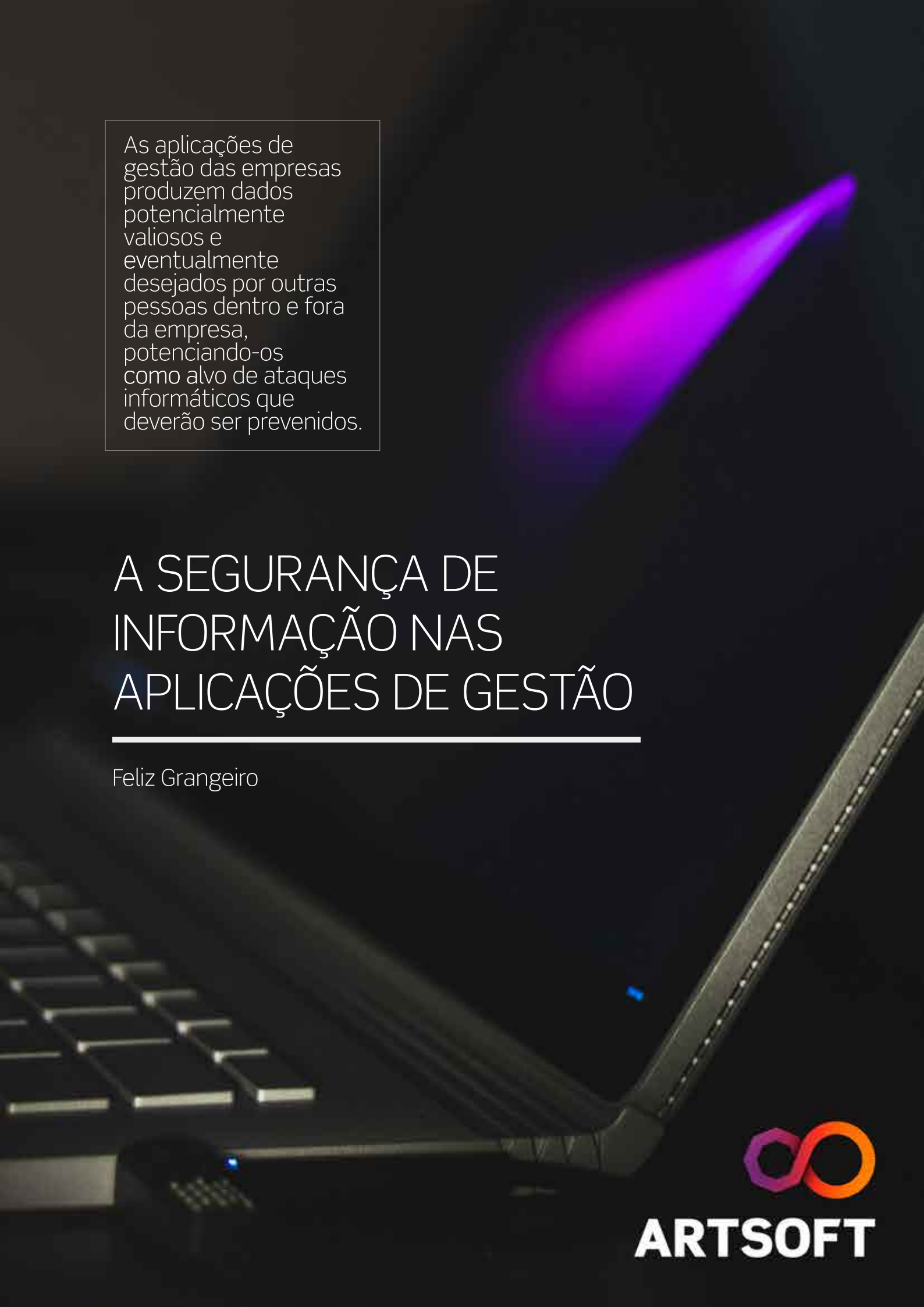


As aplicações de gestão das empresas produzem dados potencialmente valiosos e eventualmente desejados por outras pessoas dentro e fora da empresa, potenciando-os como alvo de ataques informáticos que deverão ser prevenidos.

A SEGURANÇA DE INFORMAÇÃO NAS APLICAÇÕES DE GESTÃO

Feliz Grangeiro

ARTSOFT

Conteúdo

A segurança de informação

As instituições e empresas possuem nos seus sistemas dados confidenciais sobre os seus colaboradores, clientes, fornecedores, produtos, investigação, dados financeiros, etc. que são inseridos, processados e guardados em computadores e, muitas vezes, transferidos através de redes para outros computadores.

Se esses dados caírem em mãos erradas, a empresa ou os seus clientes poderão sofrer danos pessoais ou financeiros irreparáveis, para além da perda de reputação da empresa.

A importância deste tema nas organizações de qualquer dimensão é tal que para muitas, não é suficiente terem identificação dos riscos e definição de *políticas de segurança*, mas pretendem verificar regularmente se essas políticas continuam adequadas aos desafios que a organização enfrenta diariamente, implementando o processo de certificação de Sistemas de Gestão da Segurança de Informação, regulado pela norma ISO/IEC 27001.

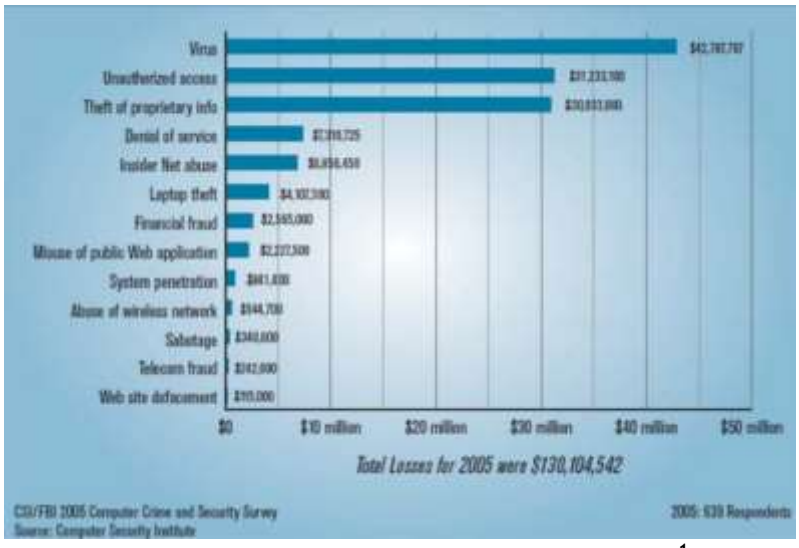


Fig. 1 – Totais de perdas por tipos de ataques[1]

As ameaças a que os dados estão sujeitos provêm de:

- Ataques de *software* (vírus, vermes, ataques de 'phishing', cavalos de troia);
- Roubo de propriedade intelectual (documentos, bases de dados, software);
- Acesso não autorizado / roubo de identidade (obter as credenciais de alguém que tenha acesso a dados vitais ou privilegiados no sistema);
- Roubo de equipamento informático (computadores portáteis, discos externos, discos USB, *tablets*, telemóveis, etc.);
- Extorsão (roubo de dados[2] ou alteração de funcionamento de sistemas, com o intuito de receber um pagamento pela devolução ou reposição do funcionamento dos sistemas afetados).

