cap:2

Capítulo 2

Estratégias para Proteção da Privacidade de Dados Armazenados na Nuvem

Eliseu C. Branco Jr., Javam C. Machado e Jose Maria Monteiro

Abstract

This short course describes the problem of data privacy in cloud computing. Besides, a short review of the main concepts about security and privacy in cloud environments is presented and discussed. In addition, it presents methods and techniques concerning privacy preserving of data stored or processed in the cloud. Finally, an original solution to ensure data privacy in cloud computing environments is discussed.

Resumo

Este minicurso discute o problema da privacidade de dados na computação em nuvem. Além disto, uma revisão dos principais conceitos relacionados à segurança e à privacidade em computação em nuvem é apresentada e discutida. Adicionalmente, os principais métodos e técnicas atualmente existentes para a proteção da privacidade dos dados armazenados ou processados na nuvem são apresentados. Por fim, uma solução original para assegurar a privacidade de dados em ambientes de computação em nuvem é discutida.

2.1. Introdução

A computação em nuvem é uma tecnologia que tem como objetivo proporcionar serviços de Tecnologia da Informação (TI) sob demanda com pagamento baseado no uso. A nuvem computacional é um modelo de computação em que dados, arquivos e aplicações residem em servidores físicos ou virtuais, acessíveis por meio de uma rede em qualquer dispositivo compatível (fixo ou móvel), e que podem ser acessados a qualquer hora, de qualquer lugar, sem a necessidade de instalação ou configuração de programas específicos.

Contudo, para que todo o potencial da computação em nuvem possa ser explorado pelas organizações, é de fundamental importância garantir a segurança e a privacidade dos dados armazenados na nuvem. O relatório *Top Threats to Cloud Computing, 2013,* produzido pela *Cloud Security Alliance-CSA* [1], lista as 10 maiores ameaças para a computação em nuvem. Em primeiro lugar no *ranking* desta pesquisa ficou o "roubo de dados"e em segundo lugar a "perda de dados". Portanto, não será possível atingir todo o potencial da computação em nuvem sem o desenvolvimento de estratégias que assegurem a proteção da privacidade dos dados de seus usuários.

2.1.1. O que é Privacidade?

Os trabalhos de pesquisa sobre privacidade abrangem disciplinas da filosofia à ciência política, teoria política e legal, ciência da informação e, de forma crescente, engenharia e ciência da computação. Um aspecto comum entre os pesquisadores do tema é que privacidade é um assunto complexo. Privacidade é um conceito relacionado a pessoas. Trata-se de um direito humano, como liberdade, justiça ou igualdade perante a lei. Privacidade está relacionada ao interesse em que as pessoas têm em manterem um espaço pessoal, sem interferências de outras pessoas ou organizações. Segundo [Jr et al. 2010], existem basicamente três elementos na privacidade: o sigilo, o anonimato e o isolamento (ou solidão, o direito de ficar sozinho).

Inicialmente, é importante fazer uma distinção entre dados e informação. O padrão RFC-2828 define informação como "fatos e ideias que podem ser representados (codificados) sob vários formatos de dados"e dados como "informações em uma representação física específica, normalmente uma sequência de símbolos que possuem um significado; especialmente uma representação da informação que pode ser processada ou produzida por um computador."

[Jr et al. 2010] definem 3 dimensões para a privacidade:

a) Privacidade Territorial: proteção da região próxima a um indivíduo.

b) Privacidade do Indivíduo: proteção contra danos morais e interferências indesejadas.

c) Privacidade da Informação: proteção para dados pessoais coletados, armazenados, processados e propagados para terceiros.

A privacidade, em relação aos dados disponibilizados na nuvem, pode ser vista como uma questão de controle de acesso, em que é assegurado que os dados armazenados

[1] https://downloads.cloudsecurityalliance.org/initiatives/top_threats/The_Notorious_Nine_Cloud_Computing_Top_Threats_in_2013.pdf, acessado em março de 2014.

estarão acessíveis apenas para pessoas, máquinas e processos autorizados. A privacidade assegura que os indivíduos controlam ou influenciam quais informações relacionadas a eles podem ser coletadas e armazenadas por alguém e com quem elas podem ser compartilhadas [Stallings 2007].

Adicionalmente, privacidade em computação em nuvem é a habilidade de um usuário ou organização controlar que informações eles revelam sobre si próprios na nuvem, ou seja, controlar quem pode acessar qual informação e de que forma isto pode ocorrer. Neste contexto, a proteção de dados está relacionada ao gerenciamento de informações pessoais. De modo geral, informações pessoais descrevem fatos, comunicações ou opiniões relacionadas ao indivíduo, as quais ele desejaria manter em segredo, controlando sua coleta, uso ou compartilhamento. Informações pessoais podem ser associadas a um indivíduo específico tais como nome, cpf, número do cartão de crédito, número da identidade. Algumas informações pessoais são consideradas mais sensíveis do que outras. Por exemplo, informações sobre saúde (registros médicos) são consideradas sensíveis em todas as circunstâncias. Também são exemplos de informações sensíveis: aquelas relacionadas à biometria de um indivíduo e os resultados de uma avaliação de desempenho realizada com os funcionários de uma determinada empresa. Este tipo de informação necessita de proteção adicional em relação à privacidade e segurança.

Para que seja possível discutir em detalhes os principais aspectos relacionados à privacidade dos dados armazenados na nuvem, necessitamos definir precisamente o que é computação em nuvem. Existem várias definições para computação em nuvem. Contudo, neste trabalho, será utilizada a definição apresentada em [Hon et al. 2011]:

a) A computação em nuvem fornece acesso flexível, independente de localização, para recursos de computação que são rapidamente alocados ou liberados em resposta à demanda.

b) Serviços (especialmente infraestrutura) são abstraídos e virtualizados, geralmente sendo alocados como um *pool* de recursos compartilhados com diversos clientes.

c) Tarifas, quando cobradas, geralmente, são calculadas com base no acesso, de forma proporcional, aos recursos utilizados.

À medida em que grandes volumes de informações pessoais são transferidas para a nuvem, cresce a preocupação de pessoas e organizações sobre como estes dados serão armazenados e processados. O fato dos dados estarem armazenados em múltiplos locais, muitas vezes de forma transparente em relação à sua localização, provoca insegurança quando ao grau de privacidade a que estão expostos.

Segundo [Pearson 2013], a terminologia para tratar questões de privacidade de dados na nuvem inclui a noção de controlador do dado, processador do dado e sujeito proprietário do dado. Estes conceitos serão descritos a seguir:

a) Controlador de Dado: Uma entidade (pessoa física ou jurídica, autoridade pública, agência ou organização) que sozinha ou em conjunto com outros, determina a maneira e o propósito pela qual as informações pessoais são processadas.

b) Processador de Dado: Uma entidade (pessoa física ou jurídica, autoridade pública, agência ou organização) que processa as informações pessoais de acordo com as instruções do Controlador de Dado.

c) Sujeito do Dado: Um indivíduo identificado ou identificável ao qual a informação pessoal se refere, seja por identificação direta ou indireta (por exemplo por referência a um número de identificação ou por um ou mais fatores físicos, psicológicos, mentais, econômicos, culturais ou sociais).

2.2. Conceitos Fundamentais

Nesta seção, iremos apresentar os principais conceitos relacionados à segurança e à privacidade em computação em nuvem.

2.2.1. Privacidade de Dados na Nuvem

Diversos estudos têm sido realizados para investigar os problemas relacionados à privacidade e segurança em ambientes de computação em nuvem. [Liu et al. 2012] estudou o assunto nas áreas de saúde e energia elétrica. Já o trabalho apresentado por [Gruschka and Jensen 2010] sugeriu modelar o ecosistema de segurança baseado em três participantes do ambiente de nuvem: o usuário do serviço, a instância do serviço e o provedor do serviço. Os ataques podem ser classificados em 6 categorias, conforme descrição na Tabela 2.1. Em cada categoria representa-se a origem e o destino dos ataques. Por exemplo, "usuário -> provedor" indica ataques de usuários a provedores de nuvem.

Tabela 2.1. Tipos de Ataques na Nuvem

	Usuário do Serviço	Instância do Serviço	Provedor de Nuvem
Usuário do Serviço		usuário →serviço	usuário → provedor
Instância do Serviço	serviço → usuário		serviço → provedor
Provedor de Nuvem	provedor → usuário	provedor → serviço	

[Spiekermann and Cranor 2009] classificou 3 domínios técnicos para o armazenamento de dados na nuvem: esfera do usuário, esfera da organização e esfera dos provedores de serviços. O autor relacionou áreas de atividades que causam grande preocupação em relação à privacidade de dados com as 3 esferas de privacidade, conforme ilustrado na Tabela 2.2, a seguir.

O *National Institute of Standards and Technology (NIST)* propõe uma terminologia para a classificação de problemas relacionados à privacidade e a segurança em ambientes de computação em nuvem. A terminologia proposta contém 9 áreas: governança, conformidade, confiança, arquitetura, gerenciamento de acesso e identidade, isolamento de *software*, proteção de dados, disponibilidade e resposta a incidentes [Jansen and Grance 2011]. Em relação à proteção de dados, o NIST recomenda que sejam avaliadas a adequação de soluções de gerenciamento

Tabela 2.2. Esferas de Influência Associadas às Preocupações com Privacidade de Dados. Adaptado de [Spiekermann and Cranor 2009].

Esfera de Influência	Preocupação com a Privacidade de Dados
Esfera do usuário	• Coleção e armazenamento de dados não autorizados • Acesso não autorizado a dados • Exposição de dados • Entrada indesejada de dados
Esfera da organização	• Exposição de dados • Mau julgamento a partir de dados parciais ou incorretos • Acesso não autorizado a dados pessoais • Uso não autorizado de dados por terceiros envolvidos na coleta dos dados ou por outras organizações com as quais os dados foram compartilhados
Esfera dos provedores de serviços de nuvem	• Uso não autorizado de dados por terceiros envolvidos na coleta dos dados • Uso não autorizado por outras organizações com as quais os dados foram compartilhados • Acesso não autorizado de dados pessoais • Erros acidentais ou deliberados em dados pessoais • Mau julgamento a partir de dados parciais ou incorretos • Combinação de dados pessoais, a partir de banco de dados diferentes para recriar o perfil de um sujeito

de dados do provedor de nuvem para os dados organizacionais envolvidos e a capacidade de controlar o acesso aos dados, para proteção dos dados em repouso, em movimento e em uso, incluindo o descarte dos dados.

