(9)

Methods of cryptanalysis

Classical cryptanalysis:

- Frequency analysis
- Index of coincidence
- Kasiski examination

Symmetric algorithms:

- Boomerang attack
- Brute force attack
- Davies' attack
- Differential cryptanalysis
- Impossible differential cryptanalysis
- Improbable differential cryptanalysis
- Integral cryptanalysis
- Linear cryptanalysis
- Meet-in-the-middle attack
- Mod-n cryptanalysis
- Related-key attack
- Sandwich attack
- Slide attack
- XSL attack

Hash functions:

- Birthday attack
- Rainbow table

Attack models:

- Chosen-ciphertext attack
- Chosen-plaintext attack
- Ciphertext-only attack
- Known-plaintext attack

Side channel attacks:

- Power analysis
- Timing attack

Network attacks:

- Man-in-the-middle attack
- Replay attack

External attacks:

- Black-bag cryptanalysis

Rubber-hose cryptanalysis

Attack model

Attack models or **attack types** specify how much information a cryptanalyst has access to when cracking an encrypted message (also known as *ciphertext*). Some common attack models are:

- Ciphertext-only attack
- Known-plaintext attack
  - During WWII, the Allies used known-plaintexts ("cribs") in their successful Cryptanalysis of the Enigma machine cipher.
- Chosen-plaintext attack
- Chosen-ciphertext attack
  - Adaptive chosen-ciphertext attack
  - Indifferent chosen-ciphertext attack

The ciphertext-only attack model is the weakest because it implies that the cryptanalyst has just the encoded message. Modern ciphers rarely fail under this attack type. Different attack models are used for other cryptographic primitives, or more generally for all kind of security systems. Examples for such attack models are:

- Adaptive chosen-message attack for digital signatures

Ciphertext-Only Attack

In cryptography, a **ciphertext-only attack (COA)** or **known ciphertext attack** is an attack model for cryptanalysis where the attacker is assumed to have access only to a set of ciphertexts.

The attack is completely successful if the corresponding plaintexts can be deduced, or even better, the key. The ability to obtain any information at all about the underlying plaintext is still considered a success. For example, if an adversary is sending ciphertext continuously to maintain traffic-flow security, it would be very useful to be able to distinguish real messages from nulls. Even making an informed guess of the existence of real messages would facilitate traffic analysis.

In the history of cryptography, early ciphers, implemented using pen-and-paper, were routinely broken using ciphertexts alone. Cryptographers developed statistical techniques for attacking ciphertext, such as frequency analysis. Mechanical encryption devices such as Enigma made these attacks much more difficult (although, historically, Polish cryptographers were able to mount a successful ciphertext-only cryptanalysis of the Enigma by exploiting an insecure protocol for indicating the message settings).

Every modern cipher attempts to provide protection against ciphertext-only attacks. The vetting process for a new cipher design standard usually takes many years and includes exhaustive testing of large quantities of ciphertext for any statistical departure from random noise. *See:* Advanced Encryption Standard process. Also, the field of steganography evolved, in part, to develop methods like mimic functions that allow one piece of data to adopt the statistical profile of another. Nonetheless poor cipher usage or reliance on home-grown proprietary algorithms that have not been subject to thorough scrutiny has resulted in many computer-age encryption systems that are still subject to ciphertext-only attack. Examples include:

- Early versions of Microsoft's PPTP virtual private network software used the same RC4 key for the sender and the receiver (later versions had other problems). In any case where a stream cipher like RC4 is used twice with the same key it is open to ciphertext-only attack. *See:* stream cipher attack
- Wired Equivalent Privacy (WEP), the first security protocol for Wi-Fi, proved vulnerable to several attacks, most of them ciphertext-only.
- Some modern cipher designs have later been shown to be vulnerable to ciphertext-only attacks. For example, Akelarre.
- A cipher whose key space is too small is subject to brute force attack with access to nothing but ciphertext by simply trying all possible keys. All that is needed is some way to distinguish valid plaintext from random noise, which is easily done for natural

languages when the ciphertext is longer than the unicity distance. One example is DES, which only has 56-bit keys. All too common current examples are commercial security products that derive keys for otherwise impregnable ciphers like AES from a user-selected password. Since users rarely employ passwords with anything close to the entropy of the cipher's key space, such systems are often quite easy to break in practice using only ciphertext.

