One-time receiver address in IPv6 for protecting unlinkability

Atsushi Sakurai, Takashi Minohara,
Ryota Sato and Keisuke Mizutani

Takushoku University

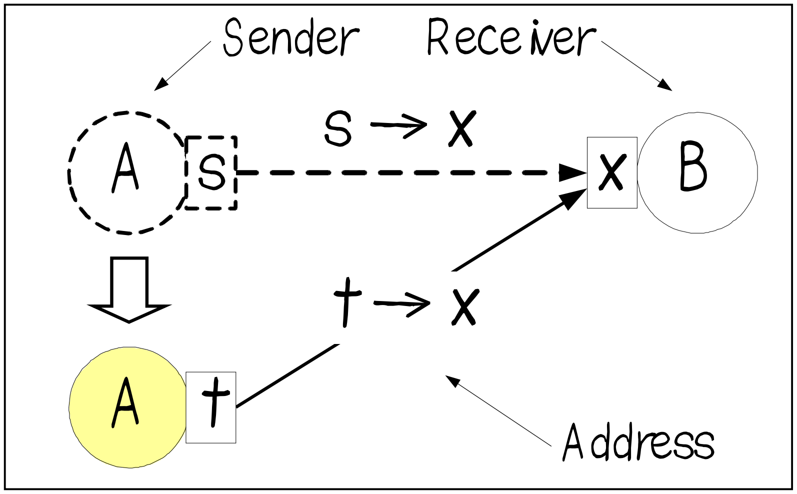
Takushoku University

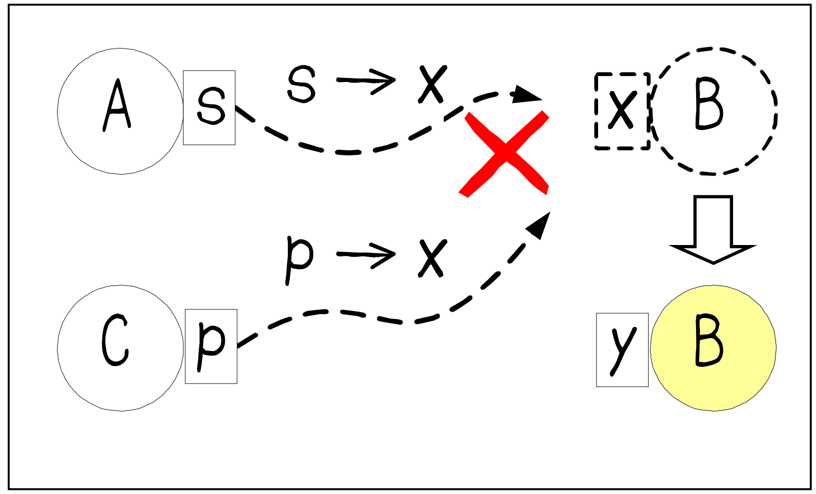
Background

- Privacy is one of the most desirable properties in the Internet communication.
  - Many encryption methods, for example IPsec, PGP and so on, are proposed for protecting privacy in message contents.
  - However, it is difficult to protect address of the message, because it is necessary to deliver the message.
  - It is important to protect privacy in message address.
- We focus on the unlinkability of the Internet address
  - An eavesdropper cannot distinguish whether two or more messages are sent from or sent to the same node.

Changing addresses for unlinkability

- To protect unlinkability, one cannot use the same address for long time.



