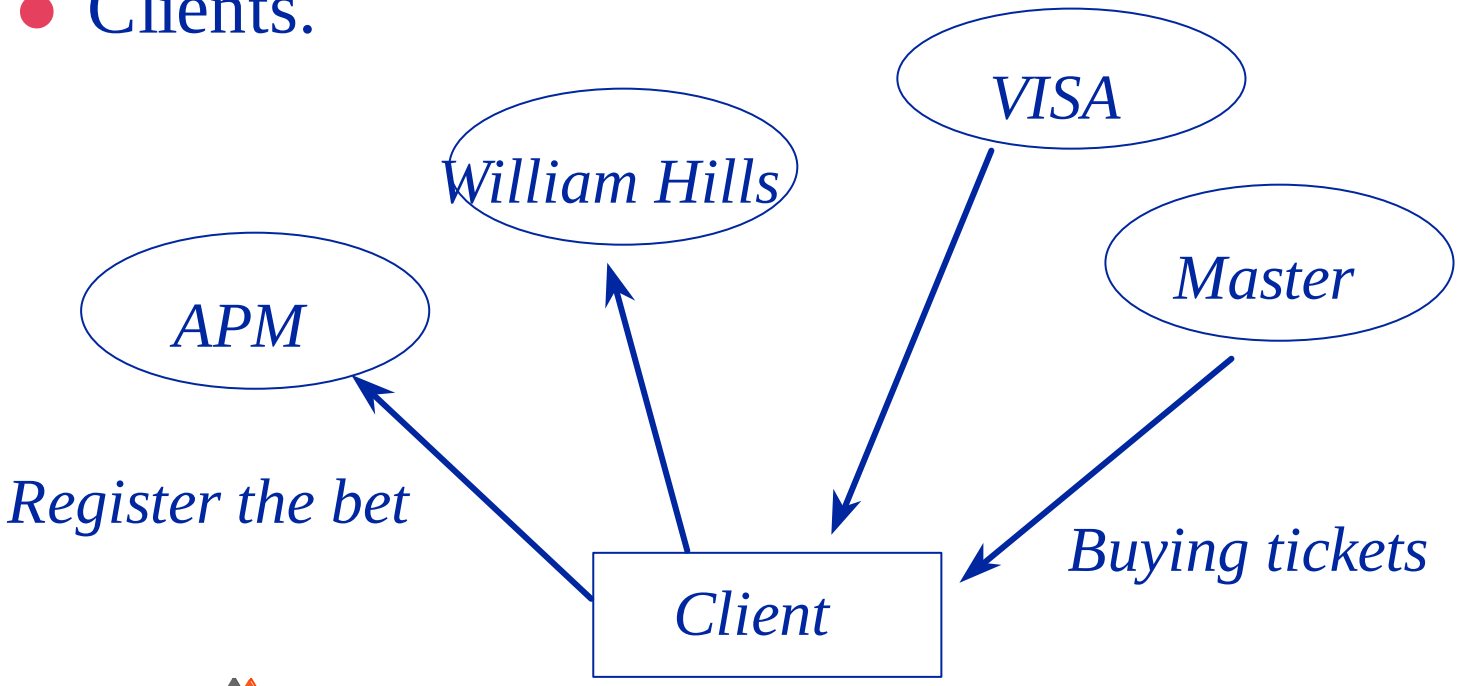
Mobile Object Security Implementation

21 July 1998

Takanori Ugai

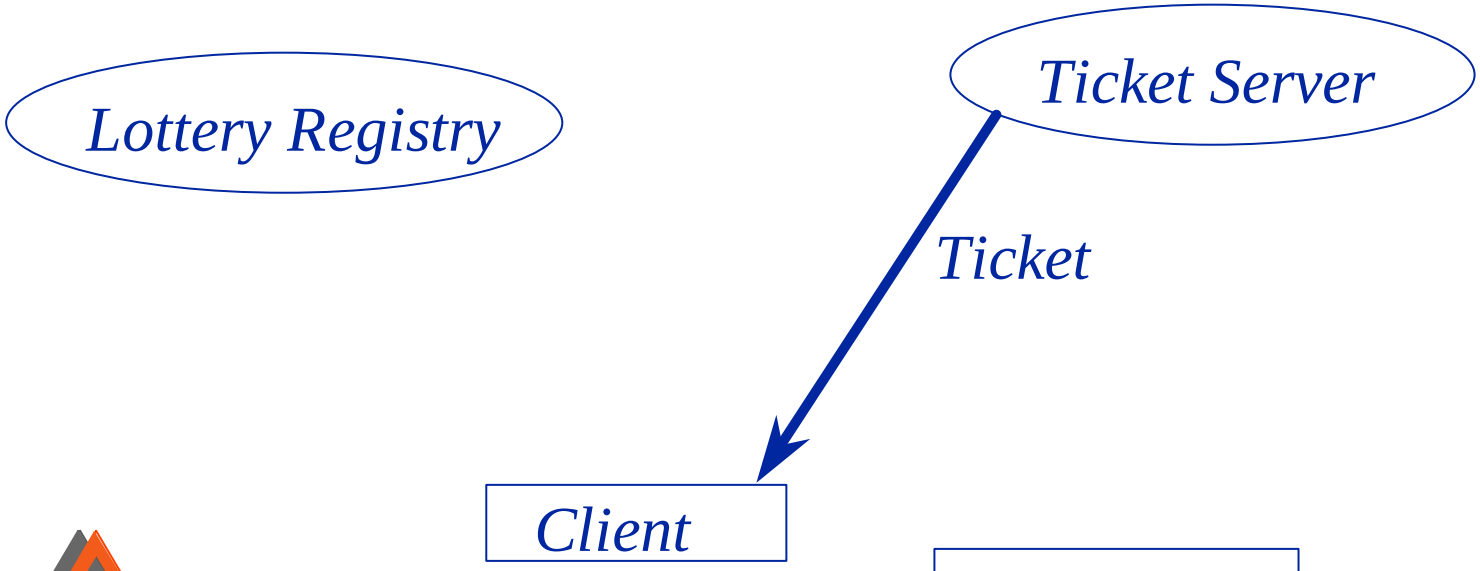
Lottery Example

- A couple of ticket vendors.
- Lots of lottery service sites.