---

1 Ver documento 'Computer crime and security survey / Dollar Amount Losses by Type', na pág. 15 de: http://www.cpppe.umd.edu/Bookstore/Documents/2005CSISurvey.pdf

- Sabotagem (paragem de sites, alteração do conteúdo de sites visando a perda de confiança dos seus utilizadores, ataque DDoS[3] tornando o site indisponível aos seus utilizadores);

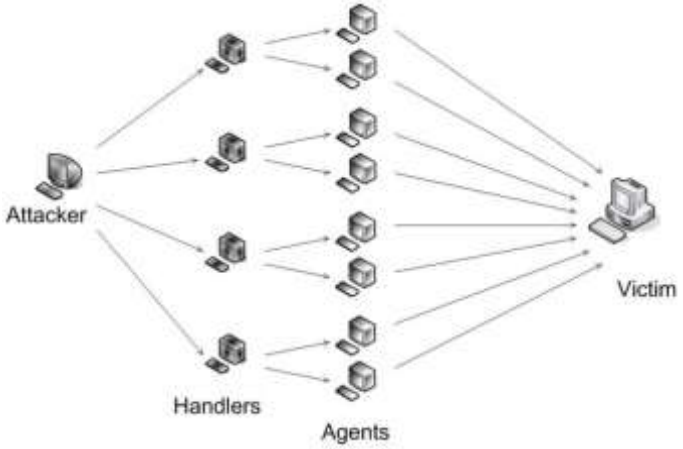


Fig. 2 – Esquema de um ataque DDoS[4]

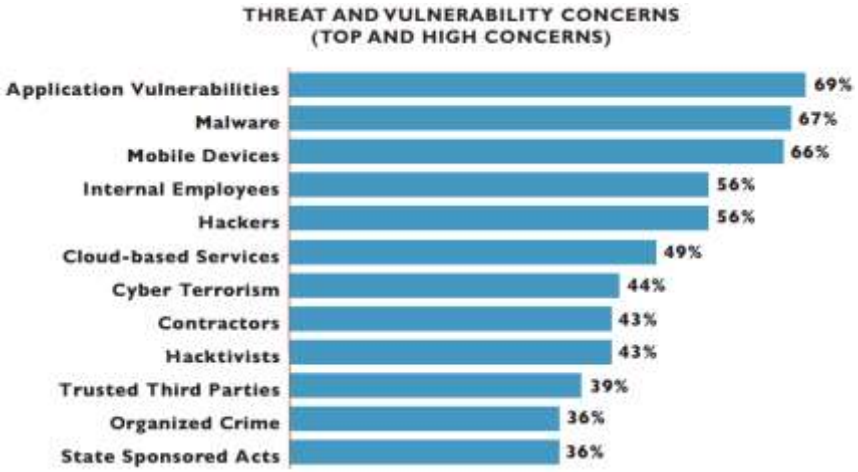


Fig. 3 – Ameaças de segurança[5]

---

2 No caso da extorsão, os dados são copiados e eliminados do sistema. Mesmo depois do resgate, a confidencialidade dos documentos já está irremediavelmente perdida.

3 'Distributed Denial of Service' – Ataque efetuado por milhares de computadores pedindo recursos do servidor, de forma a consumir todos os seus recursos ou a largura de banda da sua ligação à internet.

4 Artigo 'Protection against Denial of Service Attacks' (maio de 2009) na pág. 2 de: http://staffweb.cms.gre.ac.uk/~lg47/publications/LoukasOke-DoSSurveyComputerJournal.pdf

5 Ver pág. 6 de: https://www.isc2.org/uploadedfiles/(isc)2_public_content/2013%20global%20information%20security%20workforce%20study%20feb%202013.pdf

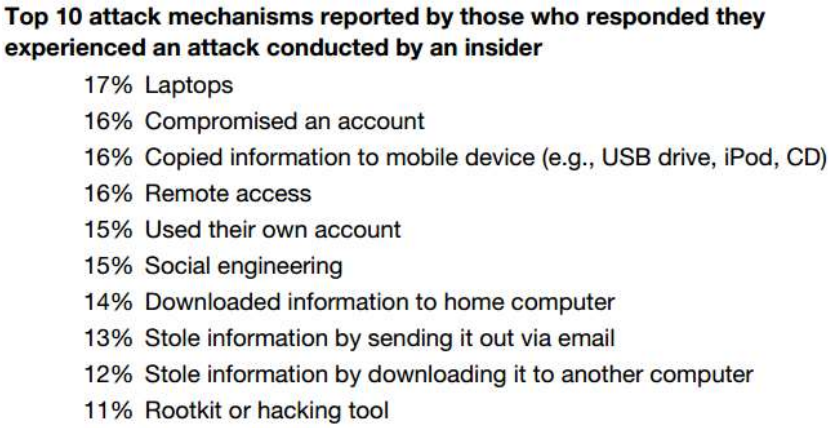


Fig. 4 – Os principais 10 mecanismos de ataque interno[6]

As ameaças a que os sistemas estão sujeitos, por ordem de importância, provêm de:

- Vulnerabilidade dos sistemas e aplicações[7] - exploradas por vírus, vermes e cavalos de troia, que uma vez instalados permitem efetuar as mais variadas operações a partir de fora da rede;
- Dispositivos móveis - o roubo de telemóveis, *tablets*, discos externos, memórias USB, se não estiverem devidamente protegidos contra intrusos expõem os seus dados;
- Colaboradores internos (por terem acesso interno à intranet, quando executam qualquer aplicação não autorizada, permitem a instalação no seu computador ou mesmo no sistema de software malicioso que coloca à disposição do atacante todas as suas permissões sempre que estiver conectado à rede interna);
- Fornecedores contratados - conhecendo o preço das propostas de outros concorrentes, permite-lhes apresentar propostas que maximizem o interesse do fornecedor;
- Hackers e crackers - um 'hacker' atua por curiosidade, necessidade profissional, vaidade, espírito competitivo, patriotismo, ativismo ou crime. Os hackers que usam seu conhecimento para fins imorais, ilegais ou prejudiciais são chamados 'crackers';
- Serviços expostos na nuvem - estando expostos na nuvem, se não estiverem devidamente protegidos rapidamente se tornam alvo de curiosos, hackers e crackers;
- 'Hacktivistas' (As suas atividades são quase sempre consideradas como crimes cibernéticos como ataques DoS[8], não exploram dados confidenciais e são normalmente inofensivos).

