Triple DES revisited

Chris Mitchell
www.chrismitchell.net

Agenda

1. DES – a brief history
2. Double and triple DES
3. The Merkle-Hellman attack
4. The van Oorschot-Wiener attack
5. Generalising the van Oorschot-Wiener attack
6. Other issues
7. Concluding matters

Agenda

1. DES – a brief history
2. Double and triple DES
3. The Merkle-Hellman attack
4. The van Oorschot-Wiener attack
5. Generalising the van Oorschot-Wiener attack
6. Other issues
7. Concluding matters

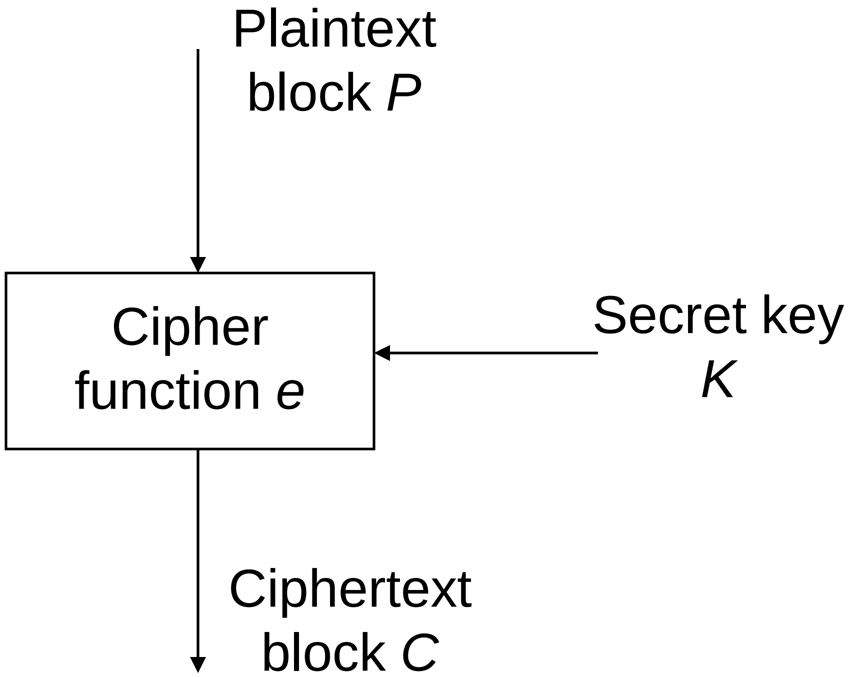
DES

- The DES (Data Encryption Standard) is a 64-bit block cipher, first published as a US federal standard in 1977 (NBS FIPS PUB 46).
- It was chosen as the result of a competition for a standard cipher.
- DES is a refined version of an IBM submission to the competition.

Block ciphers

- A block cipher is a very widely used type of cipher.
- A block cipher encrypts data a block (e.g. 64 or 128 bits) at a time.
- A well-designed block cipher is a very powerful tool – it has many uses (beyond just data encryption).
- The block length is vital for security – must be 64, or preferably 128, bits long (or more).

Block cipher – definition



Plaintext block *P*

Cipher function *e*

Secret key *K*

Ciphertext block *C*

Block ciphers

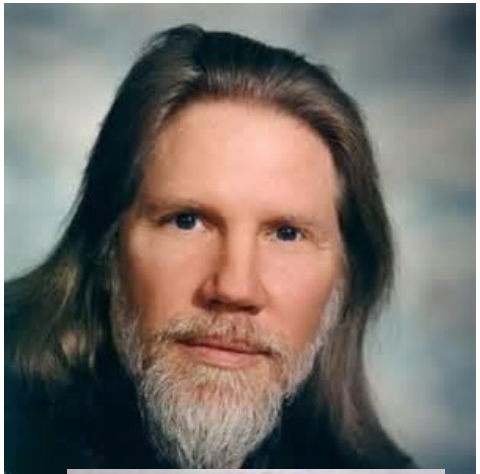
- For encryption we write: $C = e_K(P)$, where P is the plaintext block, K is the secret key, and C is the ciphertext block.
- We must also have a decryption function d which satisfies $P = d_K(C)$.
- The block size n needs to be reasonably large (e.g. $n \geq 64$) to prevent dictionary attacks.
- DES has n=64, which is why it is called a 64-bit block cipher.

