Framework para Detecção e Filtragem de Alertas de Intrusão utilizando Redes Bayesianas

Everton Gamba Lermen
Gaspare Giuliano Elias Bruno
Setembro/2008

Roteiro

- Problema a ser resolvido
- Fundamentação teórica
- Solução proposta
- Implementação
- Resultados e avaliação
- Conclusões

Problema a ser resolvido

- Crescente aumento no volume de ataques a redes de computadores
- Ataques cada vez mais elaborados e de fácil execução
- Demanda constante de atualização (regras estáticas)
- Dificuldade de identificar novos ataques (pró-atividade)
- Elevado número de falsos positivos e falsos negativos

Problema a ser resolvido

Configuração de um ambiente de rede

Simulações de ataques e acessos normais

Elaboração da Rede Bayesiana

- Escolha da ferramenta de linguagem de programação.
- Identificação das variáveis.
- Escolha do algoritmo de aprendizagem e inferência.

Problema a ser resolvido

Espera-se

- Validar a utilização de Redes Bayesianas em Sistemas de Detecção de Intrusão.
- Diminuir o número de falsos positivos e falsos negativos.
- Propor uma abordagem diferenciada de utilização.

Fundamentação teórica

Sistemas de Detecção de Intrusão

- 1980 :: James P. Anderson
- 1984 - 1986 :: Dorothy Denning e Peter Neumann

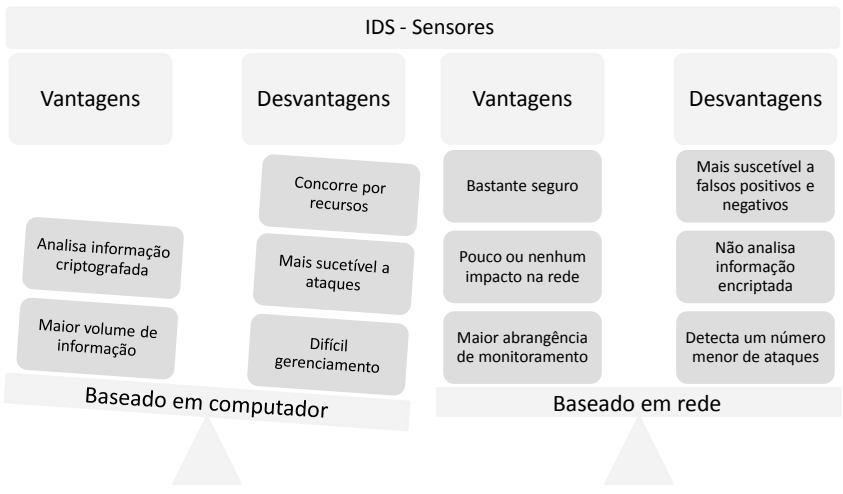
Redes Bayesianas

- Domínio de problemas que contenham incerteza.
- Pode ser descrita como um grafo acíclio dirigido.

Trabalhos relevantes

- *A Framework for an Adaptative Intrusion Detection System using Bayesian Network*. (Jemilli et al, IEEE 2007)
- *Bayesian Learning Networks Approach to Cybercrime Detection*. (Abouzakhar et al, PGNET 2003)

Fundamentação teórica



Fundamentação teórica

Fundamentação teórica

RB Construção

Independência condicional (Algoritmo PC)	Busca e pontuação (Algoritmo EM)
Dados completos	Dados Incompletos
Menor domínio do problema	Maior domínio do problema
Construção automática	Construção manual

Fundamentação teórica

RB Inferência

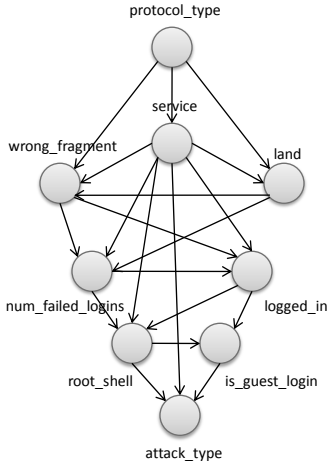
Exatos (Algoritmo Árvore de Junção)	Aproximados (Algoritmo de Monte Carlo)
Maior esforço computacional	Menor esforço computacional
Maior precisão	Precisão ligada diretamente ao tamanho da amostra

