RESUMO

Apresentam-se, nesta comunicação, os princípios básicos que suportam as modernas técnicas criptográficas. Descrevem-se com mais detalhe os sistemas DES (Data Encryption Standard) e PKS (public Key System).

Finalmente enunciam-se algumas aplicações destes sistemas que visam a segurança e autentificação da informação em redes de comunicação e bancos de dados.

1. INTRODUÇÃO

1.1 importância actual da criptografia

O desenvolvimento tecnológico no campo da electrónica digital deu um novo impacto a criptografia, saindo das aplicações de natureza puramente militar ou diplomática a que se restringia até aos finais da última guerra mundial, a criptografia de dados estende-se agora irreversivelmente na direcção dos mais variados campos tais como a transmissão digital, comunicações via-satélite, transmissão segura de voz por canal telefónico, manipulação de dados por computador, armazenamento de dados em ficheiros, etc..

Todo este novo mercado exige a definição de sistemas criptográficos a escala nacional. Mencionaremos dois: o DES (Data Encryption Standard) e PKS (Public KeySystem) respectivamente nos capítulos 2 e 3 deste artigo.

Previamente, no entanto, impunha-se definir conceitos tais como privacidade e autenticidade, critérios de segurança realistas etc.. É o que faremos no capitulo 1.

Concluiremos por uma breve referência a aplicações no capitulo 4

1.2 Taxonomia

A figura 1 ilustra o fluxo de informação num sistema criptográfico. O texto gerado pelo emissor devera ser comunicado através de um canal a um dado destinatário. A possibilidade de acesso a essa informação da parte de um receptor ilegítimo leva ao processamento do texto em claro por um algoritmo de cifragem, produzindo-se assim o texto cifrado ou criptograma que será transmitido em seu lugar. O algoritmo em questão e o correspondente algoritmo de cifragem é seleccionado de uma família de algoritmos estruturalmente idênticos por intermédio de um dado parâmetro que designaremos por chave.

Criptoanálise será precisamente a tentativa de cifragem ou decifragem realizada por um receptor ilegítimo.

O criptoanalista podara restringir-se a um papel passivo de extracção de informação. Em certos casos, no entanto, a injecção de criptogramas anteriormente registados e eventualmente modificados é também possível.

A prevenção da primeira hipótese relaciona-se com o problema da privacidade e a da segunda com o da autenticidade, dado que o destinatário deve possuir meios de verificar a legitimidade do criptograma.

Por outras palavras, pretende-se o equivalente de uma assinatura com a respectiva data num sistema criptográfico. Esta questão relaciona-se, também, com o problema da distribuição de chave- pelo que será abordada quando estudarmos o segundo sistema proposto.

1.3Criptanalise

1.3..1 Tipos de Ataque

o criptanalista que dispõe apenas de um texto cifrado e desconhece a chave ou o próprio sistema, poderá

tentar a decifragem com base numa analise de frequência, i.e., numa estimativa das letras ou palavras mais prováveis que poderia ser, por exemplo, as formulas normalmente usadas no inicio e fim de uma carta. O sistema que sucumbir a este tipo de ataque - ataque com criptograma apenas - pode ser considerado inseguro.

Por sua vez, o criptanalista que tem alguma informação a priori sobre o texto emclaro, poderá mais facilmente detectar a correspondência entre este e o texto cifrado –é o caso, por exemplo da cifragem de programas em linguagens avançadas. Este tipo de ataque com conhecimento de um texto em claro, é apenas uma extensão do primeiro caso descrito.
Finalmente, o chamado críptanalista activo pode injectar qualquer tipo de mensagens e obter o correspondente criptograma. Este é o tipo de ataque mais forte e em relação ao qual, importa testar qualquer Sistema criptográfico - ataque com escolha de texto em claro.

