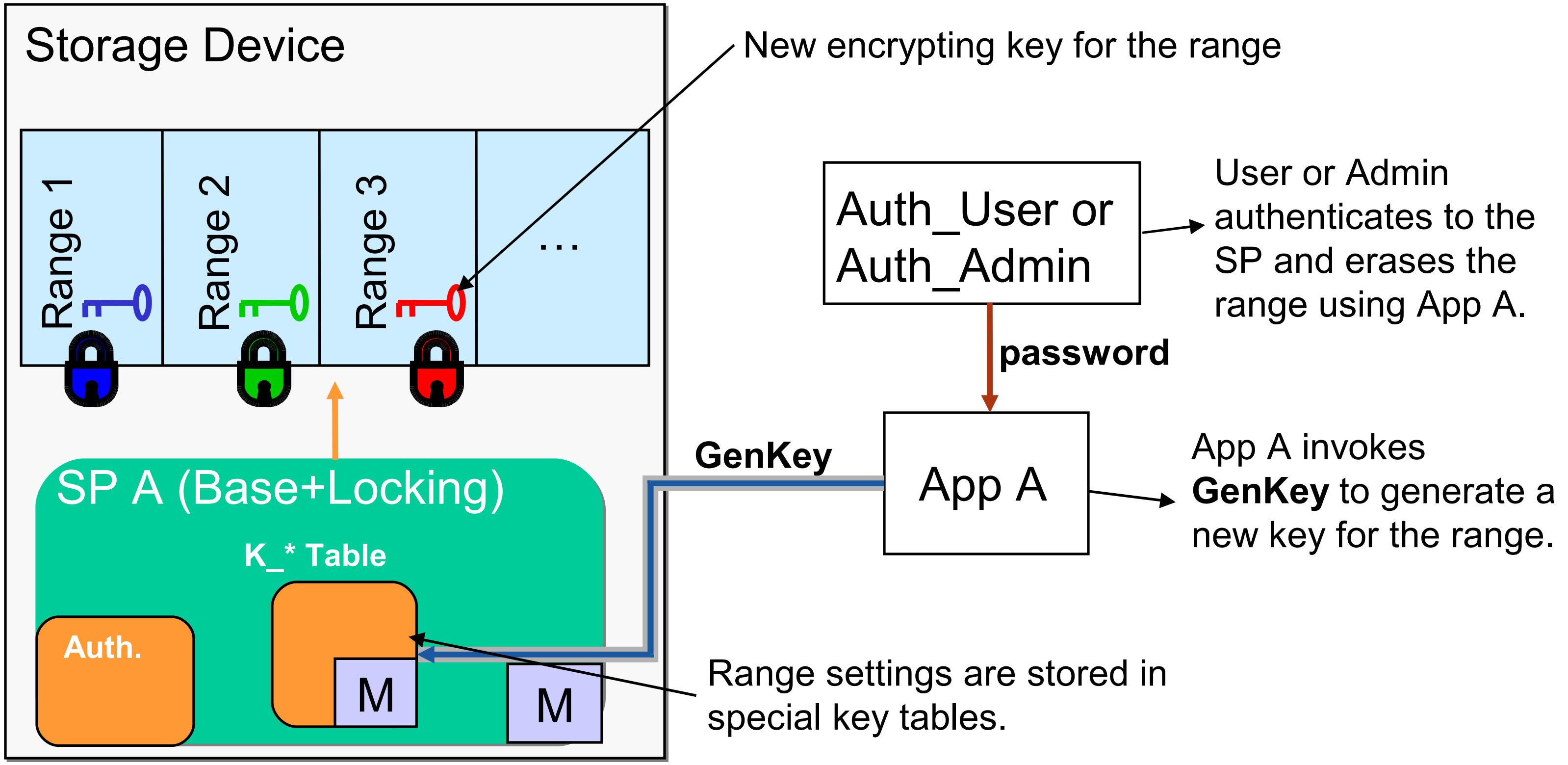
The authorized user authenticates with his password and then unlocks the ranges to which she has access.



