UNIVERSIDADE DE SÃO PAULO

UM SISTEMA ADAPTATIVO DE DETECÇÃO DE INTRUSÃO EM REDES DE COMPUTADORES

ADRIANO M. CANSIAN
EDSON DOS S. MOREIRA
ROGÉRIO B. MOURO
FÁBIO T. MORISHITA
ANDRÉ C. P. DE L. F. DE CARVALHO

Nº 22

NOTAS

Instituto de Ciências Matemáticas de São Carlos

Instituto de Ciências Matemáticas de São Carlos

ISSN - 0103-2577

UM SISTEMA ADAPTATIVO DE DETECÇÃO DE INTRUSÃO EM REDES DE COMPUTADORES

ADRIANO M. CANSIAN
EDSON DOS S. MOREIRA
ROGÉRIO B. MOURO
FÁBIO T. MORISHITA
ANDRÉ C. P. DE L. F. DE CARVALHO

Nº 22

NOTAS DO ICMSC
Série Computação

São Carlos
Fev./1996

Um Sistema Adaptativo de Detecção de Intrusão em Redes de Computadores

Adriano M. Cansian (adriano@nimitz.ibilce.unesp.br)[1]
Edson d.S.Moreira, Rogério B. Mouro, Fábio T. Morishita, André C.P.de L.F. de Carvalho
(edsmorei, mouro, teruo, andre@icmsc.sc.usp.br)[2]

Resumo

O grande crescimento da Internet tanto em termo do número de máquinas conectadas quanto ao número e tipo de serviços oferecidos, segurança tem se tornado um tema chave para os projetistas destas technologias. Este projeto pretende desenvolver um sistema que, localizado em pontos estratégicos da rede, captura e avalia os pacotes passantes, tentando localizar conexões suspeitas. Ele provê uma lista destas conexões ao administrador, ajudando-o na tomada de decisão no estágio inicial da intrusão. Redes neurais são utilizadas para o reconhecimento de padrões de ações de intrusão dentro das seqüências de bytes passantes pelo sistema de observação. O sistema é altamente adaptativo, fazendo com que a adição de um perfil diferente de intrusão pode ser fácil e rapidamente ensinada à rede neural.

Abstract

As the Internet expands both in number of hosts connected and in terms of the number of services provided, security has become a key issue for the technology developers. This project intends to develop a system which, positioned at key points of the network, will keep looking to the passing packets, searching for suspicious connections. It provides a list of such connections for the administrator, enabling him/her to take the proper action in the early stage of the intrusion. Neural Networks are used to look for profiles of intrusion within the data streams analised. The assessment is made by comparison to well known profiles of intrusion. The system is highly adaptive, since new profiles can be added to the data base and the Neural Network re-trainned to consider the new profile.

[1] DCCE-IBILCE/UNESP Cx Postal 136
15054-000 São José do Rio Preto SP
Fone: +55 17 2244667
FAX: +55 17 2248692

[2] ICMSC-USP Cx Postal 668
13560-970 São Carlos SP
Fone: +55 16 2749136
FAX: +55 16 274 9150

1. Introdução

Com o crescimento das redes de computadores, crescem também as preocupações com a segurança das informações armazenadas nos computadores conectados a elas. Além disso, uma quantidade cada vez maior de atividades essenciais é realizada por intermédio das redes (principalmente a Internet), e seu funcionamento correto e confiável torna-se de fundamental importância. Paralelamente, ações de pirataria, tentativas de intrusão, invasões consumadas e ações de *break-in* aumentam a cada dia, envolvendo um número cada vez maior de máquinas [1-2]. Isso faz com que técnicas especiais de segurança se tornem indispensáveis nos sistemas computacionais modernos.

Diversos Sistemas de Detecção de Intrusão (SDIs) têm sido desenvolvidos e alguns implementados experimentalmente. Estes sistemas são divididos em "baseados em *hosts*" (*host-based*) [3-7] e "baseados em rede" (*network-based*) [8-10]. Os sistemas *host-based* empregam os registros de auditoria (chamados de *audit trails*, normalmente representados por *daemons*) como principal fonte para identificar intrusões, enquanto os sistemas *network-based* constroem seu próprio conjunto de informações utilizando, transparentemente, o tráfego de pacotes da rede. Os sistemas *network-based* podem também utilizar análise dos registros de auditoria dos *hosts* da rede sobre a qual atuam para melhorar sua eficácia.

Este trabalho descreve o projeto de um sistema de segurança do tipo *network-based* que utiliza técnicas de inteligência artificial e de redes neurais para descobrir intrusos, registra-los e fornecer elementos para auxiliar no processo de decisão sobre possíveis ações a serem tomadas pelo administrador do sistema. O sistema de detecção é composto por um ou mais **agentes de segurança**, que, colocados em locais estratégicos da rede, oferecem as informações a serem mostradas em **estações de gerenciamento de segurança.**

No contexto deste trabalho, algumas definições devem ser estabelecidas:

- ***Conexão***, indica um conjunto de pacotes de informações cujos endereços origem-destino são os mesmos. Embora os endereços apenas não sejam suficientes para determinar uma conexão, eles bastam para uma análise inicial. Para uma avaliação mais fina, o número de portas e a identificação da conexão pode ser usada.
- ***Domínio,*** indica um endereço IP, uma rede, ou uma faixa de números IP que serão considerados como pertencentes a um mesmo grupo, para ações de limitação de fronteiras de segurança. Não tem nenhuma função como as definidas em contexto do NIS ou do DNS.
- ***Host,*** uma máquina para a qual foi atribuído um número IP.

2. Detecção de Comportamentos Suspeitos

Existem algumas técnicas de acesso pelas quais as tentativas de ataque se desenvolvem. Na grande maioria dos cenários, o intruso se encontra fisicamente distante do sistema sob ataque [2], e assim utiliza algum ponto de rede durante sua tentativa. A *figura 1* ilustra estes 4 tipos de comportamentos. No ataque simples o intruso usa uma máquina na rede para acessar outra

máquina. No ataque *doorknob*, diversas tentativas de acesso (com ou sem sucesso) se desenvolvem a partir de uma única máquina. No ataque em cadeia, o intruso procura passar por diversas máquinas para ocultar sua verdadeira identidade ou seu ponto de ataque inicial; no ataque em *looping*, a máquina atacada é a mesma originadora do ataque.