Adoption

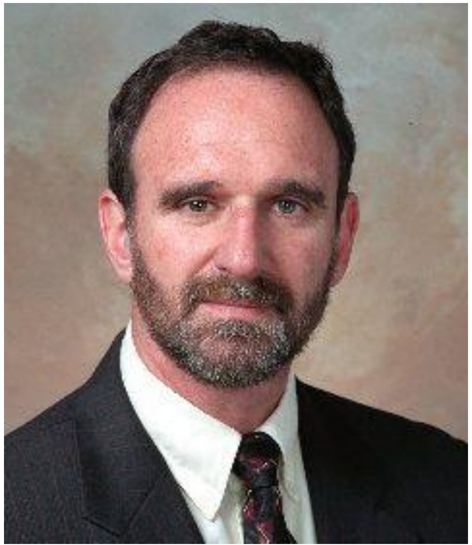
- DES was originally intended for use by the federal government.
- However, it was adopted much more widely:
  - ANSI made it a US standard (ANSI X3.92);
  - it was widely adopted for retail banking security internationally;
  - for a number of years it was the only prominent standardised cipher.

DES and 56-bit keys

- From the beginning, there was heavy criticism of its short key length (56 bits).
- That is, even in 1977, 2^{56} trials, as necessary to do a brute force search for the key using a known plaintext/ciphertext pair, seemed just about possible.
- In 1977, Whit Diffie and Martin Hellman published a very critical paper, sketching the design of a device which they claimed could find a key in a day and could be built at a cost of around $10 million.



Diffie and Hellman



Breaking DES in software

- It was some 20 years before breaking DES became a reality, at least in public.
- In June 1998, a 3-month distributed search organised by the DESCHALL project found the DES key for a 'challenge' plaintext-ciphertext pair.
- More recent, similar, efforts have completed much more quickly.

Breaking DES in hardware

- A few months after the DESCHALL break, the Electronic Frontier Foundation (EFF) announced the completion and successful use of *Deep Crack*.
- Deep Crack was a special-purpose hardware device designed to do brute force DES key searches, a complete search taking around a week.
- The claimed cost was less than $250,000.
- Similar, but cheaper and faster, machines have since been designed.

The end of single DES

- By 1998, the use of single DES was already widely seen as insecure, and the software and hardware breaks confirmed this.
- The breaks accelerated the replacement of DES by others schemes, notably by triple DES (three iterations of DES using at least two different keys).
- Triple DES forms the main focus of this talk.

The success of DES

- Despite issues with the key length, the design of DES has been a great success.
- It was clearly designed with great care, using understanding of design and cryptanalysis principles only rediscovered (sometimes decades) later.
- Whilst attacks are known which are ‘in theory’ slightly faster than the 2^{56} brute force search, in practice brute force is still the most effective way to break DES.
- This is a huge compliment for a 40-year old design.

Agenda

1. DES – a brief history
2. Double and triple DES
3. The Merkle-Hellman attack
4. The van Oorschot-Wiener attack
5. Generalising the van Oorschot-Wiener attack
6. Other issues
7. Concluding matters

Multiple iterations

- The idea of using multiple iterations of DES using more than one key has been around since the 1970s.
- The idea is mentioned in the 1977 Diffie-Hellman paper.
- This is an ‘obvious’ way of increasing the effective key length for a cipher.
- It also allows simple upgrades to existing systems (no new cipher to implement).

Why not double DES?

- The most obvious approach is simply to encrypt twice, using two distinct keys.
- However, this is not much more secure than single DES because there is a simple meet-in-the-middle attack on double DES.
- This attack was known back in the 1970s, and is outlined by Diffie and Hellman in their 1977 paper.

Meet-in-the-middle I

- Suppose we have a plaintext-ciphertext pair (P, C); then we know $C = e_{K_2}(e_{K_1}(P))$, where K_1 and K_2 are DES keys.
  1. Make a table of the values of $e_L(P)$ for every possible key L, which is sorted or hashed for easy searching (costs 2^{56} DES encryptions). Each table entry contains $e_L(P)$ and L.
  2. Go through all the possible DES keys again, and for each key M compute $d_M(C)$ and check if it is in the table. If it is, then the corresponding value of L, together with M, are a candidate for (K_1,K_2). Check every candidate using one more plaintext-ciphertext pair.