---

6 Ver pág. 10 de: http://www.pwc.com/en_US/us/increasing-it-effectiveness/publications/assets/us-state-of-cybercrime.pdf

7 Aplicações que não foram projetadas para se defenderem de ataques externos, como, por exemplo, *SQL injection*. Ver exemplos em: http://www.w3schools.com/sql/sql_injection.asp

8 Uma analogia interessante sobre o DDoS de autoria de Richard Stallman, que considera este tipo de ataque como um protesto do mundo físico: um ataque DDoS seria equivalente a um protesto efetuado por um grande grupo de pessoas em frente a um prédio, dificultando a entrada e saída de pessoas, e o único dano que causam é apenas inconveniência e embaraço para as pessoas do prédio. Ver mais em: http://pt.wikipedia.org/wiki/Hacktivismo

- Cyber terrorismo - grupos especializados em provocar prejuízos a empresas ou ao estado, por motivos económicos, políticos ou religiosos;
- Crime organizado - grupos especializados em furto ou extorsão a empresas, por motivos económicos, para seu benefício, ou para financiar atividades políticas, religiosas, ou outras;

Um sistema considera-se seguro se conseguir[9]:

- **Integridade**: garantir que os dados são efetivamente os que foram fornecidos a um sistema / disponibilizados por um sistema pressuposto, e que não sofreram alterações de forma fortuita ou maliciosa; *No caso de uma ação maliciosa, o utilizador poderá estar a receber ou fornecer dados de/a um sistema alternativo malicioso, pressupondo tratar-se do verdadeiro*;
- **Confidencialidade:** assegurar que só as pessoas certas têm acesso aos dados disponibilizados de forma inteligível; *Potenciais estranhos que intercetem a mensagem deverão vê-la de forma cifrada*;
- **Disponibilidade**: Assegurar que os dados estão sempre disponíveis para os seus legítimos utilizadores, com uma qualidade de serviço mínima garantida (num tempo mínimo pré-definido); *manter os dados em equipamentos desadequados, sob sistemas vulneráveis ou desadequados, com largura de banda insuficiente, ou expostos a ataques internos ou externos, pode causar danos à credibilidade dos dados e do sistema, bem como á reputação da empresa.*
- **Não repúdio:** assegurar que um documento emitido ou transação efetuados por determinada pessoa não possam vir a ser negados por essa pessoa; *Se isso não for assegurado, toda a responsabilidade que tal documento ou transação implique deixa de ter qualquer efeito;*
- **Autenticidade:** assegurar que os utilizadores só têm acesso aos recursos autorizados, e assegurar que os participantes de uma comunicação ou transação sejam efetivamente quem se acredita que sejam. Para que um sistema seja credível deverá possuir processos de credenciação e controlo de acessos suficientemente seguro para fornecer essas garantias.

Um sistema considera-se que está sob ameaça de segurança se houver violação de uma das seguintes caraterísticas:

- **Perda de Integridade**: quando documentos ou dados ficam expostos a pessoas não autorizadas, poderão existir modificações sem o conhecimento do seu autor.
- **Perda de Confidencialidade**: quando existe quebra de sigilo de determinado dado, quer sejam as credenciais de um utilizador ou dados não autorizados do sistema.
- **Perda de Disponibilidade**: perda de comunicação com um servidor da empresa, pela falha de um serviço ou aplicação crítica do negócio, causada por motivo interno ou externo ao equipamento ou ao serviço ou aplicação, ou por ação não autorizada de pessoas, de forma intencional ou não.

[9] Ver: http://pt.wikipedia.org/wiki/Segurança_da_informação e http://en.wikipedia.org/wiki/Information_security

Para assegurar estas caraterísticas, existem mecanismos de segurança para:

- Cifra: Transformar de forma reversível os dados tornando-os ininteligíveis para terceiros: através de algoritmos de cifra e uma chave secreta ou um par de chaves privada / pública produzindo uma nova sequência de dados decifráveis apenas por quem possua a chave secreta ou a chave privada do destinatário.
- Assinatura digital: Associar a um documento um conjunto de dados dependentes do conteúdo desse documento e da chave privada de quem o assina. A assinatura digital garante a integridade e autenticidade do documento assinado.
- Controle de acesso: Para aceder a um sistema seguro, deverá ser utilizado um protocolo seguro que garanta que o sistema seja o que se acredita que seja, fornecendo a identificação do sujeito que pretende o acesso, através do uso de pares 'identificação' / 'palavra chave', de sistemas biométricos, ou outros, desde que sejam suficientemente seguros para o efeito.

Algoritmos de segurança

Referem-se apenas estes dois mecanismos muito utilizados em segurança de informação, na perspetiva de melhor entender o que fazem e para que servem, mas não com a profundidade tal que os torne inacessíveis à maioria dos utilizadores.

Cifra

Os algoritmos de cifra transformam uma mensagem (ou documento) a partir de uma chave, produzindo uma mensagem ininteligível para terceiros, e decifrável apenas por quem possua a chave secreta ou a chave privada (mais detalhes em 'Apêndice B – Técnicas Cifra', na pág. 17).

Há três processos de recuperação do texto original: a *decriptação*, a *criptanálise* e a '*força bruta*':

- A *decriptação* é o processo legítimo efetuado por quem possua a chave fornecida por quem produziu a mensagem original.
- A *criptanálise* é o processo ilegítimo efetuado por alguém que não deveria ter acesso ao conteúdo da mensagem, mas tenta consegui-lo, através do conhecimento e aplicação das técnicas de encriptação.
- A recuperação através de *força bruta* é um processo ilegítimo efetuado por alguém que não deveria ter acesso ao conteúdo da mensagem, mas tenta consegui-lo efetuando uma série exaustiva de tentativas até conseguir obter algo com algum sentido, e deduzindo, a partir daí, a técnica e / ou a chave a empregar para recuperar a mensagem completa.

Hash

Os algoritmos de 'hash' processam uma mensagem (ou documento), produzindo um resultado baseado no seu conteúdo de tal forma que, qualquer modificação ocorrida na mensagem original ou documento produza um resultado diferente, e a partir desse resultado não deve ser possível obter o conteúdo original (ver exemplos no 'Apêndice A – Técnicas de Hash', na pág. 15).

Se um algoritmo de 'hash' produz um resultado diferente por mais simples que seja a modificação de uma mensagem, então, juntando a mensagem com o resultado, está a enviar-se ao destinatário uma forma deste verificar se o documento não foi modificado desde que foi disponibilizado pelo seu emissor (dispõe de uma forma de verificar a ***integridade*** do mesmo).

Mas nada impede que alguém, por ação maliciosa e conhecendo o algoritmo que produz o resultado, altere o documento e também o resultado, comprometendo totalmente a ***integridade*** do documento. Por outro lado, mesmo que exista um documento que esteja integro, não é possível confirmar quem o emitiu, por não existir num tal documento qualquer mecanismo de segurança que o garanta.

Para resolver ambos os problemas, temos que recorrer à criptografia de chave privada/pública.

Se o resultado do 'hash' for cifrado com a chave privada do proprietário do documento (ou mensagem), esse resultado é a assinatura digital desse documento, que se for anexada a esse documento, obtém-se 'um documento assinado'.

Quem recebe o documento, se pretender verificar quem é o seu emissor, obtém a chave *pública* de quem diz ser o emissor, e com esta decifra a assinatura do documento. Se o conseguir fazer, obtém a garantia de quem foi o emissor. O resultado da decifra revela o resultado do 'hash' do documento no seu estado original. Calculando o 'hash' do documento no estado em que foi recebido e comparando-o com o do estado original, sendo iguais, obtém a garantia da integridade do documento e o não repúdio do emissor[10].

[10] Desta forma o emissor nunca poderá alegar não ter sido ele a emitir o documento (porque foi assinado com a sua chave privada), ou que este foi entretanto modificado (garantido pelo mecanismo de verificação de integridade do documento).

