Awdrey Vieira Vilela

Estudos de Técnicas de Detecção e Prevenção de Intrusos

Monografia de Graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências da disciplina Projeto Orientado para obtenção do título de Bacharel em Ciência da Computação.

Orientador
Prof. Joaquim Quinteiro Uchôa

Lavras
Minas Gerais - Brasil
2001

Awdrey Vieira Vilela

Estudos de Técnicas de Detecção e Prevenção de Intrusos

Monografia de Graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências da disciplina Projeto Orientado para obtenção do título de Bacharel em Ciência da Computação.

Aprovada em *29 de Julho de 2001*

Prof. Luiz Henrique Andrade Correia

Prof. Jones Oliveira Albuquerque

Prof. Joaquim Quinteiro Uchôa
(Orientador)

Lavras
Minas Gerais - Brasil

Dedico principalmente a Deus,meus pais,
dedico também a meus irmãos que sempre me apoiaram,
demais parentes que nunca me deixaram desanimar.

Agradecimentos

Agradeço este trabalho aos colegas que me ajudaram nesta tarefa,
aos meus pais por me darem condições de estar aqui,
ao professor Joaquim por ter a paciência de ser meu orientador,
colegas de república pelo companherismo
e em especial a Pâmera pela compreensão!

Resumo

Um computador que precisa de segurança deve se precaver com vários mecanismos de controle de acesso. Sistemas de Detecção de Intrusos é um dos mecanismos usados hoje em dia.

Detecção de Intrusos tem como objetivo principal reagir a uma invasão ou tentativa de invasão.

Mecanismos de Detecção de Intrusos emitem alerta assim que detectam alguma anomalia. A Detecção de Intrusos hoje é um mecanismo indispensável a várias empresas

Lista de Figuras

Capítulo 1

Introdução

Detecção de intrusos é a prática de usar ferramentas automatizadas e inteligentes para descobrir tentativas de intrusão em tempo real. Tais ferramentas são denominadas freqüentemente de Sistema de Detecção de Intruso. Como comentado em [Weber (2000)]:

> Técnicas de Detecção de Intrusos se aproximam bastante daquelas usadas em *Firewalls* e sistemas de Log, e o seu objetivo principal é reagir a uma invasão (ou suspeita de invasão) no menor intervalo de tempo possível. Isto pode ser feito, por exemplo, monitorando-se continuamente o tráfego de rede, à procura de qualquer anomalia, ou então analisando-se continuamente as últimas entradas dos arquivos de log, à procura de ações suspeitas.

Para promover a segurança necessária em sistemas computacionais são usados mecanismos de controle de acessos. Entretanto, se esses mecanismos forem burlados um acesso não autorizado pode ser feito por um invasor. Uma das formas de impedir este invasor é detectando sua presença e conseguindo fechar seus caminhos a tempo.

A complexidade nos métodos usados para se quebrar a segurança dos sistemas tradicionais tem levado as empresas a uma busca sem precedentes por implementar soluções eficazes de medidas de precaução para atender às suas necessidades de segurança. Entretanto, mesmo um exímio profissional de segurança não poderia predizer as ações de um invasor, nem poderia antecipar as falhas de sistemas, que ainda não foram exploradas ou descobertas.

Falhas na segurança podem levar a prejuízos incalculáveis. Apesar da evolução de segurança de computadores e engenharia de software, a maioria dos sistemas

de computadores ainda possuem vulnerabilidades que permitem ataques. Este problema aumentou com a proliferação das redes de computadores, pois os sistemas agora podem ser acessados por qualquer pessoa na rede.

Com o uso de detecção de intrusos uma vez detectada alguma modificação ou alteração de estado nos objetos sob monitoramento, imediatamente é notificados a ocorrência por meio de alerta. Esse alerta leva o administrador a se precaver contra uma possível invasão (ou tentativa), não tendo assim grandes prejuízos com o invasor em questão.

Uma breve navegada pela internet pode mostrar que nos últimos anos tem se desenvolvido um número grande de ferramentas de segurança (entre as quais as de detecção de intruso) para Windows e UNIX, boa parte disponível livremente na internet. Ferramentas estas possíveis de serem estudadas e comparadas podendo assim escolher uma melhor opção para o seu interesse.

Este trabalho tem por objetivo estudar e comparar aplicativos de detecção de intruso. Ele encontra-se dividido da seguinte forma: o Capítulo 2 apresenta uma breve noção sobre segurança em redes, qual a necessidade de seu uso e alguns pontos para se ter uma boa segurança. No Capítulo 3 fala-se mais exclusivamente sobre detecção de intruso, apresentando aplicativos, analisando alguns e mostrando como instalá-los e gerênciá-los, bem como suas respostas. A resposta do Tripwire e do AIDE, devido seu tamanho, vem nos apêndices A e B, respectivamente.

Uma observação a ser feita aqui é que o tempo para o desenvolvimento do mesmo foi insuficiente para os resultados esperados. Assim o projeto deixa vários pontos em aberto, não apresentando todas as conclusões de interesse no projeto. Entretanto os resultados obtidos permitem facilmente que ele seja enxergado como um ponto razoavelmente adiantado para um melhor desenvolvimento futuro.