Meet-in-the-middle II

- Candidates will arise for one value of *M* in every 2^8=256 instances of step 2, and so the cost of checking is dwarfed by the other costs of the scheme.
- The total attack cost is 2^{57} DES encryptions (just twice as many as for a single DES brute force).
- The main extra cost will be for the table, which has 2^{56} entries, each containing 15 bytes.
- Even today, this is non-trivial, but attack trade-offs can be achieved to reduce the storage cost while correspondingly increasing the computations.

Triple DES and E-D-E

- Because of the meet-in-the-middle attack, at least three iterations of DES is the minimum effective multiple-iteration version of DES.
- In practice, instead of three encryptions, the ‘standard’ approach is to first encrypt, then decrypt, and then encrypt again.
- That is, $C = e_{K_3}(d_{K_2}(e_{K_1}(P)))$, where K_1 , K_2 and K_3 are DES keys.
- This is backwards-compatible with single DES if $K_1 = K_2 = K_3$ – this greatly simplifies migration.

2-key triple DES

- If K_1 , K_2 and K_3 are all independently chosen, then this is known as 3-key triple DES.
- However, in the late 1970s a variant in which $K_1 = K_3$ was proposed.
- This is known widely as 2-key triple DES.
- The 2-key version has the advantage of a shorter key, but still offers greater security than double DES (the simple meet-in-the-middle no longer works).

Triple DES standards

- Triple DES (both variants) has been widely standardised, both in the US by NIST and ANSI, and also internationally in ISO/IEC 18033-3.
- Both 2-key and 3-key triple DES remain in wide use today.
- Triple DES is also an industry standard, e.g. in the EMV specifications and in ISO banking standards, and so **2-key triple DES is probably implemented in credit and debit cards in your wallet**.

Agenda

1. DES – a brief history
2. Double and triple DES
3. **The Merkle-Hellman attack**
4. The van Oorschot-Wiener attack
5. Generalising the van Oorschot-Wiener attack
6. Other issues
7. Concluding matters

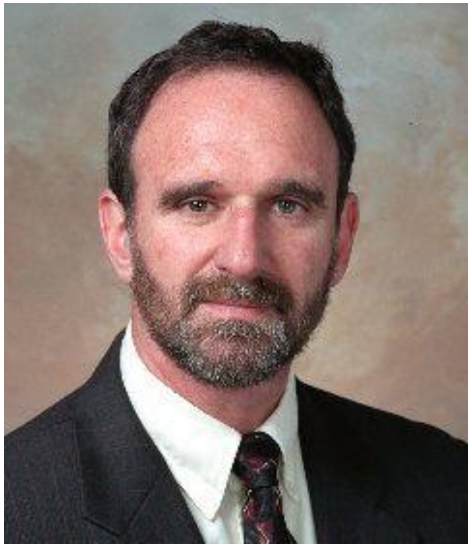
Key lengths and security

- Neither 2-key nor 3-key triple DES are as secure as one might expect from their key lengths.
- That is, in an ideal world, the most effective attacks against a cipher with a *k*-bit key would be a size 2^k brute force search (or one of the brute force time-space trade-off attacks with product complexity 2^k.
- In such a case a cipher is said to offer *k* bits of security.
- However, neither 2-key nor 3-key triple DES offer as many as 112 (or 168) bits of security.
- Big question: 'How many bits of security do they offer?'

Early doubts …

- In 1981, Merkle and Hellman described a **certificational** attack against 2-key triple DES which they suggested meant it should not be used.
- They claimed that their attack, whilst unrealistic (hence certificational), showed that 2-key triple DES was not much more secure than double DES.
- However, this did not stop widespread use of the 2-key variant.

Merkle and Hellman



Attack requirements

- As before, we suppose 2-key triple DES operates as: $C = e_{K_1}(d_{K_2}(e_{K_1}(P)))$, where K_1 and K_2 are DES keys.
- The attacker needs to be able to get chosen plaintexts encrypted using the genuine triple DES key (i.e. the genuine pair of DES keys).
- That is, it is a **chosen plaintext** attack.
- In fact, the attacker needs the ciphertext for as many as **2^{56} chosen plaintexts**.