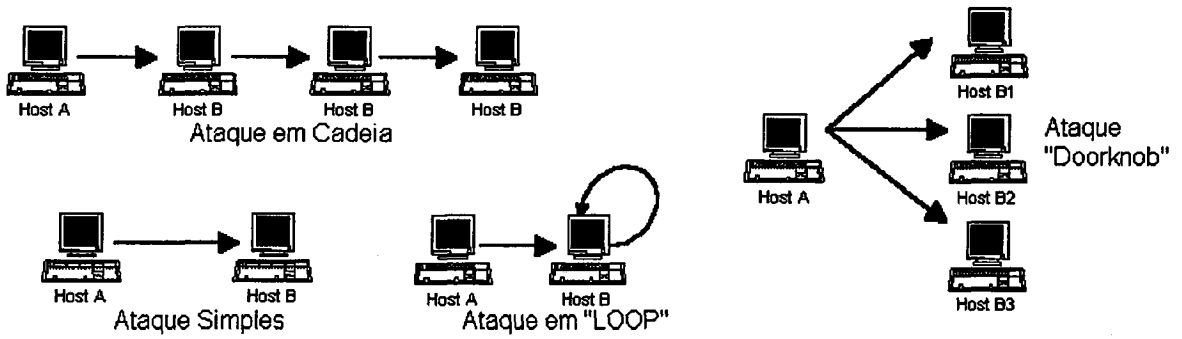


Figura 1 - Tipos de ataques

Os modelos de SDIs iniciais, projetados para atuarem em computadores isolados, empregam algoritmos básicos que incluem o cálculo de funções multinomiais e aproximações de matrizes covariantes, para detectar desvios do comportamento normal, e sistemas especialistas para detectar violações da política de segurança. Os modelos mais recentes monitoram um grande número de computadores interconectados em rede e transferem as informações monitoradas para processamento num equipamento centralizador [11-12], utilizando técnicas de processamento distribuído, além de possuírem o fluxo de tráfego da rede como parte de seus algoritmos de detecção [14-16]. A maioria destes SDIs possue em algum estágio, um processo auditor (*daemon*) em cada máquina, responsável pela captura das ações de segurança e que transmite os dados relevantes para análise numa máquina centralizadora. Os sistemas baseados em redes, ao invés de usarem os registros de auditoria, analisam o tráfego de pacotes na rede para detectar o comportamento intrusivo. Este tipo de sistema possui uma série de vantagens dentre as quais podemos destacar as seguintes:

- Apesar de existirem diversos sistemas de redes, existem certos protocolos padronizados, tais como TCP e UDP. Estes padrões podem ser usados para monitoramento de um conjunto heterogêneo de máquinas e sistemas operacionais.
- Os registros de auditoria muitas vezes não estão disponíveis instantaneamente, retardando o tempo de resposta.
- A própria natureza de *broadcast* das redes permite acesso aos dados assim que eles são transmitidos, disparando imediatamente o processo de detecção de ataque.
- Os processos e registros de auditoria são freqüentemente vulneráveis a ações de *hacking*.
- Freqüentemente a coleta dos registros de auditoria são desabitados pelos gerentes, em virtude da perda de performance causada (de 5 a 20%). Se a análise destes registros é realizada na própria máquina, mais degradação ocorre. Se os dados são transmitidos pela rede para análise em outra máquina, perde-se banda de passagem. Os sistemas de captura de pacotes não afetam os sistemas monitorados e não sofrem a influência dos administradores, uma vez que são independentes das configurações individuais das máquinas.

Uma das propostas de inovação em nosso sistema de gerenciamento é a introdução de um agente de segurança, capaz de detectar comportamento intrusivo nas conexões. Este agente atua capturando e decifrando pacotes que são transmitidos através da rede sob monitoramento.

Para realizar inferências acerca do estado de segurança das conexões, utiliza um sistema especialista e uma rede neural. Redes neurais artificiais [22-25] são sistemas paralelos distribuídos, compostos por unidades de processamento simples (neurônios) que computam certas funções matemáticas, dispostas em uma ou mais camadas, interligadas por um grande número de conexões unidirecionais. Geralmente estas conexões estão associadas a pesos, os quais armazenam o conhecimento representado no modelo, e servem para ponderar a entrada recebida por cada neurônio da rede. O funcionamento destas redes é inspirado numa estrutura física concebida pela natureza: o cérebro humano. Como resultado, a rede fornecerá um número que, baseado em informações de intrusões anteriores, dará uma idéia sobre a severidade de um determinado ataque.

O sistema se baseia no fato de que um comportamento intrusivo pode ser detectado a partir da análise de padrões pré-determinados. Os seguintes exemplos ilustram alguns destes comportamentos suspeitos [5]:

- Alguém tentando acessar um determinado sistema deve gerar uma taxa anormal de falhas de *password*, seja com relação a um computador em particular ou o sistema como um todo.
- Alguém que ganhe acesso indevido ao sistema através de um *account* e *password* não autorizados deve desenvolver uma sessão diferente do usuário legítimo, em termos de tempo de *login*, localização do acesso ou tipo da conexão. Além disso o comportamento do intruso pode diferir consideravelmente do usuário legítimo em termos de acesso aos diretórios ou execução de utilitários de *status* do sistema.
- Um usuário tentando penetrar em mecanismos de segurança num sistema operacional deve executar programas diferentes e disparar mais violações de proteção através da tentativa de acesso a arquivos e programas não autorizados. Se há sucesso, ele terá acesso a programas e arquivos normalmente não permitidos a ele.
- Um usuário tentando acessar documentos sensíveis à segurança pode realizar *logins* em horários não usuais e rotear dados para discos ou impressoras normalmente não usados por ele.
- Um usuário tentando obter dados não autorizados de uma base de dados através de agregação e inferência deve recuperar mais registros que o normal.
- *Trojan Horses* e Vírus: o comportamento de um *Trojan Horse* ou Vírus plantado ou substituído no sistema, pode diferir do programa legítimo em termos de tempo de CPU ou atividade de I/O.
- Um intruso capaz de monopolizar um recurso (por exemplo uma rede) deve possuir alta atividade anormal com relação ao recurso, enquanto a atividade de todos os outros usuários é, em contraste, normalmente baixa.