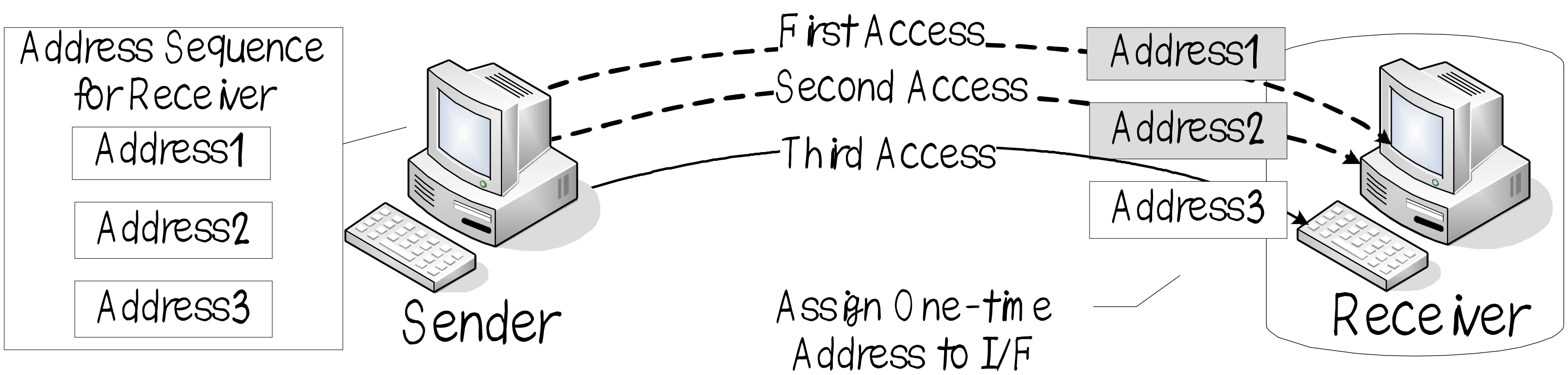


- Sender may change its address.
  - RFC3041: “Privacy extensions for stateless address autoconfiguration in IPv6”
- Changing receiver address is not easy because sender need to know the receiver’s new address before initiating a communication.

Takushoku University

Unlinkability in receiver addresses

- One-time receiver address for unlinkability
  - Receiver frequently changes its address one after another.
  - Proper senders can only follow the change.
  - Those addresses are kept from being linked by the third persons.
- Shared a secret sequence of addresses
  - Each pair of sender and receiver shares a secret sequence of addresses.



- This scheme requires a large address space, and we limit our target to the IPv6