2.2.2. Segurança de Dados na Nuvem

O *NIST Computer Security Handbook* define segurança computacional como sendo "a proteção conferida a um sistema de informação automatizado, a fim de atingir os objetivos propostos de preservação da integridade, disponibilidade e confidencialidade dos recursos do sistema de informação (incluindo *hardware*, *software*, *firmware*, informações/dados e telecomunicações)"[Guttman and Roback 1995]. Esta definição contém 3 conceitos chaves para segurança computacional: confidencialidade, disponibilidade e integridade.

Além dos riscos e ameaças inerentes aos ambientes tradicionais de TI, o ambiente de computação em nuvem possui seu próprio conjunto de problemas de segurança, classificados por [Krutz and Vines 2010] em sete categorias: segurança de rede, interfaces, segurança de dados, virtualização, governança, conformidade e questões legais. Os princípios fundamentais da segurança da informação: confidencialidade, integridade e disponibilidade, definem a postura de segurança de uma organização e influenciam os controles e processos de segurança que podem ser adotados para minimizar os riscos. Estes princípios se aplicam também aos processos executados na nuvem.

O processo de desenvolvimento e implantação de aplicações para a plataforma de computação em nuvem, que seguem o modelo *software* como um serviço (*Software as a Service - SaaS*), deve considerar os seguintes aspectos de segurança em relação aos dados armazenados na nuvem [Subashini and Kavitha 2011]:

a) Segurança dos dados: no modelo *SaaS*, os dados são armazenados fora dos limites da infraestrutura de tecnologia da organização, por isso o provedor de nuvem deve prover mecanismos que garantam a segurança dos dados. Por exemplo, isso pode ser feito utilizando técnicas de criptografia forte e mecanismos de ajuste preciso para autorização e controle de acesso.

b) Segurança da rede: os dados do cliente são processados pelas aplicações *SaaS* e armazenados nos servidores da nuvem. A transferência dos dados da organização para a nuvem deve ser protegida para evitar perda de informação sensível. Por exemplo, pelo uso de técnicas de encriptação do tráfego de rede, tais como *Secure Socket Layer (SSL)* e *Transport Layer Security (TLS)*.

c) Localização dos dados: no modelo *SaaS*, o cliente utiliza as aplicações *SaaS* para processar seus dados, mas não sabe onde os dados serão armazenados. Isto pode ser um problema, devido à legislação sobre privacidade em alguns países proibir que os dados sejam armazenados fora de seus limites geográficos. O que ocorre, por exemplo, em relação ao armazenamento de dados médicos na nuvem, em alguns países da União Européia.

d) Integridade dos dados: o modelo *SaaS* é composto por aplicações multi-inquilino hospedadas na nuvem. Estas aplicações utilizam interfaces baseadas em *API-Application Program Interfaces XML* para expor suas funcionalidades sob a forma de serviços *Web* (*web services*). Embora existam padrões para gerenciar a integridade das transações com *web services*, tais como *WS-Transaction* e *WS-Reliability*, estes padrões não são amplamente utilizados pelos desenvolvedores de aplicações *SaaS*.

e) Segregação dos dados: os dados de vários clientes podem estar armazenados no mesmo servidor ou banco de dados no modelo *SaaS*. A aplicação *SaaS* deve garantir a segregação, no nível físico e na camada de aplicação, dos dados dos clientes.

f) Acesso aos dados: o ambiente muilti-inquilino da nuvem pode gerar problemas relacionados à falta de flexibilidade de aplicações *SaaS* para incorportar políticas específicas de acesso a dados pelos usuários de organizações clientes do serviço *SaaS*.

2.2.3. Vulnerabilidades da Nuvem

[Zhifeng and Yang 2013] relaciona características da nuvem que causam vulnerabilidades de segurança e privacidade, as quais são descritas a seguir:

a) Máquinas virtuais de diferentes clientes compartilhando os mesmos recursos físicos (*hardware*) possibilitam ataque de canal lateral (*side-channel attack*), situação em que o atacante pode ler informações do *cache* da máquina e descobrir o conteúdo de chaves criptográficas de outros clientes.

b) Perda de controle físico da máquina pelo cliente, que não pode se proteger contra ataques e acidentes. Por exemplo: alteração ou perda de dados.

c) Sub-provisionamento da largura de banda da rede, o que provocou o surgimento de um novo tipo de ataque de negação de serviço (*DOS- Denial of Service*) que se aproveita do fato da capacidade de rede do provedor de nuvem ser menor do que a quantidade de máquinas alocadas na mesma sub-rede [Liu 2010].

2.3. Modelos de Preservação da Privacidade

As soluções propostas para preservação da privacidade de dados armazenados ou processados na nuvem, que são discutidas neste documento, são classificadas em quatro categorias. A primeira categoria trata da proteção da privacidade de dados privados que devem ser disponibilizados na nuvem de forma pública. Neste contexto, pode-se aplicar os modelos de anonimização apresentados na Seção 2.4. A segunda categoria se refere a situações nas quais deseja-se realizar consultas sobre dados criptografados disponibilizados na nuvem, sem revelar o conteúdo destes dados, nem o conteúdo da consulta para o provedor de nuvem. Para este cenário, a técnica de Busca Criptográfica, apresentada na Seção 2.5, pode ser aplicada. A terceira categoria engloba os cenários nos quais é necessário assegurar a privacidade do acesso na recuperação de dados armazenados na nuvem. Neste contexto, pode-se aplicar a a técnica de proteção *PIR-Private Information Retrieval*, discutida na Seção 2.6. A quarta categoria, trata do problema de manter a privacidade em transações distribuidas na nuvem. Neste contexto, a técnica *SMC-Secure Multiparty Computation*, apresentada na Seção 2.7, pode ser utilizada. Por fim, a Seção 2.8 apresenta uma proposta de uma nova técnica para assegurar a privacidade de dados armazenados na nuvem, utilizando decomposição e fragmentação de dados.

2.4. Anonimização de Dados na Nuvem

Organizações públicas e privadas têm, cada vez mais, sido cobradas para publicar seus dados "brutos"em formato eletrônico, em vez de disponibilizarem apenas dados estatísticos ou tabulados. Esses dados "brutos"são denominados microdados (*microdata*). Neste caso, antes de sua publicação, os dados devem ser “sanitizados”, com a remoção de identificadores explícitos, tais como nomes, endereços e números de telefone. Para isso, pode-se utilizar técnicas de anonimização.

O termo anonimato, que vem do adjetivo "anônimo", representa o fato do sujeito não ser unicamente caracterizado dentro de um conjunto de sujeitos. Neste caso, afirma-se que o conjunto está anonimizado. O conceito de sujeito refere-se a uma entidade ativa, como uma pessoa ou um computador. Conjunto de sujeitos pode ser um grupo de pessoas ou uma rede de computadores [Pfitzmann and Köhntopp 2005]. Um registro ou transação é considerada anônima quando seus dados, individualmente ou combinados com outros dados, não podem ser associados a um sujeito particular [Clarke 1999].

Os dados sensíveis armazenados em sistemas de banco de dados relacionais sofrem riscos de divulgação não autorizada. Por este motivo, tais dados precisam ser protegidos. Os dados são normalmente armazenados em uma única relação r, definida por um esquema relacional R(a_1,a_2,a_3,...,a_n), onde a_i é um atributo no domínio D_i, com i = 1,..,n. Na perspectiva da divulgação de dados de indivíduos, os atributos em R podem ser classificados da seguinte forma [Camenisch et al. 2011]:

a) Identificadores: atributos que identificam unicamente os indivíduos (ex.: CPF, Nome, Número da Identidade).

b) Semi-identificadores (SI): atributos que podem ser combinados com informações externas para expor alguns ou todos os indivíduos, ou ainda reduzir a incerteza sobre suas identidades (ex.: data do nascimento, CEP, cargo, função, tipo sanguíneo).

c) Atributos sensíveis: atributos que contêm informações sensíveis sobre os indivíduos (ex.: salário, exames médicos, lançamentos do cartão de crédito).

2.4.1. Operações de Anonimização

[Clarke 1999] conceitua privacidade da informação como sendo "o interesse que um indivíduo tem em controlar, ou, ao menos, influenciar significativamente, o conjunto de dados a seu respeito". Com o crescimento da oferta de serviços de armazenamento de dados e programas em nuvem, preocupações com segurança e privacidade dos dados tem requerido, dos provedores destes serviços, a implementação de estratégias para mitigar riscos e aumentar a confiança dos usuários. Existe a preocupação de que dados privativos coletados e armazenados em bancos de dados na nuvem estejam protegidos e não sejam visualizados por pessoas não autorizadas, citados na literatura como "bisbilhoteiros de dados", "espião de dados", intruso ou atacante [Duncan et al. 2001].

As técnicas atualmente existentes para a proteção de dados, (generalização, supressão, embaralhamento e perturbação), propostas pela comunidade acadêmica, podem ser utilizadas e/ou combinadas com o objetivo de anonimizar os dados. Essas técnicas são apresentadas a seguir:

a) Generalização: para tornar o dado anônimo, esta técnica substitui os valores de atributos semi-identificadores por valores menos específicos, mas semanticamente consistentes, que os representam. A técnica categoriza os atributos, criando uma taxonomia de valores com níveis de abstração indo do nível particular para o genérico. Como exemplo, podemos citar a generalização do atributo Código de Endereçamento Postal (CEP), o qual pode ser generalizado de acordo com os seguintes níveis: CEP (60.148.221) -> Rua -> Bairro -> Cidade -> Estado -> País.

b) Supressão: esta técnica exclui alguns valores de atributos identificadores e/ou semi-identificadores da tabela anonimizada. Ela é utilizada no contexto de bancos de dados estatísticos, onde são disponibilizados apenas resumos estatísticos dos dados da tabela, ao invés dos microdados [Samarati 2001].

c) Encriptação: esta técnica utiliza esquemas criptográficos normalmente baseados em chave pública ou chave simétrica para substituir dados sensíveis (identificadores, semi-identificadores e atributos sensíveis) por dados encriptados.

d) Perturbação (Mascaramento): esta técnica é utilizada para preservação de privacidade em *data mining* ou para substituição de valores dos dados reais por dados fictícios para mascaramento de bancos de dados de testes ou treinamento. A idéia geral é alterar randomicamente os dados para disfarçar informações sensíveis enquanto preserva as

características dos dados que são críticos para o modelo de dados. Duas abordagens comuns desta técnica são a randomização (*Random Data Perturbation - RDP*) e a condensação dos dados [Chen and Liu].