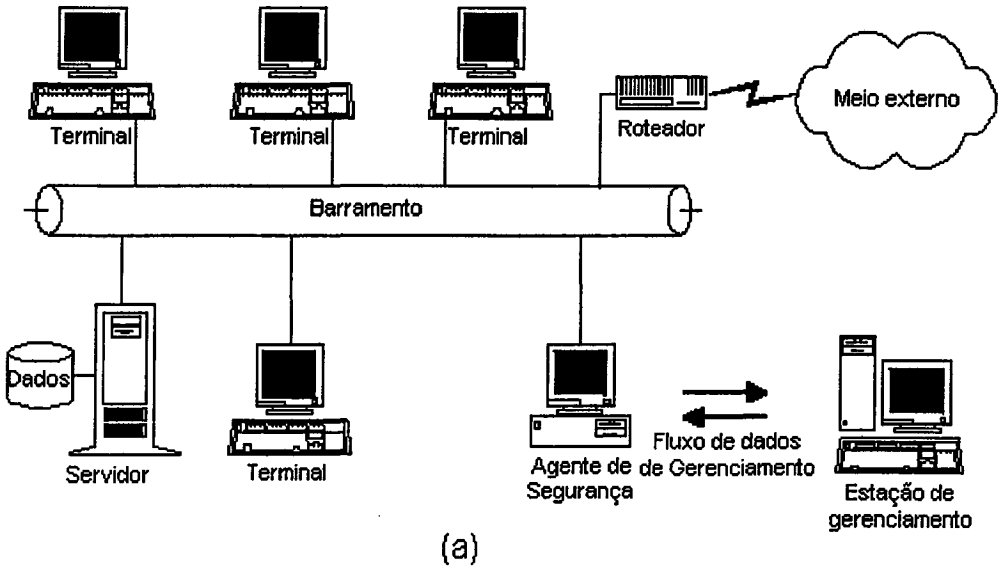
Nossa proposta é que a utilização de redes neurais pode fornecer mecanismos de reconhecimento de ataque eficazes, além de inserir uma capacidade adaptativa que acompanhe as mudanças nas técnicas de *hacking*.

3. O Agente de Segurança

O agente é implantado em uma máquina segura, ou seja, que é logicamente invisível às outras máquinas, e se encontra num local onde o acesso físico é restrito, podendo assim ser acessado apenas em determinadas condições especiais. Também podem ser utilizadas técnicas especiais de ocultação, utilizando por exemplo uma alocação dinâmica de IP. Esta máquina segura é posicionada em pontos sensíveis do sistema de rede. Estes pontos sensíveis podem ser:

- Diretamente na subrede sob monitoramento (*Fig.2a*).
- Junto a pontos de interconexão de subredes internas (*Fig.2b*);
- Compartilhando o *backbone* onde se ligam roteadores que forneçam, às subredes sob monitoramento, acesso ao mundo exterior (*Fig.3a*);
- Avaliação de pontos de acesso de interconexão de WANs (linhas seriais). Neste caso estamos propondo o desenvolvimento de um sistema de monitoração que opere sobre linhas seriais em conexões ponto-a-ponto, de maneira similar a um "grampo" telefônico (*Fig.3b*). Note-se a importância desta configuração devido ao fato de que, geralmente, roteadores podem ter portas ligadas a mais do que uma subrede.

Em todas as situações acima, o agente passivamente monitora a rede, e capta os pacotes em circulação através da utilização da interface de rede em modo promíscuo.



