UNIVERSIDADE FEDERAL DE SANTA CATARINA
CURSO DE GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Iuri Campana
Victor Simas Silva

Sistema Seguro de Compras

Trabalho de Conclusão de Curso

Fernando Carlos Pereira
Orientador

Prof. Ricardo Felipe Custódio, Dr.
Co-Orientador

Florianópolis, Fevereiro de 2003

Sistema Seguro de Compras

Iuri Campana
Victor Simas Silva

Este Trabalho de Conclusão de Curso foi aprovado em sua forma final pelo Curso de Ciência da Computação da Universidade Federal de Santa catarina.

Prof. José Mazzuco Júnior, Dr.

Coordenador do Curso

Banca Examinadora

Fernando Carlos Pereira

Orientador

Prof. Ricardo Felipe Custódio, Dr.

Co-Orientador

Prof. Carlos Roberto De Rolt, Dr.

Prof. Júlio da Silva Dias, M.Eng.

”Só as pessoas de valor lutam contra os obstáculos em busca da felicidade!”(Autor desconhecido)

À todas as pessoas que se interessam em criptografia.

Agradecimentos

Primeiramente gostaria de agradecer às pessoas que me ajudaram mais diretamente a desenvolver este projeto com muita responsabilidade e dedicação, meu orientador Fernando Carlos Pereira, meu co-orientador Prof. Dr. Ricardo Felipe Custódio e meu colega Iuri Campana.

Ao prof. Custódio, seu profundo conhecimento no assunto que sanou todas nossas dúvidas, não chegaríamos tão longe se não fosse por suas ajudas. E aos membros da banca prof. Carlos Roberto De Rolt e prof. Júlio da Silva Dias.

Gostaria de dar um agradecimento especial ao meu pai por além de todo incentivo que recebo, nos concedeu um micro na Internet para realizarmos nossos testes.

A minha namorada Fernanda por me ajudar tanto nas revisões do texto quanto emocionalmente durante este período do projeto

A minha família, por me ajudar durante todo esse caminho me dando o embasamento para tudo o que sou hoje.

E por último aos meus amigos pelos momentos de descontração que temos juntos.

Victor Simas Silva

Gostaria de agradecer a o meu orientador Fernando Carlos Pereira, pelo apoio e todo ensinamento passado. Meus colegas que contribuíram com o trabalho: Victor Simas Silva e Marcelo Brocardo. Agradeço também aos meus amigos, que sempre estiveram ao meu lado durante mais essa jornada: Daniel Nizer, Adriano Ferlin, Thiago Linhares e Joáber Cavichioli. Um agradecimento especial para o meu co-orientador Prof. Dr. Ricardo Felipe Custódio por todo o tempo dispensado para aprimorar meu aprendizado, ao prof Dr. Carlos De Rolt e ao prof. M.Eng. Júlio Dias.

A minha mãe, Regina Campana pelo incentivo e apoio, aos meus irmãos Priscila, Dimitri e Samya. Ao meu pai Nivaldo Campana que sempre esteve ao meu lado.

E agradeço especialmente à minha esposa Sabrina Gonçalves Campana, pela compreensão e carinho durante a realização deste trabalho.

Iuri Campana

Sumário

Lista de Figuras

Lista de Siglas

AC	Autoridade Certificadora
AR	Autoridade de Registro
CPF	Cadastro de Pessoa Fsica
ICP	Infra-estrutura de Chave Pblica
LabSEC	Laboratrio de Segurana em Computao - UFSC
LCR	Lista de Certificados Revogados
SGBD	Sistema de Gerenciamento de Banco de Dados
API	Application Programming Interface
ASP	Active Server Pages
ASCII	Amsterdam Subversive Code for Information Interchange
BASIC	Beginner’s All-purpose Symbolic Instruction Code
CGI	Common Gateway Interface
CLR	Common Language Runtime
COM	Component Object Model
CSP	Cryptographic Service Providers
DLL	Dynamic-link library
ERWin	Entity Relationship for Windows
FI	Form Interpreter
FTP	File Transfer Protocol
HTML	Hypertext Markup Format
HTTP	Hypertext Transfer Protocol
IIS	Internet Information Service
IP	Internet Protocol

ISAPI	Internet Server Advanced Programming Interface
JSP	JavaServer Pages
MAC	Middle Access Control
NCSA	National Center for Supercomputing Applications
NIST	National Institute of Standards and Technology
PHP	Hypertext Preprocessor
SQL	Structured Query Language
RSA	Rivest-Shamir-Adelman
SSL	Secure Socket Layer
TCP	Transmission Control Protocol
VB	Visual Basic

Resumo

Este trabalho pretende informatizar, de modo seguro, os processos de compras, que se baseiam em modelos de licitação, das organizações privadas ou públicas. A implementação do trabalho será através de uma aplicação web e um protocolo criptográfico que garantirá aos usuários maior segurança e confiabilidade.

Desta maneira, os processos de compras poderão ser realizados via internet, onde os participantes das licitações sejam compradores ou fornecedores, contarão com a agilidade e comodidade que este sistema oferecerá. Assim o tempo e o custo serão consideravelmente reduzidos na produção e análise de propostas, além de todos estarem certos de que o sistema é íntegro e imune a violações.

O presente trabalho é uma monografia de conclusão de curso do programa de graduação da Computação da UFSC. Este documento trata de protocolos criptográficos como alternativa de solução ao processo de compras via Internet.

Abstract

This work intends to turn today's process into a electronic process, in a safe way, the purchase processes, which are based on private or public organizations auction models. The work implementations were made through a web application and a cryptology protocol that assures more safety and trust to the users. This way, the purchase processes can be done through the Internet, where the auctions participants, either buyers or sellers, will be able to count on the easiness and commodity of the purchases. So the costs and time of the purposes production and analysis will be reduced, besides the fact that everyone will be sure that the process is trustable and immune to violations.

Capítulo 1

Introdução

A cada dia pode-se evidenciar os grandes câmbios que podem realizar-se com novas tecnologias cada vez mais poderosas de informática, mas desde que as mesmas operem com efetividade [IBA 03].

Os ataques provocados por fontes internas em um sistema causam mais danos do que intrusos. A maior exposição a ameaças à segurança na computação se deve a vários motivos [LIN 03]:

- Novos modelos de negócios. O setor público está seguindo o exemplo do setor privado, com uns cinco anos de atraso;
- Crescimento exponencial do uso da informática: os computadores e redes se insinuaram em quase todas as áreas das nossas vidas;
- Custos reduzidos: A tecnologia atual é barata. Não importa qual seja a medida usada, os custos da informática básica estão muito mais baixos do que em qualquer época passada, e o custo das novas tecnologias está decrescendo mais rapidamente do que decrescia poucos anos atrás, por causa do rápido progresso e da maior concorrência.

À medida que as tecnologias evoluem, a segurança na informática continuará a melhorar em termos de eficácia e este deve ser o objetivo dos profissionais da informática.

1.1 Processos de Compra

Compras são transações comerciais e podem ser definidas como sendo uma negociação entre duas partes: comprador; e fornecedor. O fornecedor transfere o domínio de um produto ou presta um serviço ao comprador em troca de remuneração financeira imediata, parcelada ou futura conforme acordado previamente.

Transações comerciais envolvendo compra de produtos e contratação de serviços é usualmente envolvida por quatro etapas básicas, como a verificação de necessidades, a análise das propostas, a seleção e a compra, as quais são necessárias para que sejam assegurados a satisfação e os direitos do comprador e do fornecedor, durante e após a transação comercial.

Este trabalho considera dois tipos de compradores, o comprador privado sendo ele uma pessoa jurídica e representado pelas empresas privadas e o comprador público representado pelas Administrações Públicas Federais, Estaduais, Municipais e Distritais.

Os processos de compras podem ser definidos como sendo a aquisição de bens por meio de remuneração. Neste trabalho será enfatizado os processos de licitações.

Denomina-se licitação a todo processo administrativo, através do qual a Administração Pública, realiza a coleta de propostas para fornecimento de materiais, ou serviços, dentre as quais selecionará aquele que julgar mais vantajosa para a celebração de um contrato [FED 03].

Os princípios da licitação são [FED 03]:

- Legalidade: Obriga o administrador a somente fazer o que a lei autoriza, não prevalecendo na Administração Pública a idéia de que o não proibido é necessariamente permitido;

- Impessoalidade: Veda os ”apadrinhamentos”, impondo que o processo licitatório deva estar ao alcance de todos os interessados;

- Moralidade: O licitador e licitante devem observar uma conduta honesta e honrada;

- Igualdade: Assegura iguais oportunidades a todos os interessados na licitação;

- Publicidade: Obriga a divulgação plena de todos os atos da licitação;
- Probidade Administrativa: Obriga o administrador a sempre visar o interesse do povo;
- Vinculação ao Instrumento Convocatório: impede a utilização, após iniciado o procedimento licitatório, de critérios diferentes daqueles estabelecidos no ato convocatório;
- Julgamento Objetivo: Afasta qualquer tipo de discricionariedade na avaliação das propostas.

Em organizações públicas, a participação em licitações requer todo um planejamento detalhado dos projetos básicos e executivos.

Estes projetos incluem: segurança, funcionalidade e adequação ao interesse público, economia na execução, conservação e operação, possibilidade de emprego de mão-de-obra, materiais, tecnologia e matérias-primas existentes no local para execução, conservação e operação, facilidade na execução, conservação e operação, sem prejuízo da durabilidade da obra ou do serviço, adoção das normas técnicas, de saúde e de segurança do trabalho adequadas e impacto ambiental. (Redação dada pela Lei n 8.883, de 8.6.94)

Apesar de algumas organizações privadas adotarem a legislação como procedimentos no processo de compras, a maioria possui procedimentos personalizados, ou seja, utilizam os mesmos procedimentos das organizações públicas, como a licitação, porém adaptados à sua realidade.

Neste sentido, será abordada com mais ênfase o processo de compras do setor público.

Basicamente, as licitações são compostas de etapas como [FED 03]:

1. Requisição do Interessado: É o documento que dá origem ao processo licitatório;
2. Estimativa de Valor: Permite concluir se há a obrigatoriedade de adoção de Licitação, permitindo a escolha exata da modalidade de Licitação a ser utilizada. É peça fundamental na fase de classificação, pois é utilizada para avaliação das propostas,

permitindo ao licitador verificar se a proposta apresentada é inexequível ou superfaturada;

3. Autorização de Despesa: Demonstra que uma autoridade competente avaliou a proposta de compra e concordou com a necessidade do objeto pretendido;

4. Elaboração do Instrumento Convocatório: É o meio através do qual a Administração Pública leva o certame ao conhecimento público;

5. Análise e Aprovação Jurídica do Edital: O instrumento convocatório é levado para análise e aprovação da assessoria jurídica da Administração;

6. Divulgação: Aprovada juridicamente, elabora-se o documento definitivo e passa-se à fase de divulgação, através da publicação do Aviso na imprensa oficial.

Segundo a legislação, as compras, sempre que possível, devem [FED 02]:

- Atender ao princípio da padronização, que imponha compatibilidade de especificações técnicas e de desempenho, observadas, quando for o caso, as condições de manutenção, assistência técnica e garantia oferecida;
- Ser processadas através de sistema de registro de preços;
- Ser subdivididas em tantas parcelas quantas necessárias para aproveitar as peculiaridades do mercado, visando economicidade;
- Balizar-se pelos preços praticados no âmbito dos órgãos e entidades da Administração Pública.

1.2 Composição e Compreensão

Para obter-se uma boa compreensão e um aprofundamento do texto é necessário que o leitor compreenda os conceitos de **concorrência, tomada de preço e carta convite**, que são modalidades de licitações. A seguir seguem seus conceitos baseados na legislação [FED 02].

Concorrência *é a modalidade de licitação entre quaisquer interessados que, na fase inicial de habilitação preliminar, comprovem possuir os requisitos mínimos de qualificação exigidos no edital para execução de seu objeto.*

Tomada de preços *é a modalidade de licitação entre interessados devidamente cadastrados ou que atenderem a todas as condições exigidas para cadastramento até o terceiro dia anterior à data do recebimento das propostas, observada a necessária qualificação.*

Carta Convite *é a modalidade de licitação entre interessados do ramo pertinente ao seu objeto, cadastrados ou não, escolhidos e convidados em número mínimo de 3 (três) pela unidade administrativa, a qual afixará, em local apropriado, cópia do instrumento convocatório e o estenderá aos demais cadastrados na correspondente especialidade que manifestarem seu interesse com antecedência de até 24 (vinte e quatro) horas da apresentação das propostas.*

Outro conceito fundamental para compreensão do modelo de licitação é o seu processamento e julgamento [FED 02]:

- Abertura dos envelopes contendo a documentação relativa à habilitação dos concorrentes, e sua apreciação;

- Devolução dos envelopes fechados aos concorrentes inabilitados, contendo as respectivas propostas, desde que não tenha havido recurso ou após sua denegação;

- Abertura dos envelopes contendo as propostas dos concorrentes habilitados, desde que transcorrido o prazo sem interposição de recurso, ou tenha havido desistência expressa, ou após o julgamento dos recursos interpostos;

- Verificação da conformidade de cada proposta com os requisitos do edital e, conforme o caso, com os preços correntes no mercado ou fixados por órgão oficial competente, ou ainda com os constantes do sistema de registro de preços, os quais deverão ser devidamente registrados na ata de julgamento, promovendo-se a desclassificação das propostas desconformes ou incompatíveis;

- Julgamento e classificação das propostas de acordo com os critérios de avaliação constantes do edital;
- Deliberação da autoridade competente quanto à homologação e adjudicação do objeto da licitação.

1.3 Objetivos

Este trabalho tem como objetivo geral desenvolver um sistema que permita a realização de processos de licitação pública de modo seguro via Internet. Neste sentido, o trabalho deverá:

1. Realizar um estudo sobre o mecanismo de funcionamento de processos de licitação pública;
2. Pesquisar sobre técnicas e protocolos de Criptografia;
3. Pesquisar sobre as ferramentas de desenvolvimento necessárias à implementação do sistema;
4. Implementar um protocolo criptográfico, o qual será responsável por manter a confidencialidade das propostas comerciais durante o tempo necessário;
5. Retratar o ambiente de compras de licitações no ambiente digital;
6. Utilizar recursos de criptografia para garantir a segurança do sistema.

1.4 Materiais e Métodos

Este trabalho foi desenvolvido com base em estudos realizados sobre processos de licitações pública, tecnologias de desenvolvimento voltadas à Internet, protocolos e técnicas de criptografia.

Estes estudos foram realizados através de livros, artigos científicos e manuais das tecnologias utilizadas.

No trabalho será desenvolvido um site em que compradores e fornecedores terão acessos a áreas distintas, onde eles terão disponíveis os recursos necessários para interagirem em processos de licitação.

O sistema utiliza recursos de criptografia, tais como certificados digitais e cifragem de dados, e com isso disponibiliza aos seus usuários um ambiente robusto e confiável para o envio e recebimento de informações.

1.5 Trabalhos relacionados

Este trabalho implementa um protocolo criptográfico proposto na dissertação de mestrado de **??**. Os modelos para a solução são todas baseados e fundamentados nesta dissertação.

Um Site interessante que está muito relacionado é o Portal de Compras do Governo Federal - ComprasNet, citado na bibliografia, que funcionalmente se assemelha muito com o trabalho aqui desenvolvido.

1.6 Motivação e Justificativa

Nos dias de hoje, muitas das licitações são realizadas através de papéis. Ou seja, as propostas são enviadas através de envelopes, armazenadas até uma determinada data, para depois então, serem abertos e analisados.

Como podemos perceber, há muita margem para a insegurança dos participantes. Podemos simular inúmeros métodos de corromper este sistema para beneficiar um ou outro participante. Este problema merece grande atenção, pois é o agente causador de grandes prejuízos financeiros em quaisquer organizações.

Com o avanço da tecnologia, os meios eletrônicos passaram a necessitar de modos seguros de armazenagem e envio de dados. Neste sentido, com o advento da criptografia, é possível que o processo de compras seja implementado via Internet.

1.7 Organização do Texto

Será feita uma rápida abordagem no capítulo 2 sobre os principais fundamentos da criptografia, onde serão conceituados a criptografia simétrica, criptografia assimétrica, função resumo, assinatura digital, Infra-estrutura de chaves públicas, esquemas de compartilhamento de segredos e redes misturadoras. No capítulo 3 é apresentada a conceituação e o funcionamento das diversas tecnologias utilizadas neste trabalho. Por fim no capítulo 4 será apresentada a solução escolhida e implementada.

Capítulo 2

Fundamentos de Criptografia

2.1 Introdução

As comunicações seguras são compostas por três áreas: privacidade, autenticação, e integridade.

Privacidade - Para ter a privacidade, os usuários necessitam que mensagens fiquem incompreensíveis a terceiros, exceto ao receptor. Assegurar a privacidade envolve geralmente algum protocolo criptográfico. Para fornecer a privacidade, é feita uma cifra (mistura) na mensagem, de modo que a mensagem possa ser armazenada e transmitida com segurança.

A cifragem de dados transforma uma mensagem escrita (texto original) de modo que apareça como uma mistura de caracteres. Um sistema bom de cifragem de dados faz com que seja difícil de reverter os dados cifrados no texto original sem uma chave secreta. Os dados a serem cifrados podem ser texto do ASCII (Amsterdam Subversive Code for Information Interchange), um campo de uma base de dados, ou quaisquer outros dados que você queira armazenar com segurança ou transmitir. O termo mensagem é referente a qualquer informação que deve ser tratada, onde a mensagem original refere-se aos dados que não foram cifrados, e a mensagem cifrada refere-se aos dados que foram cifrados [MIC 03a].

Os dados cifrados podem ser armazenados em um local não seguro ou ser trans-

mitidos sobre uma rede não segura e ainda possuir a característica confidencial. Mais tarde, os dados podem ser decifrados em sua forma original. Este processo é mostrado na ilustração 2.1.

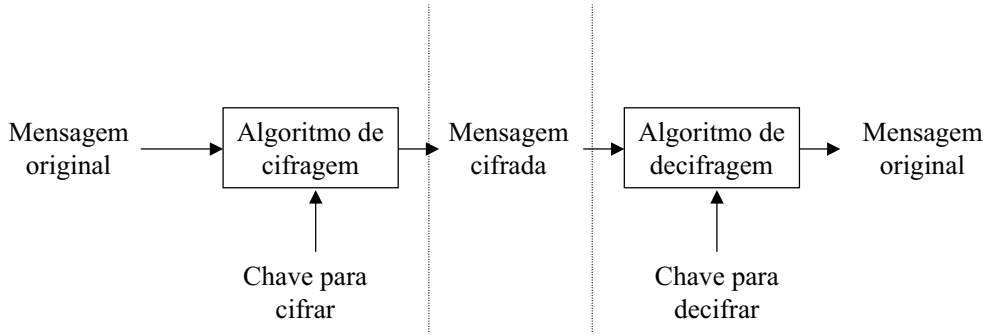

Figura 2.1: Cifrando e decifrando dados

A cifragem de dados e a decifragem são processos simples. Quando os dados são cifrados, uma chave de cifragem é usada. Esta chave é análoga a uma chave física que seja usada para travar um cadeado. Para decifrar os dados, uma chave da decifragem é usada. A chave da decifragem é análoga a usar uma chave para destravar um cadeado.

A cifragem e a decifragem são feitas freqüentemente usando a mesma chave, que neste caso chamaríamos de cifragem simétrica, pois utilizaríamos algoritmos simétricos. Entretanto a cifragem e a decifragem podem usar chaves diferentes, como o par de chaves pública e privada, utilizando neste caso, algoritmos assimétricos.

As chaves sigilosas devem ser mantidas de forma segura, e devem ser transmitidas com segurança a outros usuários. O desafio principal está em restringir corretamente o acesso à chave da decifragem porque qualquer um que a possuir poderá decifrar todas as mensagens que foram cifradas com sua chave correspondente à cifragem.