Capítulo 2

Segurança em Redes

O termo *segurança* é freqüentemente usado com o significado de minimizar a vulnerabilidade de bens (qualquer coisa de valor) e recursos. A *segurança* está relacionada à necessidade de proteção de suas informações confidenciais de elementos não autorizados, ou seja, está relacionada à proteção contra o acesso ou manipulação, intencional ou não, de informações por elementos não autorizados.

A necessidade de proteção deve ser definida em termos das possíveis ameaças e riscos e dos objetivos de uma organização, formalizados nos termos de uma política de segurança. Isso é extremamente válido em redes de computadores. Nesse caso, constituem pontos para uma boa segurança em redes:

- uma boa política de segurança, somente através da política de segurança pode-se ter uma idéia da necessidade de segurança da organização;
- um *firewall* bem instalado e configurado, onde os filtros serão melhor implementados;
- uma boa criptografia nos dados cruciais, já que dados criptografados são difíceis de serem quebrados;
- um bom detector de intruso instalado e bem configurado, talvez até dois, onde se usa um passivo e um ativo, cujo o ativo tenta detectar até mesmo uma tentativa de invasão e o passivo caso tenha sido iniciado a invasão.

Este capítulo apresentará estes métodos, salvo detecção de intruso que é apresentado em um capitulo próprio, já que é o objetivo deste trabalho.

2.1 Política de Segurança

Uma *política de segurança* é um conjunto de leis, regras e práticas que regulam como uma organização gerencia, protege e distribui suas informações e recursos. A política de segurança define o que é, e o que não é permitido em termos de segurança, durante a operação de um dado sistema.

Portanto um dado sistema é considerado seguro em relação a uma política de segurança, caso garanta o comprimento das leis, regras e práticas definidas nessa política. Uma organização sem uma política de segurança fica inviabilizada de ter uma boa segurança, sendo portanto o elemento fundamental em segurança de redes. A organização deve seguir normas e regras regulamentadas através da política de segurança as quais permitem uma boa implantação da mesma.

Uma política de segurança deve incluir regras detalhadas definindo como as informações e recursos da organização devem ser manipulados ao longo do seu ciclo de vida, ou seja, desde o momento que passam a existir no contexto da organização até quando deixam de existir. A política de segurança define o que é, e o que não é permitido em termos de segurança, durante a operação de um dado sistema. Uma política de segurança pode ser classificada da seguinte forma:

- *Política de Segurança baseada em regras*: Apóia-se em informações sobre sensibilidade, ou seja, em um sistema seguro. Os dados ou recursos devem ser marcados com rótulos de segurança que indicam seu nível de sensibilidade,

- *Política de Segurança baseada na identidade*: Representa o tipo de controle de acesso mais encontrado nos computadores atuais. A base desse tipo de segurança é que um indivíduo, ou processo operando sob seu controle, pode especificar explicitamente os tipos de acesso que outros indivíduos podem ter às informações e recursos sob seu controle.

Uma política de segurança define, portanto, uma *política de uso*. A política de uso é o documento (ou um conjunto de documentos) que diz ao usuário o que pode e o que não pode fazer. Ou seja, se uma determinada ação pode ser realizada ou em caso de realizá-la está infringindo as normas. Um exemplo de restrição em uma política de uso é a definição de qual conteúdo pode ser acessado pelo usuário, ou seja por qual página pode-se navegar.

2.2 Firewalls

Um *firewall* é um conjunto de componentes colocados entre duas redes e que coletivamente implementam uma barreira de segurança. Um *firewall* pode ser considerado um *check point* (local de checagem de passagem de um determinado objeto, exemplo em um rally local que checa passagem do carro em um determinado local) pois todo o tráfego entre as redes interna e externa deve passar por ele. Assim, é um ótimo local onde aplicar a política de segurança. Por exemplo, se a política de segurança impede o uso de FTP para o exterior, todos os pedidos de conexão com servidores FTP externos serão filtrados pelo *firewall* e não serão passados à Internet. Como observado em [Weber (2000)]

> A idéia básica de um firewall é a monitoração constante das atividades de rede através da análise do tráfego IP e, a partir de uma série de regras, a decisão se o trafego deve ser permitido ou bloqueado.

O objetivo básico da implementação de um *firewall* é defender a organização de ataques externos. Como efeito secundário, ele também pode ser utilizado para o uso de recursos externos pelos usuários internos. Entretanto um *firewall* não pode proteger contra usuários internos. Qualquer forma de ataque proveniente da própria rede interna não pode ser detectada pelo *firewall*. A razão é óbvia: o fluxo de dados de um ponto da rede interna para outro ponto da rede interna não passa pelo *firewall*, como pode ser visto na figura 2.1.

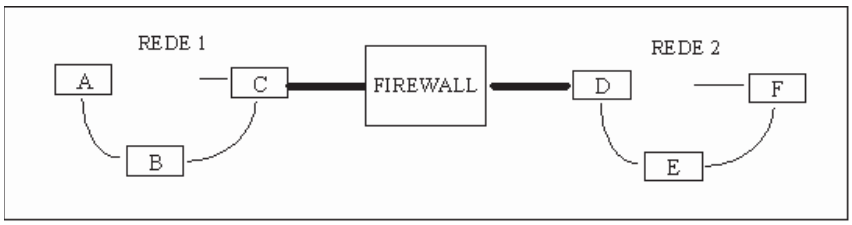

Figura 2.1: Ilustração de um Firewall

Um *firewall* pode ser configurado mais rigorosamente, dependendo da necessidade da rede. Pode-se, por exemplo, configurá-lo para não aceitar comunicações somente para determinados serviços (ftp, por exemplo) ou simplesmente não aceitar nenhuma comunicação. Neste caso uma máquina da sub-rede deverá estabelecer esta comunicação e nunca uma de fora. Vemos então que um *firewall* mal configurado pode ser um grande transtorno para os usuários legítimos.