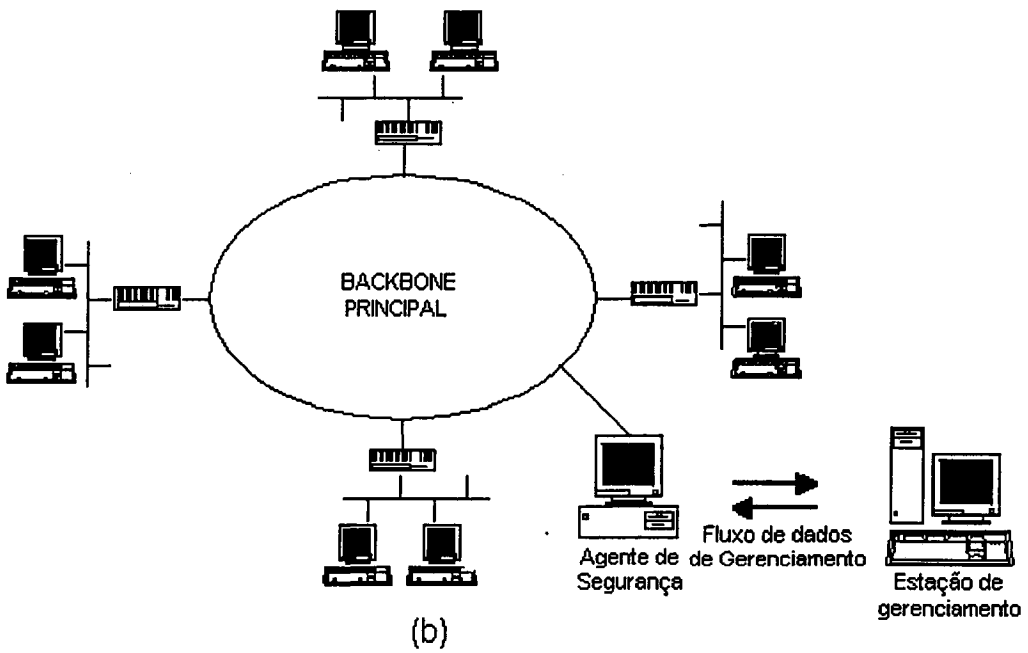
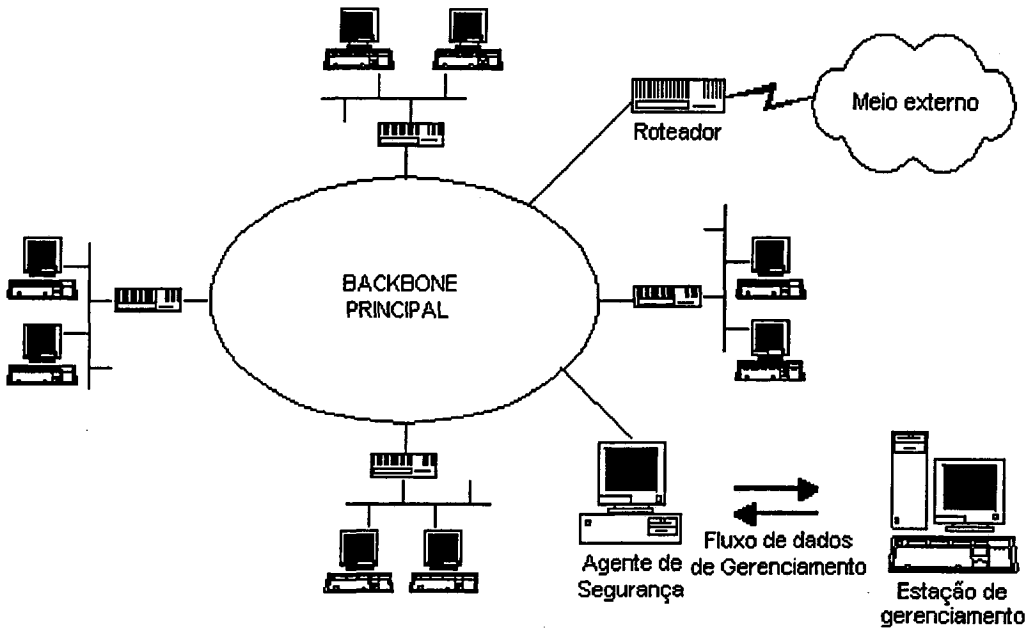
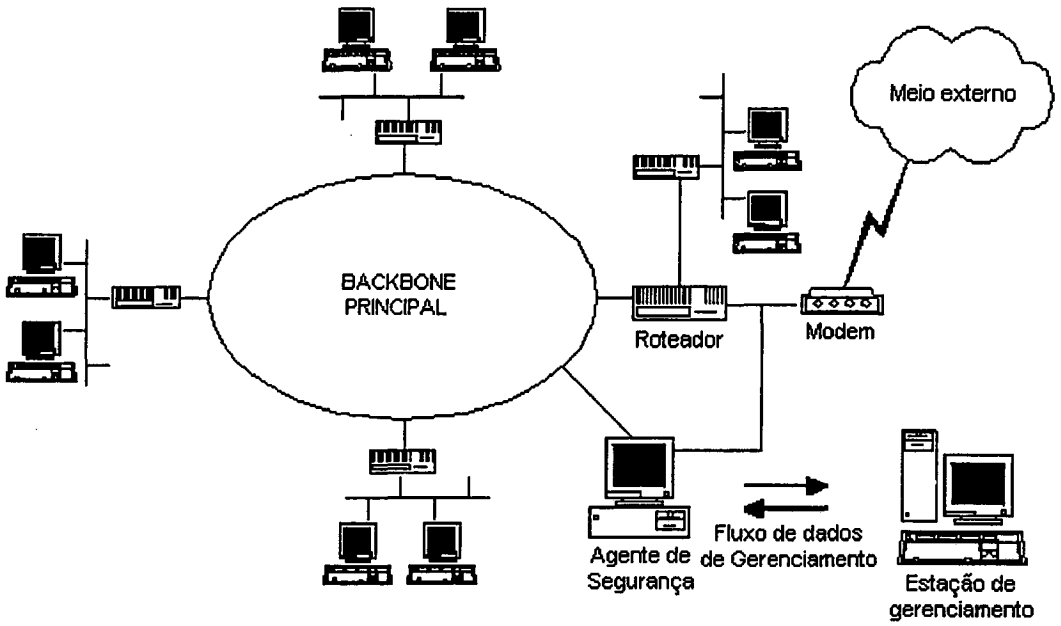


Figura 2 - Cenários Possíveis de monitoramento. (a) Monitoramento de uma subrede. (b) Monitoramento de um conjunto de subredes



(a)



(b)

Figura 3 - Cenários Possíveis de monitoramento. (a) Monitoramento de roteador de acesso a subredes. (b) Monitoramento de uma ligação serial externa.

O agente é organizado num modelo formado por 4 camadas (*Fig.4*), que tratam o fluxo de pacotes e que fornecem o vetor de estímulo para a rede neural. O nível mais baixo apenas captura um fluxo de dados na rede e passa os pacotes ordenados para a segunda camada.

