|  |  |
| --- | --- |
|  |  |

UNIVERSIDADE NOVE DE JULHO UNINOVE-SP

TECNICO EM ANALISE E DESENVOLVIMENTO DE SISTEMA

**Projeto de conclusãode curso 2014**

**Aluno: Clecio Barboza**

PASSWORD SECURITY

**SÃO PAULO**

**2013/3º Semestre**

ClecioBarboza RA:

PASSWORD SECURITY

Trabalho apresentado como requisito parcial para aprovação na disciplina de Projeto em informática composta no currículo do 4º semestre de Técnico em Analise e Desenvolvimento de Sistema, Universidade Nove de Julho – SP – Campus Santo Amaro.

Orientador: Jakov Trofo Surjan

Diretoria de Exatas

**SÃO PAULO**

**2013/3º Semestre**

|  |
| --- |
| SUMÁRIO |
| 1 QuEM SOMOS 4  2 FORMA DE TRABALHO 4  3 MISSÃO 4  4 POLITÍCA DE SEGURANÇA 4  5 SEGURANÇA FÍSICA 4  6 CONTROLE DE ACESSO 5  7 LOCALIZAÇÃO DOS SERVIDORES 5  8 CONTROLE DE LIXO 5  9 ENGENHARIA SOCIAL 5  10 CONTROLE BIOMÉTRICO 5  11 MAPA DA REDE 6  12 SEGURANÇA LÓGICA 6  13 FERRAMENTAS DE MONITORAMENTO 6  14 FIREWALL 7  15 IDS 7  16 BACKUP 7  17 IPV6 7  18 FIREWALL 7  19 IPV6 7  20 HONEY POTY 8  21 DIAGRAMA DE CASO DE USO 9  22 TELA PRINCIPAL 10  23 TELA DE CADASTRO 12 |

**Quem somos**

Uma empresa que trabalha no ramo de segurança da informação, prestando consultorias para empresas.

**Forma de trabalho**

Através de ferramentas desenvolvidas pela nossa própria empresa, monitoramos diariamente as empresas parceiras, dando feedback de todas possíveis vulnerabilidades que possa existir em cada setor da empresa e onde esta sendo a área mais afetada. E RESOLVER O PROBLEMA!

**Missão**

Conscientizar as empresas da área de tecnologia, sobre os riscos de não se adotar medidas de segurança

**Política de Segurança da Informação**

Com base na norma ISO 17799, manual de boas praticas, NBR 1333, de 12/1990 controle de acesso físico a CPDs (Centro de Processamento de Dados), NBR 1334, de 12/1990 critérios de segurança física para armazenamentos de dados, NBR 1335, de 07/1991 segurança física de microcomputadores e terminais em estações de trabalho de segurança vamos implantar a segurança da informação na Password Security para a proteção do CPD e para cada local da empresa, também definimos diversas restrições de acesso tanto dentro da empresa quanto nos sistemas que possuímos, as senhas terão o prazo de validade de 30 dias e terá que ter no mínimo 10 caracteres incluindo letras, números e um caractere especial.

**Segurança física**

**Planta da empresa**



**Controle de acesso**

Com base nas informações da planta baixa decidimos que cada funcionário terá seu próprio crachá com identificação para poder entrar na empresa e para usar algumas dependências.

No CPD existem dois métodos de identificação, biometria e uma senha que será fornecida, quem tem acesso são somente funcionários de infraestrutura e o dono da empresa.

Em pontos estratégicos foi colocado câmeras de vigilância e vão ser vigiadas 24 horas por dia.

**Localização dos Servidores**

**Teremos dois servidores**:

* O servidor local será em uma sala separada restrita com controle de acesso via cartão eletrônico e impressão digital.
* Terá um datacenter em um local distante da empresa