Existem várias formas de se construir um *firewall*. Talvez a mais simples seja apenas um filtro de pacotes implementado em uma máquina com duas placas de rede. Ele simplesmente verifica a origem e o destino dos pacotes não aceitando conexões de determinadas máquinas ou então só aceitando de algumas, ou ainda, só aceitando para determinados serviços. Um *firewall* será inútil se existirem outras entradas para a rede, como por exemplo, através de *modems* [Schlemer (2001)].

Entre os *firewalls* mais utilizados podem ser citados *iptable*s e *ipchains*, que são implementações de *firewall* em Linux. Entanto existem várias outras implementações disponíveis na internet.

2.3 Criptografia

Criptografia é um método que modifica o texto original da mensagem a ser transmitida (texto normal), gerando texto criptografado na origem, através de um processo de codificação definido por um método de criptografia, ou seja é escrever texto através de códigos conhecidos apenas pelos interessados.

A Criptografia tem suas origens há muitos anos. Comenta-se que o imperador romano Júlio César teria sido o primeiro a empregá-la quando enviava cartas criptografadas, pois não confiava no mensageiro e havia o risco dele ser capturado, no caso de uma guerra. O método utilizado por César era simples: ele rescrevia a carta somando 3 a posição da letra, ou seja, o "A" (1) passaria a ser "D" (4), o "B" (2) "E" (5) e assim sucessivamente, imaginado as letras dispostas em círculo, ou seja, a lista não termina no "Z" mas retorna ao "A" novamente [Schlemer (2001)].

O texto (ou a mensagem) criptografado deve então ser transmitido e, no destino, o processo inverso deve ocorrer, isto é, o método de criptografia é aplicado agora para decodificar o texto criptografado transformando-o no texto normal original.

O método de criptografia usado por Júlio César é facilmente quebrado por isso foram desenvolvidos outros métodos. Alguns tipos de criptografia usados atualmente são:

- *Criptografia com chave secreta* - consiste em substituir as letras de uma mensagem pela terceira letra após sua posição no alfabeto (considerando o a como sucessor do z). Uma generalização desse método seria substituir as letras pela n-ésima letra após sua posição no alfabeto,

- *Criptografia com chave pública* - baseia-se na utilização de chaves distintas: uma para a codificação (E) e outra para a decodificação (D), escolhidas de

forma que a derivação de D a partir de E seja, em termos práticos, senão impossível, pelo menos muito difícil de ser realizada.

2.4 Comentários Finais

Este capítulo apresentou noções de segurança e métodos que deve-se usar para implementar uma segurança em uma rede. Mostrou que com uma boa política de segurança pode-se conseguir uma segurança satisfatória para uma determinada organização.

Capítulo 3

Detecção de Intrusos

3.1 Comentários iniciais

A maioria das informações no mundo atual estão guardadas em computadores e por sua vez a maioria desses computadores estão ligados à grande rede mundial. Tais computadores estão cada vez mais precisando de segurança para que suas informações não sejam vistas por outras pessoas e para isto necessita-se de segurança.

A detecção de intrusos é um dos método usados hoje em dia. Desde que combinado com outros modos de segurança (e.g.: criptografia) imprime grande segurança. Para isso, entretanto, necessita-se saber qual aplicativo é melhor e porque é melhor para aquela situação.

A metodologia de detecção de intrusão tem como objetivo detectar atividades que violem a política de segurança ou comprometam a segurança do sistema. Como nenhum sistema de segurança é invulnerável, a detecção de intrusos apresenta-se como uma linha defensiva alternativa dos sistemas relativamente a falhas no processo de autenticação dos utilizadores. A implementação de mecanismos de detecção de intrusos e registos de actividades apresenta muitas vantagens:

- ao ser detectado o intruso pode ter a sua sessão terminada, evitando ou pelo menos reduzindo os danos provocados;

- ao ser do conhecimento geral a existência de detecção de intrusos e registro de atividades, tal funciona como um fator dissuasor, quer para intrusos quer para os utilizadores lícitos do sistema;

- depois da ocorrência de um ataque torna-se possível analizar a técnica usada e toma-se medidas adequadas para evitar a mesma situação no futuro;
- mediante algum tipo de situação anormal reportada por um utilizador é possivel analizar a ocorrência e chegar aos responsáveis;
- permite detectar situações de violação de privacidade, sem registros de atividade esta situação poderia não ser detectada.

Os atacantes têm perfeita consciência de todos estes fatores, se conseguirem. Uma das preocupações que vão ter é terminar o sistema de registro de atividades e remover as respectivas bases de dados, depois podem trabalhar mais "à vontade". É atraves dos registros de atividades que os administradores vão tentar evitar um ataque do mesmo tipo e vão tentar identificar o intruso.

