How to Break „Fraud-Detectable Key-Escrow“

Birgit Pfitzmann
Universität Saabrücken

Michael Waidner
IBM Research Division

Ad:

Eric R. Verheul et. al.: *Binding ElGamal: A Fraud-Detectable Alternative to Key-Escrow Proposals;* Eurocrypt 1997

Slides presented at *Rump Session of Eurocrypt ’97, Konstanz, Germany, May 13, 1997*.
Our attack was independently also discovered by *H. Tiersma*, who gave a presentation at the same Rump Session entitled *“Unbinding ElGamal – An Alternative to Key-escrow?”*

1 Key Recovery Scheme with Binding Information

1.1 Key Recovery Scheme

Ciphertext =

$$< E_{S}(Message) \;||\; E_{Recipient}(S) \;||\; \overbrace{\{E_{TRP}(S)\}^{*}}^{KRF} \;||\; bind >$$

(Key recovery field)

- Trusted Recovery Parties *TRP* as virtual addressees
- No secret decryption keys are escrowed, only session keys, *S*
- „Fraud is detectable:“ Based on *bind*, one can check correctness of *KRF*, without knowing *S*

1.2 Given Model: What Users Can Do

- If *Alice* wants to send something secretly to *Bob*, she has to use KR-scheme

  This means: criminals are too lazy and stupid to use, e.g., PGP

- Only KR keys exchanged in advance.

But

- Dishonest senders can delete *KRF*, or copy *KRF* from other messages. (They cannot perform cryptographic operations.)
- Dishonest recipients can decrypt even if redundant fields do not fit.

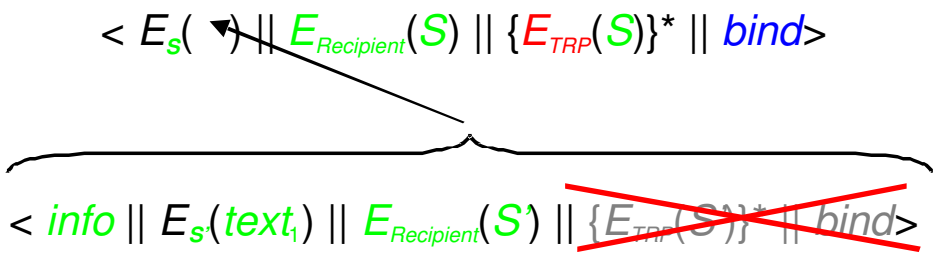
  p. 121: „… abuse … through one-time manipulation of this validation in software — is still easily possible“

  (*Otherwise previous schemes ok*)

2 Simple Attack

KR-Scheme used twice

- *Alice* encrypts $text_1$ for *Bob*, but discards the resulting KRF_1; the result is $text_2$
- *Alice* adds *info* to $text_2$ to tell *Bob* what to do with it, i.e., to decrypt it *again*; the result is $text_2$‘
- *Alice* encrypts $text_2$‘ for Bob again and sends it as supposed, i.e., including KRF_2



2.2 Remarks

- Attack does not require any preestablished agreement between *Alice* and *Bob*
- Would even work if *bind* proved that *S* is used in encryption (it *is* used)
- Assuming that *bind* even proves that *S* is used to encrypt some English text

  a) is unreasonable

  b) $text_2$ might be encoded in some natural and easy-to-describe way („zero zero one …“)

2.3 Countermeasure

Detect and prevent double-encryption

- All communication must be intercepted, decrypted and automatically checked: *infeasible and (hopefully) politically not desired*
- Assume that recipient *cannot* decrypt if *KRF* does not fit: *no need for „bind“ anymore*

Enable wiretapping in spite of double-encryption

- Law-enforcement agencies would need access to user‘s secret decryption key:
  - Real escrow of secret keys
  - Organized „Man-in-the-middle attack“ against all users: *infeasible*

3 Conclusion

- Concept of *binding* does not solve any of the problems of key recovery schemes
- In realistic model even more attacks: whoever might change message fields can also use PGP …
  - Use PKI for signatures to exchange authenticated PGP keys
  - Hide result using legal scheme
- In the worst case: hide the fact of encrypting by hiding the ciphertext in a picture (steganography)
- Plus other risks: key recovery agencies increase risk of key compromise by hackers, bribery, …