1.3.2Critério de Segurança

Não estamos interessados aqui, em definir um critério de segurança absoluto. De facto, o único sistema criptográfico seguro é o chamado - "ONE TIME TAPE" - em que, normalmente, O texto em claro é combinado com uma chave aleatória, usada uma única vez. Optemos, pois, por um critério de segurança relativo. Para o definir atendamos a que, hoje em dia, com a evolução vertiginosa dos computadores, a tentativa sistemática de todas as chaves por parte de um criptanalista com conhecimento da estrutura básica do sistema não pode ser posta de lado, por maior que seja a complexidade do sistema. Consideraremos, pois, como como "computacionalmente seguro" um sistema cuja ruptura exigiria um esforço computacional demasiado exaustivo para o objecto em questão.

1.4 Configuração de Sistemas

Embora extensível a qualquer tipo de sistema criptografico de dados, as configurações a seguir indicadas aplicam-se, em particular ao sistema DES. Mencionaremos três tipos: a cifragem em blocos, a sequencial e a mista.

1.4.1 Cifragem em Blocos

Nesta configuração os bits de entrada são divididos em blocos de dimensão fixa que serão tratados independentemente um
Dos outros. É um sistema em que cada bit de saída é uma função de todos os bites do bloco de entrada ("data dependemt"). Como tal a propagação de erros é considerável mas limitada a cada bloco. A sua principal desvantagem e a possibilidade de idênticos blocos de entrada produzirem criptogramas iguais, o que facilita a tarefa do criptanalista. Uma variante deste tipo de fórmula, designado por cifragem em bloco com realimentação (fig. 2a) ultrapassa este problema somando (mod. 2) cada bloco com o próximo bloco de entrada. 'É um sistema com memória, dado que a saída é função, tanto de entrada como de estado. É também "data dependent", com o inconveniente adicional de um único erro na entrada se propagar independentemente por todos os blocos posteriores.

1.4.2 cifragem Sequencial

No segundo tipo de configurações, a saída é uma função Simultânea de entrada e do estado do sistema. Distinguem-se duas variantes:

a - Auto-Sincronizado (fig.2-b)

Neste formato, os bits de entrada são somados (mod.2) com alguns dos bits cio saída anteriores, sendo o resultado realimentado para a entrada. E "data dependent" mas com uma memória finita, donde, a propagação de erros acaba ao fim de um certo número de bits, motivo porque se designa de auto-sincronizado.

b - Sincrono (fig. 2-c)

O papel do algoritmo de cifragem nesta configuração é de gerador de bits pseudo-aleatório. Isto, porque a saída e a soma bit-a-bit dos novos bits de entrada com uma corrente de bits que e permanentemente realimentada. O sistema é 'Data independent", uma vez que a realimentação não depende da entrada, i.e. cada estado é unicamente função do anterior.

1.4.3Cifragem Mista

Como o nome Indica, este terceiro tipo inclui sistemas emque existem combinações das configurações anteriores.

A escolha entre formatos do tipo "data dependent" ou "data independent" ditada pela aplicação em vista. Os primeiros apresentam uma grande vantagem do ponto de vista de resistência à criptanilise. No entanto, ha que optar por sistemas do tipo "independent" quando se está, por exemplo, a trabalhar com sinais de voz transmitidos através de canais ruidosos

Outro factor a considerar nesta escolha é a velocidade de cifragem, e, qual depende do número de bits realimentados: com uma realimentação de 8 bits, por exemplo, a velocidade atingida é de 1/8 da velocidade de cifragem em blocos.

Esta ligação pode ser feita fundamentalmente de três modos distintos, dependendo do tipo de aplicação

a - "Link-by-Link" (.Fig. 3-a)

O cifrador é colocado em série entre o transdutor e modem de modo que todo o trafego passara ainda na forma não cifrado através da unidade central de processamento de qualquer nó onde é susceptível de ser interceptado.

b - "Node-by-Node" (fig. 3-b)

Para ultrapassar a desvastagem referida, atribui-se a cada nó uma dada chave e a conversão da chave emissora para a receptora processa-se num módulo seguro.

e - "End-to-End"

A impractibilidade de um sistema como o anterior quando o número de utilizadores aumenta levou a uma nova técnica. Embora a cada terminal seja atribuida uma chave fixa, as mensagens são cifradas de acordo com uma chave temporária gerada pelo controlador central que a envia, a pedido do emissor, a este e ao receptor, cifrada nas respectivas chaves que sé ele conhece e que os dois interlocutores decifrarão.