Attack idea I

- As described in the 1981 paper, a simple brute force attack requires going through all possibilities for K_1, and for each such possibility, checking all possible value for K_2.
- That is, the attack complexity is $2^{56} \times 2^{56} = 2^{112}$.
- However, if there was a way to check K_2 quickly independently of the choice of K_1, then the attack complexity would go down to $O(2^{56})$.

Attack idea II

- Merkle and Hellman also noted that, if the attacker knew $A = e_{K_1}(P)$ as well as P and C, then (A,C) would essentially be a known plaintext-ciphertext pair for double DES, and the double DES attack could be used.
- This led them to the attack in which they choose a possible A, and make sure that $A = e_{K_1}(P)$ for **one** of a set of available plaintext-ciphertext pairs.
- They just don't know which one …

Attack operation

- The attack operates as follows:
  1. The attacker chooses a 64-bit value A (which can be anything) and computes $P_L=d_L(A)$ **for every DES key L**.
  2. The attacker now obtains the triple DES encryption of P_L for every L – call the result C_L – and for each such C_L then computes $d_L(C_L)$ – call this B_L.
  3. The values (B_L, L) are tabulated, sorted or hashed on the values of B_L for easy searching.
  4. For every possible DES key M, the attacker computes $d_M(A)$ and looks it up in the table; if there is a match, then the pair (L, M) is a candidate for (K_1,K_2), and can be checked using another plaintext /ciphertext pair.

Complexity

- The attack complexity very closely resembles that of the meet-in-the-middle attack on double DES.
- The attacker has to perform 2^{57} DES calculations, and a table is needed containing 2^{56} entries, each of 15 bytes.
- The ‘only’ extra is the need for the ciphertexts for 2^{56} chosen plaintexts, which of course makes the attack completely unrealistic.
- However it is interesting and worrying that the attack complexity looks like only $O(2^{56})$.

Agenda

1. DES – a brief history
2. Double and triple DES
3. The Merkle-Hellman attack
4. The van Oorschot-Wiener attack
5. Generalising the van Oorschot-Wiener attack
6. Other issues
7. Concluding matters

A more realistic attack

- The Merkle-Hellman attack, although interesting, did not pose a serious threat to 2-key triple DES, which was rapidly adopted.
- However, almost ten years after Merkle-Hellman, in 1990 van Oorschot and Wiener described an attack (vOW) which only requires **known** plaintext-ciphertext pairs.
- The idea is rather similar to that of the Merkle-Hellman attack.

van Oorschot and Wiener

Attack idea

- Their idea is to obtain a large-ish set of known plaintext-ciphertext pairs (P,C), choose an A, and hope that by random chance $A = e_{K_1}(P)$ for at least one of the values P.
- If the attacker is lucky, then the Merkle-Hellman attack applies.
- If the attacker is unlucky, then try with another value of A, and go on until he/she gets lucky.

Attack requirements

- The attack requires a set of matching known plaintext-ciphertext pairs (P,C), the more the better!
- To simplify complexity calculations we suppose the attacker has 2^t pairs, for some t.
- The attacker keeps the 2^t pairs (P,C) in Table 1, sorted or hashed on P for easy searching.
- The attack operates in a series of **phases** where, in each phase, the probability of successfully finding the triple DES key (K_1,K_2) is approximately $1/2^{64-t}$.
- That is, the attack will require around 2^{64-t} phases to be performed before the key is found.

Attack operation

- One phase of the attack operates as follows:
  1. The attacker chooses a 64-bit value A (which can be anything) and computes $P_L=d_L(A)$ for every DES key L.
  2. If P_L = one of the P values in Table 1, then the attacker computes $B_L= d_L(C)$ for the corresponding value of C from Table 1.
  3. The values (B_L, L) are tabulated in Table 2, sorted or hashed on the values of B_L for easy searching.
  4. Once Table 2 is complete, the attacker computes $d_M(A)$ for every possible DES key M, and looks it up in the table; if there is a match, then the pair (L, M) is a candidate for (K_1,K_2), and can be checked using another plaintext /ciphertext pair.

Complexity

- As mentioned previously, the chances of one phase successfully finding the key is $1/2^{64-t}$. So $O(2^{64-t})$ attack phases will need to be performed.
- A phase involves 2^{57} DES calculations, and Table 1 contains 2^{t} entries, each of 16 bytes. Table 2 is much smaller than Table 1 so can be ignored.
- That is, the attack complexity is:

$$(\#\text{ of phases})\times(\text{cost of one phase}) = 2^{64-t}\times 2^{57} = 2^{121-t}\text{ DES calculations}$$

with storage only as necessary to store the known plaintext/ciphertext pairs.

