Hack.lu 2010 Cryptanalysis Workshop

E. Filiol (filiol@esiea.fr)

Breaking weak or misimplemented systems

An open source cryptanalysis Library

Introduction

- Many encryption systems are
  - Either weakly implemented
  - Or badly used
  - Or crippled with mathematical or implementation backdoors
- How to detect such cases and break the cryptography
  - Without the knowledge of the encryption
  - Without any time-consuming brute force key search

Introduction (2)

- This situation occurs far more frequently than expected
  - Satellite communications
  - Encrypted malware
  - Encryption software
  - Dynamic trapdoors
- Applies for stream ciphers and block ciphers in stream cipher mode.

Bibliography

- E. Filiol - *How to Operationally Detect the Misuse of Stream Ciphers (and even Block Ciphers Sometimes) and Break them*. Black Hat Europe 2010.
- E. Filiol – *Dynamic cryptographic trapdoors: how to trap encryption on-demand*. H2HC 2010 Sao Paulo, Nov, 2010.

Aim of the workshop

- Learn on practical cryptanalysis
- Be able to detect any weak or dangerous encrypted traffic
- Be able to break it without effort.
- Present the Megiddo cryptanalysis library
- Source code and samples provided
- http://code.google.com/p/mediggo/

Megiddo

- Open source cryptanalysis library
- Written in C
- At the present time
  - Detection and cryptanalysis of weakly implemented or trapped systems
- To come
  - Automatic detection of statistical biases in cryptographic algorithms.
  - More to come later...

Basics in cryptography

- Stream ciphers or block ciphers in stream cipher mode
- Weak use/implementation
- Refer to Black Hat 2010 Slides and white paper.

Step I: detection

- Among thousands of encrypted texts, how to detect the weak subsets
- File detect.c

  ./detect <ciphertext_dirname> <outputfile>
- Compute the coincidence indices
- Apply the equivalence relationship to find subsets.

Step II: build a corpus

- The aim is to have a statistical model of the plaintext language.
- File create_corpus.c

  ./create_corpus <ref_text_dirname> <corpus output file>
- Optimal values 4-grams over a 96-character alphabet
  - Covers most of the Western languages.

Step III: decrypt

- From a weak encrypted texts subset, we launch the cryptanalysis
- File decrypt_para.c

  ./decrypt_para <corpus> <sequence _file> <crypto1> <crypto 2>.....
- You have obtained the pseudo-running sequence. You can use it now to decipher each ciphertext:
  - Program decipher.c

decipher ciphertext_file pseudo-random sequence_file plaintext_file

Additional stuff

- texte_extract.c
  - Extract encrypted data in MS Word and MS Excel document (up to Office 2003)
- hack_lu2010_tutorial.pdf
  - The present slides
- Black Hat 2010 slides and white paper
- More to come...

Enjoy cryptanalysis!
Stay tuned with further developments in Megiddo

Thanks for your attention