Sistema para Identificação de Alertas Falso Positivos por meio de Análise de Correlacionamentos e Alertas Isolados

A. A. A. Silva e A. E. Guelfi

Abstract - **Among the problems encountered in the use of Intrusion Detection Systems (IDS), can be listed the difficulty to understand the strategies of the attacks and the huge amount of alerts generated. The analysis of alerts can be difficult because many alerts are kind of false positive and eventually lead to false results. Correlate alerts issued by an IDS based on its causes and consequences is a way of highlighting strategies and establish links between the attacks. However, many alerts, called isolates did not correlate to others. Within this context, the ideal is to check the veracity of alerts (isolated and correlated) through other sources (cross-correlation), for example, logs taken from the operating system. Therefore, the objective is to propose an Event Analysis System (SAE) which allows multi-correlate security alerts information from a SDI with the operating system logs and analyze the alerts isolates for the identification of false positives alerts.**

Keywords **– Intrusion Detection Systems (IDS), alerts correlation, isolated alerts and false positive alerts.**

I. Introdução

Uma invasão pode ser entendida como um conjunto de ações com o objetivo de comprometer a integridade, confidencialidade ou a disponibilidade de um sistema computacional ou de seus recursos. É causada por usuários ou programas que tentam obter acesso ou fazer uso de sistemas de forma não autorizada, ou que estejam tentando ultrapassar os limites de seus privilégios [1].

Entre as ferramentas usadas para monitoramento de invasões estão os Sistemas de Detecção de Intrusão (SDI). Muitos dados são gerados na monitoração realizada pelo SDI, e consequentemente, há dificuldades na análise destas informações. Este tipo de problema está diretamente ligado à emissão de alertas Falso Positivos (FP).

Uma forma de facilitar a análise dos alertas é levantar os correlacionamentos entre eles, buscando características similares que possam ser relacionadas [2][3][4]. Com isso eliminam-se dados redundantes e podem ser descobertos padrões de ataque [5][6][7][8][15].

Um dos problemas da correlação de alertas é que seus resultados dependem, em grande parte, da qualidade dos alertas detectados. Portanto, o nível de precisão das ferramentas de detecção é um fator preponderante para que ocorram correlações válidas e, como consequência, um melhor entendimento dos ataques. Falhas na detecção ou altos índices na emissão de FP podem comprometer toda a análise [7][9].

A multi-correlação, integração dos alertas gerados pelo SDI com informações vindas de fontes diferentes como ferramentas de monitoração ou *logs* do sistema operacional, podem melhorar o nível da detecção dos alertas [2][7].

Porém, há alertas que mesmo passando por processos de correlação não se relacionam a nenhum outro. Estes alertas são chamados alertas isolados e podem consistir de ataques individuais sem relação alguma com outros eventos. Não fazem, portanto, parte de uma estratégia maior de ataque. Muitos destes alertas podem ser também FP [5][8].

Trabalhos que usaram a multi-correlação [2][7][9], não fazem uma análise detalhada da influência dos alertas isolados nos índices de emissão de alertas FP.

Portanto, o objetivo deste trabalho é propor um sistema de análise de eventos (SAE) que possibilite multi-correlacionar alertas de segurança da informação de um SDI com *logs* do sistema operacional e verificar a influência dos alertas isolados visando a identificação dos alertas FP de um SDI.

Este artigo está organizado em seções da seguinte forma: a seção II mostra a base teórica sobre correlacionamento de alertas, a seção III compara os principais trabalhos pesquisados, a seção IV mostra a arquitetura e as atividades implementadas em cada sub-módulo do SAE, a seção V mostra o ambiente montado para validação do SAE e a coleta de resultados e a seção VI mostra um resumo do trabalho e as conclusões levantadas.

II. Correlacionamento de Alertas

As técnicas de correlação de alertas de SDI podem ser divididas em três categorias: baseadas em regras, baseadas em similaridades e baseadas em causas e conseqüências (*Prerequisites and Consequences* – PC) [7].

O método baseado em regras requer algum conhecimento prévio sobre o ataque. A máquina alvo passa por uma preparação denominada treinamento. Com isso, torna-se capaz de detectar a vulnerabilidade treinada com precisão, sendo esta sua grande vantagem. Exemplos do uso deste método podem ser observados nos trabalhos de [2] e de [10].

Uma das dificuldades do método baseado em regras é a geração de enormes volumes de dados, aliada ao alto poder computacional que isto exige. Outro fator limitante é que este método funciona apenas para vulnerabilidades conhecidas. Além disso, não há garantias de que um eventual atacante siga

o cenário estabelecido no treinamento, ou que simples variações nos algorítimos dos programas usados para o ataque não acabem passando despercebidas pelo sistema de detecção.