Autenticação - As comunicações seguras requerem que ao se comunicar os indivíduos saibam a identidade daqueles com quem estão se comunicando. A autenticação envolve verificar a identidade de um uma pessoa ou entidade. Para verificar a identidade de uma pessoa, as carteiras de identidades, CPFs (Cadastro de Pessoa Física) e outros, são usados freqüentemente. Quando uma verificação é feita, a pessoa que está fazendo a verificação confia na entidade que emitiu o documento de identificação.

Os passaportes são outro exemplo. Um oficial olha um passaporte e aceita-o como a prova que uma pessoa é quem diz que ser. O oficial confia que o governo fez um trabalho adequado de identificar o proprietário do passaporte antes de emitir o passaporte. Em ambos os exemplos, um nível de confiança existe na entidade emissora de identificações.

A autenticação envolve também certificar-se de que os dados recebidos são os dados que foram emitidos. Se A emitir uma mensagem a B, B necessita poder provar que a mensagem recebida era a mensagem que A emitiu, e A não pode negar que a enviou.

Desde que as comunicações sobre uma rede de computadores surgiram com nenhum contato físico entre os comunicantes, verificar a identidade depende freqüentemente de uma credencial que possa ser emitida e recebida sobre uma rede. Tal credencial deve ser emitida por uma autoridade de confiança dos usuários.

Os Certificados Digitais são justamente estas credenciais. É uma maneira de verificar a identidade e conseguir a autenticação em uma rede de computadores.

Um certificado digital é uma credencial emitida por uma entidade confiável denominada Autoridade Certificadora (AC). Este certificado contem a chave pública e os dados que identificam uma determinada entidade. Um certificado é emitido por uma AC somente depois que a AC verificou a identidade da entidade que solicitou a emissão do certificado e confirmou que tal entidade tem realmente a posse da chave privada referente àquela chave pública.

A comunicação entre uma AC e um solicitante de certificado é realizada geralmente através da troca de mensagens sobre uma rede.

Integridade - Todos os dados emitidos sobre um meio não seguro, podem ser mudados acidentalmente ou propositalmente. No mundo real, os lacres são usados para fornecer e provar a integridade.

Da mesma maneira, um receptor de dados necessita não somente poder verificar a identidade do remetente dos dados, mas deve também estar certo de que os dados recebidos são exatamente os dados que foram emitidos; que não foram alterados.A integridade dos dados é alcançada através da emissão de não somente os dados originais, mas também uma mensagem de verificação, chamada de resumo, sobre aqueles dados. Os dados e a mensagem da verificação podem ser emitidos como uma assinatura digital que prove a

origem de ambos.

2.2 Criptografia

As chaves criptográficas são o cerne das operações criptográficas. No caso das chaves simétricas, por exemplo, devem ser mantidas secretas, porque quem quer que possua esta chave tem o acesso a todos os dados que a chave está associada. Por exemplo, se uma chave for usada para cifrar um campo, qualquer um com uma cópia dessa chave pode decifrar o campo. Além disso, qualquer um que possui uma chave pode usá-la para assinar mensagens.

Fazendo uma analogia com o mundo real, guardamos um conteúdo muito valioso em um baú e o lacramos com um cadeado e sua chave. A princípio, apenas quem possuir a chave do cadeado, possuirá acesso ao conteúdo do baú. Neste caso, o segredo de como abrir o baú está no formato da chave do cadeado. Nem mesmo o fabricante do cadeado, ou o fabricante do baú poderia abrí-lo teoricamente.

Trazendo esta idéia para a informática, possuímos um algoritmo que pode ser de domínio público, porém teríamos uma senha que seria utilizada pelo algoritmo para misturar os dados. Desta forma, o segredo está na senha, e não no algoritmo. Quem a possuir, possuirá acesso aos dados cifrados por ela.

Estas senhas são chamadas de chaves criptográficas.

Há dois tipos de Criptografias: Criptografia Simétrica e Criptografia Assimétrica.

2.2.1 Criptografia Simétrica

As chaves simétricas são usadas com algoritmos simétricos de cifragem. Os algoritmos simétricos são o tipo mais comum de algoritmos de criptografia. São chamados simétricos porque usam a mesma chave para cifrar e decifrar. As chaves simétricas são mudadas freqüentemente, geralmente usando uma chave diferente para cada mensagem cifrada.

Os algoritmos simétricos são mais rápidos do que algoritmos assimétricos. Assim,

são mais utilizados para cifrar quantidades grandes de dados. Alguns dos algoritmos simétricos mais comuns são o DES [AES 03a], e o AES [AES 03b].

Ao contrário das chaves assimétricas, as chaves simétricas são temporárias. As aplicações podem armazenar estas chaves para seu uso mais tarde ou para a transmissão a outros usuários usando a função de ”Exportação de Chaves”e exportá-los do CSP (*Cryptographic Service Provider*) para o espaço da aplicação[MIC 03a]. Ou seja, para exportar dados, entre instâncias do mesmo aplicativo, a chave simétrica é utilizada para cifrar e decifrar.

2.2.2 Criptografia Assimétrica

Os pares de chaves assimétricos são usados por um método mais seguro de cifragem. A cifragem assimétrica é usada principalmente para cifrar e decifrar chaves simétricas e assinaturas digitais. A cifragem assimétrica usa algoritmos próprios que utilizam duas chaves diferentes: uma chave pública e uma chave privada. A chave privada do par, deve ser mantida em sigilo e segura. A chave pública, entretanto, pode ser distribuída a qualquer um que a pede. A chave pública de um par de chaves é distribuída freqüentemente por meio de um certificado digital. Quando uma chave de um par de chaves é usada para cifrar uma mensagem, a outra chave deste par será requerida para decifrar a mensagem. Assim se a chave pública do usuário A for usada para cifrar dados, somente o usuário A (ou alguém que tem o acesso à chave privada do usuário A) pode decifrar os dados. Se a chave privada do usuário A for usada para cifrar dados, somente a chave pública do usuário decifrará os dados, assim indicando que o usuário A (ou alguém com acesso à chave privada do usuário A) cifrou.

Se a chave privada for usada para assinar uma mensagem, a chave pública desse par deve ser usada para validar a assinatura. Por exemplo, se Alice quisesse emitir a alguém uma mensagem digital assinada, assinaria a mensagem com sua chave privada, e a outra pessoa poderia verificar sua assinatura usando sua chave pública. Já que somente Alice tem o acesso a sua chave privada, o fato de que a assinatura pode ser verificada com chave pública de Alice indica que Alice criou a assinatura.

Infelizmente, os algoritmos assimétricos são muito lentos, aproximadamente 1000 vezes mais lento do que os simétricos [STI 02]. É pouco prático usá-los para cifrar quantidades grandes de dados. Na prática, os algoritmos assimétricos são usados para cifrar chaves simétricas. Os algoritmos simétricos são usados para cifrar/decifrar a maioria dos dados.

Cada usuário tem geralmente um par de chaves assimétricas. Uma chave (pública) é usada para cifrar chaves simétricas e a outra (privada) para criar assinaturas digitais.

Assim, se Alice quisesse enviar dados sigilosos para Bob, deveria requerer a chave pública de Bob. Com esta chave pública de Bob cifraria os dados , e enviaria à Bob o texto cifrado. Desta forma, apenas quem possui a chave privada de Bob acessará os dados sigilosos enviados por Alice.

Alguns exemplos dos algoritmos de criptografia assimétrica são o RSA, Elgamal, etc.

2.2.3 Função Resumo

Um resumo de um texto ou de uma cadeia de bits é um valor associado ao texto original com um tamanho fixo e menor. Existem funções que fornecem meios de criar um resumo para todo o texto ou uma cadeia de bits. Esse resumo, então, pode ser usado como um identificador original de seus dados associados.

Para assegurar a integridade de um texto, um resumo de um texto pode ser emitido juntamente com o texto. O receptor pode então computar o resumo dos dados recebidos e comparar o resumo computado com o resumo recebido. Se realmente a mensagem estiver íntegra, então a comparação dos resumos deve ser igual.

Desta forma, não se pode determinar o texto apenas com a posse de seu resumo. E como era de se esperar, o resumo possui pouca semelhança com o texto, de tal maneira que a qualquer modificação do texto, geraria um resumo completamente diferente.

As funções resumo são comumente utilizadas como um lacre resguardando a integridade da mensagem original, pois qualquer falsificação, ou fraude sobre a mensagem original, é facilmente detectada pela função resumo.

Alguns exemplos dos algoritmos da função resumo são o SHA-1, MD5, etc.

2.2.4 Assinatura Digital

A Assinatura Digital pode ser usada para distribuir uma mensagem no formato de texto plano quando os receptores desejam identificar e verificar o remetente da mensagem. Assinar uma mensagem não altera a mensagem, gera simplesmente uma seqüência de bytes que será a assinatura digital. Uma assinatura digital é uma parte curta de dados que são cifrados com a chave privada do remetente. Decifrar os dados da assinatura utilizando a chave pública do remetente prova que os dados estiveram cifrados pelo remetente ou por alguém que teve o acesso à chave privada do remetente.

As Assinaturas de Digitais são geradas usando algoritmos assimétricos. Uma chave privada cifra e gera a assinatura, e a chave pública correspondente deve ser usada decifrar e validar assim a assinatura. Este processo é mostrado na ilustração 2.2.

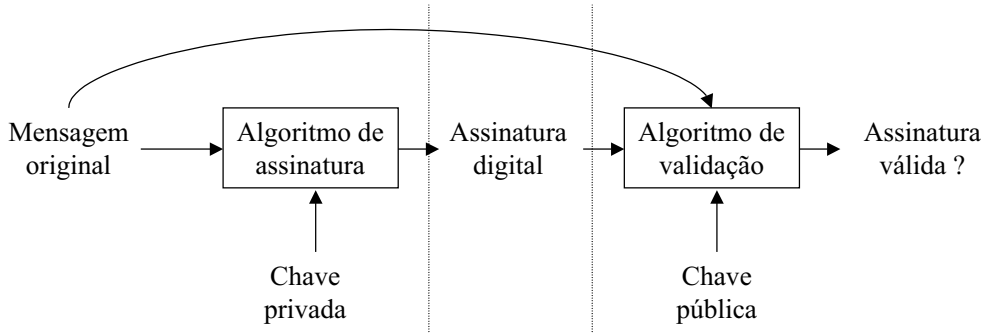

Figura 2.2: Assinando e verificando uma mensagem

Há duas etapas envolvidas em criar uma assinatura digital de uma mensagem. A primeira etapa envolve criar um resumo da mensagem. Este resumo é cifrado usando a chave privada do assinante. A ilustração a seguir é uma das etapas envolvidas em criar uma assinatura digital.

Para verificar uma assinatura, a mensagem e a assinatura são requeridas. Primeiramente, um resumo deve ser criado da mensagem da mesma maneira que a assinatura foi criada. Este valor do resumo é comparado então com a assinatura decifrada com a chave pública do assinante. Se o resumo gerado e a assinatura decifrada combinarem, você pode

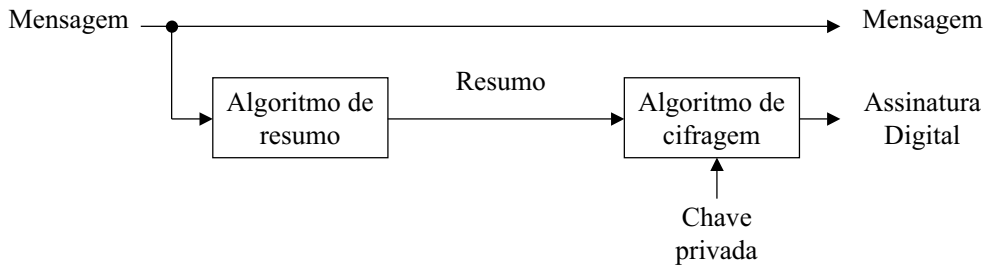

Figura 2.3: Assinando uma mensagem

afirmar que a mensagem é certamente do assinante e que esta não foi alterada. O seguinte diagrama ilustra o processo envolvido em verificar uma assinatura digital.

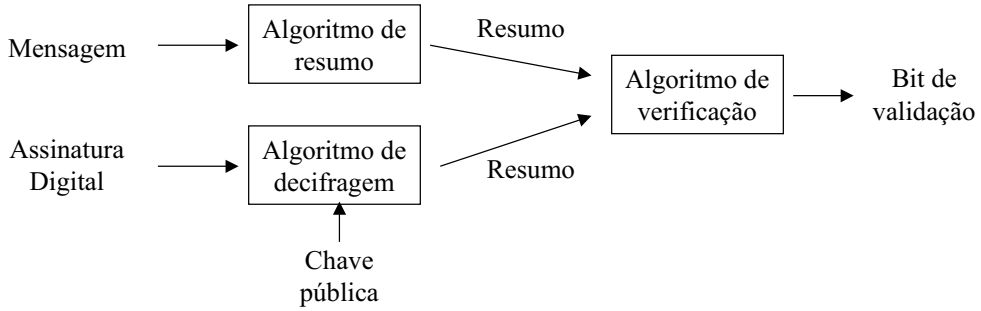

Figura 2.4: Verificando a assinatura

Um resumo consiste em um pouco de dados binários, tipicamente cerca de 160 bits. Isto é produzido usando um algoritmo de resumo.

O tamanho do resumo é determinado pelo tipo de algoritmo usado, e seu comprimento não varia com o tamanho da mensagem. Os comprimentos mais comuns são 128 ou 160 bits.

Cada parte da mensagem traduz um valor completamente diferente de resumo, mesmo se as duas partes da mensagem diferem somente por um único bit. Usando a tecnologia de hoje, é inviável descobrir uma parte da mensagem que traduza o valor do resumo sem quebrar o algoritmo de resumo.

Todos os algoritmos de resumo possuem um propósito único. Dado um resumo, não é possível recuperar a mensagem original. De fato, nenhuma das partes da mensagem original pode ser determinada a partir do resumo.

No Brasil, a empresa Bry Tecnologia produziu um software que faz assinaturas digitais.

2.2.5 Infra-estrutura de Chaves Públicas

Uma Infra-estrutura de Chaves Públicas (ICP) é um conjunto de padrões cujo objetivo é fornecer suporte às aplicações de manipulação de chaves públicas. Ou seja, para haver troca de informações confiáveis ou sigilosas entre partes distintas, se faz necessário a utilização de serviços, protocolos, e padrões [dB 03].

2.2.5.1 Certificados Digitais

A autenticação é a base de toda a comunicação que requer confiabilidade. Autenticidade é quando um usuário pode provar sua identidade e verificar a identidade com quem se comunica. Para a comunicação segura, a autenticidade é muito importante, pois na comunicação via rede, como não há o encontro físico, pode haver interceptação da mensagem e alteração da mesma.

O certificado digital é uma credencial que fornece meios de verificar a identidade. Uma organização de confiança atribui um certificado a uma pessoa física ou a uma organização que então associa uma chave pública a esta entidade. A organização confiada que emite o certificado é uma Autoridade Certificadora (AC) que será melhor abordado no ítem 2.2.5.2.

Os certificados usam técnicas criptográficas para sanar o problema da falta do contato físico entre entidades que comunicam-se. Usar estas técnicas limita a possibilidade de um terceiro interceptar para alterar, ou falsificar mensagens.

Os dados em um certificado incluem a chave criptográfica pública do par de chaves. Uma mensagem assinada com chave privada pelo seu remetente pode somente ser lida pelo receptor da mensagem usando a chave pública do remetente. Decifrar uma assinatura com uma chave pública de um certificado prova que a assinatura foi produzida usando a chave privada do certificado. Se o remetente for cuidadoso e mantiver o segredo chave privada, o receptor pode confiar na identidade do remetente da mensagem.

Os certificados de digitais incluem também as informações adicionais do proprietário do certificado tais como o endereço, E-mail, as atividades que o certificado pode executar, entre outras [MIC 03b].

2.2.5.2 Autoridade Certificadora

Naturalmente deve existir confiança entre o receptor de uma mensagem assinada e do emissor que assinou a mensagem. como já foi visto, um método de estabelecer esta confiança é através de um certificado, um documento eletrônico que confirma que entidades ou pessoas são quem dizem ser. Este certificado é emitido por um terceiro que seja de confiança de ambas as partes. Então cada receptor da mensagem decide se deve confiar nesta entidade que emitiu o certificado do remetente [MIC 03c].

A esta terceira entidade chamamos de Autoridade Certificadora (AC). AC é uma entidade de confiança que certifica-se da identidade de uma pessoa física ou jurídica e à esta emite um certificado digital. Também é de responsabilidade de uma AC a revogação de certificados, listando-os em uma Lista de Certificados Revogados (LCR).

O processo de emissão de certificados se dá primeiro com uma requisição de uma entidade. Assim, é realizada uma série de procedimentos, normas e padrões que comprovam a identidade do requisitante. Com base nestas informações, é gerado o certificado, contendo informações do requisitante, propósitos do certificado, a assinatura da AC que emitiu o certificado, uma data de validade, a chave pública e outras propriedades. Este certificado é enviado a um diretório público, e outra cópia ao requisitante.

De posse de um certificado digital é possível assinar mensagens, códigos, documentos, etc, as quais se poderá confiar no autor que foi certificado pela AC.

2.2.5.3 Autoridade de Registro

A Autoridade de Registro (AR) é uma organização na qual a Autoridade Certificadora confia trabalho de conferir a identidade de um requisitante que deseja um certificado. A qualidade do processo de conferência das informações determina o nível de confiança que deve ser atribuído ao certificado [HUN 00]. A conferência das informações

compreende uma série de procedimentos, normas e padrões pré - estabelecidos que comprovam a identidade do requisitante [MIC 03c].

2.2.5.4 Diretório Público

Os Diretórios Públicos são entidades que armazenam os certificados contendo as chaves públicas e LCR tornando-os acessíveis a qualquer usuário que requere estes dados.

Desta forma, um diretório público não fornece qualquer tipo de segurança com relação aos dados nele armazenado. Porém estes diretórios públicos devem possuir os dados sempre atualizados, e com um nível de disponibilidade alto.

2.2.6 Compartilhamento de Segredos

Esta técnica baseia-se na divisão de um segredo pelo número de entidades confiáveis. Cada entidade receberá portanto, uma ”parte”do segredo. Quando houver a necessidade do uso do segredo, será necessário reconstituí-lo, e para isto, alguma das entidades devem fornecer a sua parte que lhes foi entregue.

O propósito desta técnica é certificar-se de que a utilização de um certo dado seja utilizado somente quando há a concordância de um número mínimo de entidades que participaram do compartilhamento.

Um exemplo de esquemas de compartilhamento de segredos é área militar, quando se deseja que uma tomada de decisão importante tenha a aprovação de um número mínimo de membros da organização [HAR 99].

2.2.6.1 Divisão do Segredo

De acordo com o *esquema de divisão de segredos* definido em [SCH 96], a partir do número de participantes da divisão *n*, são criadas *n* - 1 *strings* com mesmo tamanho do segredo identificadas por *Xi*. Com o segredo e as identificações, são feitas um *XOR* (OU exclusivo) e obtem-se *y*.

Cada identificador *Xi* é entregado a cada participante, sendo que *y* é entregue ao último participante.

