UNIVERSIDADE PRESBITERIANA MACKENZIE

NELSON HERMENEGILDO POLETTO

UTILIZANDO *WEB SERVICES* NA INTEGRAÇÃO DE SISTEMAS: UMA VISÃO DO CONHECIMENTO ELETRÔNICO DE CARGA – CT-e

São Paulo

2009

NELSON HERMENEGILDO POLETTO

UTILIZANDO *WEB SERVICES* NA INTEGRAÇÃO DE SISTEMAS: UMA VISÃO DO CONHECIMENTO ELETRÔNICO DE CARGA – CT-e

Monografia apresentada à Universidade Presbiteriana Mackenzie do curso de Especialização em Tecnologia da Informação – Análise de Sistemas como requisito parcial para obtenção do título de Especialista.

ORIENTADOR: Prof. Ms. Ubiraci Brasil Matta

São Paulo

2009

Dedico este trabalho a todas as pessoas que sabem dar valor às pessoas que procuram algum tipo de acolhimento.

AGRADECIMENTOS

À Deus, fonte de toda sabedoria, pela força e pela coragem que me concedeu. Na sua infinita bondade, enviou uma centelha de luz emanada pelo nascimento de Jesus Cristo para permanecer ao meu lado e ao lado de minha família e de meus amigos no percurso dessa jornada maravilhosa que é a vida.

Agradeço em particular minha esposa Elizete, incansável companheira, minha maior incentivadora, aos meus filhos Fernando e Suzana pela paciência e compreensão nos momentos mais difíceis.

Ao meu sogro e sogra, Rigolvino e dona Luzia que considero meus segundos pais, pelo prazer de tê-los como família.

Aos meus pais Althemistoncles e Eunice por tudo o que eles representam e pelos investimentos feitos ao longo da vida para culminar no ponto dessa pós-graduação.

Da minha querida irmã, Denilze acompanhados do Jose Luiz, Mayara e o Jan que mesmo a distância puderam colaborar.

Agradeço aos professores do Mackenzie que foram presentes em situações que foram necessárias para me posicionar sobre as direções corretas a seguir.

Ao prof. Ubiraci B. Matta pelo apoio e incentivo na realização desse trabalho.

RESUMO

Visa-se com esta monografia apresentar o conceito de *Web Services*, onde a plataforma adotada não interfere no modo com que os sistemas se comunicam. Apresentam-se, também, informações relevantes no que tange à segurança da *Web*, visto que, hoje em dia, em todos os sistemas a segurança é parte principal no planejamento de desenvolvimento ou manutenção de sistemas. Apresenta também informações do projeto do Conhecimento Eletrônico de Carga (CT-e), onde os participantes principais, os remetentes e destinatários, as transportadoras e o governo federal do Brasil estão imbuídos no objetivo de racionalizar e revolucionar o setor de cargas criando um documento fiscal eletrônico, facilitando uma série de obrigações fiscais. Perante os novos e sempre crescentes desafios da tecnologia da informação e da comunicação, os *Web Services* associado ao projeto Conhecimento Eletrônico de Carga proporciona uma evolução no campo dos sistemas que envolvem a operacionalidade da gestão de processos e controles, tanto no campo empresarial como no campo governamental.

Palavras-chave: Web Services, Segurança na Web, Conhecimento Eletrônico de Carga.

ABSTRACT

This paper presents the concept of Web Services, where the platform adopted did not interfere with the way the systems communicate. It also presents important information when it comes to Web security, since today, all systems security is the main part in the planning of development or maintenance of systems. It also details the project Eletronic Invoicing Transportation, where the main participants, the senders and receivers, carriers and the federal government of Brazil are imbued with the goal to streamline and revolutionize the industry of creating a document electronic facilitating a series of tax obligations. Based on the new and ever-growing challenges of information technology and communications, Web services associated with the project Eletronic Invoicing Transportation provides an evolution in the field of systems involving the operational management processes and controls, both in the industry and the plan government.

Keywords: Web Services, Web Security, Eletronic Invoicing.

LISTA DE FIGURAS

LISTA DE ABREVIATURAS

ASP	*Active Server Pages*
B2B	*Business to Business*
C++	*Classes*
CA	Autoridade de Certificação
CORBA	*Common Object Request Broker Architecture*
COTEPE	Comissão Técnica Permanente
CT-e	Conhecimento de Transporte Eletrônico
CTRC	Conhecimento de Transporte Rodoviário de Carga
DACTE	Documento Auxiliar do Conhecimento de Transporte Eletrônico
DCOM	*Distributed Component Object Model*
DMZ	*DeMilitarized Zone*
DTD	*Document Type Definition*
HTLM	*Hypertext Markup Language*
HTTP	*Hypertext Transfer Protocol*
MD5	*Message Digest*
RMI	*Remote Method Invocation*
SHA-1	*Secure Hash Algorithm*
SEFAZ	Secretaria de Fazenda
SINIEF	Sistema Nacional Integrado de Informações Econômico-Fiscais
SMTP	*Simple Mail Transfer Protocol*
SOAP	*Simple Object Access Protocol*
UDDI	*Universal Description Discovery and Integration*
URI	*Uniform Resource Identifier*
URL	*Uniform Resource Locator*
W3C	*World Wide Web Consortium*
WSDL	*Web Services Description Language*
WWW	*World Wide Web*
XML	*Extensible Markup Language*
XML Schema	*Extensible Markup Language Schema*

SUMÁRIO

1 INTRODUÇÃO

Na área do conhecimento, e porque não dizer, também na área de Tecnologia da Informação, os desafios são sempre constantes, seja na forma de manter os conhecimentos adquiridos através das experiências profissionais vividas, ou na condição de aprender novas tecnologias. Dentro desse contexto, o estudo do tema *Web Services* contribuirá para o enriquecimento e desenvolvimento em análise de sistemas. A escolha deu-se em virtude do apaixonante mundo da *World Wide Web*, onde temos uma infinidade de opções de estudo e aprimoramento profissional e pessoal.

O tema “Utilizando *Web Services* na Integração de Sistemas: Uma visão do Conhecimento Eletrônico de Cargas CT-e”, sucinta grande interesse por se tratar de uma tecnologia relativamente nova. O estudo científico a ser apresentado visa evidenciar a condição de aplicá-lo em qualquer empresa que se interesse pelo seu uso, ampliando assim minha experiência científica e profissional.

A integração de diferentes aplicações corporativas é um desafio constante que preocupam profissionais da área de Tecnologia da Informação, pois um dos maiores problemas atuais das grandes corporações que hospedam sistemas em várias plataformas, com diferentes arquiteturas tecnológicas e também em diferentes ambientes de servidores, são a comunicação desses sistemas entre si. A estratégia utilizada para unificação dos sistemas tem um elevado custo e demanda muito esforço e tempo, inviabilizando esse processo.

O desafio de integrar e disponibilizar informações de vários sistemas distribuídos, cuja definição é “uma coleção de computadores independentes que aparecem para seus usuários como um simples e coerente sistema” (TANENBAUM, 2003, p. 2), tornou a tecnologia de *Web Services* um dos caminhos para facilitar a integração de sistemas e disponibilização de informações para o mundo.

Nos últimos anos, o avanço tecnológico aliado à popularização do uso da Internet e o crescente desenvolvimento de sistemas distribuídos, na qual permite a utilização de diversas tecnologias no seu desenvolvimento, tais como, ASP, C++, JAVA, FORMS entre outras, resulta na necessidade de compartilhar as informações entre os usuários desses sistemas. Um dos focos de preocupação dos profissionais da Tecnologia da Informação é resolver o problema da interoperabilidade entre os sistemas, pois em grandes companhias há vários sistemas desenvolvidos em várias plataformas.

A arquitetura *two-tier*, cliente/servidor ainda é muito utilizada. Com o crescimento da programação orientada a objeto, surgiram também os softwares *middlewares*, que têm como principal função possibilitar que os sistemas possam ser desenvolvidos de uma maneira mais independente possível do *hardware* e do sistema operacional, fazendo com que o código da aplicação possa ser executado em outros computadores. Exemplos mais comuns de arquiteturas de objetos para computação distribuída são o CORBA, DCOM e RMI.

Diante deste cenário, surge no final da década de 1990, os *Web Services* para resolverem a comunicação, não só de sistemas distribuídos, como também, qualquer tipo de sistema desenvolvido. Existem várias definições para *Web Services*, uma delas é:

> um *Web Service* é um sistema de *software* identificado por uma URI (*Uniform Resource Identifier*), cujas interfaces públicas e associações, são definidas e descritas usando XML. Sua definição pode ser descoberta por outros sistemas de software. Estes sistemas podem então interagir com o serviço *Web* de uma forma prescrita pela sua definição, baseado em XML, utilizando mensagens transmitidas por protocolos da Internet (AUSTIN, 2004, p. 3).

A figura abaixo demonstra a evolução do número de *hosts* no Brasil nos últimos seis anos.

	2009	2008	2007	2006	2005	2004
Janeiro	14.678.982	10.151.592	7.422.440	5.094.730	3.934.577	3.163.349
Julho		9.572.594	8.264.709	6.508.431	4.392.693	3.485.773

Figura 1: Evolução do número de *Hosts* do Brasil
Fonte: CETIC-Evolução Hosts, 2009.

O crescimento dos *hosts* no Brasil teve uma evolução significativa entre os anos de 2006 a 2009. De 2006 a 2007 um crescimento de cerca 24,38%. De 2007 a 2008, crescimento de 36,77% e entre 2008 a 2009 uma forte expansão de 44,59%.

O conjunto das novas tecnologias e conceitos de interoperabilidade como a XML (*eXtensible Markup Language*), SOAP (*Simple Object Access Protocol*), WSDL (*Web Services Description Language*) e o UDDI (*Universal Description Discovery and Integration*) que compõem a arquitetura dos *Web Services*, surgiram para melhorar a comunicação entre sistemas que estão baseadas em várias plataformas e também em vários sistemas operacionais e em diferentes linguagens de programação. Este conceito é um diferencial muito grande, pois, acaba com os problemas de comunicação das aplicações distribuídas citadas acima.

Web Services é a tecnologia que viabiliza a comunicação entre sistemas bem como a interoperabilidade entre processos que são acessados através da *Web*. Sua tecnologia permite a utilização de processos totalmente padronizados. Vale ressaltar que o órgão que regulamenta as tecnologias que são aplicadas à *Web* é o *Wide Web Consortium* (W3C), que atualmente conta com cerca de 500 membros e fundado por Timothy John Berners-Lee, inventor da WWW (*World Wide Web*).

Com a crescente demanda do Governo Federal por gerir as informações fiscais, as Secretarias de Fazenda dos Estados e Receita Federal do Brasil, desenvolveram de forma integrada, o projeto denominado Conhecimento de Transporte Eletrônico (CT-e). Este projeto visa substituir gradativamente, o atual Conhecimento de Transporte (CTRC), que corresponde à Nota Fiscal de Serviços de Transportes, pelo Conhecimento de Transporte Eletrônico (CT-e). O conceito do Conhecimento Eletrônico de Transporte (CT-e) é definido pelo Manual de Integração – Contribuinte como sendo:

> um documento de existência exclusivamente digital, emitido e armazenado eletronicamente, com o intuito de documentar uma prestação de serviços de transportes, cuja validade jurídica é garantida pela assinatura digital do emitente e a Autorização de Uso fornecida pela administração tributária do domicílio do contribuinte. (BRASIL, 2008, p. 8)

De forma simplificada o modelo operacional do projeto segue os seguintes passos: A empresa emissora do CT-e gera um arquivo eletrônico contendo todas as

informações fiscais da prestação de serviços de transportes. Esse arquivo é assinado digitalmente, de forma a garantir a integridade dos dados nele contido e a autoria da empresa emissora. O arquivo, gerado em XML, é transmitido, pela Internet para Secretaria de Fazenda Estadual de jurisdição da empresa emitente, que faz a pré-validação do arquivo e o devolve com uma Autorização de Uso. Na ausência dessa Autorização, não há possibilidade de haver a prestação de serviço de transporte. Depois de recebido o CT-e, a Secretaria de Fazenda Estadual disponibiliza consulta, via Internet, para o tomador do serviço de transporte e outros legítimos interessados que tenham a chave de acesso do documento eletrônico.

O CT-e é ainda transmitido, pela Secretaria de Fazenda Estadual para a Receita Federal do Brasil, na qualidade de repositório nacional de todos os CT-e emitidos, e também para as Secretarias de Fazendas do destinatário, desde que seja de jurisdição diferente da emissora, além da Superintendência da Zona Franca de Manaus.

Os portais das Secretarias de Fazenda Estaduais disponibilizam vários *Web* Services para cada serviço solicitado, ou seja, recepção, cancelamento, inutilização de numeração, consulta de situação, carta de correção e consulta dos CT-e enviados.

O diagrama a seguir mostra o fluxo conceitual de comunicação entre o aplicativo do contribuinte, seja ele qual for, e o Portal da Secretaria de Fazenda Estadual:

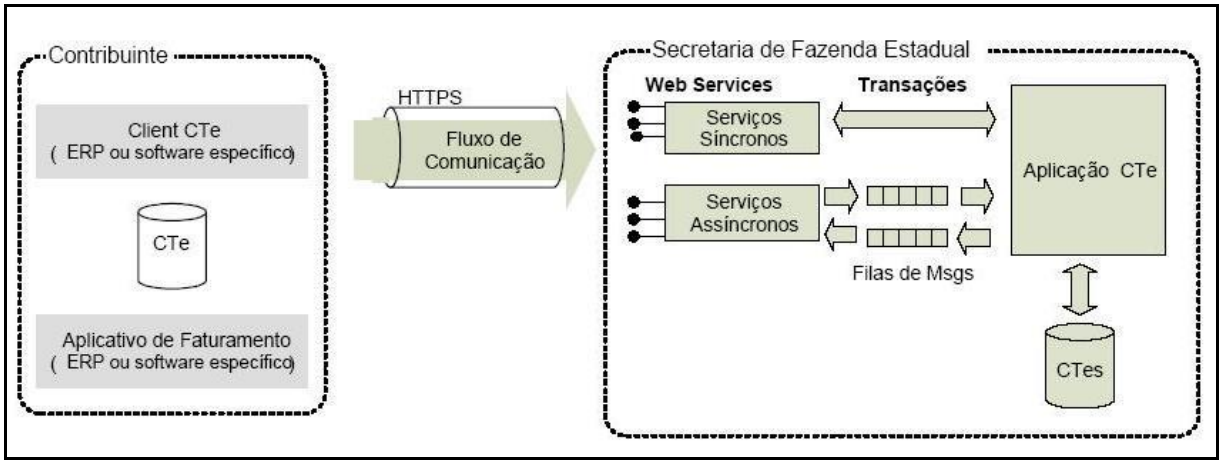


Figura 2: Arquitetura de Comunicação - Visão
Fonte: BRASIL (2008, p. 11).

Pretende-se ao longo dessa pesquisa científica, tornar conhecida a tecnologia dos *Web Services* e ampliar os níveis de conhecimento para profissionais que buscam esse aperfeiçoamento.

O objetivo principal dessa pesquisa científica é tornar a tecnologia dos *Web Services* mais difundida nos meios empresariais, sociais e científicos, demonstrando seus conceitos, benefícios e também sua utilização no projeto Conhecimento de Transporte Eletrônico CT-e.

O embasamento teórico deste trabalho está fundamentado em livros e artigos nacionais e internacionais, onde os autores na sua maioria mestres e doutores discorrem sobre os Web Services, Segurança na Web e também sobre o projeto do Conhecimento Eletrônico de Cargas.

Em síntese, diante do exposto, o foco principal desse trabalho é o *Web Services*, concomitantemente com o projeto Conhecimento de Transporte Eletrônico CT-e.

Esta monografia está organizada da seguinte forma: No capítulo 2 são apresentados os conceitos básicos dos *Web Services*, bem como sua arquitetura e tecnologias envolvidas; no capítulo 3 é abordado o tema de segurança na Web com conceitos sobre criptografia, protocolos criptográficos, assinaturas digitais, certificados digitais e autoridades certificadoras. Os *Web Services* se utilizam desses critérios para que tenham um funcionamento seguro; por fim, no capítulo 4 é discorrido sobre o tema do Conhecimento Eletrônico de Carga com seus conceitos, objetivos, estruturas e benefícios.

2 *WEB SERVICES* - CONCEITOS FUNDAMENTAIS

De acordo com ABINADER & LINS (2006) os *Web Services* nasceram naturalmente pelo uso em larga escala da Internet. Muitos concordam que essa utilização em massa leva a um processo de evolução desse meio entre comunicação entre pessoas e evolução também na arquitetura dos computadores e com isso leva naturalmente a possibilidade de escrever para um público maior e em grande escala.

No início, a Internet foi formada por páginas estáticas com muitas informações interligadas, sendo consultadas por usuários com utilização de programas chamados por navegadores *Web*, sendo popularmente denominado de *browser*, que em inglês significa navegador, e que tem por objetivo principal a visualização das páginas *Web*. Hoje em dia, os mais conhecidos *browsers* são os Internet Explorer, Firefox, Netscape, Opera e o Google Chrome (MORIMOTO, 2009).

Em um passo adiante na evolução da Internet, fez com que as páginas *Web*, que eram estáticas, fossem atualizadas para serem disponibilizadas e tornassem aptas a interagir, de chamarem outros programas geradores de informações dinâmicas, com acesso a banco dados e outras fontes. Além dessas dinâmicas, foi possível ao consumidor de conteúdos da Internet, interagir com mais informações, no que tange a inserir, alterar ou excluir informações, bem como possibilitar consultas de terceiros.

Nesta fase da Internet, criou-se o modelo de aplicações distribuídas, que é uma aplicação sendo executada simultaneamente em várias máquinas, sendo que os processos estão trabalhando de forma cooperativa e coordenada para realização de determinada tarefa (CIRNE, 2009). O modelo de aplicações distribuídas propiciou o surgimento de infra-estrutura de padrões e que incentivaram o aparecimento de aplicações ao cliente comum, com browser e funcionando sobre as camadas HTLM (*Hypertext Markup Language*) e HTTP (*Hypertext Transfer Protocol*). A solução de uso elaborada pelo modelo de aplicações distribuídas apresentou desvantagens por ter alto custo, de serem proprietárias e muito complexas, que afastaram as

empresas que desejavam se utilizarem da Internet como meio de aplicação nas transações comerciais. Gerou-se, então, uma necessidade de negócios que poderiam ser executados pela Internet. Demanda essa, definida como necessidade de trocar informações entre empresas de todos os portes que pudessem se comunicar através de padrões simples e públicos. A solução para essa demanda é o que se propõe a tecnologia de *Web Services* e a linguagem XML (*Extensible Markup Language*). Esse conjunto de tecnologia, hoje em dia, é considerado totalmente presente em toda Internet, pois é muito simples de ser utilizada.

De modo amplo pode-se conceituar os *Web Services* como uma tecnologia simples, constituída de softwares com baixo acoplamento, reutilizáveis e com componentes gerados para serem facilmente acessados pela Internet e que representam um modo de integração de tarefas de negócios entre pessoas, empresas e entre os próprios servidores *web*.

2.1 ARQUITETURA DOS *WEB SERVICES*

Segundo KREGER (2001) os *Web Services* baseiam-se numa arquitetura de interação de três funções: O provedor de Serviço (*service provider*), o consumidor de serviço (*service requestor*) e o registro de serviços (*service registry*). Nessa interação de funções, estão envolvidas as operações de publicação, pesquisa e ligação.