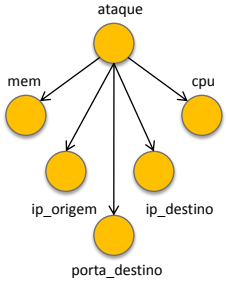
Solução proposta

- IDS
  - Baseado em rede
  - Deteção por anomalia
- Rede Bayesiana
  - naive-RB
  - Algoritmo EM
  - Algoritmo Árvore de Junção
- Variáveis consideradas
  - IP Origem
  - IP Destino
  - Porta Destino
  - Utilização Memória (%)
  - Utilização CPU (%)
  - Ataque
- Ambiente Desenvolvimento
  - Netica API C
- Saída
  - Identificação Acesso Normal
  - Identificação Ataque
  - Regra Estática (Snort)

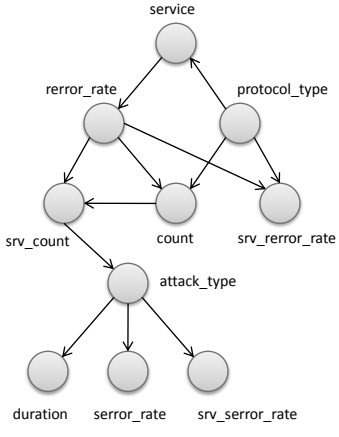
Solução proposta

Jemili et al (2007)

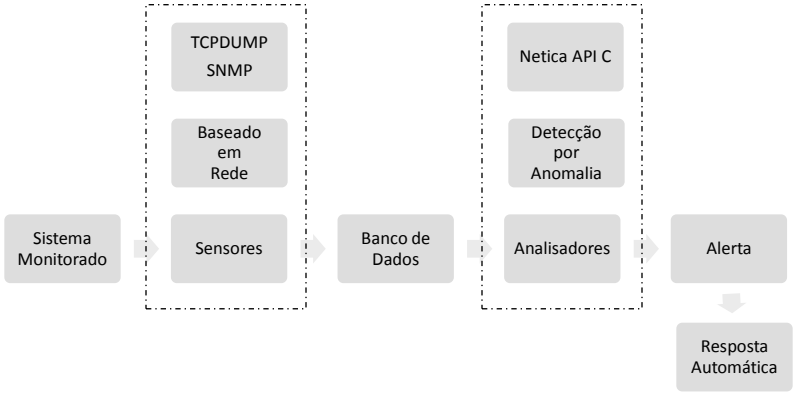




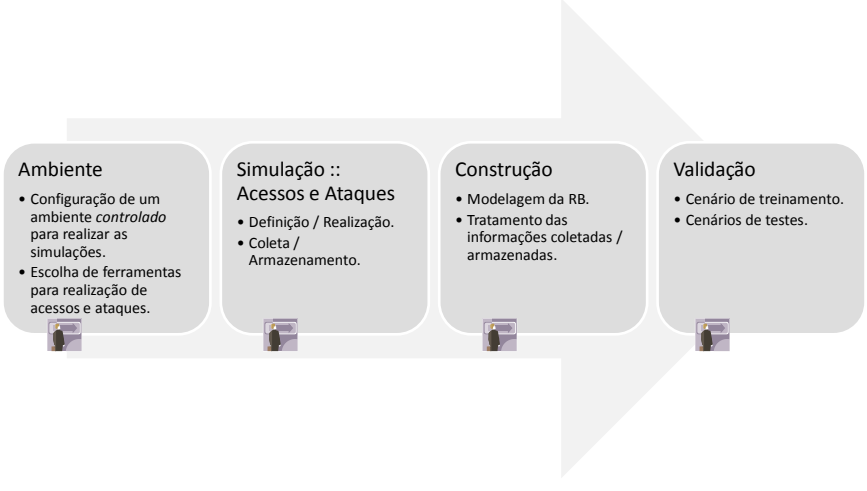
Abouzakhar et al (2003)