2.SISTEMAS DES

O sistema cuja análise propomos em primeiro lugar, o DES, apareceu como resposta à necessidade de um sistema resistente ao criptanalista que dispondo de recursos computacionais elevadíssimos e não possuindo a chave conhece, no entanto, a estrutura Interna do algoritmo.

Adoptado pela NBS (National Bureau of Standars) em Janeiro de 1977 como sistema de cifragem de dados nacional (EUA) o DES não pretendia, dada a sua ampla divulgação, abranger os serviços militares ou de espionagem.

2.1 Estrutura Básica

A ideia que está na base do DES não e nova - de facto, os projectistas da IBM foram busca-la à Shannon que, já em 1944, sugeria a elaboração de um sistema "forte" à custa da associação de blocos criptograficamente simples.

Consideremos uma aplicação que, a um texto em claro de n bits faz corresponder um criptograma de n bits também. O resultado desta 'cifra-bloco" e uma de 2n configurações possíveis. O número total de transformações n bits - n bits é, de facto, de (2n)2n mas desta:. só 2n são transformações não singulares, sendo as restantes, regra geral irreversíveis. Na prática, um dispositivo capaz de realizar uma qualquer das transformações do primeiro tipo, tendo como entrada de selecção uma chave de n-bits, necessitaria de um número de blocos lógicos que aumenta de uma maneira critica com n, o que restringe a aplicação do sistema a um n reduzido, logo, criptograficamente pouco seguro.

Oconceito de "cifra produto" surge precisamente como uma aproximação do sistema ideal com o infinito e não é mais do que a aplicação sucessiva de cifragens muito simples. Um caso particular de cifra-produto é a classe de aplicações designadas por involuções. Por definição, a aplicação n é uma involução se n2= I em que I é a aplicação identidade.

Consideremos um bloco x constituído pelos subblocos

a aplicação que consiste nos seguintes pontos:

-aplicação da transformação T ao subbloco direito x2

-adição bit-a-bit (mod 2) do resultado com o subbloco esquerdo

-concatenação de x1 O T(x2) com o subbloco direito

oque significa que a decifragem se consegue passando o bloco cifrado pelos n sucessivos andares por ordem inversa, resultando deveras vantajoso. Este sistema de que representamos um dos andares na fig. 4, é de facto, a estrutura básica do DES.

2.2 Descrição sumaria do Algoritmo

Basicamente, o algoritmo consiste no processamento de um bloco de entrada de 64 bits através do 18 andares de manipulação do dados. Destes, 16 são estruturalmente idênticos, diferindo apenas nas 16 chaves geradas internamente a partir da chave activa de 55 bits. Os outros dois - as permutações inicial e final são fixas, não necessitando, portanto, da chave.

Da analise da figura 5-a e b ressaltamos seguintes factos:

-Os andares inicial e final são simples transposições inversas uma da outra.

-Os 64 bits trocados que se apresentam A entrada do andar de manipulação 1 vão ser divididos em dois blocos: o bloco esquerdo ko e o bloco direito R0.

-O bloco r0 sofre em seguida uma expansão de 32 bits para 48 bits, o que é representado pela caixa E.

Esta expansão processa-se de um modo muito simples que consiste em repetir os bits

1,4,5,6,9,12,..., 24,27,28,32.

isto e, os bits da ponta" de cada bloco de 4 bits. Os bits adicionais são postos de forma a formar blocos de 6 bits do seguinte modo:

-O bloco resultante vai ser adicionado (mod.2) com uma chave derivada-k1-de 48 bits também, e sofre em seguida uma permutação P

-Opera-se então uma nova divisão: desta vez em 8 blocos de 6 bits, cada um dos quais constitui a entrada de uma das 8 caixas 5. Nestas efectua-se uma substituição não-linear que produz uma saída de apenas 4 bits.