O provedor de serviços disponibiliza o *Web Services* para que seja utilizado por alguma pessoa ou empresa. O provedor de serviços, numa visão comercial é o proprietário do serviço e pela visão arquitetural é quem hospeda esse serviço. Para ter acessibilidade, uma descrição do serviço precisa ser publicada (*publish*), em um registro central, de modo que o serviço solicitante possa encontrá-lo.

O consumidor de serviços é toda e qualquer pessoa, empresa ou até mesmo um programa sem a interferência do usuário, nesse exemplo pode ser um outro *Web Service*, que faça uso do *Web Service* que está sendo solicitado, ou seja, consumido. Para fazer uso dessa funcionalidade deve-se conhecer o serviço a partir

da descrição disponibilizada pelo *Web Service* em um registro publicado, que serve também para obter o mecanismo de ligação com o *Web Services*.

RECKZIEGEL (2006) argumenta que o registro dos serviços é o ponto central de localização onde o provedor de serviços pode relacionar todos os seus *Web Services* e desse modo, o consumidor de serviços pode fazer pesquisas (*find*). O registro de serviços possui informações mais detalhadas como qual serviço disponível, dados da empresa, descrições técnicas e uma outra gama de informações.

Por consequência, o provedor de serviços define “do que se trata” o serviço e o publica no registro de serviços. O consumidor de serviços usa a descrição publicada para acessar o provedor de serviços e recuperar a informação solicitada (*bind*). A figura a seguir, ilustra o funcionamento de todo esse processo.

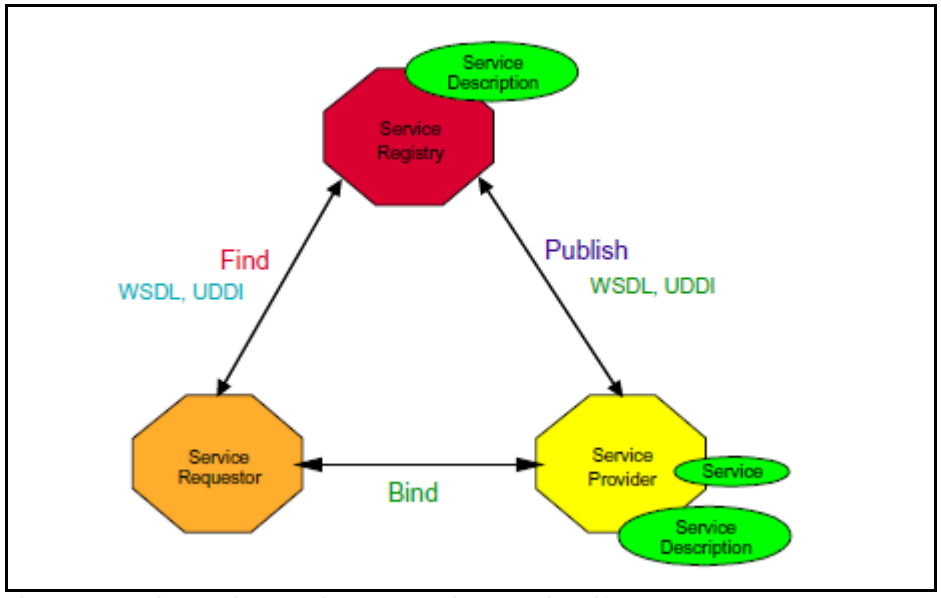


Figura 3: Web Services Roles, Operations and Artifacts
Fonte: KREGER, Heather (2001, p. 7).

2.2 TECNOLOGIAS UTILIZADAS - *WEB SERVICES*

Conforme ABINADER & LINS (2006) o padrão de linguagem dos *Web Services* é a XML *(Extensible Markup Language)* que faz parte do núcleo básico dessa arquitetura. A padronização segue com os componentes, constituídos pelas

tecnologias, WSDL *(Web Services Definition Language)*, SOAP *(Simple Object Access Protocol)* e UDDI *(Universal Discovery Description Integration)*.

2.2.1 XML (*Extensible Markup Language*)

RAY (2001) define XML (*Extensible Markup Language*) como sendo um protocolo de contenção e de gestão de informação, é constituída por uma família de tecnologias muito poderosas, onde é possível fazer formatação de documentos e até filtragem de dados. Possui uma filosofia de dados que dá tratamento com a máxima utilidade e flexibilidade para refinar informações, utilizando-se de sua forma estruturada.

Para JORGENSEN (2002) a linguagem XML é um padrão avançado para troca de informações, através da Internet, permitindo tags (marcadores) definidas pelo próprio usuário, o que permite tratamento muito flexível para documentos XML. As tags são termos ou palavras chaves relevantes para associação de alguma informação. Um arquivo texto contendo as informações, “F10 Shimano Calcutta 47.76 e F20 Bantam Lexica 49.99”, convertido no formado da linguagem XML, a interpretação torna-se mais simples. A figura 4 é um exemplo simples da linguagem XML, demonstrado a seguir.

```
<?xml version="1.0"?>
<Catalog>
        <Product>
                <ProductID>F10</ProductID>
                <ProductName>Shimano Calcutta </ProductName>
                <ListPrice>47.76</ListPrice>
        </Product>
        <Product>
                <ProductID>F20</ProductID>
                <ProductName>Bantam Lexica</ProductName>
                <ListPrice>49.99</ListPrice>
        </Product>
</Catalog>
```

Figura 4: Código em linguagem XML
Fonte: JORGENSEN, David (2002, p.63).

Percebe-se claramente nesse exemplo, que existe um catálogo com a identificação, nome e preço de produtos. Deste modo é perceptível que as informações estão

estruturadas, organizadas, identadas e aninhadas, de fácil compreensão e leitura. A linguagem XML resolve um problema complexo, na qual nos dá uma formatação simples para dados ou informações, além de possibilitar que os dados sejam mantidos independente dos processos que vão ser utilizados.

Os principais componentes de um documento XML são:

- Declaração: É uma declaração opcional, porém, recomendada pelo W3c. No exemplo citado é: <?xml version="1.0"?>;
- Comentário: Possibilita comentários. Instrução: <!– comentário –>;
- Esquema ou DTD (*Document Type Definition*): Em certas situações um esquema ou DTD pode anteceder o documento XML. Contém regras sobre os elementos do documento XML;
- Elementos: Um elemento contém tags de início e fim, e entre o início e fim temos o conteúdo dos elementos. No exemplo citado é: <ProductName>Shimano Calcutta </ProductName>;
- Elementos raiz: No documento XML deve aparecer somente um elemento raiz. No exemplo é: <Catalog>;
- Atributos: Fornece as informações adicionais aos elementos. No exemplo é: <ProductName> e <ListPrice>.

2.2.2 SOAP (*Simple Object Acess Protocol*)

Segundo GUDGIN, HADLEY & MENDELSOHN (2007), SOAP *(Simple Object Access Protocol)* é um protocolo leve, projetado para troca de informações estruturadas em ambiente distribuído e descentralizado. Utiliza a linguagem XML para definir e construir os quadros de mensagens. Esses quadros são concebidos para serem independentes de qualquer modelo particular de programação.

ALONSO (2004) afirma que o SOAP troca informações utilizando-se de mensagens. Essas mensagens são usadas como se fossem um envelope, onde a aplicação inclui, nesse envelope, as informações a serem enviadas. Cada envelope é formado por duas partes, um cabeçalho (*header*) e corpo (*body*).

O protocolo SOAP assume que toda mensagem tem um emissor, um destinatário e um número n, determinados por convenção, de intermediários, chamados de *nodes*. A figura a seguir ilustra a estrutura de mensagem no protocolo SOAP.

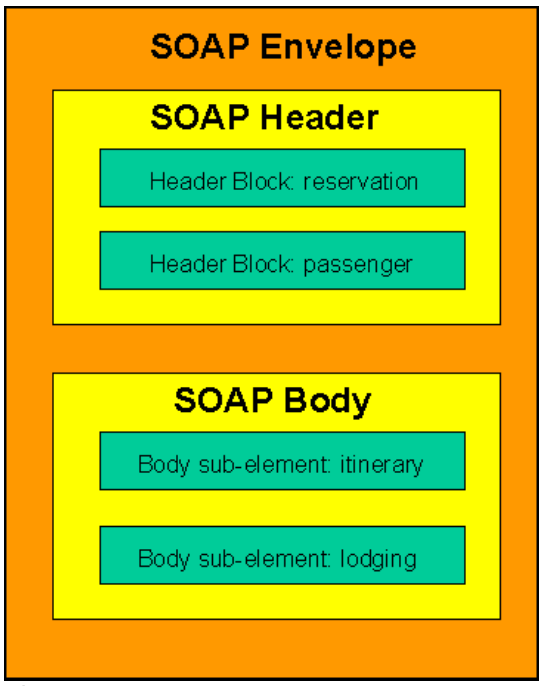


Figura 5: SOAP message structure
Fonte: W3C-SOAP (2007, p.8).

CRUZ (2005) aponta que o SOAP contém três partes principais, o modelo de empacotamento, onde está o conteúdo da mensagem SOAP, o mecanismo de serialização que é composto pelo conjunto de regras de codificação e definição dos tipos de dados a serem utilizados na aplicação, e o mecanismo de comunicação que define as chamadas e respostas de acordo com os procedimentos remotos. A mensagem SOAP é um documento XML e obrigatoriamente possui os elementos <Envelope>, <Body> e opcionalmente os elementos <Header> e <Fault>.

> O elemento <Envelope> é a raiz do documento XML e representa a mensagem propriamente dita. O elemento <Body> contém a carga de informações (operações e parâmetros) que são entregues ao destinatário da mensagem. Este elemento, de acordo com a especificação SOAP, pode conter um elemento <Fault>, que, quando presente, pode ser utilizado no

> processamento de falhas do serviço *Web*. O elemento <Fault> descreve erros de chamada de métodos remotos ou mantém informações acerca do tipo de erro. Portanto, ele conta com os elementos <FaultCode>, <FaultActor>, <FaultString> e <Detail>. O elemento <Header> expande uma mensagem SOAP. Ele define algumas características opcionais e acordos negociáveis entre as partes; seu conteúdo deve ser aceito pelas aplicações que estiverem se comunicando. (CRUZ, 2005).

No exemplo a seguir, o campo SOAPAction é utilizado por servidores de páginas ou *firewalls* para filtragem ou roteamento da mensagem. O campo SOAPAction ainda pode ser nulo ou então conter algum método SOAP.

```
POST /HopliasBooks HTTP/1.1
Host: localhost
Content-Type: text/xml; charset="utf-8"
Content-Length: nnnn
SOAPAction: http://equipe.nce.ufrj.br/serra
<soapenv:Header>
<soapenv:criptograph
soapenv:mustUnderstand="0" xsi:type="xsd:string">yes
</soapenv:criptograph>
<soapenv:priority
soapenv:mustUnderstand="0" xsi:type="xsd:string">high
</soapenv:priority>
</soapenv:Header>
<soapenv:Body>
<validateInfo
soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
<Username xsi:type="xsd:string">SergioSerra</Username>
<BirthDate xsi:type="xsd:string">03/02/1965</BirthDate>
<IDCardType xsi:type="xsd:string">CPF</IDCardType>
<IDCardNumber xsi:type="xsd:string">45608</IDCardNumber>
</validateInfo>
</soapenv:Body>
</soapenv:Envelope>
```

Figura 6: Mensagem SOAP com elementos <Envelope>, <Header> e <Body>
Fonte: CRUZ, Sérgio Manuel da Serra da (2005, p. 2).

2.2.3 WSDL (*Web Services Description Language*)

O W3C adotou a linguagem WSDL (*Web Services Description Language*) como sendo um padrão base de descrição. A linguagem WSDL especifica as características operacionais de um *Web Services* utilizando um documento gerado em XML, sustentando uma notação para saber, o que o *Web Service* faz, qual a localização e como ele pode ser acessado (PUTTE, 2004).

CHRISTENSEN (2001) identifica que um documento WSDL define serviços como sendo um conjunto de *endpoints*, ou seja, pontos de acesso na rede. Através desses *endpoints* são realizadas operações (*operations*), que são as trocas das mensagens (*messagens*) que utilizam os tipos (*types*). Um documento WSDL é constituído pelos elementos, *types*, *message*, *operation*, *port type*, *binding* e *port*, organizados em duas seções lógicas, a descrição abstrata (*abstract*) e descrição concreta (concrete).

PIRES (2005) enumera que a estrutura principal de um documento WSDL é composta do elemento <*definitions*> que contém os elementos:

- <*portType*>: O mais importante, pois, define no *Web Service* as operações que este pode executar e quais as mensagens envolvidas. Define o ponto de conexão com o *Web Service*. Este elemento pode servir de comparação a uma biblioteca de funções de uma linguagem de programação;

- <*message*>: Define os elementos de dados de uma determinada operação. Este elemento pode ter uma ou mais seqüências de informação. São igualadas aos parâmetros de uma função;

- <*types*>: São os tipos de informações que são utilizados pelo *Web Service*. Os tipos de dados são definidos, fazendo-se uso da sintaxe XML SCHEMA para definição de tipos de dados;

- <*binding*>: Descreve a forma de acesso ao serviço, por intermédio do protocolo SOAP. Possui dois atributos, o *name*, que é o nome do *binding*, e o *type*, que aponta para os *endpoints*;

- <*service*>: Define o *access point*, ou *endpoints* dos serviços.

Dentro do elemento <*portType*>, temos um padrão de troca de mensagens que são formados por quatro tipos de operações. As operações de *request-response*, que são as operações que podem receber uma requisição e retornar uma resposta. A

operação *one-way*, que é uma operação onde se pode receber uma mensagem e não retornar nenhuma resposta. A operação *solicit-response* que pode enviar uma requisição e espera por uma resposta. E por fim, a operação *notification*, que pode enviar uma mensagem e não aguarda por uma resposta. A figura 7 mostra um exemplo de um documento de WSDL.