- Clients.



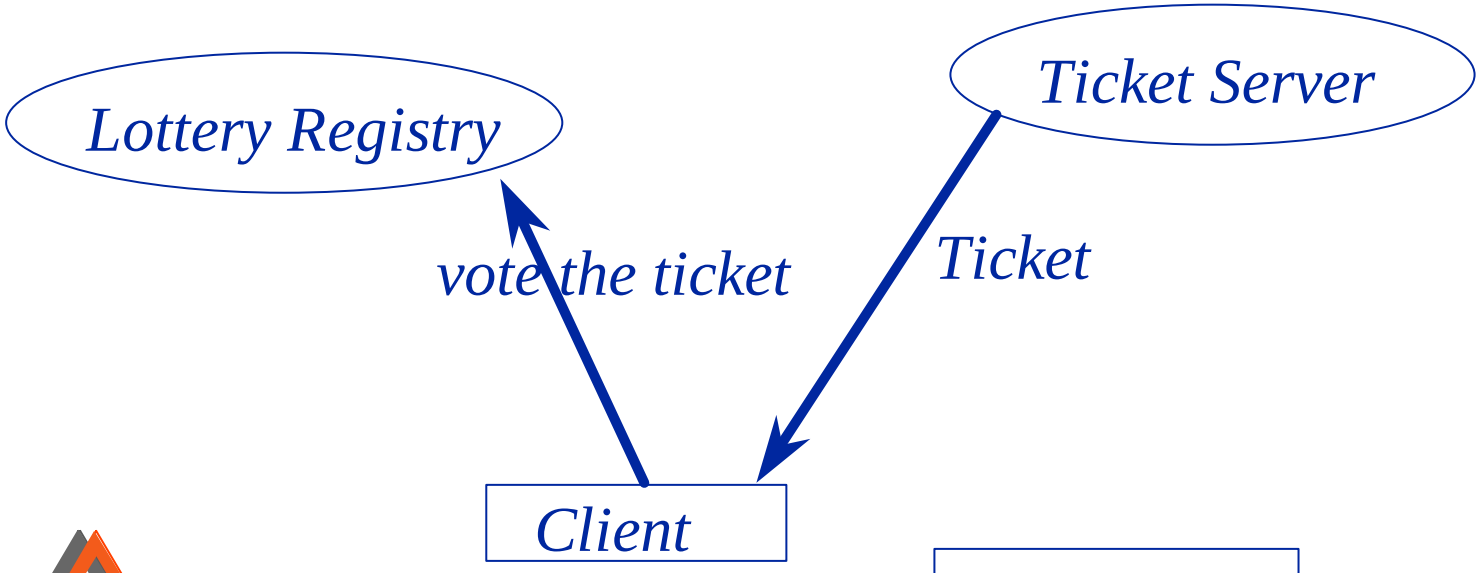
Ticket Server/Lottery Registry/Client

- Client(C) buys a ticket from a Ticket Server(TS).
- Client makes a choice and puts it on the ticket.