**Controle do lixo**

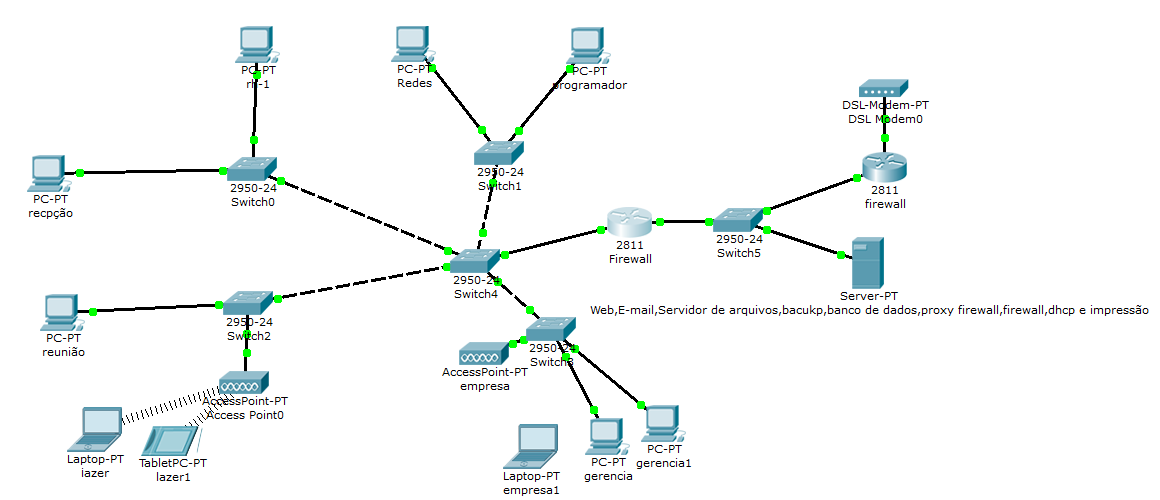
Todo e qualquer lixo da empresa será triturado e incinerado para que as pessoas não autorizadas e engenheiros sociais não tenham acesso á informação indevida.

**Engenharia social**

Com o intuito de minimizar definitivamente os ataques de engenheiro sociais os funcionários recebem aulas, palestras com orientações sobre esses tipos de ataque e como agir caso isso aconteça.

**Controle biométrico**

Pensando em segurança em cada setor da empresa vamos utilizar controle biométrico via impressão digital, com um banco de dados com informações de cada funcionário, os funcionários somente poderão adentrar no setor correspondente, afim de ter um controle de acesso.

**Mapa da rede**

Com informações levantadas sobre o mapa da rede vamos implementar um servidor com firewall roteado e o programa squid para bloqueios de todos os computadores.

Os funcionários que usarem os computadores da rede terão seu login e senha que será fornecido pelo pessoal de infraestrutura também com um prazo de trinta dias para a troca de senha.

**Segurança lógica**

Na Password Security vamos utilizar oito servidores:

1. Servidor Web
2. Servidor de E-mail
3. Servidor de Arquivos
4. Servidor de Backup
5. Servidor de Banco de Dados Oracle
6. Servidor Proxy e Firewall
7. Servidor Firewall
8. Servidor DHCP e Impressão

**Ferramenta para monitoramento**

Para que pudéssemos monitorar os servidores e para ver se os servidores estão funcionando corretamente utilizamos uma ferramenta chamada Nagios que traz em tempo real os servidores, localização de cada um e o sistema operacional.

**Firewall**

Temos dois servidores Firewall para bloqueio dos computadores e da rede em si o primeiro firewall é os acessos que vem da internet onde tem um controle do que entra e sai o segundo Firewall vai atuar com o Squid para bloqueio de sites indevido em que o usuário não tem direito de baixar tudo da internet e sim consultar para ajudar a empresa a detectar tipos de falha em que a empresa precisa para prestar consultoria.

**Ids sistema de detecção de intruso**

Visando proteger a empresa vamos implantar em todos os servidores o sistema de detecção de intruso que vai funcionar em conjunto com o Honeypot e o Firewall para analisar os pacotes que passam pelo firewall e identificar os que são normais ou suspeitos.

Temos dois tipos de ids os HIDS e NIDS:

* Hids esse tipo de IDS monitora o tráfego do segmento de rede. Em geral com a interface de rede no modo promíscuo – como um Sniffer.
* Nids esse tipo de IDS procura por indícios de intrusão na máquina onde está instalado. Em geral, com base em informações de arquivos de log do sistema operacional eles procuram por atividades que não são comuns, como tentativas de login ou tentativas de alteração em privilégios do sistema.
* O software utilizado para essa ação vai ser o SNORT uma ferramenta gratuita baseada em Linux ele possui três módulos de trabalho.
* SNIFFER:
* PACKET LOGGER.

**Backup**

Será feito o backup diariamente de todos os arquivos para que seja restaurado quando houver algum tipo de problema. Junto com o backup utilizaremos uma ferramenta Raid porque quando tivesse uma queda de um dos nossos servidores ele já subisse automaticamente sem que haja qualquer perda nos dados ele vai funcionar como espelho dos servidores.

**Ipv6**

Para ataques do tipo baseado em redes vamos implementar o IPV6 um protocolo seguro para as conexões .

Um endereço padrão IPv6 deve ser formado por um campo provider ID, subscribe ID, subnet ID e node ID. O node ID (ou identificador de interface) deve ter 64bits, e pode ser formado a partir do endereço físico (MAC) no formato EUI 64.