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<definitions xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:SOAP-
ENC="http://schemas.xmlsoap.org/soap/encoding/" xmlns:tns="urn:server.hello"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" xmlns="http://schemas.xmlsoap.org/wsdl/"
targetNamespace="urn:server.hello">
<types>
<xsd:schema targetNamespace="urn:server.hello">
  <xsd:import namespace="http://schemas.xmlsoap.org/soap/encoding/" />
  <xsd:import namespace="http://schemas.xmlsoap.org/wsdl/" />
</xsd:schema>
</types>
<message name="helloRequest">
  <part name="name" type="xsd:string" />
</message>
<message name="helloResponse">
  <part name="return" type="xsd:string" />
</message>
<portType name="server.helloPortType">
<operation name="hello">
  <documentation>Retorna o nome</documentation>
  <input message="tns:helloRequest" />
  <output message="tns:helloResponse" />
</operation>
</portType>
<binding name="server.helloBinding" type="tns:server.helloPortType">
  <soap:binding style="rpc" transport="http://schemas.xmlsoap.org/soap/http" />
<operation name="hello">
  <soap:operation soapAction="urn:server.hello#hello" style="rpc" />
<input>
  <soap:body use="encoded" namespace="urn:server.hello"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />
  </input>
<output>
  <soap:body use="encoded" namespace="urn:server.hello"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />
  </output>
  </operation>
  </binding>
  <service name="server.hello">
  <port name="server.helloPort" binding="tns:server.helloBinding">
    <soap:address location="http://localhost/imasters2/nuSOAP/server2.php" /> </port>
  </service>
  </definitions>
```

Figura 7: Documento WSDL
Fonte: RECKZIEGEL, Mauricio (2006, p. 6).

2.2.4 UDDI (*Universal Description Discovery & Integration*)

CLEMENT (2004) aponta que os *Web Services* são funcionais e executáveis, se e somente se, os mesmos são acessíveis e possam ser encontrados por outros *Web Services*. O foco principal do UDDI (*Universal Description Discovery & Integration*) é a definição de um conjunto de serviços de apoio à descrição e descoberta de empresas, organizações e outros prestadores de serviços da *Web*, para que estes serviços sejam disponibilizados na *Web*, bem como as interfaces técnicas que podem ser utilizados para acessar tais serviços. Baseado num conjunto comum de padrões da indústria de software, incluindo HTTP, XML, XML Schema e SOAP, UDDI fornece uma interoperabilidade fundamental para infra-estrutura de *Web Services* baseados em software para ambiente públicos, bem como os ambientes expostos internamente dentro de uma organização.

ABINADER & LINS (2006) sustenta que o UDDI foi criado e implementado pelas maiores indústrias, lideradas pelas empresas Ariba, IBM e Microsoft para oferecer soluções rápidas que dificultavam a adoção da Internet nas transações em B2B (*Business to Business*). O UDDI é formado por um conjunto de especificações que define o que deve ser feito e como deve ser feito, livre da plataforma, para que os negócios possam fazer a descrição publicamente dos serviços disponíveis. Operacionalmente são procedimentos que propõem organizações identificarem outras organizações para que possam efetuar transações de negócio com muita rapidez. No segmento da indústria procura atender as necessidades do B2B, ou seja, padrões apropriados para condução de negócios na Internet.

Um registro UDDI possui informações sobre os dados das organizações, seus negócios e serviços. Em seu diretório, o UDDI contém referências para acesso dos serviços oferecidos pelos *Web Services*, sustentados por segmentações, taxonomias, sistemas de identificação, esquemas de identificação D-U-N-S e outros, utilizados para classificar e categorizar empresas, segmentos de atividades para que os serviços oferecidos sirvam de referencia para um banco de busca de outros *Web Services*. A literatura compara o UDDI a uma lista telefônica, onde temos as páginas brancas, as amarelas e as verdes. Nas páginas brancas temos as informações da empresa, seu endereço e telefones. Para se ter acesso ao número do telefone,

realiza-se uma consulta informando o nome da empresa. As páginas amarelas contêm as informações dos serviços prestados pela empresa. Assim quando realizarmos uma consulta por um determinado tipo de serviço encontraremos muitas empresas que prestam o serviço pesquisado. A classificação das empresas são baseadas na taxionomia padrão na qual a empresa pertence e também possuem taxionomia geográfica para possibilitar seu georeferenciamento. Nas páginas verdes encontram-se uma lista completa dos serviços oferecidos, seguidos de referencias para os processos nelas realizadas.

Quatro tipos de informações compõem o registro UDDI. As entidades de negócios, os serviços de negócios, especificação de ponteiros e tipos de serviços. A figura 8 detalha o conteúdo de cada informação:

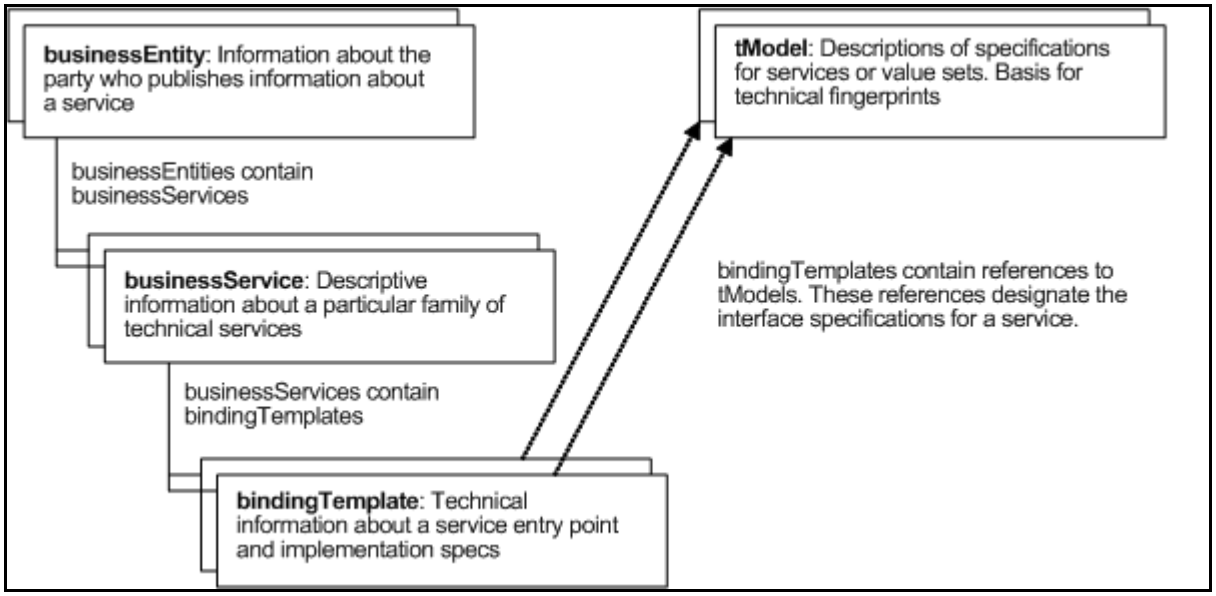


Figura 8: UDDI core data structures
Fonte: CLEMENT, Luc (2004, p. 27).

As entidades de negócios (*businessEntity*) são constituídas de identificador único, informações básicas, tais como, descrição breve dos serviços, informações simples de contato e descrição resumida dos negócios. Os serviços de negócios (*businessService*) fornece a descrição do serviço disponibilizado, lista de categorias, que permite classificação do serviço. Cada entidade de serviços de negócios inclui uma URL para ligação (*bindingTemplate*) e busca de informações técnicas adicionais a partir de modelos (*templates*). Os tipos de serviços (*Service Type, Technical Model*), abreviado por tModel contém a tipificação dos serviços, o nome

da organização, as categorias dos serviços, ponteiros para especificação de serviços, definição de interface, protocolos de mensagens e formatos e por fim os protocolos de segurança.

A seguir, um exemplo muito didático para abstrair como os *Web Services* funcionam (SOTOMAYOR, 2005). No propósito de manter e distribuir uma base de informações das condições meteorológicas dos Estados Unidos é disponibilizado um *Web Service* Meteorológico, que será acessado através de um código postal. A figura 9 ilustra as etapas e tecnologias para acesso da informação das condições climáticas:

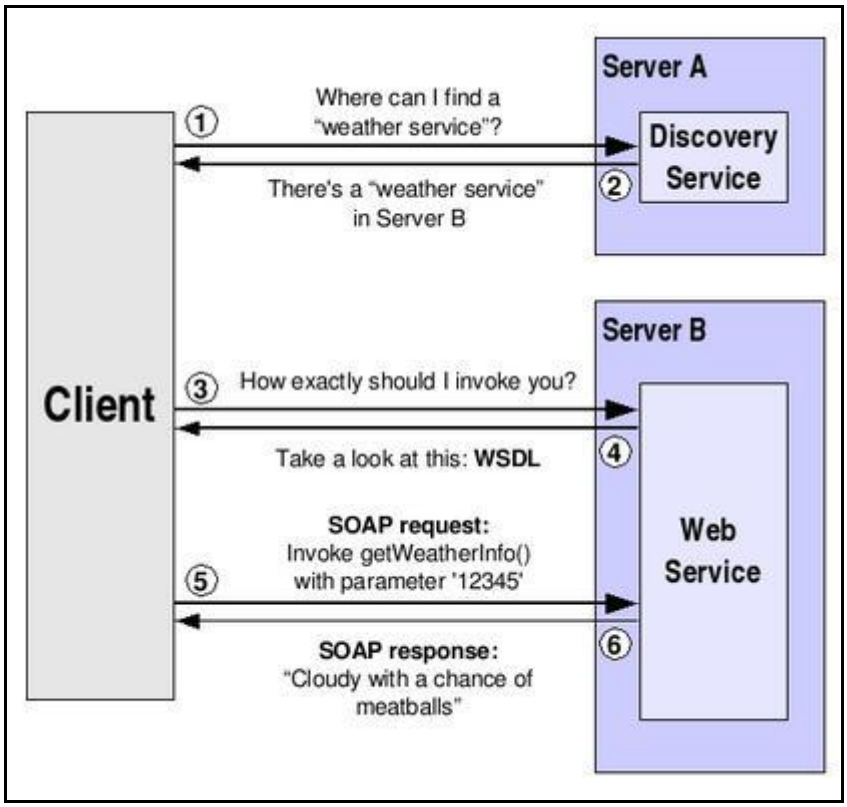


Figura 9: Como funcionam os *Web Services*
Fonte: SOTOMAYOR, Borja (2005, p. 3).

- Passo 1: O cliente pode não conhecer o *Web Service* que vai recorrer, sendo assim, é necessário descobrir qual *Web Service* atende suas necessidades. No exemplo, precisa ser encontrado um serviço que dê a previsão do tempo. Esse é também um *Web Service*, porém, de descoberta.

- Passo 2: A descoberta de serviço irá responder ao cliente, dizendo que servidores pode nos fornecer o serviço solicitado.

- Passo 3: Agora que é conhecida qual a localização do *Web Service*, é necessário perguntar ao *Web Service*, qual exatamente é o método que tenho que invocar para obter a informação.

- Passo 4: O *Web Service* responde qual o método para obter a informação.

- Passo 5: Sabendo qual o *Web Service* e qual o método de invocação a informação é solicitada.

- Passo 6: O *Web Service* responderá a solicitação da informação e até mesmo uma mensagem de erro será enviada, caso não encontre a informação.

3 SEGURANÇA NA *WEB*

A palavra segurança é oriunda do latim, e significa sem preocupação. Sua etimologia inspira de maneira sutil o sentido de ocupar de você próprio. Uma definição utilizada no Dicionário de Filosofia, Moral e Política sobre o que define segurança é:

> Segurança é a ausência do risco, a previsibilidade, a certeza quanto ao futuro. Risco é qualquer fator que diminui a previsibilidade e portanto a certeza sobre o futuro. (MATOS, 2004, p. 8).

Na última década, a literatura faz questão de considerar para efeito de exemplos os nomes, tais como “Alice”, “Bob”, “Eve” e outros em praticamente todos os artigos e livros que falam sobre criptografia. Os criptógrafos seguem uma tradição. (TANENBAUM, 2003, cap. 8, p. 19).

FARRELL (2009) reforça que dentro dos padrões da informática, segurança é entendido como controle de risco, não significa eliminação total de risco, isso raramente acontece, para que se tenha um bom projeto de segurança é necessário ter claro quais os pontos do projeto que devem ter esse tipo de dispositivo para que não haja um custo muito desproporcional dentro desse projeto. Para isso é primordial que se faça um planejamento dos tipos de riscos que os ativos da organização estão sujeitos e assim direcionar recursos para que os riscos sejam mitigados. A origem da segurança da informação é:

> A segurança da informação teve origem no ambiente militar, onde o sigilo é geralmente o requisito mais importante. Isso levou muitas pessoas a concluir erradamente que não precisam de segurança, pois todas as suas informações são públicas. Sempre que as pessoas dependem de informações fornecidas por um serviço, existe um requisito de integridade. Afinal, embora Alice tenha a preocupação de que seu banco divulgue informações sobre sua conta, ela certamente ficaria mais preocupada se um invasor roubasse o dinheiro da conta. (FARRELL, 2009, p. 4).

A segurança da informação está baseada em três tipos distintos de riscos, cada um culminando por um requisito de segurança específico com sua tecnologia. A figura

10 demonstra que, para cada risco há um requisito específico de segurança e a tecnologia correspondente a ser utilizada.

Risco	Requisito de Segurança	Tecnologia
Divulgação	Confidencialidade	Criptografia
Modificação	Integridade	Autenticação, Assinaturas digitais
Serviço	Controle de Acesso	Controle de Acesso

Figura 10: Tipos de Riscos
Fonte: FARRELL, Sthepen et al (2009, p. 4).

COULOURIS, DOLLIMORE & KINDBERG (2007) enfatiza que o principal objetivo da segurança é limitar o acesso aos recursos e informações somente para os principais envolvidos que tenham autorização. As principais ameaças contra a segurança recaem em três amplas categorias. São elas:

- VAZAMENTO: O acesso das informações por usuários que não têm autorização para tal.

- FALSIFICAÇÃO: O acesso das informações e consequentemente, alteração da mesma.

- VANDALISMO: O acesso no operacional do sistema, pelo simples prazer da interferência.

Os ataques em sistemas distribuídos têm êxito através da obtenção de acesso aos canais de comunicação, mascarados como um canal de comunicação autorizado. O termo canal é definido para referir qualquer mecanismo de comunicação entre processos de um sistema. Os métodos de ataque são classificados como intromissão, mascaramento, falsificação de mensagens, reprodução e negação de serviço.

CHEWICK & BELLOVIN (apud COULOURIS, DOLLIMORE & KINDBERG, 2007) identifica que são 42 deficiências que podem ser considerados riscos sérios em sistemas usados na Internet e que variam desde a descoberta de senhas até aos ataques que executam o protocolo de sincronização de envio e recebimento de mensagens eletrônicas. Se na transmissão de alguma mensagem entre dois processos puder ser minuciosamente verificada, essa informação poderá ser analisada. O vazamento de informações aparece quando os resultados de uma transmissão podem ser colhidos.

FARRELL (2009) destaca que dentro da política de segurança na troca de informações através de rede de computadores ou até mesmo rede interna de uma empresa, um tipo de tecnologia vem sendo muito utilizada, trata-se da criptografia de informações, que consiste em possibilitar a confidencialidade das informações, protegidas por encriptação e a integridade do conteúdo, assegurada pela autenticação.