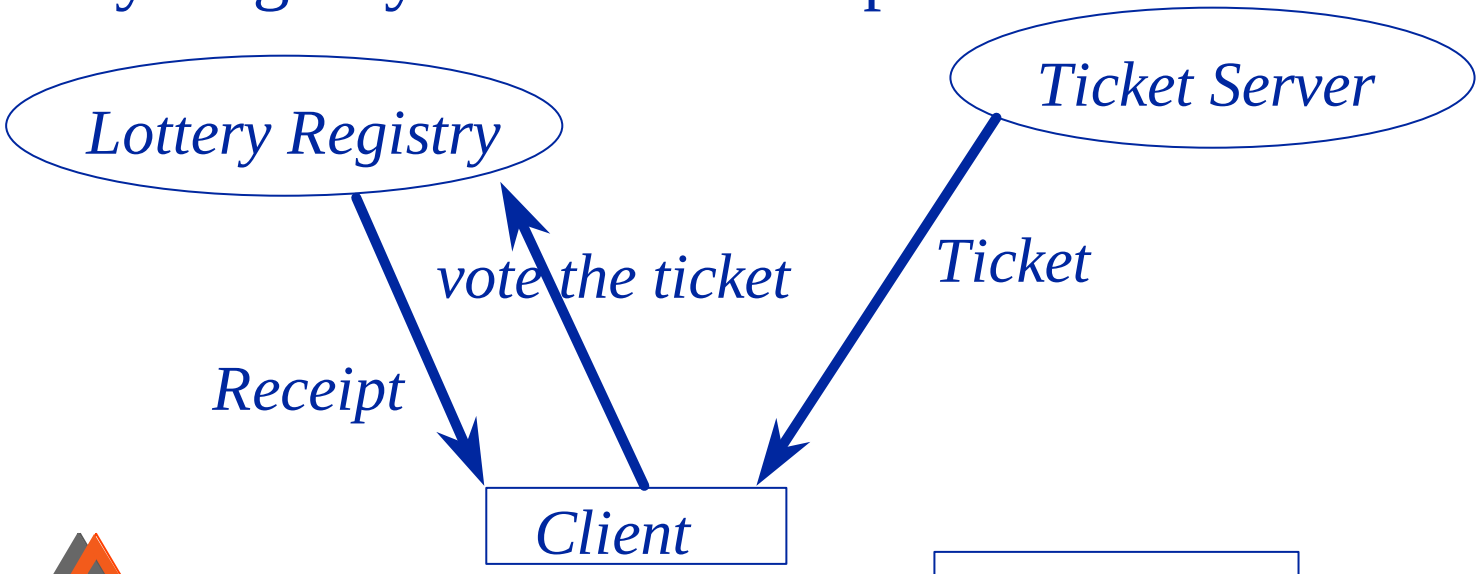
Ticket Server/Lottery Registry/Client

- Client(C) buys a ticket from a Ticket Server(TS).
- Client makes a choice and puts it on the ticket.
- Client sends the ticket to a Lottery Registry(LR).



Ticket Server/Lottery Registry/Client

- Client(C) buys a ticket from a Ticket Server(TS).
- Client makes a choice and puts it on the ticket.
- Client sends the ticket to a Lottery Registry(LR).
- Lottery Registry returns a receipt.



Ticket
Data Fields and Access Metrics

- Ticket Identifier(TID) TS:RW LR:R C :R
  - anonymous read key TS:RW LR:R C:R
  - anonymous write key TS:RW C:R
- Vote LR:R C:RW
- TID and Vote is counter signed by LR

R = Read

RW = Read and Write

What is implemented

- Buying a ticket from a ticket server.
- Anonymous voting to lottery registry

- Not implemented
  - payment for the ticket.
  - payment for the prize.

Secure and Insecure

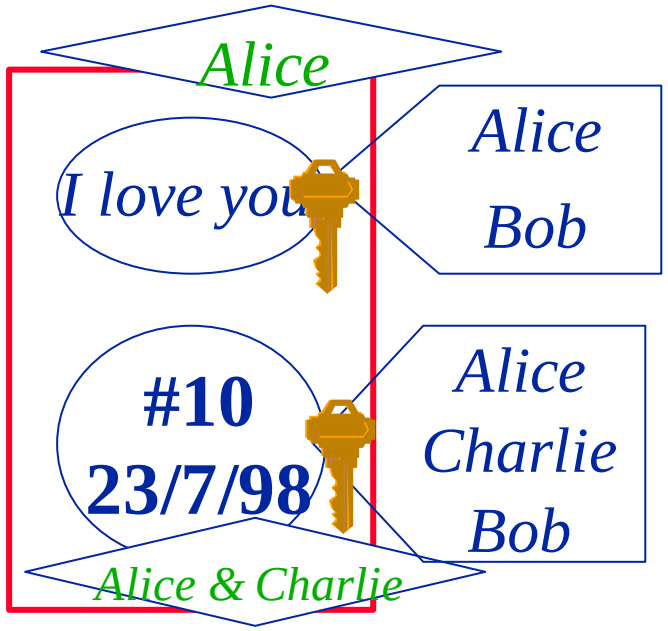
- Secure
  - Validity of the ticket.
  - Validity of the vote
  - TS knows who buys the ticket.
  - LR does not knows who is voter.
  - Ticket’s duplication is detected.
- Insecure
  - LR can cheat about the lottery result.
  - LR may not send back the receipt.

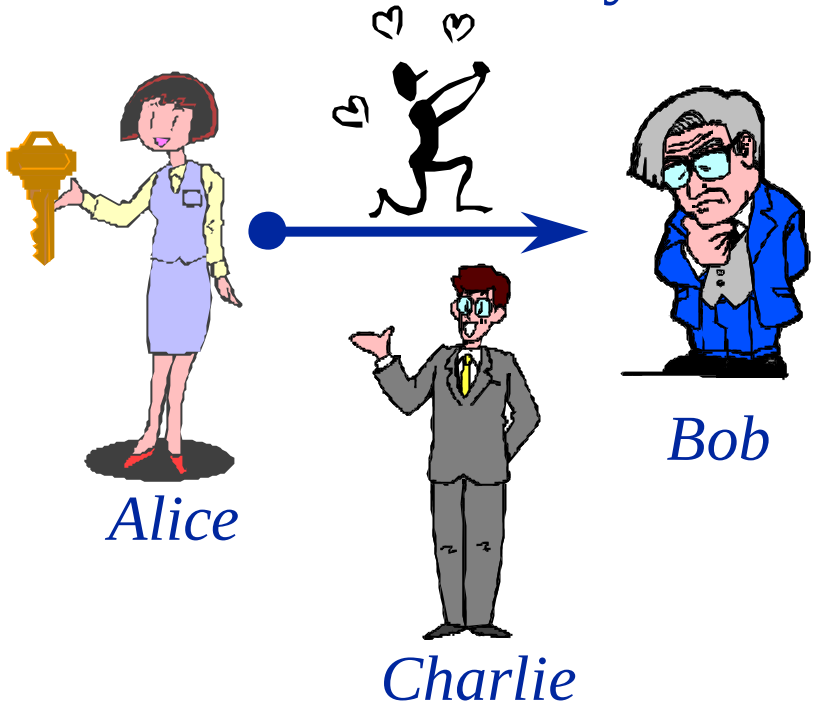
Secure Mobile Object

- **Carrying Secrets**
  - **protecting the data from hostile hosts**
- **Secure Communication between Mobile Objects**
  - **protecting communications and object migration from hostile third parties.**