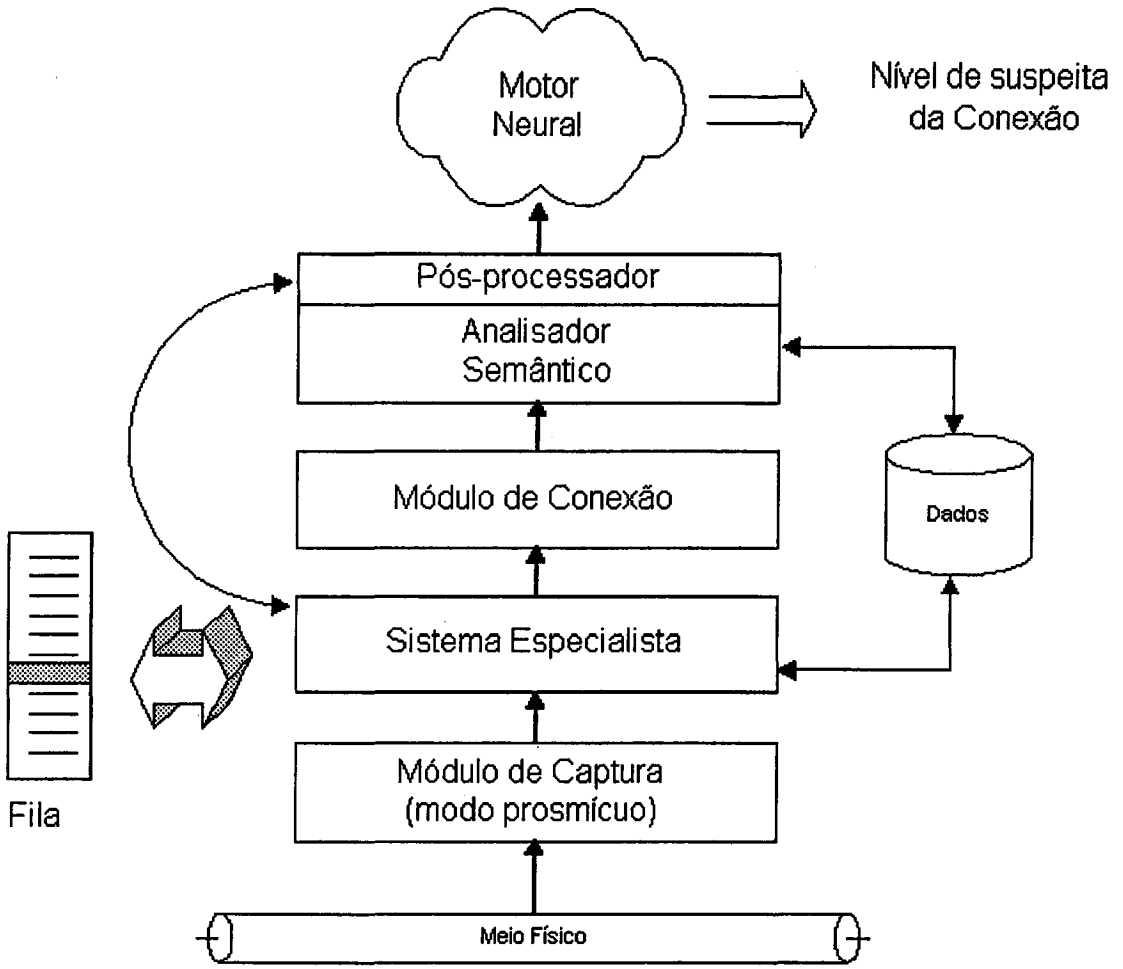


Figura 4 - Organização do Sistema

A segunda camada é formada por 2 módulos: módulo de pré-seleção de pacotes e módulo do sistema especialista. O módulo de pré-seleção realiza uma filtragem inicial dos pacotes que possam representar eventos de interesse, como por exemplo que tipo de protocolo será monitorado, ou quais destinos ou origens devem ser considerados. Os pacotes previamente filtrados são então enviados para análise pelo sistema especialista.

O sistema especialista utiliza as seguintes informações na tomada de decisões:

- Quais os caminhos esperados de conexão (origem e destino da conexão), e quais podem representar risco;
- Portas de origem e destino;
- Sensibilidade das máquinas e confiabilidade dos domínios.

Estes dados servem como elementos de análise para monitoração dos eventos de risco, que são armazenados numa fila de permanência que os mantêm durante um tempo definido. Os dados colocados nesta fila tratam-se de vetores que incluem fonte e destino da conexão, portas envolvidas, um "grau de segurança" (GS) e um *time stamp* (*Fig.5*). Cada conexão é representada por um vetor na fila.

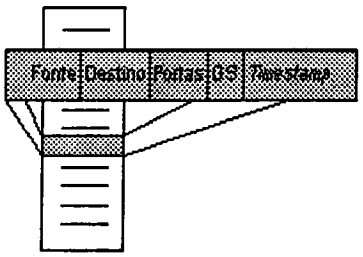


Figura 5 - Representação da fila de verificação

O "grau de segurança" é um valor numérico que vai sendo acrescido à medida que são monitorados eventos de interesse vindos de um mesmo par origem-destino (ou de um mesmo domínio). Por exemplo, o sistema detecta um *finger* vindo do host A para host B, e inclui o seu vetor na fila, com um valor de grau de segurança pré-determinado para o *finger* (GS=10, p.ex). Ao detectar um novo evento, por exemplo um *telnet* (GS=15, p.ex), a fila é pesquisada e, encontrando-se uma entrada proveniente da mesma origem-destino, o grau de segurança deste vetor é acrescido do valor do grau de segurança do *telnet*, ficando com GS=25. Neste caso, não é criado um novo vetor na fila.

Outro caso a se considerar é aquele em que o evento detectado vem de um domínio de onde já houve algum outro evento anterior (já registrado na fila), mas não necessariamente proveniente da mesma máquina. Neste caso um novo vetor, é inserido na fila, com o grau de segurança equivalente, acrescido de um peso que indica que já há alguma ocorrência vinda daquele domínio.

Ciclicamente esta fila é verificada, e quando o campo do grau de segurança de um vetor atinge um *threshold* pré-estabelecido, o sistema especialista passa a monitorar aquela conexão ou, opcionalmente, todas as conexões vindas daquele domínio. Ainda há a possibilidade de se realizar diferentes tipos de buscas e verificações na fila, considerando-se não apenas o *threshold* para um par origem-destino, mas também relativamente a domínios ou intervalos de endereços. Estas buscas podem resultar não apenas em atividades de monitoramento, mas também em ações de *logging* e auditoria, conforme a necessidade.

Desta maneira, quando o sistema especialista identificar os comportamentos que diferem dos padrões aceitáveis, todos os pacotes provenientes da conexão são enviados para o próximo nível (Camada Manipuladora de Conexão).

Esta terceira camada baseia-se no modelo hierárquico do *Network Security Monitor* (NSM) [13] e recebe os pacotes e os organiza numa relação de causa e efeito, identificando um fluxo de dados unidirecional. Isso é feito através da análise dos campos origem, destino, portas e número de seqüência dos pacotes, resolvendo assim o problema de fragmentação. Este tratamento pode ser realizado tanto a nível de IP como a nível de TCP.

Uma vez identificado e ordenado, o fluxo consiste dos dados sendo transferidos de uma máquina para outra, através de um dado conjunto de portas, por um determinado protocolo. Estes pacotes são mapeados num vetor de fluxo, que é a transcrição dos dados de uma conexão em particular. Se for necessário, são formados pares de vetores de fluxo, de modo a se representar bi-direcionalmente o tráfego de dados, ou seja, uma conexão *host-to-host* (*Fig.6*).