O método mais usual e simples de criptografia é o *message digest* (resumo de mensagem), comumente chamado de *hash function (*função hash*)*, ou *one way function* (função de sentido único). O resumo de mensagem tem valor igual criptográfico de uma operação de verificação. O resumo de mensagem trabalha no documento para comprimi-lo, criando um *digest value* (valor de resumo). As funções de resumo mais utilizadas são MD5 (message digest 5) e SHA-1 (secure hash algorithm), pois geram respectivamente valores de resumo de mensagem de 128 e 160 bits, independente do tamanho do documento a ser criptografado. O modo mais utilizado e simples de criptografia é conhecido como *shared secret encryption* (criptografia de segredo compartilhado), onde a usuária Alice criptografa um documento com uma chave secreta anteriormente informada ao usuário Bob. Quando o usuário Bob recebe o documento, ele o decriptografa utilizando-se da mesma chave secreta.

A criptografia, de acordo com COULOURIS, DOLLIMORE & KINDBERG (2007. p. 244), é:

> processo de codificar uma mensagem de maneira a ocultar seu conteúdo. A criptografia moderna inclui vários algoritmos de segurança para cifrar e decifrar mensagens baseados no uso de segredos chamados chaves. Uma chave de criptografia é um parâmetro usado em um algoritmo de criptografia de tal maneira que a criptografia não possa ser revertida sem o consentimento da chave.

Há duas principais classes de algoritmo de criptografia de uso geral. A primeira classe utiliza-se de chaves secretas compartilhadas, ou seja, o remetente e o destinatário devem ter o conhecimento da chave utilizada e essa chave não deve ser compartilhada com mais ninguém. A segunda classe usa pares de chaves pública e privada, onde o remetente se utiliza de uma chave pública, que já foi publicada pelo destinatário para cifrar a mensagem. O destinatário usa uma chave privada, correspondente para decifrar a mensagem. No entanto, muitos principais podem verificar a chave pública, somente o destinatário decifra a mensagem, pois é o destinatário que tem a chave privada.

A utilização da criptografia tem importância em diversos momentos na implementação de sistemas seguros. São eles:

- Segredo e Integridade:

> A criptografia é usada para manter o segredo e a integridade da informação, quando ela é exposta a ataques em potencial: por exemplo, durante a transmissão em redes vulneráveis à intromissão e à falsificação da mensagem. Esse uso da criptografia corresponde à sua função tradicional em atividades militares e de inteligência. Ele explora o fato de que uma mensagem cifrada com uma chave de criptografia em particular só pode ser decifrada por um destinatário que conheça a chave para decifrar correspondente. Assim, ele mantém o segredo da mensagem cifrada, desde que a chave para decifrar não seja comprometida (exposta a quem não é participante da comunicação) e que o algoritmo de criptografia seja poderoso o suficiente para anular todas as tentativas possíveis de violá-lo. A criptografia também mantém a integridade da informação cifrada, já é possível incluir e verificar algum tipo de informação redundante, como uma soma de verificação. (COULOURIS, DOLLIMORE & KINDBERG. 2007. p. 245).

- Autenticação: A criptografia é utilizada em procedimentos de autenticação da comunicação em pares. Um remetente cifra a mensagem com uma chave particular, contendo um valor de soma correto, ou algum outro valor. O destinatário em conseguindo decifrar a mensagem, conclui que é autentica a mensagem enviada pelo remetente.

- Assinaturas Digitais: A criptografia realiza um mecanismo que simula as assinaturas convencionais, verificando se um documento ou uma mensagem é uma cópia sem alteração do que foi enviado pelo remetente.

3.1 ALGORITMOS DE CRIPTOGRAFIA

SILVA FILHO (2009) afirma que para uma mensagem onde o remetente quer ter a certeza que somente o destinatário possa ler, é necessário que se tome algumas medidas para que o sigilo seja mantido. Uma das soluções é a criptografia. Na criptografia, a terminologia usada para mensagem original é chamada de texto claro, ou simplesmente mensagem. O processo de mudança para esconder o conteúdo da mensagem e denominado de cifração, promovendo transformações matemáticas adequadas sobre a mensagem. A mensagem cifrada é também chamada de texto cifrado ou criptograma. O processo inverso para recuperação da mensagem é denominado de decifração. Os processos para cifração e decifração usam um algoritmo e um parâmetro de controle chamado de chave criptográfica. A princípio, para que possa haver a decifração, somente quem tiver o conhecimento da chave utilizada e o algoritmo pode decifrar a mensagem. O processo para decifração de mensagens onde são conhecidos os algoritmos, mas sem conhecimento da chave, é chamado de Criptoanálise. A criptografia trata também de soluções de outras aplicações que utilizam autenticação, assinatura digital, dinheiro eletrônico e outros.

A classe de algoritmos de criptografia está dividida em duas categorias. Os simétricos, chamados também de chave-secreta, e os assimétricos, chamados de chave-pública. Os algoritmos simétricos usam idêntica chave para cifrar e decifrar. Para os algoritmos assimétricos há chaves diferenciadas, uma para cifrar e a outra chave para decifrar. A chave de decifração não pode ser obtida através do conhecimento da chave de cifração. Nos algoritmos assimétricos, as chaves são sempre criadas aos pares, ou seja, uma para cifrar e outra para decifrar.

A chave gerada pelo algoritmo simétrico exige que a chave seja mantida em sigilo pelos dois envolvidos no envio e recebimento da mensagem. Com isso, o manuseio

desta chave gera algumas dificuldades no sentido de manter um canal seguro que permita a transmissão desta chave. A figura 11 demonstra a forma que o algoritmo simétrico processa a informação numa mensagem enviada para Alice.

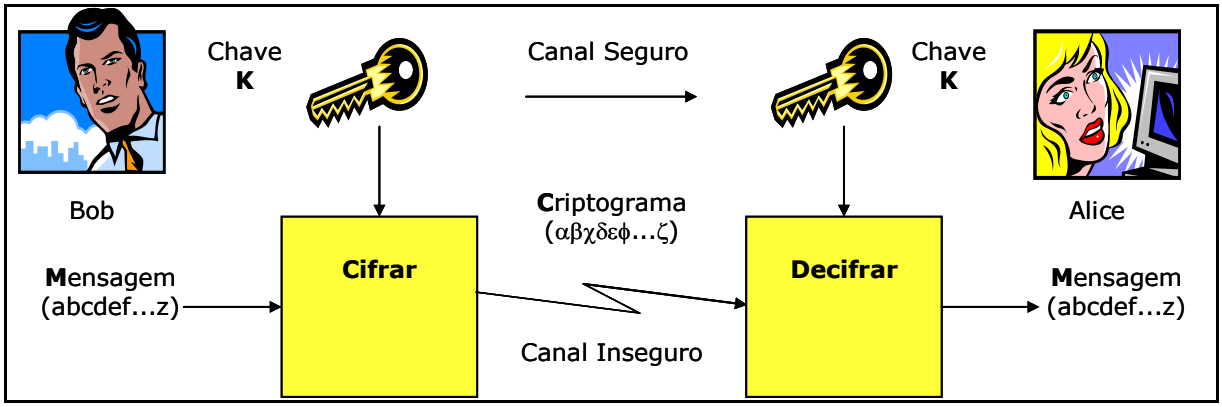


Figura 11: Uso de algoritmo simétrico (chave secreta)
Fonte: SILVA FILHO, Joel Guilherme da. (2009, p. 2).

Segue abaixo alguns algoritmos simétricos. São eles:

Algoritmo Simétrico	Bits	Descrição
DES	56	O Data Encryption Standard (DES) é o algoritmo simétrico mais disseminado no mundo. Foi criado pela IBM em 1977 e, apesar de permitir cerca de 72 quadrilhões de combinações (2^{56}), seu tamanho de chave (56 bits) é considerado pequeno, tendo sido quebrado por "força bruta" em 1997 em um desafio lançado na Internet. O NIST (National Institute of Standards and Technology), que lançou o desafio mencionado, recertificou o DES pela última vez em 1993 e desde então está recomendando o 3DES. O NIST está também propondo um substituto ao DES que deve aceitar chaves de 128, 192 e 256 bits, operar com blocos de 128 bits, ser eficiente, flexível e estar livre de "royalties". O novo padrão, denominado AES (Advanced Encryption Standard), está sendo estudado desde 1997 a partir de vários algoritmos apresentados pela comunidade. Os finalistas são: Serpent, Mars, RC6, Twofish e Rinjdael, e o resultado deverá ser divulgado no final de 2000.
Triple DES	112 ou 168	O 3DES é uma simples variação do DES, utilizando-o em três ciframentos sucessivos, podendo empregar um versão com duas ou com três chaves diferentes. É seguro, porém muito lento para ser um algoritmo padrão.
IDEA	128	O International Data Encryption Algorithm foi criado em 1991 por James Massey e Xuejia Lai e possui patente da suíça ASCOM Systec. O algoritmo é estruturado seguindo as mesmas linhas gerais do DES. Mas na maioria dos microprocessadores, uma implementação por *software* do IDEA é mais rápida do que uma implementação por *software* do DES. O IDEA é utilizado principalmente no mercado financeiro e no PGP, o programa para criptografia de e-mail pessoal mais disseminado no mundo.
Blowfish	32 a 448	Algoritmo desenvolvido por Bruce Schneier, que oferece a escolhe entre maior segurança ou desempenho através de chaves de tamanho variável. O autor aperfeiçoou-o no Twofish, concorrente ao AES.
RC2	8 a 1024	Projetado por Ron Rivest (o R da empresa RSA Data Security Inc.) e utilizado no protocolo S/MIME, voltado para criptografia de e-mail corporativo. Também possui chave de tamanho variável. Rivest também é o autor do RC4, RC5 e RC6, este último concorrente ao AES.

Figura 12: Algoritmos simétricos
Fonte: MAIA, Luiz Paulo & PAGLIUSI, Paulo Sergio (2009, p. 3).

Os algoritmos assimétricos possibilitam que a chave de cifração torne-se pública, sendo a chave, disponível em um repositório de acesso público, e por isso que é chamada de chave pública. Com a chave pública qualquer pessoa pode cifrar uma determinada mensagem, todavia, somente o destinatário que possui a chave de decifração correspondente é que pode decifrar a mensagem. A chave de decifração pode ser denominada de chave privada ou secreta. A chave privada deve ser mantida em segurança e em segredo pelo responsável pela geração do par de chaves, por conseguinte, a chave pública pode ser de conhecimento de todos. A figura 13 mostra a operação do algoritmo assimétrico, onde Alice cria o par de chaves, e envia a chave pública à Bob. Bob então cifra a mensagem com a chave pública de Alice, na qual, somente ela será capaz de decifrar a mensagem, fazendo uso da sua chave privada.

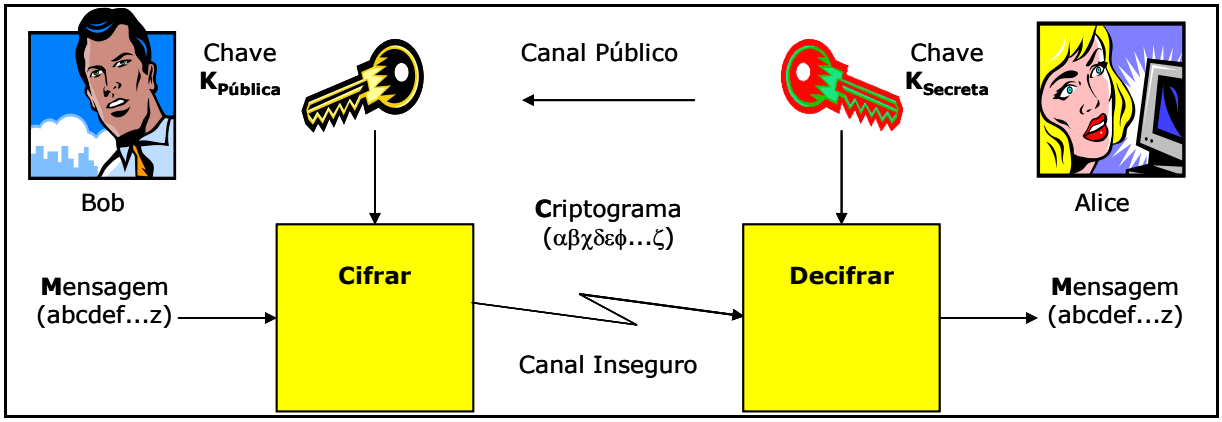


Figura 13: Uso de algoritmo assimétrico (chave pública)
Fonte: SILVA FILHO, Joel Guilherme da. (2009, p. 2).

Os algoritmos simétricos normalmente são bem rápidos na sua execução, com altas taxas de cifração, chegando a ordem de 10 gigabits por segundo (10^9 bits/s). Os algoritmos assimétricos são menos eficientes computacionalmente, mas geralmente a tendência é a utilização em conjunto dos dois algoritmos, ou seja, um algoritmo de chave pública é usado para cifrar uma chave criptográfica utilizada na cifração de uma mensagem via um algoritmo simétrico. Primeiro o destinatário decifra a chave simétrica, através de sua chave privada via o sistema de chave pública, e logo após decifra a mensagem fazendo uso da chave recuperada no sistema simétrico. Nesse processo de utilização, não se tem o problema de compartilhar o segredo da chave com outra pessoa. Para cada mensagem nova, pode-se fazer todo o processo novamente. No caso de Bob querer enviar uma mensagem para Alice, primeiro

deve-se escolher uma chave K, e a envia através do algoritmo assimétrico (chave pública) cifrada com a chave pública de Alice. Alice recupera a chave, decifrando o criptograma com sua chave privada. Nesse caso, Bob envia a mensagem através do algoritmo simétrico, que é mais eficiente para esse feito, cifrando a mensagem com a chave que Alice já possui, e que foi enviada de forma segura. A figura 14 ilustra todo esse o procedimento de uso do algoritmo simétrico com o assimétrico.

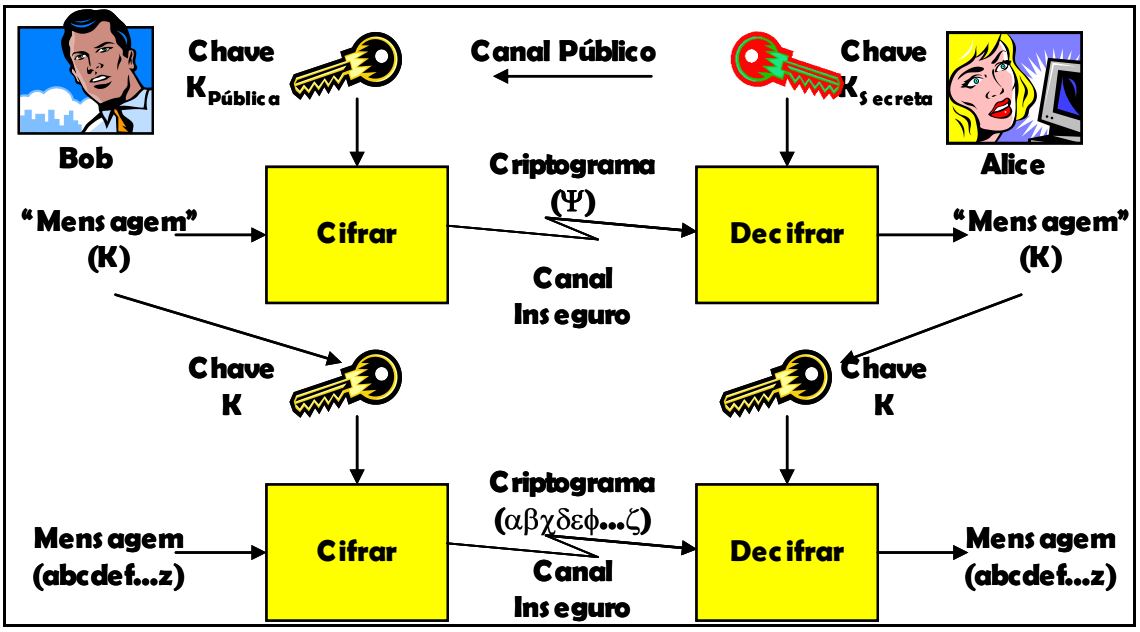


Figura 14: Uso misto algoritmo simétrico e assimétrico
Fonte: SILVA FILHO, Joel Guilherme da. (2009, p. 3).

Segue abaixo alguns algoritmos assimétricos. São eles:

Nome	Descrição
RSA	Inventado por Ron Rivest, Adi Shamir e Len Adleman, no MIT. O algoritmo é o mais utilizado, e o mais poderoso, pois, realiza transformações matemática com números primos. Sua filosofia está em multiplicar dois números primos para alcançar um terceiro número. Este terceiro número fica muito difícil de recuperar a partir da fatoração. A dificuldade em fatorar números grandes é a base desse algoritmo, pois, demanda muito tempo para ser recuperado. Um exemplo foi uma experiência, onde uma chave de 512 bits necessitou sete meses para ser descoberta, onde atuaram cientistas de seis paises com mais de 300 estações de trabalho.
ElGamal	Utiliza-se de manuseio de grandes quantidades numéricas para gerenciamento de suas chaves. A segurança está no problema do logaritmo discreto. A complexidade para calcular logaritmos discretos em um corpo finito é o cerne da sua criação.