Carrying/Passing Secret

- Data is encrypted by a secret key.
- Only allowed principals can use the secret key.





Passing Secret

- Alice asks Charlie to pass the secret to Bob.
  - Bob can read the sentence.
  - Bob can recognize the secret is made by Alice.
  - Bob will pay #10 if he receive the message before the expiry date.
  - Charlie cannot read the secret message.
  - Everyone understands that the price and expiry date are agreed by Alice and Charlie
- Charlie can
  - steal the secret and not pass it to Bob.
  - make a copy and try to get another #10.

Secure Object Implementation

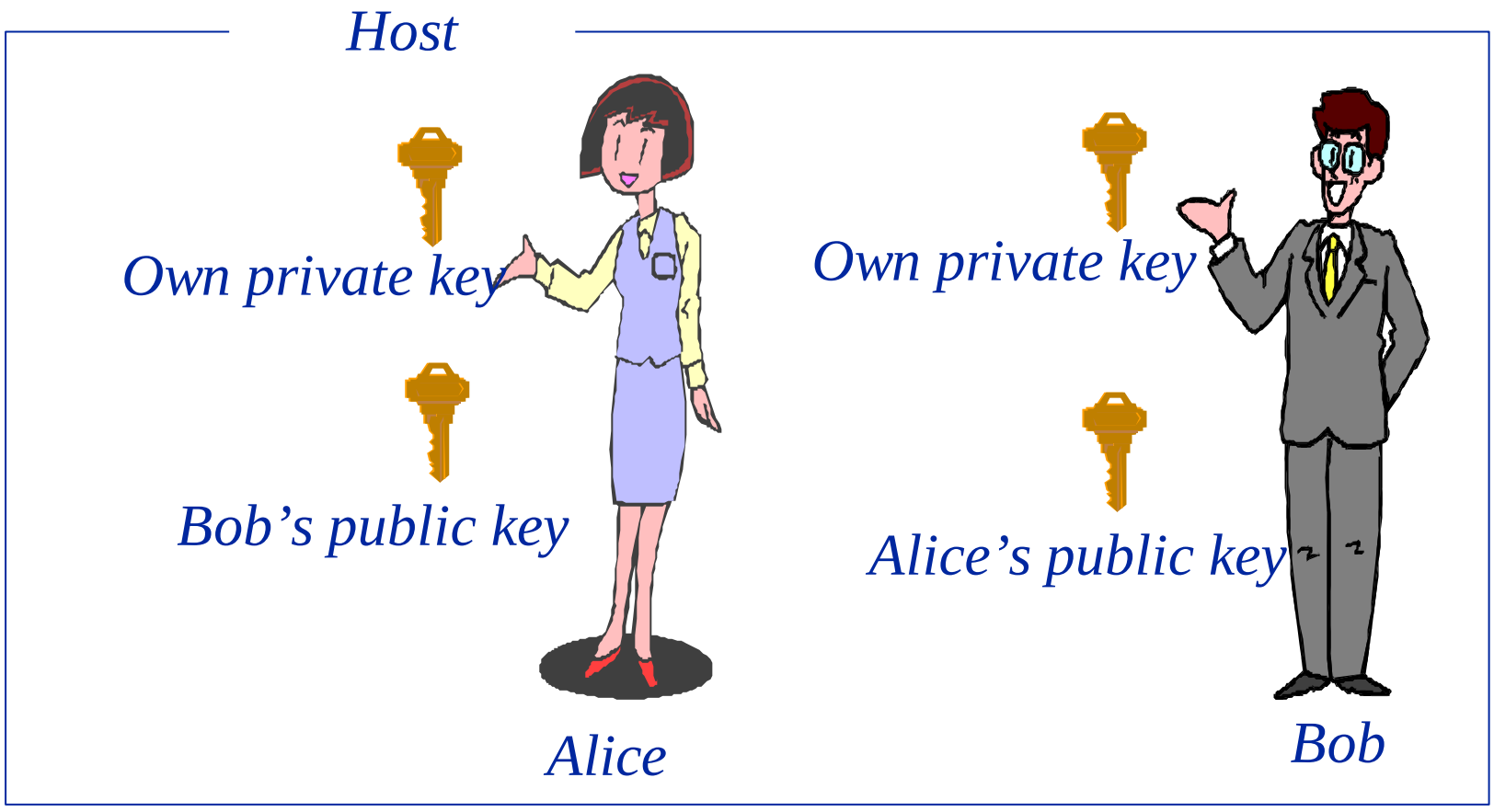
- Assumes a public key infrastructure.
- Uses an X509 certificate as an identifier and a public key.
- Uses the FlexiNet serialiser.
- Independent of crypto algorithms
- Defaults
  - Asymmetric key algorithm RSA
  - Symmetric key algorithm DES
  - Signature algorithm MD5/RSA