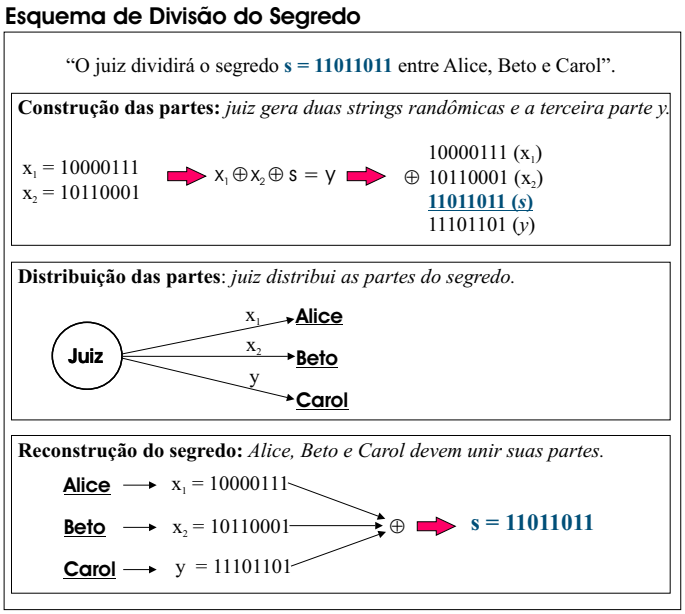

Figura 2.5: Exemplo do Protocolo de Divisão do Segredo

Desta forma, para recuperar-se o segredo é necessário que todos os participantes forneçam as partes que lhe foi confiada, e através de um *XOR* entre estas partes (o *y* está incluso) obtém-se o segredo.

2.2.6.2 Esquema de Limiar do Shamir

Este esquema é um sistema de divisão de segredos criados por George R. Blakley e Adi Shamir [SHA 79], o qual é definido um número de participantes (inferior ao número de total de participantes), chamado de valor de limiar, que poderá recuperar o segredo.

Desta forma, estipula-se com antecedência o número total de participantes *n* e o valor de limiar *t*, onde *s* é o segredo.

Uma entidade de confiança escolhe aleatoriamente um número primo *t* e constrói um polinômio de grau *t*-1 cujos coeficientes são determinados randomicamente com exceção do primeiro coeficiente que será *s*. Cada uma das partes geradas será entregue aos participantes.

Para a reconstrução do segredo será necessária as partes de no mínimo *t* participantes. Com estas partes, será utilizada a interpolação de Lagrange, e através desta, será

recuperado o segredo.

2.2.7 Redes de Misturadores

As Redes de Misturadores é uma técnica baseada na criptografia de chaves públicas que permite ocultar a identidade de um usuário bem como a própria comunicação. Esta técnica não requer uma autoridade universal de confiança. Um usuário pode permanecer anônimo a um outro usuário, ao permitir que este responda através de um endereço do retorno que não pode ser rastreado [CHA 81].

Esta técnica onde uma entidade recebe uma série de mensagens eletrônicas com um tamanho fixo, para torná-las irreconhecíveis quanto ao tamanho, e assinadas com a chave pública da entidade (para que somente a entidade misturadora possa decifrá-las).

Esta entidade misturadora, após receber as mensagens, elimina a identificação do remetente e mistura a ordem de chegada das mensagens. Estas são enviadas aos seus respectivos destinatários.

2.2.8 Conclusão

Neste capítulo conceituamos uma série de aspectos da criptografia, cerne deste trabalho, com o intuito de esplanar as principais técnicas que serão utilizadas para garantir a eficácia do sistema que será desenvolvido.

Capítulo 3

Tecnologias Utilizadas

3.1 Introdução

Este capítulo destina-se a um estudo detalhado sobre as ferramentas e tecnologias que utilizamos para o desenvolvimento de nosso projeto, incluindo linguagens de programação; ferramentas e aplicativos; servidores Web e sistema de gerenciamento de banco de dados. Fizemos um estudo detalhado sobre cada uma das tecnologias e ferramentas empregadas procurando encontrar no histórico, vantagens e desvantagens, comparativos com outras tecnologias da mesma linha de funcionamento e o estado da arte, que representa o atual estado em que encontra-se cada tecnologia.

3.2 CryptoAPI

API (*Application Programming Interface* - Programação de interfaces de aplicativos) consiste em um grupo especial de DLLs (*Dynamic-link library* - biblioteca dinâmica) chamadas de *Standard DLLs*. As *Standard DLLs* são diferentes da interface COM (*Component Object Model* - modelo de interação com objetos), por isso deve-se instanciá-las diretamente através de uma chamada de função [MIC 99]. Mais adiante na seção 3.8.2 teremos a definição de DLL e na seção 3.8.3 o conceito de API.

A CryptoAPI é uma *Standard DLL* com métodos e propriedades relacionadas com

criptografia. O sistema de arquitetura da CryptoAPI é composta por cinco áreas funcionais. Estas funções básicas de criptografia são usadas para conectar com um CSP, permitindo assim, que aplicações escolham um CSP específico que forneça uma classe de funcionalidades necessárias para a aplicação.

- Funções de geração de chaves, utilizadas para gerar e armazenar chaves criptográficas;
- Funções de trocas de chaves utilizadas para trocar ou transmitir chaves;
- Funções de codificação e decodificação com certificados utilizados para cifrar ou decifrar dados;
- Funções de armazenamento de certificados utilizados para o gerenciamento de conjuntos de certificados digitais;
- Funções simples de manipulação de mensagens para cifrar e decifrar mensagens ou dados.

  [MIC 03d]

As aplicações podem utilizar qualquer destas funções, e todas estas funções formam a CryptoAPI. A base das funções de criptografia usam os CSPs que são necessários aos algoritmos de criptografia e o armazenamento seguro de chaves.

Embora uma aplicação possa se comunicar diretamente com qualquer uma das cinco áreas de funções, ela não pode se comunicar diretamente com um CSP. Toda e qualquer comunicação entre uma aplicação e um CSP ocorre através de uma base de funções criptográficas. Com estas funções, passa-se apenas o parâmetro que especifica qual CSP que será utilizado. Se este parâmetro for um valor nulo, o CSP selecionado será o CSP padrão [MIC 03d].

3.3 Capicom

O Capicom é um modelo de interação com objetos (COM - Component Object Model) que interage com a biblioteca CryptoAPI. A vantagem de se utilizar um COM em aplicativos é a facilidade com que estas DLLs são manipuladas pelo desenvolvedor.

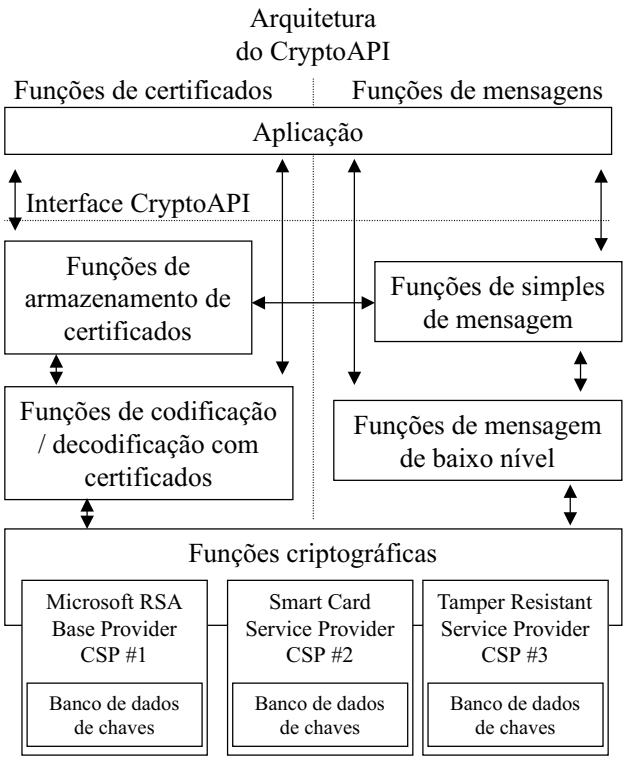

Figura 3.1: Arquitetura do CryptoAPI

A CryptoAPI e a CAPICOM fornece serviços que permitem os desenvolvedores de aplicativos adicionarem funções de cifragem/decifragem de dados e autenticação utilizando certificados digitais. Este é o método utilizado para especificar objetos abstratos que são esperados pela transmissão serial - base dos aplicativos Microsoft Windows.

Através do Capicom os desenvolvedores de aplicativos podem utilizar as funções da CryptoAPI sem conhecer os detalhes subjacentes implementados, além de se poder utilizar bibliotecas gráficas sem conhecer qualquer particularidade da configuração gráfica do hardware. A CryptoAPI trabalha com um número de CSP que realizam as reais funções criptográficas [MIC 03a].

As operações básicas do Capicom, como criação de assinatura digital, e decodificação de uma mensagem codificada, estão prontas para serem utilizadas no computador de qualquer usuário, devendo apenas este possuir um certificado digital válido que contenha uma chave privada associada. Se não houver um certificado com a chave privada associada, então qualquer operação criptográfica citada não funcionará. Os usuários de aplicativos

com o Capicom devem se certificar que possuem certificados válidos quando os aplicativos executarem [MIC 03e].

Devido à uma série de peculiaridades próprias do Capicom, o que o Capicom cifra, somente o Capicom decifra [MIC 03f].

3.4 Servidor Web

Geralmente um aplicativo Web envolve de alguma maneira a obtenção de dados de um usuário, esta obtenção de dados normalmente ocorre com um formulário HTML. O usuário preenche um formulário qualquer, o navegador processa estes dados e envia uma requisição para o servidor Web.

O servidor Web é acionado, ou seja, executa um programa servidor, quando um navegador empacota determinadas informações e as envia através de uma requisição HTTP ou HTTPs. Esta requisição é basicamente o endereço da página que o usuário quer acessar ou o script, os dados do formulário, e outras informações contidas no cabeçalho.

Estas requisições devem especificar o seu método de acesso, normalmente os métodos podem ser POST, GET ou HEAD. O método HEAD obtém informações de um documento e nunca seu conteúdo, o método GET é usado para emitir requisições para executar um programa Web, esta requisição é enviada, através do adicionamento de uma *string* no formato chave=valor à *string* de requisição (ex.: http://www.endereco.com?-acao=valoracao). O método POST tem o mesmo propósito do método GET porém a requisição é enviada através de valores dentro do arquivo de entrada, permitindo assim maior clareza na requisição quando uma grande quantidade de dados é enviada.

O servidor Web tem uma importante função que é a de passar a requisição do usuário a um script ou programa para que esta possa então ser processada. O mapeamento das requisições para os programas ou scripts necessários é feita analisando a extensão do arquivo enviado ou definindo-se uma pasta no servidor onde qualquer arquivo dentro desta deve ser processado por um determinado programa. Após processar os dados com o programa ou script correspondente à requisição o servidor Web retorna uma resposta ao navegador, o servidor basicamente especifica o tipo de conteúdo e grava a resposta

em um fluxo de saída. Após receber esta resposta o navegador analisa o cabeçalho da resposta para que possa então definir de que maneira irá mostrar os dados para o usuário que enviou a requisição, por exemplo, um áudio, imagem ou texto html.

3.4.1 Servidor IIS (Internet Information Service)

O IIS é o servidor Web desenvolvido pela Microsoft e executado sobre a plataforma Windows. O IIS executará um serviço no Windows NT/2000 que o transformará em um servidor Web. Após a instalação, as páginas desejadas devem ser colocadas dentro das pastas de um servidor ou mais servidores e o IIS deve ser configurado para acessar estas páginas quando solicitado [GS 99].

Para disponibilizar o acesso à uma pasta virtual através da Internet, como por exemplo http://meuservidor.com/teste e considerando uma pasta que se encontra no diretório c:\inetpud\wwwroot\teste, devem ser seguidos os seguintes passos:

- Ativar o Internet Services Manager;
- Clicar com o botão da direita em Default Web Site;
- Clicar em ”new”e depois em ”virtual directory”;
- Em seguida aparecerá uma nova janela (virtual directory create wizard) onde deve-se clicar no botão next;
- Na tela Virtual Directory Alias, deve-se colocar o nome fácil para acessar a pasta virtual, no nosso caso teste;
- Na próxima tela (Web Site Content Directory) deve-se especificar o caminho da pasta onde se encontra a pasta a ser publicada neste caso c:\inetpud\wwwroot\teste;
- Na tela Access Permissions, devem ser definidas as permissões que serão atribuídas aos usuários que acessam esta página. As opções são:
  - read - permite que os usuários leiam a página HTML e a executem;

  - run scripts (such as ASP) - permite que os usuários executem páginas com scripts com ASP;
  - execute (such as ISAPI applications or CGI) - permite que usuários rodem os scripts baseados nos formatos ISAPI (Internet Server Advanced Programming Interface), CGI (Common Gateway Interface) ou outros modelos parecidos;
  - write - permite que os usuários alterem o conteúdo da página sobre a qual eles possuem esta permissão;
  - browse - permite que os usuários vejam os arquivos contidos na pasta onde se encontra a página.

- Na tela Virtual Directory Create Wizard clicar no botão Finish para encerrar a criação do diretório.

Seguindo os passos acima temos no nosso site uma nova pasta acessível a partir do endereço http://meuservidor.com/teste. Para alterar alguma propriedade da nova pasta virtual, clica-se com o botão direito em cima da pasta e escolhe a opção properties, que viabiliza uma janela com as seguintes guias:

- **General** : especifica o local da pasta virtual, as permissões existentes, o nome da aplicação e outros;
- **Documents** : determina os arquivos que devem ser procurados para ativar a página sem que seja citado o nome de um arquivo;
- **Directory Security** : permite que se determine o método de acesso e autenticação dos usuários, permite negar ou conceder acesso baseando-se nos endereços IP dos usuários e estabelecer o uso do protocolo SSL (Secure Socket Layer) ou o HTTPs, viabilizando um canal seguro para troca de informações entre cliente e servidor, via chaves de criptografia de dados pública e privada;
- **HTTP Headers** : permite estipular de quanto em quanto tempo a página deve ser atualizada pelo IIS, pois acredita-se que a página sofreu alguma modificação;

- **Custom Errors** : permite que se padronizem os erros de acordo com as necessidades da empresa ou entidade.

3.4.2 Apache

O servidor Web Apache é resultado de um projeto cujos objetivos eram criar um servidor web robusto, com capacidade comercial, com diversas funcionalidades e com código fonte aberto e livre, este projeto foi desenvolvido em equipe [MON 03]. Atualmente este projeto é gerenciado por um grupo de voluntários ao redor do mundo, e que utilizam a internet para planejar e desenvolver o servidor bem como a sua documentação. Diferente do IIS, o Apache funciona em diversas plataformas, entre elas Linux e Microsoft Windows.

Em fevereiro de 1995 o servidor mais popular na Web era o de domínio público, HTTP daemon, que foi desenvolvido por Rob McCool no NCSA (National Center for Supercomputing Applications) da Universidade de Illinois. Após ele ter deixado o NCSA, vários programadores contribuíram para o melhoramento do seu projeto, acrescentando funcionalidades e corrigindo erros. Um pequeno grupo desses programadores se organizou e começaram a fazer os trabalhos de maneira coordenada em uma máquina com login e senha na Califórnia. Oito membros desse grupo inicial fundaram o ”Apache Group”(Grupo Apache) que é um grupo destinado a gerenciar os trabalhos que são desenvolvidos por programadores ao redor do mundo todo, este grupo era originalmente composto por Brian Behlendorf, Roy T. Fielding, Rob Hartill, David Robinson, David Robinson, David Robinson, Cliff Skolnick, Randy Terbush, Robert S. Thau e Andrew Wilson. Em 1999, os membros do Grupo Apache formaram o Apache Software Foundation que é uma organização sem fins lucrativos que organiza o Servidor Web Apache.

3.5 SSL - Secure Socket Layer

O SSL é um protocolo de segurança desenvolvido pela Netscape que provê serviços de privacidade na Internet e consiste em duas fases, autenticação do servidor uma autenticação

do cliente opcional [HIC 95] que serão explicadas mais adiante. Ele é utilizado para gerenciar a segurança de mensagens transmitidas pela rede. A idéia da Netscape foi a de que a programação para se ter mensagens confidenciais transitando pela rede deveria estar no nível de aplicação da camada OSI do TCP/IP. O termo socket se refere ao método socket de transmitir dados de um lado para o outro entre um cliente e um servidor em uma rede. O SSL da Netscape utiliza o algoritmo RSA em conjunto com certificados digitais. O SSL é integrado aos navegadores Netscape e Internet Explorer. Sua utilização depende da existência do SSL no servidor Web assim como da solicitação deste serviço pelo sistema que opera no cliente.

A principal vantagem do protocolo SSL é que ele é independente de aplicação, um protocolo do nível de aplicação como HTTP, FTP ou *telnet* podem ficar em uma camada acima do SSL O protocolo SSL pode negociar o algoritmo de cifragem e a chave de sessão assim como autenticar o servidor antes que o protocolo do nível de aplicação transmita ou receba algum dado. Todos os dados são cifrados antes de serem transmitidos o que garante a privacidade.

O objetivo principal do SSL é propiciar confiabilidade e privacidade às comunicações realizadas através da Internet. O SSL possui três propriedades básicas:

- O processo de cifragem é utilizado depois do contato inicial para definir uma chave secreta. A criptografia simétrica é usada pra cifrar os dados. Isso torna a conexão privada;

- O estabelecimento da chave simétrica que será utilizada sessão ocorre no início das comunicações e é feita através de criptografia assimétrica;

- A transmissão de mensagens inclui a checagem de integridade de mensagens usando um protocolo MAC (Controle de Acesso ao Meio) protegido por chave, o que torna a conexão confiável.

O SSL é instalado no servidor Web, e funciona basicamente da seguinte maneira, o servidor Web faz uma solicitação de um certificado para uma entidade certificadora, como o LabSEC por exemplo, este por sua vez processa a requisição e envia o certificado

emitido ao servidor Web. Uma vez com o certificado o servidor o instala e escolhe quais pastas virtuais ou quais sites irão utilizar o protocolo SSL para garantir a segurança, lembrando que não se deve utilizar SSL para qualquer site, apenas os que a segurança for imprescindível já que o SSL retarda a velocidade de execução de requisição de conteúdo HTML.

Exemplo de requisição de um certificado:

```
-----BEGIN CERTIFICATE REQUEST-----
MIIB3zCCAUgCAQAwgZ4xCzAJBgNVBAYTAlVTMRMwEQYDVQQIEwpDYWxpZm9ybmlh
MQ4wDAYDVQQHEwVEYXZpczEoMCYGA1UEChMfVW5pdmVyc2l0eSBvZiBDYWxpZm9y
bmlhLCBEYXZpczElMCMGA1UECxMcSW50ZXJuc2hpcCBhbmQgQ2FyZWVyIENlbnRl
cjEZMBcGA1UEAxMQaWNjMi51Y2RhdmlzLmVkdTCBnzANBgkqhkiG9w0BAQEFAAOB
jQAwgYkCgYEAp6X6vQqiBTL4IHTi7DlScrtK6sctGP5PjkKq9NGTRaQut9rOexli
FweYt5q9Ix7Cje7hW4B/0ABPelXV4twmnoAcgGL5vK8N+ZEWFG/OqdiZNnnSQ9XP
TWd/tM/5TWwDuftllx8oML+YF2ugKdNY55Jevu7W1rGUoiOk/JjT7+UCAwEAAaAA
MA0GCSqGSIb3DQEBBAUAA4GBAJTffqs61+u4/0lISTuB6dm4E3sK68oORmGiTLkU
uAmxxTZc9UFAyJUlgowthbYOxsG5UpAHbsGh9l15CJnrxEPj4Z38YMJ4ILUkzHJl
/Tx/cTsYRL6DiTRo4QjS0q0v4Y8pequQl3A2p4eJr0mthB7rNnLH0vgv6AHB0OGV
ZOpP
-----END CERTIFICATE REQUEST-----
```

Este certificado foi solicitado por um servidor Web a uma entidade certificadora qualquer, neste caso, ao LabSEC.