Implications

- If the attacker has as many as 2^{32} known plaintext-ciphertext pairs, this means that the attack complexity is 2^{89} DES computations.
- This is large, but not really large enough.
- Of course, getting 2^{32} known plaintext-ciphertext pairs all created using the same key is unlikely, but …
- This fact has led to pressure to move away from 2-key triple DES.

NIST and de-standardisation

- Indeed, in late 2015 NIST announced that it could no longer support continued use of 2-key triple DES, recommending a move to either 3-key triple DES or a newer and more secure algorithm such as AES.
- This is in line with previous announcements.
- NIST has always stated that 2-key triple DES should be regarded as giving only 80 bits of security.

The ISO/IEC response

- ISO/IEC 18033-3:2010 (a standard devoted to block ciphers) gives both 2-key and 3-key triple DES, and there are no current plans to withdraw support for the 2-key version.
- However, an ISO 'standing document' on key lengths states that (for 2-key triple DES):
  - 'depending on the required security level, the maximum number of plaintexts encrypted under a single key should be limited'; and
  - 'the effective key-length of two-key Triple-DES in specific applications can only be regarded as 80 bits (instead of 112 bits)'.

A lack of clarity?

- That is, there is a lack of consistency in the message from standards bodies.
- NIST says stop using the scheme, whereas ISO/IEC still says ‘use with care’.
- The most obvious conclusions are that:
  - the scheme is probably safe if you keep changing the key regularly; and
  - ‘80 bits’ seems like a safely conservative lower bound for the security of 2-key triple DES.
- In the remainder of this talk we challenge these assumptions.

Agenda

1. DES – a brief history
2. Double and triple DES
3. The Merkle-Hellman attack
4. The van Oorschot-Wiener attack
5. Generalising the van Oorschot-Wiener attack
6. Other issues
7. Concluding matters

An observation

- An apparently novel observation is that the vOW attack still works even if the plaintext-ciphertext pairs have not all been generated using the same key.
- In the attack, each plaintext/ciphertext pair is used independently of all the others, except when checking candidate key pairs.
- Checking can be done as long as the attacker knows which plaintext-ciphertext pairs ‘belong together’, i.e. have been created using the same key.

Generalising the attack

- In the scenario where the plaintext-ciphertext pairs have been created using a range of keys, the attack works with one minor modification.
- In Tables 1 and 2, a label needs to be kept with each entry, indicating which key has been used (to enable checking of candidate keys).

Complexity

- The attack complexity is identical to the regular vOW attack, except the two tables are slightly larger.
- That is, if 2^t known plaintext-ciphertext pairs are available, even generated with many different keys, one of the keys can be found in 2^{121-t} DES operations.
- The possibility that as many as 2^{32} pairs are available in this scenario seems much more plausible than in the single key scenario.

Implications

- This means that the ISO/IEC advice:

  > … depending on the required security level, the maximum number of plaintexts encrypted under a single key should be limited …

  has limited value!

- Of course, it is always good to change keys regularly, but changing keys will not prevent the attack.

A further generalisation

- The DES complementation property is well known:
  - if, for a plaintext P and key K, we have:

    $C = e_K(P)$

    then: $C^* = e_{K^*}(P^*)$, where the * simply indicates that the block has been complemented, i.e. every one has been changed to a zero and vice versa.
- Hence if (P,C) is a known plaintext-ciphertext pair for the key K, then (P^*,C^*) is a known plaintext-ciphertext pair for the key K^*.

Implications

- The fact that the key is different in the complemented pair does not matter, from our previous observation.
- This means we get two plaintext-ciphertext pairs to use in the attack from every pair.
- This means that the overall attack complexity reduces to 2^{120-t} DES computations.

Using partially known plaintext

- In ‘real life’, it is often the case that ciphertext will be available for which only partial information about the plaintext is known.
- For example, we might know 56 out of the 64 plaintext bits for a 64-bit ciphertext block, but not the other eight.
- Such information cannot be used in the ‘vanilla’ vOW attack.