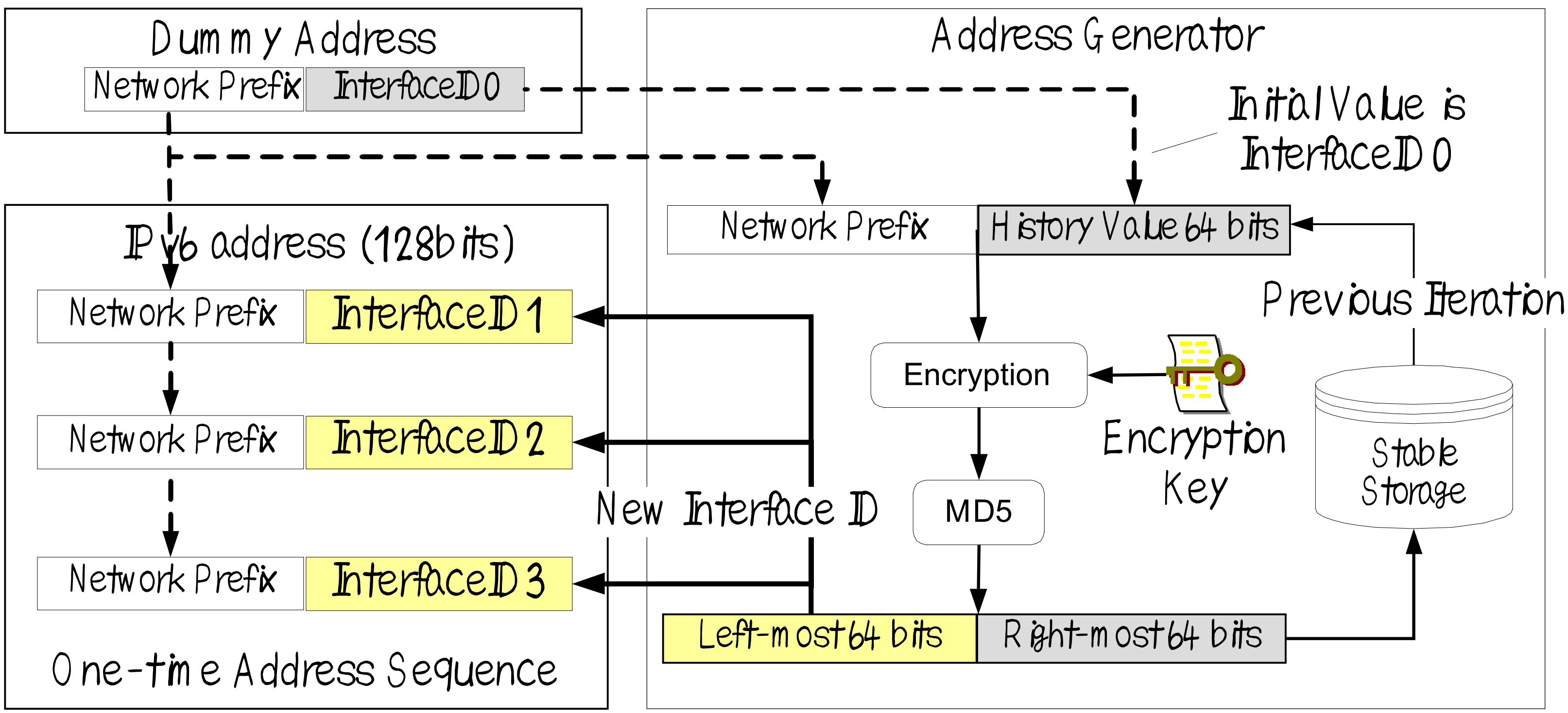
Takushoku University

Sharing a secret address sequence

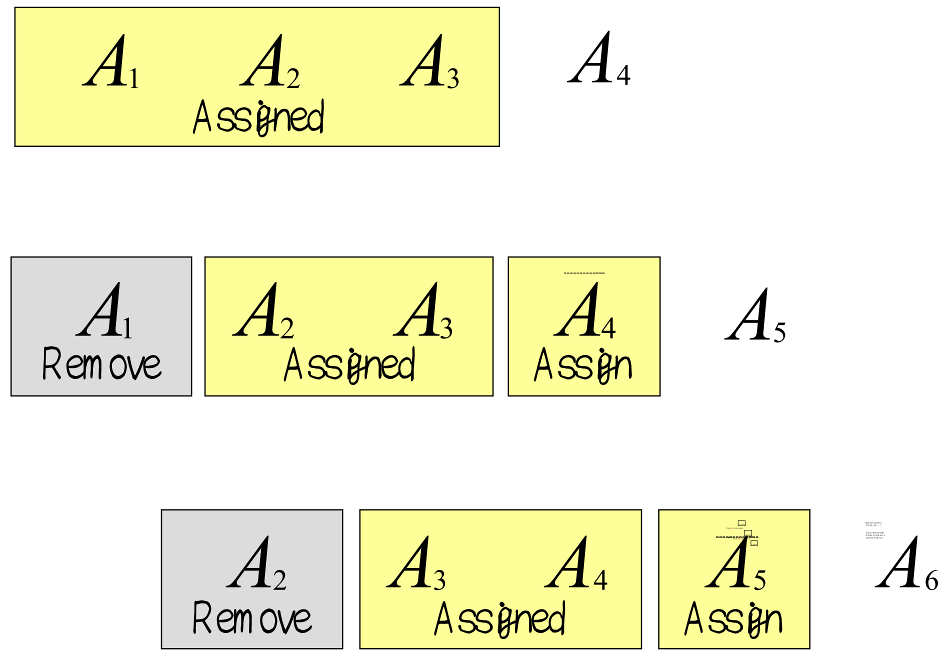
- We assume unlinkability in a receiver address is basically required for a kind of closed community like a friend to friend network.
  - The number of potential senders is limited.
  - Sender and receiver can share a secret encryption key by the method of Diffie-Hellman.
- Receiver generates a sequence of addresses for each potential sender by using the different encryption key.
- Sender independently generates the identical sequence of addresses with the encryption key.

Generation of an address sequence

- Receiver registers a dummy address to public server as a seed of sequence, and sender obtains it.
- The first I/F ID is generated from the dummy address by encryption and calculation of MD5 value.
- And following addresses are generated from the stored value of previous iteration.



On demand assignment of addresses



- Limited range of addresses is assigned to a network interface in order to reduce the number of assigned addresses.
- Assignment, generation and removal of address are triggered by the first access to the address within the range.

Treatment of duplicated address

- Duplication of generated address will rarely happen, but it is unavoidable.
  - Receiver can detect a duplicate address, but sender has no way to detect the duplication unless it is informed from the receiver.
- A new ICMP message is used for skipping the duplicate address
  - Receiver send an ICMP message to inform sender about the duplicate address before it is used.
  - Sender skips the address informed by receiver.