Solução proposta



Implementação

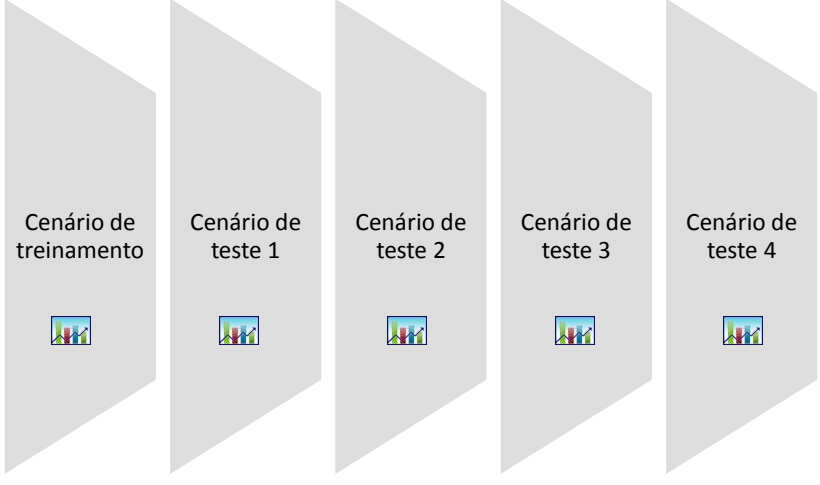


Resultados e avaliação

O comportamento do framework foi analisado quanto

- Ao número de falsos positivos e falsos negativos, em relação ao tráfego conhecido.
- Ao número de falsos positivos e falsos negativos, em relação ao tráfego desconhecido.

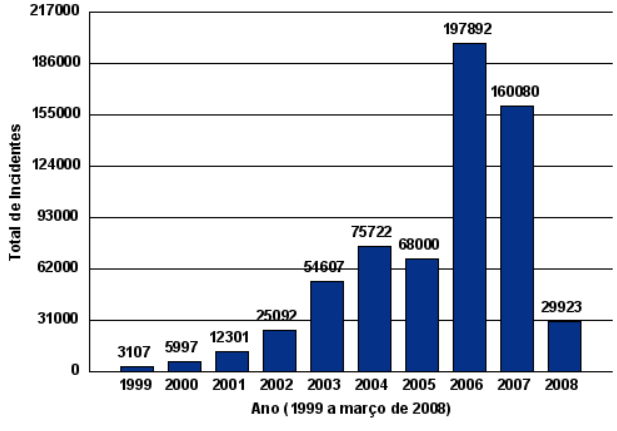
Resultados e avaliação



Conclusões

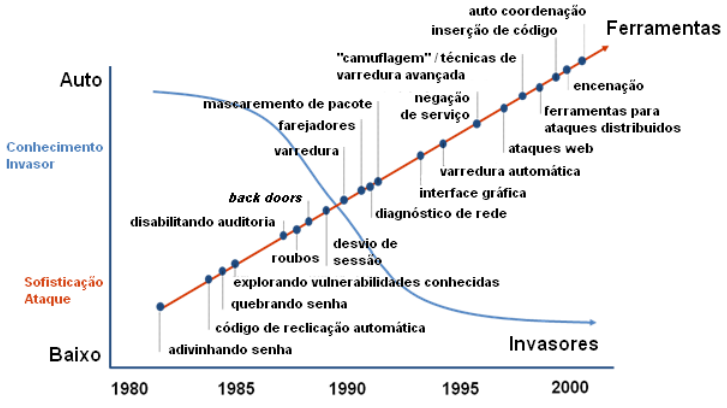
O framework proposto obteve	Limitações impostas	Melhorias
• 0% de falsos positivos e 3,15% de falsos negativos.	• Base de conhecimento – 3 ataques. • Ferramenta Netica – licença limitada. • Não considera o tempo de resposta. • Não ocorre realimentação da RB.	• Interface gráfica de controle e gerenciamento. • Ferramenta sem limitação – Weka/Java. • Aprimorar coleta de informações (MEM/CPU). • Agregar variáveis – detalhes do pacote e no. de conexão.

Total de incidentes reportados ao CERT.br por ano.