Known-plaintext attack

The **known-plaintext attack (KPA)** or **crib** is an attack model for cryptanalysis where the attacker has samples of both the plaintext and its encrypted version (ciphertext), and is at liberty to make use of them to reveal further secret information such as secret keys and code books. The term "crib" originated at Bletchley Park, the British World War II decryption operation.

History

The usage "crib" was adapted from a slang term referring to cheating—thus, "I cribbed my answer from your test paper." A "crib" originally was a literal or interlinear translation of a foreign-language text — usually a Latin or Greek text — that students might be assigned to translate from the original language.

The idea behind a crib is that cryptologists were looking at incomprehensible ciphertext, but if they had a clue about some word or phrase that might be expected to be in the ciphertext, they would have a "wedge"—a test to break into it. If their otherwise random attacks on the cipher managed to sometimes produce those words or (preferably) phrases, they would know they might be on the right track. When those words or phrases appeared, they would feed the settings they had used to reveal them back into the whole encrypted message, to good effect.

In the case of Enigma, the German High Command was very meticulous about the overall security of the Enigma system, but nonetheless understood the possible problem of cribs. The day-to-day trench operators, on the other hand, were less careful. The Bletchley Park team would guess some of the plaintext based upon when the message was sent. For instance, a daily weather report was transmitted by the Germans, at the same time every day. Due to the regimented style of military

reports, it would contain the word "Wetter" (German for "weather") at the same location in every message, and knowing the local weather conditions helped Bletchley Park guess other parts of the plaintext as well. For example, an officer in the Africa Corps helped greatly by constantly sending: “Nothing to report.” Other operators too would send standard salutations or introductions. Standardized weather reports were also particularly helpful.

At Bletchley Park in World War II, strenuous efforts were made to use and even force the Germans to produce messages with known plaintext; schemes to force the Germans to produce them were called "gardening". For example, when cribs were lacking, Bletchley Park would sometimes ask the Royal Air Force to “seed” a particular area in the North Sea with mines (a process that came to be known as gardening, by obvious reference). The Enigma messages that were shortly sent out would most likely contain the name of the area, or the harbour threatened by the mines.

When a captured German revealed under interrogation that Enigma operators had been instructed to encode numbers by spelling them out, Alan Turing reviewed decrypted messages, and determined that the number “*eins*” ("1") appeared in 90% of messages. He automated the crib process, creating the Eins Catalogue, which assumed that “*eins*” was encoded at all positions in the plaintext. The catalogue included every possible position of the various rotors, starting positions, and keysettings of the Enigma.

The Polish Cipher Bureau had likewise exploited "cribs" in the "ANX method" before World War II (the Germans' use of "ANX" — German for "To," followed by "X" as a spacer.)[3]

Classical ciphers are typically vulnerable to known-plaintext attack. For example, a Caesar cipher can be solved using a single letter of corresponding plaintext and ciphertext to decrypt entirely. A general monoalphabetic substitution cipher needs several character pairs and some guessing if there are fewer than 26 distinct pairs.

Present day

Modern ciphers such as Advanced Encryption Standard are not susceptible to known-plaintext attacks.

Encrypted file archives such as ZIP are prone to this attack. For example, an attacker with an encrypted ZIP file needs only one unencrypted file from the archive which forms the "known-plaintext". Then using some publicly available software they can quickly calculate the key required to decrypt the entire archive. To obtain this unencrypted file the attacker could search the website for a suitable file, find it from another archive they can open, or manually try to reconstruct a plaintext file armed with the knowledge of the filename from the encrypted archive.