A classificação dos aplicativos de detecção de intrusos pode ser feita de várias maneiras, neste trabalho será abordada uma onde se classifica detecção de intrusos em:

1. *Detecção Passiva ou Prevenção de Intrusos*: ferramentas que analisam ações que possam ser prenúncios de tentativas de invasões. Essa visão baseia-se na idéia que, em um servidor relativamente seguro, a invasão geralmente não é instantânea: ela é fruto de uma série de tentativas. Exemplos analisados neste trabalho são Tripwire [Tripwire (2001)] e AIDE [AIDE (2001)].
2. *Detecção Ativa ou Detecção de Intrusos*: ferramentas que analisam ações que possam ser tentativas efetivas de invasões. Nesse caso, o sistema deve tomar decisões imediatas sobre qual ação tomar. O exemplo analisado neste trabalho é o Snort [Forster (2001)]. Outros aplicativos foram analisados apenas parcialmente devido, principalmente à problemas de instalação.

Observe que todo o projeto foi desenvolvido rodando os aplicativos sobre o sistema operacional Linux Red Hat 7.1.

3.2 Detecção Passiva de Intrusos

3.2.1 Tripwire

O tripwire pode ser adquirido em `http://www.tripwire.com` ou então em `http://www.sourceforge.net/projects/tripwire/`. É um sistema de detecção de intrusos passivo que tem como pontos positivos usar senha para

que o administrador tenha acesso a seu banco de dados, ser de fácil instalação e atualmente ser gratuito (salvo que não era no passado). Tem como pontos negativos ser lento e gastar muito tempo para ser configurado, visto que seu arquivo de configuração deve ser especificado arquivo por arquivo.

Instalando o Tripwire

- Inicialmente obtenha o arquivo, em formato rpm de um dos dois sites citados.
- Instale o Tripwire com o comando
  `rpm -Uvh` tripwire.rpm
- Configure o tripwire para o sistema atual, digitando
  `/etc/tripwire/twinstall.sh`
  O Tripwire irá pedir várias frases de senha, começando com sua frase de senha de arquivo chave, e após inserir sua frase de senha será pedido confirmação. Depois pedirá a frase de senha de chave local e novamente a confirmação. Por fim pedirá a frase de senha do site. Terminando, informará que a instalação foi bem sucedida.

Utilizando o tripwire

Antes de utilizá-lo precisa personalizar dois arquivos:

1. O arquivo de configuração do Tripwire.
2. O arquivo de diretivas do tripwire.

O arquivo de configuração é o twcfg.txt, este arquivo armazena informações específicas do sistema, este arquivo encontra-se em `/etc/tripwire/`. O arquivo de diretivas é o `twpol.txt`, este arquivo armazena a especificação de quais objetos (arquivos, diretórios e assim por diante) o Tripwire deve monitorar bem como suas localizações.

Para configurar e executar o tripwire, vá para o diretório `/etc/tripwire/` e emita o comando

`twadmin --create-cfgfile -site-keyfile site.key twcfg.txt`

Em resposta ele vai te pedir sua frase de senha. Agora é necessário atualizar o arquivo de diretiva para isto digite

`twadmin --create-polfile twpol.txt`

Novamente irá te pedir a frase de senha.

Agora deve gerar seu banco de dados para isto digite

`tripwire --init`

Aqui também será pedido a frase de senha. O que acontece a seguir depende da configuração do seu sistema. Se você não edita adequadamente o arquivo de diretiva, pode-se ver vários erros. Anote-os para que você corrija mais tarde mudando as regras do arquivo de diretiva. Estes erros são os arquivos que foram configurados um a um no arquivo de diretiva.

Por fim, para verificar a integridade do arquivo emita o comando

`tripwire --check`

Em resposta o tripwire faz uma varredura no seu sistema e te devolve os resultados. Um exemplo de relatório de checagem pode ser verificado no Apêndice A.

3.2.2 AIDE

O AIDE (Advanced Intrusion Detection Enviroment) pode ser adquirido no site `http://www.cs.tut.fi/~rammer/aide.html`. Ele é um sistema de detecção de intrusos passivo que tem como pontos positivos ser mais rápido para fazer verificação que o Tripwire, ser gratuito (já que foi desenvolvido para ser a versão gratuita do Tripwire), ser de fácil instalação, e já vir previamente configurado.

Ele tem como ponto negativo não assinar seu banco de dados (não ter uma senha para acesso ao banco de dados), deficiência esta que é suprida executando o banco de dados por meio de disco remóvivel e protegido contra gravação.

Instalação

- Para instalar primeiro adquira o aplicativo em formato rpm em seu site.
- Instale o arquivo com o comando `rpm -Uvh` AIDE.rpm

Configurando

O arquivo de configuração do AIDE é `/etc/aide.conf`, cujo conteúdo *default* deve ser suficiente para maioria das aplicações.

Para gerar o banco de dados, execute os comandos:

`/usr/bin/aide -i`

`mv /var/aide/aide.db.new /var/aide/aide.db`.

Logo após execute o comando
`/usr/sbin/aide-md5` [dispositivo de boot]
Exemplo:
`/usr/bin/aide-md5 /dev/hda`,
ou
`/usr/bin/aide-md5 /dev/floppy`.
Para verificar a integridade do sistema, execute o próprio AIDE, desta forma:
`/usr/bin/aide -C`.