Investimento em segurança de informação

O investimento em segurança de informação deverá ter em conta duas vertentes: o valor real ou potencial que os dados protegidos possam ter para terceiros, e o intervalo de tempo em que os dados se devem manter confidenciais.

Neste ponto de vista, um sistema considera-se seguro se os dados que contém forem públicos, ou, tendo determinado valor, se o investimento necessário para os obter for superior a esse valor, ou o tempo necessário para os obter seja superior à data de validade desses dados[11].

Assim, se os dados forem de acesso público o investimento necessário em segurança será apenas o essencial para garantir a sua integridade e disponibilidade, por serem estes os fatores que, sendo atacados, poderão criar problemas de credibilidade à empresa que os disponibiliza.

Se os dados forem confidenciais, terão um valor potencial 'x'. O investimento em segurança, para além do já referido para os dados públicos, deverá ser aquele que garanta que um atacante terá de investir mais do que 'x' ou demore tanto tempo para o conseguir que ultrapasse a data em que esses dados se tornem públicos.

Portanto, investir um valor mais elevado que este em segurança não tornará o sistema mais seguro.

[11] A data em que esses dados se tornarão públicos, como, por exemplo, os resultados das empresas cotadas em bolsa, ou a apresentação de contas de todas as empresas.

Vulnerabilidades dos dados de ERP

Administração de utilizadores do ERP

Num ERP que não respeite as regras de segurança, quando o administrador de sistemas, ou alguém em quem tenha delegado essa tarefa, efetuar alterações de perfil ou de permissões de determinado utilizador, deverá testar se as mesmas correspondem ao esperado. Para tal, se o utilizador estiver presente, poderá pedir-lhe para introduzir as suas credenciais em determinada máquina e efetuar o teste; Se houver alguma coisa a corrigir, tem que voltar a colocar as credenciais de administrador, efetuar as correções, voltar a pedir ao utilizador para voltar a inserir as suas credenciais, voltar a testar, repetindo o processo enquanto houver alguma coisa a corrigir. Se o utilizador não estiver presente, ***tem que lhe redefinir a palavra-passe***, testar, e fornecer-lhe a nova palavra-passe, *devendo este modificá-la de imediato*;

Na vertente 'eficiência' o primeiro caso, envolvendo dois recursos para esta tarefa, não pode ser considerado como um processo eficiente.

Na vertente 'segurança' estas operações apresentam os seguintes riscos:

- Quer o utilizador esteja presente ou não, uma vez que os testes são feitos com a conta do utilizador, todas as operações sujeitas a registo irão ser marcadas como tendo sido efetuadas por esse utilizador, violando o princípio do não repúdio;
- Se o utilizador não estiver presente, a sua palavra-passe é destruída e são efetuadas operações em nome do utilizador alvo; se o utilizador estiver presente, quanto mais correções forem efetuadas, mais vezes este terá de inserir a mesma palavra-passe, expondo ao administrador de sistema as palavras-passe mais fracas. Ambos os casos violam os princípios de confidencialidade, não repúdio e autenticação.

Solução:

O sistema ERP deverá permitir que o administrador informe o sistema que está em modo 'teste', pretendendo utilizar a conta do utilizador 'x' mas com a palavra-chave do administrador, marcando as eventuais operações efetuadas como tendo sido executadas pelo administrador, ou na impossibilidade desta última, que permita comutar automaticamente para uma base de dados de testes.

Autenticação de utilizadores

A autenticação do ERP deve possuir um mecanismo de segurança 'controlo de acesso' que consiga assegurar que só os utilizadores autorizados tenham acesso ao mesmo, e aos dados mantidos e gerados pelo mesmo[12] - através do ERP ou de quaisquer outras ferramentas externas - e criar a máxima frustração às tentativas de acesso maliciosas.

Quantas mais vezes as credenciais estejam guardadas no sistema, mais locais existem para os manter, e outros tantos para atacar. Se existir um serviço seguro de autenticação de utilizadores e o ERP consiga usá-lo, deve ser este o mecanismo de segurança preferido, evitando existir mais um local onde a palavra-passe possa estar exposta, e que tenha que ser mantida[13].

Se o sistema não possuir um serviço seguro de autenticação de utilizadores ou o ERP o consiga usar, então deve ser capaz de detetar, suportar e frustrar os ataques mais conhecidos às credenciais dos seus utilizadores:

- Um atacante sabe que os utilizadores costumam usar palavras-passe com sequências simples de dígitos[14], nomes de pessoas, palavras ou pequenas frases numa determinada língua[15]. Se souber qual o identificador da conta de um determinado utilizador, pode usar esse conhecimento para criar um script ou aplicação que injete na aplicação a atacar pares identificação / palavra-passe obtidos a partir das sequências referidas, ficheiros de dicionários, ou mesmo todas as combinações possíveis (ataques de força bruta).

  Para que este ataque possa ser bem-sucedido, necessita de efetuar um grande número de tentativas num intervalo de tempo até que seja detetado.

  Para frustrar esse ataque, o serviço de autenticação do ERP deverá contar o nº de vezes que a credencial dessa conta falhou, e se ultrapassar determinado número, suspender temporariamente ou mesmo bloquear essa conta. Assim, mesmo que o atacante apresente as credenciais corretas, estas serão rejeitadas.

- Se tiver acesso a isso, o atacante poderá atacar o local da base de dados onde as credenciais estão guardadas, para as obter, ou tentar modificar as autorizações de determinada conta.

  Para frustrar esse ataque, o registo de utilizador não deverá conter a palavra-passe mas um 'hash' da mesma, e o conteúdo das colunas a proteger deve ser autenticado com outro 'hash'. Desta forma, sempre que o registo do utilizador for lido, o respetivo 'hash' deve ser recalculado e comparado com o guardado. Se estes não forem iguais, o registo desse utilizador deve ser ignorado, comportando-se o sistema como se o registo estivesse suspenso ou anulado, mesmo que se trate da conta de administrador (a primeira a ser atacada). Mas se esta conta ficar bloqueada, o ERP deverá prever a entrada no sistema através de credenciais de super-adminstrador.

---

[12] Bases de dados, ficheiros de dados exportados para instituições tributárias, financeiras, etc.
[13] Conhecido como 'Single sign-on'. Ver: http://en.wikipedia.org/wiki/Single_sign-on
[14] '12345', '9999', 'abcdefgh', 'qwertyuiop', '159753', '48621793', '123abc', etc.
[15] As mais usadas: português e inglês.

Documentos legais

Para que um documento de faturação seja considerado como emitido na forma legal, ou é emitido manualmente em livros impressos por tipografias autorizadas, impresso através de programa certificado, ou emitido através de um sistema de intercâmbio eletrónico de dados (EDI), ou através da emissão de um documento em formato digital autenticado com uma assinatura eletrónica avançada[16].

A emissão de documentos em formato digital que não sejam autenticados com assinatura eletrónica avançada, não satisfaz as condições exigidas pelo CIVA, art.º 36, nº 10.

Mesmo que seja possível garantir a integridade do seu conteúdo[17], não se consegue garantir a autenticidade da sua origem, porque esta garantia só pode ser dada pela assinatura eletrónica avançada do documento, anexada ao documento com recurso à chave privada do certificado digital da empresa, e verificável por qualquer pessoa através da chave pública da mesma.