- Condensação de Dados: técnica proposta por [Aggarwal and Philip 2004], condensa os dados em múltiplos grupos de tamanhos predefinidos. Informações estatísticas sobre média e correlações entre diferentes dimensões de cada grupo são preservadas. Dentro de um grupo não é possível distinguir diferenças entre os registros. Cada grupo tem um tamanho mínimo k, que é o nível de privacidade obtido com esta técnica.
- *Random Data Perturbation (RDP)*: esta técnica adiciona ruídos, de forma randômica, aos dados numéricos sensíveis. Desta forma, mesmo que um bisbilhoteiro consiga identificar um valor individual de um atributo confidencial, o valor verdadeiro não será revelado. A maioria dos métodos utilizados para adicionar ruído randômico são casos especiais de mascaramento de matriz. Por exemplo, seja o conjunto de dados X, o conjunto Z dos dados randomizados é computado como $Z = AXB + C$, onde A é uma máscara de transformação de registro, B é um máscara de transformação de atributo e C é um mascara de deslocamento (ruído) [Domingo-Ferrer 2008, Muralidhar and Sarathy 1999].

O mascaramento de dados é utilizado para disponibilizar bases de dados para teste ou treinamento de usuários, com informações que pareçam reais, mas não revelem informações sobre ninguém. Isto protege a privacidade dos dados pessoais presentes no banco de dados, bem como outras informações sensíveis que não possam ser colocadas a disposição para a equipe de testes ou usuários em treinamento. Algumas técnicas de mascaramento de dados são descritas a seguir [Lane 2012]:

a) Substituição: substuição randômica de conteúdo por informações similares, mas sem nenhuma relação com o dado real. Como exemplo, podemos citar a substituição de sobrenome de família por outro proveniente de uma grande lista randômica de sobrenomes.

b) Embaralhamento (*Shuffling*): substituição randômica semelhante ao item anterior, com a diferença de que o dado é derivado da própria coluna da tabela. Assim, o valor do atributo A em uma determinada *tupla* c_1 é substituído pelo valor do atributo A em uma outra *tupla* c_n, selecionada randomicamente, onde $n \neq 1$.

c) *Blurring*: esta técnica é aplicada a dados numéricos e datas. A técnica altera o valor do dado por alguma percentagem randômica do seu valor real. Logo, pode-se alterar uma determinada data somando-se ou diminuindo-se um determinado número de dias, determinado randomicamente, 120 dias, por exemplo; valores de salários podem ser substituídos por um valor calculado a partir do valor original, aplicando-se, para mais ou para menos, uma percentagem do valor original, selecionada randomicamente, por exemplo, 10% do valor inicial.

d) Anulação/Truncagem (*Redaction/Nulling*): esta técnica substitui os dados sensíveis por valores nulos (*NULL*). A técnica é utilizada quando os dados existentes na tabela não são requeridos para teste ou treinamento.

2.4.2. Tipos de Ataque à Privacidade dos Dados

O controle de inferência em banco de dados, também conhecido como *Statistical Disclosure Control* (SDC), trata da proteção de dados que podem ser publicados sem revelar informações confidenciais que possam ser relacionadas a pessoas específicas aos quais os dados publicados correspondem. A proteção que as técnicas de SDC proporcionam, provocam, em algum grau, modificação nos dados publicados, dentro dos limites de nenhuma modificação (máxima utilidade para os usuários e nenhuma proteção dos dados) e encriptação de dados (máxima proteção e nenhuma utilidade para o usuário sem a chave criptográfica). O desafio para SDC é prover a proteção necessária e suficiente para as informações divulgadas com o mínimo de perda de informação possível [Domingo-Ferrer 2008]. Existem dois tipos de divulgação de informações que podem ocorrer em dados anonimizados: divulgação de identidade, que ocorre quando a identidade de um indivíduo pode ser reconstruída e associada com um registro em uma tabela; e divulgação de atributo, que ocorre quando o valor de um atributo pode ser associado a um indivíduo (sem necessariamente poder ser associado a um registro específico). Modelos de privacidade para proteção da divulgação de informações propostos por [Fung et al. 2010] são classificados em duas categorias, com base nos tipos de ataques possíveis:

a) A primeira categoria considera que a ameaça à privacidade dos dados ocorre quando um adversário consegue ligar um proprietário de dados a um registro da tabela, ou a um atributo sensível da tabela ou ainda à tabela inteira. Estes tipos de ataques são denominados de: ataque de ligação ao registro, ataque de ligação ao atributo e ataque de ligação à tabela.

b) A segunda categoria considera a variação no conhecimento do adversário antes e depois de acessar os dados anonimizados. Caso esta variação seja significativa, esta situação configura o ataque probabilístico.

A remoção ou encriptação de atributos do tipo **Identificadores** é o primeiro passo para anominização. A tabela resultante desta alteração é chamada tabela anonimizada. Para ilustrar os ataques e os modelos de anominização, considere a Tabela 2.3, que apresenta dados fictícios de infrações de trânsito. A Tabela 2.3 é anonimizada, utilizando-se as técnicas de generalização e supressão, sendo os atributos classificados da seguinte forma: atributos identificadores: Motorista, Número da Placa e CPF. Esses atributos serão suprimidos da tabela anonimizada; atributos semi-identificadores: data nascimento e data infração. Esses atributos serão generalizados; atributos sensíveis: tipo de infração e valor da multa.

A Tabela 2.4 ilustra o resultado do processo de anonimização aplicado sobre a Tabela 2.3. Os modelos de ataque (de ligação ao atributo, ao registro e à tabela) contra a tabela com dados anonimizados (Tabela 2.4) são apresentados a seguir.

2.4.2.1. Ataque de Ligação ao Atributo

Neste tipo de ataque, valores de atributos sensíveis são inferidos a partir dos dados anonimizados publicados. Caso o atacante saiba que o Sr. José Sá nasceu em 05/1978 e recebeu

Tabela 2.3. Dados Privados sobre Infrações de Trânsito

Número Placa	Motorista	CPF	Data Nascimento	Data Infração	Tipo Infração	Valor Multa
HXR-1542	José Pereira	258.568.856	14/03/1977	03/01/2013	1	170,00
HTS-5864	Jorge Cury	566.548.584	04/03/1977	03/01/2013	2	250,00
HUI-5846	Paula Maria	384.987.687	24/05/1977	03/01/2013	1	170,00
HTR-5874	Joatan Lima	054.864.576	20/04/1978	04/01/2013	1	170,00
HOI-6845	José Sá	244.684.876	22/05/1978	04/01/2013	2	250,00
HQO-5846	Kilvia Mota	276.684.159	13/05/1978	05/01/2013	2	250,00
HUY-8545	José Pereira	538.687.045	15/05/1978	05/01/2013	1	170,00

Tabela 2.4. Dados Públicos Anonimizados sobre Infrações de Trânsito

Número Placa	Motorista	CPF	Data Nascimento	Data Infração	Tipo Infração	Valor Multa
*	*	*	03/1977	01/2013	1	170,00
*	*	*	03/1977	01/2013	2	250,00
*	*	*	05/1977	01/2013	1	170,00
*	*	*	04/1978	01/2013	1	170,00
*	*	*	05/1978	01/2013	2	250,00
*	*	*	05/1978	01/2013	2	250,00
*	*	*	05/1978	01/2013	1	170,00

uma multa de trânsito em 01/2013, por exemplo, analisando a Tabela 2.4, ele pode inferir com 2/3 de confiança que o valor da multa paga pelo Sr. José Sá foi de R$ 250,00, conforme descrito na Tabela 2.5. Para evitar este ataque, a estratégia geral é diminuir a correlação entre os atributos sensíveis e os atributos semi-identificadores.

Tabela 2.5. Ataque de Ligação ao Atributo

Data_Nascimento	Data_Infração	Tipo_Infração	Valor_Multa
03/1977	01/2013	1	170,00
03/1977	01/2013	2	250,00
05/1977	01/2013	1	170,00
04/1978	01/2013	1	170,00
05/1978	01/2013	2	250,00
05/1978	01/2013	2	250,00
05/1978	01/2013	1	170,00

2.4.2.2. Ataque de Ligação ao Registro

Neste tipo de ataque, registros com os mesmos valores para um determinado conjunto de atributos semi-identidicadores formam um grupo. Se os valores dos atributos semi-identificadores estiverem vulneráveis e puderem ser ligados a um pequeno número de registros no grupo, o adversário poderá identificar que um determinado registro refere-se a um indivíduo (vítima) particular.

Por exemplo, considere a Tabela 2.6, a qual contém dados de infrações de trânsito. Suponha que a Secretaria da Fazenda do Ceará - SEFAZ publicou uma relação dos proprietários de veículos de Fortaleza (Tabela 2.7). Supondo que cada pessoa com um registro na Tabela 2.6 tenha um registro na Tabela 2.7. O conjunto de registros dos atributos semi-identificadores (data nascimento e data infração) do grupo {05/1977,01/2013} possui apenas um registro. Neste caso é possível ligar este registro com o registro da Sra. Paula Maria na Tabela 2.7.

Tabela 2.6. Ataque de Ligação ao Registro na Tabela de Multas

Data Nascimento	Data Infração	Tipo Infração	Valor Multa
03/1977	01/2013	1	170,00
03/1977	01/2013	2	250,00
05/1977	01/2013	1	170,00
04/1978	01/2013	1	170,00
05/1978	01/2013	2	250,00
05/1978	01/2013	2	250,00
03/1985	01/2013	1	170,00

Tabela 2.7. Tabela dos Proprietários de Veículos (Dados Externos)

Número da Placa	Motorista	CPF	Data Nascimento
HXR-1542	José Pereira	258.568.856	14/06/1977
HTS-5864	Jorge Cury	566.548.584	04/06/1977
HUI-5846	Paula Maria	384.987.687	24/05/1977
HTR-5874	Joatan Lima	054.864.576	20/05/1978
HOI-6845	Leonardo Sá	244.684.876	22/05/1978
HQO-5846	Kilvia Mota	276.684.159	13/06/1977
HUY-8545	José Pereira	538.687.045	15/06/1977

2.4.2.3. Ataque de Ligação à Tabela

O ataque de ligação à tabela acontece quando o adversário consegue inferir a ausência ou a presença de um registro da vítima na tabela anonimizada. No caso de registros médicos ou financeiros, a simples identificação da presença do registro da vítima na tabela já pode causar prejuizo a ela.