Implementation of Prototype Receiver

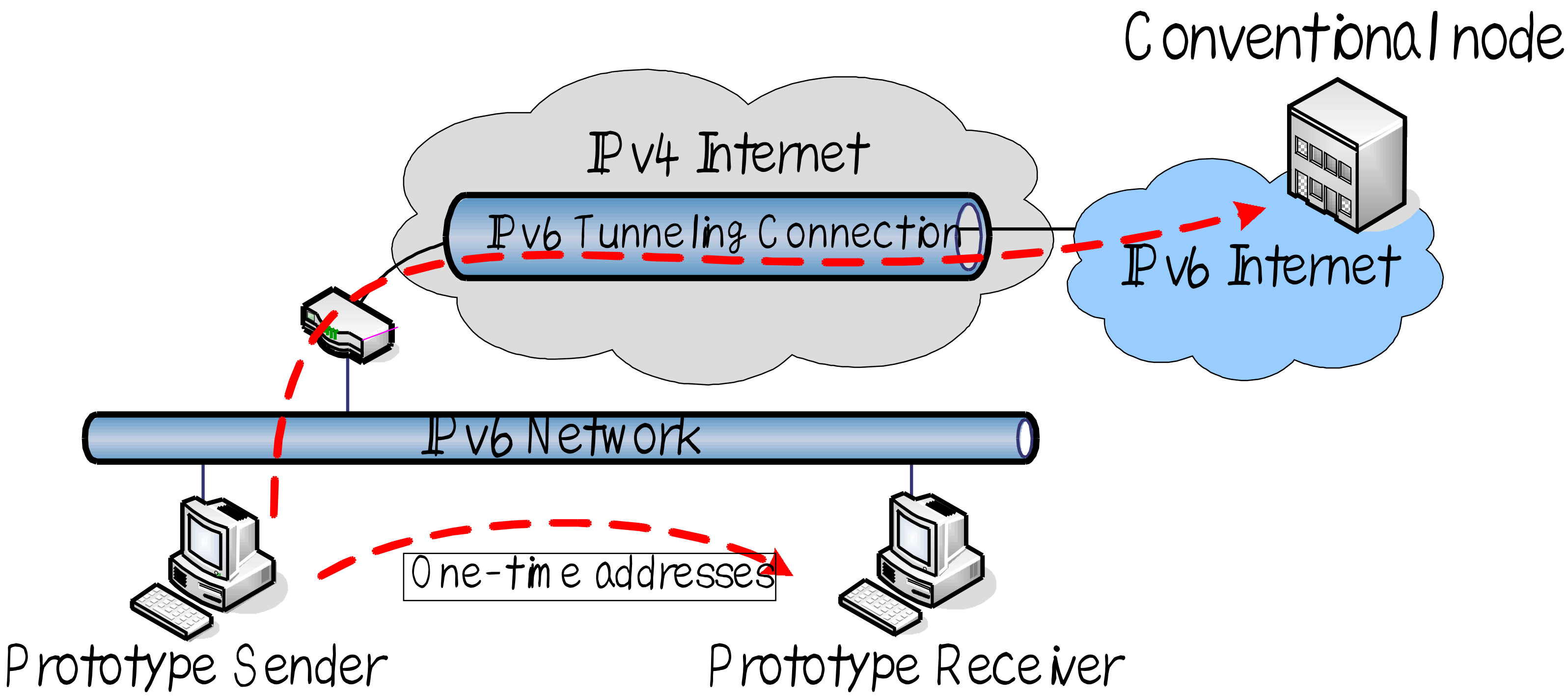
- Implement on the Linux kernel (2.6)
- One-time address generation
  - Obtains a network prefix from a RA (router advertisement) message.
  - Generates a sequence of I/F IDs from a random initial I/F ID and a encryption key.
  - Publishes a dummy address (the network prefix and the initial I/F ID).
- Attachment/detachment of one-time address
  - Attaches an address from the sequence.
  - Detaches the address that is no longer used.