Modifying the attack

- We build again on the observation that the attack treats each plaintext-ciphertext pair independently.
- We can generate a set of all possible plaintext-ciphertext pairs consistent with a partially known pair.
- As long as enough partial information is available (e.g. 48 out of 64 bits), surprisingly this does not affect the overall computational complexity (although it does increase the storage complexity).

Implications I

- Suppose have 2^t known plaintext-ciphertext pairs, where some of the plaintext blocks may not be completely known, and the pairs may have been generated using multiple keys.
- We can discover one of the keys with 2^{120-t} DES computations.
- If t=40, then this means we can find a key pair in only 2^{80} DES computations.

Implications II

- The ISO statements:
  - ‘depending on the required security level, the maximum number of plaintexts encrypted under a single key should be limited’; and
  - ‘the effective key-length of two-key Triple-DES in specific applications can only be regarded as 80 bits (instead of 112 bits)’.

  both now look very shaky.
- Whilst 2-key triple DES still has 80 bits of security, this is no longer a conservative estimate with a margin of error.

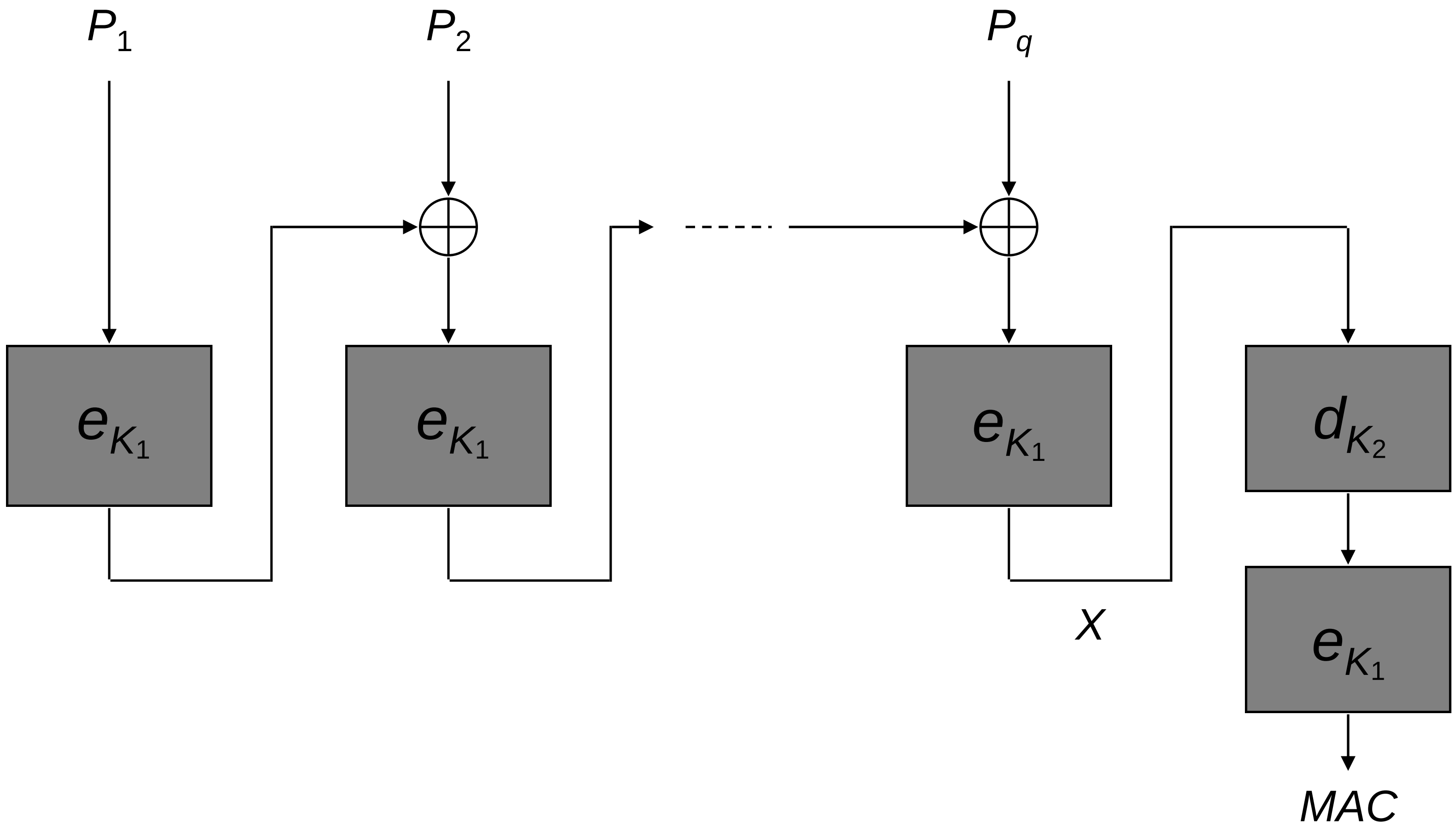
Agenda

1. DES – a brief history
2. Double and triple DES
3. The Merkle-Hellman attack
4. The van Oorschot-Wiener attack
5. Generalising the van Oorschot-Wiener attack
6. Other issues
7. Concluding matters

ANSI retail MAC

- A ‘triple DES’ type construction is also widely used to compute Message Authentication Codes (MACs).
- The ANSI retail MAC is a CBC-MAC construction, i.e. it use Cipher Block Chaining to compute a MAC.
- Single DES is used in CBC mode to process all but the last block, and the last block is then triple DES encrypted.

ANSI retail MAC calculation



Known attacks

- A number of authors have described attacks against the ANSI Retail MAC.
- Probably the most important, due to Preneel and van Oorschot (1996), relies on the simple observation that if two messages give the same MAC, then the values of X (shown on the previous slide) will also be the same.
- However, X is a function purely of K_1, i.e. this allows a single DES brute force search for K_1.
- If K_1 is known, then K_2 can be found with another single DES brute force attack.

Preneel-van Oorschot operation

- How likely is it that two messages will give the same MAC?
- Well, given that the MAC is 64 bits long, standard probabilistic arguments say that if 2^{32} message-MAC pairs are available, then the chances are better than 50% (with the probability near 1 if 2^{33} or more pairs are available).