Um exemplo deste tipo de ataque ocorre quando uma tabela anonimizada T (Tabela 2.8) é disponibilizada e o adversário tem acesso a uma tabela pública P (Tabela 2.7) em que T $\subseteq$ P. A probabilidade da Sra. Kilvia Mota Maria estar presente na Tabela 2.8 é de 3/4 = 0,75, uma vez que há 3 registros na Tabela 2.8 contendo a data de nascimento "06/1977"e 4 registros na Tabela 2.7 com data de nascimento "06/1977"

2.4.3. Modelos de Anonimização

Com a finalidade de evitar os ataques discutidos anteriormente, vários modelos de anonimização foram propostos por [Wong et al. 2010], [Tassa et al. 2012], [Last et al. 2014] e

Tabela 2.8. Ataque de Ligação à Tabela

Data Nascimento	Data Infração	Tipo Infração	Valor da Multa
06/1977	01/2013	1	170,00
06/1977	01/2013	2	250,00
06/1977	01/2013	1	170,00
04/1978	01/2013	1	170,00
05/1978	01/2013	2	250,00
05/1978	01/2013	2	250,00
03/1978	01/2013	1	170,00

[Gionis et al. 2008]. A seguir, são apresentados os principais modelos de anonimização encontrados na literatura: *k-anonymity*, *l-diversity*, *LKC-Privacy*, *t-closeness* e *b-likeness*.

2.4.3.1. k-anonymity

O modelo $k-anonymity$ requer que qualquer combinação de atributos semi-identificadores (grupo SI) seja compartilhada por pelo menos k registros em um banco de dados anonimizado [Samarati 2001], onde k é um valor inteiro positivo definido pelo proprietário dos dados, possivelmente como resultado de negociações com outras partes interessadas. Um valor alto de k indica que o banco anonimizado tem baixo risco de divulgação, porque a probabilidade de re-identificar um registro é de $1/k$, mas isto não protege o banco contra divulgação de atributos. Mesmo que o atacante não tenha capacidade de re-identificar o registro, ele pode descobrir atributos sensíveis no banco anonimizado.

[Samarati and Sweeney 1998] apresentam dois esquemas de transformação dos dados por generalização e supressão. O primeiro esquema substitui os valores de atributos semi-identificadores por valores menos específicos, mas semanticamente consistentes, que os representam. Como exemplo, pode-se trocar datas (dd/mm/aaaa) por mês/ano (mm/aaaa). A supressão é um caso extremo de generalização, o qual anula alguns valores de atributos semi-identificadores ou até mesmo exclui registros da tabela. A supressão deve ser utilizada como uma forma de moderação para a técnica de generalização, quando sua utilização provocar um grande aumento de generalização dos atributos semi-identificadores em um conjunto pequeno de registros com menos de k ocorrências. [Fung et al. 2007] propõe discretizar os atributos SI que apresentem valores contínuos, substituindo-os por um intervalo que contenha estes valores. Como exemplo, pode-se substituir o preço de produtos de supermercado por uma faixa de valores [1 a 3], [3 a 7], etc.

O esquema de generalização para o grupo *SI* (data nascimento, data infração) faz o mapeamento destes valores para um nível hierárquico mais genérico, conforme ilustrado na Tabela 2.9, em que vários valores diferentes de um domínio inicial são mapeados para um valor único em um domínio final.

Utilizando o modelo *k-anonymity* na Tabela 2.10 com o valor de $k = 2$ para o grupo SI = {data nascimento, data infração}, os registros 3 e 4 foram excluídos porque a

Tabela 2.9. Mapeamento de Valores entre Domínios

Domínio Inicial	Domínio Final
dd/mm/aaa	mm/aaaa
14/03/1967	03/1967
20/03/1967	03/1967
30/03/1967	03/1967

quantidade de registros dos grupos SI_1 = {05/1977,01/2014} e SI_2 ={04/1978, 01/2014} é menor do que k.

Tabela 2.10. Dados Públicos Anonimizados sobre Infrações de Trânsito

Número do Registro	Motorista	CPF	Data Nascimento	Data Infração	Tipo Infração	Valor da Multa
1	*	*	03/1977	01/2013	1	170,00
2	*	*	03/1977	01/2013	2	250,00
3 - excluído	*	*	05/1977	01/2013	1	170,00
4 - excluído	*	*	04/1978	01/2013	1	170,00
5	*	*	05/1978	01/2013	2	250,00
6	*	*	05/1978	01/2013	2	250,00
7	*	*	05/1978	01/2013	2	250,00

$k-anonymity$ não protege os atributos sensíveis de serem descobertos quando um grupo SI não possui diversidade nos valores de atributos sensíveis. Por exemplo, se Bob sabe que Alice foi multada em janeiro/2013 e que Alice nasceu em 1978, consultando a tabela anonimizada (registros 5, 6 e 7), Bob descobre que Alice recebeu uma multa do tipo 2 (avançar sinal vermelho), pois este é o único valor do atributo sensível "tipo de infração" do grupo SI ={05/1978, 01/2013} na Tabela 2.10.

O modelo $k-anonymity$ assume como pressuposto que cada registro representa apenas um indivíduo. O problema aqui posto é que se vários registros na tabela representarem um único indivíduo, um grupo de k registros pode representar menos do que k indivíduos, colocando em risco a proteção da privacidade de algum indivíduo. Para resolver esta questão, [Wang and Fung 2006] propuseram o modelo *(x,y)-anonymity* em que x e y representam conjuntos de atributos disjuntos, onde cada valor de x descreve um conjunto de registros (ex.: x =data nascimento) e está ligado a pelo menos k valores distintos de y (ex.: y = data infração). A associação entre x e y proposta pelo modelo dificulta a descoberta de atributos sensíveis.

2.4.3.2. l-diversity

O modelo $l-diversity$ proposto por [Machanavajjhala et al. 2006] captura o risco da descoberta de atributos sensíveis em um banco de dados anonimizado. o modelo $l-diversity$ requer que, para cada combinação de atributos semi-identificadores (grupo SI), deva existir pelo menos l valores "bem representados" para cada atributo sensível.

A definição de $l-diversity$ proposta por [Machanavajjhala et al. 2006] é a seguinte: um grupo *SI* é l-diverso se contiver pelo menos l valores bem representados para os atributos sensíveis. Uma tabela é l-diversa se cada grupo SI for l-diverso. Proporciona privacidade, mesmo quando não são conhecidas quais informações o atacante possui, pois garante a existência de pelo menos l valores de atributos sensíveis em cada grupo *SI*.

Dado um grupo *SI* = {data nascimento, data infração}, considere uma tabela a ser anonimizada onde existem registros de motoristas com datas de nascimento de 1960 a 1980, totalizando 240 meses e datas de infração do ano de 2013 (12 meses). Nesta tabela, podem existir no máximo 240 x 12 = 2.880 grupos *SI* distintos. Se for definido um limite $k = 5$ para a k-anonimização destes dados, cada grupo *SI* deverá ter pelo menos 5 registros com valores identicos para os atributos semi-identificadores. Por exemplo, utilizando-se o modelo $l-diversity$, os atributos sensíveis (tipo de infração e valor da multa) deverão ter l valores distintos em cada grupo *SI*. A Tabela 2.11 ilustra um exemplo dos registros do grupo SI = {03/1977,01/2013}. Se for definido o valor de $l = 3$, este grupo deverá ser excluído pois a diversidade dos atributos sensíveis tem apenas 2 valores distintos ({1,2},{170,250}) neste grupo.

Tabela 2.11. Grupo SI = 03/1977,01/2013

Data Nascimento	Data Infração	Tipo Infração	Valor da Multa
03/1977	01/2013	1	170,00
03/1977	01/2013	2	250,00
03/1977	01/2013	1	170,00
03/1977	01/2013	1	170,00
03/1977	01/2013	2	250,00
03/1977	01/2013	2	250,00
03/1977	01/2013	2	250,00

A interpretação do princípio de valores "bem representados" para os atributos de cada grupo *SI* pela métrica *l-diversity* originou 3 variações desta métrica: [Ninghui et al. 2007]

a) $l-diversity$ com valores distintos: neste caso existem l valores distintos para cada grupo *SI*. Um grupo *SI* pode ter um valor que apareça mais frequentemente que outros valores, possibilitando ao atacante re-identificar este valor ao sujeito que está presente no grupo *SI*. Este ataque é denominado ataque de probabilidade de inferência.

b) $l-diversity$ com entropia: neste caso, a entropia da tabela inteira deve ser pelo menos $log(l)$ e a entropia de cada grupo *SI* deve ser maior ou igual a $log(l)$. Esta definição é mais forte do que a definição anterior e pode ser muito restritiva se existirem poucos valores com alta frequência de ocorrência na tabela. A entropia de cada grupo *SI* é definida pelo indíce de diversidade de *Shannon*:

   $Entropia(SI) = -\sum_{s \in SI} f(s)log(f(s))$ onde $f(s)$ é a fração de registros do grupo *SI* que contem atributo sensível com valor igual a s.

c) (c,l)-*diversity* recursivo: Considere um grupo *SI* em que existem diversos valores para o atributo *S*, dado pelo conjunto $\{s_1,...,s_m\}$. Considere o conjunto de contagem destes valores (Ex.: 'r_1 = count(*) where S=s_1'), ordenados em ordem decrescente $\{r_1,...r_m\}$, neste caso, dada uma constante *c*, cada grupo *SI* satisfaz recursivamente (c,l)-*diversity* se $r_1 < c(r_l + r_{l+1} +...+ r_m)$. Este procedimento assegura que valores muito frequentes não apareçam tão frequentemente e que valores mais raros não apareçam tão raramente nos grupos *SI*. Um grupo *SI* satisfaz (c,l)-*diversity* recursivo se for possível eliminar um valor sensível e mesmo assim o grupo *SI* continuar *(c,l-1)-diverso*.

[Ninghui et al. 2007] apresenta alguns problemas do modelo $l-diversity$:

a) O modelo é limitado na pressuposição do conhecimento do adversário sobre os atibutos sensíveis. Por exemplo, não considera a possibilidade do adversário obter informações sobre um atributo sensível a partir da informação da frequência da distribuição global deste atributo na tabela.

b) O modelo assume que todos os atributos sensíveis são categorizados, desconsiderando atributos numéricos, nos quais, apenas pode ser suficiente a descoberta de valores aproximados.