Chosen-Plaintext Attack

A **chosen-plaintext attack (CPA)** is an attack model for cryptanalysis which presumes that the attacker has the capability to choose arbitrary plaintexts to be encrypted and obtain the corresponding ciphertexts. The goal of the attack is to gain some further information which reduces the security of the encryption scheme. In the worst case, a chosen-plaintext attack could reveal the scheme's secret key.

This appears, at first glance, to be an unrealistic model; it would certainly be unlikely that an attacker could persuade a human cryptographer to encrypt large amounts of plaintexts of the attacker's choosing. Modern cryptography, on the other hand, is implemented in software or hardware and is used for a diverse range of applications; for many cases, a chosen-plaintext attack is often very feasible. Chosen-plaintext attacks become extremely important in the context of public key cryptography, where the encryption key is public and attackers can encrypt any plaintext they choose.

Any cipher that can prevent chosen-plaintext attacks is then also guaranteed to be secure against known-plaintext and ciphertext-only attacks; this is a conservative approach to security.

Two forms of chosen-plaintext attack can be distinguished:

- **Batch chosen-plaintext attack**, where the cryptanalyst chooses all plaintexts before any of them are encrypted. This is often the meaning of an unqualified use of "chosen-plaintext attack".
- **Adaptive chosen-plaintext attack**, where the cryptanalyst makes a series of interactive queries, choosing subsequent plaintexts based on the information from the previous encryptions.

Non-randomized (deterministic) public key encryption algorithms are vulnerable to simple "dictionary"-type attacks, where the attacker builds a table of likely messages and their corresponding ciphertexts. To find the decryption of some observed ciphertext, the attacker simply looks the ciphertext up in the table. As a result, public-key definitions of security under chosen-plaintext attack require probabilistic encryption (i.e., randomized encryption). Conventional symmetric ciphers, in which the same key is used to encrypt and decrypt a text, may also be vulnerable to other forms of chosen-plaintext attack, for example, differential cryptanalysis of block ciphers.

A technique termed *Gardening* was used by Allied codebreakers in World War II who were solving messages encrypted on the Enigma machine. Gardening can be viewed as a chosen-plaintext attack.

Chosen Ciphertext Attack

A **chosen-ciphertext attack (CCA)** is an attack model for cryptanalysis in which the cryptanalyst gathers information, at least in part, by choosing a ciphertext and obtaining its decryption under an unknown key. In the attack, an adversary has a chance to enter one or more known ciphertexts into the system and obtain the resulting plaintexts. From these pieces of information the adversary can attempt to recover the hidden secret key used for decryption.

A number of otherwise secure schemes can be defeated under chosen-ciphertext attack. For example, the El Gamal cryptosystem is semantically secure under chosen-plaintext attack, but this semantic security can be trivially defeated under a chosen-ciphertext attack. Early versions of RSA padding used in the SSL protocol were vulnerable to a sophisticated adaptive chosen-ciphertext attack which revealed SSL session keys. Chosen-ciphertext attacks have implications for some self-synchronizing stream ciphers as well. Designers of tamper-resistant cryptographic smart cards must be particularly cognizant of these attacks, as these devices may be completely under the control of an adversary, who can issue a large number of chosen-ciphertexts in an attempt to recover the hidden secret key.

When a cryptosystem is vulnerable to chosen-ciphertext attack, implementers must be careful to avoid situations in which an adversary might be able to decrypt chosen-ciphertexts (i.e., avoid providing a

decryption oracle). This can be more difficult than it appears, as even partially-chosen-ciphertexts can permit subtle attacks. Additionally, some cryptosystems (such as RSA) use the same mechanism to sign messages and to decrypt them. This permits attacks when hashing is not used on the message to be signed. A better approach is to use a cryptosystem which is provably secure under chosen-ciphertext attack, including (among others) RSA-OAEP, Cramer-Shoup and many forms of authenticated symmetric encryption.