Implementation in the sender side

- It is difficult for user to specify a receiver’s changing one-time address.
  - We provide the mechanism that users can specify target host names in stead of addresses.
  - The address selection mechanism on sender must be transparent to application programs
- We have developed an one-time address resolver, and integrated into name resolver library (glibc-2.4)
  - Returns an one-time address of receiver to user’s program on sender (e.g. when getaddrinfo() is used)

Experimental evaluation

- Prototype nodes works well with conventional IPv6 networks.
  - A prototype receiver is accessed with one-time addresses.
  - Conventional nodes are accessed without any difference.



Takushoku University

Overheads of proposed one-time address

- RTTs are measured between a sender and a receiver connected on a segment.
- Only negligible overheads(< 1ms) are observed for one-time receiver address.
- The overhead of proposed system is negligible.

Table 1. Difference of RTT of between original kernel and modified kernel

Sequence number	1	2	3	4
Normal address on original kernel (ms)	1.115	0.151	0.149	0.150
One-time address on modified kernel (ms)	1.291	0.190	0.191	0.192
Difference of RTT	0.176	0.039	0.042	0.042

Conclusion

- We have developed one-time receiver address for unlinkability.
  - Receiver uses one address after another, and sender follows the change.
  - By using a shared encryption key, both nodes can generate identical sequence of addresses.
- Negligible overheads exist only at the both ends of communication.
  - Our system requires neither multiple relays nor multiple receiver.

Remaining problems and approaches

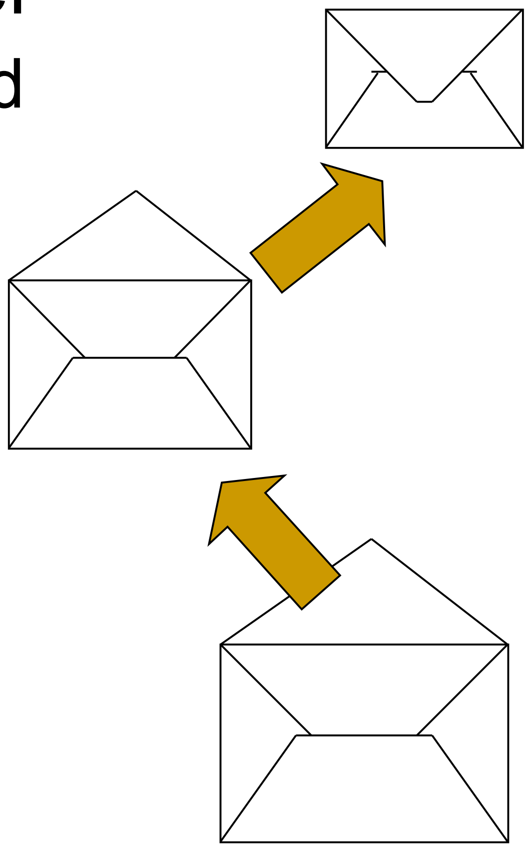
- Linkage by the network prefix
  - Network prefix potion of address remains unchanged.

    ⇒ multiple paths with one hop relays
- Linkage by the link-level header
  - MAC address may disclose a relation between messages.

    ⇒ one-time MAC address for unlinkability

Related work : Onion Routing [D. M. Goldschlag et. al. ,1996]

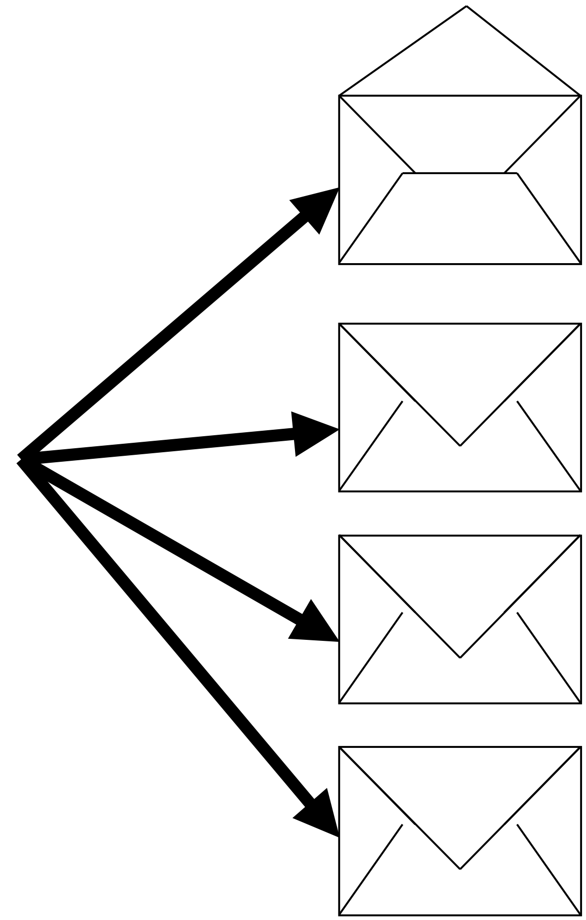
- Anonymity of sender and receiver
  - A message is enclosed in encrypted envelopes over and over.
  - Each envelope can be opened only by the addressed relay node.
- Disadvantage
  - Decryption process is required at every relay node.
  - Large delay may be occurred by multiple relays.