Os endereços IPv6 são normalmente escritos como oito grupos de 4 dígitos hexadecimais.

2001:0db8:85a3:08d3:1319:8a2e:0370:7344.

Se um grupo de vários dígitos seguidos for 0000, pode ser omitido.

2001:0db8:85a3:0000:0000:0000:0000:7344 é o mesmo endereço IPv6 que:

2001:0db8:85a3::7344.

• unicast - cada endereço corresponde a uma interface (dispositivo).

• multicast - cada endereço corresponde a múltiplas interfaces. É enviada uma cópia para cada interface.

• anycast - corresponde a múltiplas interfaces que partilham um prefixo comum. Um datagrama é enviado para um dos dispositivos, por exemplo, o mais próximo.

Segurança: O IPv6 inclui a seguridade nas próprias especificações como a encriptação da informação e a autenticação do que envia a referida informação.

Aplicações de tempo-real: Para dar melhor suporte ao tráfego de tempo-real (i.e. videoconferência), o IPv6 inclui o rotulado dos fluxos nas suas especificações. Com este mecanismo os routers podem reconhecer a qual fluxo pertence os pacotes transmitidos.

**Honeypot**

Com o intuito de ver como os Hackers trabalham foram colocado o Honeypot que nada mais é que um sistema para fazer uma copia de todos os servidores.

A maioria dos servidores de HONEYPOT não possui serviço ativo, caso algum intruso tente invadir a rede ou ter acesso de forma remota aos dados, ficaremos sabendo pelo processamento do servidor.

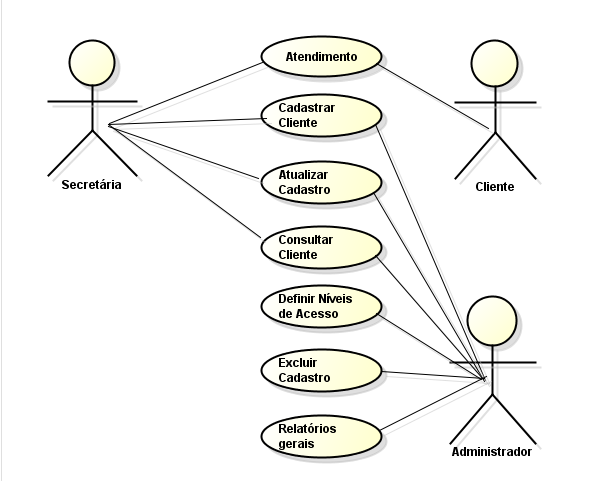
**Vpn**

Para que os funcionários tenham acesso á rede da empresa foram implementado uma infraestrutura VPN para que em qualquer parte do mundo os funcionários tenham acesso á rede da empresa e também para fazer os backups em um servidor externo onde as conexões são criptografadas usando a internet.

**Descrição do Projeto**

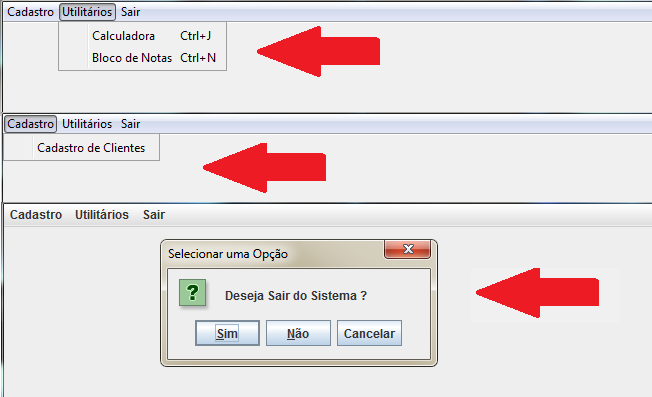
1. **Diagrama de Caso de Uso**

* Pensando em melhoria do sistema definimos o modelo em qual foi desenvolvido.



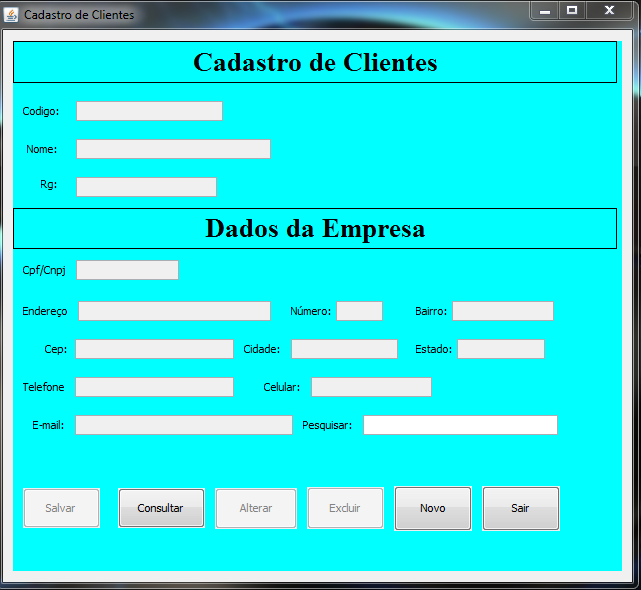
1. **Tela Principal do Sistema**

* O usuário vai poder navegar pelos menus onde vai poder:
* Cadastrar clientes
* Utilizar a calculadora
* Utiliza o bloco de notas
* Sair do sistema