Secure Erase

HITACHI Inspire the Next

The Locking-enabled SP provides the admin and users with the ability to securely erase data, securely and quickly, by replacing the encryption key for a range with a new key randomly generated securely in the drive.



TCG Specifications

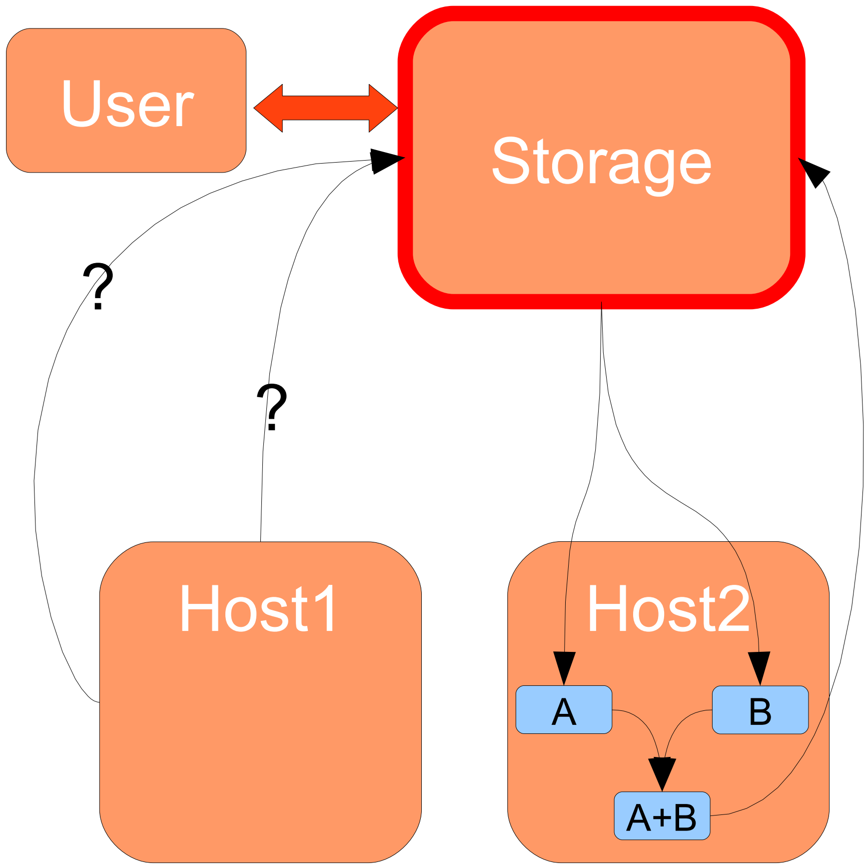
- **Specifications published a week ago!**
- **Available at:**
  - https://www.trustedcomputinggroup.org/specs/Storage/
    - PC Client (Desktop/Laptop drives):
      - https://www.trustedcomputinggroup.org/specs/Storage/Opal_SSC_1.0_rev1.0-Final.pdf
    - Traditional Enterprise (Fiber channel drives):
      - https://www.trustedcomputinggroup.org/specs/Storage/TCG_SWG_SSC_Enterprise-v1r1-090120.pdf

Future challenges in Cloud Storage security

Hitachi Global Storage Technologies

Multiple challenges

- **Data hosts and data users are typically separate entities with different views on how the data should be used...**
- **Many cloud storage providers have business models that rely on information about the stored data**
  - Advertising!
  - Storage optimization
  - ...
- **Fundamentally, secure storage cryptographic algorithms are designed to prevent "adversaries" from gaining knowledge about stored data**
  - "Semantic security"
- **So solutions to those challenges will require relaxing requirements**



