COMx501: Computer Security and Forensics

Achim D. Brucker

a.brucker@sheffield.ac.uk https://www.brucker.ch/

Software Assurance & Security Research
Department of Computer Science, The University of Sheffield, Sheffield, UK
https://logicalhacking.com/

February 20, 2018

{* LogicaλHacking *}.com

The University Of Sheffield.

COMx501: Computer Security and Forensics

Part 5: A Short Story on Attacking Crypto

Achim D. Brucker

a.brucker@sheffield.ac.uk **https://www.brucker.ch/**

Software Assurance & Security Research
Department of Computer Science, The University of Sheffield, Sheffield, UK
https://logicalhacking.com/

February 20, 2018

{* LogicalHacking *}.com

The University Of Sheffield.

Announcements

- If you **cannot** access the course material in MOLE, solve the problem now!
  - problem sheets (with solutions)
  - discussion forums
  - MOLE quizzes (weeks 7 and 11)
- SURE
- No lecture/tutorial on Thursday

Outline

Motivation & Disclaimer

- Cryptographic schemes are not unbreakable
- To implement systems secure, it is helpful to have an idea how one attacks systems
- Let's have a lock ...

Warning:

The following slides provide only a glimpse into the subject of attacking crypto systems (using a few selected example attacks).

Outline

1. Motivation & Disclaimer
2. Examples of Attacks on Crypto Systems
3. Conclusions
4. Appendix

Cipher Text Only A (COA))

- An attack in which the attacker has only access to
  - cipher text

  and, usually, tries to gain access to the plain text
- Approaches
  - brute force (testing of all/most combinations)
    - works successful on small message sizes (lack of entropy), e.g., passwords
      John the Ripper: http://www.openwall.com/john/
    - can be based on pre-computed data (e.g., hash tables or, more efficient, rainbow tables)
  - statistical analysis (e.g., character or word frequency)
    - For an example, see the current homework paper
- Standardization processes for crypto algorithms:
  - vetting process usually takes several years
  - exhaustive testing of large quantities of ciphertext for any statistical departure from random noise.

Known Plain Text Attacks (KPA)

- An attack in which the attacker has access to
  - a plain text (could be a part of a message)
  - the cipher text of the plain text (or a message containing the plain text)

  and tries to gain access to the encryption key
- Chosen Plain Text Attack
  - the attacker can generate the cipher text for arbitrary plain texts
- The situation today:
  - Modern ciphers (e.g., AES) are currently not known to be susceptible to KPA
  - Old versions of the PKZIP stream cipher are prone to KPA [BK95]
- For an example, see the current homework paper

Birthday Attack

Idea:

- Exploit the Birthday Paradox:
  In a room with 23 people, the probability that two people have their birthday on he same day, is larger than 0.5.
- In a room with 100 people, the probability is 0.9999997

Hash codes revisited:

- A hash is a function that maps a message *m* of variable length to a fixed length hash code
- For hash codes of length l, there are 2^l possible hash codes
  (usually: *m* much longer than l, thus more than one *m* is mapped to the same hash code)
- Birthday paradox: if we generate $2^{\frac{l}{2}}$ message, the probability for a collision is larger 0.5

Birthday Attack

ALetter in 2^{37} variations

Dear Anthony,

$\begin{Bmatrix}\text{This letter is}\\ \text{I am writing}\end{Bmatrix}$ to introduce $\begin{Bmatrix}\text{you to}\\ \text{to you}\end{Bmatrix}$ $\begin{Bmatrix}\text{Mr.}\\ \text{--}\end{Bmatrix}$ Alfred $\begin{Bmatrix}\text{P.}\\ \text{--}\end{Bmatrix}$

Barton, the $\begin{Bmatrix}\text{new}\\ \text{newly appointed}\end{Bmatrix}$ $\begin{Bmatrix}\text{chief}\\ \text{senior}\end{Bmatrix}$ jewellery buyer for $\begin{Bmatrix}\text{our}\\ \text{the}\end{Bmatrix}$