Secure Object API

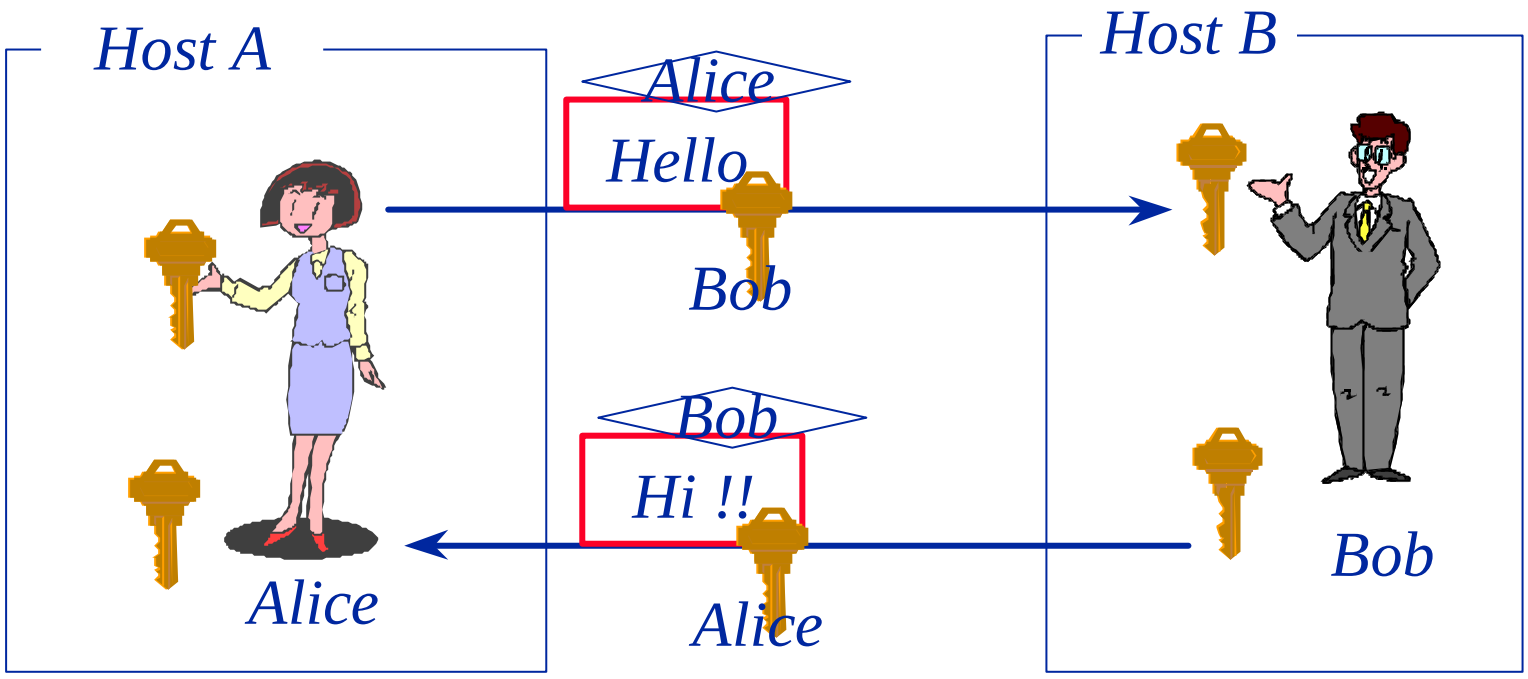
- Revealing/modifying secure data will be transparent
  - put(Object)
  - setPolicy(Policy)
  - commit(X509Certificate,PrivateKey)
  - get(X509Certificate,PrivateKey)
  - verifySignature(PublicKey)
- Policy (Hashtable)
  - X509Certificate <-> READ | WRITE | CHANGEPOLICY
  - put(X509Certificate, READ|WRITE|CHANGEPOLICY)

