School of
Information Systems

Secure and Practical Key Distribution for RFID-Enabled Supply Chains

Tieyan Li, Yingjiu Li, Guilin Wang

SecureComm 2011, London

Outline

- Motivations
- Related Works
- Contributions
- Scenario
- Desired Security Properties
- Resilient Secret Sharing (RSS)
- Our Construction
- Parameterization
- Conclusion

Motivations

- RFID-enabled supply chains
  - RFID tags, readers, and supply chains
  - RFID security and privacy issues
  - Symmetric key based solutions
  - Key distribution problem
  - Lack of pre-existing trusted relationships in large-scale dynamic supply chains

Related Works

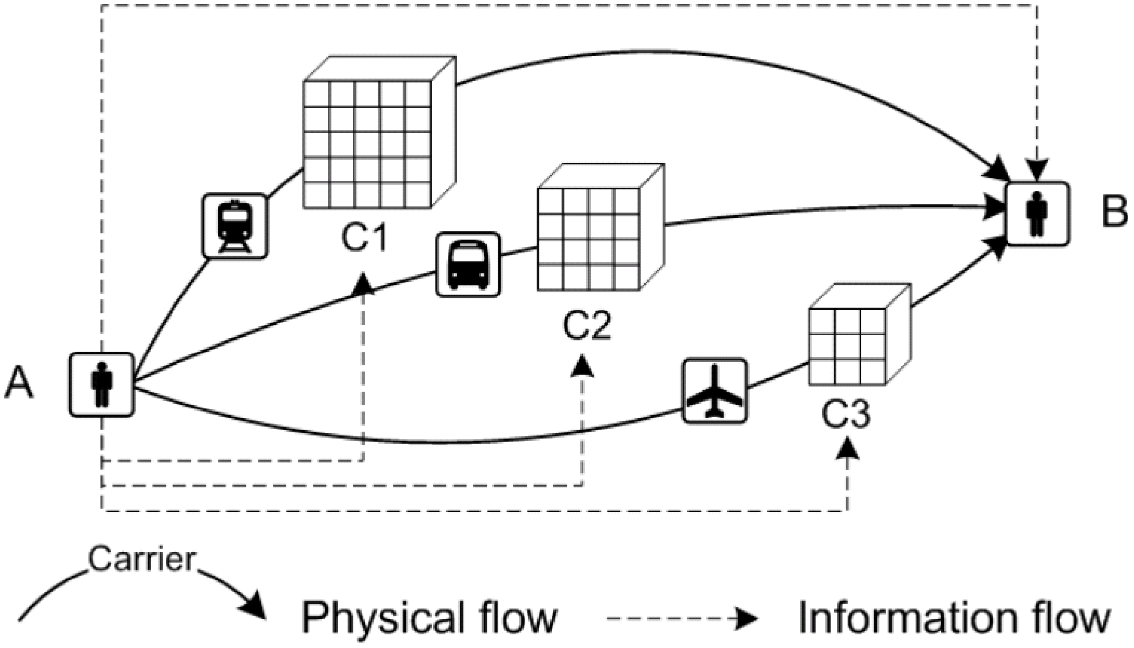
- Centralized control (OSK'04,MW'04,LD'07)
  - A centralized DB manages all tag keys
  - Not practical for large-scale dynamic supply chains
- Secret sharing on tags (LM'07,JPP'08)
  - The encryption key for a batch of tags are shared among the tags
  - Not secure due to weak adv model and clone attack

Contributions

- Secure and practical key distribution for RFID-enabled supply chain
  - Practical: focalized viewpoint on any pair of consecutive parties linked by a transaction and a 3rd party who delivers goods (auth tags with errors) from one party to the other
  - Secure: strong adv model (no clone/privacy attack even for the 3rd party)

Scenario

- Batch goods delivery from A to B by C
  - – Each item is attached with an RFID tag
  - – C can authenticate the tags (with certain errors) but cannot know tag IDs or clone tags