Esta compressão de 6 bits para 4 bits garante a não linearidade e é conseguida fazendo 4 dos bits de entrada determinarem uma coluna e os outros 2 uma linha de uma tabela (de 2 4=16 colunas e 2 2=4 linhas) que fornece para cada localização uma saída de 4 bits.

-O bloco resultante das 8 caixas S tem, portanto, 32 bits e vai ser novamente adicionado (mod.2) ao bloco esquerdo L0 que não tinha sido previamente manipulado.

-Processa-se finalmente uma troca de blocos: Ro passará a ser agora o bloco esquerdo da entrada do andar 2(L1) e o bloco resultante da soma será o bloco direito respectivo(R1).

Genericamente, podemos, portanto, representar as transformações sofridas através das eq.

L1= Ri-1

Ri = Li-1 e+ f(Ri~1, K1)

em que f sela uma função cuja saída tem 32 bits.

Tendo descrito o algoritmo de cifragem, vejamos como decifrar. O algoritmo respectivo é muito mais simples de derivado que poderia parecer à primeira vista e a razão fundamental esta no facto de que a função f não precisa de ser invertida.

Conclui-se pois, que os algoritmos de cifragem e decifragem são estruturalmente Idênticos como, aliás, se pretendia.

derivação das chaves internas (fig. 5-c)

apôs retirar os 8 bits de paridade da chave de entrada (64 bits) os 56 bits restantes constituem a chamada chave activa que gerará as 16 chaves derivadas do seguinte modo:

Em primeiro lugar a chave activa sofre uma permutação(P1) da qual resulta um bloco que vai constituir a entrada de dois registos de deslocamento de 28 bits dos quais 24 são consideradas saídas.

O conteúdo, de cada um destes registos vai sofrer um deslocamento a esquerda de 1 ou 2 bits em cada uma das 16 iterações.

Os dois juntos vão, para cada uma destas, sofrer uma permutação idêntica e o resultado é a chave derivada do andar respeitante à ordem de iteração.

O algoritmo apresentado deve a sua "força" à complexidade do processo global. As ideias básicas eram, como vimos, muito simples, mas desenvolver um algoritmo destes foi um trabalho exaustivo que teve em conta vários pontos que não ressaltam da análise breve que esboçamos. São eles, por exemplo, a escolha do número de andares de processamento, da estrutura e do número de bits dos blocos-S, do número de bits da chave, etc.

a - Atendamos ao primeiro facto considerando: do ponto de vista de hardware do terminal de processamento, poder-se-ia ter utilizado um numero infinitamente maior de andares intermédios. De facto é a velocidade de processamento e transmissão dos vários periféricos que impõe um limite máximo ao número de iterações. Este parâmetro controla a "mistura" dos bits de entrada. Verifica-se que um sistema com um único andar produziria uma saída fortemente correlacionada com alguns bits de entrada, dependência que praticamente desaparece com um mínimo de 8 andares. A escolha final, resultou, portanto, de uma solução de compromisso.

b - Os blocos-S são o ponto forte do algoritmo. No critério de projecto incluem-se varias especificações. Nomeadamente, devem ser não-lineares.

Ente ponto é fundamental dado que todas as restantes operações efectuadas em álgebra binária são lineares e as transposições também o são, pelo que a composição resultante seria linear e, por conseguinte, facilmente destrutível por um mini-computaclor em alguns segundos.

-devem ser adequadamente não-afins

Aplicações afins são susceptíveis de criptanalise e, portanto, devem ser evitadas. O limite inferior do número de bits de entrada (K) de cada bloco-S fica assim definida. Na realidade, K=2 daria origem a transformações afins e K=3 também em alguns casos. Porém, K=4 é já adequado.. Um número superior origina um sistema mais seguro e complexo.

-a alteração de um único bit de entrada deve implicar a alteração de pelo menos, dois bits de saída. Como, consequência, tem-se chamado efeito de avalanche, que consiste na propagação de erros da entrada para a saída quando se altera um único bit do bloco de entrada ou da chave. Esta propriedade do sistema dificulta tremendamente a criptanálise pois o conhecimento "aproximado" segundo um qualquer critério geométrico pouco adianta na procura da chave usada.

c – Relativamente à escolha do número de bits da chave, ela foi um ponto quente da controvérsia gerada em torno do DES. De facto, a tentativa sistemática de 256 possíveis chaves implica, de momento, um esforço computacional tão grande que não parece justificar o fim a atingir. Suportareis, por exemplo, um chip LSI capaz de implementar o teste de uma chave em 1us. Com um computador constitui do por um milhão do placas operando em paralelo, todo o espaço das chaves poderia ser testado em 10 segundos (cerca do um dia).