O método baseado em similaridades tem na correlação por análise estatística seu principal expoente, baseando-se no acúmulo de informações colhidas ao longo do tempo e nas observações dos desvios ocorridos no processo. Esta técnica permite a detecção de ataques totalmente novos.

Por exemplo, em [11] demonstra-se um sistema de detecção de anomalias caracterizado pela monitoração de diversos parâmetros simultâneos. Já em [12], apresenta-se uma proposta de correlação probabilística para SDI, com base na fusão de dados e multi-sensores.

Porém, o método baseado em similaridades pode não detectar atividade anômala escondida em processos normais, se esta for executada em níveis muito baixos. Além disso, justamente por analisar os processos normais, apenas reportando desvios, não é muito adequado para descoberta das causas dos ataques [5].

O método PC que é baseado em causas (condições para um ataque ser verdadeiro) e consequências (resultados da exploração de uma causa) relaciona alertas com base nestas informações e é adequado para a descoberta de estratégias dos ataques. Tanto as causas como as consequências são compostas por informações referentes aos atributos dos alertas (características específicas pertencentes a cada alerta). Estas informações são chamadas tuplas.

Para estas conexões serem válidas, um alerta (preparatório) precisa ter em suas consequências, ao menos uma tupla que se repita nas causas do segundo (resultante). Ou seja, o alerta preparatório contribui para a construção do resultante e, portanto, pode ser correlacionado. Neste tipo de ligação, ilustrado na Fig. 1, obrigatoriamente, o *timestamp* do alerta preparatório precisa ser anterior ao do alerta resultante [5][6][7].

Com o intuito de diminuir a complexidade, as conexões podem ser visualizadas em grafos compostos por nós representando os alertas e setas representando as correlações entre estes, conforme ilustra a Fig. 2.

Porém, existem alguns aspectos negativos no método PC, como a dificuldade na obtenção das causas e consequências [13] ou o fato de ataques perdidos dificultarem a correlação, que podem ser melhorados com o uso de técnicas complementares.

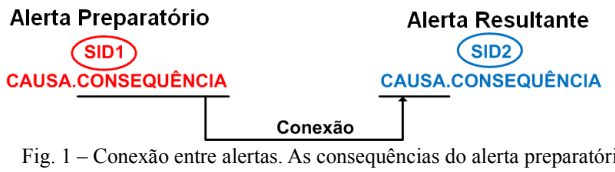

Fig. 1 – Conexão entre alertas. As consequências do alerta preparatório (SID1) são ligadas às causas do alerta resultante (SID2).

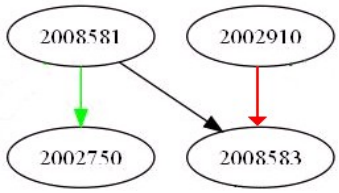

Fig. 2 – Exemplo de grafo com 4 alertas correlacionados. Os números dentro dos nós representam a numeração dos alertas de SDI.

Uma forma complementar à correlação pode ser vista em [14], que recomenda a instalação de sensores que trabalhem em cooperação, monitorando o ambiente para minimizar falhas na detecção.

Uma das maiores dificuldades no uso de sensores que trabalham em cooperação com um SDI é o mapeamento entre os *logs* destes sensores e os alertas do SDI, devido às diferenças de conteúdo e de formato nos dados.

Há duas técnicas para realização de conexões entre os alertas de SDI e os *logs* obtidos de outras fontes: análise descendente e análise ascendente [2][4].

Na técnica descendente, a partir da análise do comportamento de um ataque conhecido verifica-se em outros *logs* (sistema operacional por exemplo) onde há traços de sua atuação com base no *timestamp* do alerta. Este tipo de análise é útil para traçar evidências sobre as estratégias dos eventos mapeando o caminho realizado pelo ataque até sua fonte.

A técnica ascendente procura descobrir ataques a partir da análise de diversos *logs*. Assim que uma anomalia é detectada em um destes *logs,* os outros são checados com base no *timestamp*. Apesar de mais custosa computacionalmente, esta técnica permite a detecção de ataques novos.

Neste ponto surge outra vulnerabilidade do método PC: um evento malicioso não precisa necessariamente passar por etapas sucessivas para ocorrer. Diante disto, um ataque direto sobre um serviço, apesar de detectado, pode não ser correlacionado. No método PC, sempre que um alerta não é correlacionado a outros, trata-se de um alerta isolado [5][6][8].

III. Trabalhos Relacionados

Em [6] o objetivo é a descoberta de estratégias entre diferentes ataques usando o método de correlação PC. Uma das dificuldades é entender a forma como as tuplas utilizadas na composição das causas e consequências são definidas.

De forma complementar, em [5] apresentam-se técnicas úteis para a análise dos grafos de correlacionamento no método PC. O objetivo principal é entender as estratégias dos ataques por meio da melhoria dos grafos, pois em algumas situações há dificuldade no entendimento das estratégias dos ataques, sendo este um indicativo da necessidade de validação dos resultados gerados pelo método PC.