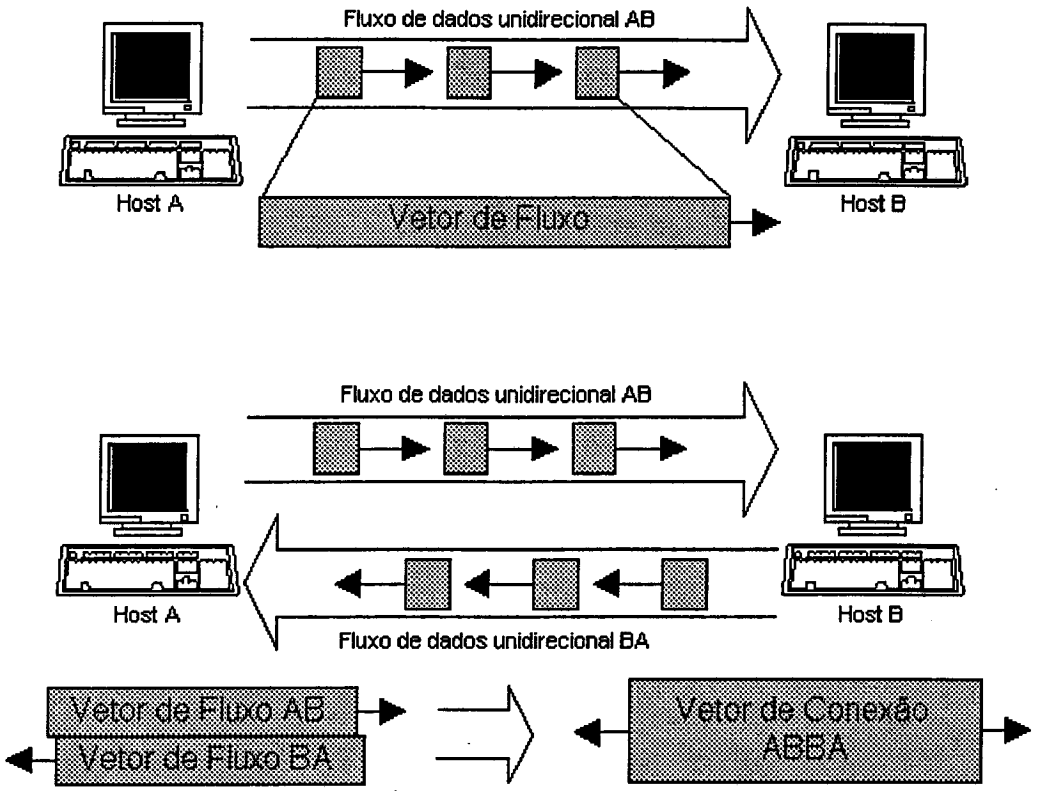


Figura 6 - Determinação do vetor de conexão

Os pares de vetores de fluxo são representados por um vetor de conexão, que contém todo o *stream* de dados em andamento entre os 2 *hosts* pela conexão sob monitoramento. Os vetores de conexão são então enviados para a quarta camada, para serem tratados por um analisador semântico.

O analisador semântico age sobre os vetores de conexão, buscando perfis de ataque que estejam presentes nos dados. Os perfis de ataque estão armazenados em uma base de dados e são atualizado conforme a necessidade. Estes perfis tratam-se de assinaturas de ataque (*attack signatures*) e contém informações sobre como uma sessão suspeita se comporta. São analisados os dados contidos na conexão, e quantos e quais deles casam com o perfil. Estas informações são enviadas para um último módulo de pós-processamento, que agrega as informações do analisador semântico (perfis casados) a informações do sistema especialista (sensibilidade dos serviços e conexões) e sua base de dados, para formar o vetor de estímulo para a rede neural.

O vetor de estímulo (*Fig. 7*) conterá valores sobre a conexão:

- Nível de capacidade do serviço. Por exemplo, *telnet* oferece mais capacidades que FTP;
- Nível de autenticação do serviço, isto é, *telnet* exige mais autenticação do que SMTP;
- Nível de segurança das máquinas destino e origem, que pode ser um valor numérico fornecido por um sistema analisador padronizado, como por exemplo COPS, SATAN ou *Tigger Scripts*, ou ainda um valor *default* ou um valor referente ao Sistema operacional, atribuído por organizações internacionais de segurança como FIRST [19], CERT [20] ou COAST [21];
- Quantidade de dados transmitida, que pode ser um importante indicativo de ações suspeitas, se associado ao tipo de serviço utilizado;

- Horário da conexão. Atividades, serviços e quantidades de dados transmitidos num determinado horário (por exemplo no horário comercial) podem ser suspeitas se ocorrerem em outro horário (por exemplo, à noite);
- E, finalmente, a quantidade de *strings* suspeitos localizados pelo analisador semântico nos dados, tais como: *su*, *root*, *passwd*, *telnet*, *debug*, *rlogin*, *rsh*, *chown*, *service disable*, dentre outros possíveis.

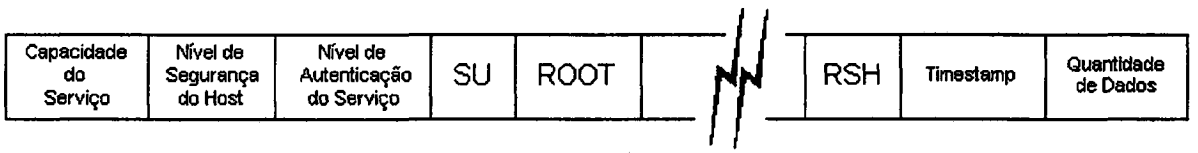


Figura 7 - Representação resumida do vetor de estímulo

A rede neural analisa o vetor de estímulo, com os respectivos pesos que representam a importância da ocorrência dos eventos, e procura atribuir um grau de suspeita, que representa o estado de segurança da conexão. Antes que a rede neural possa identificar potenciais *hackers*, ela precisa ser treinada com um conjunto significativo e suficientemente grande de vetores de estímulo, que representem tanto o comportamento de conexões são suspeitas quanto legítimas. Uma vez treinada, a rede deve utilizar sua propriedade de generalização para identificar corretamente usuários que apresentem características semelhantes àquelas presentes nas ações de intrusão utilizados para treiná-la.

O sistema de gerenciamento (remoto) recebe então o valor numérico que representa o estado de segurança das conexões suspeitas e as mapeia, classificando-as pelo nível de segurança, por intermédio de um código de cores. A partir deste ponto várias ações podem ser tomadas, tais como emissão de vários níveis de alertas aos administradores (dependendo do nível de risco dos eventos detectados), disparo de processos de *logging*, ativação de contra-medidas para isolar o *host* ou o domínio originador do ataque (ajuste ativo de *wrappers* [17], *firewalls* ou filtros [18]), dentre outros.