First Challenge: Search applications

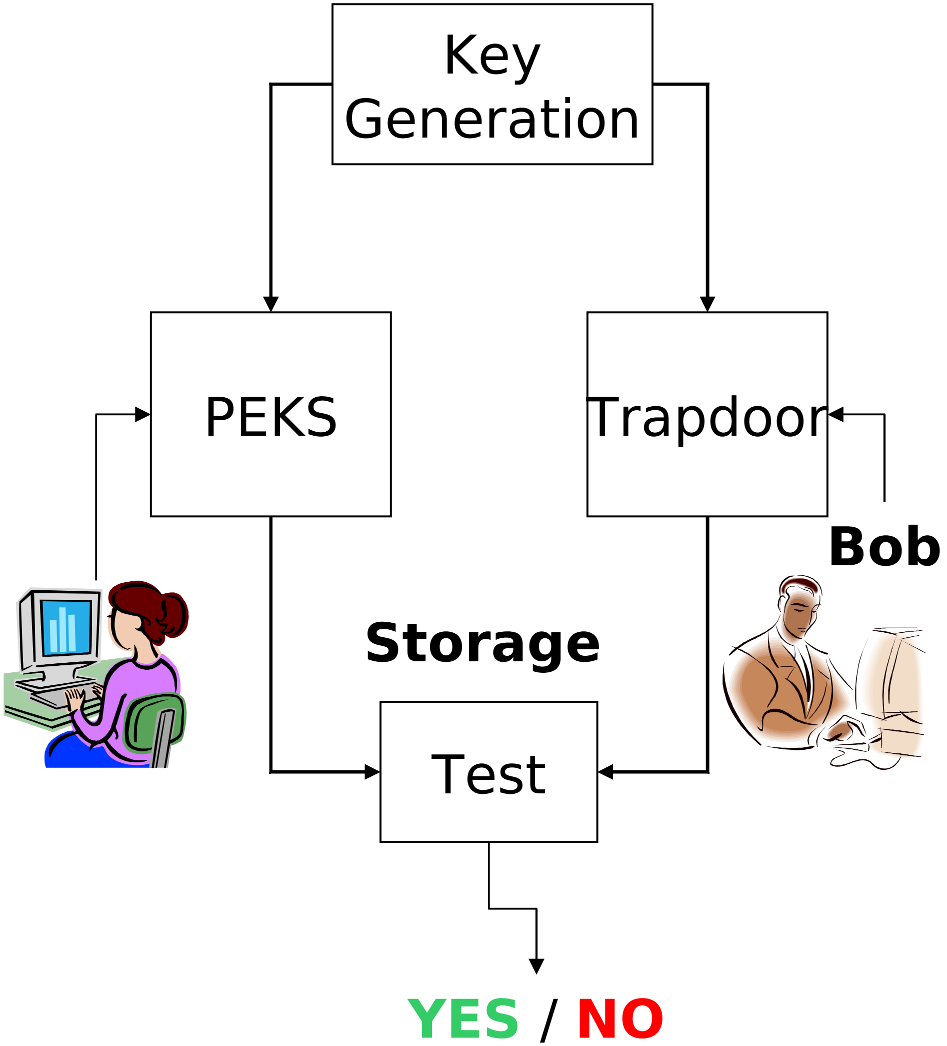
- **Motivation (communication paradigm)**
  - Bob wants to send an encrypted email to Alice
  - Alice's server/gateway wants to test for the presence of some keywords to determine how to route the email properly (urgent, mailing list...)
  - But Alice does not want the server/gateway to be able to decrypt her messages

  ➔ Asymmetric algorithms
- **In the cloud storage case, Alice and Bob might be the same person...**

  ➔ Symmetric algorithms
- **Goal:**
  - Allow a third party to test for presence of specific keywords
- **Cryptographic solutions**
  - Searchable encryption!
- **Public Key Encryption with Keyword Search (PEKS)**
  - Introduced by Boneh, Di Crescenzo, Ostrovsky and Persiano[BDOP04]
- **Searchable Symmetric Encryption (SSE)**
  - Applicable directly to storage
  - Studied by [SWP00], [Goh03], [CM05], [CGKO]