Diffie-Hellman	Basea-se também no problema do logaritmo discreto, porem não permite ciframento e assinatura digital. O código foi desenhado para permitir dois indivíduos compartilharem informações assim como uma chave.
Curvas Elípticas	Criado por, Neal Koblitz e V.S. Miller, desenvolveram a utilização de curvas elípticas. Implementaram algoritmos de chave pública já existentes, tal como o algoritmo Diffie-Hellman. O sistema criptográfico de curvas elípticas tem sua base na modificação de outros sistemas (ElGamal). São seguros com chaves de menor tamanho.

Figura 15: Algoritmos assimétricos
Fonte: MAIA, Luiz Paulo & PAGLIUSI, Paulo Sergio (2009, p. 4).

3.2 ASSINATURAS DIGITAIS

MAIA & PAGLIUSI (2009) destaca que um dos grandes benefícios da criptografia com chave pública é a assinatura digital, pois ela garante a autenticidade, associada à integridade do conteúdo de quem está enviando uma mensagem. Num dos exemplos clássicos é a suposição de que a remetente Alice, deseja enviar a comunicação de que seu filho nasceu para todos os destinatários, no caso Bob. Alice quer a garantia de que a mensagem chegue integra e sem alterações e que Bob tenha certeza que foi ela que enviou.

> Alice então cifra a mensagem com sua chave privada e a envia, em um processo denominado de assinatura digital. Cada um que receber a mensagem deverá decifrá-la, ou seja, verificar a validade da assinatura digital, utilizando para isso a chave pública de Alice. Como a chave pública de Alice apenas decifra (ou seja, verifica a validade de) mensagens cifradas com sua chave privada, fica garantida assim a autenticidade, integridade e não-repudiação da mensagem. Pois se alguém modificar um bit do conteúdo da mensagem ou se outra pessoa assiná-la ao invés de Alice, o sistema de verificação não irá reconhecer a assinatura digital de Alice como sendo válida. (MAIA & PAGLIUSI, 2009, p. 5).

Quanto ao exemplo acima, é importante ressaltar que a assinatura digital não garante a confidencialidade. Um intruso de nome Eve poderá acessá-la simplesmente usando a chave pública de Alice. Para obter a confidencialidade deve-se combinar o procedimento tal que, Alice assina a mensagem utilizando a chave privada, criptografa a mensagem outra vez com sua assinatura, agora utilizando a chave pública de Bob. Quando Bob receber a mensagem primeiramente ele deve decifrá-la com sua chave privada garantindo a privacidade, depois, deve decifrá-la novamente checando assim a assinatura utilizando a chave pública da Alice, tendo a garantia da sua autenticidade. Os algoritmos usados na assinatura digital são os RSA e ElGamal, estes já descritos acima, e DSA. O DSA (*Digital Signature Algorithmn*) é estritamente utilizado para as assinaturas digitais, foi proposto pelo NIST, inventado pelo NSA e patenteado pelo governo Americano.

A figura a seguir ilustra o procedimento da assinatura digital utilizando o RSA:

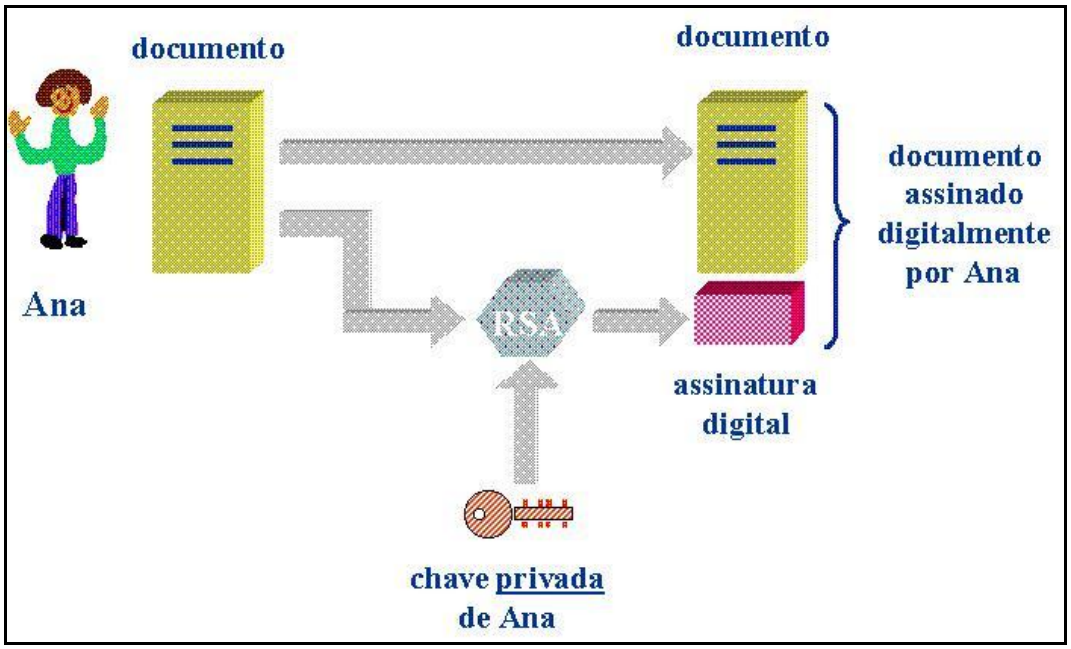


Figura 16: Assinatura digital
Fonte: MAIA, Luiz Paulo & PAGLIUSI, Paulo Sergio (2009, p. 6).

3.3 FUNÇÃO *HASHING*

De acordo com SILVA FILHO (2009) uma função de *hashing* é uma função criptográfica que cria uma saída com tamanho fixo, normalmente de 128 a 256 bits, independente do tamanho do documento ou mensagem de entrada.

Para MAIA & PAGLIUSI (2009) a assinatura digital obtida pelo uso de chave pública ou criptografia assimétrica não pode ser utilizada isoladamente. Deve-se utilizar a função *hashing*, pois, os algoritmos assimétricos são demasiadamente lentos para serem gerados. São normalmente cerca de 1.000 vezes mais lentos que os algoritmos simétricos. A função *hashing* produz um arquivo de valor pequeno, de tamanho fixo, a partir de mensagens ou arquivos de qualquer tamanho que se deseja assinar. A função *hashing* também é chamada de *Message Digest*, *One-Way Hash Function*, Função de Condensação, ou até mesmo de Função de Espalhamento Unidirecional.

O arquivo gerado é o *digest* ou valor *hash*, que é útil para garantir a integridade do conteúdo da mensagem ou arquivo. Sendo assim, após o valor *hash* ter sido calculado, através da função *hashing*, qualquer tipo de modificação no seu conteúdo, será descoberto, pois, se um novo valor *hash* for calculado após qualquer

que seja a alteração, esse valor terá com certeza valores totalmente diferentes do valor *hash* original. A função *hashing* dispõe de rapidez nas assinaturas digitais bem como garante a integridade confiável do documento ou mensagem. Destacam-se duas funções *hashing* mais utilizadas. São elas:

- MD5: A função MD5 (*Message Digest)* é uma função de espalhamento unidirecional inventada por Ron Rivest e que gera um valor *hash* de 128 bits para mensagens de entrada de tamanho arbitrário. Projetado para ser rápido, simples e seguro. Seus detalhes são públicos, o que levou análise pela comunidade de criptografia. Uma fraqueza foi descoberta, porém, não houve consequências para a segurança do algoritmo.

- SHA-1: A função SHA-1 (*Secure Hash Algorithm)* é uma função de espalhamento unidirecional inventada pela NSA, que gera um valor *hash* de 160 bits para mensagens de tamanho arbitrário. A fraqueza do MD5 não acontece no SHA-1. Hoje em dia, não há notícia de nenhum ataque de criptoanálise contra a função SHA-1. Um ataque de força bruta é impraticável, devido ao seu valor *hash* de 160 bits.

A figura 17 ilustra a utilização de funções *hashing*:

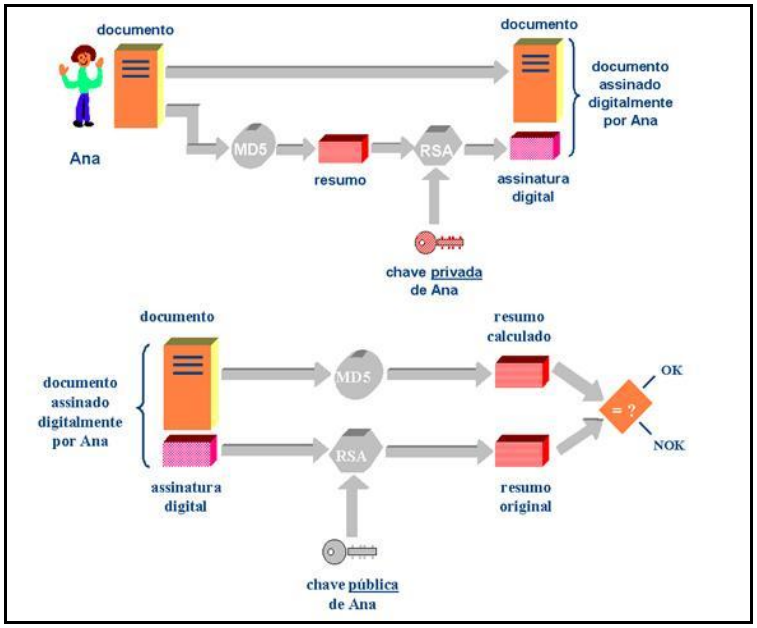


Figura 17: Funções *Hashing*
Fonte: MAIA, Luiz Paulo & PAGLIUSI, Paulo Sergio (2009, p. 7).

3.4 PROTOCOLOS CRIPTOGRÁFICOS

Para implementação dos três mecanismos criptográficos básicos, o ciframento, assinatura digital e a função *hashing*, os algoritmos criptográficos podem ter a combinação com protocolos criptográficos, encaixados dentro da arquitetura de segurança de produtos com perfil de comércio eletrônico. Tais protocolos criptográficos dão sustentação ao comércio eletrônico, no sentido amplo da segurança na *web*, ou seja, garantindo assim a disponibilidade, o sigilo, o controle de acesso, a autenticidade, a integridade e o não-repúdio. Alguns exemplos dão a seguir uma solução aos sistemas que utilizam sistemas criptográficos chamados de híbridos. São eles:

Protocolo	Descrição
IPSec	Padrão de protocolos criptográficos desenvolvidos para o IPv6. Realiza também o tunelamento de IP sobre IP. É composto de três mecanismos criptográficos: Authentication Header (define a funçãoHashing para assinatura digital), Encapsulation Security Payload (define o algoritmo simétrico para ciframento) e ISAKMP (define o algoritmo assimétrico para Gerência e troca de chaves de criptografia). Criptografia e tunelamento são independentes. Permite Virtual Private Network fim-a-fim. Futuro padrão para todas as formas de VPN.
SSL e TLS	Oferecem suporte de segurança criptográfica para os protocolos NTTP, HTTP, SMTP e Telnet. Permitem utilizar diferentes algoritmos simétricos, message digest (hashing) e métodos de autenticação e gerência de chaves (assimétricos).
PGP	Inventado por Phil Zimmermman em 1991, é um programa criptográfico famoso e bastante difundido na Internet, destinado a criptografia de e-mail pessoal. Algoritmos suportados: hashing: MD5, SHA-1, simétricos: CAST-128, IDEA e 3DES, assimétricos: RSA, Diffie-Hellman/DSS. Versão mais recente: 6.5.3.
S/MIME	O S/MIME (Secure Multipurpose Internet Mail Extensions) consiste em um esforço de um consórcio de empresas, liderado pela RSADSI e pela Microsoft, para adicionar segurança a mensagens eletrônicas no formato MIME. Apesar do S/MIME e PGP serem ambos padrões Internet, o S/MIME deverá se estabelecer no mercado corporativo, enquanto o PGP no mundo do mail pessoal.
SET	O SET é um conjunto de padrões e protocolos, para realizar transações financeira seguras, como as realizadas com cartão de crédito na Internet. Oferece um canal de comunicação seguro entre todos os envolvidos na transação. Garante autenticidade X.509v3 e privacidade entre as partes.
X.509	Recomendação ITU-T, a especificação X.509 define o relacionamento entre as autoridades de certificação. Faz parte das séries X.500 de recomendações para uma estrutura de diretório global, baseada em nomes distintos para localização. Utilizado pelo S/MIME, IPSec, SSL/TLS e SET. Baseado em criptografia com chave pública (RSA) e assinatura digital (com hashing).

Figura 18: Protocolos criptográficos híbridos
Fonte: MAIA, Luiz Paulo & PAGLIUSI, Paulo Sergio (2009, p. 8).

3.5 CERTIFICADO DIGITAL

O gerenciamento de chaves num sistema de chave pública, tem duas novas características. A primeira deve-se localizar antecipadamente a chave pública da pessoa com que se deseja trocar informações. A segunda característica é que deve-se ter garantia de que a chave pública localizada seja da pessoa na qual será trocada as informações. Caso essa garantia não exista, um intruso, no caso Eve, pode facilmente convencer o remetente e o destinatário de que chaves públicas falsas pertencem a ambos e sendo assim, estabelecendo um elo de confiança entre os dois, Eve pode se passar por um deles ou ambos. Nesse sentido, quando o remetente Alice, enviar uma mensagem a Bob, o intruso Eve, pode atuar entre os interlocutores, enviando uma chave pública criada por ele e assim podendo decifrar, cifrar novamente a mensagem com outro conteúdo, causando um enorme problema. Por esse ataque, o intruso Eve, pode causar muitos prejuízos, poderia causar até mais danos, caso conseguisse quebrar o algoritmo utilizado pelos interlocutores.

Os certificados de chave pública são a garantia para que esses ataques não obtenham êxito. O certificado resume-se em chaves públicas assinadas por alguém de confiança. Geralmente está no padrão ITU X.509v3 (*International Telecommunications Union*). O certificado digital tem como objetivo evitar substituição de uma chave pública por outra qualquer. No caso, o certificado de Bob, possui alem da chave pública, alguns outros atributos, tais como, nome, endereço e outras informações pessoais e é assinado por uma Autoridade de Certificação ou mesmo CA (*Certification Authority*), que funciona similarmente como um cartório eletrônico. Por definição, um certificado digital é um documento eletrônico, assinado digitalmente por uma outra parte de cunho confiável, onde é associada além da chave pública, alguns outros atributos que somente quem expediu o certificado tem como provar que o mesmo é autêntico.

> Pela assinatura da chave pública e das informações sobre Bob, a CA garante que a informação sobre Bob está correta e que a chave pública em questão realmente pertence a Bob. Alice, por sua vez, confere a assinatura da CA e então utiliza a chave pública em pauta, segura de que esta pertence a Bob e a ninguém mais. Certificados desempenham um importante papel em um grande número de protocolos e padrões utilizados na proteção de sistemas de comércio eletrônico. (MAIA & PAGLIUSI, 2009, p. 9).

Grandes empresas como a Verisign, Cybertrust e Nortel são autoridades de certificação (CA) e assinam os certificados digitais assegurando a autentica validade. As CA têm por obrigação e responsabilidade, manter e divulgar a lista com os certificados revogados (*Certificate Revocation List*-CRL).

Os certificados que constam nessa lista, podem ter tido algum tipo de problema, pois podem ser roubados, perdidos e até mesmo estar sem uso. Geralmente as CAs utilizam-se de uma hierarquia de certificação, onde a validade de uma CA de nível inferior valida sua assinatura com a assinatura de uma CA de hierarquia superior.

A manutenção do gerenciamento de chaves públicas utiliza-se da Public Key Infrastructures (PKI) ou Infra-estrutura de chaves públicas. A PKI define onde serão armazenados e recuperados os certificados digitais. De qual forma estarão armazenados esses certificados e também como um certificado tem sua revogação.