A rede neural também pode ser ajustada, de maneira a refletir as mudanças nos padrões de comportamento das ações de *hacking*. Os administradores podem detectar comportamentos que são altamente intrusivos, mas que a rede neural ainda não está identificando corretamente (ou está identificando com uma severidade inferior do que o esperado). Neste caso a rede pode ser alimentada com os novos padrões, e passar por um novo treinamento de maneira a aprender a reconhecer o novo padrão adequadamente, refletindo a mudança de comportamento. Esta capacidade adaptativa é a inovação mais importante e que não se encontra presente nos SDIs conhecidos.

4. Sistema de Gerenciamento

O módulo de gerenciamento é independente do módulo de segurança. Os dois módulos se comunicam via TCP/IP que, se por um lado lhe dá a capacidade de ser colocado em qualquer ponto da internet, por outro abre espaço a ações de ferramentas do tipo *sniffers* que, comandadas por *hackers*, podem possibilitar o vazamento de informações importantes para mãos erradas. No entanto, a possibilidade de colocar o módulo de gerência em qualquer lugar é uma característica interessante, inclusive para que o agente possa servir a uma comunidade

de gerentes (como num campus, com vários institutos e departamentos) e nós estamos trabalhando para estabelecer um método seguro de transmissão de informações agente-gerente.

Correntemente, o módulo de gerenciamento está sendo desenvolvido como uma ferramenta extra no MultView [26,27,28] e, além de aplicações de detecção de intrusão, faz o monitoramento de outras informações e dados da rede, também oriundos do agente de segurança:

- Qual a utilização da conexão externa;
- Quais as máquinas, redes ou domínios mais acessados interna e externamente;
- Quais os horários de pico de utilização;
- Quais os serviços mais utilizados e por quem.

MultView, é um sistema gerenciador que roda em máquinas UNIX/XWindows e usa a arquitetura SNMP [30] para busca e troca de informações de gerenciamento. Uma versão atualizada deverá estar disponível em breve e utiliza o protocolo SNMPv2 que, entre outras coisas, corrige algumas deficiências da versão anterior e implementa algumas funções adicionais (comunicação gerente-gerente, mecanismos de segurança, transferência de dados mais ágil, etc). O MultView possui ainda recursos multimídia, através dos quais são apresentadas imagens (fotos digitalizadas de pessoas - normalmente administradores - e equipamentos, imagens fornecidas por câmeras de monitoração localizadas em pontos estratégicos, videoconferência [29] para comunicação gerente-gerente e gerentes-usuários, etc) e áudio (mensagens de alerta gravadas, beeps, e conversação on-line, etc).

As informações e dados relativos à rede são apresentados ao gerente graficamente, onde os elementos de redes são diferenciados por figuras (hosts, gateways, rede, etc) e cores diferentes que caracterizam os estados dos elementos de rede (up, down, não-gerenciavel, suspeito de intrusão, etc) de forma a simplificar a visualização e facilitar o acesso as informações de monitoramento.

Desta forma o agente de segurança irá se comunicar com o sistema de gerenciamento através do envio de notificações sobre a ocorrência de algum evento importante (detecção de intrusão), ou por pedido explicito do sistema de gerenciamento. No momento, não prevemos a utilização de SNMP para a comunicação entre as duas máquinas. O administrador de uma rede qualquer, pode visualizar, em determinado momento, o status de cada conexão que passa pelo agente de segurança, podendo filtrar aquelas que mais lhe interessem. Através de um sistema gradativo de coloração, ele poderá' ter uma idéia de quais conexões são consideradas normais (tendendo para verde) e quais são vistas pelo agente como potencialmente perigosas (tendendo ao vermelho). A cor seria derivada do parâmetro *grau de segurança* nos vetores armazenados na fila de permanência, fig.5.

Adicionalmente, se o gerente, através do monitoramento tradicional da rede (ou por notificação de alguém), conhecer alguma tentativa inovadora de intrusão que o agente não tenha detectado, ele poderá inseri-lo na base de dados do agente de segurança forma que este possa se atualizar.

5. Conclusões

O sistema de detecção de intrusão proposto pretende fornecer um instrumento ágil e versátil para atacar o problema cada vez mais preocupante de intrusão em computadores via rede. Ele se baseia no fato de que a maioria dos *hackers* segue um determinado padrão de atuação. Estes padrões são armazenados numa base de dados e utilizados dinamicamente no acompanhamento da atividade de um intruso, com o auxílio de sistemas especialistas e de redes neurais. Embora não tenhamos ainda dados contundentes sobre a eficácia do método, podemos prever uma boa chance de sucesso do sistema, pois ele não procura dar uma resposta definitiva para a avaliação das conexões, mas sim atribuirá um número que indicará a possibilidade dela ser oriunda de atividades de *hacking*.

Outra característica importante do método é a adaptividade. Os conhecimentos dos administradores podem ser facilmente introduzidos no sistema (e a rede re-treinada) de modo que novas informações importantes no processo de tomada de decisão podem ser incorporadas dinamicamente.

Finalmente, o sistema abre a possibilidade de se criar um ambiente automático de aprendizado que permitiria que houvesse um acúmulo significante na expertise na área de detecção e manipulação de intrusões.

6. Agradecimentos

Agradecemos à FAPESP e à CAPES pelo suporte dado a este projeto através de bolsas de estudos e equipamentos.

7. Referências

[1] R. Bace. *A New Look at Perpetrators of Computer Crime*. In Proc. 16th Department of Energy Computer Security Conference, 1994.

[2] P. Neumann & D. Parker. *A Summary of Computer Misuse Techniques*. In Proc. 12th National Computer Security Conference, pages 396-407, 1989.

[3] H.S.Javitz & A.Valdez. *The SRI IDES Statistical Anomaly Detector*. In Proc. 1991 IEEE Symposium on Research in Security and Privacy, Oakland, CA, May, 1991.

[4] J.R.Winkler & W.J.Page. *Intrusion and Anomaly Detection in Trusted Systems*. In Proc. Fifth Annual Computer Security Applications Conference, Tucson, AZ, Dec,1990. 115-124.