Para autenticar um documento com uma assinatura eletrónica avançada, é necessário obter um certificado digital apropriado para assinatura de documentos passado por uma entidade certificadora.

Estando na posse deste certificado, o ERP deverá utilizá-lo para assinar quaisquer documentos em formato digital (o formato 'pdf' suporta assinatura digital e é o mais usado para este efeito).

Ordens de pagamento

Num ERP, o processamento de pagamentos a fornecedores e / ou a colaboradores gera um ficheiro em formato texto a enviar à entidade bancária com as instruções de pagamento, facilmente legível e / ou modificável por quem lhe tiver acesso. Pela sua natureza, este ficheiro é um alvo fácil para um atacante poder obter dados confidenciais da empresa (observando os ficheiros) ou obter grandes quantias (modificando os números de conta beneficiários) com um esforço mínimo.

Para frustrar este ataque, o ERP deverá possuir os mecanismos de segurança (integridade e confidencialidade), suficientes para dificultar ao máximo as tarefas do atacante:

- Na fase de preparação do ficheiro, este não deverá ser gerado em nenhuma parte do sistema de ficheiros (onde poderia estar acessível a alguém), mas antes, deverá ser gerado em memória, assinado, cifrado e transferido para a base de dados.
- Na fase de envio, o ficheiro deverá ser decifrado, verificado contra modificações (comparando o ficheiro com a sua assinatura), gerado no diretório 'temp' do sistema

[16] Ver definições no regime jurídico dos documentos eletrónicos e da assinatura eletrónica republicados pelo DL 62/2003, na pág. 2177 de http://dre.pt/pdf1sdip/2003/04/079A00/21702185.pdf, e as definições no art.º 2º, e em Perguntas Frequentes de: http://www.scee.gov.pt/ECEE/pt/faq/

[17] Por exemplo, um documento em formato PDF com opção de não modificação e encriptação.

operativo no computador do utilizador que o vai submeter à instituição financeira, e destruído, logo que o sistema do banco tenha reportado o sucesso da transferência.

Ficheiros de reporte a entidades externas

O envio de diversos ficheiros de texto a várias entidades[18] com dados relativos a empregados e terceiros, poderá ser alvo fácil de obter dados confidenciais por quem lhe tiver acesso.

Para manter a confidencialidade destes dados, o ERP deverá possuir o mecanismo de segurança (confidencialidade) suficiente para dificultar ao máximo as tarefas do atacante, bastando para isso usar o mesmo processo que o já referido para as ordens de pagamento.

Base de Dados - Tabelas com dados pessoais

De acordo com a legislação em vigor (Código do Trabalho[19], art. 17, nº 4, e Lei 67/98[20], art.14 e 15), a empresa deve assegurar a proteção dos dados pessoais de empregados e terceiros, podendo utilizá-los apenas para os fins a que se destinam.

Infelizmente, esta é uma área muito apetecível para obtenção de dados confidenciais (remunerações, comissões, prémios, etc.), que poderão ter algum valor transacional para os concorrentes da empresa, comunicação social, etc.. A distribuição dos dados pelas tabelas deverá ser de tal forma que os dados públicos e dados privados estejam em tabelas separadas, devendo estas últimas serem cifradas para garantirem total confidencialidade[21]. Desta forma, só quem conhece a palavra-passe de acesso a essas tabelas, poderá efetuar processamentos e obter relatórios provenientes das mesmas.

Ligações externas

Se o ERP possuir serviços de comunicação[22] com aplicações externas, deve continuar a assegurar todos os mecanismos de segurança integridade, confidencialidade, disponibilidade, não repúdio e autenticação.

Se os serviços não possuírem outro mecanismo de confidencialidade, os dados trocados deverão circular num túnel VPN, ou protegidos por comunicações (por exemplo, baseadas no protocolo IPsec[23]).

Em nenhuma circunstância as credenciais dos utilizadores remotos deverão circular na rede. Para evitar que isso aconteça, essas credenciais deverão ser trocadas mediante um protocolo do tipo 'CHAP' (ver 'Apêndice C – Algoritmo CHAP', na pág. 25).

---

18 Seguradoras, Sindicatos, Ministério do Trabalho, Segurança Social, Banco de Portugal, etc.
19 Ver: https://dre.pt/application/dir/pdf1s/2009/02/03000/0092601029.pdf
20 Ver: https://dre.pt/application/dir/pdf1sdip/1998/10/247A00/55365546.pdf
21 A cifra é o único mecanismo de segurança que garante a confidencialidade.
22 SOAP, REST, HTTP, etc.
23 IPSEC - IP Security Protocol. Ver: http://en.wikipedia.org/wiki/IPsec

Apêndice A – Técnicas de Hash

Os algoritmos de 'hash' processam determinada mensagem (ou documento), produzindo um resultado baseado no conteúdo do mesmo de tal forma que, a partir do resultado não seja possível obter o conteúdo original.

Algoritmo 'somadígitos'

Podemos definir o algoritmo 'somadígitos' como a soma do valor de todas as letras do alfabeto maiúsculo, de acordo com a seguinte tabela:

Letra	*Valor*
Espaço	0
A	1
B	2
C	3
D	4
E	5
F	6

Letra	*Valor*
G	7
H	8
I	9
J	10
K	11
L	12
M	13

Letra	*Valor*
N	14
O	15
P	16
Q	17
R	18
S	19
T	20

Letra	*Valor*
U	21
V	22
W	23
X	24
Y	25
Z	26

Ao aplicar 'somadigitos' ao texto 'SEGURANCA INFORMATICA', devemos adicionar os valores correspondentes a cada letra, obtendo-se o valor '198':

S	E	G	U	R	A	N	C	A		I	N	F	O	R	M	A	T	I	C	A		
19	5	7	21	18	1	14	3	1	0	9	14	6	15	18	13	1	20	9	3	1	=	198

Ao aplicar 'somadigitos' ao texto 'ABCDEFGH TUVWXYZA', também se obtém 198:

A	B	C	D	E	F	G	H		T	U	V	W	X	Y	Z	A		
1	2	3	4	5	6	7	8	0	20	21	22	23	24	25	26	1	=	198

Pode verificar-se que tendo apenas o valor '198', não é possível obter nem sequer conhecer qual o conteúdo original. Neste caso, poderia pertencer a qualquer destas mensagens, mas poderia pertencer a muitas outras que produzam resultado similar.

Se aplicar 'somadigitos' ao texto 'INSEGURANCA INFORMACAO', obtém-se o valor '221':

I	N	S	E	G	U	R	A	N	C	A		I	N	F	O	R	M	A	T	I	C	A		
9	14	19	5	7	21	18	1	14	3	1	0	9	14	6	15	18	13	1	20	9	3	1	=	221

Apesar de muito simples, estes exemplos demonstram como se obtém um 'hash', a dependência do resultado relativamente ao texto e a irreversibilidade do algoritmo. Porém, este não é usado por ser muito fraco para utilização em segurança de informação, sendo usados outros mais seguros, como o '*MD5*', '*SHA1*', etc.