Em [2] tem-se como objetivo testar a eficácia do uso da multi-correlação com o método por regras. Bons resultados na precisão das detecções são conseguidos com a multi-correlação e reforçam a ideia de que os resultados de um SDI devem ser validados.

Em [8], que também utiliza o método PC para

correlacionamento, os alertas isolados são citados e descartados sem análise. Os próprios autores chamam a atenção para a periculosidade desta ação já que podem ser eliminados alertas Verdadeiro Positivos (VP) neste processo.

Em [7] apresenta-se um trabalho complementar que usa o método PC em conjunto com o levantamento de eventos do sistema operacional. Estes *logs* são então correlacionados com os alertas de SDI para a descoberta de possíveis ataques Falso Negativos (FN) por análise inversa.

Os principais trabalhos pesquisados mostram-se complementares, convergindo para a necessidade da utilização da multi-correlação. Entre outras constatações podem ser listadas a falta de tratamento dada aos alertas isolados e a complexidade no levantamento de tuplas de causas e consequências. A Tabela 1 mostra a comparação do trabalho desenvolvido neste artigo com os principais trabalhos pesquisados e pode-se perceber que a proposta deste trabalho tem como principais contribuições combinar a multi-correlação com a análise dos alertas isolados para diminuição de alertas FP, usar duas formas de análise de *logs* (ascendente e descendente) e adotar tuplas padronizadas para elaboração de causas e consequências dos alertas.

TABELA 1
COMPARAÇÃO ENTRE TRABALHOS RELACIONADOS

Metodologias	Trabalhos					
	[6]	[5]	[2]	[8]	[7]	*
Método	PC	PC	Regras	PC	PC	**PC**
Tuplas padronizadas						**X**
Estratégias dos ataques	X	X		X	X	**X**
Análise inversa					X	
Descoberta FN					X	
Diminuição FP	X	X	X	X	X	**X**
Logs de outras fontes			X		X	**X**
Alertas isolados						**X**
Análise ascendente			X			**X**
Análise descendente			X		X	**X**

(*) TRABALHO DESENVOLVIDO PELOS AUTORES DESTE ARTIGO

IV. ARQUITETURA DO SAE

A arquitetura do SAE possui quatro módulos funcionais: Conversor, Atualizador, Correlacionador e Calculador (vide Fig. 3).

Conversor (a): Este módulo tem como objetivo manipular os dados de entrada do sistema (assinaturas de SDI, alertas detectados e *logs* do sistema operacional).

Atualizador (b): Este módulo tem como objetivo controlar os dados a serem utilizados pelo sistema.

Correlacionador (c): Este módulo tem como objetivo realizar as operações necessárias (correlação, mapeamento, identificação de FP e identificação de isolados) para a análise dos dados presentes no módulo Atualizador (b).

Calculador (d): A função deste módulo é analisar e comparar os índices de alertas FP com base nos resultados produzidos pelo módulo Correlacionador (c).

Para avaliar a ocorrência de alertas FP durante a execução do SAE, três etapas (FP1, FP2 e FP3) são definidas para levantamento de resultados:

a) FP1: após o correlacionamento dos alertas de SDI, procura identificar eventuais alertas isolados e alertas FP referentes a operações normais do sistema.

b) FP2: procura identificar eventuais alertas isolados resultantes da etapa FP1.

c) FP3: procura identificar alertas isolados e alertas FP após o multi-correlacionamento e levanta os índices de emissão de FP entre os alertas isolados.

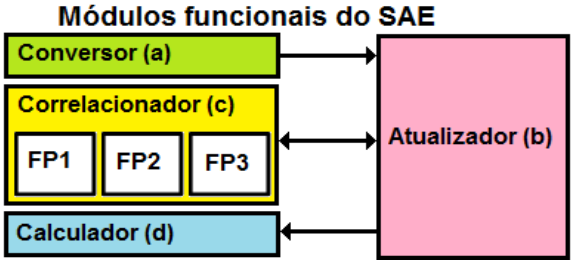

Fig. 3 – Arquitetura do SAE

A. ATIVIDADES DO MÓDULO CONVERSOR (a)

O módulo Conversor (a) executa a atividade de conversão e classificação de campos: que reorganiza as assinaturas do SDI em uma tabela de banco de dados. Os campos desta tabela são classificados em campos principais e campos secundários, que por sua vez podem ser dos seguintes tipos: causas (P), consequências (C) ou ambos (PC) e servem para uso do módulo Correlacionador (c).

Os campos principais contém as principais informações do alerta, indicando informações de origem e destino do ataque, são todos do tipo PC e estão descritos na Tabela 2.