Segundo [Ninghui et al. 2007], $l-diversity$ é vulnerável a dois tipos de ataques: ataque de assimetria (*Skewness attack*) e ataque de similaridade (*Similiarity attack*). Esses ataques são discutidos a seguir:

a) Ataque de assimetria: ocorre quando existe grande assimetria na distribuição dos valores dos atributos sensíveis. Por exemplo, um atributo com 2 valores em que existe 99% de ocorrencia de um valor e 1% de ocorrência do outro valor.

b) Ataque de similaridade: ocorre quando os valores em um grupo *SI* são distintos mas semanticamente equivalentes. Por exemplo, o atributo salário poderia ser discretizado por faixa de valores, mas as faixas de valores mais altas indicariam que os indivíduos ocupavam funções de chefia, enquanto faixas de valores mais baixas poderiam indicar pessoas recém-contratadas que ocupavam funções operacionais.

2.4.3.3. LKC-Privacy

Uma das maneiras de evitar o ataque de ligação de atributo consiste em utilizar a técnica de generalização de dados em grupos *SI*, de forma que cada grupo contenha *k* registros com os mesmos valores de semi-identificadores e diversificação dos atributos sensíveis para desorientar inferências do atacante sobre atributos da vítima conhecidos por ele [Fung et al. 2010]. O problema de aplicar esta técnica quando a quantidade de atributos semi-identificadores é muito grande é que a maior parte dos atributos tem que ser suprimida para se obter k-anonimização, o que diminue a qualidade dos dados anominizados. Este problema foi identificado por [Aggarwal 2005] e é conhecido como problema da alta dimensionalidade dos dados em k-anonimização.

[Mohammed et al. 2009] criaram o modelo $LKC-Privacy$ como uma proposta de solução para o problema da alta dimensionalidade dos dados. Este modelo parte do pressuposto de que o atacante não possui todas as informações dos atributos semi-identificadores do seu alvo. Neste caso, é razoável supor que o atacante possui conhecimento de pelo menos "L"valores de atributos semi-identificadores.

O modelo $LKC-Privacy$ assegura que cada combinação de valores de atributos SI em $SI_j \subseteq SI$ com tamanho máximo L em uma tabela T seja compartilhada por pelo menos K registros, e a confiança da inferência de qualquer valor sensível em S não seja maior do que C, onde L, K e C são limites e S é um conjunto de valores de atributos semi-identificadores. O modelo limita a probabilidade de sucesso na identificação do registro da vítima a ser menor ou igual a $1/K$ e a probabilidade de sucesso no ataque de ligação de atributo a ser menor ou igual a C, considerando que o conhecimento prévio do adversário não excede o valor de L.

[Fung et al. 2010] apresenta propriedades do modelo $LKC-Privacy$ que são adequadas para anominização de dados com alta dimensionalidade:

a) Requer que apenas um subconjunto de atributos semi-identificadores seja compartilhado por k registros. Este relaxamento da restrição tradicional de k-anonimização baseia-se na premissa de que o adversário tem limitado conhecimento dos atributos sensíveis da vítima.

b) Generaliza vários modelos tradicionais de k-anonimização. Por exemplo: *k-anonymity* é um caso especial de $LKC-Privacy$ onde L = Conjunto de Todos os Atributos semi-identificadores, $K = k$ e C = 100%; $l-diversity$ é um caso especial de $LKC-Privacy$ onde L= conjunto de todos os atributos semi-identificadores, $K = 1$ e $C = 1/l$.

c) É flexível para ajustar o dilema entre privacidade de dados e utilidade de dados. Aumentando L e K ou diminuindo C pode-se aumentar a privacidade, embora isso reduza a utilidade dos dados

d) É um modelo de privacidade geral que evita ataques de ligação de registro e ligação de atributos. É aplicável para anonimização de dados com ou sem atributos sensíveis.

A seguir, na Tabela 2.14, é apresentado um exemplo de anonimização utilizando o modelo $LKC-Privacy$ que satisfaz (2,2,50%)-privacidade pela generalização de todos os valores dos atributos SI da Tabela 2.12, de acordo com a taxonomia proposta na Tabela 2.13.

Cada possível valor de SI_j (SI_1 = {Sexo, Idade}, SI_2 = {Sexo, Data Infração}, SI_3 = {Idade, Data Infração}) com tamanho máximo igual a 2 na Tabela 2.14 é compartilhado por pelo menos 2 registros. Neste caso, a confiança do atacante inferir o valor sensível de tipo de infração = 1 não é maior do que 50%.

2.4.3.4. t-closeness

Este modelo propõe-se a corrigir algumas limitações de $l-diversity$ no que diz respeito à proteção contra divulgação de atributo. O objetivo é limitar o risco de descoberta a

Tabela 2.12. Multas de Trânsito do Mês de Janeiro/2014

	Semi-identificadores			Atributos Sensíveis	
Id	Sexo	Idade	Data Infração	Tipo Infração	Valor Multa
1	M	37	03/01/2013	1	170,00
2	F	22	30/01/2013	2	250,00
3	F	37	03/01/2013	1	170,00
4	M	18	13/01/2013	1	170,00
5	M	19	04/01/2013	2	250,00
6	M	36	05/01/2013	2	250,00
7	F	22	05/01/2013	1	170,00
8	F	47	20/01/2013	1	170,00

Tabela 2.13. Mapeamento de Valores entre Domínios

Sexo		Idade		Data Infração	
Domínio Inicial	Domínio Final	Domínio Inicial	Domínio Final	Domínio Inicial	Domínio Final
M	P	27	[1-30]	DD/MM/AAAA	MM/AAAA
F	P	57	[30-60]		

Tabela 2.14. Dados Anonimizados (L = 2, K = 2, C = 0,5)

	Semi-identificadores			Atributos Sensíveis	
Id	Sexo	Idade	Data Infração	Tipo Infração	Valor Multa
1	P	[30-60]	01/2013	1	170,00
2	P	[1-30]	01/2013	2	250,00
3	P	[30-60]	01/2013	1	170,00
4	P	[1-30]	01/2013	1	170,00
5	P	[1-30]	01/2013	2	250,00
6	P	[30-60]	01/2013	2	250,00
7	P	[1-30]	01/2013	1	170,00
8	P	[30-60]	01/2013	1	170,00

um nível aceitável. A técnica $t-closeness$ utiliza o conceito de "conhecimento global de retaguarda", que pressupõe que o adversário pode inferir informações sobre atributos sensíveis, a partir do conhecimento da frequência de ocorrência destes atributos na tabela.

Como os dados anonimizados disponibilizados devem conter a maior parte ou todos os registros da tabela original, é possível para o atacante calcular a medida da distribuição do atributo sensível em relação ao total de registros da tabela. Por exemplo, considere uma tabela anonimizada de registros de multas de trânsito em que foram disponibilizados 9.000 registros. A Tabela 2.15 apresenta a frequência do atributo "Tipo de Infração".

O modelo $t-closeness$ estima o risco de divulgação computando a distância entre a distribuição de atributos confidenciais dentro do grupo *SI* e a tabela inteira. Esta métrica requer que a distribuição de um atributo sensível em qualquer grupo *SI* seja um valor pró-

Tabela 2.15. Frequência do Atributo "Tipo de Infração"

Tipo de Infração	Quant. Registros	Frequência
1	1000	11%
2	1000	11%
3	4000	44%
4	3000	33%
Total	9000	100%

ximo do valor da distribuição do atributo em relação à tabela inteira [Ninghui et al. 2007]. Sendo Q a medida da distribuição do atributo sensível em toda a tabela e P a medida da distribuição do atributo sensível em um grupo SI, quanto mais próximas estas medidas estiverem, menor será o conhecimento que o atacante poderá ter sobre indivíduos específicos e maior será o grau de privacidade dos grupos SI. A distância entre as duas distribuições não pode ser maior que um limite t.

$t-closeness$ limita as possibilidades de um adversário obter informações sobre atributos sensíveis pela análise da distribuição de valores globais destes atributos. [Ninghui et al. 2007] sugere o uso da fórmula da distância variacional para calcular a distância entre $P = \{p_1,p_2,p_3,...,p_m\}$ e $Q = \{q_1,q_2,q_3,...,q_m\}$, definida pela medida *Earth-Mover Distance* (EMD), que mede a quantidade mínima de esforço necessário para mover uma distribuição de massa entre pontos de um espaço probabilístico [Liang and Yuan 2013]. O valor de EMD entre duas distribuições em um espaço normalizado é um número entre 0 e 1. A fórmula EMD é apresentada a seguir: $D[P,Q] = \sum_{i=1}^{m} \frac{1}{2}|p_i - q_i|$.

Por exemplo, considere o grupo SI = {03/1977,01/2013} da Tabela 2.16 que contém apenas infrações do tipo 1 e 2. A distribuição de frequência do atributo "tipo de infração"no grupo SI é P = {50%,50%,0%,0%}. A distribuição do atributo "tipo de infração"na tabela toda, de acordo com a tabela 2.16 é Q = 11%,11%,44%,33%. Calculando a distância entre P e Q utilizando EMD, obtem-se o valor de 0,775 (77,5%). Quanto maior for a distância entre P e Q, maior a probabilidade da descoberta de atributos sensíveis.

Tabela 2.16. Grupo SI = (03/1977,01/2013)

Data Nascimento	**Data Infração**	**Tipo Infração**	**Valor da Multa**
03/1977	01/2013	1	170,00
03/1977	01/2013	2	250,00
03/1977	01/2013	1	170,00
03/1977	01/2013	1	170,00
03/1977	01/2013	2	250,00
03/1977	01/2013	2	250,00

2.4.3.5. b-likeness

O modelo $b-likeness$, proposto por [Cao and Karras 2012], assegura que a confiança de um atacante no valor de um atributo sensível não aumenta em termos relativos, mais que um limite b pré-estabelecido, depois que o atacante tem conhecimento dos dados publicados.