Secure Communication between Mobile Objects

Secure Communication

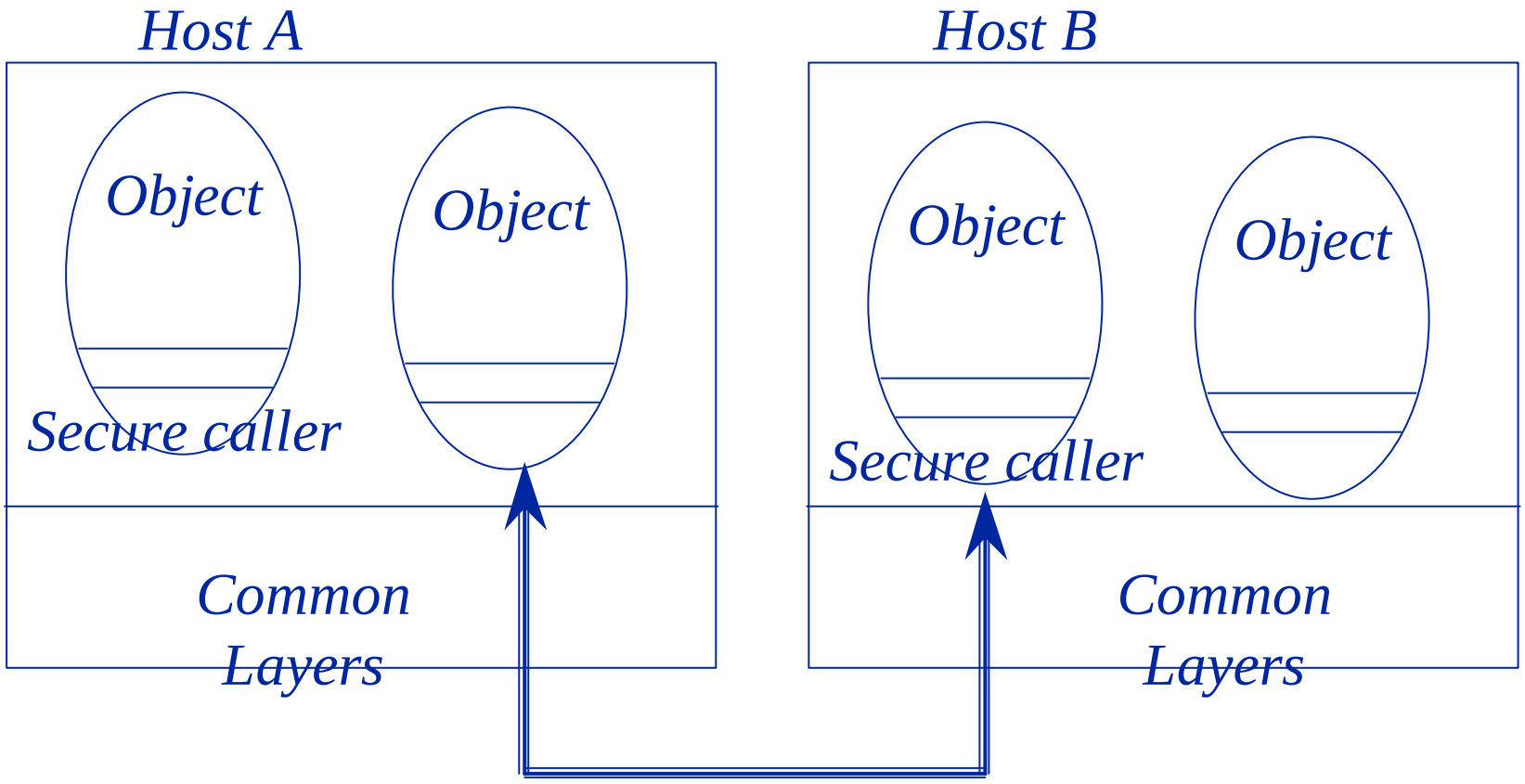


Secure Communication



Messages are encrypted by receiver's public key and signed by sender's private key.

Implementation Architecture



Design

- host to host
  - Using an existing SSL implementation for the communication layer
- object to object
  - require object identity
  - object must reveal proof of identity to host it is on
    - We assume some public key infrastructure and use the X509Certificatefor objects
  - only reasonable at trusted hosts

Implementation

- Caller and ClientCall Layer
  - When a method is called, arguments are encrypted and signed. ClientCall extracts and verifies the arguments.

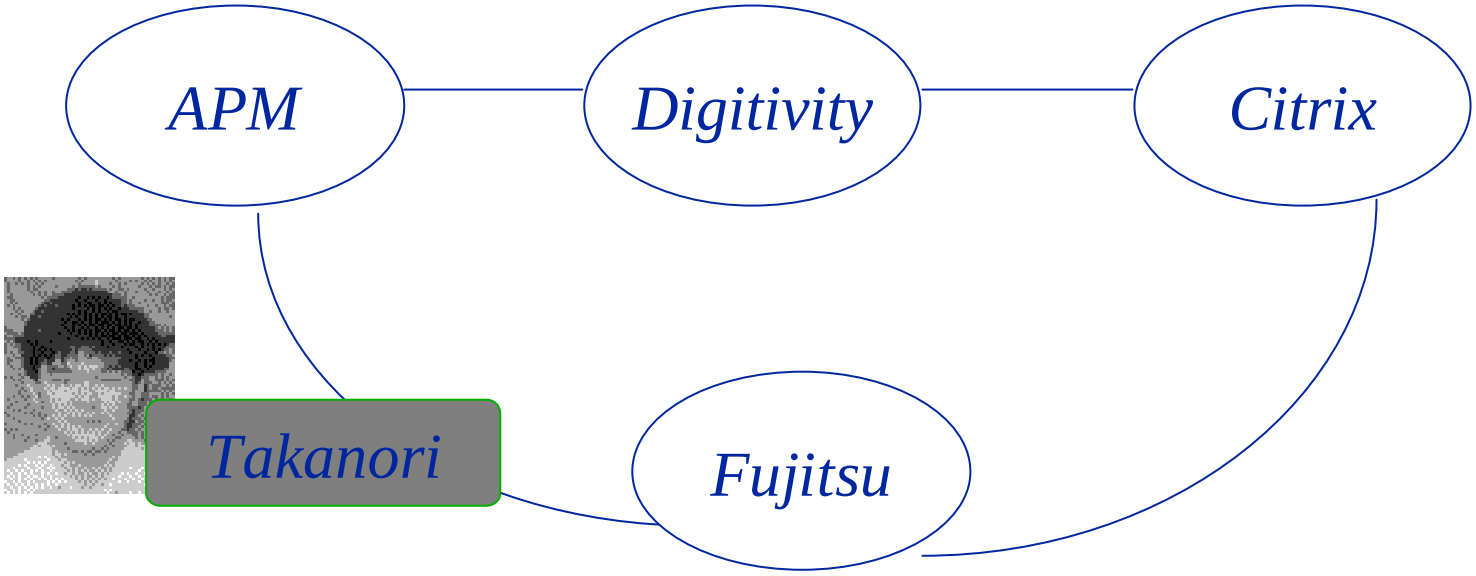
- Class loader is responsible for code integrity. Moving objects can keep some evidence like fingerprint of class data.

