Introdução à Criptografia

- Campos de estudo
- Algoritmos
- Aplicações e praticas de privacidade

Campos de estudo

- Criptologia
  - Criptografia
    - Encriptação
    - Desencriptação
    - Algoritmos
    - Chaves
  - Criptoanalise

Criptoanalise (ou quase)

Criptoanalise é o ramo da criptologia que se dedica a analise de sistemas de encriptação de modo que com um texto encriptado (ciphertext) posse se chegar ao texto basico original (plaintext) mesmo sem o conhecimento da chave (key).
Existem diversas tecnicas de criptoanalise conhecidas, sendo algumas delas:

Criptoanalise (ou quase)

brute force: Se tratando de algoritmos simetricos consiste em se tentar todo o espectro possivel de chaves que certo algoritmo suporta, por exemplo o algortimo DES usa chaves de 56 bits sendo 2^56 (72.057.594.037.927.936 chaves possiveis). Já com algoritmos assimetricos varia entre problemas como fatorisação de numeros inteiros ou no calculo de um logaritmo discreto, por exemplo o algoritmo RSA que usa o produto de 2 numeros primos, sendo o brute force se fatorar esse produto em seus 2 termos originais (o numero de chaves possiveis aumenta exponencialmente quanto maior (mais digitos) os primos tiverem.

Criptoanalise (ou quase)

ciphertext only: Nesse ataque se tenta chegar ao plaintext ou mesmo a key tendo se somente o ciphertext ou parte do mesmo. Algoritmos antigos manuais conhecidos como algoritmo de papel-e-caneta são extremamente sucetiveis a esse ataque, como por exemplo o algoritmo de Cesar. Todos os algoritmos modernos oferecem proteção contra esse ataque no entando alguns (WEP, versões antigas de PPtP) se mostraram vulneraveis a esse ataque.

Criptoanalise (ou quase)

known text: Consite em se ter acesso ao plaintext e ao ciphertext, e assim se deduzir a key.

chosen plaintext: Consiste no “atacante” tendo a capacidade de escolher arbitrariamente parte do plaintext ou mesmo todo ele e então analisando o ciphertext deduzir a key, algoritmos simetricos são sucetiveis a esse tipo do ataque. Sabe-se que qualquer algoritmo não sucetivel a esse ataque também não será a ataques de “known text” e “ciphertext only”

Criptoanalise (ou quase)

timing: nesse ataque se tenta deduzir a key apartir do tempo que levado para se executar um algoritmo, levando em conta que a execução de algoritmos assimetricos é extremamente dependente do tamanho da key. Apesar de parecer improvavel, diversas execuções em diferentes plaintext acabam por “vazar” informações estatisticas sobre a key.

Algoritmos

- Algoritmos de chave publica (algoritmos assimetricos)
- Algoritmos de chave privada (algoritmos simetricos)

Algoritmos de chave publica

- RSA
- Diffie-Hellman
- Elgamal

RSA

Criado por Ron Rivest, Adi Shamir e Len Adleman em 1977 no MIT, baseia-se basicamente na multiplicação entre dois numeros primos e depende da dificuldade em se fatorar o produto dessa multiplicação, por isso sua segurança depende imensamente do tamanho de chave escolhida. Composto de um sistema de criação de chaves, um algoritmo de encriptação e um algoritmo de desencriptação.
O algoritmo RSA é vulneravel a ataques “chosen plaintext” e “timing”.

Diffie-Hellman

Em 1976 Whitfield Diffie e Martin Hellman introduzem o conceito de criptografia assimetrica com o artigo *New Directions in Cryptography, e também um sistema de troca de chaves baseado no problema do logaritmo discreto, cuja dificuldade de computação é igual ou superior à fatoração do produto de numeros primos. Foi conhecido posteriormente que tal algoritmo havia sido criado algum anos antes pela inteligencia britanica.*

Contudo é um sistema anonimo, usado de base para outros metodos de autenticação.

O algoritmo é vulneravel a ataques do tipo "timing".

ElGamal

Taher Elgamal em 1984 introduziu o algoritmo que recebeu seu nome, também baseado num problema de logaritmo discreto usando as propriedades de corpos ciclicos finitos. O sistema em si é composto por 3 componentes, um gerador de chaves, um algoritmo de encriptação e um algoritmo de desencriptação

Vulneravel a ataques de “chosen ciphertext”

Algoritmos de chave privada

- DES
- AES
- BLOWFISH/TWOFISH

DES

Criado em 1971 Horst Feistel trabalhando para a IBM criou o algoritmo LUCIFER, que em 1976 foi adotado como padrão pelo governo americano após alguma modificações sob o nome de DES (Data Encryption Standard).

DES é um algoritmo de bloco de 64bits usando chaves de 64bits (56 reais os ultimos 8 são usados como verificadores da chave), baseado em redes Fiestel cada bloco pode ser cifrado ou individualmente (eletronic code book ou ECB), ou tornando os blocos dependentes entre si (chain block coding e cipher feedback, CBC e CFB respectivamente) atravez de operaçoes XOR.