Algoritmos MD5 e SHA1

A descrição destes algoritmos, pela sua complexidade, está fora de âmbito, apresentando-se apenas o resultado do algoritmo aplicado a algumas entradas:

Texto original	*Resultado do Hash MD5*	*Resultado do Hash SHA1*
(texto vazio)	d41d8cd98f00b204e9800998ecf8427e	da39a3ee5e6b4b0d3255bfef95601890afd80709
a	0cc175b9c0f1b6a831c399e269772661	86f7e437faa5a7fce15d1ddcb9eaeaea377667b8
b	92eb5ffee6ae2fec3ad71c777531578f	e9d71f5ee7c92d6dc9e92ffdad17b8bd49418f98
pequeno texto	03b5ae764ced6932a3a401a050c89197	2439196dc1714de686bc687fc75f7720decd3eb4
Pequeno texto	cbafd9da47cf9b450e97d4a237edc623	0cff7f6d23392cbf38880a3a10979b420abec8cf
Pequeno Texto	c57235bbdca0215f370a3e08d3a3748f	97ed6a2ac42c46f590034648ac48f6d6663eba82
Os algoritmos de 'hash' processam determinada mensagem (ou documento), produzindo um resultado baseado no conteúdo do mesmo de tal forma que, a partir do resultado não seja possível obter o conteúdo original.	c5727bebecb4af4ad1169f054b903c68	ef698edcfd7469b551ccfa3a2999a8e67c5b258d
Os algoritmos de 'hash' processam determinada mensagem (ou documento), produzindo um resultado baseado no conteúdo do mesmo de tal forma que, a partir do resultado não seja possível obter o conteúdo original	871d7c83d564e486c1eab75a403a8e7f	f1406366b001fcc93781684dfdad94b050648fbd

Pode verificar-se que um texto vazio possui um 'hash' válido, a variabilidade do 'hash' da letra 'a' com o da letra 'b' é enorme, o mesmo se passando na alteração de uma ou duas minúsculas para maiúsculas em 'pequeno texto'.

Na mensagem 'Os algoritmos…' o facto de a segunda não conter o ponto final resultou num 'hash' absolutamente diferente, e em todos eles, a variabilidade do resultado foi independentemente de ter sido usado o algoritmo 'MD5' ou o 'SHA1'.

Apêndice B – Técnicas Cifra

Apenas para quem tenha interesse em saber algo mais sobre criptografia, apresentamos alguns exemplos muito simples das técnicas de cifra / decifra.

Criptografia simétrica

Na criptografia simétrica, é utilizada a mesma chave para cifrar e decifrar a mensagem.

Se esta chave for usada muitas vezes pode ser alvo de criptanálise, comprometendo a confidencialidade do texto ou mensagem. Por essa razão, uma chave usada em criptografia simétrica não deverá ser reutilizada para outra mensagem.

Técnica de César

Uma das técnicas de transformação de uma mensagem é conhecida como a '*técnica de César*', que consiste em transformar um determinado carater, em outro que esteja à distância de 'x' carateres. A tabela seguinte representa uma transformação do alfabeto maiúsculo, com uma distância de 3 carateres:

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C

Se considerarmos a mensagem 'SEGURANCA INFORMATICA', transformada através desta tabela, obteríamos:

S	E	G	U	R	A	N	C	A		I	N	F	O	R	M	A	T	I	C	A
V	H	J	X	U	D	Q	F	D		L	Q	I	R	U	P	D	W	L	F	D

À primeira vista, o texto 'VHJXUDQFD LQIRUPDWLFD' será ininteligível para todas as pessoas que desconheçam que este texto foi transformado através da 'técnica de César', com uma distância de '3'. Mas esta técnica é muito vulnerável e fraca, podendo ser atacada através de criptanálise ou mesmo por força bruta.

Técnica de Vigenère

Nesta técnica, os carateres da mensagem original são cifrados por meio de diferentes alfabetos, permitindo o uso de uma chave de cifra, resolvendo a limitação do deslocamento fixo da técnica de César.

Para usar esta técnica, é necessário a criação de uma tabela constituída pelos 26 alfabetos possíveis, deslocados uns relativamente aos outros em uma posição: o primeiro começa por A, o segundo por B, e assim sucessivamente.

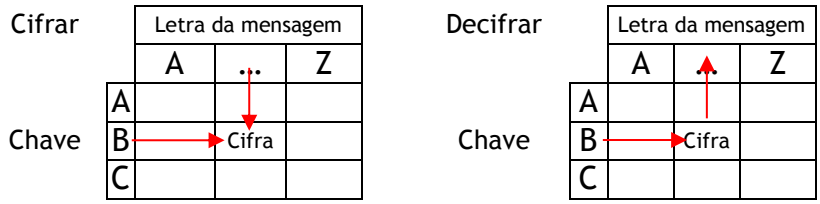
Cada letra da mensagem inicial é cifrada no alfabeto correspondente ao carater fornecido por uma chave de cifra, que deve ser o mais comprida possível, podendo ter carateres repetidos. Quando esgotados todos os carateres da chave, volta-se a utilizar o primeiro carater da chave (daí o interesse em que seja o mais comprida possível).

A tabela seguinte é uma tabela de Vigenère modificada que inclui o carater '_' representando um espaço, que torna a cifra e decifra mais fácil para os exemplos seguintes.

Tabela de Vigenère (modificada para incluir o carater 'espaço'):

	A	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**	**J**	**K**	**L**	**M**	**N**	**O**	**P**	**Q**	**R**	**S**	**T**	**U**	**V**	**W**	**X**	**Y**	**Z**	**_**
A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	_
B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	_	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	_	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	_	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	_	A	B	C	D
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	_	A	B	C	D	E
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	_	A	B	C	D	E	F
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	_	A	B	C	D	E	F	G
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	_	A	B	C	D	E	F	G	H
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	_	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	_	A	B	C	D	E	F	G	H	I	J
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	_	A	B	C	D	E	F	G	H	I	J	K
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	_	A	B	C	D	E	F	G	H	I	J	K	L
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	_	A	B	C	D	E	F	G	H	I	J	K	L	M
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	_	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	P	Q	R	S	T	U	V	W	X	Y	Z	_	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	Q	R	S	T	U	V	W	X	Y	Z	_	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	R	S	T	U	V	W	X	Y	Z	_	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	S	T	U	V	W	X	Y	Z	_	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	T	U	V	W	X	Y	Z	_	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	U	V	W	X	Y	Z	_	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	V	W	X	Y	Z	_	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	W	X	Y	Z	_	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	X	Y	Z	_	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Y	Z	_	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	Z	_	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y
_	_	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z

Utilizando esta técnica, cada letra da mensagem será cifrada através de um alfabeto diferente, variando consoante os carateres da chave:



Para cifrar a mensagem 'SEGURANCA INFORMATICA', com a chave 'CRIPTOGRAFIA', irá usar-se o alfabeto da linha correspondente à respetiva letra da chave, escolhendo-se o carater da coluna correspondente à letra a cifrar:

Chave	C	R	I	P	T	O	G	R	A	F	I	A	C	R	I	P	T	O	G	R	A
Mensagem	S	E	G	U	R	A	N	C	A	_	I	N	F	O	R	M	A	T	I	C	A
Cifra	U	V	O	I	J	O	T	T	A	E	Q	N	H	E	Z	A	T	G	O	T	A