[5] D.E. Denning. *An Intrusion-Detection Model*. IEEE Trans. on Software Engg. Vol. SE-13, pp. 222-232, Feb.1987.

[6] T.F. Lunt et al. *IDES: A Progress Report*. In Proc. Sixth Annual Computer Security Applications Conference. Tuscon, AZ, Dec. 1990.

[7] T.F. Lunt et al. *A Real Time Intrusion Detection Expert System (IDES)*. Interim Progress Report, Project 6784, SRI International, May 1990.

[8] L.T. Heberlein et al. *A Network Security Monitor*. In Proc.1990 IEEE Symposium on Research in Security and Privacy, Oakland, CA, May, 1990, pp 296-304.

[9] L.T. Heberlein et al.. *Towards Detecting Intrusions in a Networked Environment*. In Proc. 14th DOE Conference on Computer Security. Concord, CA, May 1991, pp 17-47.

[10] L.T. Heberlein, K.N. Levitt and B. Mukherjee. *A Method to Detect Intrusive Activity in a Networked Environment*. In Proc. 14th National Computer Security Conference. Washington, DC. Oct. 1991, pp 362-371.

[11] S.R. Snapp et al. *DIDS (Distributed Untrusion Detection System) - Motivation, Archieteture, and an Early Prototype*. In Proc. 14th National Computer Security Conference. Washington, DC. Oct. 1991.

[12] S.R. Snapp et al. *A System for Distributed Intrusion Detection*. In Proc. IEEE COMPCON 91. San Francisco, CA. Feb.1991, pp 170-176.

[13] L.T. Heberlein, B. Mukherjee and K.N. Levitt. *Internet Security Monitor: An Intrusion Detection System for Large-Scale Networks*. In Proc. 15th National Computer Security Conference. Baltimore, MD. Oct. 1992.

[14] C. Ko et al. *Analysis of an Algorithm for Distributed Recognation and Accountability*. In Proc. First ACM Conference on Computer and Communicatons Security. Faifax, VA, Nov.1993. pp 154-164.

[15] S.G.Staniford-Chen. *Distributed Tracing of Intruders*. M.S.Thesis. Divison of Computer Science, University of California, Davis, CA. 1995.

[16] S.E.Smaha. *Haystack: An Intrusion Detection System*. In Proc. IEEE Fourth Aerospace Computer Security Applications Conference, Orlando FL, Dec. 1988.

[17]W.Venema. *TCP Wrapper: Networking Monitoring, Access Control and Booby Traps*. ftp://cert.sei.cmu.edu/pub/network_tools/tcp_wrapper.txt

[18] S.Garfinkel & G.Spafford. *Pratical UNIX Security*. O'Reilly and Associates, 1991.

[19] http://www.first.org/first/

[20] http://www.sei.cmu.edu/technology/cert.cc.html

[21] http://www.cs.purdue.edu/coast/coast.html

[22] Rumelhart, D.E.; McClelland, J.L. and The PDP Research Group. *Parallel Distributed Processing: Exploration in the Microstructure of Cognition*, MIT Press 1986.

[23] Carvalho, A.C.; Fairhurst, M.C. e Bisset, D.L. An integrated Boolean neural network for pattern classification, Pattern Recognition Letters, Vol. 15, páginas 807-813, agosto de 1994.

[24] Lupo, C.J. Defense Applications of Neural Networks, IEEE Communications Magazine, Vol. 27, No. 11, páginas 82-88, novembro de 1989.

[25] Wong A. J. Recognition of General Patterns Using Neural Networks, Biological Cybernetics, Springer-Verlag, Vol. 58, páginas 361-372, 1998.

[26] Oda, C. S., *Desenvolvimento de um Sistema Monitor Gráfico Baseado em Protocolo de Gerenciamento SNMP*, Tese de Mestrado - ICMSC/USP, 1994

[27] Oda, C. S., Moreira, E. dos S., *Representação Dinâmica de Objetos em Sistemas de Gerenciamento de Redes,* Anais do XIII SBRC, Belo Horizonte, 1995

[28] Oda, C. S., Moreira, E. dos S., *A XWindows-Based Network Monitor with Intelligent Topology Discovery*, Proceedings of The International Conference on Intelligent Infomation Management Systems, Washington DC, USA, 1995

[29] Lieira, J., Moreira, E. dos S., *Utilização de Som e Imagem em Sistemas de Gerenciamento de Redes de Computadores*, Anais do II Congresso de Informática e Telecomunicações de Mato Grosso, Cuiabá MT, 1995

[30] Stalling, W., *SNMP, SNMPv2 and CMIP - The Pratical Guide to Network-Management Standards*, Addison-Welley, 1993

NOTAS DO ICMSC

SÉRIE COMPUTAÇÃO

021/95 BEZERRA, L.A.F.; SANTANA, R.H.C.; SANTANA, M.J. - Sistema auxiliar de arquivos baseado em disco WORM para ambientar computacional distribuído.

020/95 NUNES, M.G.V.; HASEGAWA, R. - PROTEMA: intelligent tutoring systems for mathematics.

019/95 OLIVEIRA, M.C.; TURINE, M.A.S.; MASIERO, P.C. - A statechart - based model for hypertext.

018/95 PIMENTEL, M.G.C. - Alternative operations for browsing hypertext.

017/94 ROMEIRO, N.M.L.; CASTELO FILHO, A. - Análise Comparativa de Métodos Numéricos de equações algebrico-diferenciais.

016/94 MAGALHÃES, A.L.C.C.; SIQUEIRA, M.F.; OLIVEIRA, M.C.F. - Operadores de Euler na modelagem por fronteira: conceito, aplicação, estudos de casos.

015/94 ODA, C.S.; MOREIRA, E.S. - ASNMP graphical network monitor with automatic topology discovery.

014/94 FELIPE, L.S.G.; FRANCO, N.M.B. - Sobre a ordem de convergência para as equações integrais de volterra de segunda espécie tipo Abel com soluções não suaves.

013/94 PIMENTEL, M.G.C. - A framework for user-hypertext interaction.

012/94 TURINE, M.A.S.; MENDES, M.D.C.; NUNES, M.G.V. - TEGRAM: a geometry tutoring system based on Tangram.