Os arquivos que sofrerem qualquer mudança, seja no tamanho, conteúdo, permissão ou data de criação serão listados. Um exemplo de relatório é mostrado no Apêndice B. Para verificar a integridade do próprio AIDE, deve-se executar novamente o programa aide-md5, como no exemplo:
`/mnt/floppy/aide-md5 /dev/hda`.

Se algum dos códigos MD5 não estiver igual com aqueles gerados anteriormente, o(s) respectivo(s) componentes podem estar comprometidos, e isto é um problema **muito** sério.

3.3 Detecção Ativa de Intrusos

3.3.1 Snort

O Snort pode ser obtido em `http://www.snort.org`. É um sistema de detecção de intrusos ativo que tem como pontos positivos ter o maior cadastro de assinaturas, ser leve, pequeno, de fácil instalação, já vir configurado, fazer escanamento do micro e verificar anomalias dentro de toda a rede ao qual seu computador pertence.

O Snort é um filtro de pacotes baseado em libcap, um *sniffer* e registrador em log que fornece detecção de invasão de rede básica. Ele é uma ferramenta de detecção de invasão baseada em regras que adota tanto a abordagem preemptiva como a reacionária. Ele ouve tráfego de rede em tempo real e corresponde esse tráfego com regras predefinidas [Anonymous (1999)].

Você pode usar o Snort para implementar funções sensoriais e de análise em um sistema de detecção de intrusão: pode registrar pacotes, analisar pacotes ou as duas coisas. O registro pode consistir de pacotes “brutos” ou de informações de pacotes decodificadas e convenientemente armazenadas. Usando o segundo método, o Snort decodifica os pacotes e os classifica por IP remoto em subdiretórios do diretório de registro [Northcutt, Novak, McLachlan (2000)]. Como comentado em [Anonymous (1999)]:

> Snort é uma ferramenta de detecção de invasão rápida e confiável que exige poucos recursos de sistema. Você pode adicionar assinaturas de ataque obtendo, compilando e executando explorações contra seu sistema enquanto também executa um *sniffer*. O *Sniffer* irá capturar o texto característico ou string binária passada pela ferramenta de ataque para seu servidor. Pegue os últimos poucos caracteres significativos ou caracteres únicos dessa string adicione-os como um descritor de conteúdo em Snort.

Snort devolve respostas principalmente através de logs, mas quando há uma detecção de algo mais sério ele também emite alertas por e-mail.

Instalação

Para instalar obtenha o aplicativo e após isto execute o comando:

`rpm -Uvh snort.rpm`.

Geralmente não é preciso configurar nada, entanto, suas regras estão no arquivo `/etc/snort.conf`. Os arquivos de logs do Snort ficam armazenados em /var/log/snort. Um exemplo de *log* pode ser visto na Figura 3.1 onde há uma detecção de uso do Napster (que é considerado ilegal em algumas instituições).

```
      [**] Napster Client Data [**]
06/11-12:20:29.415133 200.231.172.15:16583 -> 200.131.251.32:6699
TCP TTL:118 TOS:0x0 ID:21786 IpLen:20 DgmLen:111 DF
***AP*** Seq: 0x9347D28A  Ack: 0x881EE9  Win: 0x446F  TcpLen: 20
65 64 70 69 70 6F 6C 65 20 22 31 3A 5C 31 5C 4D  edpipole "1:\1\M
75 73 69 63 61 73 5C 4E 65 6E 68 75 6D 20 64 65  usicas\Nenhum de
20 4E 6F 73 20 2D 20 44 61 73 20 63 6F 69 73 61   Nos - Das coisa
73 20 71 75 65 20 65 75 20 65 6E 74 65 6E 64 6F  s que eu entendo
2E 6D 70 33 22 20 30                             .mp3" 0
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=
```

Figura 3.1: Exemplo de Detecção do Snort

A figura é composta do nome de onde foi acessado (Naspter em questão), Data, hora, de qual micro usando e qual foi sua porta para qual micro usando e qual foi sua porta e O que foi realizado em forma Hexadecimal e em forma de texto.

3.4 Comentando aplicativos não usados

Estes aplicativos foram vistos mas não escolhidos a serem instalados e testados. Segue nome e o porque da não utilização

- **Claymore**

  Semelhante ao Tripwire, é baseado nele e está em versão muito inicial.
  `http://linux.rice.edu/magic/claymore/`

  Licença: GNU General Public License (GPL)

- **IDSA**

  Analisador de log, é semelhante ao Snort. Está em fase experimental.
  `http://jade.cs.uct.ac.za/idsa/`

  Licença: GNU General Public License (GPL)

- **Imsafe**

  Detecta Operações anormais, através de processos, está ainda em fase experimental.
  `http://imsafe.sourceforge.net/`

  Licença: GNU General Public License (GPL)

- **Integrit**

  Alternativa para progaramas como Tripwire e Aide. O uso do Tripwire e do AIDE fez com que o mesmo não fosse testado.
  `http://integrit.sourceforge.net/`

  Licença: GNU General Public License (GPL)

- **Toby IDS**

  É uma versão do tripwire em Perl, só diferencia na linguagem ao qual foi desenvolvido por isto não testado. É uma reimplementação do Tripwire em uma linguagem interpretada.
  `http://www.buttsoft.com/~thumper/software/sysadmin/Toby/`

  Licença: Artistic License

- **Siden**

Necessita ser instalado em redes como o projeto se destinava a servidores, não foi usado, é ativo e usa arquitetura cliente servidor.
`http://siden.sourceforge.net/`

Licença: GNU General Public License (GPL)

- **Free Veracity**

  Semelhante ao Tripwire, descoberto somente no final do prjeto por isto não testado. Talvez seja feito testes futuros, já fora deste projeto.
  `http://www.freeveracity.org/`

  Licença: Free World Licence

- **Psionic - HostSentry**

  Parte do Abacus Project, é uma ferramenta de detecção de invasão que observa anomalias de login.