A definição básica de $b-likeness$, formulada por [Cao and Karras 2012], é de que dada uma tabela T que contem atributos sensíveis (*sensitive attributes-SA*), seja $V = \{v_1,v_2,v_3,...,v_m\}$ o domínio de SA e $P = \{p_1,p_2,p_3,...,p_m\}$ a distribuição global de SA em T. Uma classe de equivalência G com distribuição de atributos sensíveis $Q = \{q_1,q_2,q_3,...,q_m\}$ satisfaz um limite básico $b-likeness$, se e somente se $max\{D(p_i,q_i)|p_i \in P, p_i < q_i\} \leq b$ onde $b > 0$ é um limite e D é uma função de distância entre p_i e q_i. A distância D deve ser grande o suficiente para proteger os dados de ataques de assimetria (*Skewness attack*) e de similaridade (*Similiarity attack*). Esta técnica difere das anteriores em relação ao uso da função de distância D para estabelecer o limite de distância máximo, ao invés da distância cumulativa entre os atributos sensíveis. É utilizada uma medida relativa, ao invés das medidas absolutas utilizadas pelas funções cumulativas de diferenças de frequências dos outros modelos anteriores. D é calculado pela fórmula $D(p_i,q_i) = \frac{p_i,q_i}{p_i}$

O modelo $b-likeness$ apresenta-se como uma solução ao problema da exposição de privacidade de valores de atributos sensíveis que ocorrem com menor frequência. Em geral, modelos de privacidade, como o *t-closeness*, que utilizam funções cumulativas de diferenças de frequências entre as distribuições não conseguem fornecer uma relação compreensível entre o limite t e a privacidade proporcionada pelo modelo. Tais modelos não dão atenção aos valores de atributos sensíveis que são menos frequentes e que são mais vulneráveis a exposição de privacidade.

A restrição imposta à função $D(p_i,q_i)$ de ser menor ou igual ao limite b, tem como consequência, a criação de um limite superior para a frequência de $v_i \in V$ em qualquer classe de equivalência G, conforme descrito na expressão $(q_i - p_i)/p_i \leq b \Rightarrow q_i \leq p_i \times (1+b)$. Esta função representa um limite de proteção de privacidade compreensível apenas se $p_i \times (1+b) < 1$, neste caso, valores de $p_i < 1/(1+b)$ devem ser monitorados, pois pode ocorrer de tais valores assumirem valor igual a 1 na classe de equivalência, tornando possível ao atacante, que saiba que o registro da vítima está presente na classe de equivalência, a inferência do valor atributo sensível com 100% de confiança.

2.5. Busca Criptográfica

Busca Criptográfica (*Searchable Encryption*) é uma técnica que provê funcionalidades de pesquisa em dados encriptados sem requerer a chave de encriptação. Esta técnica utiliza duas partes: um cliente e um servidor que armazena um banco de dados D encriptado, onde o cliente possui uma chave de acesso Q e a utiliza para obter o resultado da consulta $Q(D)$ sem revelar o texto e o resultado da consulta para o servidor. Uma chave de acesso é um conjunto de palavras codificadas que estão relacionadas a palavras-chaves associadas aos registros da tabela pesquisada no banco de dados. A consulta retornará os registros em que houver coincidência entre as palavras da chave de acesso Q e as palavras dos registros da tabela.

Como exemplo de um cenário de uso de busca encriptada, suponha que um determinado cliente deseja armazenar seus dados médicos criptografados em um banco de dados na nuvem, de forma que possa recuperar os registros seletivamente. O cliente associa um conjunto de palavras-chaves para cada registro da tabela, por exemplo tipo da doença. Para usar a busca criptográfica, o cliente criptografa o conjunto de palavras chaves que estão associadas aos registros da tabela. Os registros dos dados médicos também são criptografados usando algum esquema de criptografia padrão. As palavras-chaves e os dados médicos são armazenados em uma tabela no banco de dados. Para consultar registros que estejam associados com a palavra "diabetes", o cliente cria uma chave de consulta Q usando a palavra "diabetes"e envia a consulta para o servidor, que verificar cada palavra-chave da tabela e seleciona os registros onde existe correspondência entre a chave de consulta e a palavra-chace "diabetes", retornando estes registros para o cliente, caso existam. Neste caso, o servidor obtém a informação de quais registros foram retornados, mas não aprende nada sobre o conteúdo destes registros.

Esquemas de busca criptográfica podem utilizar esquemas criptográficos baseados em chave simétrica ou chave pública. Esquemas de chave pública são adequados para atributos multi-usuário, em que qualquer cliente pode encriptar os dados utilizando parâmetros públicos, mas somente um usuário pode realizar consultas aos dados. No esquema de chave simétrica, apenas o proprietário da chave secreta pode criar as palavras-chaves. A Tabela 2.17 mostra uma comparação entre os esquemas de criptografia de chave pública e chave simétrica.

Tabela 2.17. Comparação entre Esquemas de Busca Criptográfica. Fonte:[Sedghi 2012]

	Busca criptográfica com chave simétrica	Busca criptográfica com chave pública
Construção do texto cifrado pesquisável	Criado por uma chave secreta	Criado por parâmetros públicos
Gerenciamento da chave	Atributos de usuário único	Atributos de multiusuário
Funcionalidade	Busca por um palavra chave	Busca por uma palavra chave e decriptação parcial dos dados
Desempenho	Mais eficiente	Menos eficiente

2.6. Private Information Retrieval

Segundo [Yang et al. 2011a], para proteger a privacidade do padrão de acesso a dados, a intenção de cada operação de acesso a dados deve ficar escondida de forma que quem estiver observando a transação, não obtenha nenhuma informação significativa. *PIR - Private Information Retrieval* é uma técnica de consulta em bancos de dados públicos não criptografados com proteção à violação de privacidade de acesso dos usuários. Uma violação de privacidade de acesso ocorre quando, além de aprender as propriedades dos dados estatísticos agregados, o provedor de nuvem pode, com alta probabilidade de acerto, saber determinada informação privada do usuário a partir de dados criptografados armazenados. Protocolos *PIR* permitem que clientes recuperem informações de bancos de dados públicos ou privados sem revelarem para os servidores de banco de dados quais registros são

recuperados. [Olumofin and Goldberg 2012] argumentam que, pela proteção do conteúdo das consultas, *PIR* pode proteger importantes domínios de aplicações, tais como banco de dados de patentes, banco de dados farmaceuticos, censo *online*, serviços baseados em localização e análise de comportamento *online* para propaganda pela rede.

Um esquema *PIR* modela o banco de dados como uma *string* binária $x = x_1, x_2, x_3, \ldots, x_n$ de tamanho n. Cópias idênticas desta *string* são armazenadas em k servidores, sendo $k \geq 2$. Os usuários possuem um índice i (um inteiro entre 1 e n) e estão interessados em obter o valor do *bit* x_i fazem consultas aleatórias aos servidores e obtém respostas com as quais podem computar o bit x_i. As consultas realizadas aos servidores são distribuidas independentemente do valor de i para que os servidores não obtenham nenhuma informação sobre i. As consultas não recuperam necessariamente um *bit* em particular ou conjuntos de *bits*. Elas podem definir funções computadas pelos servidores, como por exemplo, uma consulta pode especificar um conjunto de índices entre 1 e n e a resposta do servidor pode ser o XOR dos *bits* que possuem estes índices.

O parâmetro de maior relevância nos esquemas *PIR* é a complexidade da comunicação entre o usuário e os servidores. Os protocolos mais eficientes para comunicaçao com 2 servidores têm complexidade de comunicação de $O(n^{\frac{1}{3}})$ [Chor et al. 1998]. Devido ao fato dos esquemas *PIR* utilizarem dados não criptografados, [Yang et al. 2011b] argumenta que eles não são adequados para uso em ambientes de nuvens não confiáveis.

2.7. SMC-Secure Multiparty Computation

SMC (*Secure Multiparty Computation*) é técnica de processamento distribuido de dados, com garantia de privacidade. No SMC, um conjunto de partes interessadas deseja avaliar alguma função de interesse comum ao grupo e para tal processa dados individuais privados sem revelar estes dados uns aos outros. Apenas a saída da função é disponibilizada para todas as partes. O processamento de dados de forma colaborativa é muitas vezes necessário em ambiente de nuvem. No processamento distribuído, as partes podem ser adversários passivos que tentam obter informação "extra"sobre os dados de outras partes. Neste método, cada cliente C_i possui uma entrada privada x_i, e todos os clientes computam uma função pública $f(x_1,x_2,x_3,...,x_n)$ sem revelar x_i para os outros, exceto o que pode ser derivado da entrada ou saída da função.

2.8. Anonimização por Decomposição

A criptografia é uma ferramenta útil para proteção da confidencialidade de dados sensíveis. Entretanto, quando os dados são encriptados, a realização de consultas se torna um desafio. Assim, embora a encriptação dos dados proporcione confidencialidade, os dados encriptados são muito menos convenientes para uso do que os dados descriptografados. Quando utilizada com bancos de dados relacionais, a criptografia cria dois grandes problemas. O primeiro problema é que os bancos relacionais requerem que os tipos de dados sejam definidos antes do seu armazenamento. O segundo problema é que consultas ou funções não podem ser executadas sobre dados criptografados. Não é possível avaliar faixas de datas ou fazer comparações de valores em dados criptografados. As estruturas de índice também não podem ser utilizadas.

Adicionalmente, os métodos baseados em criptografia precisam incluir estraté-

gias de geração e distribuição de chaves [Tian and Zhang 2012]. Porém, existem várias desvantagens relacionadas com a gestão de chaves criptográficas, tais como:

a) A necessidade de guardar as chaves pelo mesmo tempo em que os dados permanecerem criptografados.

b) A atribuição ou a revogação de chaves para o acesso aos dados por parte dos usuários.

c) A necessidade de manter múltiplas cópias encriptadas do mesmo arquivo, para acesso multi-usuário utilizando chave-pública.

Neste sentido, novas técnicas para assegurar a privacidade dos dados armazenados na nuvem, que não sejam baseadas em criptografia, tornam-se necessárias em diversos cenários de aplicação. Desta forma, esta seção apresenta uma estratégia para preservar a privacidade dos dados armazenados na nuvem, denominada "decomposição", que utiliza decomposição e dispersão de arquivos para separar dados em partes irreconhecíveis e armazená-las em servidores distribuídos na nuvem. Além disso, a abordagem proposta não criptografa os dados a serem armazenados e processados na nuvem.