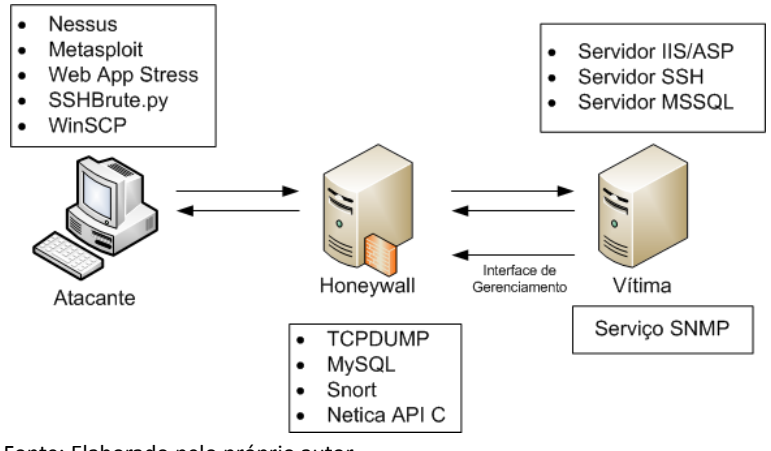
Fonte: CERT.br, 2008.

Sofisticação de ataque *versus* Conhecimento do invasor



Fonte: CERT/CC, 2002.

Ambiente



Fonte: Elaborado pelo próprio autor.

Simulação

- Nessus
- 192.168.47.1-5
- 100 conexões

Ataque Servidor IIS/ASP

- SSH Brute.py
- 192.168.47.1-5
- 100 senhas
- 100 conexões

Ataque Servidor SSH

- Nessus
- 192.168.47.129
- 1 conexão

Ataque Servidor MSSQL

- Web App Stress
- 192.168.47.5-9
- 50 conexões

Acesso Normal Servidor IIS/ASP

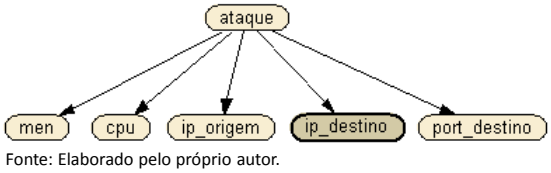
- WinSCP
- 192.168.47.5-9
- 50 conexões

Acesso Normal Servidor SSH

- TCPDUMP
- SNMP
- MySQL

Coleta / Armazenamento

Construção



Fonte: Elaborado pelo próprio autor.

Variável	Faixa de valores (1ª. conversão)	Faixa de valores (2ª. conversão)	Descrição
MEM	1 a 35	1 a 35	Baixa
	36 a 65	36 a 50	Média
	66 a 100	51 a 100	Alta
CPU	1 a 25	1 a 25	Baixa
	26 a 50	26 a 50	Média
	51 a 100	51 a 100	Alta

Fonte: Elaborado pelo próprio autor.

Validação

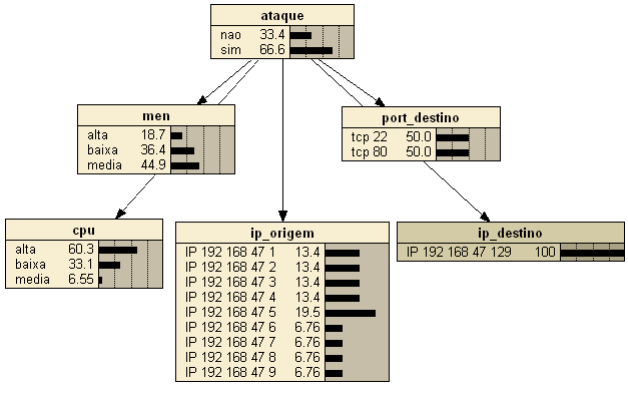
Cenário de Treinamento

- 200 ataques
- 100 acessos normais

Cenários de Testes

- Ataque conhecido partindo de um computador conhecido por realizar ataques.
- Acesso conhecido partindo de um computador conhecido por realizar acesso normal.
- Ataque conhecido partindo de um computador conhecido por realizar acesso normal.
- Ataque e computador desconhecidos.