  O HostSentry espera por logins e gera sua própria informação de logs. Portanto se caso observa discrepância entre o log de seu sistema e do HostSentry é porque uma invasão ocorreu.

  Foi feita a tentativa de Instalação, mas ao instalar ocorreu um erro de execução. Foi enviado e-mail aos desenvolvedores na data de 24/05/2001 e este foi respondido somente na data 13/06/2001 o que gerou a falta de tempo hábil para sua analise. O email contia:

  ```
  Hello, we'are trying to install hostsentry on our
  servers.  We're running RedHat 7.1 with
  python 1.5.2.

  When we try to use hostsentry, we get the following
  message

  ./hostsentry.py:  from:  command not found

  What is the problem???
  ```

  a resposta:

  ```
  Well it may not work under RedHat 7.1.  You should
  do it this way though:
  ```

```
python ./hostsentry.py

- Craig
```

executando desta forma, o comando funcionou, entanto sem tempo hábil para testá-lo adequadamente. Entanto deve-se estar atento ao fato que este programa é bastante comentado em listas de discussão sobre segurança, sendo bastante elogiado.
`http://www.psionic.com/abacus/hostsentry/`

Licença: Livre para cópias e distribuição, porém não há permissão para mudar código - caso haja mudanças, o autor tem que ser avisado.

- **AAFID**

  AAFID (Autonomous Agents for Intrusion Detection) é um sistema distribuído de monitoramento e detecção de invasão que emprega pequenos programas standalone para realizar monitoramento de funções nos hosts de uma rede.

  Necessita de Perl e suas bibliotecas, informou erro na biblioteca tk ao ser arrumada informou erro na biblioteca tcl ao qual não conseguiu a correção.

  Erros mostrados na instalação:

  `Can't locate Tk.pm in @INC`

  `Can't locate Tcl.pm in @INC`

  Foi abandonado por insucesso em instalação.

  Em seu manual é comentado que o técnico que pretende instalá-lo deve ter bom conhecimento de Perl.
  `http://http://www.cerias.purdue.edu/homes/aafid/`

  Licença: Livre para copias e distribuição, porém não há permissão para mudar o código.

- **TARA**

  Aplicativo semelhante ao Cops. Faz uma varredura no micro achando falhas, não checa integridade de arquivos.
  `http://www-arc.com/tara/`

  Licença: GNU General Public License (GPL)

3.5 Comentários Finais

Os sistemas de detecção de intrusos têm o mesmo problema dos software antivírus: os novos ataques não são detectados porque ainda não existe asinatura para eles. Este problema se agrava ainda mais quando vemos o baixo número de assinaturas existente. O Snort que é o aplicativo com maior número de asssinaturas e têm por volta de duas mil assinaturas apenas.

Uma boa notícia em relação à detecção de intrusos é que os *hackers* estão quase alcançando alguns limites que deve desacelerá-los na construção de ferramentas de invasão. A má notícia é que ferramentas de ataques amadurecidas e confiáveis estão amplamente disponíveis [Northcutt, Novak, McLachlan (2000)].

O terrorismo cibernético que tanto se falou não mostra evidências de que seja uma ameaça a curto prazo. Já existem indicações e noções disso, mas a ênfase do terrorismo parece continuar fixa em bombas e armas. A conclusão sobre o terrorismo virtual por enquanto é que sua organização deve ter um plano de emergência.

Uma das tendências não somente futura mas que já está sendo usada é o compartilhamento de informações. Isto pode até parecer errado mas já está acontecendo e através destas, as pessoas da área de segurança estão podendo se defender de melhor forma.

As tendências atuais e emergentes apontam para uma ameaça crescente. As empresas estão sofrendo perdas financeiras devido a uma variedade de ataques e fraudes baseado em computador. A partir do nosso estudo de risco, "Aspectos organizacionais", sabemos que quanto mais cresce a expectativa de perda anual, mais faz sentido investir em contramedidas como sistemas de detecção de intruso. Também sabemos que os sitemas de detecção de intrusão atuais são muito limitados, como observado em [Northcutt, Novak, McLachlan (2000)].

Capítulo 4

Conclusão

O uso da tecnologia é cada dia maior e mais indispensável ao mundo atual, porém o uso de informação tem levado a uma maior concorrência e a uma maior necessidade de segurança das informações o que leva a necessidade de um provedor seguro. Para tal necessidade, usam-se de varias técnicas e uma entre elas é a detecção de intrusos, uma ferramenta forte no combate aos *hackers* ou invasores de sistemas.

Devido a esta necessidade muitas pessoas tem feito estudos na área de segurança de redes de computadores. O detector de intruso é mais uma ferramenta no combate aos invasores que estão cada dia com melhores aplicativos, é um dos metodos usados na luta contra os invasores e junto ao firewall e de um sistema bem configurado, talvez sua segurança nesta corrida virtual.