Exemplo de resposta para a requisição de um certificado:

```
 -----BEGIN CERTIFICATE-----
MIIDEzCCAfugAwIBAgIBCjANBgkqhkiG9w0BAQQFADB/MQswCQYDVQQGEwJVUzEw
MC4GA1UEChMnTmF0aW9uYWwgQ29tcHV0YXRpb25hbCBTY2llbmNlIEFsbGlhbmNl
MSAwHgYDVQQLExdDZXJ0aWZpY2F0aW9uIEF1dGhvcml0eTEcMBoGA1UEAxMTQ2Vy
dGlmaWNhdGUgTWFuYWdlcjAeFw0wMDAxMTMxNzI4NTNaFw0wMjAxMTIxNzI4NTNa
MFcxCzAJBgNVBAYTAlVTMTAwLgYDVQQKEydOYXRpb25hbCBDb21wdXRhdGlvbmFs
```

IFNjaWVuY2UgQWxsaWFuY2UxFjAUBgNVBAMTDVNjb3R0IEtvcmFuZGEwgZ8wDQYJ
KoZIhvcNAQEBBQADgY0AMIGJAoGBANMVxxyXIXF5Xt1kLwhnk1R2m8M8UUbj3CQY
NDyo6i7ojVYusxGZFD7qye+vE18g7rt7jolz8ifJvOucWGliYTncLcxZdkv7PLib
V02ThoQWC9Ahw15PCJEl0ZkJmYuMZ+/PVdGvOUvtj233C0B86sEPnrI7RrnZorpB
mN2eHJhBAgMBAAGjRjBEMBEGCWCGSAGG+EIBAQQEAwIGwDAOBgNVHQ8BAf8EBAMC
BPAwHwYDVR0jBBgwFoAUZNPfeYI3ff8hfdGUOHSu+CotEkgwDQYJKoZIhvcNAQEE
BQADggEBABY2Np2qZruxhGzewQOTfWiUb09frwUsLFIqym2C1amV9iu6vw5k3N8i
u8x4+JX/KIoHFkYoBDVN42Vr3MkxtsHZf/QrkNuXkS4eny2tjZABkiK6AcQbQOe3
k9RSoFNTNt+AIYmLf7kXJv/VHlwFlb3H9eMqRcckTg7vVc0yHeBLzQhe1ZRTxcBz
FqKNTWNlmho4FFxRJ7wySoCxi4JNDfufd++tsrXfgv+6kOzk/sBLMDn/VsOb8MTc
fGbPtF596PLIOhB0u6RC+tkM9VSznrXfnqb9iMtFZvYqkX3kaNLL2lUEE9vjDscR
SSZ/8F6yOxSG+I2ipPaXNxKtXWgNikQ=
-----END CERTIFICATE-----

Esta foi a resposta ao pedido de certificado emitida pela entidade certificadora, este é o arquivo do certificado e deve ser salvo com a extensão .cer para que possa ser instalado no servidor Web que enviou a requisição.

3.6 Scripts da Web (Cliente)

A World Wide Web começou como uma mídia de texto, a primeira versão da especificação de HTML nem mesmo tinha a capacidade de incluir imagens gráficas em uma página. Porém nos últimos anos o uso da Web cresceu de maneira exponencial, e embora ainda não esteja pronta para competir com a televisão, hoje é uma das mais ativas áreas da Internet.

Em 1991, Tim Beners-Lee introduziu o Hypertext Transfer Protocol (Protocolo de Transferência de Hipertexto) ou http. Hoje este protocolo é a vase da World Wide Web. Porém nesta época a Web somente comportava texto. Em 1994 a World Wide Web mudou drasticamente com a introdução do navegador Web Mosaic que possibilitava a inclusão de imagens no documento. Esta pequena mudança fez a Web crescer repentinamente e

atingir o seu patamar atual onde documentos HTML podem incluir uma variedade de recursos: imagens gráficas, sons, animação, vídeo e, texto.

3.6.1 Introdução

Scripts de páginas Web são pequenos programas interpretados, ou seja, não são compilados, cada instrução é lida, analizada e executada em separado seguindo uma certa ordem, pelo navegador que está sendo utilizado pelo cliente. Estes programas são incorporados no HTML de uma página Web.

Os scripts são sempre executados no lado cliente, diminuindo assim a carga do servidor, repassando parte do processamento para o cliente. Por rodar no lado do cliente, estes scripts (através de controles ActiveX) podem acessar dados específicos do cliente, como por exemplo, no projeto que desenvolvemos, os certificados digitais instalados no computador do cliente.

Uma limitação dos scripts web á o fato que seu código-fonte está sempre disponível a qualquer pessoa que baixe o conteúdo da página. Outra limitação é que sua interação só acontece com o navegador, embora isso possa ser contornado com a utilização de controles ActiveX.

Algumas funcionalidades dos scripts de páginas Web:

- Adicionar mensagens que rolam na tela ou alterar as mensagens na linha de status do navegador;
- Validar os conteúdos de um formulário e fazer cálculos. (Por exemplo, um formulário de pedido pode exibir automaticamente uma soma total à medida que você insere quantidades de item);
- Exibir mensagens para o usuário, tanto como parte de uma página da Web como em caixas de alertas;
- Fazer animações de imagens ou criar imagens que mudam quando o usuário move o mouse sobre elas;

- Detectar o navegador em utilização e exibir conteúdo diferente para navegadores diferentes;

- Detectar plug-ins instalados e notificar o usuário se um plug-in foi exigido.

3.6.2 Java Script

O JavaScript foi a primeira linguagem de scripts de páginas Web. Foi desenvolvido no início de 1996 pela Netscape Communications Corporation para possibilitar interatividade em um documento Web. A Netscape utilizou a sintaxe e o paradigma de orientação a objetos da linguagem de programação já existente, Java para criar uma linguagem de programação análoga para ser utilizada em documentos HTML. Essa linguagem, que deveria ser fácil de aprender e usar, voltou-se para aqueles que já dominavam o HTML, mas não se sentiam confiantes o suficiente para enfrentar Java [MON 99].

3.6.3 JScript

Pouco tempo depois de a Netscape ter desenvolvido o JavaScript, a Microsoft ”copiou”o JavaScript, tanto a sintaxe quanto a estrutura, mas como a Netscape já tinha direitos autorais sobre o nome e implementação JavaScript, a Microsoft ”batizou”este novo script de JScript, assim qualquer script que era desenvolvido em JavaScript para o navegador Netscape poderia também ser executado no Internet Explorer [MON 99].

3.6.4 VBScript

Um pouco depois de introduzido o JScript a Microsoft introduziu o VBScript (Visual Basic Scripting Edition), que é baseado no produto Microsoft, Visual Basic. O VBScript usa uma boa parte da sintaxe e estrutura do Visual Basic, o que facilita que programadores Visual Basic escrevam scripts em VBScript [MCD 98].

Embora que muito parecido com o Visual Basic, o VBScript é muito mais limitado para evitar que usuários executem sem saber scripts maliciosos. Por tal razão não é

possível criar, alterar ou excluir arquivos utilizando este. Uma limitação do VBScript é que ele é suportado apenas no Internet Explorer.

3.7 Tecnologias do Lado do Servidor

Até alguns anos, a única solução real para se ter dados dinâmicos da Web era com CGI, CGI significa Common Gateway Interface, ou seja Interface de Interconexão Comum, possibilita que se adicione elementos de interatividade a suas páginas HTML. Tanto a Netscape quanto a Microsoft desenvolveram APIs para serem executadas no lado do servidor. São elas ASP (Microsoft) e JSP (Netscape). Porém a solução avaliada como melhor solução para este trabalho foi o PHP que funciona de maneira semelhante ao ASP e JSP, os scripts são embutidos em instruções <?php ?> e incorporados dentro de uma página HTML. Estes scripts são executados no servidor antes que a página seja enviada ao navegador, de modo que não há problema de suporte de navegador para as páginas PHP. Diferente do ASP, o PHP é independente de plataforma, gratuito e com código-fonte aberto.

O processamento e geração de páginas Web do lado do servidor oferecem várias vantagens com relação às tecnologias de processamento do lado do cliente, que são: diminui o tráfego da rede, porque não precisa que o servidor e o navegador conversem com freqüência; diminui o tempo de carregamento, porque no final só se faz download de uma página HTML; evita problemas de compatibilidade de navegadores; Pode fornecer dados do cliente que não estão no cliente. Porém, como estes scripts são executados no lado do servidor Web, eles podem aumentar a carga do servidor, fazendo o site ficar lento. Além disso, eles não possibilitam uma interatividade rápida e contínua com o usuário.

3.7.1 PHP

O PHP Hypertext Preprocessor (PHP) é uma linguagem de elaboração de scripts embutida que opera do lado do servidor. Isso significa que ela funciona dentro de um documento HTML para dar a ele a capacidade de gerar instruções específicas. Você pode

transformar seu site em um aplicativo Web e não apenas em uma coleção de páginas estáticas com informações que não podem ser atualizadas dinamicamente e com freqüência, o que é inaceitável em um site comercial ou de educação.

A vantagem do PHP sobre as outras opções como ASP, Cold Fusion, Perl, Java, Python, etc. é a simplicidade, a forma quase natural de usar bancos de dados (suporta diversos bancos de dados), a independência de plataforma e a velocidade. Além de ter código-fonte aberto.

3.7.1.1 Histórico

Em 1994 quando Rasmus Lerdorf juntou alguns scripts Perl para monitorar quem estava espiando seu resumo. Pouco a pouco as pessoas foram ficando interessadas nos scripts, que foram mais tarde lançados em um pacote de ferramentas ”Personal Home Page”. Por causa do grande interesse, em 1995, ele escreveu um sistema de processamento de scripts e incorporou outra ferramenta para analisar a entrada vinda de formulários HTML: FI, Form Interpreter (interpretador de formulários).

As pessoas começaram a usar essas ferramentas para fazer coisas mais complicadas e o desenvolvimento passou de uma só pessoa para um grupo de desenvolvedores encarregados do projeto e sua organização. Esse foi o começo da linguagem PHP3. Esse grupo de desenvolvedores (Rasmus Lerdorf, Andi Gutmans, Zeev Suraski, Stig Bakken, Shane Caraveo e Jim Winstead) aperfeiçoaram e ampliaram o sistema de processamento de scripts e adicionaram uma API simples que dava a liberdade a outros programadores de adicionar mais funcionalidades à linguagem, escrevendo módulos para ela. Foi melhorada também, a sintaxe da linguagem, as estruturas ficaram parecidas com a de linguagem orientadas a objetos ou a procedimentos já conhecidas como C, C++ ou Java o que torna o PHP mais familiar para os programadores [JC 01].

3.7.1.2 Estado da Arte

O PHP4 é baseado no sistema de processamento de scripts Zend. Este sistema de processamento de scripts foi projetado desde o zero para poder ser facilmente incorporado

em aplicativos diferentes.

O PHP4 é o primeiro aplicativo usando o sistema de processamento Zend. O PHP4 já está disponível em http://www.php.net/ . O PHP4 deveria ser completamente compatível com o PHP3, tendo apenas funcionalidades adicionais, porém na prática percebe-se que diversos programas escritos em PHP3 não funcionam com PHP4.

3.7.1.3 Exemplos e Sintaxe

Existem quatro maneiras diferentes de se adiconar código PHP em um documento HTML[MAN 03]:

1. <?php echo(”Primeira”); ?>
2. <? echo (”Segunda”); ?>
3. <script language=”php”> echo (”Terceira”); </script>
4. <% echo (”Quarta”); %>

Sendo que o primeiro código é o que deve ser utilizado pois funciona em qualquer tipo de documento que dá suporte a PHP.

A sintaxe do código PHP é muito parecida com a de linguagens já existentes como JAVA, C e C++. Segue abaixo um exemplo de código envolvendo algumas das estruturas da linguagem php:

```
<HTML><HEAD></HEAD><BODY>
<?php function valor()   {
      return true;
} $expressao = true; ?> <?php  if ($expressao) {  ?>
    <b>Verdadeiro.</b>
<?php       } else {                          ?>
    <b>Falso</b>
<?php       }       /* isso {\e} um coment{\a}rio */    ?>
```

```
<?php
    for ($i = 1; $i <= 10; $i++) { print $i; }
    //isso tamb{\e}m {\e} um coment{\a}rio
?> </BODY></HTML>
```

3.8 Visual Basic

O Visual Basic é uma linguagem de programação que possibilita a criação de programas executáveis (arquivos .EXE). Este arquivo pode ser todo o código escrito compilado, tornando-se um enorme e único arquivo, mas o VB permite que o programador divida o projeto em diversos arquivos menores que podem ser usados por diversos aplicativos, estes arquivos menores podem ser DLLs, OCXs ou EXEs. Este capítulo destina-se a descrever a linguagem de programação, Visual Basic.

3.8.1 Histórico

A história do Visual Basic começa com a criação do BASIC (Beginner's All-purpose Symbolic Instruction Code) em 1964, que é uma linguagem bastante fácil de ser aprendida e usada por programadores iniciantes. Esta linguagem tornou-se muito popular e utilizada por vários programadores durante os 15 anos seguintes, durante os quais, diversos compiladores e interpretadores para BASIC foram criados.

Em 1975, quando a Microsoft era uma empresa nova, uma versão da linguagem BASIC foi um dos primeiros produtos que ela criou, e que se tornou um sucesso. O BASIC da Microsoft e seu sucessor, o Quick BASIC tornaram-se as versões do BASIC mais amplamente disponíveis nos PCs. O Quick BASIC estava disponível para Windows, mas era muito trabalhoso escrever código pra interface Windows fazendo com que ele não fosse apropriado para este novo ambiente. Para tal, a Microsoft criou um novo produto que juntava a facilidade do BASIC com um ambiente de programação que permitia criar interfaces gráficas para um programa. Este produto era o Visual Basic 1.0.

O VB ficou muito popular com o passar do tempo. Esta popularidade deve-se principalmente a possibilidade de a Microsoft, o programador e outros fornecedores criarem componentes de interface personalizada que poderiam ser adicionados a diferentes programas. Esses componentes permitiam que o programador criasse aplicativos mais robustos que continham vários componentes desses o que tornava o desenvolvimento mais rápido e fácil, aumentando a produtividade.

Até a versão 4.0 do Visual Basic, o código era interpretado o que tornava sua execução muito lenta em relação a outras linguagens como o Delphi ou o C++, a partir da versão 5, em 1997 era possível criar versões compiladas ou interpretadas de programas o que melhorou o desempelho [MAC 03].

3.8.2 DLL

Uma maneira de melhor compartilhar o código é criar uma biblioteca de códigos armazenados em um arquivo separado dos programas que o utilizam e que possa ser alterado de maneira independente. Esta biblioteca é chamada de componente e é normalmente criada na forma de um arquivo .dll. Utilizar uma biblioteca é o melhor modo de compartilhar códigos.

No decorrer do desenvolvimento de novas versões do Visual Basic, sua capacidade de criar componentes foi avançando, hoje este é um recurso extremamente essencial em projetos de desenvolvimento.

3.8.3 API

Quando a Microsoft escreveu o Windows, eles colocaram uma enorme quantidade de código dentro de bibliotecas as quais podem ser acessadas por programadores, não importando que linguagem eles estivessem usando, aumentando assim o poder dos aplicativos.

Como vimos no tópico anterior (DLLs), é possível criar sua própria biblioteca de procedimentos no VB através do uso de arquivos .dll permitindo que diversos aplicativos acessem o mesmo código. A Microsoft faz algo parecido com o Windows, muitos ar-

quivos executáveis no Windows não contém todo o código necessário para funcionarem, utilizando arquivos .dll para seu funcionamento.

Um programador avançado em Visual Basic deve saber usar as APIs do Windows já que existem coisas que o VB não dá suporte e um bom programador VB não deve limitar-se. Existem várias DLLs disponíveis mas as três mais comunmente utilizadas são:

- User32.dll - que controla objetos visíveis (aquilo que pode ser visto na tela);
- Gdi32.dll - é a casa da maioria das APIs orientadas a gráfico;
- Kernel32.dll - provê acesso para funcionalidade de baixo nível do sistema operacional.

3.8.4 Estado da Arte

Todos os recursos complementares que foram sendo adicionados ao Visual Basic, foram feitos em cima de uma base existente. O que possibilitava a compatibilidade com versões anteriores, porém causava a acumulação de lixo. Porém reescrever uma linguagem a partir do zero é muito difícil e caro. Mas foi o que a Microsoft fez na passagem do Visual Basic 6.0 para o Visual Basic .NET. Ela reescreveu a linguagem para criar uma versão limpa que eliminasse o lixo que se acumulou por uma década de atualizações sucessivas.

Á primeira vista, o .NET parece ser um jogo de marketing, uma maneira de evitar um outro número após o Visual Basic, mas o .NET representa uma série de novas tecnologias e conceitos que formam uma plataforma para desenvolver aplicativos.

A plataforma .NET é uma camada que existe abaixo de seus programas e fornece um conjunto de serviços e funções básicas. Esta camada contém um conjunto de aplicativos e sistemas operacionais chamados de servidores .NET, um conjunto de objetos chamado .NET Framework e um conjunto de serviços que dá suporte a diversas linguagens (as linguagens .NET) que é o Common Language Runtime (CLR).

3.9 ActiveX

ActiveX é uma especificação desenvolvida pela Microsoft que permite aos programas Windows comuns executar dentro de uma página Web. Os programas ActiveX podem ser escritos em linguagens como Visual C++ e Visual Basic e são compilados antes de serem instalados no servidor Web.

Aplicativos ActiveX, denominados controles, são baixados e executados pelo navegador Web. Estes controles podem ser instalados permanentemente, como programas, eliminando a necessidade de baixá-los novamente.

A principal vantagem do ActiveX é que ele pode fazer quase qualquer coisa, o que pode vir a ser uma desvantagem já que o programador pode fazer controles maliciosos que podem causar danos ao computador do usuário.

Felizmente, o ActiveX inclui um recurso de assinatura que identifica a fonte do controle e impede que os controles sejam modificados. Embora isso não impeça que um controle danifique seu sistema, você pode especificar as fontes em que confia.

O ActiveX possui duas desvantagens significativas. Primeiro, não é tão fácil de programar como uma linguagem de script ou Java. Segundo, o ActiveX é proprietário: funciona somente no Microsoft Internet Explorer e apenas em plataformas Windows.

3.10 Sistemas de Gerenciamento de Banco de Dados

Para armazenar dados de um sistema qualquer que interage com um usuário poderíamos utilizar um sistema de arquivos que guarda os dados que define-se como importantes e que devem poder ser consultados ou alterados em um momento futuro, porém a utilização de arquivos texto para este propósito podem trazer uma série de problemas como Inconsistência e redundância de dados, dificuldade de acesso aos dados, isolamento dos dados, problemas de integridade, problemas de atomicidade, anomalias no acesso concorrente e problemas de segurança [SIL 99]. Para resolver estes problemas é necessário utilizar-se de sistemas de gerenciamento de banco de dados que nada mais é do que um conjunto de dados associados a um conjunto de programas para acesso a estes dados. Eles são proje-

tados para gerenciar de forma conveniente e eficiente grandes volumes de dados, possibilitando a estruturação das informações a serem armazenadas assim como sua segurança. Os sistemas de banco de dados fazem suas funções dentro de um modelo de três níveis, nível físico, nível lógico e nível de visão, este último seria sua interface com o usuário, abstraindo assim a necessidade do usuário de conhecer detalhes sobre o armazenamento, inter-relacionamento entre os dados ou até partes dos dados que certos usuários não precisam conhecer. Este capítulo destina-se a descrição de sistemas de gerenciamento de banco de dados em geral e no caso específico do MySQL.

3.10.1 A Linguagem SQL

A linguagem SQL é uma linguagem para modelagem da estrutura de um banco de dados assim como para consultas no mesmo.

A SQL foi desenvolvida pela IBM com o nome original de Sequel no início dos anos 70. A evolução do Sequel resultou na SQL, Structured Query Language ou Linguagem de Consulta Estruturada.

Mais detalhes sobre a SQL serão dados no próximo capítulo onde explicaremos melhor esta linguagem através de exemplos.

3.10.2 MySQL

O MySQL não é um banco de dados como muitos pensam, na verdade é um sistema que possibilita que um usuário, crie, mantenha e gerencie banco de dados eletrônicos, é um SGBD (Sistema de Gerenciamento de Banco de Dados).