Para decifrar esta mensagem, recorre-se ao processo inverso: Sabendo qual o alfabeto (linha) a usar a partir da respetiva letra da chave, localizar a coluna onde se encontra o carater cifrado, escolhendo a primeira letra dessa coluna:

Chave	C	R	I	P	T	O	G	R	A	F	I	A	C	R	I	P	T	O	G	R	A
Cifra	U	V	O	I	J	O	T	T	A	E	Q	N	H	E	Z	A	T	G	O	T	A
Mensagem	S	E	G	U	R	A	N	C	A		I	N	F	O	R	M	A	T	I	C	A

Na técnica de César, uma determinada letra origem é transformada sempre na mesma letra destino, tornando a criptanálise muito fácil. Na técnica de Vigenère, pode verificar-se que a mesma letra origem 'A' aparece codificada no destino como 'O,A,T', mas a letra 'O' também serviu de destino às letras origem 'G,A,I'.

A técnica de Vigenère manteve-se inquebrável durante mais de 300 anos, mas a repetição da chave na determinação do alfabeto a partir do qual se cifra, levou ao sucesso da sua criptanálise, conseguida por Charles Babbage.

Em 1918 Arthur Scherbius patenteou uma máquina eletromecânica, a Enigma[24], que deu origem às máquinas de criptografia ***Enigma*** (alemã) e ***Purple*** (japonesa) na II Guerra Mundial.

A máquina ***Enigma*** usava um rotor que funcionava com um conjunto de 3 cilindros conseguindo 26^3 = 17,576 alfabetos; foi posteriormente aumentada para 5 cilindros produzindo 11,881,376 alfabetos (a tabela original não incluía o carater 'espaço').

As cifras produzidas pela máquina ***Enigma*** foram quebradas pelos serviços secretos britânicos, que, segundo Winston Churchill, permitiu aos aliados ganharem a guerra[24].

Técnica XOR

As técnicas de cifra através de computador utilizam maioritariamente a operação lógica 'XOR', que possui uma propriedade extraordinária: aplicando a chave à mensagem produz uma mensagem cifrada, e voltando a aplicar a essa mensagem a mesma operação com a mesma chave produz a mensagem original.

A operação XOR define-se de forma muito simples: a conjunção de dois bits iguais resulta no bit '0' e inversamente, a conjunção de dois bits diferentes resulta no bit '1'.

Exemplo 1 - Chave 10101010:

Cifrar		
Bits	***Chave***	*Mens.*
0	***1***	1
1	***0***	1
1	***1***	0
0	***0***	0
1	***1***	0
1	***0***	1
1	***1***	0
0	***0***	0

Decifrar		
Mens.	***Chave***	*Bits*
1	***1***	0
1	***0***	1
0	***1***	1
0	***0***	0
0	***1***	1
1	***0***	1
0	***1***	1
0	***0***	0

Exemplo 2 - Chave 00001111:

Cifrar		
Bits	***Chave***	*Mens.*
0	***0***	0
1	***0***	1
1	***0***	1
0	***0***	0
1	***1***	0
1	***1***	0
1	***1***	0
0	***1***	1

Decifrar		
Mens.	***Chave***	*Bits*
0	***0***	0
1	***0***	1
1	***0***	1
0	***0***	0
0	***1***	1
0	***1***	1
0	***1***	1
1	***1***	0

Através do exemplo 2, nesta técnica, quando a chave é composta exclusivamente por bits '0', a saída é igual à entrada, e inversamente, quando a chave é composta exclusivamente por bits '1', a saída é igual ao inverso da entrada.

Tal como em todas as técnicas criptográficas, quanto maior for a variedade (aleatoriedade) de bits da chave, maior é a aleatoriedade de bits do resultado final.

[24] Ver http://en.wikipedia.org/wiki/Enigma_machine

Técnicas de cifra inquebráveis

Frank Miller descreveu pela primeira vez em 1882 a técnica 'One-time pad', uma técnica de cifra que, se usada corretamente, é inquebrável[25], conforme constatou Joseph Mauborgne, chefe do departamento de investigação criptográfica do exército dos EUA, no final da I Grande Guerra: *se uma chave for composta por carateres totalmente aleatórios e tiver tamanho igual ou superior ao tamanho da mensagem*, a criptanálise é impossível.

Porém, a dificuldade da sua utilização está exatamente na criação e distribuição de chaves totalmente aleatórias e de grande tamanho.

Criação de chaves pseudoaleatórias de tamanho infinito

A dificuldade em obter uma chave absolutamente aleatória e que tenha um tamanho maior ou igual ao da mensagem levou à criação de técnicas de criação de chaves de tamanho igual ao da mensagem e pseudoaleatórias.

Para se criar uma chave 'pseudoaleatória', pode usar-se o resultado da cifra de um bloco de dados com o mesmo tamanho da chave[26], e usar esse resultado como a nova chave de cifra do bloco seguinte. Porém, a utilização de uma mensagem de conteúdo constante permite uma criptanálise fácil. Para que o sistema de criptografia não fique dependente do conteúdo da mensagem, alguns algoritmos juntam um bloco de dados chamado 'vetor de inicialização' ou 'IV-initialization vector', ou, um nome mais culinário, mas adequado: 'sal'. Este bloco, tal como a chave, tem que ser conhecido no lado da cifra bem como na decifra, e pode ser calculado a partir de qualquer elemento conhecido, como uma sequência numérica, ou da própria chave.

A forma como a cifra é aplicada a cada bloco de mensagem[27] é muito flexível, não comprometendo a segurança[28], dependendo da complexidade algorítmica, capacidade de execução em múltiplos processadores, capacidade de recuperação em caso de erros de transmissão, etc.

---

[25] Ver http://en.wikipedia.org/wiki/One-time_pad

[26] Em substituição da chave, podem usar-se algoritmos fortes de hash aplicados a esta, para maior aleatoriedade.

[27] Ver http://en.wikipedia.org/wiki/Block_cipher_mode_of_operation

[28] Excepto o modo ECB. Ver: http://en.wikipedia.org/wiki/Block_cipher_mode_of_operation

Criptografia assimétrica

Na criptografia assimétrica, é utilizado um par de chaves, uma para cifrar e a outra para decifrar a mensagem ou o documento.

Uma variante da criptografia assimétrica é a criptografia de chave pública / privada:

- Quando se cifra um documento com a chave pública, só quem detiver a respetiva chave privada conseguirá decifrar o mesmo. Esta técnica garante a confidencialidade do documento.
- Quando se cifra um documento com a chave privada, qualquer pessoa que possuir a respetiva chave pública conseguirá decifrar o mesmo. Como, por princípio, se a chave é pública, qualquer pessoa a poderá deter, e assim, qualquer pessoa poderá decifrar o mesmo. Esta técnica garante a identidade do emissor e o não repúdio do documento pelo mesmo, mas não garante qualquer confidencialidade.
- Se um documento for cifrado com a chave pública do recetor e de seguida voltar a ser cifrado com a chave privada do emissor, quando receber o documento, o recetor irá decifrá-lo, utilizando a chave pública do emissor que lhe garante a veracidade da proveniência. Nesta fase, o documento continua cifrado com a chave pública do recetor, que lhe garante a confidencialidade. Se o decifrar novamente com a sua chave privada, obtém o documento original.