Desired Security Properties

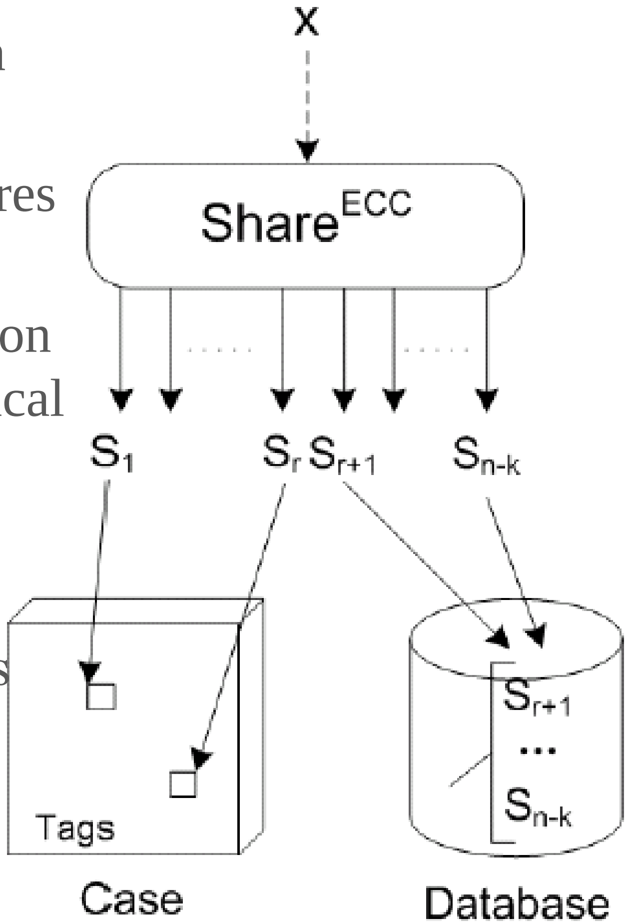
- Authenticity of tags by C
  - Case based, or individual tag authentication
  - Tolerate certain reading errors or tag missing
  - No access to tag content or clone of tags
- Authenticity/accessibility of tags by B
  - Authenticate tags in batch (with robustness)
  - Obtain all secret information to access/update individual tags
- Privacy protection against C/adversary
  - Tag IDs encrypted by A can be accessed/ decrypted by B only (not C or any adversary)

Resilient Secret Sharing (RSS)

- McEliece's RSS based on Reed-Solomon codes (CACM'81)
  - Let B=(b1,b2,…bk) be the secret, where bi in BF(2^m)
  - There exists D=(d1,d2,…dn) in (k,n)-RS code, where di=bi for i<k+1.
  - The last n-k symbols in D are available for distribution to those sharing the secret.
  - At least k shares are required to recover the secret

RSS at High Level

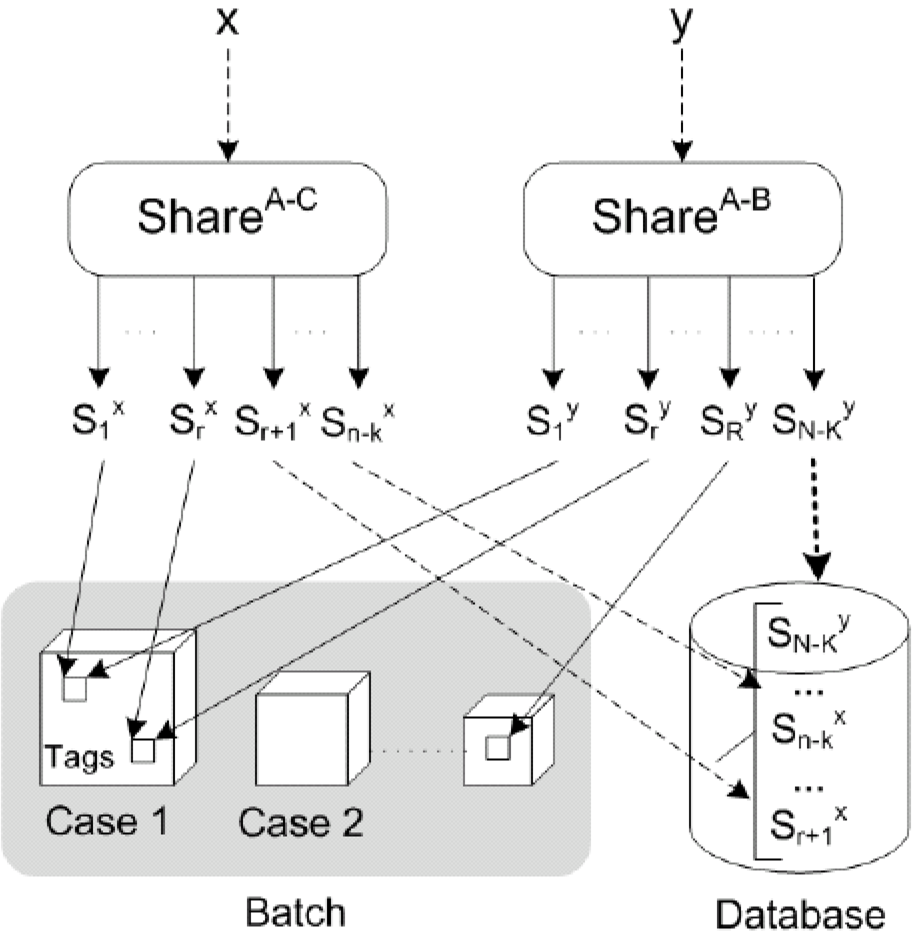
- RSS scheme. A secret x is shared into $n-k$ available shares, in which r shares are distributed into r tags respectively and the other $n-k-r$ shares are stored in a database.
- Recover secret by combining shares from both physical flow and information flow
- Any single flow cannot contribute enough shares (r<k and n-k-r<k)
- Allow more shares contributed from information flow to compensate the missing shares in physical flow
- A minimum number of shares should be contributed from information flow so that an adversary's guessing attack on missing shares is difficult