Northern $\begin{Bmatrix}\text{European}\\ \text{Europe}\end{Bmatrix}$ $\begin{Bmatrix}\text{area}\\ \text{division}\end{Bmatrix}$. He $\begin{Bmatrix}\text{will take}\\ \text{has taken}\end{Bmatrix}$ over $\begin{Bmatrix}\text{the}\\ \text{--}\end{Bmatrix}$

responsibility for $\begin{Bmatrix}\text{all}\\ \text{the whole of}\end{Bmatrix}$ our interests in $\begin{Bmatrix}\text{watches and jewellery}\\ \text{jewellery and watches}\end{Bmatrix}$

in the $\begin{Bmatrix}\text{area}\\ \text{region}\end{Bmatrix}$. Please $\begin{Bmatrix}\text{afford}\\ \text{give}\end{Bmatrix}$ him $\begin{Bmatrix}\text{every}\\ \text{all the}\end{Bmatrix}$ help he $\begin{Bmatrix}\text{may need}\\ \text{needs}\end{Bmatrix}$

to $\begin{Bmatrix}\text{seek out}\\ \text{find}\end{Bmatrix}$ the most $\begin{Bmatrix}\text{modern}\\ \text{up to date}\end{Bmatrix}$ lines for the $\begin{Bmatrix}\text{top}\\ \text{high}\end{Bmatrix}$ end of the

market. He is $\begin{Bmatrix}\text{empowered}\\ \text{authorized}\end{Bmatrix}$ to receive on our behalf $\begin{Bmatrix}\text{samples}\\ \text{specimens}\end{Bmatrix}$ of the

$\begin{Bmatrix}\text{latest}\\ \text{newest}\end{Bmatrix}$ $\begin{Bmatrix}\text{watch and jewellery}\\ \text{jewellery and watch}\end{Bmatrix}$ products, $\begin{Bmatrix}\text{up}\\ \text{subject}\end{Bmatrix}$ to a $\begin{Bmatrix}\text{limit}\\ \text{maximum}\end{Bmatrix}$

of ten thousand dollars. He will $\begin{Bmatrix}\text{carry}\\ \text{hold}\end{Bmatrix}$ a signed copy of this $\begin{Bmatrix}\text{letter}\\ \text{document}\end{Bmatrix}$

as proof of identity. An order with his signature, which is $\begin{Bmatrix}\text{appended}\\ \text{attached}\end{Bmatrix}$

$\begin{Bmatrix}\text{authorizes}\end{Bmatrix}$ you to charge the cost to this company at the $\begin{Bmatrix}\text{above}\end{Bmatrix}$

- It is, e.g., not difficult to generate 2^{37} documents that convey the same message
- Could be used for forging digital signatures
- Might be even easier for real hash algorithms (e.g., old members of the MD family) [BK04]

Observations:

- The security of many cryptographic schemes relies on strong random number generators (i.e., sequences of unpredictable random numbers that cannot be distinguished from "noise")
- Humans are bad in generating random numbers (think of passwords ...)
- Computers as well: many pseudo-random-number generators (PRNG) can easily predicted, e.g.,
  do not use `java.util.Random` for security critical implementations
- Random generators for security-relevant implementations should include entropy from physical measurements and/or hardware devices.

Known examples:

- Netscape seed: early versions of Netscape's SSL implementation used a PRNG that used three inputs as seeds:
  the time of day, the process ID, and the parent process ID
- The Java class SecureRandom could generate collisions in the *k* nonce values used for ECDSA in implementations of Bitcoin on Android (2013)

Outline

Conclusion

- Implementing crypto correctly is hard
- Many implementations that look secure on the first sight (e.g., `java.util.Random`) are actually insecure
- Attacks on the “heart” of cryptographic schemes are usually difficult

Thank you for your attention!
Any questions or remarks?

Contact:

Dr. Achim D. Brucker
Department of Computer Science
University of Sheffield
Regent Court
211 Portobello St.
Sheffield S1 4DP, UK

a.brucker@sheffield.ac.uk
@adbrucker
https://de.linkedin.com/in/adbrucker/
https://www.brucker.ch/
https://logicalhacking.com/blog/

Bibliography I

Ross J. Anderson.
Security Engineering: A Guide to Building Dependable Distributed Systems.
John Wiley & Sons, Inc., New York, NY, USA, 1st edition, 2001.
The complete book is available at: http://www.cl.cam.ac.uk/~rja14/book.html.

Eli Biham and Paul C. Kocher.
A known plaintext attack on the PKZIP stream cipher, pages 144–153.
Springer Berlin Heidelberg, Berlin, Heidelberg, 1995.

Mihir Bellare and Tadayoshi Kohno.
Hash Function Balance and Its Impact on Birthday Attacks, pages 401–418.
Springer Berlin Heidelberg, Berlin, Heidelberg, 2004.

Alfred J. Menezes, Scott A. Vanstone, and Paul C. Van Oorschot.
Handbook of Applied Cryptography.
CRC Press, Inc., Boca Raton, FL, USA, 5th edition, 2001.
The complete book is available at: http://cacr.uwaterloo.ca/hac/.

Document Classification and License Information