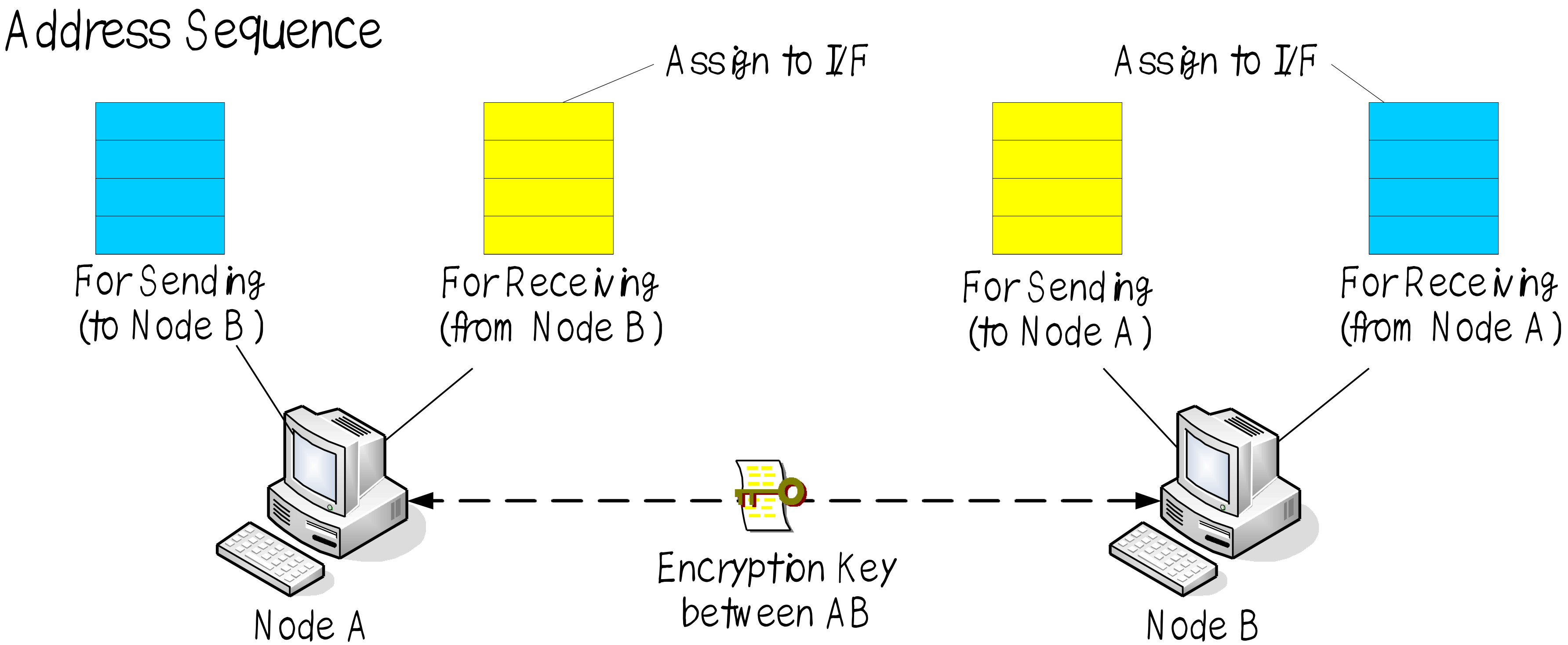
Takushoku University

Related work: Incomparable public keys [B. R. Waters et. al., 2003]

- Anonymity of receiver
  - A message is encrypted and send to multiple nodes (with multicasting.)
  - Only proper receiver can decrypt the message.
- Disadvantage
  - Messages sent to other nodes waste the network resources.