A estimativa de $20 milhões de dólares feita por Hellman não é universalmente aceite, mas dá uma ideia do esforço pré-computacional envolvido. Dentro de 3 a 10 anos, porém, ele poderá considerar-se realizável.

Por outro lado é sempre argumentável que o tamanho efectivo da chave pode ser duplicado através da cifragem repetida do mesmo bloco. De facto, o DES é frequentemente implementado de uma forma recorrente, o que dificulta o ataque exaustivo.

2.3 critica do DES

Os oponentes do DES criticam em primeiro lugar, a não publicação integral dos critérios de projecto nomeadamente no que se refere aos blocos S, P e derivação das chaves internas. Embora ainda se desconheça qualquer técnica criptanalitica para destruir o sistema, o facto é que há suspeita de várias propriedades dos blocos S não publicadas que facilitariam a criptanálise.

Outra critica frequente e que já focamos, é o numero relativamente pequeno de iterações e de bits da chave.

Aponta-se finalmente uma outra propriedade: existe de facto simetria entre os blocos cifrados correspondentes a textos em claro complementares. Explorando esta propriedade, a tarefa criptanalitica vem reduzida a metade.

Contra esta critica, que tem o projectista do DES para contrapôr?

E para já o “melhor" sistema capaz de ser implementado em hardware e continuará a sê-lo pelos próximos 5 a 10 anos, havendo sempre a possibilidade de aumentar a dimensão efectiva da chave.

Em segundo lugar, o DES resistiu até ao momento aos vários testes criptanaliticos a que foi sujeito:

-Sendo uma transformação de blocos de 64 bits em blocos da mesma dimensão, houve que testar se a aplicação sucessiva do mesmo algoritmo com chave idêntica conduziria mais tarde ou mais cedo à recuperação do texto em claro. Verificou-se que o comprimento médio de um "ciclo" é da ordem de 1018 que torna este ataque impraticável.

-Testes destinados à medida da correlação entre a entrada e a saída revelaram também uma "independência" satisfatória e, por outro lado, a medição da distancia de Hamming (aH) entre dois criptogramas correspondentes a textos em claro (ou chaves) diferindo num bit (aH=1) conduziu a valores do cerca de 50% do número total de bits.

-Recorreu-se também a uma representação Booleana das várias transformações efectuadas. Verificou-se, neste teste, que o numero de termos ao fim ao fim de duas iterações é demasiada elevado para que este ataque seja realizável. De facto, devido à complexa estrutura interna que mistura integralmente tanto os bits de entrada como os da chave nenhum dos últimos é utilizado menos de 12 vezes ou mais de 15 vezes, minimizando a possibilidade de reconhecimento de padrões), não foi ainda demonstrado como "quebrar" o algoritmo sem prévio conhecimento da chave.

Este é, de facto, um mérito indiscutível de um sistema cuja estrutura esta tão amplamente divulgado.

2.4implementação

Do ponto de vista hardware, a implementação do DES pode, apesar da sua complexidade, levar apenas cerca de 6 us para cifrar ou decifrar um bloco de 64 bits. Esta velocidade ultrapassa de longe a da execução em software.

Existem variadíssimas implementações do DES, cada qual adaptada a um fim especifico, algumas do tipo "stand-alone" outras que funcionarão apenas acopladas entre o modem e o terminal de dados ou o computador. Há também que mencionar a existência do hardware próprio para teste do sistema DES.

3.SISTEMAS DE CHAVE PÚBLICA (PKS)

Este tipo de sistemas pretende fundamentalmente resolver o problema da distribuição de chaves por um número potencialmente elevado de utilizadores.