O futuro parece promissor, mas todo cuidado é pouco. Porém uma certeza pode-se ter: há muito trabalho a ser feito. Ferramentas, técnicas e treinamentos estão sendo desenvolvidos para conter as ameaças, resta agora lutar e vencer está batalha. A ameaça virtual embora freqüente não chega a temer tanto quanto a ameaça química mas os valores da informação hoje são bem maiores que os valores materiais, o mundo da informação precisa ser protegido.

Empresas bem estruturadas hoje contam com uma equipe de segurança onde exitem especialistas em todas áreas e a de detecção apesar de ser uma área ainda criança está em grande crescimento e já é necessária.

Este trabalho mostrou ser de boa prática a instalação de pelo menos dois sistemas de detecção de intruso, sendo um ativo e um passivo, talvez o melhor seja colocar os dois passivos. É indicado a instalação do ativo Snort ou HostSentry pelo poder e pela facilidade de instalação, já o passivo indica se AIDE ou Tripwire

ou até mesmo os dois.

Referências Bibliográficas

[AIDE (2001)] Lehti, Rami [alcunha: Rammer]. *Advanced Intrusion Detection Environment*. 2001. [url: `http://www.cs.tut.fi/~rammer/aide.html`].

[Anonymous (1999)] *Ed. Scott D. Meyers* Anonymous. *Maximum Linux Security*.Indianapolis, Indiana, 1999.

[Forster (2001)] Forster, Jim. *Snort - The Open Source Network Intrusion Detection System*. 2001. [url: `http://www.snort.org`].

[Nortcutt (1999)] Nortcutt, Stephen. *Network Intrusion Detection: an analyst's Handbook* New Riders, 1999, 268 p.

[Northcutt, Novak, McLachlan (2000)] Northcutt, Stephen & Novak, Judy & McLachlan, Donald. *Segurança e Prevenção em Redes*. Siciliano, São Paulo, 2000.

[Santos, Caminhas, Errico (1999)] Santos, Ricardo Bernardo & Caminhas, Walmir Matos & Errico, Luciano. Detecção de Intruso: Uma abordagem usando redes neurais. *InfoComp - Anais da II Secicom - IP Semana de Ciência da Computação. UFLA* Lavras MG

[Schlemer (2001)] Schlemer, Elgio. *Segurança em Redes*. Porto Alegre, UFRGS, 2001. [url: `http://www.inf.ufrgs.br/~elgio/trabs-html/redes/seg\protect\T1\textunderscorerede.html`].

[Toxen (2000)] Toxen, Bob. *Real World Linux Security : Intrusion Prevention, Detection and Recovery*. Prentice Hall, New York, 2000.

[Tripwire (2001)] Forrester, Ron. *Project: Tripwire*. 2001. [url: `http://www.sourceforge.net/projects/tripwire/`].

[Weber (2000)] Weber, Raul Fernando. Segurança na Internet. *Anais da XIX JAI - Jornada De Atualização em Informática. PUCPR campus Curitiba PR*, 17 a 21 de julho de 2000. p. 43-82.

Apêndice A

Resposta do Tripwire

Após fazer a checagem no Tripwire é devolvido a resposta nesta forma.

```
Tripwire(R) 2.3.0 Integrity Check Report

Report generated by:          root
Report created on:            Qui 21 Jun 2001 14:11:57 BRT
Database last updated on:     Never

=======================================================
Report Summary:
=======================================================

Host name:                    prolog
Host IP address:              127.0.0.1
Host ID:                      None
Policy file used:             /etc/tripwire/tw.pol
Configuration file used:      /etc/tripwire/tw.cfg
Database file used:           /var/lib/tripwire/prolog.twd
Command line used:            tripwire --check

=============================================================
Rule Summary:
=============================================================

-------------------------------------------------------------
  Section: Unix File System
-------------------------------------------------------------

  Rule Name                       Severity Level    Added
  Removed  Modified
```

```
 ---------                          --------------    -----
-------  --------
 Invariant Directories              66                0
0        0
 Temporary directories              33                0
0        0
 Tripwire Data Files                100               0
0        0
 Critical devices                   100               0
0        0
 User binaries                      66                0
0        0
 Tripwire Binaries                  100               0
0        0
 Libraries                          66                0
0        0
 File System and Disk Administraton Programs
                                    100               0
0        0
 Kernel Administration Programs     100               0
0        0
 Networking Programs                100               0
0        0
 System Administration Programs     100               0
0        0
 Hardware and Device Control Programs
                                    100               0
0        0
 System Information Programs        100               0
0        0
 Application Information Programs
                                    100               0
0        0
 Shell Related Programs             100               0
0        0
 Critical Utility Sym-Links         100               0
0        0
 Critical system boot files         100               0
0        0
 Critical configuration files       100               0
0        0
 System boot changes                100               0
0        0
 OS executables and libraries       100               0
0        0
 Security Control                   100               0
```

```
 0        0
  Login Scripts                   100               0
 0        0
  Operating System Utilities      100               0
 0        0
  Shell Binaries                  100               0
 0        0
* Root config files               100               0
 0        1

Total objects scanned:  28155
Total violations found:  1

============================================================
Object Summary:
============================================================

------------------------------------------------------------
# Section: Unix File System
------------------------------------------------------------

------------------------------------------------------------
Rule Name: Root config files (/root)
Severity Level: 100
------------------------------------------------------------

Modified:
"/root"

============================================================
Error Report:
============================================================

No Errors

------------------------------------------------------------
*** End of report ***
```

Severity Level - Nível de segurança das pastas, Added - Numero de arquivos adicionado na pasta, Removed - Numero de arquivos removido na pasta, Modified - Numero de arquivos modificados na pasta.