Existem vários tipos de certificados, destacam-se os Certificados de CA, que são usados para validação de outros certificados, os Certificados de Servidor, que são usados para identificação de servidores seguros, os Certificados Pessoais, que como o próprio nome diz, contém informações do portador e os Certificados de Desenvolvedores de Software, que são certificados para validação de assinaturas ligadas a programas de software.

De acordo com BRASIL-ITI (2009) o controle e gerenciamento no Brasil da infra-estrutura de chaves públicas é feito pela ICP-Brasil (Infra-Estrutura de Chaves Públicas).

A seguir a estrutura de nível 1 e nível 2 com suas autoridades certificadoras:

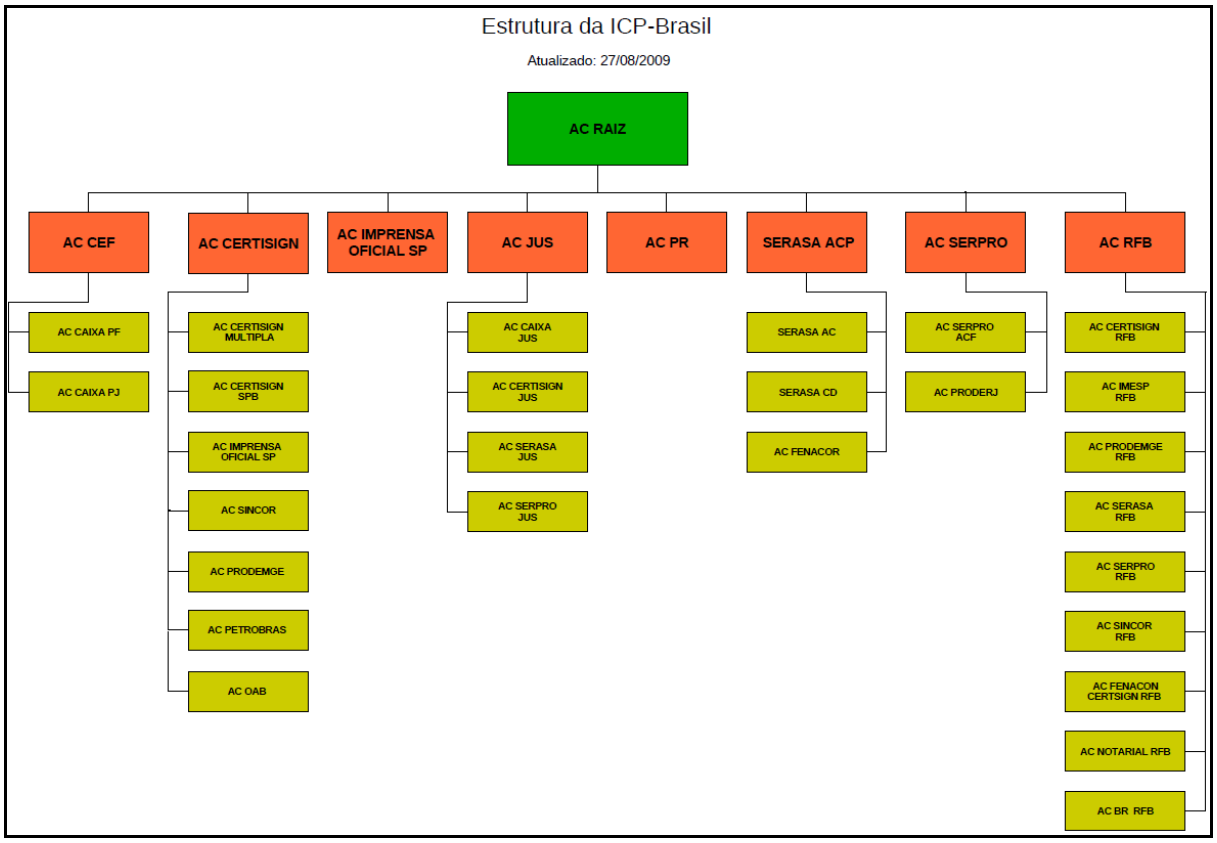


Figura 19: Estrutura da ICP-Brasil
Fonte: BRASIL-ITI (2009).

Dentre as autoridades certificadoras citadas na figura acima, destaca-se a SERASA, por se tratar de um AC que fornece segurança com seus certificados digitais, pois trabalha atualmente com a maioria das instituições financeiras que participa do Sistema de Pagamentos Brasileiros (SPB).

3.6 FIREWALLS

A definição de *firewall* é:

> dispositivos de segurança que protegem os recursos de hardware e software da empresa dos perigos (ameaças) aos quais o sistema está exposto. Em redes de computadores, *firewalls* são barreiras interpostas entre a rede privada e a rede externa com a finalidade de evitar intrusos (ataques). Estes mecanismos de segurança são baseados em hardware e software e seguem a política de segurança estabelecida pela empresa (SOUSA JR & PUTTINI, 2009).

Segundo COULOURIS, DOLLIMORE & KINDBERG (2007) o principal objetivo do *Firewalls* é controlar e examinar toda e qualquer comunicação para dentro e para

fora de uma intranet. Um *firewall* é implementado aplicando-se a política de segurança determinada pela organização. Os *firewalls* interceptam toda a comunicação externa, produzindo assim, um ambiente de comunicação segura, onde as mensagens autorizadas são endereçadas para o destinatário.

PINHEIRO (2004) considera que a segurança em um projeto de rede de computadores é uma preocupação constante. Para evitar problemas de acessos indesejáveis os projetistas definem que os serviços de uma rede devem ser distribuídos em vários outros servidores para que esses serviços não sejam todos expostos à internet, pois, se a maioria dos serviços forem disponibilizados num único servidor, esses serviços ficam vulneráveis a ataques pela Internet.

A solução encontrada para disponibilizar serviços da rede interna e rede externa para que sejam acessados seguramente e que não reflitam em maiores problemas, criam-se as DMZ (*DeMilitarized Zone*), ou seja, as zonas desmilitarizadas. Um exemplo para ilustrar é a separação do servidor de e-mail do servidor da *web*. O servidor de e-mail é o mais requisitado por ataques externos, porém, o mais importante é o servidor *web*, por sustentar os sistemas nele baseados.

Em uma rede interna, algumas máquinas precisam acessar a rede externa, é o caso, por exemplo, dos serviços baseados em servidores SMTP (*Simple Mail Transfer Protocol*) e servidores *Web*. Nesse caso há necessidade de manter os computadores que não tem acesso à rede externa seguros. Para tanto, a DMZ é criada, colocando os servidores nessa área para oferecer acesso à rede externa, com a segurança também para os servidores da área interna da organização.

Um *firewall* implementado na DMZ permite um único ponto de acesso à rede, e esse tráfego de rede pode ser monitorado, analisado e controlado pelo software de firewall, visando auditar quem tem acesso a quais serviços e por qual usuário ou endereço do mesmo.

A seguir um exemplo de uma rede com *firewall*:

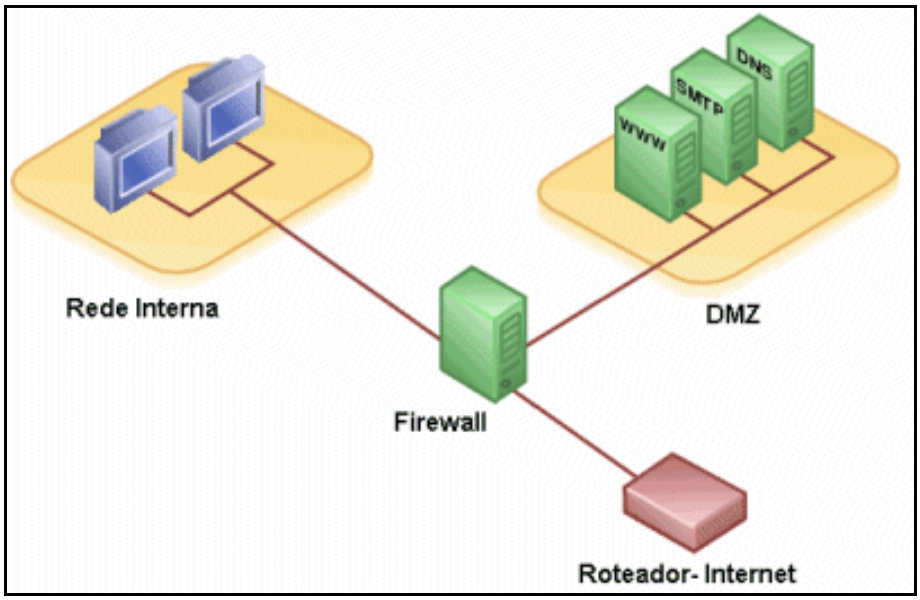


Figura 20: Rede com firewall e DMZ
Fonte: PROJETO-Firewall (2009, p.2).

4 CONHECIMENTO DE TRANSPORTE ELETRÔNICO

De acordo com BRASIL-SC (2009) a União, os Estados, o Distrito Federal e os municípios têm autonomia política, administrativa e financeira e, por isso, cada uma dessas instâncias tem como estabelecer regras para administrar seus recursos e tributos.

O processo de globalização das informações faz com que as administrações tributárias se adaptem ao desafio de controlar as movimentações fiscais sempre crescentes, no que tange à necessidade do Estado de prevenir e detectar a evasão tributária.

Dentro desse contexto, a racionalização e modernização da integração das informações pela administração tributária brasileira, vê na criação do projeto de documento fiscal eletrônico, a solução para reduzir custos, facilitar o cumprimento das obrigações tributárias e pagamento correto dos impostos, bem como o controle e fiscalização de informações entre as administrações tributárias e contribuintes.

As administrações tributárias brasileiras visam com a implantação do projeto CT-e, de forma gradativa, a substituição da atual sistemática de emissão de documentos fiscais em papel pelo documento fiscal eletrônico.

O documento eletrônico tem validade jurídica, garantida por leis nacionais e estaduais e também garantida por assinaturas digitais, simplificando as obrigações acessórias dos contribuintes, permitindo o acompanhamento das prestações de serviço de transportes, em tempo real pelo Fisco e contribuinte.

A implantação do CT-e, assim como a NF-e (nota fiscal eletrônica) é um marco histórico na vida dos contribuintes, facilitando os envolvidos nos processos, atividades como fiscalização e acompanhamento de prestação de serviços.

Conforme BRASIL-PortalCTe (2009) o Conhecimento de Transporte Eletrônico (CT-e) e o Documento Auxiliar do Conhecimento de Transporte Eletrônico (DACTE) foi instituído pelo Ajuste SINIEF 09/2007 (Sistema Nacional Integrado de Informações Econômico-Fiscais) e também pelo ato COTEPE 08/2008 (Comissão Técnica Permanente). O ato COTEPE 08/2008 disponibiliza o manual de integração do contribuinte e dispõe sobre as especificações técnicas para a comunicação entre os envolvidos no processo de emissão do CT-e.

A legislação vigente do CT-e e DACTE é nacional. No entanto, para cada estado da federação que vier a implantar o CT-e, deve-se instituir seu próprio decreto estadual.

O CT-e, cujo modelo é o 57, substitui os documentos a seguir. São eles:

- Conhecimento de Transporte Rodoviário de Cargas, modelo 8;
- Conhecimento de Transporte Aquaviário de Cargas, modelo 9;
- Conhecimento Aéreo, modelo 10;
- Conhecimento de Transporte Ferroviário de Cargas, modelo 11;
- Nota Fiscal de Serviço de Transporte Ferroviário de Cargas, modelo 27;
- Nota Fiscal de Serviço de Transporte, modelo 7, quando utilizada em transporte de cargas.

De acordo com BRASIL-MS (2009) o CT-e é um documento estritamente digital criado para documentar uma prestação de serviço de transportes de carga, independente do modal transportado (Rodoviário, Aéreo, Ferroviário, Aquaviário ou Dutoviário).

O DACTE é uma representação gráfica que contém todas as informações gravadas pelo CT-e, ou seja, é um espelho fiel das informações. O DACTE é um instrumento

auxiliar para fazer consultas no site da SEFAZ (Secretaria da Fazenda) do estado autorizador da emissão do CT-e. Nele existe a chave de acesso com 44 caracteres e representação gráfica em código de barras linear padrão CODE-128C.

A figura 21 representa uma consulta resumida de um CT-e. A divulgação está autorizada pela empresa.

CONSULTA RESUMIDA CT-e

Chave de acesso	Número	Série	Versão XML
4309.0395.5917.2300.3053.5700.0000.0000.0100.0000.0000	1	0	1.02

Dados do CT-e

Número	Série	Data Emissão
1	0	02/03/2009 - 07:15:00

VALORES

Valor Total Serviço	Base Cálculo ICMS	Valor ICMS
59,69	0,00	0,00

EMITENTE

CNPJ	Nome / Razão Social	Inscrição Estadual	UF
95.591.723/0030-53	EXPRESSO MERCURIO S.A.	1090137912	RS

TOMADOR DO SERVIÇO

CNPJ	Nome / Razão Social	Inscrição Estadual	UF
94.477.882/0001-24	RODOAUTO COM PNEUS LTDA	1090173145	RS

REMETENTE

CNPJ	Nome / Razão Social	Inscrição Estadual	UF
94.477.882/0001-24	RODOAUTO COM PNEUS LTDA	1090173145	RS

DESTINATÁRIO

CNPJ	Nome / Razão Social	Inscrição Estadual	UF
94.160.991/0001-13	DINECAR AUTO E MOTO PECAS LTDA	0390062065	RS

CARACTERÍSTICAS

Modal	Tipo Serviço	Finalidade	Forma
01 - Rodoviário	0 - Normal	0 - CT-e Normal	1 - Normal
CFOP	**Natureza da Prestação**	**Digest Value do CT-e**	
5353	SERVICO DE TRANSPORTE	7yHH+jyreFQC9LQFC5lAkPha4qw=	
Início da Prestação		**Fim da Prestação**	
RS - 4316907 - SANTA MARIA		RS - 4307005 - ERECHIM	

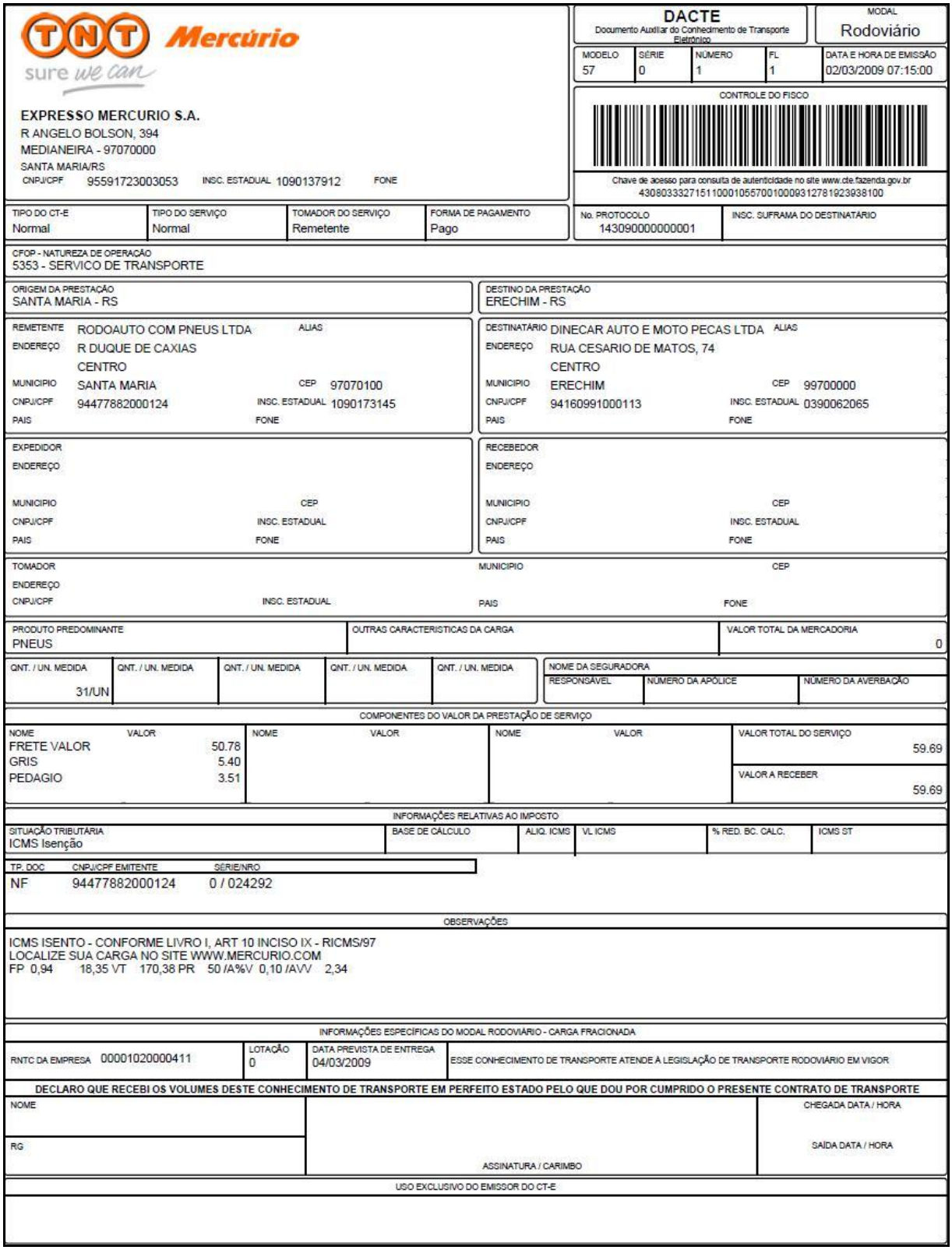
SITUAÇÃO ATUAL : AUTORIZADO O USO DO CT-E

Administração Tributária : Receita Estadual RS

Data/Hora da Consulta : 03/10/2009 16:12:19

Figura 21: CT-e resumido
Fonte: BRASIL-RS (2009).

A seguir, um DACTE é demonstrado pela figura 22 e apresenta a consulta de informações do CT-e. A divulgação está autorizada pela empresa.



TNT Mercúrio
sure we can

EXPRESSO MERCURIO S.A.
R ANGELO BOLSON, 394
MEDIANEIRA - 97070000
SANTA MARIA/RS
CNPJ/CPF 95591723003053 INSC. ESTADUAL 1090137912 FONE

DACTE
Documento Auxiliar do Conhecimento de Transporte Eletrônico

MODAL: Rodoviário

MODELO	SÉRIE	NÚMERO	FL	DATA E HORA DE EMISSÃO
57	0	1	1	02/03/2009 07:15:00

CONTROLE DO FISCO

Chave de acesso para consulta de autenticidade no site www.cte.fazenda.gov.br
43080333271511000105570010009312781923938100

TIPO DO CT-E	TIPO DO SERVIÇO	TOMADOR DO SERVIÇO	FORMA DE PAGAMENTO	No. PROTOCOLO	INSC. SUFRAMA DO DESTINATÁRIO
Normal	Normal	Remetente	Pago	143090000000001	

CFOP - NATUREZA DE OPERAÇÃO
5353 - SERVICO DE TRANSPORTE

ORIGEM DA PRESTAÇÃO
SANTA MARIA - RS

DESTINO DA PRESTAÇÃO
ERECHIM - RS

REMETENTE RODOAUTO COM PNEUS LTDA ALIAS
ENDEREÇO R DUQUE DE CAXIAS
CENTRO
MUNICIPIO SANTA MARIA CEP 97070100
CNPJ/CPF 94477882000124 INSC. ESTADUAL 1090173145
PAIS FONE

DESTINATÁRIO DINECAR AUTO E MOTO PECAS LTDA ALIAS
ENDEREÇO RUA CESARIO DE MATOS, 74
CENTRO
MUNICIPIO ERECHIM CEP 99700000
CNPJ/CPF 94160991000113 INSC. ESTADUAL 0390062065
PAIS FONE

EXPEDIDOR
ENDEREÇO
MUNICIPIO CEP
CNPJ/CPF INSC. ESTADUAL
PAIS FONE

RECEBEDOR
ENDEREÇO
MUNICIPIO CEP
CNPJ/CPF INSC. ESTADUAL
PAIS FONE

TOMADOR
ENDEREÇO
CNPJ/CPF INSC. ESTADUAL
MUNICIPIO CEP
PAIS FONE

PRODUTO PREDOMINANTE	OUTRAS CARACTERISTICAS DA CARGA	VALOR TOTAL DA MERCADORIA
PNEUS		0

QNT. / UN. MEDIDA	QNT. / UN. MEDIDA	QNT. / UN. MEDIDA	QNT. / UN. MEDIDA	QNT. / UN. MEDIDA
31/UN				

NOME DA SEGURADORA

RESPONSÁVEL	NÚMERO DA APÓLICE	NÚMERO DA AVERBAÇÃO

COMPONENTES DO VALOR DA PRESTAÇÃO DE SERVIÇO

NOME	VALOR
FRETE VALOR	50.78
GRIS	5.40
PEDAGIO	3.51

NOME	VALOR

NOME	VALOR

VALOR TOTAL DO SERVIÇO: 59.69
VALOR A RECEBER: 59.69

INFORMAÇÕES RELATIVAS AO IMPOSTO

SITUAÇÃO TRIBUTÁRIA	BASE DE CÁLCULO	ALIQ. ICMS	VL ICMS	% RED. BC. CALC.	ICMS ST
ICMS Isenção					

TP. DOC	CNPJ/CPF EMITENTE	SÉRIE/NRO
NF	94477882000124	0 / 024292

OBSERVAÇÕES

ICMS ISENTO - CONFORME LIVRO I, ART 10 INCISO IX - RICMS/97
LOCALIZE SUA CARGA NO SITE WWW.MERCURIO.COM
FP 0,94 18,35 VT 170,38 PR 50 /A%V 0,10 /AVV 2,34

INFORMAÇÕES ESPECÍFICAS DO MODAL RODOVIÁRIO - CARGA FRACIONADA

RNTC DA EMPRESA	LOTAÇÃO	DATA PREVISTA DE ENTREGA	
00001020000411	0	04/03/2009	ESSE CONHECIMENTO DE TRANSPORTE ATENDE À LEGISLAÇÃO DE TRANSPORTE RODOVIÁRIO EM VIGOR

DECLARO QUE RECEBI OS VOLUMES DESTE CONHECIMENTO DE TRANSPORTE EM PERFEITO ESTADO PELO QUE DOU POR CUMPRIDO O PRESENTE CONTRATO DE TRANSPORTE

NOME
RG
ASSINATURA / CARIMBO
CHEGADA DATA / HORA
SAÍDA DATA / HORA

USO EXCLUSIVO DO EMISSOR DO CT-E

Figura 22: DACTE
Fonte: BRASIL-RS (2009).

4.1 OBJETIVOS

O CT-e tem como principal objetivo:

> a implantação de um modelo nacional de documento fiscal eletrônico para a substituição da sistemática atual de emissão dos documentos fiscais em papel que atualmente acobertam os serviços de transporte interestadual e intermunicipal, reduzindo custos, simplificando as obrigações acessórias dos contribuintes e permitindo, ao mesmo tempo, o acompanhamento em tempo real das operações comerciais pelo Fisco. (BRASIL-SP, 2009, p.1).