A técnica de "decomposição"extrai informações dos arquivos de dados sobre quantidade, qualidade e medida. Os arquivos de dados são considerados objetos. Cada objeto, segundo Hegel, na doutrina do SER [HEGEL 1988], possui três características que o determinam: a qualidade, a quantidade e a medida. Em um arquivo de dados, a qualidade é representada pelas 256 combinações possíveis dos 8 *bits* que formam os *bytes* que compõem o arquivo. A quantidade é o número de vezes que cada *byte* é encontrado no arquivo e a medida é a ordem em que os *bytes* estão dispostos no arquivo. Por exemplo, em um arquivo de 256 *bytes* onde ocorrem apenas os *bytes* que representam as letras "A","B","C"e "D"em igual proporção, por exemplo:

Arquivo: "ABCDABCDABCDABCDABCDABCDABCD...ABCD"(256 bytes)
Quantidade: 64(A), 64(B), 64(C) ,64(D)
Qualidade: A,B,C,D
Medida: A (1^o,5^o,9^o,13^o...253^o), B (2^o,6^o,10^o,14^o...,254^o), C (3^o,7^o,11^o,15^o...,255^o), D (4^o,8^o,12^o,16^o,256^o)

A seguir, apresentamos as etapas que compõem a técnica de "decomposição":

1) O algoritmo de decomposição lê sequências de 256 *bytes* do arquivo de dados. Iremos nos referir de agora em diante a este conjunto de *bytes* como *I-Objeto*.

2) O algoritmo extrai as informações de qualidade, quantidade e medida do *I-Objeto*, armazenando estas informações em dois *arrays* com tamanho de 256 elementos cada um: o *array* de inteiros Quantidade-Qualidade[256] e o *array* de caracteres Medida[256]. Iremos nos referir a estes *arrays* como vetores daqui por diante.

3) O vetor Quantidade-Qualidade[256] irá armazenar, para cada um dos diferentes *bytes* existentes no *I-Objeto*, o número de vezes que este *byte* é encontrado no *I-Objeto*. Por exemplo, se o *byte* $00001111_2 = 15_{10}$ estiver presente 20 vezes no *I-Objeto*, o item Quantidade-Qualidade[15] será igual a 20. Caso o *byte* 15_{10} não estiver presente, o valor do item Quantidade-Qualidade[15] será igual a zero.

4) Para cada item do vetor Quantidade-Qualidade, o algoritmo de decomposição, converte o valor do item em uma sequência de *bits* '1', caso o elemento do vetor seja maior que zero. Exemplo: Quantidade-Qualidade[25]=3 $\Rightarrow$ VetorBits[25] = '111'. Caso o elemento do vetor Quantidade-Qualidade seja igual a zero, o VetorBits não irá armazenar nenhum valor.

5) Os itens do VetorBits são concatenados da seguinte forma: VetorBits[0]+'0'+ VetorBits[1]+'0'+,...+'0'+VetorBits[255], produzindo um vetor de 512 elementos, que é usado como entrada em uma função que lê o vetor em sequências de 8 itens e converte para a representação *ASCII* correspondente, criando uma sequência de 64 *bytes*, que é gravada no arquivo quantidade-qualidade.dec. Ex: '01000001' é convertido para a letra 'A'

6) O vetor de caracteres Medida[256] irá armazenar, para cada elemento do vetor Quantidade-Qualidade[256] > 0, a ordem em que os *bytes* aparecem no *I-Objeto*. A posição dos *bytes* irá variar de 0 a 255, representando do 1º ao 256º byte contido no bloco de dados. O vetor usará o valor decimal do *byte* para representar os valores das posições dos bytes do *I-Objeto*. A Tabela 2.18 mostra um exemplo em que o *byte-1* ocorre 3 vezes e o *byte-3* ocorre 1 vez no *I-Objeto* e não há ocorrência dos *bytes* 0, 2 e 255. O vetor Medida[256] é gravado no arquivo medida.dec.

Tabela 2.18. Exemplo de Preenchimento de Vetores de Qualidade-Quantidade e Medida

Itens Vetor Quantidade_Qualidade[256]	Itens Vetor Medida[256]
Quantidade_Qualidade[0]=0	
Quantidade_Qualidade[1]=3	Medida[1] = 5_{10} = 0000 0101_2 Medida[2] = 27_{10}=0001 1011_2 Medida[3] = 43_{10} =0010 1011_2
Quantidade_Qualidade[2]=0	
Quantidade_Qualidade[3]=1	Medida[4] = 54_{10} =0011 0110_2
...	
Quantidade_Qualidade[255]=0	

Os arquivos medida.dec e quantidade-qualidade.dec são armazenados em provedores de nuvem diferentes. Neste caso, cada um dos arquivos é insuficiente para reconstruir o arquivo original. Por exemplo, supondo que o provedor que possua o arquivo quantidade-qualidade.dec tentasse reconstruir um bloco de 256 *bytes* do arquivo original. Utilizando o método de força-bruta para tentar reconstruir a sequência de 256 *bytes* de um *I-Objeto*, a probabilidade do provedor descobrir a sequência correta dos *bytes*, conhecendo a quantidade e a qualidade é uma permutação repetida *P* de 256 elementos: Prob =

$1/P_{256}{}^{n1,n2,n3...}$, onde $n1, n2, n3...$ são os itens de quantidade e qualidade conhecidos. Para um *I-Objeto* com apenas 1 *byte* de qualidade ou quantidade, a probabilidade é de 1/256. Para um *I-Objeto* com 2 *bytes* diferentes, a probabilidade é de aproximadamente $1/10^{76}$. A medida que a quantidade de itens de quantidade ou qualidade aumenta, a probabilidade de descoberta da ordem dos *bytes* tende a zero. Com 10 *bytes* diferentes no *I-Objeto*, a probabilidade já chega a $1/10^{256}$. Para o provedor de nuvem que armazena o arquivo medida.dec, ou seja, a ordem em que os *bytes* estão dispostos no bloco, a probabilidade de recomposição do *I-Objeto*, utilizando força-bruta é 1/256!, ou seja, aproximadamente $1/10^{506}$. Quanto maior for o arquivo, maior será a dificuldade do atacante para reconstruí-lo.

As vantagens desta técnica sobre as técnicas convencionais que utilizam criptografia para garantir confidencialidade dos dados armazenados na nuvem são as seguintes:

a) Não utilização de chaves criptográficas.

b) Aplicabilidade da técnica para soluções *SaaS*, *PaaS* e *IaaS* sem que haja nenhuma alteração nas interfaces dos aplicativos do usuário.

c) A técnica pode ser aplicada a qualquer formato de dado armazenado (dados e programas).

d) Não há limitação máxima para o tamanho do arquivo a ser anonimizado.

e) A solução suporta expurgo de dados da nuvem, pois os arquivos disponibilizados em provedores distintos não revelam informações sobre os dados originais. Caso o usuário deixe a nuvem, os dados podem ser considerados automaticamente expurgados.

A recomposição do arquivo original é feita pela recuperação dos arquivos quantidade-qualidade.dec e medida.dec que estão armazenados em 2 provedores distintos na nuvem, que são utilizados como entrada para o algoritmo de remontagem do arquivo original. A complexidade computacional do algoritmo de recomposição é de $O(n)$. A complexidade de comunicação é de $O(\frac{5\times n}{4})$, devido a soma do tamanho dos arquivos quantidade-qualidade.dec e medida.dec ser 25% maior do que o tamanho do arquivo original.

2.9. Considerações Finais

Para que todo o potencial da computação em nuvem possa ser explorado pelas organizações, é de fundamental importância garantir a segurança e a privacidade dos dados armazenados na nuvem. Nos últimos anos, vários mecanismos para assegurar privacidade dos dados armazenados na nuvem têm sido propostos [Stefanov and Shi 2013, Li et al. 2013, Yang et al. 2013, Jung et al. 2013, Yeh 2013, Nimgaonkar et al. 2012]. Este minicurso discutiu o problema da privacidade dos dados armazenados e processados nos ambientes de computação em nuvem, bem como as principais abordagens atualmente existentes para solucionar este importante problema. Por fim, uma nova técnica para assegurar a privacidade dos dados processados e armazenados nos ambientes de computação em nuvem foi apresentada.

Referências

[Aggarwal 2005] Aggarwal, C. C. (2005). On k-anonymity and the curse of dimensionality. In *Proceedings of the 31st international conference on Very large data bases*, pages 901–909. VLDB Endowment.

[Aggarwal and Philip 2004] Aggarwal, C. C. and Philip, S. Y. (2004). *A condensation approach to privacy preserving data mining*, pages 183–199. Springer.

[Camenisch et al. 2011] Camenisch, J., Fischer-Hübner, S., and Rannenberg, K. (2011). *Privacy and identity management for life*. Springer.

[Cao and Karras 2012] Cao, J. and Karras, P. (2012). Publishing microdata with a robust privacy guarantee. *Proc. VLDB Endow.*, 5(11):1388–1399.

[Chen and Liu] Chen, K. and Liu, L. Privacy preserving data classification with rotation perturbation. In *Proceedings of the Fifth IEEE International Conference on Data Mining*, pages 589–592. IEEE Computer Society.

[Chor et al. 1998] Chor, B., Kushilevitz, E., Goldreich, O., and Sudan, M. (1998). Private information retrieval. *Journal of the ACM (JACM)*, 45(6):965–981.

[Clarke 1999] Clarke, R. (1999). Introduction to dataveillance and information privacy, and definition of terms.

[Domingo-Ferrer 2008] Domingo-Ferrer, J. (2008). *A survey of inference control methods for privacy-preserving data mining*, pages 53–80. Springer.

[Duncan et al. 2001] Duncan, G. T., Keller-McNulty, S. A., and Stokes, S. L. (2001). Disclosure risk vs. data utility: The ru confidentiality map. In *Chance*. Citeseer.

[Fung et al. 2010] Fung, B. C., Wang, K., Fu, A. W.-C., and Yu, P. S. (2010). *Introduction to Privacy-Preserving Data Publishing: Concepts and Techniques*. Chapman-Hall.

[Fung et al. 2007] Fung, B. C. M., Ke, W., and Yu, P. S. (2007). Anonymizing classification data for privacy preservation. *Knowledge and Data Engineering, IEEE Transactions on*, 19(5):711–725.

[Gionis et al. 2008] Gionis, A., Mazza, A., and Tassa, T. (2008). k-anonymization revisited. In *Data Engineering, 2008. ICDE 2008. IEEE 24th International Conference on*, pages 744–753. IEEE.

[Gruschka and Jensen 2010] Gruschka, N. and Jensen, M. (2010). Attack surfaces: A taxonomy for attacks on cloud services. In *Cloud Computing (CLOUD), 2010 IEEE 3rd International Conference on*, pages 276–279.

[Guttman and Roback 1995] Guttman, B. and Roback, E. A. (1995). *An introduction to computer security: the NIST handbook*. DIANE Publishing.