Varieties of Chosen-Ciphertext Attacks

Chosen-ciphertext attacks, like other attacks, may be adaptive or non-adaptive. In a non-adaptive attack, the attacker chooses the ciphertext or ciphertexts to decrypt in advance, and does not use the resulting plaintexts to inform their choice for more ciphertexts. In an adaptive chosen-ciphertext attack, the attacker makes their ciphertext choices adaptively, that is, depending on the result of prior decryptions.

Lunch Time Attacks

A specially noted variant of the chosen-ciphertext attack is the "lunchtime", "midnight", or "indifferent" attack, in which an attacker may make adaptive chosen-ciphertext queries but only up until a certain point, after which the attacker must demonstrate some improved ability to attack the system. The term "lunchtime attack" refers to the idea that a user's computer, with the ability to decrypt, is available to an attacker while the user is out to lunch. This form of the attack was the first one commonly discussed: obviously, if the attacker has the ability to make adaptive chosen ciphertext queries, no encrypted message would be safe, at least until that ability is taken away. This attack is sometimes called the "non-adaptive chosen ciphertext attack"; here, "non-adaptive" refers to the fact that the attacker cannot adapt their queries in response to the challenge, which is given after the ability to make chosen ciphertext queries has expired.

Adaptive Chosen-Ciphertext Attack

A (full) adaptive chosen-ciphertext attack is an attack in which ciphertexts may be chosen adaptively before and after a challenge ciphertext is given to the attacker, with only the stipulation that the challenge ciphertext may not itself be queried. This is a stronger attack notion than the lunchtime attack, and is commonly referred to as a CCA2 attack, as compared to a CCA1 (lunchtime) attack. Few practical attacks

are of this form. Rather, this model is important for its use in proofs of security against chosen-ciphertext attacks. A proof that attacks in this model are impossible implies that any realistic chosen-ciphertext attack cannot be performed.

A practical adaptive chosen-ciphertext attack is the Bleichenbacher attack against PKCS#1.

Cryptosystems proven secure against adaptive chosen-ciphertext attacks include the Cramer-Shoup system[1] and RSA-OAEP.

Adaptive Chosen-Ciphertext Attack

An **adaptive chosen-ciphertext attack** (abbreviated as **CCA2**) is an interactive form of chosen-ciphertext attack in which an attacker sends a number of ciphertexts to be decrypted, then uses the results of these decryptions to select subsequent ciphertexts. It is to be distinguished from an indifferent chosen-ciphertext attack (CCA1).

The goal of this attack is to gradually reveal information about an encrypted message, or about the decryption key itself. For public-key systems, adaptive-chosen-ciphertexts are generally applicable only when they have the property of ciphertext malleability — that is, a ciphertext can be modified in specific ways that will have a predictable effect on the decryption of that message.

Practical attacks

Adaptive-chosen-ciphertext attacks were largely considered to be a theoretical concern until 1998, when Daniel Bleichenbacher of Bell Laboratories demonstrated a practical attack against systems using RSA encryption in concert with the PKCS#1 v1 encoding function, including a version of the Secure Socket Layer (SSL) protocol used by thousands of web servers at the time.

The Bleichenbacher attacks took advantage of flaws within the PKCS #1 function to gradually reveal the content of an RSA encrypted message. Doing this requires sending several million test ciphertexts to the decryption device (eg, SSL-equipped web server.) In practical terms, this

means that an SSL session key can be exposed in a reasonable amount of time, perhaps a day or less.

Preventing attacks

In order to prevent adaptive-chosen-ciphertext attacks, it is necessary to use an encryption or encoding scheme that limits ciphertext malleability. A number of encoding schemes have been proposed; the most common standard for RSA encryption is Optimal Asymmetric Encryption Padding (OAEP). Unlike ad-hoc schemes such as the padding used in the early versions of PKCS#1, OAEP has been proven secure in the random oracle model[2]. OAEP was incorporated into PKCS#1 as of version 2.0 published in 1998 as the now-recommended encoding scheme, with the older scheme still supported but not recommended for new applications.

Mathematical model

In complexity-theoretic cryptography, security against adaptive chosen-ciphertext attacks is commonly modeled using ciphertext indistinguishability (IND-CCA2).