3.10.2.1 Histórico

Michael Widenius, o inventor do MySQL, da empresa sueca TcX trabalhava com banco de dados desde 1979. Em 1994 sua empresa começou a desenvolver aplicações baseadas na web e utilizou o UNIREG para tal. Porém, este era muito caro para desenvolver suas tarefas que era a de gerar páginas da web dinamicamente. Então a TcX

começou a dar atenção ao SQL e mSQL. Contudo o mSQL estava em suas versões iniciais e seu desempenho era pobre comparado ao UNIREG.

Então Widenius entrou em contato com David Hughes, autor do mSQL, para tentar conectar o mSQL ao UNIREG, sem sucesso. Então a TcX criou seu próprio servidor de banco de dados compatível com suas exigências e compatível com o mSQL, seu nome era MySQL 1.0 [RJY 00].

3.10.2.2 Vantagens e Desvantagens

O MySQL não é gratuito como muitos pensam, embora seja gratuito para uso pessoal ou por universidades ele deve ser pago por empresas comerciais, embora tenha código fonte aberto (open source) ou seja, o usuário pode analizar e alterar o código sem custos adicionais.

A principal vantagem do MySQL é seu desempenho porém esta vantagem causa desvantagens pra usuários de médio e grande porte que precisam sacrificar o desempenho para ter outros recursos. Este recursos são transações, gatilhos, procedimentos de armazenagem, subseleções (já implementado no MySQL 4.1) e objetos. Embora o MySQL tenha o objetivo de incorporar alguns destes recursos, dando a opção para que o usuário o desabilite.

3.10.2.3 Segurança

Além de o usuário querer ter seus dados armazenados de forma segura, ele também quer ter certeza de que qualquer pessoa tenha acesso a eles. O MySQL utiliza seu próprio servidor de banco de dados para implementar esta segurança. O processo de instalação do MySQL cria um banco de dados chamado ”mysql”, este banco de dados tem as tabelas: db, host, user, tables-priv e colums-priv. Estas tabelas são utilizadas para decidir a quem é permitido fazer o que.

3.10.2.4 PHP e MySQL

O PHP tem uma biblioteca pronta de acesso à base de dados MySQL muito fácil, prático e robusta. Abaixo descrevemos algumas das funções que podem ser utilizadas assim como exemplos de um código que faz acesso à base de dados MySQL.

mysql_connect – Abre uma conexão com o servidor MySQL mysql_close – Fecha a conexão com o MySQL mysql_fetch_row – Retorna o resultado de uma linha numa matriz numérica mysql_query – Realiza uma query MySQL e retorna um resultado mysql_db_query – Envia uma query ao MySQL

```
<?php
$cn = mysql_connect($endereco,$usuario,$senha);
$sql = ”insert into tabela values(’$valor1’,’$valor2’)”;
$sucesso = mysql_db_query($site_owner,$sql,$cn);
$sql = ”select campo1,campo2 from tabela ”;
$sql .= ”where campo4=’Carlos’ and campo9>=’3’”;
$resultado = mysql_query($sql);
while ($linha = mysql_fetch_row($resultado))
echo (”campo1=”. $linha[0] . ”e campo2=”. $linha[1]);
mysql_close($cn);
?>
```

3.11 ERwin

O Erwin (Entity Relationship for Windows) é um programa para modelagem de banco de dados muito popular originalmente criado pela Logic Works, Inc e mais tarde comprado pela Platinum Technology a qual foi adquirida mais tarde pela Computer Associates. Esta última lançou recentemente uma nova versão do Erwin, o AllFusion Erwin Data Modeler.

A principal utilidade deste programa é a possibilidade da criação de um Diagrama Entidade-Relacionamento que pode ilustrar de maneira gráfica, tabelas de uma base de

dados e seus relacionamentos (um para um, um para muitos, muitos para muitos).

Este programa é de extrema utilidade visto que fornece uma interface bastante intuitiva de modelar qualquer base de dados e uma vez modelada esta base, podemos visualizá-la de maneira clara e fácil, podemos alterá-la utilizando uma interface amigável e quando necessária a migração para outro banco de dados, a conversão fica muito fácil já que o Erwin suporta uma série de banco de dados diferentes. Além de permitir que programadores visualizem a modelagem da base de dados com a qual tem de lidar para desenvolver seu aplicativo, sem precisar ter o banco de dados em questão instalado em seu microcomputador, mas apenas o Erwin.

3.12 Outras Ferramentas

Além das ferramentas descritas anteriormente utilizamos outras ferramentas que valem a pena ser descritas neste capítulo, são elas EditPlus 2.11, Adobe Photoshop 7.0 e phpMyAdmin 2.3.1. Outras ferramentes também foram utilizadas mas são conhecidas o suficiente e não são relevantes para este documento, tais como WS-FTP e Microsoft FrontPage que são respectivamente um cliente de FTP e um aplicativo para construção de páginas Web.

O EditPlus é um excelente editor de textos de 32 bits com navegador imbutido, editor de HTML e editor de algumas linguagens de programação. Ao mesmo tempo que pode ser um bom substituto para o bloco de notas do Windos, ele também oferece muitas ferramentas poderosas para autoria de páginas Web e programas. Algumas de suas características são a capacidade de destacar a sintaxe, através de colorações diferentes, para HTML, CSS, PHP, ASP, Perl, C/C++, Java, JavaScript e VBScript. Também contém um navegador para visualizar páginas HTML, e suporta comandos de FTP para fazer o upload de arquivos para um servidor de FTP, além de outras ferramentas que auxiliam a confexão de páginas da Web.

O Adobe Photoshop 7.0 é uma ferramenta de manipulação de imagens que dispensa introduções por ser bastante popular e amplamente utilizada por designers profissionais e produtores gráficos assim como por usuários de imagens comuns e Webmasters.

Em outras palavras o Adobe Photoshop é indispensável para dar uma aparência bonita e comercial para qualquer sítio da web, embora existam outros aplicativos com a mesma funcionalidade, encontramos no Photoshop a ferramenta mais fácil de lidar e robusta ao mesmo tempo.

O phpMyAdmin 2.3.1 é um conjunto de páginas PHP que servem para administrar uma ou diversas bases de dados MySQL. Esta ferramenta contém diversas funcionalidades, tais como criar, modificar e deletar tabelas ou campos na mesma, visualizar estruturas e dados de tabelas podendo incluir, alterar ou deletar dados nas mesmas, além de diversas outras funcionalidades, como gerar arquivos com script de criação da base de dados, fazer backup do banco de dados, entre outros.

3.13 Conclusão

Neste capítulo abordamos todas as tecnologias e ferramentas que devemos ter como base para o desenvolvimento de nosso projeto. Descrevemos cada tecnologia e ferramenta utilizada detalhadamente e pudemos adquirir uma série de novos conhecimentos nestes diversos assuntos.

Descrevemos as ferramentas de segurança que utilizamos que são a CrytoAPI, uma dll que executa diversas funções como geração de chaves, cifragem e decifragem e a Capicom que é uma COM que executa funções de segurança proporcionando uma boa interface ao programador.

Abordamos, também, neste capítulo, os Servidores Web, da Microsoft (IIS) e o Apache que funcionam como um serviço para disponibilização de páginas Web na Internet ou Intranet. Estudamos, também, a segurança aplicada a estes servidores através do protocolo SSL da Netscape.

Estudamos alguns scripts que são executados na máquina do cliente que está acessando um sítio. que podem ser Javascript, Jscript ou VBScript assim como os que são executados diretamente no servidor, em nosso caso o PHP.

O Visual Basic foi estudado como uma ferramenta capaz de gerar componentes ActiveX, que é uma peça fundamental para aplicações Web mais sofisticadas.

O sistema de gerenciamento de banco de dados abordado foi o MySQL, que além de rápido e ”grátis”, é bastante fácil de ser utilizado através da linguagem PHP.

Outras ferramentas foram estudadas neste capítulo como o Erwin, EditPlus, etc.

Capítulo 4

Sistema Implementado

4.1 Introdução

Para melhor compreendermos as técnicas de segurança em computação, mais especificamente em sítios da Web, desenvolvemos um sistema para a realização de licitações através de uma interface Web segura.

Este sistema desenvolvido apresenta uma alternativa ao modelo atual, que é feito manualmente, o qual dá uma margem de falha de segurança. Nosso sistema além de ser seguro, é prático e ágil. Mostraremos, neste capítulo, a maneira pela qual desenvolvemos nosso sistema, descrevendo cada etapa do desenvolvimento e as tecnologias utilizadas em cada uma delas. Procuramos ilustrar da melhor maneira possível para que o processo seja compreendido por qualquer indivíduo, seja da área de informática ou de qualquer outra área.

4.2 Primeiro Protótipo

Após fazermos o levantamento dos dados junto ao nosso orientador, demos início aos estudos necessários para o conhecimento das tecnologias e ferramentas a serem utilizadas. Para exercitar os conhecimentos adquiridos e conhecer melhor o sistema que deveria ser implementado, desenvolvemos um protótipo inicial que deveria simular o

funcionamento do sistema como um todo, porém sem a utilização de tecnologias de segurança, esta foi a primeira etapa do projeto.

A primeira etapa, compreendida entre o final do mês de março e meados de maio de 2002, desenvolvemos um protótipo com poucas funcionalidades e com um caráter mais ilustrativo com o intuito de fazer uma apresentação ao cliente em potencial. Este protótipo foi abandonado, tanto sua funcionalidade quanto sua interface e serviu apenas para que compreendêssemos melhor como iria funcionar o projeto como um todo e pudéssemos nos familiarizar com as tecnologias que seriam utilizadas no decorrer de todo o projeto. Nesta primeira etapa, estudamos apenas tecnologias necessárias para desenvolver um sítio comum na Web, sem a utilização de técnicas de segurança. As tecnologias estudadas nesta etapa foram PHP, MySQL, JavaScript e servidor Web IIS.

A figura 4.1 ilustra a primeira tela do protótipo desenvolvido. que foi de extrema importância para o desenvolvimento do segundo protótipo que confeccionamos em seguida, porque além de nos familiarizarmos melhor com as tecnologias e ferramentas, pudemos ter um conhecimento muito mais detalhado das necessidades do projeto, aplicando correções ao projeto inicial e incrementando outras necessidades.

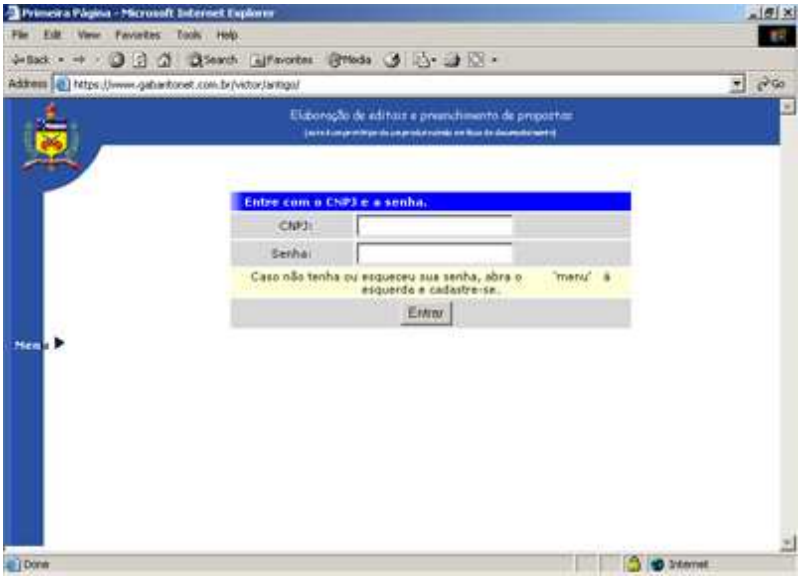

Figura 4.1: Tela para logon no primeiro protótipo do sistema

Neste protótipo, o usuário deveria cadastrar-se no sistema, como sendo comprador

ou fornecedor, escolhendo uma senha neste cadastro. Para entrar no site e começar a utilizar o sistema, o usuário deveria logar-se entrando com o CNPJ de sua empresa e a senha escolhida na fase cadastral. Após logar-se, o usuário escolhe se quer iniciar o processo de licitação, caso seja um comprador, ou fazer uma proposta em algum edital existente no sistema, caso seja um fornecedor. Estes dois procedimentos diferentes serão melhor explicados na seção seguinte. As páginas neste protótipo foram todas feitas em HTML, os scripts do lado do servidor foram escritos em PHP 3 e os scripts do lado do cliente em JavaScript. O servidor Web utilizado foi o IIS 5.0 da Microsoft em um computador com o sistema operacional Windows 2000 Server. Os dados inseridos pelo usuário eram cadastrados na base de dados MySQL. Fizemos uma modelagem dos dados bastante, básica que foi praticamente substituída no segundo protótipo.

4.3 Definição e instalação dos pré-requisitos

Após definidos os pré-requisitos necessários para a implementação de nosso projeto junto a nosso orientador, que consistiam de uma série de ferramentas necessárias para o funcionamento de um sítio com tais características, iniciamos a instalação dos pré-requisitos. Conseguimos então um servidor com Windows 2000 Server instalado para começarmos nosso trabalho.

Em primeiro lugar, criamos um sítio de testes em um servidor Web Microsoft, o IIS 5.0, para tal seguimos os seguintes passos:

1. Instalamos o IIS em nosso servidor Windows 2000 Server;
2. Instalamos o PHP3 no mesmo servidor;
3. Abrimos o IIS 5.0, selecionamos a pasta Site da Web padrão;
4. Adicionamos uma nova pasta virtual com o alias TCC e a direcionamos para uma pasta localizada no mesmo micro;
5. Definimos as permissões de acesso para permitir leitura e execução de Scripts (para permitir a execução de páginas com a extensão .php).

Uma vez seguidos estes passos, poderíamos testar nossas páginas PHP na Internet. Partimos então para um próximo pré-requisto, que era a instalação da base de dados MySQL, instalamos este sistema de gerenciamento de banco de dados em nosso servidor de testes, após o download desta ferramenta que é gratuita. Fizemos alguns testes de conexão para garantir que estava funcionando e seguimos para o próximo passo.

Seguindo sugestões, instalamos o phpMyAdmin 2.3.1 com a intenção de fazer uma interface entre o PHP e o MySQL para melhor administrar a base de dados.

Para garantir o sigilo dos dados que trafegam em nosso site, instalamos o SSL em nosso servidor Web, para tal foi necessário preparar uma requisição de certificado, enviá-la a uma entidade certificadora e aguardar a emissão do certificado do servidor pela mesma. Após recebido o certificado, instalamos este no servidor. Para exigir que nosso sítio utilize o protocolo SSL, entramos nas propriedades de nossa pasta virtual e em seguida na pasta ”Segurança de pasta”, em Comunicações seguras clica-se em Editar e marca-se a primeira opção Requerer canal seguro (SSL). A partir disso, esta pasta virtual irá funcionar apenas com o protocolo SSL, ou seja, o endereço ao invés de começar com http:// , irá começar com https:// e no rodapé do navegador irá aparecer um cadeado indicando que a conexão com o sítio é segura. Ao clicar duas vezes no cadeado pode-se visualizar o certificado do servidor como é mostrado na figura 4.2.

Figura 4.2: Certificado instalado no servidor para conexões seguras (SSL)

Neste ponto temos o servidor pronto para dar início ao nosso trabalho de modelagem de dados, programação do código das páginas PHP, testes e correções.

4.4 Modelagem dos dados

Nesta parte, definimos as tabelas necessárias para guardar os dados do usuário e seus relacionamentos, utilizamos o programa Erwin 4.0 para fazer esta modelagem, a figura 4.3 mostra o diagrama Entidade-Relacionamento gerado por este programa.

Utilizamos uma série de tabelas e esta seção destina-se a explicar cada uma delas.

A tabela TbCadEmpresa é a tabela com o cadastro de todas as empresas que são compradoras, ou empresas que preenchem editais assim como as empresas que são possíveis fornecedoras de produtos para estes compradores. Esta tabela contém o campo cnpjEmpresa que é a chave primária desta tabela e representa o cnpj da empresa; o campo nomeEmpresa que guarda a razão social da empresa; flgCompradorFornecedor que indica se a empresa cadastrada é fornecedora ou compradora; os campos nomePessoa e emailPessoa correspondem respectivamente ao nome e email da pessoa que será o contato da empresa responsável pelo manuseamento do sítio; nomeRua, nomeBairro, nomeCidade, codigoEstado, numeroTelefone e numeroFax, correspondem a dados de endereço e telefones de contado da empresa em questão; o campo areaAtuacao é um campo que indica a área de atuação da empresa, como por exemplo informática, alimentos ou construção civil; generoProdutos guarda o gênero de produtos com o qual a empresa lida; identidadeDigital, o certificado digital que a empresa utiliza para logar-se no sítio.

TbLog é uma tabela responsável por armazenar as descrições das transações efetuadas pelas empresas durante a navegação pelo sítio para facilitar a administração do mesmo em caso de erros ou má fé de participantes. As chaves primária são cnpjEmpresa que é o campo que faz o relacionamento desta tabela com a TbCadEmpresa e nuLog que é um campo auto incremental que dá um número diferente para cada registro de log. Os outros campos são deLog que é a descrição da ação que está sendo efetuada e dthrLog que é a data e hora na qual esta ação foi efetuada.

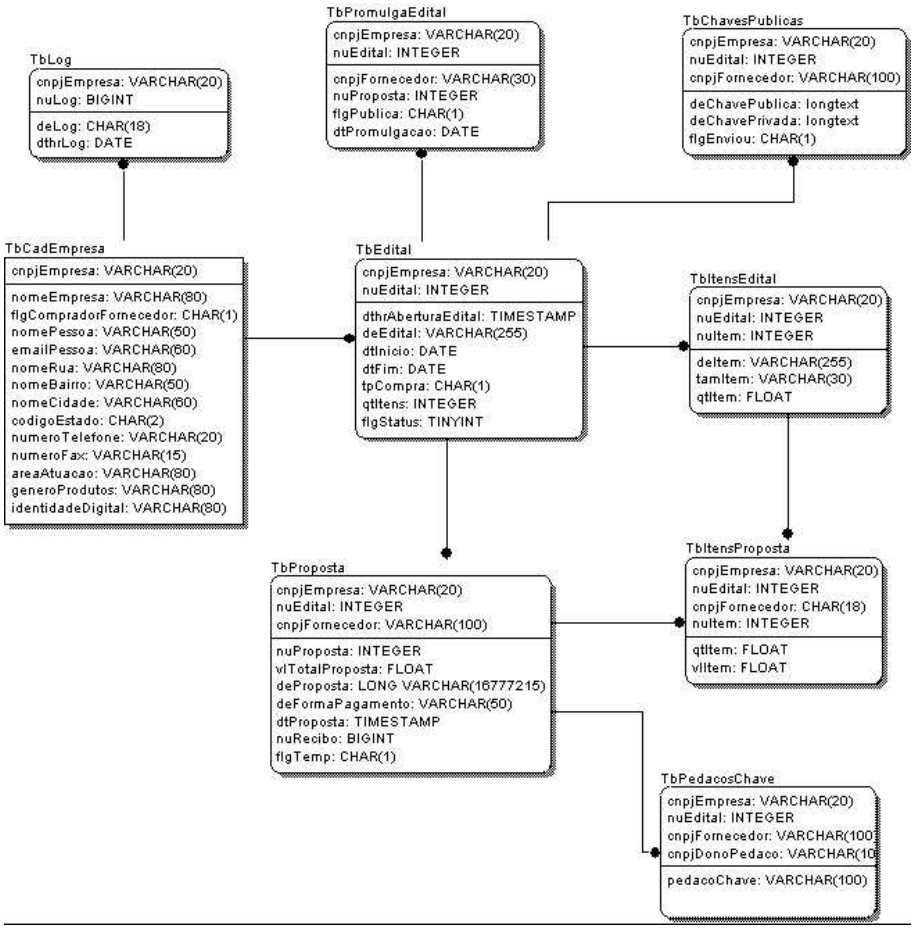

Figura 4.3: Modelagem dos dados no ERwin

A TbEdital contém os dados gerais dos editais formulados pelas empresas compradoras, esta tabela relaciona-se com a tabela TbCadEmpresa através da chave estrangeira cnpjEmpresa e contém outra chave primária, o campo nuEdital que representa o número do edital proposto pela empresa compradora. O campo dthrAberturaEdital armazena a data e hora em que o edital foi aberto; deEdital representa a descrição geral do edital proposto; dtInicio é a data de início de recebimento das propostas; dtFim a data de encerramento do recebimento de propostas; tpCompra é o tipo de compra podendo ser 1 - ’Concorrência’, 2 - ’Tomada de Preço’, 3 - ’Carta Convite’; qtItens é a quantidade de ítens que o edital possui; flgStatus define o estado do edital como sendo -1 - edital apenas foi incluído, ainda não foi confirmado, 0 - já foi confirmado, mas não foi publicado, 1 - edital foi publicado, 2 - o recebimento das propostas do edital foi iniciado, o edital está

em andamento, 3 - o recebimento das propostas foi finalizado, o edital está encerrado, 4 - a proposta vencedora já foi promulgada.