Apêndice B

Resposta AIDE

Após fazer a checagem no Tripwire é devolvido a resposta nesta forma.

```
AIDE found differences between database and filesystem!!
Start timestamp: 2001-06-22 12:06:20
Summary:
Total number of files=29086,added files=0,removed files=0,
changed files=14

Changed files:
changed:/boot
changed:/boot/System.map
changed:/etc
changed:/etc/sysconfig/hwconf
changed:/etc/mail/virtusertable.db
changed:/etc/mail/access.db
changed:/etc/mail/domaintable.db
changed:/etc/mail/mailertable.db
changed:/etc/aliases.db
changed:/etc/adjtime
changed:/etc/mtab
changed:/etc/ioctl.save
changed:/etc/issue
changed:/etc/issue.net
Detailed information about changes:

File: /boot
Mtime: old = 2001-06-21 13:28:50, new = 2001-06-22 09:43:32
Ctime: old = 2001-06-21 13:28:50, new = 2001-06-22 09:43:32
```

```
File: /boot/System.map
Mtime: old = 2001-06-21 13:28:50, new = 2001-06-22 09:43:32
Ctime: old = 2001-06-21 13:28:50, new = 2001-06-22 09:43:32

File: /etc
Mtime: old = 2001-06-21 13:28:46, new = 2001-06-22 11:57:39
Ctime: old = 2001-06-21 13:28:46, new = 2001-06-22 11:57:39

File: /etc/sysconfig/hwconf
Mtime: old = 2001-06-21 13:29:07, new = 2001-06-22 09:43:48
Ctime: old = 2001-06-21 13:29:07, new = 2001-06-22 09:43:48

File: /etc/mail/virtusertable.db
Mtime: old = 2001-06-21 13:29:34, new = 2001-06-22 09:44:17
Ctime: old = 2001-06-21 13:29:34, new = 2001-06-22 09:44:17
MD5: old = DToO6uXqoXTfcLWHxue6ng== ,
 new = amGTawC/ZNHgBlvFwp3tkQ==

File: /etc/mail/access.db
Mtime: old = 2001-06-21 13:29:35, new = 2001-06-22 09:44:17
Ctime: old = 2001-06-21 13:29:35, new = 2001-06-22 09:44:17
MD5: old = TbGSsMdUZLCfvy4Zq3l6zQ== ,
 new = kdN+BfgCtUuqA+MFmGskTQ==

File: /etc/mail/domaintable.db
Mtime: old = 2001-06-21 13:29:35, new = 2001-06-22 09:44:17
Ctime: old = 2001-06-21 13:29:35, new = 2001-06-22 09:44:17
MD5: old = PHWI2tD7rS40xfKcxmVHBQ== ,
 new = hA9I2fZNlShycxa8I/r6vQ==

File: /etc/mail/mailertable.db
Mtime: old = 2001-06-21 13:29:35, new = 2001-06-22 09:44:17
Ctime: old = 2001-06-21 13:29:35, new = 2001-06-22 09:44:17
MD5: old = QumKzBs/4MRFvmlvRmTyPA== ,
 new = jnWRXvH0/bCpIkZm/5ErNA==

File: /etc/aliases.db
Mtime: old = 2001-06-21 13:29:34, new = 2001-06-22 09:44:17
Ctime: old = 2001-06-21 13:29:34, new = 2001-06-22 09:44:17
MD5: old = fEWhHJ+6dzBH0SrBYwMbig== ,
 new = yj/gK+lzAYKkXijfSzb1Vw==

File: /etc/adjtime
Mtime: old = 2001-06-21 09:17:11, new = 2001-06-22 09:30:29
Ctime: old = 2001-06-21 09:17:11, new = 2001-06-22 09:30:29
MD5: old = SzLz5bYcp0rIkPAu43rRtg== ,
```

```
 new = Y3MvHKvAVLWmxTKXbvb6zQ==

File: /etc/mtab
Mtime: old = 2001-06-21 13:28:46, new = 2001-06-22 11:57:39
Ctime: old = 2001-06-21 13:28:46, new = 2001-06-22 11:57:39
Inode: old = 230282 , new = 230324

File: /etc/ioctl.save
Mtime: old = 2001-06-21 13:28:51, new = 2001-06-22 09:43:33
Ctime: old = 2001-06-21 13:28:51, new = 2001-06-22 09:43:33

File: /etc/issue
Mtime: old = 2001-06-21 13:29:41, new = 2001-06-22 09:44:23
Ctime: old = 2001-06-21 13:29:41, new = 2001-06-22 09:44:23

File: /etc/issue.net
Mtime: old = 2001-06-21 13:29:41, new = 2001-06-22 09:44:23
Ctime: old = 2001-06-21 13:29:41, new = 2001-06-22 09:44:23
```

A resposta nos mostra arquivos adicionados, apagados e modificados informando data de quando e data nova caso seja apenas modificado.