PEKS: definitions

- **A non-interactive public-key encryption with keyword search scheme consist of the following (polynomial time) algorithms:**
  - Key Generation(s)
    - Takes a security parameter s and generates a pub/priv key pair SK/PK
  - PEKS(PK, W)
    - Takes a public key PK and a word W and generates a searchable encryption of W
  - Trapdoor(SK, W)
    - Takes a private key SK and a word W, produces a trapdoor T_w
  - Test(PK, S, T_w)
    - Given a public key PK, a searchable encryption S and a trapdoor T_w, outputs whether W=W'



PEKS: research results

- **Security (IND-CPA)**
  - The ciphertext should not reveal any information about the encrypted keyword
  - The trapdoor should only allow the trapdoor entity to know whether the specific keyword is inside the ciphertext
- **Cryptographic result:**
  - A non-interactive searchable encryption scheme that is semantically secure against an adaptive chosen keyword attack gives rise to a chose ciphertext secure Identity Based Encryption scheme
  - Or in clearer terms, secure PEKS construction are at least as hard as IBE constructions!

PEKS: actual construction

- **Multiple constructions exist**
  - Generic ones – without Random Oracle assumptions – are rather inefficient
  - A fairly efficient one, assuming the RO, and based on a slightly modified Decision Diffie Hellman assumption for bilinear maps
    - Given g^a, g^b then g^{ab} "looks like" a random element of the group
- **Bilinear map**
  - Let G_1, G_2 be two groups
  - A map $G_1 x G_1 \text{->} G_2$ is a bilinear map if
    - It is efficiently computable
    - It is bilinear:
      - $e(g^x, g^y)=e(g,g)^{xy}$
    - And it is non-degenerate
      - $e(g,g)$ generates G_2
- **Let H_1 and H_2 be two hash functions:**
  - H_1: {0,1}* -> G_1 and H_2: G_2 -> $\{0,1\}^{\log p}$
- **Then**
  - KeyGeneration
    - Pick a random value a in Z_p^* and a generator g of G1
      - Then PK=[g, $h=g^a$] and SK=a
  - PEKS
    - Compute $t=e(H_1(w), h^r)$ for a random r in Z_p^*
      - Then output [g^r, $H_2(t)$]
  - Trapdoor
    - Compute $T_w=H_1(w)^a$
  - Test
    - If S=[A,B], test if $H_2(e(T_w, A))=B$
      - If so output "yes" otherwise "no"
        - Indeed: $H_2(e(H_1(w)^a, g^r)) = H_2(e(H_1(w),g)^{ar}) = H_2(e(H_1(w),h^r))$ QED