Entre as várias implementações conhecidas focaremos duas:

Os PKC (Public Key Cryptosystems) e os PKDS (Publie Key distribution Systems)

3.1 Estrutura básica do Sistema

Tal como foi feito em relação ao DES começaremos por uma brevíssima referencia à teoria da complexidade computacional subjacente a este tipo de sistemas.

definiremos, em primeiro lugar, varias classes de problemas computacionais:
-a classe polinomial, P, à qual pertencem as funções f(x) para as quais existe um algoritmo que as calcula em tempo polinomial, i.e., inferior a um dado polinómio p(.x.),

-a classe exponencial EXP, cujo tempo de solução é inferior a uma dada exp {'p'(x) }

-e, finalmente, uma classe intermédia NP A qual pertencem as funções solúveis em tempo polinomial com computação nã deterministica usando paralelismo limitado. Uma subclasse destas, NP-completo, resolve qualquer dos problemas com base em modificações, em tempo polinomial, de um mesmo algoritmo.

Obviamente, verifica-se a relação P C.NP C NPC C EXP.

Vejamos qual a relação entre as duas classes mencionadas e os sistemas de chave pública. De facto, a ideia básica destes sistemas está na utilização de funções de dois tipos: as que são fáceis de calcular mas cuja inversão é difícil (on-way-functions) e as que são fáceis se for conhecido um dado parâmetro mas que se reduzem à classe anterior no caso contrário(trap door functions).

Quantificando as noções de fácil e difícil atras mencionadas, diremos que 'fácil" será um problema susceptível de ser resolvido em tempo polinomial e "difícil" o que não o é'.

Embora com a mesma estrutura básica que representamos na fig. 6 os sistemas de chave pública podem classificar-se em 2 tipos.

No primeiro, "Public Key Cryptosystemn", dispõe-se de duas famílias de algoritmos de cifragem e decifragem que designaremos por EK o {D K} e das quais a chave K(K) seleccionará um par correspondente tal que DK(EK(M)= M) sendo N uma mensagem qualquer.

Os projectistas deste sistema escolheram estas funções dentro da classe "Trap door functions". Deste modo, cada utilizador gera um par de transformações inversas EK e DK no seu terminal. A transformação DK é absolutamente privada enquanto que Ek e publicada num ficheiro juntamente com a identificação do utilizador. Qualquer outra pessoa poderá assim cifrar menagens e envia-las ao utilizador que é o único a poder decifra-las.

Sendo de mais difícil implementação que o segundo tipo de sistem as, este tipo é contudo vantajoso de um ponto de vista de autentificaçao, i.e., sempre que se pretende gerar uma assinatura que todos reconheçam como autêntica mas que apenas o signatário legitimo possa produzir. Suponhamos que o utilizador A deseja comunicar a B a mensagem M e a decifra com a sua chave secreta enviando DA(M). B, por sua vez, é capaz de cifrá-la com a chave pública de A, EA certificando-se assim da legitimidade de M. De modo a garantir privacidade, além de autentificaçao, o utilizador A poderá cifrar DA(M) com a chave pública de B, enviando EB(DA(M)) dado que B é o único a conhecer DB como exemplo deste tipo de sistemas mencionaremos o problema do "Knapsack".

No segundo tipo de sistemas (Public Key Distribution Systems) os dois processos de cifragem e decifragem são regulados. por uma única chave operativa que é acordada entre os dois utilizadores. Garante-se através da utilização de "trap door functions " na determinação da chave que o criptanalista, embora ouvindo toda a troca de inform ações, é incapaz de a calcular com base nestas. Como exemplo, abordaremos a variante baseada no problema logarítmico.

3.2Sistema PKC Baseado no Problema "Trap door Knapsack"

O problema clássico da analise combinatorial conhecido por "Knapsack" pode ser enunciado do seguinte modo: dado um número real S e um vector de dimensão n a, determine x tal que

S=a.x (produto interno)

o grau de dificuldade deste problema Depende crucialmente de a.

Se as componentes deste vector forem, por exe mplo,

x não é mais do que a representação binaria de S.