O modelo nacional do documento fiscal eletrônico segue padronização elaborada pelos órgãos governamentais e também por empresas do setor de transportes que foram convidadas a participar pela importância deste projeto. O documento eletrônico substitui o documento atual que é impresso em cinco vias, reduzindo significativamente o custo para as empresas de transportes.

4.2 ARQUITETURA DE COMUNICAÇÃO

BRASIL (2008) afirma que os serviços disponibilizados pelos portais das Secretarias de Fazenda Estaduais são os serviços de recepção, cancelamento, inutilização, carta de correção, consulta de status do serviço e consulta do CT-e.

O fluxo de comunicação dos serviços se inicia sempre pelo aplicativo do cliente, através de envio de mensagem com solicitação do serviço que o cliente solicitou. A forma de processamento dos serviços podem ser síncronos ou assíncronos, de acordo com a solicitação do serviço.

Os serviços síncronos são processados e concluídos na mesma conexão, com retorno de mensagem e o resultado do processamento do serviço. Para o processamento da solicitação de serviços assíncronos existe um retorno de mensagem com resposta de que o serviço foi solicitado, porém, o retorno do processamento não é executado na mesma conexão.

Sendo assim, para obter o resultado do serviço solicitado anteriormente, deverá ser feito nova solicitação para obter este serviço.

A figura 23 mostra o serviço e a implementação correspondente.

Serviço	Implementação
Recepção de CT-e	Assíncrona
Cancelamento de CT-e	Síncrona
Inutilização de Numeração de CT-e	Síncrona
Consulta da situação atual do CT-e	Síncrona
Carta de Correção de CT-e	Síncrona
Consulta do status do serviço	Síncrona

Figura 23: Tipo de serviço e sua implementação
Fonte: BRASIL (2008, p.18).

Os serviços solicitados na sua totalidade estão especificados em documentos XML. A especificação adotada é recomendada pela W3C para XML 1.0 e disponível em www.w3.org/TR/REC-xml. Existe ainda uma série de padronizações para otimização dos documentos em XML. Essas otimizações são feitas em declarações de *namespace*, prefixo de *namespace*, na montagem do arquivo e na validação de *schema*, entre outras.

A comunicação está baseada totalmente em Web Services. Os participantes são o contribuinte e a Secretaria de Fazenda Estadual. A Secretaria de Fazenda Estadual de circunscrição do contribuinte, por sua vez, disponibiliza portais para acesso às informações.

O meio físico utilizado é a Internet, utilizando ainda o protocolo SSL, versão 3.0 com autenticação recíproca que permite estabelecer um canal seguro de comunicação, dando lugar a utilização de certificados digitais com identificação do servidor e cliente, suprimindo a identificação do usuário pelo nome, código e senha.

O protocolo padrão para troca de mensagens entre os Web Services do portal da Secretaria de Fazenda Estadual e o contribuinte é o SOAP versão 1.2.

A figura a seguir ilustra os Web Services disponíveis pelo portal do CT-e.

Figura 24: *Web Services* disponíveis
Fonte: BRASIL-PortalCTe (2009).

Os certificados digitais utilizados pelo CT-e são do tipo A1 ou A3 para confirmação de identidade na *Web*, e devem ser emitidos por uma Autoridade Certificadora credenciada pela ICP-Brasil. Os certificados digitais são exigidos em duas situações distintas para o projeto CT-e. São eles:

> a) Assinatura de Mensagens: O certificado digital utilizado para essa função deverá conter o CNPJ do estabelecimento emissor do CT-e ou o CNPJ do estabelecimento matriz. Por mensagens, entenda-se: o Pedido de Autorização de Uso (Arquivo CT-e), o Pedido de Cancelamento de CT-e, o Pedido de Inutilização de Numeração de CT-e e demais arquivos XML que necessitem de assinatura. O certificado digital deverá ter o "uso da chave" previsto para a função de assinatura digital, respeitando a Política do Certificado.
>
> b) Transmissão (durante a transmissão das mensagens entre o servidor do contribuinte e o Portal da Secretaria de Fazenda Estadual): O certificado digital utilizado para identificação do aplicativo do contribuinte deverá conter o CNPJ do responsável pela transmissão das mensagens, mas não necessita ser o mesmo CNPJ do estabelecimento emissor do CT-e, devendo ter a extensão Extended Key Usage com permissão de "Autenticação Cliente". (BRASIL, 2008, p.14).

O portal da Secretaria de Fazenda Estadual aceita que as mensagens enviadas sejam assinadas digitalmente pelo certificado digital que contenha o CNPJ da empresa matriz ou da empresa emissora do CT-e previamente credenciada. Para o processo de assinatura digital o contribuinte não necessita do fornecimento da Lista

de Certificados Revogados (LCR), visto que o portal se incumbe de validar essa lista no momento da conferência da assinatura digital.

A validação da assinatura digital segue regras básicas adotadas pelas Secretarias de Fazenda Estaduais. Algumas delas são:

- Extrair chave pública do certificado;
- Verificar prazo de validade do certificado;
- Montar e validar a cadeia de confiança dos certificados;
- Validar o uso da assinatura digital;
- Garantir que o certificado utilizado é de um usuário contribuinte válido.

4.3 MODELO OPERACIONAL DOS *WEB SERVICES* - SEFAZ

De acordo com BRASIL (2008) os *Web Services* das SEFAZ disponibilizam os serviços para que os contribuintes os acessem. Os serviços disponíveis para o CT-e são os serviços de recepção, retorno da recepção, cancelamento, inutilização, consulta de protocolo, consulta de status do serviço e consulta de cadastro.

A figura 25 mostra os serviços e a URL disponibilizados pela SEFAZ para cada estado da federação.

UF	Serviço	url
MT	CTeRecepcao	https://cte.sefaz.mt.gov.br/ctews/services/CteRecepcao
MT	CTeRetRecepcao	https://cte.sefaz.mt.gov.br/ctews/services/CteRetRecepcao
MT	CTeCancelamento	https://cte.sefaz.mt.gov.br/ctews/services/CteCancelamento
MT	CTeInutilizacao	https://cte.sefaz.mt.gov.br/ctews/services/CteInutilizacao
MT	CTeConsultaProtocolo	https://cte.sefaz.mt.gov.br/ctews/services/CteConsulta
MT	CTeStatusServico	https://cte.sefaz.mt.gov.br/ctews/services/CteStatusServico
RS	CTeRecepcao	https://cte.sefaz.rs.gov.br/ws/cterecepcao/CteRecepcao.asmx
RS	CTeRetRecepcao	https://cte.sefaz.rs.gov.br/ws/cteretrecepcao/cteRetRecepcao.asmx
RS	CTeCancelamento	https://cte.sefaz.rs.gov.br/ws/ctecancelamento/ctecancelamento.asmx
RS	CTeInutilizacao	https://cte.sefaz.rs.gov.br/ws/cteinutilizacao/cteinutilizacao.asmx
RS	CTeConsultaProtocolo	https://cte.sefaz.rs.gov.br/ws/cteconsulta/CteConsulta.asmx
RS	CTeStatusServico	https://cte.sefaz.rs.gov.br/ws/ctestatusservico/CteStatusServico.asmx
SP	CTeRecepcao	https://nfe.fazenda.sp.gov.br/cteWEB/services/cteRecepcao.asmx
SP	CTeRetRecepcao	https://nfe.fazenda.sp.gov.br/cteWEB/services/cteRetRecepcao.asmx
SP	CTeCancelamento	https://nfe.fazenda.sp.gov.br/cteWEB/services/cteCancelamento.asmx
SP	CTeInutilizacao	https://nfe.fazenda.sp.gov.br/cteWEB/services/cteInutilizacao.asmx
SP	CTeConsultaProtocolo	https://nfe.fazenda.sp.gov.br/cteWEB/services/cteConsulta.asmx
SP	CTeStatusServico	https://nfe.fazenda.sp.gov.br/cteWEB/services/cteStatusServico.asmx

Figura 25: Relação dos serviços da SEFAZ
Fonte: BRASIL-PortalCTe (2009).

Caso haja problemas técnicos de responsabilidade da SEFAZ por indisponibilidade de acesso aos Web Services que impeça a emissão do conhecimento eletrônico de carga, o contribuinte tem como contingência a emissão em formulário contínuo do documento e posteriormente a regularização do mesmo junto à SEFAZ.

A seguir mais detalhes sobre cada um dos serviços disponibilizados pela SEFAZ para o CT-e.

4.3.1 *Web Service* – CteRecepção

Os Web Services de recepção de CT-e dos portais das Secretarias da Fazenda dos Estados são assíncronos e proporcionam a recepção de lotes de até 50 CT-e e tamanho máximo de 500kb para cada documento XML.

Na recepção é gerado um retorno de mensagem da transmissão com um recibo com as informações do número, data, hora, local de recebimento e tempo médio de

resposta do serviço dos últimos cinco minutos. Com o número do recibo é possível consultar o resultado do processamento do lote.

O serviço de recepção da SEFAZ realiza, no lote de CT-e, uma série de pré-validações antes de retornar a mensagem com o recibo. Algumas delas são:

- Validação do certificado de transmissão (protocolo SSL);
- Validação da versão do leiaute;
- Validação do arquivo em formato XML;
- Validação do Certificado e Assinatura digital.

Caso haja alguma divergência na validação inicial, uma mensagem de retorno com o devido código de erro é gerada e retornada ao contribuinte. Não havendo erro dessas validações iniciais, o lote de CT-e segue para a fila de entrada, onde o Web Service da SEFAZ executará o processamento de outras validações para cada CT-e enviado pelo lote.

Há nesse processo validações de regras de negócio, onde será visto se os campos obrigatórios estão presentes, se as datas estão válidas, se os CNPJ/CPF dos participantes estão de acordo com a legislação, se as inscrições estaduais estão corretas e uma série de outras checagens.

Os leiautes e mensagens de erros do Web Service de recepção e as regras de validação estão disponíveis no manual de integração do contribuinte. Ver em <http://www.cte.fazenda.gov.br/docs/Manual_CTe_v1.02.pdf> páginas 27 a 37.

Após a validação de CT-e será gerada uma mensagem de retorno com o número do protocolo e o resultado das validações para cada CT-e com os seguintes possíveis motivos. São eles:

- **Rejeição** - CT-e rejeitado por ocorrência de erros;
- **Autorização de uso** - CT-e válido;
- **Denegação de uso** - CT-e com alguma irregularidade fiscal.

A figura 26 mostra o fluxo do serviço de envio do lote de CT-e.

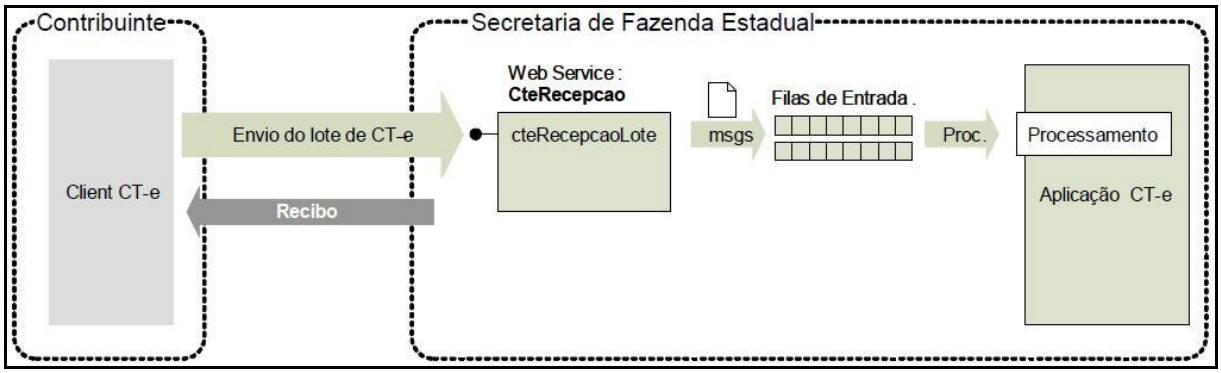


Figura 26: Transmissão de lote de CT-e
Fonte: BRASIL (2008, p. 27).

4.3.2 *Web Service* – CteRetRecepção

Os Web Services de retorno de recepção de CT-e dos portais das Secretarias da Fazenda dos Estados são assíncronos e tem como principal objetivo retornar o resultado do processamento do lote com seus respectivos CT-es.

A SEFAZ sugere que o intervalo de envio do lote até a consulta do resultado esteja em torno de quinze segundos. Caso o intervalo seja menor que esse período, a consulta poderá retornar com erro de código 105 que é “Lote em Processamento”.

O serviço de retorno executa algumas validações antes de retornar a mensagem com o resultado do processamento de CT-e. Uma dessas validações é a validação do certificado de transmissão.

No final do processamento as seguintes mensagens são possíveis para o lote. São elas:

- **Lote processado** - contém também o resultado individual de cada CT-e;

- **Lote em processamento** - o aplicativo do contribuinte deve fazer nova consulta;

- **Lote não localizado** - o aplicativo do contribuinte deve reenviar a mensagem;

- **Recibo ou CNPJ do requisitante com problemas** - o aplicativo do contribuinte deve sanar o problema.

A figura 27 mostra o fluxo do serviço de retorno da consulta do lote de CT-e.

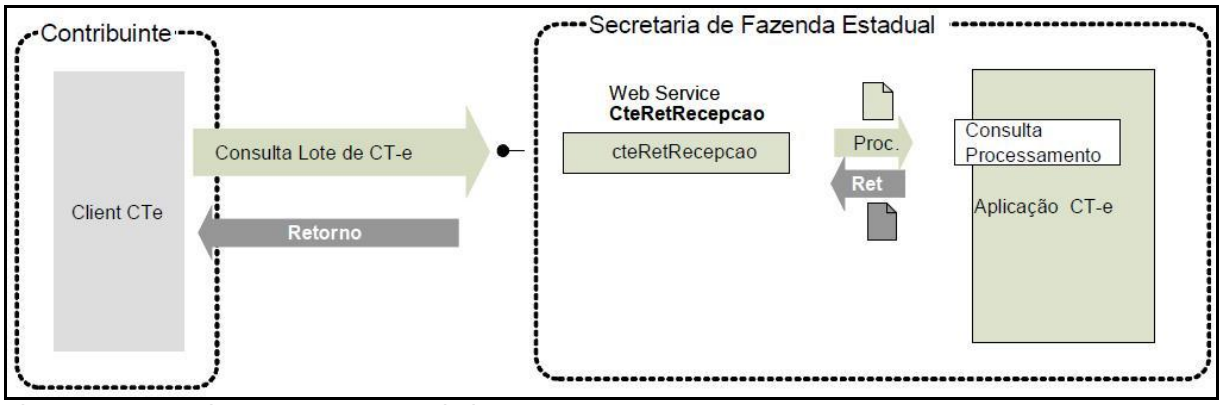
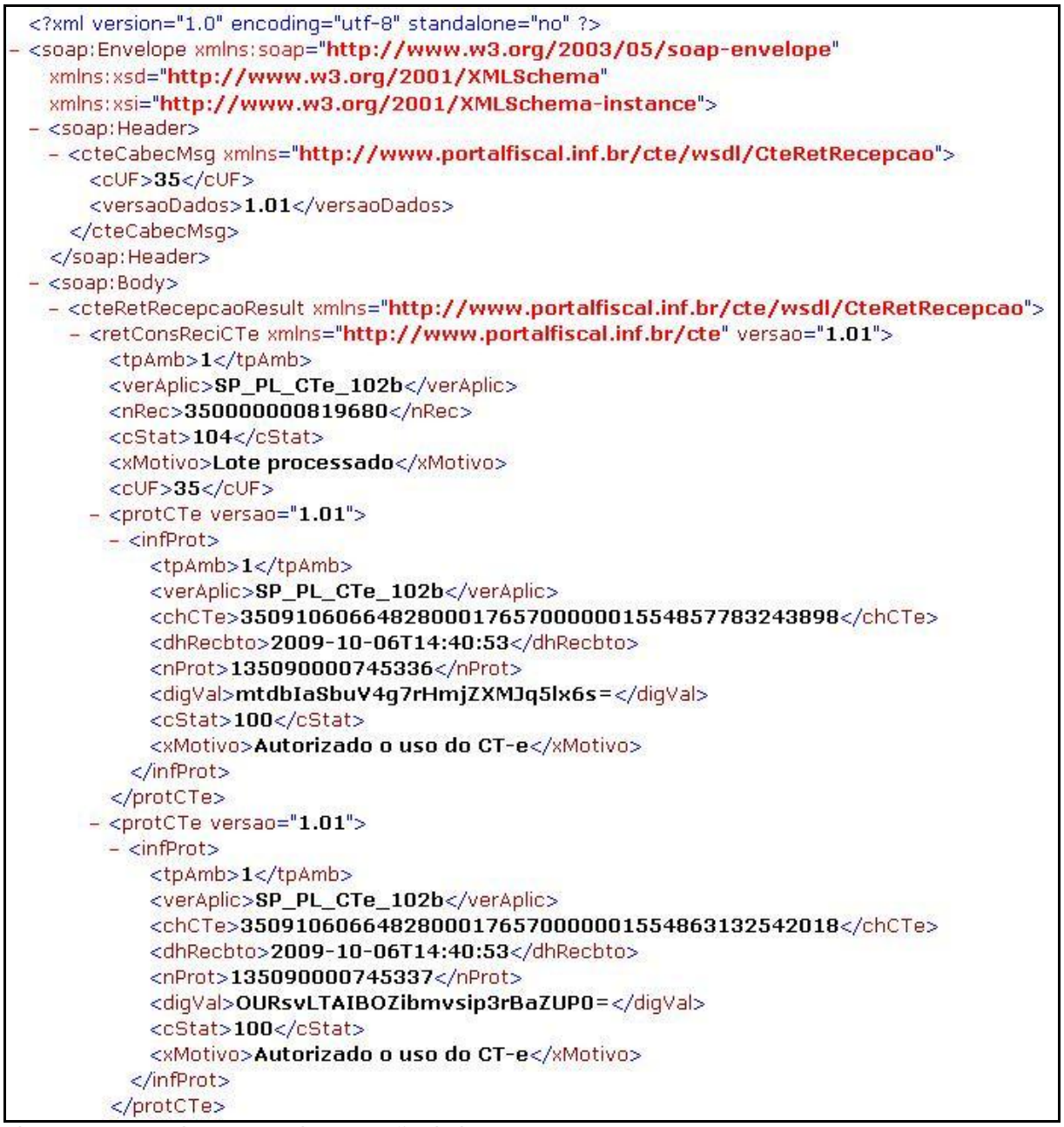


Figura 27: Consulta processamento de lote
Fonte: BRASIL (2008, p. 39).

A figura 28 exemplifica uma mensagem de retorno da recepção do lote de CT-e. Note que nesse exemplo, o campo <chCTE> contém a chave de acesso do CT-e e no campo <cStat> contém o código da mensagem, e no campo <xMotivo> contém a descrição. A mensagem “Autorizado o uso do CT-e” é uma das mensagens mais aguardadas de todo o processamento, pois ela reflete o êxito do modelo de envio, recepção e retorno da recepção do lote com seus respectivos CT-es, enviados pelo contribuinte.