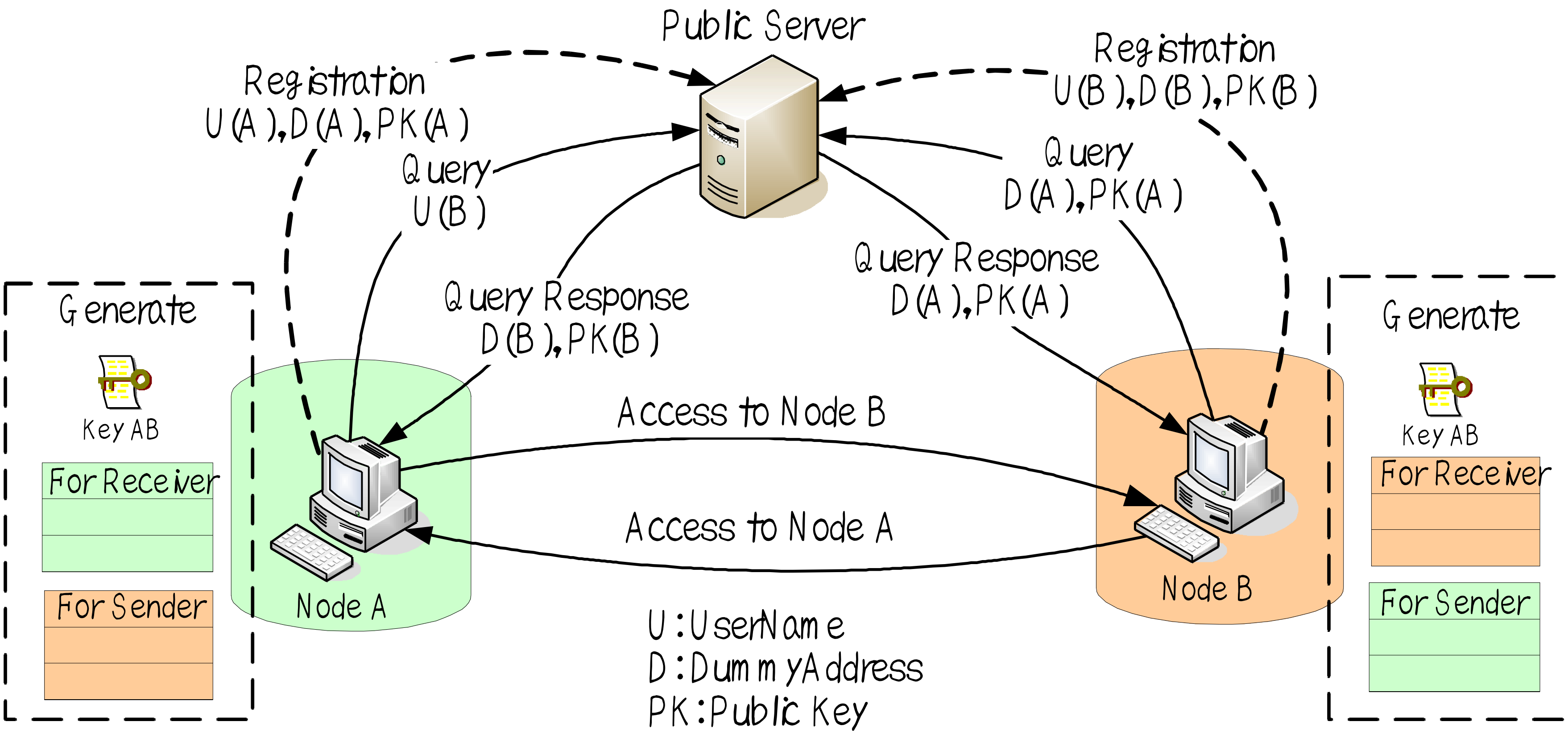
Node has two or more address sequences



- Node becomes sender and receiver so that it has a pair of address sequences for each corresponding node.
- The number of sequences = 2 * (The number of nodes)

Takushoku University

Node generates a pair of address sequences



- Node A and node B register each dummy address and public key to public server.
- Each of node gets the public key of the other node, generates the same encryption key.

Takushoku University