DES

O DES foi quebrado atravez de um ataque “brute force” em 1997 pelo grupo DESCHALL e em 1998 o EFF quebra uma chave em 56 horas. O FIPS republica o DES especificando uso preferencial ao 3DES (basicamente o mesmo algoritmo usando 3 chaves de 56bits em 3 ciclos encriptacao, desencriptação com a segunda chave, e uma segunda encriptação).
O DES é considerado inseguro, o 3DES é vulneravel a ataques “chosen plaintext” e “know plaintext”

AES

O algoritmo Rijndael surgiu em 1998 criado por Vincent Rijmen e Joan Daemen, consistindo de uma cifra de blocos baseado em uma rede de permutação em blocos de 128, 160, 192, 224, e 256 bits e chaves de 128, 160, 192, 224, e 256 bits, sendo submetido ao National Institute of Standards and Technology com o objetivo de ser aceito como padrão do governo americano em sucessão ao DES. Em 2001 ao final do processo de seleção foi escolhido entre 12 algoritmos como padrão sob o nome de AES e somente com blocos de 128 bits e chaves de 128, 192 e 256 bits. O algoritmo é baseado em um trabalho anterior de Rijmen e Daemen chamado Square, que por sua vez é derivado do algoritmo Shark também de ambos.

AES

Os blocos consistem de matrizes de 4x4 bytes (blocos de Rijndael com mais de 128bits usam matrizes maiores). As chaves de cada iteração são calculadas em operações de campo finito (a maioria das operações dentro desse algoritmo são feitas dessa forma).

Cada iteração (com excessão da ultima) consiste em 4 etapas, primeiro cada byte da matriz é substituido em uma S-Box, então cada linha da matriz é deslocada N posições, em seguida as colunas são substituidas numa operação de campo finito (com excessão da ultima iteração) e então é aplicada a chave da iteração a matriz resultante. Este processo é repitido 10, 12 e 14 vezes dependendo do tamanho da chave utilizada (128, 192, 256).

AES

Não existem ataques efetivos conhecidos contra o AES, em 2002 um ataque teorico conhecido como “XLT attack” foi proposto por Nicolas Courtois porém estudos consequentes não reproduziram os termos de Courtois, ataques “XLT” são considerados especulativos e nunca foram reproduzi-los, em Abril de 2005 Daniel J. Bernstein propos um ataque chamado “cached timing”, que devido a impraticidade de reprodução (foram usando 200 milhões de “chosen plaintexts”) foi considerado impraticavel.
O governo americano considera AES como utilizavel em proteção de dados considerados secretos.

BLOWFISH/TWOFISH

Blowfish é um algoritmo criado em 1993 por Bruce Schneier, criado como um algoritmo de uso geral e como substitudo do DES, Blowfish é livre de patentes e sua implementeção é totalmente publica (fato raro na época da sua criação). É um algoritmo de bloco de 64 bits com chave variante de 32 até 448 bits em degraus de 128 bits com usod e redes de Feistel e faz uso de S-boxes dependentes da chave. O sistema consiste de 16 ciclos atravez de um sistema de 4 S-boxes de 8 bits com uma gerando 32 cada sendo os resuldados sendo somados em com modulus 2^32 e passam por um xor ao final, cada ciclos tem um chave distinta gerada apartir da chave inicial.

BLOWFISH/TWOFISH

Twofish é um algoritmo baseado em blowfish do mesmo autor publicado em 1998 e inscrito no AES contest ficando como um dos 5 finalistas, é uma cifra de bloco de 128 bits e com chaves de até 256 bits e faz uso também de Pseudo-Hadamard transforms onde strings de bytes (precisar estar pareadas) são divididas e computadas entre si, processo também usado no algoritmo SAFER.

BLOWFISH/TWOFISH

Não existem ataques bem sucedidos a nenhum desses dois algoritmos, em 1996 Serge Vaudenay encontrou um known-plaintext attack com 2^(8r+1) plaintexts necessarios para se quebrar o Blowfish e também uma serie de chaves fracas que podiam ser detectadas e quebradas em 2^(4r+1) plaintexts com o mesmo ataque, onde r é igual ao numero de ciclos completos, esse ataque não é efetivo contra os 16 ciclos completos.
Sabe-se que devido ao tamanho do bloco no algoritmo Blowfish pode-se “vazar” informação da chave quando se encripta mais do que 2^32 blocos, portanto esse algoritmo não é recomendado para encriptação de grandes quantidades de dados.

Aplicações e praticas de privacidade

- Onde é necessaria
- Qual algoritmo usar
- Qual tamanho de chave usar
- Praticas seguras
- Web of trust e entidades certificadoras

Onde é necessaria

- Criptografia tem um custo em processamento e tamanho dos dados que vão trafegar.
- Dados interceptados podem ser usados em ataques de ciphertext, timing, ou brute force entre outras técnias.
- Dados confidenciais devem ser confidenciais

Qual Algoritmo usar

A escolha do algoritmo deve levar em conta a importancia dos dados, e o quanto se pode gastar computacionalmente com eles, não adianta ter um sistema seguro que leva 3 dias para executar qualquer tarefa.

Qual tamanho de chave usar

A politica de tamanho de chaves é tão importante quanto o algoritmo, o sistemas simetricos em geral fazem um “pad” das chaves pequenas oque diminue a confiabilidade das mesmas e descartam bits em excesso, em sistemas assimetricos chaves grandes dimuniem a eficiencia e aumentam o custo de encriptação e chaves pequenas são obviamente inseguras.

Praticas seguras

- Certificados de revogação.
- Chaves secretas são secretas.
- Verificação de identidade em chaves publicas e certificados.

Web of trust, entidades certificadoras

Web of trust é uma rede onde usuarios se regulam, determinando niveis de confiança entre si (atraves de assinaturas digitais em chaves publicas).

Entidades certificadoras são organizações que regulam a confiança em certificados (chaves publicas assinadas por essas entidades)

IN GOOGLE WE TRUST