Object to Object Policy

- reflective access to supplied credentials

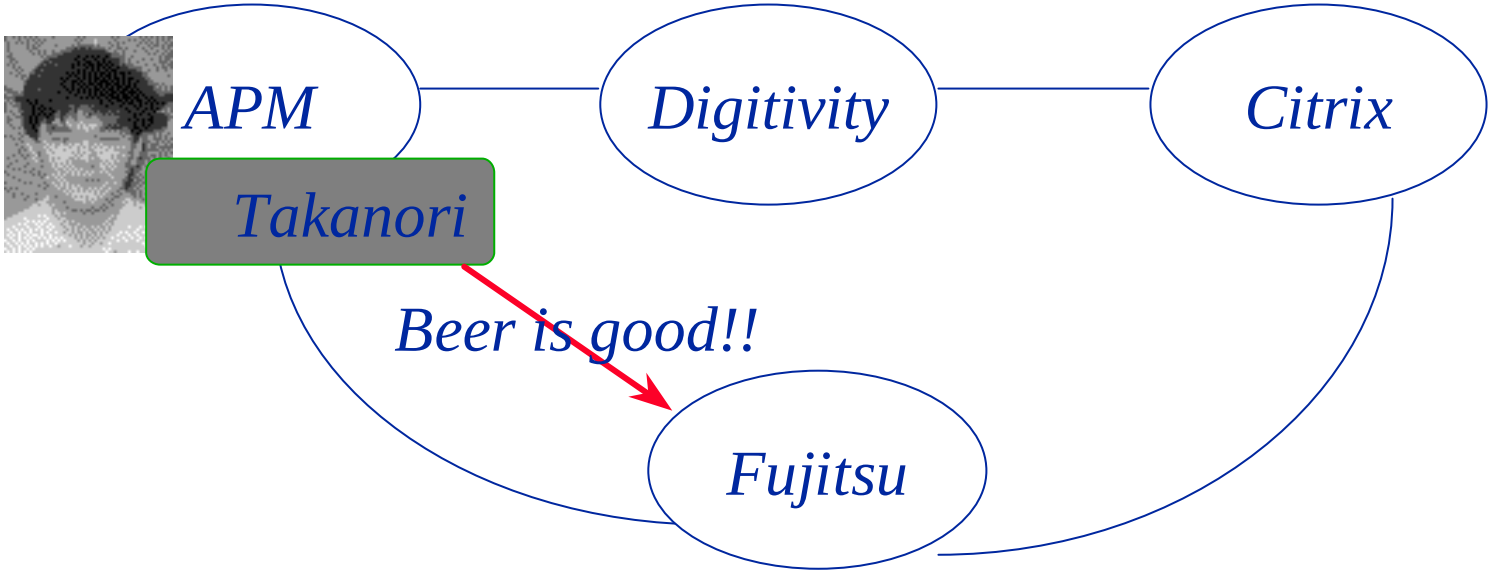
```
public void checkAccess(Object o,
                        Method m,
                        Object args,
                        Certificate client) throws
PolicyException
```