School of
Information Systems

Our Construction

- R tags in a batch, allocated equally in l cases, with r tags per case (R=l*r)



Our Construction

- EPC C1G2 tags
  - EPC Memory 🡐 48 bits pseudo-ID (PID)
    - 1 share (16 bits) for x
    - 1 share (16 bits) for y
    - 1 sequence order (16 bits)
    - Adversary can access PID, which changes for different peers
  - User Memory 🡐 encryption of 96-bit EPC code
    - Encryption key e=H(y)
    - Accessed and decrypted by B only (ID secrecy, anti-clone)
  - Reserved Memory 🡐 32-bit APIN and 32-bit KPIN
    - APIN = H(x,PID)[15:0] || H(y,PID)[15:0]
    - KPIN = H(x,PID)[31,16] || H(y,PID)[31:16]
    - C can obtain half APIN and KPIN for authentication
    - B can obtain whole APIN and KPIN for auth//acc/ident/upd

Comparison of Security Properties

- [9]: OSK'04
- [8]: MW'04
- [6]: LD'07
- [4]: JPP'08

	Key Storage (DB/Tag)	Authentication (Group/Tag)	Anti-Cloning (Tag Corruption)	Type of Privacy (Unlinkability/ID Secrecy)
[9][8][6]	Central DB	Tag	No	Unlinkability betw. auth. sessions
TSS [4]	Tag	Group	No	ID Secrecy
RSS	Partner DB & Tag	Group & Tag	Yes	Unlinkability betw. diff. peers

Parameterization

- Philips UCODE Gen2 tag (512 bits)
  - EPC (96 bits), TID (32 bits), User (128 bits), Reserved (64 bits for access and kill PINs)
- Running Example
  - 100 tags/batch ➔ 5 cases with 20 tags/case
  - Case level authentication with secret x
    - (28,60)-RSS: 32 shares with 16 bits/share
    - 448-bit secret x
    - 20 shares for tags/case and 12 shares to C
    - C tolerates up to 4 or 20% reading errors on scanning the case
  - Batch level access with secret y
    - (108,236)-RSS: 108 shares with 16 bits/share
    - 1728-bit secret y
    - 100 shares to tags/batch and 28 shares to B
    - B tolerates up to 20 or 20% errors on scanning the batch

Conclusion

- Practical and secure key distribution for RFID-enabled supply chains
  - Peer-to-peer transaction with 3PL
  - 3PL can authenticate tags (in cases) with resilience to certain reading errors
  - No adversary or 3PL can access/clone tag content
  - Receiving party can authenticate/access/update tags (in batches) with resilience to certain reading errors
  - Our solution can be easily incorporated in standard RFID appliances

Thanks!