- That is, if 2^{32} message-MAC pairs are available, then the ANSI retail MAC can be broken (key recovery) in $O(2^{56})$ DES operations.

Applying van Oorschot-Wiener

- The ‘standard’ version can be applied, but is less efficient than Preneel-van Oorschot.
- However, we can apply vOW even if the known (message, MAC) pairs are generated using multiple keys.
- Using similar arguments to before, suppose have 2^t known (message, MAC) pairs, where the pairs may have been generated using multiple keys. We can discover one of the key pairs with 2^{120-t} DES computations.

Implications

- If many large sets of message-MAC pairs are known, but each set is smaller than 2^{32}, then Preneel-van Oorschot does not work.
- However, this restriction does not apply to vOW.
- Also, the more pairs that are available, the smaller the attack complexity – this is not true for Preneel-van Oorschot.
- Fully known (message, MAC) pairs are freely available in many MAC usage scenarios – in general, they are much easier to get than known plaintext for encryption.
- This makes the ANSI retail MAC look much weaker than previously thought.

Agenda

1. DES – a brief history
2. Double and triple DES
3. The Merkle-Hellman attack
4. The van Oorschot-Wiener attack
5. Generalising the van Oorschot-Wiener attack
6. Other issues
7. Concluding matters

The future of 2-key triple DES

- Our main finding is that, perhaps surprisingly, the van Oorschot-Wiener attack works in a multiple-key and partially-known-plaintext setting.
- If an attacker is sufficiently determined and gathers enough (partially) known plaintext-ciphertext pairs, at least some of the keys can be found.
- My personal conclusion is that use of 2-key triple DES should be phased out as soon as possible.

What about the ANSI retail MAC?

- Arguably the situation is even more serious for the DES-based ANSI retail MAC.
- Here, getting (message, MAC) pairs is simply a matter of eavesdropping.
- Again if an attacker gathers enough such pairs, at least some of the keys can be found.
- Thus the DES-based ANSI Retail MAC should also be phased out as soon as possible.

Sometimes it pays to go back …

- The most recent paper on the security of 2-key triple DES (prior to the work described in this talk) was published in 1990.
- The subject seemed ‘dead’.
- However, reviewing prior art revealed new attack variants which significantly weaken the practical security of 2-key triple DES.
- Sometimes it pays to not take established wisdom for granted …

For further information …

- C. J. Mitchell, ‘On the security of 2-key triple DES’, arXiv:1602.06229 [cs.CR], February 2016, 20 pages.

Thank you and questions?