Example: Information gathering

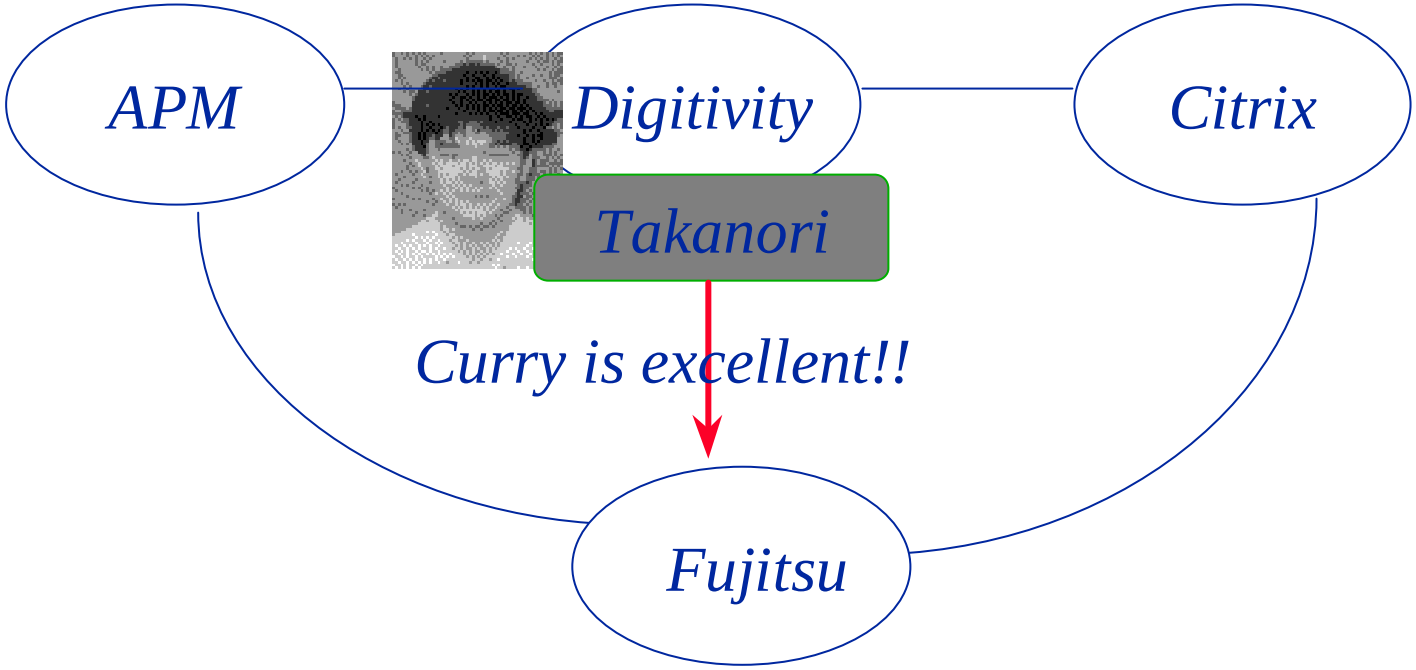


Fujitsu invites Takanori to visit, work and collect information

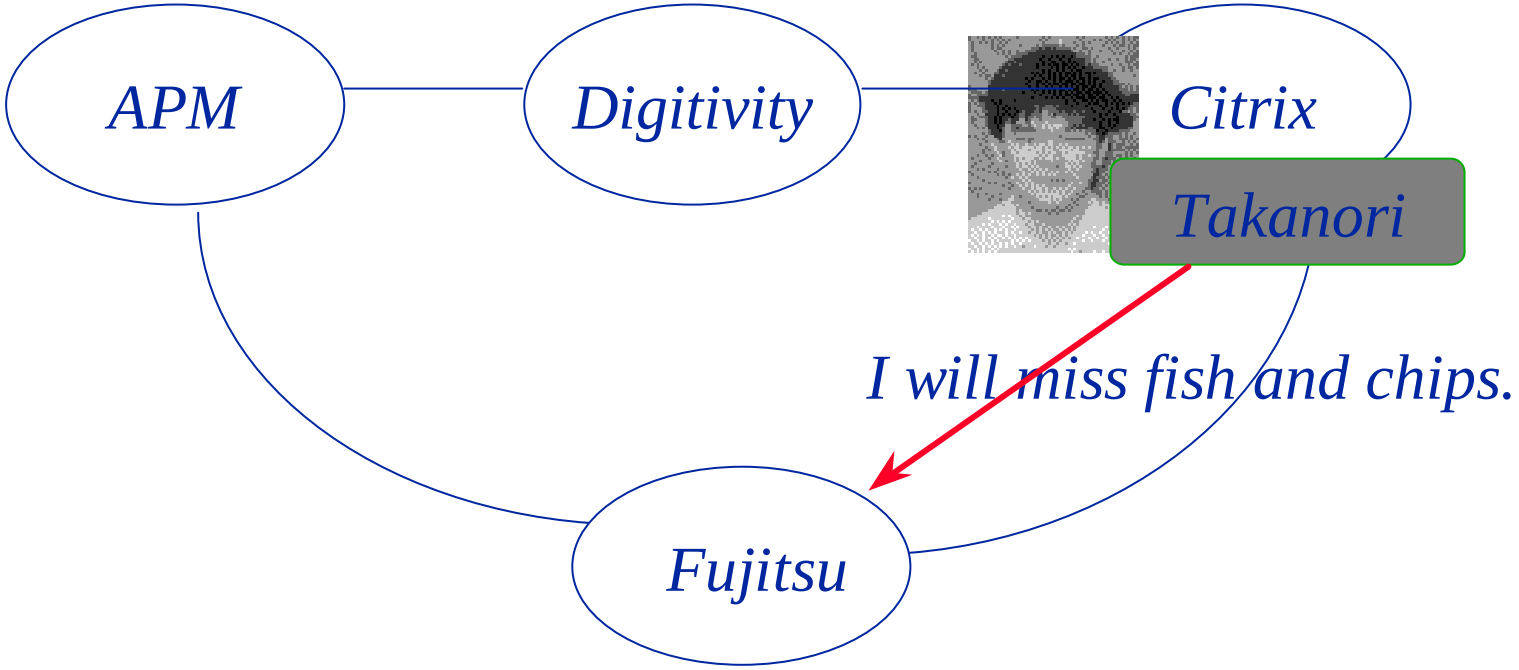
Information gathering



Information gathering



Information gathering



Secure and Insecure

- Secure
- The object is Takanori, not anyone else.
- The transmitted information is safe from other hosts.

- Insecure
- The object's behaviour relies on the host where the sender is on.
- APM, DIGITIVITY and CITRIX must be trustable by Fujitsu.

Summary

- **SSL FlexiNet (Last TC)**
- **SSL MOW (Last TC)**
- **Secure Object Implementation (This TC)**
- **Secure Communication between Mobile Object (This TC)**
- **Demonstration Programs (This TC)**
  - Lottery Example
  - Information Gathering
- **Declarative Mobile Security Pre-Processor (Not yet)**