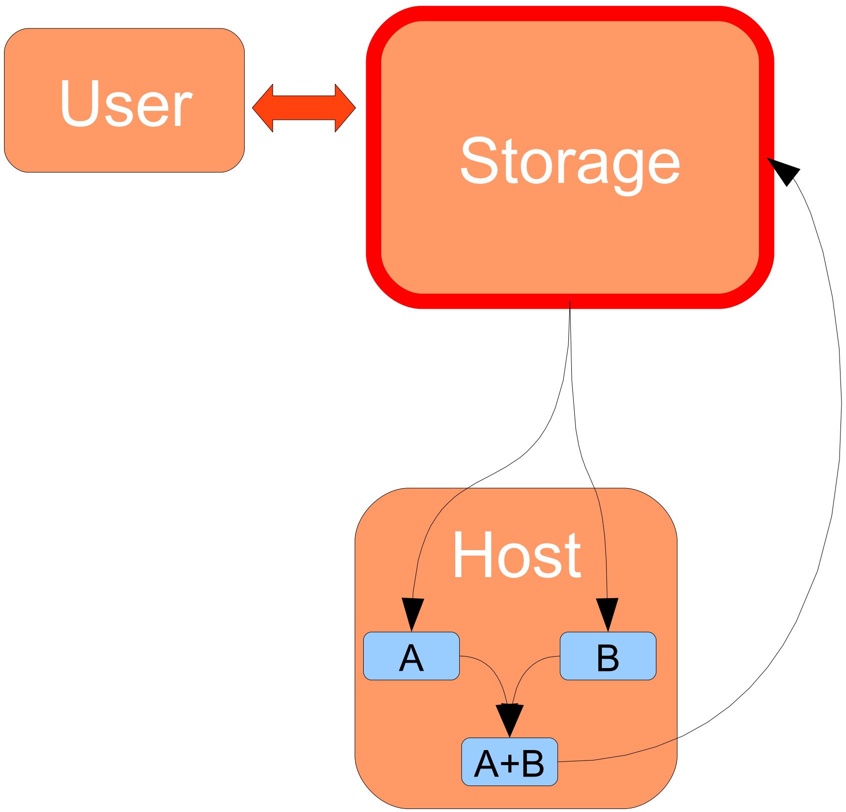
Searchable Symmetric Encryption

- **PEKS constructions are slow**
  - Public Key algorithms tend to be slow in general
- **Use Searchable Symmetric Encryption instead!**
  - For any encrypted collection of words stored in the clouds, an additional data structure is stored with it
  - The server can use this data structure to answer the query
    - Is this word W in the encrypted data?
- **Multiple constructions exist**
  - [Goh03], [CM04]
  - Difficulty lies in defining the capability of the adversary – the server –
    - Can it recover data? No
    - Can it search? Not so obvious...
    - ...

- **Keygen(1^k): outputs symmetric key K**
- **BuildIndex(K, {D_1, ..., D_n}): outputs secure index I**
- **Trapdoor(K, w): outputs a trapdoor T_w**
- **Search(I, T_w): outputs identifiers of documents containing w (id_1, ..., id_m)**

Second challenge: Calculating on encrypted data

- **Practical problem**
  - Storage hosts might want to modify stored data on behalf of the user/owner of the data
    - Incrementing values (Date, Counter)
    - Simple arithmetic
- **Cryptographic problem**
  - Any sufficiently secure encryption scheme prevents meaningful modifications to the ciphertext
  - How to relax those requirements to obtain a scheme that allows calculation/ modifications on ciphertext, while still keeping a sufficient security level for most applications



What type of computations?

- **Can we design an encryption scheme that allows any function f(x,y) to be calculated on the plaintext?**
  - Hard problem
    - Problem for generic binary operators is actually proven to be impossible
- **So let's try restricting that to simpler functions**
  - Can we find cryptosystems that allow "group" operations to be calculated on the plaintext by only acting on the ciphertext
- **Answer: Yes...**
- **Homomorphic encryption!**
- **Definition**
  - An encryption algorithm with the following property:
    - $E(A \times_1 B) = E(A) \times_2 E(B)$
- **Why is that useful**
  - It allows the user to calculate the inner operation by calculating the outer operation on the ciphertext!

Example of group-homomorphic encryption

HITACHI
Inspire the Next

- **RSA**
  - Indeed since $E(x_1)$. $E(x_2) = (x_1^e \bmod n)$. $(x_2^e \bmod n) = (x_1 . x_2)^e \bmod n = E(x_1 . x_2)$
    - Note: typical RSA padding breaks this property… And RSA encryption without padding is badly insecure, so care must be taken when designing a homomorphic encryption scheme with this property…
- **El Gamal encryption on any cyclic group**
  - $E(x_1) . E(x_2) = (g.r_1, x_1.g^{xr_1}) . (g.r_2, x_2.g^{xr_2}) = (g.(r_1+r_2), x_1.x_2.g^{x.(r_1+r_2)}) = E(x_1 . x_2 \bmod p)$
  - and many more like Goldwasser-Micali for GF(2) addition, Paillier cryptosystem for modular addition
- **Group homomorphisms are very interesting theoretically, but fairly limited in practice since only one type of operation can be done on the plaintext.**
  - How can we extend the functionality they provide?
- **Fully (ring) homomorphic encryption?**
  - Encryption preserves 2 different operations!
    - E(a.b)=E(a).E(b)
    - E(a+b)=E(a)+E(b)
- **Open problem…**
  - Though a few recent results are getting closer...
    - Addition and one multiplication in [BGN05]
      - » a.b+c.d+e.f+...+y.z