1. **Tela de cadastro de clientes**

* O Usuário vai poder:
* Cadastrar
* Alterar
* Consultar
* Excluir
* E sair do sistema



1. Codificação do Sistema

* Utilizamos os Métodos “GET,SET”.
* Melhor organização do código
* Dar referência a um determinado objeto
* Para encapsular os dados e atribuir a uma outra classe
* Código:

public class ModeloCadastro {

private int cod;

private String nome;

private int rg;

private int cpf;

private String endereco;

private int numero;

private String bairro;

private int cep;

private String cidade;

private String estado;

private int telefone;

private int celular;

private String email;

private String pesquisa;

public int getCod() {

return cod;

} public void setCod(int cod) {

this.cod = cod;

}

public String getNome() {

return nome;

}

public void setNome(String nome) {

this.nome = nome;

}

public int getRg() {

return rg;

}

public void setRg(int rg) {

this.rg = rg;

}

public float getCpf() {

return cpf;

}

public void setCpf(int cpf) {

this.cpf = cpf;

}

public String getEndereco() {

return endereco;

}

public void setEndereco(String endereco) {

this.endereco = endereco;

}

public int getNumero() {

return numero;

}

public void setNumero(int numero) {

this.numero = numero; }

public String getBairro() {

return bairro;

}

public void setBairro(String bairro) {

this.bairro = bairro;

}

public float getCep() {

return cep;

}

public void setCep(int cep) {

this.cep = cep;

}

public String getCidade() {

return cidade;

}

public void setCidade(String cidade) {

this.cidade = cidade;

}

public String getEstado() {

return estado;

}

public void setEstado(String estado) {

this.estado = estado;

}

public float getTelefone() {

return telefone;

}

public void setTelefone(int telefone) {

this.telefone = telefone; }

public float getCelular() {

return celular;

}

public void setCelular(int celular) {

this.celular = celular;

}

public String getEmail() {

return email;

}

public void setEmail(String email) {

this.email = email;

}

public String getPesquisa() {

return pesquisa;

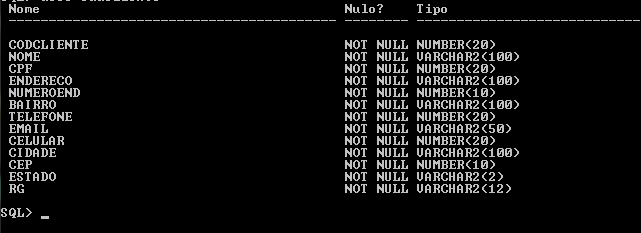
}

public void setPesquisa(String pesquisa) {

this.pesquisa = pesquisa;

} }

1. Banco de dado



* String de Conexão

import java.sql.\*;

import javax.swing.JOptionPane;

public class ConectaBanco {

public Statement stm;

public ResultSet rs;

private String driver = "oracle.jdbc.OracleDriver";

private String caminho = "jdbc:oracle:thin:@127.0.0.1:1521:XE";

private String usuario = "system";

private String senha = "senha";

public Connection conn;

public void conexao() {

try {

System.setProperty("jdbc.Drivers", driver);

conn = DriverManager.getConnection(caminho, usuario, senha);

//JOptionPane.showMessageDialog(null, "Conectado com sucesso!");

} catch (SQLException ex) {

JOptionPane.showMessageDialog(null, "Erro de conexão!\n Erro: " +ex.getMessage());

}

}

public void executaSQL(String sql){

try {

stm = conn.createStatement(rs.TYPE\_SCROLL\_INSENSITIVE,rs.CONCUR\_READ\_ONLY);

rs=stm.executeQuery(sql);

} catch (SQLException ex) {

} }

public void desconecta(){

try {

conn.close();

//JOptionPane.showMessageDialog(null, "Desconectado com sucesso!");

} catch (SQLException ex) {

JOptionPane.showMessageDialog(null, "Erro ao fechar a conexão!\n Erro:"+ex.getMessage());

}

}

}