TbItensEdital é uma tabela auxiliar à TbEdital, portanto relaciona-se a mesma através de cnpjEmpresa e nuEdital, e contém uma terceira chave, nuItem que é um seqüencial dos ítens pertencentes ao edital. Três campos são utilizados para descrever os ítens do edital, deItem fornece a descrição geral do ítem; tamItem é o tamanho especificado do ítem, a unidade de medida deve ser especificada neste mesmo campo; qtItem é a quantidade deste ítem que está sendo requerida no edital.

A TbChavesPublicas é outra tabela auxiliar da TbEdital e mais uma vez relaciona-se a ela pelos campos cnpjEmpresa e nuEdital, esta tabela conterá as chaves pública e privada dos certificados utilizados pelos fornecedores para cifragem das propostas. A chave primária será o campo cnpjFornecedor que corresponde ao cnpj da empresa fornecedora que fez a proposta. O campo deChavePublica indica o caminho físico onde se encontra o arquivo da chave pública do fornecedor e deChavePrivada para a chave privada. O flag flgEnvio indica se o fornecedor já enviou a proposta ou não, a utilidade deste campo será melhor explicada no ítem 4.6.

A TbPromulgaEdital é mais uma tabela auxiliar a TbEdital relacionando-se pelos campos cnpjEmpresa e nuEdital, ela contém os dados da promulgação do vencedor da licitação. O campo cnpjFornecedor indica o cnpj da empresa, que foi a vencedora do edital. O campo nuProposta é o número da proposta que foi encaminhada pela empresa vencedora; flgPublica diz se a promulgação do vencedor vai tornar-se pública no sítio ou não; dtPromulgacao é a data em que a promulgação do vencedor foi efetuada.

TbProposta armazena os dados da proposta elaborada pelo fornecedor, ela relaciona-se com a tabela TbEdital por cnpjEmpresa que é o cnpj da empresa que elaborou o edital do qual o fornecedor está fazendo parte e nuEdital que é o número deste edital. nuProposta é o campo com o número da proposta elaborada pelo fornecedor; vlTotalProposta contém o valor total proposto pelo fornecedor para a licitação; deProposta contém uma descrição geral da proposta elaborada; deFormaPagamento, a forma como serão pagos os ítens da licitação; dtProposta, a data em que a proposta foi encaminhada ao servidor; nuRecibo, o número do recibo enviado pelo servidor ao fornecedor como garantia do en-

vio da proposta; flgTemp indica se a proposta é a proposta definitiva a ser considerada ou apenas uma proposta temporária com direito a mudanças antes do prazo de encerramento do edital.

A tabela TbItensProposta relaciona-se à TbProposta e à TbItensEdital e portanto contém as chaves cnpjEmpresa, nuEdital, cnpjFornecedor e nuItem. Esta tabela representa os dados da proposta referentes a cada ítem do edital em separado contendo a quantidade disponível para cada ítem, o campo qtItem e o valor proposto para cada qual, campo vlItem.

A TbPedacosChave é uma tabela auxiliar a TbProposta e tem as chaves cnpjEmpresa, nuEdital, cnpjFornecedor e mais um campo de chave, o cnpjDonoPedaco que indica quem é o dono do pedaço da chave que está sendo cadastrado. Além destes, existe um campo, o pedacoChave que guarda cada pedaço da chave primária utilizada para cifrar a proposta do fornecedor. A utilidade desta tabela será explicada mais adiante.

4.5 Protocolo implementado

O protocolo criptográfico para licitação eletrônica foi desenvolvido por Fernando Carlos Pereira em [PER 03], e que com nosso auxílio, foi aplicado na prática através da sua implementação no sistema seguro de compras.

Este protocolo é aplicado ao processo de licitação pública para garantir a confidencialidade das propostas comerciais entregues ao comprador, durante o período de tempo que antecede o evento oficial da abertura e julgamento das propostas. Para alcançar este objetivo, o protocolo faz uso de técnicas e outros protocolos de criptografia.

O protocolo utilizado é baseado em criptografia temporal que é um conceito novo e pouco conhecido, mas que pode ser muito útil.

A criptografia temporal permite determinar o momento no futuro em que uma informação eletrônica poderá ser acessada [PER 03], ou seja uma informação é enviada para o futuro e durante o período em que a informação não pode ser vista, ela é protegida por criptografia e pela ocultação da a chave de decifragem, que será utilizada para decifrar a informação após transcorrido o período tempo em que ela deve permanecer secreta. A

criptografia temporal é de grande utilidade para uma série de aplicações e mais especificamente para nossa aplicação que é o sistema de licitações públicas onde a proposta comercial é selada e somente poderá ser aberta em uma data futura pré-definida junto com todas as outras propostas.

O método de ocultação da chave de decifragem da informação na criptografia temporal utilizado em nosso sistema foi o de entidades confiáveis. Este método é baseado na confiança do segredo a certas entidades sob a promessa de que o segredo será entregue em data futura, previamente determinada. O protocolo implementado é baseado em um esquema de compartilhamento de segredos onde a chave de decifragem de um documento é dividido em partes, as quais são entregues a diferentes entidades pertencentes a um grupo, que só deverão revelar suas partes em um momento futuro. A união das partes formará a chave inteira, que entçao possibilitará a decifragem do documento cifrado anteriormente.

As premissas básicas deste protocolo são a igualdade de interesses que é a necessidade dos membros do grupo de garantir a confidencialidade por um período de tempo dos documentos eletrônicos emitidos, a outra é a solidariedade que refere-se a situação em que um membro do grupo tenta prejudicar outro membro, neste caso o mesmo será prejudicado na mesma proporção. Estas premissas forçam as pessoas a respeitarem as regras dos protocolos.

Neste protocolo é utilizado o conceito de confiança distribuída onde todos os membros do grupo tem a mesma responsabilidade pela confidencialidade dos documentos eletrônicos, que é a distribuição de partes da chave de decifragem entre todos os membros do grupo. Este esquema de distribuição da chave de decifragem é feito pelo esquema de Shamir, vide capítulo 2.2.6.2.

No protocolo implementado, cada membro cria seu próprio par de chaves pública e privada, utiliza a chave pública para cifrar o documento eletrônico que será enviado a outro membro do grupo, a chave privada deve ser compartilhada por ele mesmo entre os demais membros através do esquema de Shamir.

A construção das chaves pública e privada é feita seguindo os seguintes passos:

1. Os membros escolhem o parâmetro T - período de tempo e t - total de membros

necessários para decifrar as chaves;

2. Cada membro gera um par de chaves pública e privada;
3. Cada membro quebra sua chave privada em n partes utilizando o esquema de Shamir com n e t, sendo n o número total de membros menos um;
4. Cada membro assina digitalmente as partes da sua chave privada para garantir a autenticidade das mesmas;
5. Cada membro envia seus pedaços da chave privada para todos os demais membros do grupo.

Quando o tempo T é terminado, os documentos devem ser decifrados, portanto cada membro deve entregar sua chave privada de maneira espontânea ao membro que recebeu os documentos cifrados pela chave pública, caso algum dos membros se recusar a entregar a chave, os demais membros unem-se em um grupo de t pessoas e reconstruem a chave faltante.

Embasados no protocolo descritos neste mesmo capítulo, desenvolvemos nosso sistema de licitação eletrônica. O processo de licitação foi dividido em 5 etapas que serão descritas a seguir.

Em uma primeira etapa, a fase preparatória, o documento formal que é o edital proposto pela empresa compradora é preparado e publicado em algum lugar para que todos possam vê-lo, nesta mesma etapa é definido o período no qual as propostas encaminhadas pelos fornecedores permanecerá oculta.

Após a primeira etapa ocorre a fase cadastral aonde os fornecedores que se interessaram pelo edital proposto devem assumir-se como interessados para poderem fazer parte da disputa, nesta etapa as chaves públicas que os fornecedores utilizarão para cifrar suas respectivas propostas são enviadas a um recipiente em comum a todos. É necessário que os fornecedores, neste ponto, assumam um compromisso para com a licitação, caso contrário o processo pode ser prejudicado.

Na terceira etapa que é a fase de envio das propostas, cada fornecedor envia sua proposta selada, ou seja, assinada com seu certificado digital e protegida por criptografia.

Aqui também ocorre a divisão da chave privada de cada fornecedor em n partes através do esquema de Shamir, conceituado na seção 2.2.6.2. As chaves públicas enviadas pelos fornecedores na etapa anterior agora são utilizadas para cifrar cada pedaço da chave privada de cada fornecedor. Após cifrados estes pedaços são assinados digitalmente e encaminhados para um recipiente em comum a todos os fornecedores.

A quarta etapa é quando o prazo de envio de propostas foi encerrado e as chaves privadas de todos os fornecedores são solicitadas, o motivo da solicitação desta chave é para evitar esforço computacional uma vez que seria possível obtê-las decifrando os pedaços das chaves que já se encontram disponíveis. Após solicitados de suas respectivas chaves privadas, os fornecedores devem agir de boa fé e encaminhá-las a um recipiente em comum, devidamente cifradas pela chave pública do comprador, para que suas propostas possam ser decifradas utilizando o algoritmo de criptografia assimétrica já que estas propostas foram cifradas com suas chaves públicas.

Na quinta e última etapa, o comprador recebe todas as propostas e as decifra com as chaves privadas fornecidas pelos candidatos a fornecedor do produto solicitado no edital. Caso algum fornecedor tenha agido de má fé e não tenha enviado sua chave privada, os pedaços das chaves privadas enviados anteriormente para todos os outros fornecedores que fizeram parte do processo de licitação, serão utilizados para recuperar a chave privada faltante, através do esquema de Shamir. Feita a decifragem de todas as propostas o comprador poderá escolher o vencedor do processo licitatório e promulgá-lo como tal.

4.6 Protótipo Final

Na segunda etapa do projeto, uma vez que já tínhamos um protótipo pronto, todos os pré-requisitos instalados, uma modelagem de dados pronta e as definições do funcionamento do protocolo criptográfico para a licitação, ficou mais fácil de implementarmos o protótipo utilizando os conceitos e tecnologias estudados.

Para iniciarmos a explicação do funcionamento completo do sítio na Internet desenvolvido, mostraremos todos os passos necessários tanto para elaborar um edital e ao final promulgar o vencedor como para inscrever-se como interessado em um edital, enviar

a proposta e concorrer a fornecedor dos ítens do edital.

Figura 4.4: Instalação da Capicom 2.0

Em primeiro lugar o usuário do sistema, tanto comprador quanto fornecedor, deve abrir seu navegador Web que deve ser Microsoft Internet Explorer pois usamos funcionalidades que apenas este navegador suporta além do que o sítio foi projetado esteticamente para este navegador e outros navegadores podem apresentar distorções ao serem utilizados para visualizar o sistema. Uma vez aberto o navegador, o usuário deve entrar em nosso endereço na Web, que atualmente é http://www.gabaritonet.com.br/tcc. Ao entrar neste endereço em primeiro lugar aparecerá a figura 4.4, um aviso de que o sítio utiliza a Capicom, explicada na seção 3.3. A figura 4.5 que é um aviso de que esta página irá procurar por certificados digitais instalados no computador do cliente e pergunta se o usuário permite que a página acesse seus certificados ou não, este certificado é necessário para que o usuário possa logar-se no sítio. Identidades digitais podem ser obtidas no endereço http://ac.labsec.ufsc.br/. Após permitir que os certificados sejam acessados o usuário depara-se com nossa página de frente, o index.php que está sendo mostrado na figura 4.6. Nesta página, que é a página de capa de nosso sítio, além de mostrarmos uma interface que leva o usuário a qualquer lugar dentro do sítio sendo que as opções são a página restrita dos compradores, a página restrita dos fornecedores, o cadastro de usuários, para aqueles que ainda não estão cadastrados, e um link para acesso livre, o qual pode ser acessado por qualquer pessoa que deseje melhor conhecer o sítio. Caso o

usuário já esteja cadastrado, ele deve escolher seu certificado com o qual cadastrou-se previamente no sítio, isso levará o usuário diretamente para sua devida área restrita, porém se o usuário ainda não estiver cadastrado deve clicar no link ”Cadastro”que o levará para a tela de cadastro.

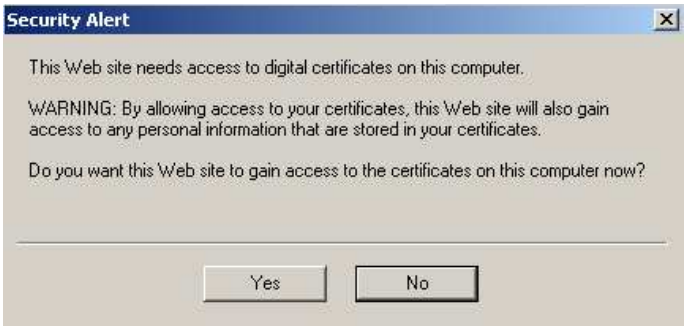

Figura 4.5: Alerta de segurança para acessar certificados digitais

Na tela de cadastro que é mostrada na figura 4.7 são requisitados os dados cadastrais da empresa fornecedora ou compradora, para que a mesma possa utilizar-se dos serviços do sítio. Para completar este passo o usuário já deve ter instalado em seu computador o seu certificado digital que utilizará para autenticar-se no sistema. A etapa de cadastro é composta de duas partes, a primeira são os dados pessoais que são automaticamente preenchidos ao escolher-se a identidade digital da lista apresentada, os dados são nome da pessoa que irá fazer o contato, e email da mesma através do qual o sistema comunicar-se-á com a empresa. A Segunda etapa são os dados da empresa propriamente dita que são Razão Social, CNPJ, Endereço, Bairro, Cidade, Unidade Federativa, Telefone, Fax, a área na qual a empresa atua e o Tipo que dirá para que fins a empresa estará utilizando o sistema, se para participar de licitações ou se para propô-las. Após clicar o botão enviar, os dados são emitidos para nossa base de dados MySQL através de um script PHP e armazenados.

Após ter se logado no sistema como um comprador o usuário entra em sua área restrita onde encontram-se todas suas licitações que já foram encerradas em uma parte da tela e as que ainda estão em andamento em outra parte da tela, como é mostrado na figura 4.8. Caso o usuário queira incluir um novo edital que passará pelo processo lici-

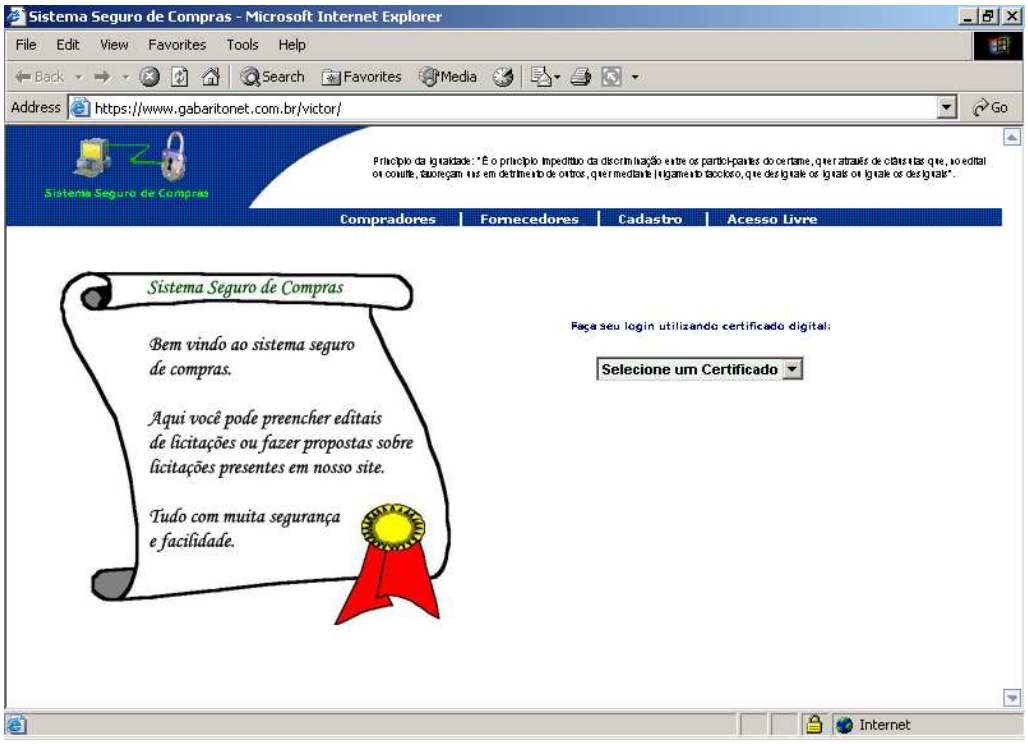

Figura 4.6: Página de capa do sítio

tatório existe um link para isso, ”incluir novo edital”, ao clicar neste link o usuário segue alguns passos para preencher completamente o documento de licitação, as figuras 4.9, 4.10 e 4.11 ilustram estes passos, no primeiro passo já vem preenchido o nome da empresa (Razão Social) e a data de inclusão do documento (a data atual), o campo objeto deve ser preenchido com a descrição geral do edital, o Tipo de Compra deve ser escolhido dentre as opções existentes que são Concorrência, Tomada de Preço, Carta Convite, Concurso, Leilão, Pregão e Compra Direta. A quantidade de ítens que compõem o edital deve ser preenchida para que a próxima tela possa definir-se automaticamente, e o período de entrega das propostas corresponde à data de início e término permitida para o encaminhamento de propostas ao servidor. Depois de clicar no botão ”Avançar”, o comprador passa para a próxima tela que além dos campos preenchidos na tela anterior, são solicitados os dados dos ítens do edital, a quantidade de ítens que é solicitada depende do número definido na tela anterior. Os dados solicitados são a

descrição de cada ítem, o tamanho (junto com a unidade de medida) e a quanti-

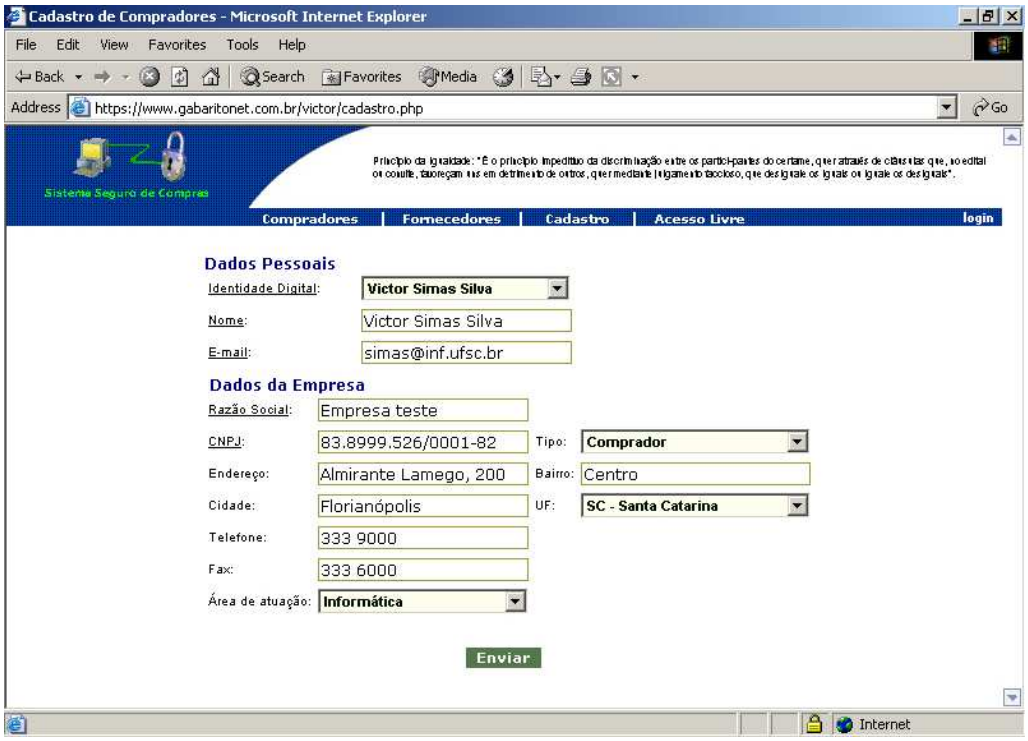

Figura 4.7: Página de cadastramento de usuários

dade de cada ítem que está sendo solicitado. No terceiro e último passo, é apresentado o documento completo, da maneira como será mostrado aos fornecedores e uma opção de ”Confirmar”para incluir o documento na base de dados ou ”Voltar”para alterar algo que esteja errado ou faltando. Após ter confirmado a inclusão do documento o comprador volta para a tela inicial de sua área restrita onde o seu edital foi incluído a sua lista de editais.