[HEGEL 1988] HEGEL, G. (1988). Enciclopédia das ciências filosóficas em epítome. 3 vols. lisboa.

[Hon et al. 2011] Hon, W. K., Millard, C., and Walden, I. (2011). The problem of 'personal data'in cloud computing: what information is regulated?—the cloud of unknowing. *International Data Privacy Law*, 1(4):211–228.

[Jansen and Grance 2011] Jansen, W. and Grance, T. (2011). Guidelines on security and privacy in public cloud computing. *NIST Special Publication*, pages 800–144.

[Jr et al. 2010] Jr, A. M., Laureano, M., Santin, A., and Maziero, C. (2010). Aspectos de segurança e privacidade em ambientes de computação em nuvem.

[Jung et al. 2013] Jung, T., Li, X.-y., Wan, Z., and Wan, M. (2013). Privacy preserving cloud data access with multi-authorities. In *INFOCOM, 2013 Proceedings IEEE*, pages 2625–2633.

[Krutz and Vines 2010] Krutz, R. L. and Vines, R. D. (2010). *Cloud security: A comprehensive guide to secure cloud computing*. Wiley. com.

[Lane 2012] Lane, A. (2012). Understanding and selecting data masking solutions: Creating secure and useful data.

[Last et al. 2014] Last, M., Tassa, T., Zhmudyak, A., and Shmueli, E. (2014). Improving accuracy of classification models induced from anonymized datasets. *Information Sciences*, 256:138–161.

[Li et al. 2013] Li, M., Yu, S., Ren, K., Lou, W., and Hou, Y. T. (2013). Toward privacy-assured and searchable cloud data storage services. *Network, IEEE*, 27(4):56–62.

[Liang and Yuan 2013] Liang, H. and Yuan, H. (2013). On the complexity of t-closeness anonymization and related problems. In *Database Systems for Advanced Applications*, pages 331–345. Springer.

[Liu 2010] Liu, H. (2010). A new form of dos attack in a cloud and its avoidance mechanism. In *Proceedings of the 2010 ACM workshop on Cloud computing security workshop*, pages 65–76. ACM.

[Liu et al. 2012] Liu, J., Xiao, Y., Li, S., Liang, W., and Chen, C. L. P. (2012). Cyber security and privacy issues in smart grids. *Communications Surveys & Tutorials, IEEE*, 14(4):981–997.

[Machanavajjhala et al. 2006] Machanavajjhala, A., Gehrke, J., Kifer, D., and Venkitasubramaniam, M. (2006). L-diversity: privacy beyond k-anonymity. In *Data Engineering, 2006. ICDE '06. Proceedings of the 22nd International Conference on*, pages 24–24.

[Mohammed et al. 2009] Mohammed, N., Fung, B. C., Hung, P. C., and Lee, C.-k. (2009). Anonymizing healthcare data: A case study on the blood transfusion service. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '09, pages 1285–1294, New York, NY, USA. ACM.

[Muralidhar and Sarathy 1999] Muralidhar, K. and Sarathy, R. (1999). Security of random data perturbation methods. *ACM Transactions on Database Systems (TODS)*, 24(4):487–493.

[Nimgaonkar et al. 2012] Nimgaonkar, S., Kotikela, S., and Gomathisankaran, M. (2012). Ctrust: A framework for secure and trustworthy application execution in cloud computing. In *Cyber Security (CyberSecurity), 2012 International Conference on*, pages 24–31.

[Ninghui et al. 2007] Ninghui, L., Tiancheng, L., and Venkatasubramanian, S. (2007). t-closeness: Privacy beyond k-anonymity and l-diversity. In *Data Engineering, 2007. ICDE 2007. IEEE 23rd International Conference on*, pages 106–115.

[Olumofin and Goldberg 2012] Olumofin, F. and Goldberg, I. (2012). *Revisiting the computational practicality of private information retrieval*, pages 158–172. Springer.

[Pearson 2013] Pearson, S. (2013). *Privacy, Security and Trust in Cloud Computing*, pages 3–42. Springer.

[Pfitzmann and Köhntopp 2005] Pfitzmann, A. and Köhntopp, M. (2005). Anonymity, unobservability, and pseudonymity—a proposal for terminology. In *Designing privacy enhancing technologies*, pages 1–9. Springer.

[Samarati 2001] Samarati, P. (2001). Protecting respondents identities in microdata release. *Knowledge and Data Engineering, IEEE Transactions on*, 13(6):1010–1027.

[Samarati and Sweeney 1998] Samarati, P. and Sweeney, L. (1998). Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression. Technical report, Technical report, SRI International.

[Sedghi 2012] Sedghi, S. (2012). *Towards provably secure efficiently searchable encryption*. University of Twente.

[Spiekermann and Cranor 2009] Spiekermann, S. and Cranor, L. F. (2009). Engineering privacy. *Software Engineering, IEEE Transactions on*, 35(1):67–82.

[Stallings 2007] Stallings, W. (2007). *Network security essentials: applications and standards*. Pearson Education India.

[Stefanov and Shi 2013] Stefanov, E. and Shi, E. (2013). Oblivistore: High performance oblivious cloud storage. In *Security and Privacy (SP), 2013 IEEE Symposium on*, pages 253–267.

[Subashini and Kavitha 2011] Subashini, S. and Kavitha, V. (2011). Review: A survey on security issues in service delivery models of cloud computing. *J. Netw. Comput. Appl.*, 34(1):1–11.

[Tassa et al. 2012] Tassa, T., Mazza, A., and Gionis, A. (2012). k-concealment: An alternative model of k-type anonymity. *Transactions on Data Privacy*, 5(1):189–222.

[Tian and Zhang 2012] Tian, M. and Zhang, Y. (2012). Analysis of cloud computing and its security. In *International Symposium on Information Technology in Medicine and Education (ITME)*, TIME '12.

[Wang and Fung 2006] Wang, K. and Fung, B. C. M. (2006). Anonymizing sequential releases. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '06, pages 414–423, New York, NY, USA. ACM.

[Wong et al. 2010] Wong, W. K., Mamoulis, N., and Cheung, D. W. L. (2010). Non-homogeneous generalization in privacy preserving data publishing. In *Proceedings of the 2010 ACM SIGMOD International Conference on Management of data*, pages 747–758. ACM.

[Yang et al. 2013] Yang, K., Jia, X., Ren, K., Zhang, B., and Xie, R. (2013). Dac-macs: Effective data access control for multiauthority cloud storage systems. *IEEE Transactions on Information Forensics and Security*, 8(11):1790–1801.

[Yang et al. 2011a] Yang, K., Zhang, J., Zhang, W., and Qiao, D. (2011a). A light-weight solution to preservation of access pattern privacy in un-trusted clouds. In *Proceedings of the 16th European Conference on Research in Computer Security*, ESORICS'11, pages 528–547, Berlin, Heidelberg. Springer-Verlag.

[Yang et al. 2011b] Yang, K., Zhang, J., Zhang, W., and Qiao, D. (2011b). *A light-weight solution to preservation of access pattern privacy in un-trusted clouds*, pages 528–547. Springer.

[Yeh 2013] Yeh, C.-H. (2013). A secure shared group model of cloud storage. In *Proceedings of the 2013 27th International Conference on Advanced Information Networking and Applications Workshops*, WAINA '13, pages 663–667, Washington, DC, USA. IEEE Computer Society.

[Zhifeng and Yang 2013] Zhifeng, X. and Yang, X. (2013). Security and privacy in cloud computing. *Communications Surveys & Tutorials, IEEE*, 15(2):843–859.

Sobre os Autores

Eliseu Castelo Branco Júnior é aluno do Curso de Doutorado em Ciências da Computação do MDCC-Mestrado e Doutorado em Ciências da Computação da UFC-Universidade Federal do Ceará e Professor do Centro Universitário Estácio do Ceará desde 2000. Possui graduação em Bacharelado em Filosofia pela Universidade Estadual do Ceará (1992),Especialização em Redes de Computadores(1994) e Mestrado em Informática Aplicada pela UNIFOR-Universidade de Fortaleza(2001). Possui e Especialização em Ciências da Computação(1995) pela UFC. Atualmente é Coordenador de Curso de Pos-graduação em Gestão de Projetos do Centro Universitário Estácio do Ceará. Possui experiência na área de Ciência da Computação, com ênfase em Sistemas de Informação, atuando nos seguintes temas: computação em nuvem, segurança da informação, privacidade de dados, gerência de projetos, multicritério, avaliação de qualidade e processos de software. Prof. Eliseu Castelo Branco Jr. é autor de artigos publicados em periódicos e conferências internacionais e nacionais.

Javam de Castro Machado possui graduação em Processamento de Dados pela Universidade Federal do Ceará (1987), mestrado em Ciências da Computação pela Universidade Federal do Rio Grande do Sul (1990), Diplome dEtudes Approfondies (Dea) em Informática - Universite de Grenoble I (1992) e doutorado em Informática também pela Universite de Grenoble I (1995). Foi diretor da Secretaria de Tecnologia de Informação da UFC por 8 anos. Atualmente é professor associado do Departamento de Computação da Universidade Federal do Ceará, Vice-diretor do Centro de Ciências da mesma Universidade e coordenador do Laboratório de Sistemas e Banco de Dados (LSBD). É também coordenador de vários projetos de pesquisa e desenvolvimento em Computação, além de atuar como pesquisador do Programa de Mestrado e Doutorado em Ciência da Computação da UFC. Prof Javam tem vários artigos publicados em veículos nacionais e internacionais e participa de projetos de cooperação internacional com universidades européias. No momento suas áreas de interesse são sistemas de banco de dados e computação em nuvens, além de sistemas distribuídos

José Maria da Silva Monteiro Filho é professor Adjunto do Departamento de Computação da Universidade Federal do Ceará, onde leciona nos cursos de graduação, mestrado e doutorado em Ciência da Computação. Possui graduação em Bacharelado em Computação pela Universidade Federal do Ceará (1998), mestrado em Ciência da Computação pela Universidade Federal do Ceará (2001) e doutorado em Informática pela Pontifícia Universidade Católica do Rio de Janeiro - PUC-Rio (2008). Tem mais de 15 anos de experiência na área de Ciência da Computação, com ênfase em Banco de Dados e Engenharia de Software, atuando principalmente nos seguintes temas: sintonia automática de bancos de dados, bancos de dados em nuvem, big data, data science e qualidade de software. Prof. José Maria Monteiro é autor de mais de 30 artigos publicados em periódicos e conferências internacionais e nacionais.