Na criptografia de chave pública/privada, o par de chaves tem que obedecer a uma regra: deverá ser praticamente impossível deduzir a chave privada a partir da chave pública.

A criptografia assimétrica recorre a algoritmos extremamente complexos para garantir que qualquer tentativa de ataque consuma tantos recursos computacionais e demore tanto tempo que torne a decifra inviável.

Como exemplo de criptografia assimétrica, segue-se um exemplo que recorre à multiplicação de matrizes em espaços modulares[29]. Uma matriz serve para chave de cifra, e a sua inversa nesse espaço modular[30] serve como chave de decifra e vice-versa.

Multiplicação de matrizes:

Matriz 1	
a	b
c	d

Matriz 2	
w	x
y	z

Resultante	
aw+by	ax+bz
cw+dy	cx+dz

Exemplo da multiplicação de duas matrizes 2x2 no espaço modular 29:

Matriz 1			Matriz 2			Resultado			Resultado módulo 29	
1	2	x	5	6	=	1x5+2x7=19	1x6+2x8=22	→	19	22
3	4		7	8		3x5+4x7=43	3x6+4x8=50		14	21

[29] A matemática modular é muito usada no cálculo de datas, horas, ângulos, etc. No caso das horas, é usado o módulo 60 para os minutos e segundos, e o módulo 24 para as horas. Como exemplo, 55 segundos + 25 segundos são: [(55+25) mod 60] = [80 mod 60] = 1 min, 20 seg.

[30] Nota: este exemplo não é suficientemente seguro pois a partir de uma chave é possível deduzir a outra, bastando inverter a matriz nesse espaço modular.

Para efetuar a cifra, é necessário converter letras em números, através de uma tabela de equivalência alfabética:

Letra	*Valor*
Espaço	0
A	1
B	2
C	3
D	4
E	5
F	6
G	7

Letra	*Valor*
H	8
I	9
J	10
K	11
L	12
M	13
N	14
O	15

Letra	*Valor*
P	16
Q	17
R	18
S	19
T	20
U	21
V	22
W	23

Letra	*Valor*
X	24
Y	25
Z	26
,	27
.	28

O par de chaves a ser usado na cifra e decifra é o apresentado nas seguintes matrizes, no espaço modular 29:

Chave E	
1	2
3	4

Chave R	
27	1
16	14

Para cifrar em módulo 29, com ‘Chave E’:

Mensagem a cifrar:

E	S	C	O	N	D	E	R
5	19	3	15	14	4	5	18

1º bloco:

ESCO	
5	19
3	15

x

Chave E	
1	2
3	4

=

Resultado	
5x1+19x3=62	5x2+19x4=86
3x1+15x3=48	3x2+15x4=66

→

Resultado mod. 29	
4	28
19	8

2º bloco:

NDER	
14	4
5	18

x

Chave E	
1	2
3	4

=

Resultado	
14x1+4x3=26	14.2+4.4=44
5x1+18x3=59	5x2+18x4=82

→

Resultado mod. 29	
26	15
1	24

Resultado:

4	28	19	8	26	15	1	24
D	.	S	H	Z	O	A	X

Decifrar com ‘Chave R’:

1º bloco:

D.SH	
4	28
19	8

x

Chave R	
27	1
16	14

=

Resultado	
4x27+28x16=556	4x1+28x14=396
19x27+8x16=641	19x1+8x14=131

→

Resultado mod. 29	
5	19
3	15

2º bloco:

ZOAX	
26	15
1	24

x

Chave R	
27	1
16	14

=

Resultado	
26x27+15x16=942	26x1+26x14=236
1x27+24x16=411	1x1+24x14=337

→

Resultado mod. 29	
14	4
5	18

Mensagem decifrada:

5	19	3	15	14	4	5	18
E	S	C	O	N	D	E	R

Tabela de pares de chaves

Para treinar esta técnica, apresenta-se de seguida uma tabela com conjuntos de matrizes de chaves e as respetivas chaves inversas no espaço modular 29.

1:	2	4	→	28	15	11:	2	3	→	24	4	21:	13	28	→	22	9
	6	8		8	7		10	11		23	7		7	25		24	1
2:	3	5	→	17	26	12:	12	16	→	11	21	22:	24	13	→	19	6
	7	9		19	25		20	22		19	6		16	12		23	9
3:	10	11	→	8	20	13:	25	28	→	4	15	23:	1	15	→	17	14
	12	13		6	24		5	8		12	27		3	17		26	1
4:	11	13	→	16	27	14:	5	9	→	9	25	24:	28	9	→	17	4
	15	17		20	24		11	13		8	28		19	27		2	23
5:	12	14	→	5	9	15:	14	19	→	19	20	25:	16	22	→	14	26
	16	18		2	13		23	24		12	28		26	6		7	18
6:	17	19	→	8	6	16:	26	27	→	15	12	26:	11	22	→	4	20
	21	23		28	16		28	17		6	11		8	13		2	19
7:	20	22	→	4	10	17:	8	16	→	17	2	27:	25	17	→	18	21
	24	26		3	12		5	9		26	28		9	2		6	22
8:	21	23	→	22	21	18:	4	22	→	9	17	28:	27	21	→	18	16
	25	27		14	1		15	2		5	18		15	9		28	25
9:	25	26	→	15	13	19:	15	26	→	27	3	29:	9	8	→	26	4
	27	28		28	2		20	27		9	15		7	6		18	10
10:	28	27	→	2	28	20:	9	6	→	15	4	30:	5	4	→	28	2
	26	25		13	15		10	21		26	23		3	2		16	12

Apêndice C – Algoritmo CHAP

Modo de funcionamento do algoritmo 'CHAP' (challenge authentication protocol) básico:

- Um computador 'Cliente' liga-se ao seu 'host' (servidor);
- 'Host' envia uma mensagem com um 'desafio aleatório', composta por 'x' carateres;
- O computador 'Cliente' junta num bloco o ID de utilizador, o desafio aleatório e a palavra-passe, e aplica a esse bloco um algoritmo de 'hash' que 'digere' estes três componentes, produzindo um resultado chamado 'Digest'[31]. Este resultado é enviado ao 'host' juntamente com o ID do utilizador.
- O 'host' recebe o ID do utilizador, e com este localiza na base de dados a respetiva palavra-passe. Como o 'host' conhece o 'desafio' (foi ele que o gerou e enviou), o ID do utilizador, e a palavra-passe (que obteve a partir da base de dados), tem todos os elementos necessários ao cálculo do 'Digest'. Depois de calculado, é comparado com o recebido do computador 'Cliente'. Se estes forem iguais, as credenciais do utilizador são válidas. Senão, ou o desafio é inválido, ou o ID de utilizador é inválido ou a palavra-passe não confere.

Feliz Grangeiro
02/01/2015

[31] Recorde-se que um algoritmo de 'hash' é irreversível: através do resultado *saído*, é impossível obter diretamente a(s) respetiva(s) *entrada(s)*.