Observando a lista de editais o comprador pode optar por, alterar o edital (caso este ainda não tenha sido publicado), excluir (caso o período de recebimento de propostas ainda não tenha sido iniciado), o comprador pode também publicar o edital o que o torna público para que qualquer pessoa que acesse o sítio possa vê-lo, além disso é criado nesse momento os recipientes (pastas) no servidor onde serão depositadas as chaves públicas, privadas e os pedaços cifrados das chaves privadas de todos os fornecedores. O comprador pode também iniciar o recebimento de propostas ou encerrá-lo, estas duas últimas opções estão disponíveis apenas para este protótipo para fins de testes, e em um sistema

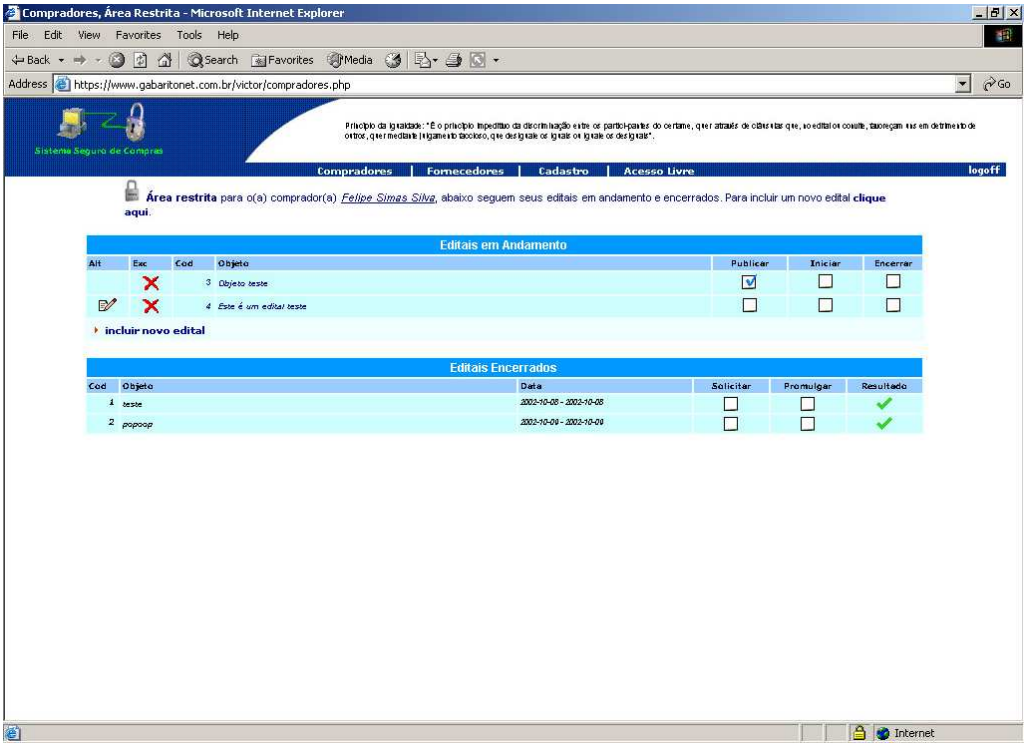

Figura 4.8: Área restrita do comprador

comercial estas funcionalidades deveriam ser disparadas automaticamente. No instante em que o comprador clica na opção ”iniciar”, o envio de propostas para este edital pelos fornecedores é autorizado, e quando a opção ”encerrar”é clicada, nenhuma proposta poderá mais ser entregue.

Depois de encerrado o período de recebimento de propostas, o edital sai da lista de ”Editais em Andamento”e vai para a lista de ”Editais Encerrados”que será explicada com mais detalhes mais adiante neste capítulo.

Quando os fornecedores entrarem em suas áreas restritas, figura 4.12, poderão ver os editais dos quais já fizeram parte e os que estão fazendo parte no momento, o status do edital indica seu estado atual diferenciando os editais já encerrados dos que estão em andamento. Para pesquisar editais publicados no sítio, existe um link chamado ”outros editais”que leva para uma página de consulta, figura 4.13.

Na página de consulta pode-se filtrar a consulta para minimizar o número de registros que serão mostrados, os campos de filtro são: Tipo de Edital que pode ser escolhido

Figura 4.9: Primeiro passo da elaboração de editais

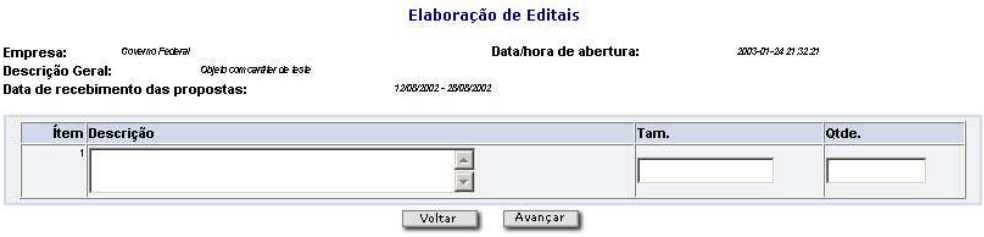

Figura 4.10: Segundo passo da elaboração de editais

dentre os tipos já vistos anteriormente ou escolher a opção ”todos”que retornará todos os tipos de editais na consulta; Status do Edital que pode ser Andamento (editais já publicados), Recebendo (editais cujas propostas já estão sendo recebidas), Encerrado (editais cujo envio de propostas já foi encerrado), Promulgado (quando o vencedor do edital já foi promulgado), Todos (consulta todos os editais); Nome do Cliente, consulta somente editais de uma determinada empresa, se for deixado em branco consulta de todas as empresas; Nome do Objeto, consulta apenas determinados objetos, como microcomputadores por exemplo; data de início e término de recebimento, consulta apenas editais dentro de um determinado período de tempo.

Depois de filtrar a consulta e clicar no botão ”Consultar’, será mostrada uma tela com todos os editais consultados (figura 4.14). Esta tela mostra os dados mais impor-

Figura 4.11: Terceiro passo da elaboração de editais

tantes do edital para economizar espaço e caberem o máximo de editais, porém existe um link que leva para uma outra página onde o edital é mostrado em detalhes (figura 4.15). Nos editais que ainda não foram encerrados, já foram publicados e não foi iniciado o recebimento de propostas, aparece o link ”tenho interesse!”. Nos editais que não foram encerrados, o recebimento de propostas foi iniciado e foi dito previamente que se tinha interesse no edital, aparece o link ”fazer proposta”. Nos editais que já foram promulgado o vencedor, aparece o link ”ver promulgação”.

Se o fornecedor escolheu o link ”tenho interesse!”, ele será encaminhada para uma outra tela onde será mostrado o edital em detalhes, um termo de compromisso e uma opção para exportar a chave pública do certificado escolhido para cifrar a proposta. Antes desta tela ser aberta, uma janela aparece (figura 4.16) pedindo autorização para instalar um componente no cliente. Este componente teve de ser assinado digitalmente com um certificado especial que tem a opção de assinar código digitalmente. O que este componente faz é exportar a chave pública, que será utilizada mais tarde para cifrar a proposta, para um diretório na máquina local do cliente. Após feito isso o cliente utiliza uma chamada a uma API do Windows para enviar, via protocolo FTP, o arquivo gerado (pelo mesmo componente) com a chave pública do certificado digital do fornecedor para o servidor, colocando este arquivo na pasta específica, gerada no momento em que o comprador publicou o edital. O motivo da exportação desta chave, como foi explicado no capítulo anterior, é para cifrar cada pedaço das chaves privada dos outros fornecedores. Este componente é um ActiveX escrito na linguagem Microsoft Visual Basic 6.0, e seu código fonte acompanha no anexo.

Todos os fornecedores que assumiram o termo de responsabilidade e disseram

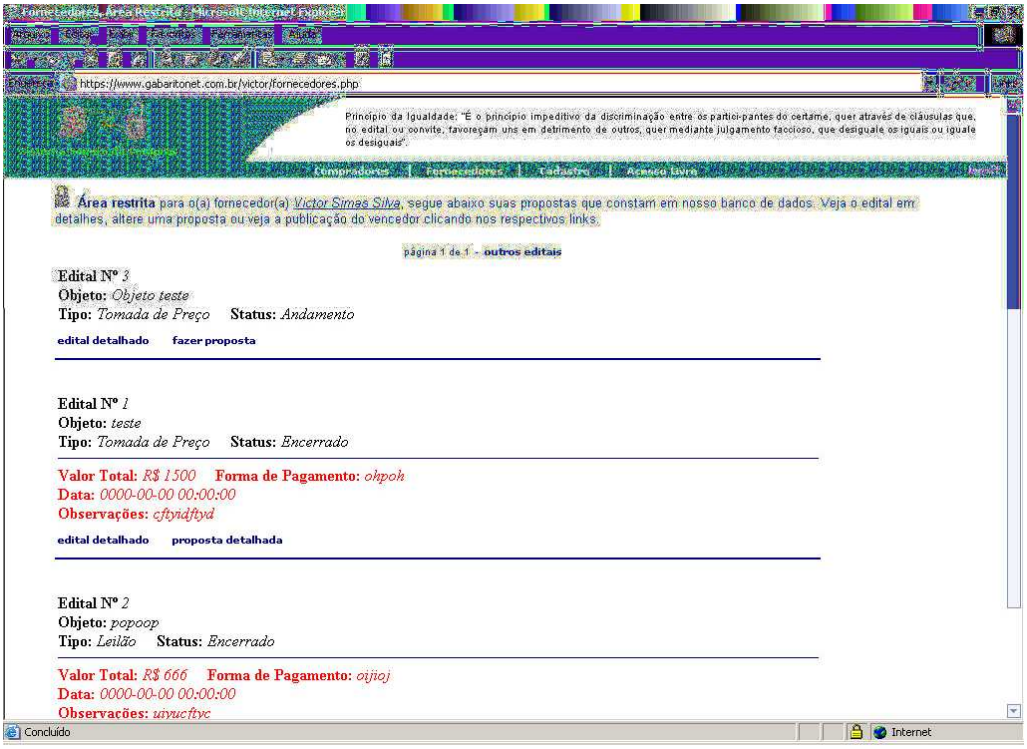

Figura 4.12: Área restrita dos fornecedores

ter interesse no edital agora devem aguardar o início do recebimento de propostas pelo fornecedor Uma vez que o comprador iniciar o recebimento das propostas, nenhum outro fornecedor poderá dizer-se interessado no edital. Agora é dado início no envio de propostas, quando em sua área restrita, o fornecedor verá o edital do qual assumiu estar interessado, além do edital em si, o fornecedor verá o link ”fazer proposta”que o levará para um formulário de preenchimento de proposta (figura 4.17). Lembrando que mesmo que o início do recebimento das propostas não tenha sido iniciado, o fornecedor poderá incluir sua proposta na base de dados com a possibilidade de fazer alterações, porém o envio para a base efetiva

poderá ser feito após o início do recebimento, após este envio nenhuma alteração será permitida. Na tela de cadastro da proposta o fornecedor deverá propor seu preço para cada ítem do edital e um valor total, que pode ser calculado através do botão ”Calcular”ou digitado manualmente caso se queira dar algum tipo de desconto. A forma de pagamento preferida deve ser preenchida assim como alguma observação adicional. Depois disso o

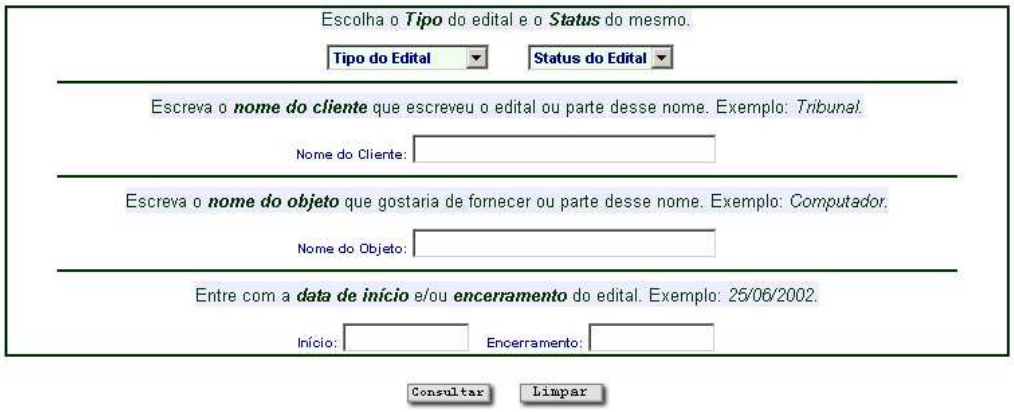

Figura 4.13: Página de consulta e editais

certificado digital que será utilizado para assinar digitalmente e cifrar a proposta deve ser escolhido dentre os mostrados na lista. Daí então o fornecedor tem a opção de ”Salvar”o documento para enviar mais tarde, ”Enviar”o documento para a base de dados sem a possibilidade de alterações futuras ou ”Carregar”um documento previamente cadastrado.

Quando o botão ”Enviar”é acionado, o mesmo componente ActiveX utilizado anteriormente para enviar as chaves públicas é acionado. Neste momento o componente fará o download de todas chaves públicas enviadas na etapa anterior a essa para um recipiente em comum a todos os fornecedores, menos a do próprio fornecedor que está enviando a proposta. Isso é feito através de uma outra chamada a mesma API do Windows que foi chamada anteriormente. O componente exporta a chave privada contida no certificado digital do fornecedor, para exportar a chave privada de um certificado digital é necessário que o usuário escolha uma senha que será utilizada mais tarde para reconstituir a chave e utilizá-la, esta senha é dividida por um algoritmo contido no componente que a quebra através do esquema de Shamir em n pedaços, sendo n o número de fornecedores participando do processo de licitação, menos um. Depois cada pedaço da chave é assinado com o certificado digital do fornecedor e cifrado com cada chave pública que foi baixada do servidor anteriormente. Feito isso, os pedaços das chaves privada são juntados em um único arquivo, que é e enviado ao servidor para uma outra pasta gerada no instante em que o edital foi publicado pelo comprador. É este componente que cifra e assina a proposta

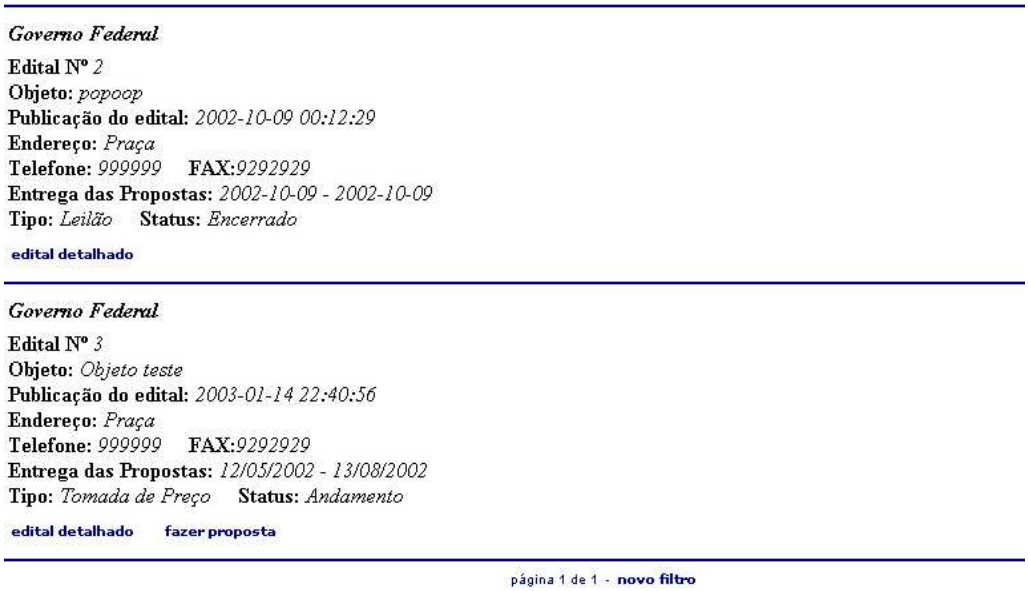

Figura 4.14: Lista de editais consultados

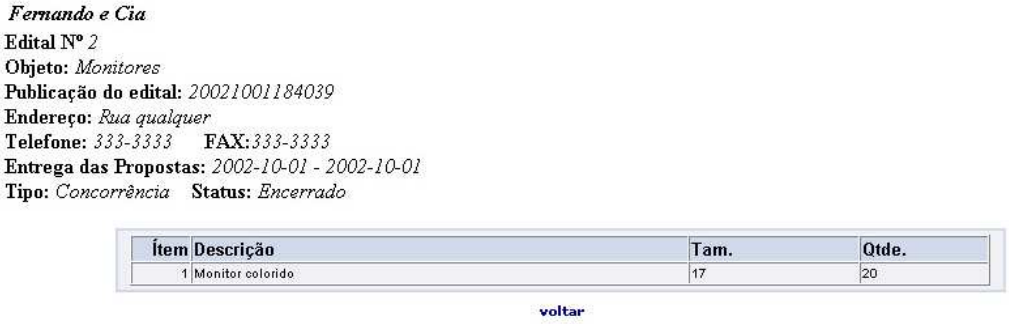

Figura 4.15: Edital detalhado

antes de ser enviada à base de dados.

No momento em que o fornecedor volta a sua página restrita, cada edital mostrado contém, além dos campos como número, descrição, tipo do edital e status e dos campos da proposta preenchida pelo próprio fornecedor (em caso de editais encerrados), links que mostram o edital de forma detalhada (”edital detalhado”); a proposta, caso exista, de forma detalhada (”proposta detalhada”); uma opção para fazer proposta, caso ainda não tenha sido feita (”fazer proposta”); uma opção para alterar ou enviar a proposta, caso ela apenas tenha sido salva e não enviada (”alterar/enviar proposta”); uma opção para excluir proposta, que entra no mesmo caso anterior (”excluir proposta”); e por último

Figura 4.16: Instalação do componente ActiveX

(”ver promulgação”) para ver a promulgação de alguma proposta a qual já encerrou-se e foi promulgada.