Embora mais difícil, mas ainda com grau de dificuldade P, é o problema em que

dado que x i = 1 se

Suponhamos então que a chave secreta de um dado utilizador A consiste no conhecimento das

componentes do vector a que dão origem a um problema 'Knapsack" relativamente fácil e ainda nos parâmetros w-1 em tais que

Um outro utilizador, B, cifrará a mensagem x recorrendo a chave publica de A que consiste no vector a:

S= a .x

e cuja decifração sem conhecimento da chave secreta é um problema de grau de dificuldade NP sendo por isso designado de "trap door Knapsack". A, por outro lado, resolve facilmente o problema calculando:

e resolve facilmente S' dado que é do tipo (a).

3.3Sistema PKDS Baseado no Problema logaritmo

Na base deste problema, esta o numero elevado de operações necessário para calcular x dado a, q (número primo suficientemente elevado) e
x= ax (mod q) (1)

Neste sistema, a chave é acordada entre o par de utilizadores i e j, sendo funções simultaneamente da chave secreta de cada um deles (xi e xj) e da chave que é publicada juntamente com a sua identificação (yi e yj, respectivamente)..

Ao pretender comunicar com j, o utilizador i constrói a chave operativa utilizando a sua chave secreta e a chave publicada de j:
x1 xj xi
K .= Yj (mod q) = a mod q (2)
lj

Por sua vez, e recorrendo também ao ficheiro público para determinar se o utilizador j calcula a chave por um processo idêntico
j xi xj
(mod q) = a (mod q)= K (3)
K ij= y ij

Conclui-se, portanto, que estamos perante um problema do tipo Mtrap door".

Para dar uma imagem do esforço criptanalitico: correspondente, referiremos um exemplo: para q da ordem de 2b o número de operações necessário ao calculado do logaritmo é de 25/2 o que, para b= 200, dá cerca de 10 operações.

4.APLICAÇÕES

Concretizaremos em três exemplos os campos de utilização da criptografia já mencionada na introdução.

-Em terminais bancários do tipo automático funcionando ininterruptamente, o uso de criptografia permite varias operações tais como depósitos, levantamentos e transferências entre contas. Nestes serviços, é vulgar a utilização de cartões magnéticos onde é gravado o número da conta ou adicionalmente, um número de identificação pessoal (NIP) que é introduzido por teclado no terminal de acesso de modo a impedir que alguém na posse de um cartão roubado tenha. acesso ao sistema. Entre o terminal e o computador central processa-se uma troca de instruções, i.e., a um pedido de transação o computador responde com uma ordem autorizando-a ou não Esta ordem deve ser dada na forma cifrada , para evitar que algum adversário possa grava-la e envia-la posteriormente ao terminal através da ligação a um outro computador, e a combinação da ordem com um dado parâmetro variável no tempo (o numero da transacção a efectuar, por exemplo) que é aplicada a cifra. O adversário não poderá assim construir uma mensagem semelhante.

-Também em cartões magnéticos de acesso a edifícios pode ser gravada a cifra (C) correspondente a um número de identificação pessoal (NIP) e a um número atribuído ao edifício (NIE)...

C(X,Y)= C(NIP, NIE-NIP)

Deste rodo, a entrada só será autorizada se depois de descodificada a mensagem se verificar

X+Y= NIE

pelo que a perda ou roubo de um único cartão implicara apenas a mudança de um NIP, não comprometendo o sistema.

-Como exemplo da utilização de criptoqrafia para protecção de ficheiro "on-line" e "off-line", citaremos o sistema IPS (Information Protection System) que usa o DES como algoritimo de cifragem.

O armazenamento é feito em blocos de 8 bytes na forma cifrada. Para evitar o acesso às chaves, estas não são armazenadas sendo da inteira responsabilidade do utilizador que é assim o único a poder decifrar os ficheiros.

Do que foi exposto não é legitimo concluir-se que a criptografia resolve integralmente os problemas de segurança e autentificaçao de informação levantados pela enorme expansão das redes e bancos de dados. Este é contudo um campo ainda "pouco" explorado e, como vimos, potencialmente muito rico.