```
<?xml version="1.0" encoding="utf-8" standalone="no" ?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soap:Header>
    <cteCabecMsg xmlns="http://www.portalfiscal.inf.br/cte/wsdl/CteRetRecepcao">
      <cUF>35</cUF>
      <versaoDados>1.01</versaoDados>
    </cteCabecMsg>
  </soap:Header>
  <soap:Body>
    <cteRetRecepcaoResult xmlns="http://www.portalfiscal.inf.br/cte/wsdl/CteRetRecepcao">
      <retConsReciCTe xmlns="http://www.portalfiscal.inf.br/cte" versao="1.01">
        <tpAmb>1</tpAmb>
        <verAplic>SP_PL_CTe_102b</verAplic>
        <nRec>350000000819680</nRec>
        <cStat>104</cStat>
        <xMotivo>Lote processado</xMotivo>
        <cUF>35</cUF>
        <protCTe versao="1.01">
          <infProt>
            <tpAmb>1</tpAmb>
            <verAplic>SP_PL_CTe_102b</verAplic>
            <chCTe>35091060664828000176570000001554857783243898</chCTe>
            <dhRecbto>2009-10-06T14:40:53</dhRecbto>
            <nProt>135090000745336</nProt>
            <digVal>mtdbIaSbuV4g7rHmjZXMJq5lx6s=</digVal>
            <cStat>100</cStat>
            <xMotivo>Autorizado o uso do CT-e</xMotivo>
          </infProt>
        </protCTe>
        <protCTe versao="1.01">
          <infProt>
            <tpAmb>1</tpAmb>
            <verAplic>SP_PL_CTe_102b</verAplic>
            <chCTe>35091060664828000176570000001554863132542018</chCTe>
            <dhRecbto>2009-10-06T14:40:53</dhRecbto>
            <nProt>135090000745337</nProt>
            <digVal>OURsvLTAIBOZibmvsip3rBaZUP0=</digVal>
            <cStat>100</cStat>
            <xMotivo>Autorizado o uso do CT-e</xMotivo>
          </infProt>
        </protCTe>
```


Figura 28: Consulta retorno de recepção do lote
Fonte: ATLAS, Transportes e Logística (2009).

4.3.3 *Web Service* – CteCancelamento

Os Web Services de cancelamento de CT-e dos portais das Secretarias da Fazenda dos Estados são síncronos e tem como principal função, o cancelamento de CT-e.

Nesse processo o Web Service de cancelamento recebe a solicitação do transmissor, processa a solicitação e retorna o resultado para o aplicativo do mesmo. O serviço de cancelamento cumpre certas validações antes de retornar a mensagem com o resultado do processamento. Uma dessas validações é a do

certificado de transmissão. No final do processamento se não ocorrer erros, a mensagem de “Cancelamento de CT-e homologado” é retornado ao transmissor.

A figura 29 ilustra o fluxo do serviço de cancelamento de lote de CT-e.

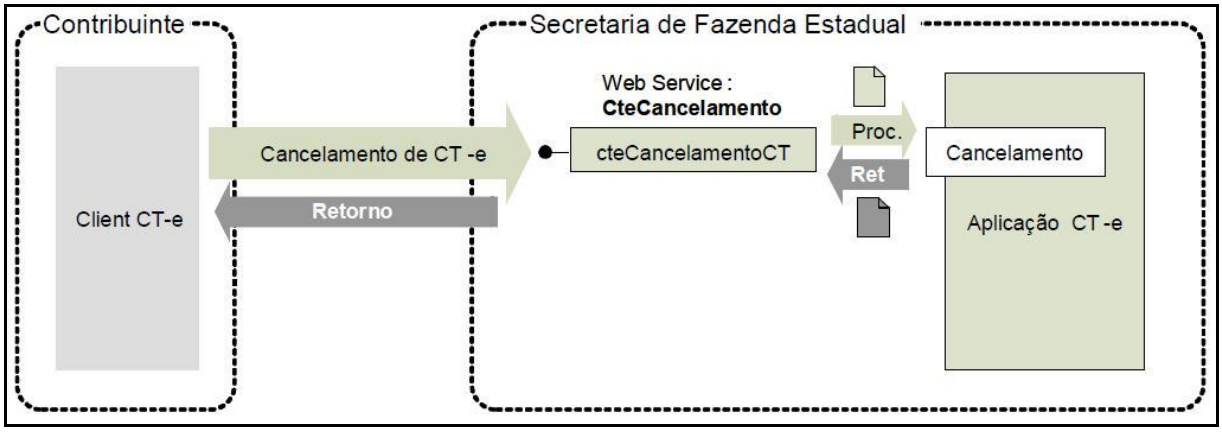


Figura 29: Cancelamento de CT-e
Fonte: BRASIL (2008, p. 44).

4.3.4 *Web Service* – CteInutilizacao

O serviço de inutilização disponibilizados pelos portais das Secretarias de Fazenda dos Estados é síncrono e tem como finalidade a inutilização de intervalo de números de conhecimentos de transportes eletrônicos.

O serviço de inutilização realiza algumas validações antes de retornar a mensagem com o resultado do processamento. Uma dessas validações é a do certificado de transmissão. No final do processamento se não ocorrer erros, a mensagem de “Inutilização de número homologado” é retornado ao transmissor. A figura 30 ilustra o fluxo do serviço de inutilização de CT-e.

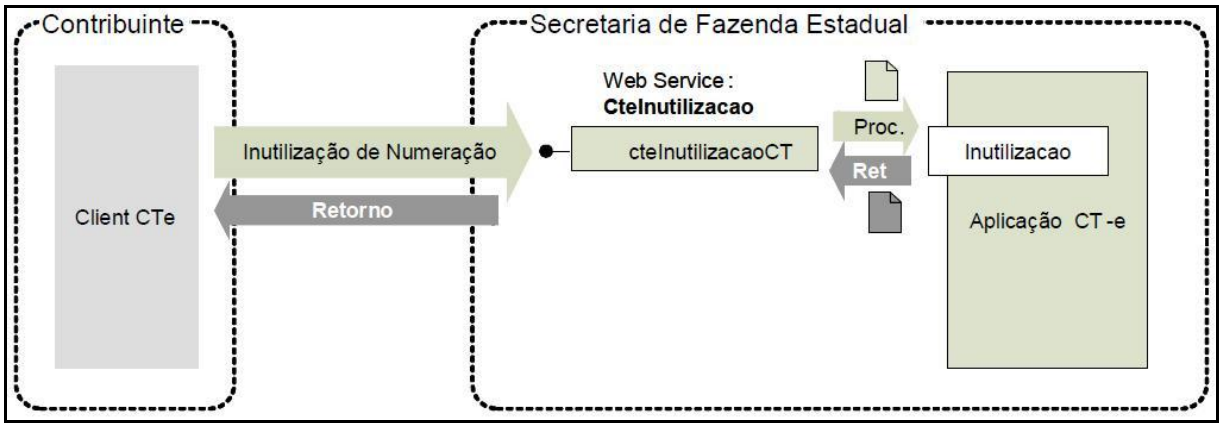


Figura 30: Inutilização de CT-e
Fonte: BRASIL (2008, p. 50).

4.3.5 *Web Service* – CteConsulta Protocolo

Os Web Services de consulta de CT-e dos portais das Secretarias de Fazenda dos Estados são síncronos e tem como principal objetivo consultar a situação atual do conhecimento eletrônico de carga. Nesse processo o Web Service de consulta recebe a solicitação do transmissor, processa a solicitação e retorna o resultado para o aplicativo do mesmo. O serviço de consulta cumpre certas validações antes de retornar a mensagem com o resultado do processamento. Uma dessas validações é a do certificado de transmissão. Após essas validações o aplicativo da SEFAZ consumirá a solicitação, validando a chave de acesso do CT-e, devolvendo a mensagem com a situação atual do CT-e.

A figura 31 ilustra o fluxo do serviço de consulta protocolo de CT-e.

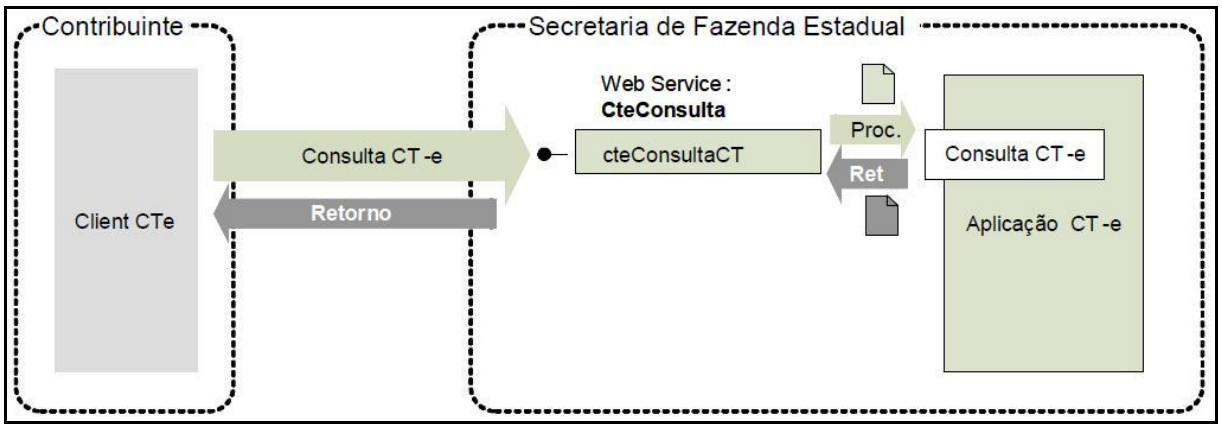


Figura 31: Consulta de protocolo CT-e
Fonte: BRASIL (2008, p. 55).

4.3.6 *Web Service* – CteStatusServico

Os Web Services de consulta de status de serviços dos portais das Secretarias de Fazenda dos Estados são síncronos e tem como função consultar a situação dos Web Services. Nesse processo o Web Service de consulta de status recebe a solicitação do transmissor, processa a solicitação e retorna o resultado para o aplicativo do mesmo. O serviço de consulta de status cumpre algumas validações antes de retornar a mensagem com o resultado do processamento. Uma dessas validações é a do certificado de transmissão. Após essas validações o aplicativo da SEFAZ consumirá a solicitação, devolvendo a mensagem com a situação atual do Web Service.

A figura 32 ilustra o fluxo do serviço de consulta de status de CT-e.

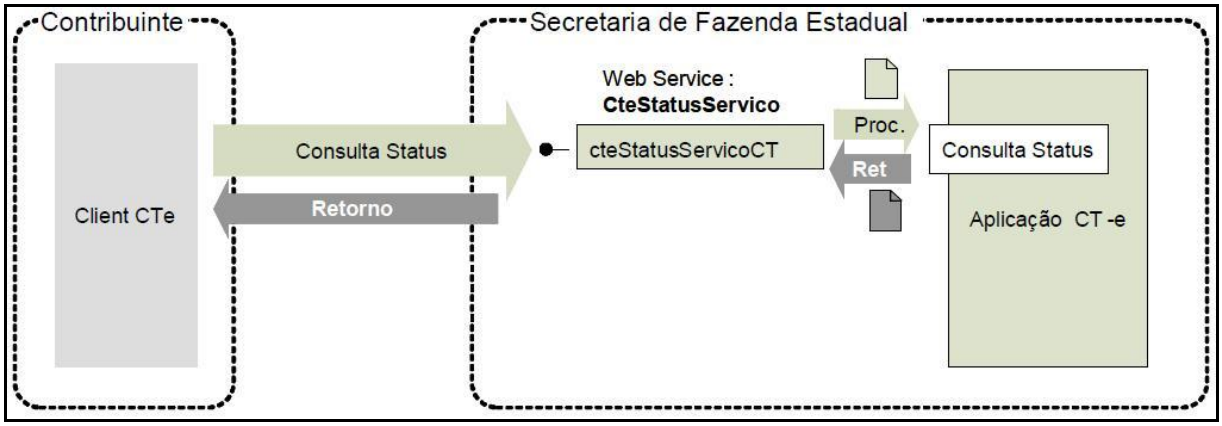


Figura 32: Consulta de status do Web Service da SEFAZ
Fonte: BRASIL (2008, p. 59).

4.3.7 *Web Service* – CadConsultaCadastro

Os Web Services de consulta de cadastro dos portais das Secretarias de Fazenda dos Estados são síncronos e tem como objetivo consultar o cadastro de contribuintes do ICMS da unidade federada. Nesse processo o Web Service de consulta de cadastro recebe a solicitação do transmissor, processa a solicitação e retorna o resultado para o aplicativo do mesmo. O serviço de consulta de cadastro cumpre algumas validações antes de retornar a mensagem com o resultado do processamento. Uma dessas validações é a do certificado de transmissão. Após essas validações o aplicativo da SEFAZ consumirá a solicitação, devolvendo a mensagem com a situação atual do contribuinte.

A figura 33 ilustra o fluxo do serviço de consulta de cadastro do contribuinte.

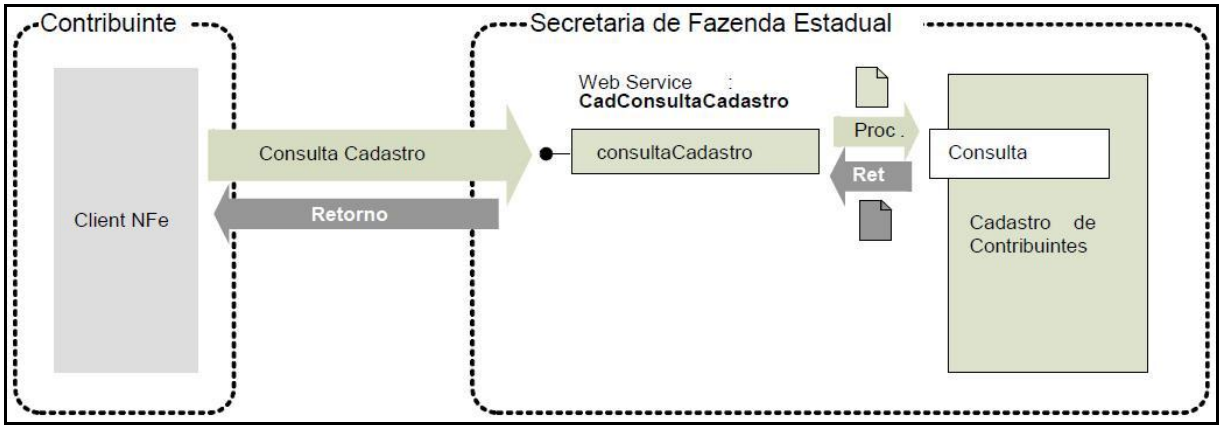


Figura 33: Consulta de cadastro do contribuinte
Fonte: BRASIL (2008, p. 63).

4.4 BENEFÍCIOS

De acordo com BRASIL-MS (2009), o Projeto Conhecimento Eletrônico de Carga beneficia todos os envolvidos no processo de transportes de carga, desde o contribuinte tomador do serviço, do prestador, do destinatário até as Secretarias de Fazendas dos Estados, as SEFAZ. Dentre os benefícios destacam-se:

A redução do custo de impressão e aquisição do documento fiscal é uma das vantagens desse projeto, visto que com a simples substituição de cinco vias do documento anterior por uma folha de sulfite tamanho A4, que é necessária para realizar o acompanhamento da prestação de serviço permite menor impacto no meio ambiente.

O custo de armazenagem e o processo logístico que envolve o arquivamento de documentos fiscais são muito grandes. Com o projeto CT-e, esse custo é reduzido drasticamente, pois, como o documento fiscal é emitido eletronicamente, não há necessidade de grandes ambientes para arquivamento.

Uma vantagem para o prestador de serviço de transportes é o fato da redução de tempo de parada de caminhões nos Postos Fiscais de Fronteira dos Estados, onde são efetuadas verificações de mercadorias e situações fiscais. Nessas paradas a simples consulta da situação do CT-e possibilita a liberação desses veículos com maior rapidez.

Para o fisco, alguns dos benefícios com a implantação do projeto são:

- Confiabilidade no documento fiscal;
- Controle fiscal com compartilhamento de informação entre as SEFAZ;
- Redução de custo no processo de controle de Fronteira;
- Diminuição da sonegação fiscal e consequente aumento da arrecadação.

Nos gráficos a seguir são mostrados por um período de quatro meses a evolução da emissão de CT-e e respectivos valores gerados.

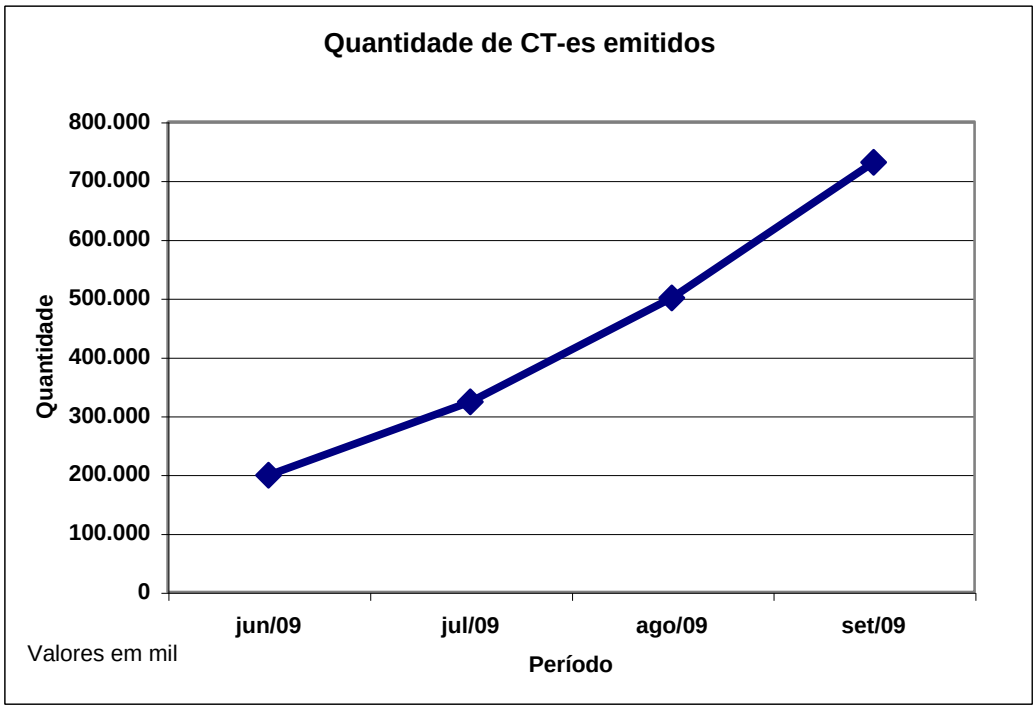


Gráfico 1 Evolução da emissão de CT-es

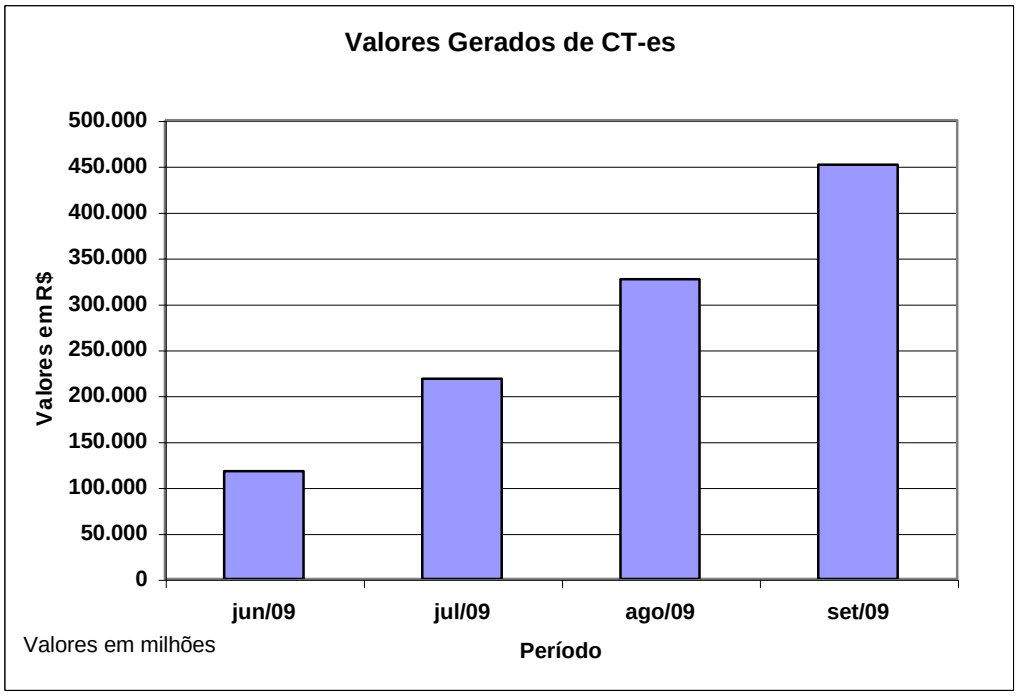


Gráfico 2 Evolução dos valores gerados

Quanto aos gráficos apresentados acima, na demonstração da evolução da quantidade de CT-es emitidos, tomando-se por base a implantação em 02/03/2009,

observa-se que o aumento da adesão ao projeto vem crescendo substancialmente mês a mês. No primeiro mês medido, o aumento foi cerca de 63% em relação ao mês anterior. O percentual médio de 54% é observado nessa medição.

O crescimento reflete a adoção pelas empresas participantes e as que estão aderindo ao projeto e evidenciando a utilização do projeto CT-e.

5 CONCLUSÃO

Os *Web Services* vieram resolver a integração de sistemas, independente de sistemas operacionais e linguagens de programação. Suas características são muito bem aceitas pela comunidade do mundo da tecnologia da informação e comunicação. Os mais importantes fabricantes da indústria de software e hardware apóiam os padrões abertos criados, ou seja, os HTTP, XML, SOAP, WSDL e UDDI. Com esse apoio as empresas ficam seguras de utilizarem essas tecnologias.

Sendo assim as empresas, utilizando os Web Services, vêem nessa tecnologia a solução para integração de sistemas heterogêneos compostos pelo legado de anos com utilização de várias tecnologias de diferentes plataformas para desenvolvimento de sistemas.

A preocupação constante com segurança na Web faz com que novas tecnologias, que até então não eram muito utilizadas passem a sê-la, pois, o comércio eletrônico e outros tipos projetos de TI, vêm ano a ano consumindo uma demanda muito forte, no sentido das empresas focarem os recursos técnicos e comerciais voltados para a tecnologia de segurança de sistemas. Há muito que temos disponíveis, as certificações digitais, as assinaturas digitais e uma série de outros artifícios para inibir e coibir a ação de vândalos da Internet.

Toda essa tecnologia disponível fez com que o governo desencadeasse uma série de estudos no sentido de utilizar esses recursos tecnológicos para o desenvolvimento de sistemas que pudessem gerir com mais rigor os processos que o governo controla e gerencia. O projeto do Conhecimento Eletrônico de Carga é o resultado de um desses estudos. O projeto foi cuidadosamente estudado, planejado, organizado e executado dentro de um cronograma bem elaborado.

As empresas já participantes do projeto e aquelas que aderirem podem imediatamente se utilizarem dos muitos benefícios diretos gerados pela utilização dos *Web Services* descritos anteriormente.

A crescente utilização das empresas de todos os tipos de segmento e também do governo brasileiro pelos *Web Services* sugere que, uma seqüência de estudos seja proposta, no sentido de ampliar e difundir o uso dos *Web Services*.

Como trabalho futuro, uma análise de evidências de utilização dos Web Services pelo governo, como forma de verificação de como os órgãos governamentais estão utilizando a tecnologia da informação e comunicação em prol do desenvolvimento. Com esse trabalho verificar também, se os impostos pagos pelos contribuintes estão sendo destinados para tal finalidade.

REFERÊNCIAS BIBLIOGRÁFICAS

ABINADER, Jorge Abílio; LINS, Rafael Dueire. ***Web Services* em Java**. 1. ed. Rio de Janeiro: Brasport, 2006.

ALONSO, Gustavo et al. ***Web Services*: *Concepts, Architectures and Applications***. 1. ed. *Germany*: *Springer-Velag Berlin Heidelberg New York*, 2004.

ATLAS, Transportes e Logística. **Empresa de Transportes**. 2009.

AUSTIN, Daniel et al. ***Web Services Arquitecture Requirements***. 2004. Disponível em <http://w3c.org/TR/wsa-reqs>. Acesso em 20/04/2009.

BRASIL. **Manual de Integração** - Contribuinte. 2008. Disponível em <http://www.cte.fazenda.gov.br/docs/Manual_CTe_v1.02.pdf>. Acesso em 22/04/2009.

BRASIL-ITI. **Presidência da República.** Instituto Nacional de Tecnologia da Informação. 2009. Disponível em <http://www.iti.gov.br>. Acesso em 06/09/2009.

BRASIL-MS. **Governo do Estado de Mato Grosso do Sul.** Conhecimento de Transporte Eletrônico. 2009. Disponível em <http://www.cte.ms.gov.br>. Acesso em 02/10/2009.

BRASIL-PortalCTe. **Ministério da Fazenda.** Conhecimento de Transporte Eletrônico. 2009. Disponível em: <http://www.cte.fazenda.gov.br/legislacao.aspx>. Acesso em 08/09/2009.

BRASIL-RS. **SEFAZ RS.** Secretaria da Fazenda. 2009. Disponível em <http://www.sefaz.rs.gov.br/CTE/CTE-Implantacao.aspx>. Acesso em 03/10/2009.

BRASIL-SC. **Governo do Estado de Santa Catarina.** Conhecimento de Transporte Eletrônico. 2009. Disponível em <http://www.cte.sef.sc.gov.br>. Acesso em 02/10/2009.

BRASIL-SP. **Governo do Estado de São Paulo.** 2009. Disponível em <http://www.fazenda.sp.gov.br/cte>. Acesso em 30/06/2009.

CETIC-**Evolução Hosts.** 2009. Disponível em <http:cetic.br/hosts/index.htm>. Acesso em 16/05/2009.

CHRISTENSEN, Erik et al. ***Web Services Description Language* (WSDL) 1.1.** 2001. Disponível em <http://www.w3.org/TR/2001/NOTE-wsdl-20010315>. Acesso em 31/05/2009.

CIRNE, Walfredo. **Desenvolvimento de Aplicações Distribuídas.** 2002. Disponível em <http://walfredo.dsc.ufcg.edu.br/talks/apdist-maceio.ppt>. Acesso em 23/05/2009.

CLEMENT, Luc et al. **UDDI *Version* 3.0.2. UDDI Spec TC.** 2004. Disponível em <http://uddi.org/pubs/uddi-v3.0.2-20041019.htm>. Acesso em 06/06/2009.

COULOURIS, George; DOLLIMORE, Jean; KINDBERG Tim. **Sistemas Distribuídos: Conceitos e Projeto.** Tradução: João Tortello. 4. ed. Porto Alegre: Bookman, 2007.

CRUZ, Sérgio Manuel Serra da. **Serviços Web** - Uma breve introdução (parte I). 2005. Núcleo de computação Eletrônica da Universidade Federal do Rio de Janeiro. Disponível em <http://www.nce.ufrj.br/conceito/artigos/2005/01p2-1.htm>. Acesso em 30/05/2009.

FARRELL, Stephen et al. **Segurança para *Web Services*.** 2009. VeriSign. Grupo de Pesquisa e Produtos Avançados. Disponível em <http://certisign.com.br/treinamento/guias-gratuitos/pdf/webservices.pdf>. Acesso em 14/02/2009.

GUDGIN, Martin; HADLEY, Marc; MENDELSOHN, Noah et al. **SOAP Version 1.2 W3C Recommendation.** 2007. Disponível em <http://www.w3.org/TR/soap12-part1/#intro>. Acesso em 30/05/2009.

JORGENSEN, David. **Desenvolvendo Serviços Web .Net com XML.** 1. ed. Rio de Janeiro: Alta Books, 2002.

KREGER, Heather. ***Web Services Conceptual Architecture (*WSCA *1.0).*** 2001. Disponível em <http://www.cs.uoi.gr/~zarras/mdw-ws/WebServicesConceptualArchitectu2.pdf>. Acesso em 23/05/2009.

MAIA, Luiz Paulo, PAGLIUSI, Paulo Sergio. **Criptografia e Certificação Digital.** 2009. Disponível em <http://www.training.com.br/lpmaia/pub_seg_cripto.htm>. Acesso em 10/08/2009.

MATOS, Luis Salgado de. **Dicionário de Filosofia, Moral e Política.** Instituto de Filosofia da Linguagem, Universidade Nova Lisboa. Portugal. Disponível em <http://www.ifl.pt/ifl_old/dfmp_files/seguranca.pdf>. Acesso em 08/08/2009.

MORIMOTO, Carlos E. **Termos Técnicos GDH.** 2009. Disponível em < http://www.guiadohardware.net/termos/>. Acesso em 23/05/2009.

PINHEIRO, José Mauricio Santos. **Redes de Perímetro.** 2004. Disponível em < http://www.projetoderedes.com.br/artigos/artigo_redes_de_perimetro.php>. Acesso em 04/09/2009.

PIRES, Facciolo Daniel. **WSDL - Conceitos.** 2005. Disponível em <http://sites.ffclrp.usp.br/ccp/(SEM%207)/MATDID/SW/Aula_03-WSDL-14934.ppt>. Acesso em 04/06/2009.

PROJETO-Firewall. **Projeto de rede com firewall.** Disponível em < http://www.projetoderedes.com.br/artigos/artigo_redes_de_perimetro.php>. Acesso em 04/09/2009.

PUTTE, Geert Van de et al. ***Using Web Services for Business Integration*.** 1. ed. *New York*: IBM, 2004. Disponível em <http://www.redbooks.ibm.com/redbooks/pdfs/sg246583.pdf>. Acesso em 31/05/2009

RAY, Erik T. ***Learning* XML.** 1. ed. Ebook. 2001. Disponível em < http://www.flazx.com/download13376.php>. Acesso em 31/05/2009.

RECKZIEGEL, Mauricio. **Descrevendo um Web Service - WSDL.** 2006. Disponível em <http://imasters.uol.com.br/artigo/4422/webservices/descrevendo_um_web_service_-_wsdl/>. Acesso em 31/05/2009.

SILVA FILHO, Joel Guilherme da. **Criptografia, Chaves Públicas e Assinatura Digital para leigos.** 2009. Disponível em <http://sbis.org.br/Criptografia.doc>. Acesso em 08/08/2009.

SOTOMAYOR, Borja. ***The Globus Toolkit 4 Programmer´s Tutorial*.** 2005. *University of Chicago, Department of Computer Science*. Disponível em <http://gdp.globus.org/gt4-tutorial/multiplehtml/ch01s02.html>. Acesso em 30/05/2009.

SOUZA JR, Rafael T. de; PUTTINI, Ricardo S. **Criptografia e Segurança de Redes de Computadores.** 2009. Disponível em <http://www.redes.unb.br/security/firewall/firewall.htm>. Acesso em 04/09/2009.

TANENBAUM, Andrew S. ***Distributed Systems*: *Principles and Paradigms*.** 4. ed. *New Jersey*: *Prentice Hall* PTR, 2003.

W3C-SOAP. ***World Wide Web Consortium*.** 2007. Disponível em <http://www.w3.org/TR/2007/REC-soap12-part0-20070427>. Acesso em 22/08/2009.