HITACHI
Inspire the Next

Conclusions

Hitachi Global Storage Technologies

Conclusions

HITACHI
Inspire the Next

- **Securing the data in the cloud is necessary!**
  - Network security is typically not sufficient...
- **There exist mechanisms to do so**
  - Indeed, good cryptographic encryption modes exist, and the industry just standardized on an extensible architecture to control the security functionality (TCG)
- **Securing the cloud does bring out some challenges**
  - Given traditional properties of symmetric encryption, it seems impossible to search/index/calculate on encrypted data
- **But cryptographers are here to find solutions!**
  - Searchable encryption
  - Homomorphic encryption
  - and more:
    - Private Information Retrieval
    - Multi-Party computations

HITACHI
Inspire the Next

Questions?

Hitachi Global Storage Technologies

Backup slides

Fully (ring) homomorphic encryption?

- **Current best result from [BGN05] but only provides evaluation of degree 2 multivariate polynomials, for some subset of all possible values:**
  - Choose two finite cyclic groups (G, *) and (G_1, *) such that
    - g is a generator of G
    - The order of G and G_1 is q1.q2 for two primes q_1 and q_2
    - There exists a bilinear map from GxG to G_1
      - In other words, there exists e: GxG -> G_1 such that $e(a^n, b^m) = e(a, b)^{nm}$ for all a, b in G and n, m in Z
    - Moreover, e(g, g) is a generator of G_1
    - Finally, e(x, y) needs to be computable in polynomial time of the inputs and parameters
  - Key generation is the following:
    - Choose two random generators g, u of G and let $h = u^{q_2}$. (Then h generates the subgroup of order q_1)
    - Let the private key be q_1 and the public key be (n, G, G_1, e, g, h)
  - Encryption is done as follows
    - Assume the plaintext message m is a bit – can easily be extended to any integer in {0..T} for $T<q_2$ –
    - Pick a random r less in {0…n-1} and calculate the following
      - $C = g^m.h^r$
  - Decryption is the following
    - Calculate $C^{q_1} = (g^m.h^r)^{q_1} = (g^{q_1})^m$
    - Finding m is only a matter of calculating the discrete log of C^{q_1} for the base g^{q_1}
      - Since m is a bit or something small, discrete log is easy…

[BGN05] cryptosystem (cont.)

- **We still need to show that this allows one to evaluate degree 2 polynomials**
- **Clearly the scheme is homomorphic since anybody can calculate**
  - $C_1.C_2.h^r$ to generate a ciphertext of $m_1 + m_2$ mod n
- **But on top of that it is possible to multiply two ciphertexts in the following way:**
  - Let $g_1=e(g,g)$ and $h_1=e(g,h)$
    - $ord(g_1)=n$, $ord(h_1)=q_1$
  - Write $h=g^{a.q_2}$
  - Then calculate $e(C_1, C_2).h_1^r = e(g^{m_1}.h^{r_1}, g^{m_2}.h^{r_2}).h_1^r = g1^{m_1 m_2}.h1^{m_1 r_2 + r_1 m_2 + r_1 r_2 a q_2 + r}$
  - $=g_1^{m_1 m_2}.h_1 r'$ which is the encryption of $m_1.m_2$ mod n, *but in the group G_1*
  - Since we are now in G_1 we can not do this trick again. We are therefore limited to one "multiplication"
- **We can calculate "additions" – group multiplication actually – and multiply once – bilinear map actually – so we can compute 2nd degree multivariate polynomial expressions**
- **How do we find groups with such pairings?**
  - Typically as groups of points on supersingular elliptic curves defined on a finite field, together with either Weil or Tate pairings into F_{p2}
    - Why supersingular?
      - » Because then the number of points is easy to calculate: it is the same as the number of elements in the field