A opção ”edital detalhado”leva a uma página que mostra o edital por completo, com seus ítens e descrições, enquanto a opção ”proposta detalhada”, além de mostrar o edital detalhado mostra também a proposta que foi feita em cima do edital em detalhes, com o preço de cada ítem, valor total, forma de pagamento, descrição, data de inclusão, status da proposta (salva ou enviada), todos estes campos destacados em vermelho na figura 4.18.

Voltando a parte do comprador agora, em que ele verá os editais que já foram encerrados, primeiramente, quando o comprador clicar na opção ”encerrar”, além de o sistema não aceitar que mais propostas sejam enviadas, o sistema também envia aos fornecedores um email pedindo que eles enviem suas chaves privadas a fim de diminuir o esforço computacional que seria necessário para obter todas as chaves privadas através dos pedaços da mesma, que já se encontram no servidor neste momento. O email contém um link que leva para uma página (figura 4.19) onde o usuário tem as informações do edital do qual está fazendo parte e da proposta que enviou, e deve escolher o certificado digital que foi utilizado para cifrar a proposta anteriormente, após escolhido o certificado, a chave privada do mesmo é enviada para um recipiente no servidor pelo mesmo componente que foi utilizado para fazer o mesmo procedimento com a chave pública.

Depois de todas as chaves privadas serem enviadas, o comprador pode agora clicar

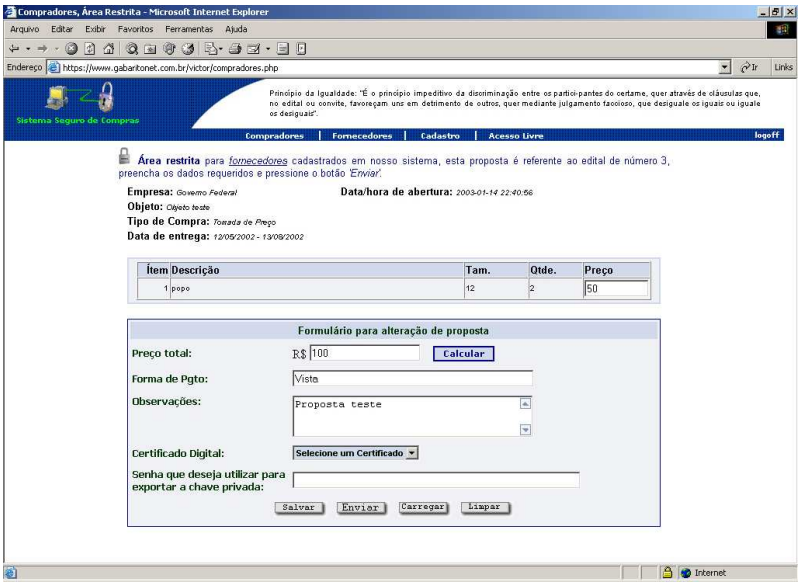

Figura 4.17: Proposta do fornecedor

na opção ”solicitar”o mesmo componente ActiveX que foi utilizado anteriormente para enviar as chaves para o servidor e dividi-la em n partes, agora é utilizado pelo comprador para decifrar cada proposta separadamente com as chaves privadas, para tal, o componente faz o download de todas as chaves privadas encontradas no servidor e dos arquivos contendo a senha para recuperar estas chaves, e as coloca em um diretório local no computador do comprador, depois recupera as chaves privadas através das senhas e decifra as propostas utilizando estas chaves que pertencem ao mesmo certificado digital, cuja chave pública foi utilizada anteriormente para cifrar as propostas.

Caso algum(ns) usuário(s) tenha(m) agido de má fé e não enviado sua chave privada no momento solicitado, se o número de chaves privadas for maior ou igual a t (do esquema de Shamir), o componente ActiveX utiliza-se do esquema de Shamir para recuperar as chaves com base nos pedaços das chaves que foram enviados previamente para o servidor cifrados com as chaves públicas de todos os fornecedores.

Completada a decifragem das propostas, o comprador visualiza uma lista de todas as propostas enviadas para o servidor (figura 4.20), após julgar a melhor proposta, declara-a vencedora. Neste momento todos os participantes recebem um email dizendo se perderam ou ganharam o edital. Se for de interesse da empresa tornar o vencedor do

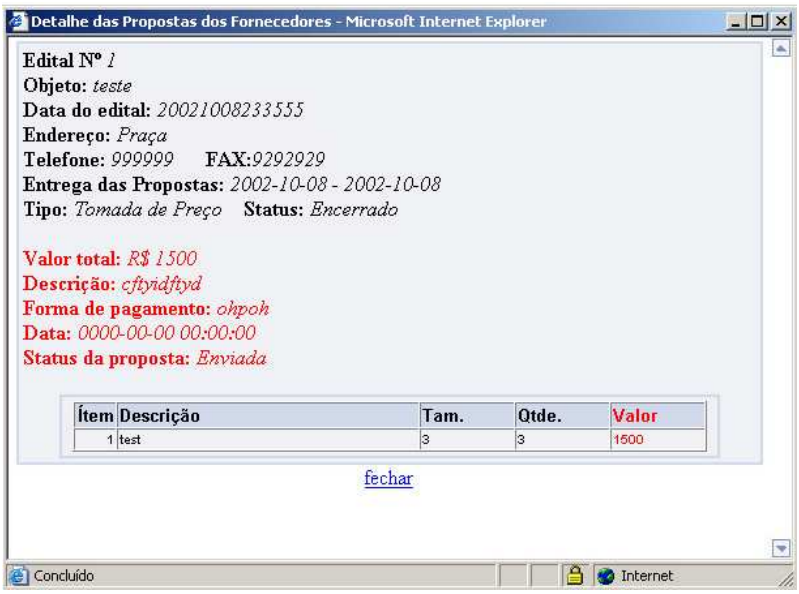

Figura 4.18: Proposta detalhada

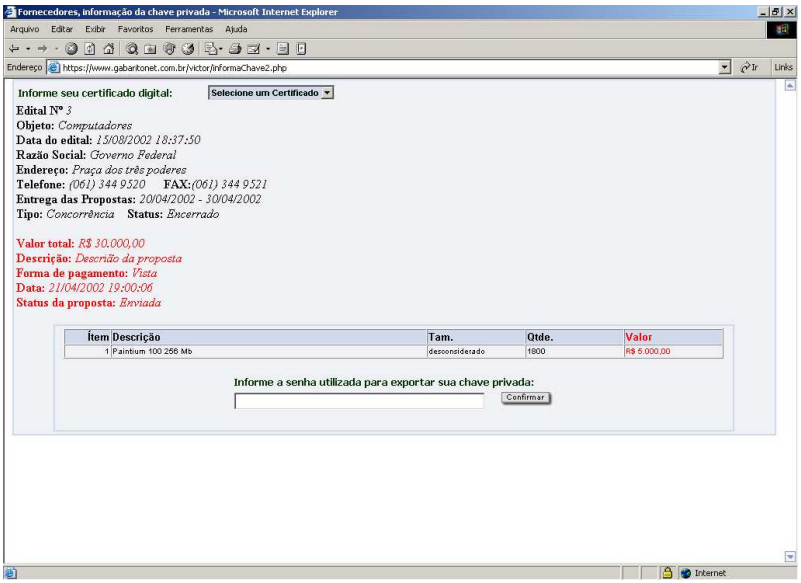

Figura 4.19: Página pedindo a senha para recuperar a chave privada

edital público em sua área restrita, o comprador tem a opção ”promulgar”, que torna o vencedor visível até para as pessoas que entrarem no sítio pelo Acesso Livre. Clicando no link ”resultado”, o comprador poderá visualizar a mesma página (figura 4.21) que todas as pessoas irão ver ao informar-se do vencedor deste edital (caso este vencedor tenha

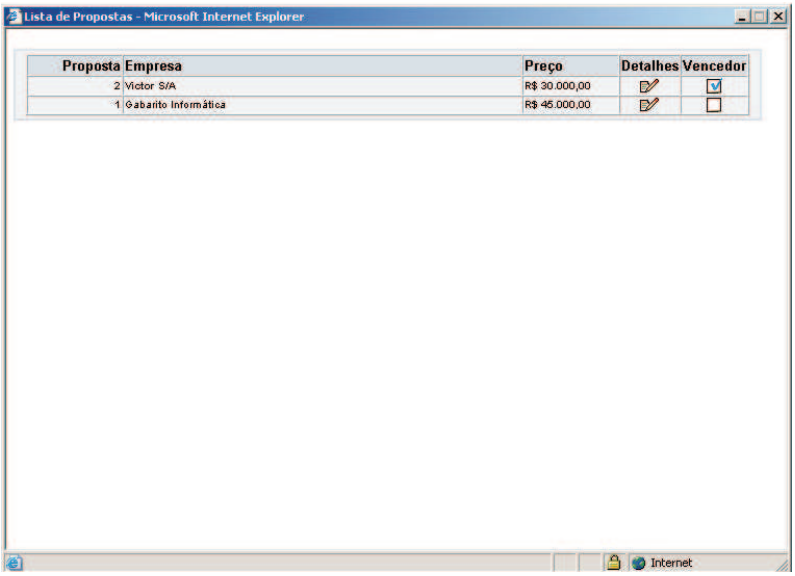

Figura 4.20: Lista das propostas

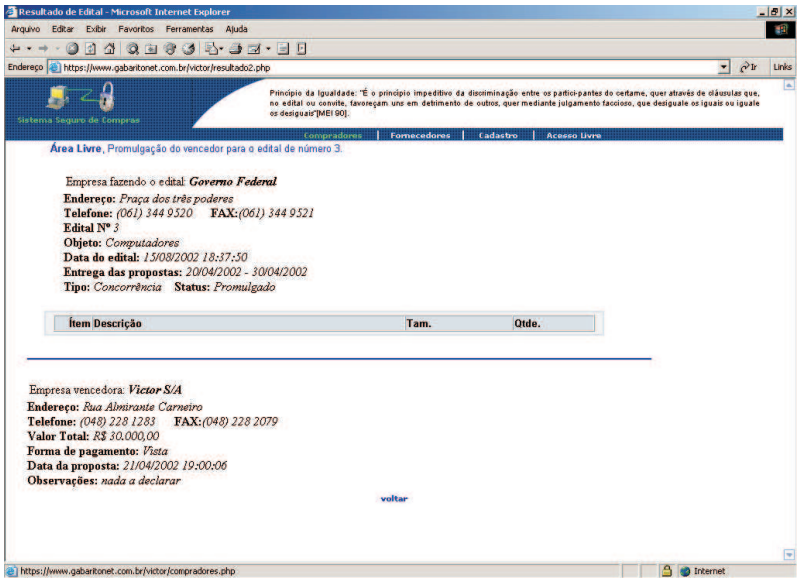

Figura 4.21: Resultado da promulgação do vencedor

sido tornado público).

Na situação dos usuários que entrarem no sítio pelo Acesso Livre, será exibida a mesma página de consulta exibida para os fornecedores, e ao filtrar sua consulta o usuário do acesso livre verá a mesma lista de editais que o fornecedor, porém não poderá dizer-se interessado em nenhuma proposta.

4.7 Conclusão

Esta foi a principal parte de nosso trabalho, onde colocamos em prática todos os conhecimentos adquiridos anteriormente em um projeto, firmando todos estes conhecimentos e adquirindo outros que só a prática pode oferecer. O projeto escolhido foi um sistema de licitações eletrônico seguro.

Primeiro realizamos um protótipo inicial para entendermos o funcionamento do processo de licitações em si, as ferramentas com as quais iríamos lidar e os princípios de segurança que melhor se aplicasse ao sistema.

Em seguida definimos e instalamos todos os pré-requisitos necessários para o funcionamento do sistema completo, como PHP, SSL, MySQL, etc.

Definimos então todas a modelagem dos dados que deveriam ser utilizados durante o processo de licitações, formando as tabelas com seus devidos campos e os relacionamentos entre si.

Partimos então para a definição do protocolo de segurança a ser utilizado na licitação eletrônica, que foi um protocolo com criptografia temporal baseado em compartilhamento de segredo, proposto em [PER 03].

Após termos um protótipo pronto, os pré-requisitos instalados, a modelagem dos dados e a definição dos protocolo de segurança, partimos para a implementação do projeto em si que resultou em um sistema bastante robusto e seguro.

Capítulo 5

Considerações Finais

A utilização da Internet para fazer comércio eletrônico é algo relativamente novo e em constante expansão, a informatização dos processos manuais que continuam existindo ainda hoje, parece ser inevitável, assim como a garantia da segurança dos dados que tramitam pela Internet durante estes processos.

Particularmente, no sistema que construímos, pudemos notar que apenas a praticidade em um processo licitatório não seria atraente o suficiente para que empresas deixassem de usar seus processos de licitação atual e passassem a utilizar o processo eletrônico, seria preciso garantir a segurança do esquema. Ao estudarmos os diversos protocolos e tecnologias de segurança nos demos conta de que a segurança em sistemas como esse não é um mero coadjuvante no processo mas sim a peça fundamental do esquema. Sabe-se que a segurança é imprescindível em processos como este, de licitação, onde os dados devem ser mantido sob completo sigilo até que o processo esteja terminado, porém conseguimos fazer algo ainda mais seguro que os procedimentos atuais. Garantimos que, sob quaisquer circunstâncias, o sigilo será mantido pelo tempo determinado, o que é sujeito a falhas no sistema atual já que as propostas são mantidas fisicamente em envelopes. Também guardamos fisicamente as propostas dos fornecedores, porém elas estão criptografadas e nem o servidor de dados nem ninguém sabe como decifrá-las até que o processo seja concluído.

Procuramos em nosso protótipo utilizar as tecnologias de programação e banco

de dados mais fáceis e práticas mantendo nosso foco sempre no aprendizado das técnicas de segurança, conseguimos então alcançar nossos objetivos que eram estudar conceitos e técnicas de segurança que melhor se aplicasse em nosso sistema que é o de licitações eletrônicas, após escolhidas as melhores técnicas conseguimos aplicá-las na prática e fazer com que funcionassem.

Esperamos que este trabalho possa vir a ser utilizado por empresas que desejem agilizar e automatizar seus processos de licitação, sempre com a garantia de segurança do processo e que outras pessoas possam utilizar-se de nossos estudos para seguir adiante com este projeto tornando-o mais comercial ou apenas utilizá-lo como ponto de referencia para outros projetos que envolvam estas técnicas de segurança.

Referências Bibliográficas

[AES 03a] AES. **National Institute of Standards and Thecnology**. www.nist.gov/des.

[AES 03b] AES. **National Institute of Standards and Thecnology**. www.nist.gov/aes.

[CHA 81] CHAUM, D. L. Untraceable electronic mail, return addresses, and digital pseudonyms. **Communications of the ACM**, [S.l.], v.24, n.2, Fevereiro, 1981.

[dB 03] DO BRASIL, G. **Infra-Estrutura de Chaves Pública Brasileira**. www.nist.gov/des.

[FED 02] FEDERAL, G. **Portal de Compras**. Disponível em <http://www.comprasnet.gov.br>. Acesso em: Setembro, 2002.

[FED 03] FEDERAL, G. **Licitações**. Disponível em <http://www.sgi.ms.gov.br/csor/paginas/licitacao.htm>. Acesso em: Janeiro, 2003.

[GS 99] GORKI STARLIN, I. A. **Microsoft Windows 2000 Server/ Advanced Server Manual Completo**. Book Express, Rio de Janeiro, 1999.

[HAR 99] HARDINGHAM, A. **Como tomar decisões acertadas**. São Paulo: Nobel, 1999.

[HIC 95] HICKMAN, K. **The SSL Protocol**. 1995.

[HUN 00] HUNT, R. **Technological infrastructure for PKI and digital certification**, p.14601471. Elsevier Science, Dezembro, 2000.

[IBA 03] IBANEZ, H. Á. M. **Norma de Segurança em Informática**. Disponível em <http://inf.unisul.br/ davi/02.htm>. Acesso em: Janeiro, 2003.

[JC 01] JESUS CASTAGNETTO, HARISH RAWAT, S. S. S. C. V. D. **Professional PHP - Programando**. Makron Books, São Paulo, 2001.

[LIN 03] LINGERFELT, J. A. **Estratégias para fazer Frente às Ameaças aos Recursos de Informática**. Disponível em <http://usinfo.state.gov/journals/itps/1198/ijpp/ip119810.htm>. Acesso em: Janeiro, 2003.

[MAC 03] MACKENZIE, D. **Aprenda Visual Basic .NET em 21 Dias**. Pearson Education do Brasil, São Paulo, 2003.

[MAN 03] **MySQLReference Manual for version 4.0.9-gamma. General Information about MySQL**. Disponível em <http://www.mysql.com/documentation/mysql/bychapter/>. Acesso em: Janeiro, 2003.

[MCD 98] MCDANIEL, R. **Como Programar em VBScript 2.0**. Makron Books, São Paulo, 1998.

[MIC 99] MICROSOFT. **Desktop Applications with Microsoft Visual Basic 6.0**, v.1, chapter9, p.297. Microsoft Press, Redmond, Washington, 1. ed., outubro, 1999.

[MIC 03a] MICROSOFT. **Microsoft Network - MSDN**. Disponível em <http://msdn.microsoft.com/library/default.asp?url=/library/en-us/security/security/cryptography-cryptoapi-and-capicom.asp>. Acesso em: Janeiro, 2003.

[MIC 03b] MICROSOFT. **Microsoft Network - MSDN**. Disponível em <http://msdn.microsoft.com/library/en-us/security/security/digital-certificates.asp>. Acesso em: Janeiro, 2003.

[MIC 03c] MICROSOFT. **Microsoft Network - MSDN**. Disponível em <http://msdn.microsoft.com/library/en-us/security/security/ca.asp>. Acesso em: Janeiro, 2003.

[MIC 03d] MICROSOFT. **Microsoft Network - MSDN**. Disponível em <http://msdn.microsoft.com/library/en-us/security/security/cryptoapi-system-architecture.asp>. Acesso em: Janeiro, 2003.

[MIC 03e] MICROSOFT. **Microsoft Network - MSDN**. Disponível em <http://msdn.microsoft.com/library/en-us/security/security/getting-ready-to-use-capicom.asp>. Acesso em: Janeiro, 2003.

[MIC 03f] MICROSOFT. **Microsoft Network - MSDN**. Disponível em <http://msdn.microsoft.com/library/en-us/security/security/managing-certificates-with-certificate-stores.asp>. Acesso em: Janeiro, 2003.

[MON 99] MONCUR, M. **Aprenda em 24 horas JavaScript 1.3**. Campus, Rio de Janeiro, 1999.

[MON 03] MONCUR, M. **The Apache Software Foundation**. Disponível em <http://httpd.apache.org/ABOUT_APACHE.html>. Acesso em: Janeiro, 2003.

[PER 03] PEREIRA, F. C. **Criptografia Temporal: Aplicação Prática em Processos de Compra**. Florianópolis: Univeridade Federal de Santa Catarina, Fevereiro, 2003. Dissertação de Mestrado.

[RJY 00] RANDY JAY YARGER, GEORGE REESE, T. K. **MySQL & mSQL**. Editora Ciência Moderna Ltda, Rio de Janeiro, 2000.

[SCH 96] SCHNEIER, B. **Applied Cryptography: Protocols, Algorithms, and Source Code in C**, chapter2. John Wiley and Sons, 1996.

[SHA 79] SHAMIR, A. **How to share a secret. Communications of the ACM**, v.22 of **11**, chapter[S.l.], p.612–613. Novembro, 1979.

[SIL 99] SILBERSCHATZ, ABRAHAM, K. H. S. **Sistema de Banco de Dados**. Makron Books, São Paulo, 1999.

[STI 02] STINSON, D. **Cryptography: Theory and Practice**. 2. ed. CRC Press, 2002.

Anexo: Artigo