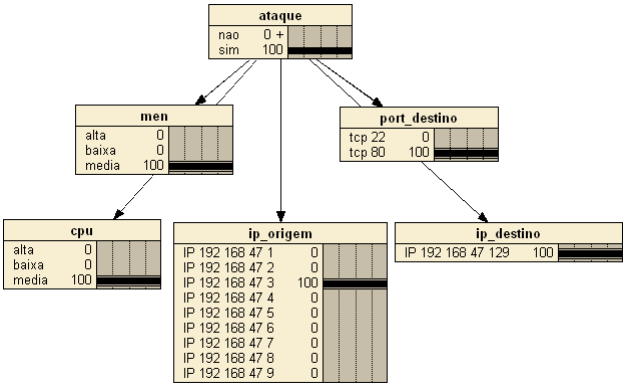
Cenário de Treinamento



Fonte: Elaborado pelo próprio autor.

Cenário de Teste 1

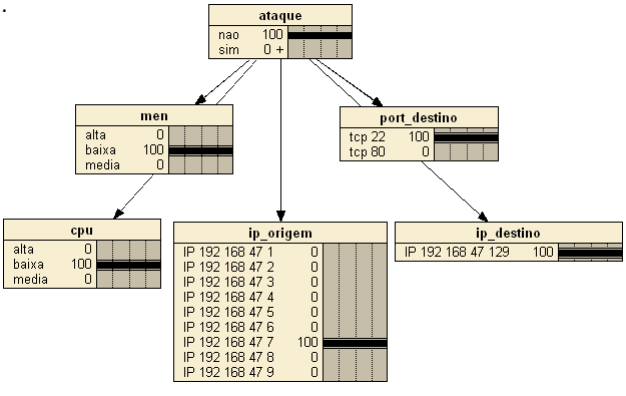
Ataque conhecido partindo de um computador conhecido por realizar ataques.



Fonte: Elaborado pelo próprio autor.

Cenário de Teste 2

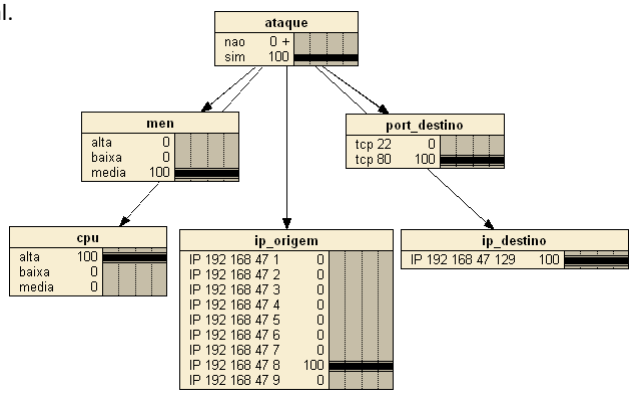
Acesso conhecido partindo de um computador conhecido por realizar acesso normal.



Fonte: Elaborado pelo próprio autor.

Cenário de Teste 3

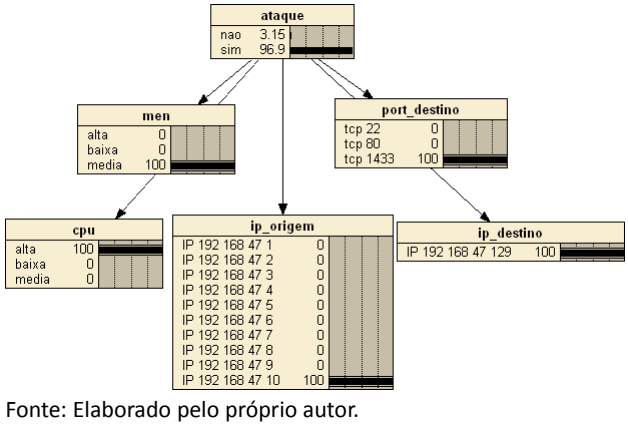
Ataque conhecido partindo de um computador conhecido por realizar acesso normal.



Fonte: Elaborado pelo próprio autor.

Cenário de Teste 4

Ataque e computador desconhecidos.



Fonte: Elaborado pelo próprio autor.