Os campos secundários: contém atributos extraídos das assinaturas que caracterizam detalhes específicos sobre os alertas diferenciando-os uns dos outros como CLASSTYPE, MSG, REFERENCE, entre outros.

TABELA 2
CAMPOS PRINCIPAIS DOS ALERTAS

Campo	descrição
proto	protocolo do alerta.
origem	IP de origem do alerta.
porta_o	porta de origem do alerta.
direcao	sentido do alerta.
destino	IP de destino do alerta.
porta_d	porta de destino do alerta.
SID	número de identificação único do alerta.

B. Atividades do Módulo Correlacionador (c)

O módulo Correlacionador (c) é responsável pela criação da tabela padrão de tuplas, da tabela de correlação e pela execução dos sub-módulos FP1, FP2 e FP3.

Tabela padrão de tuplas: mostrada na Tabela 3, contém todas as tuplas que podem servir como causas ou consequências dos alertas detectados.

Tabela de correlação: contém os alertas detectados (SID) nas linhas e as tuplas em conjunto com alguns atributos nas colunas. A partir da análise dos campos secundários CLASSTYPE, REFERENCE e MSG de cada alerta, são preenchidas as colunas com os atributos SERVICO (principal serviço do alerta), VULNERAB (principal vulnerabilidade do alerta), EXCLUSIVO (sistema operacional exclusivo do alerta) e VIA (modo pelo qual a vulnerabilidade é explorada, *download*, *trojan*, vírus, etc.)

As tuplas também são preenchidas com "P" (quando correspondem a uma causa), "C" (quando correspondem a uma consequência) ou "PC" (quando reúnem características de causa e consequências).

Para efetivar uma correlação entre diferentes alertas com o uso da tabela de correlação, um alerta que tenha uma determinada tupla preenchida com o valor "C" ou "PC" pode ser ligado a outro alerta que tiver a mesma tupla preenchida com o valor "P" ou "PC", desde que o *timestamp* do primeiro alerta (preparatório) seja anterior ao do segundo (resultante).

Por exemplo, o alerta SID=15930 têm as seguintes tuplas com valor C: EXE_ADM, EXE_USU, IND_SER e IND_REC, podendo desta forma, ser correlacionado como preparatório a quaisquer alertas que tenham o valor P ou PC em alguma destas mesmas tuplas, caso do alerta resultante SID=1594 mostrado em destaque na Tabela 4.

Como resultado, o módulo Correlacionador (c) emite um grafo denominado c1.

TABELA 3
TABELA PADRÃO DE TUPLAS

tuplas	descrição
inv_adm	invasao-administrador
inv_usu	invasao-usuario
exe_adm	execucao-administrador
exe_usu	execucao-usuario
ind_sis	indisponibilidade-sistema
ind_red	indisponibilidade-rede
ind_ser	indisponibilidade-servico
ind_inf	indisponibilidade-informacao
ind_rec	indisponibilidade-recurso
exi_vul	existe-vulnerabilidade
exi_ser	existe-servico
exi_hos	existe-host
ace_inf	acesso-informacao
ace_rec	acesso-recurso
ace_ser	acesso-servico
dis_inf	disponibilidade-informacao

TABELA 4
PARTE DA TABELA DE CORRELAÇÃO

	tuplas										
SID	inv adm	inv usu	exe adm	exe usu	ind sis	ind red	ind ser	ind inf	ind rec	exi vul	exi ser
15930			C	C			C		C	P	P
1594			P							P	P

LIGAÇÃO (EM DESTAQUE) DO ALERTA PREPARATÓRIO SID=15930 COM O ALERTA RESULTANTE SID=1594

C. Atividades do Sub-Módulo FP1

O sub-módulo FP1 executa as atividades de corte de ligações, acréscimo de ligações, identificação de isolados-FP1 e identificação de FP-FP1.

Corte de ligações: é um filtro no grafo (c1), pois muitas ligações não são verdadeiras ou estão num sentido invertido, podendo ser descartadas do grafo se nenhum dos seguintes critérios for satisfeito:

a) o conteúdo do campo MSG do alerta preparatório tem relação com o campo MSG do alerta resultante.

b) o IP de origem (ou destino) dos alertas indica um endereço específico, como o endereço de destino (ou origem) do outro alerta.

c) os tempos de execução dos alertas (*timestamp*) são próximos. Apesar de não ser uma característica definitiva, afinal alertas com com tempos distantes podem caracterizar relações preparatórias, neste caso a possibilidade de relação é maior.

d) o campo SERVICO do alerta preparatório for igual ou tiver relação com o campo SERVICO do alerta resultante;

e) o conteúdo do campo VULNERAB do alerta preparatório indicar algum tipo de relação com o campo VULNERAB do alerta resultante.

f) o conteúdo do campo VIA do alerta preparatório for igual ou tiver relação com qualquer outro campo do alerta resultante.

g) dentro da rede local os alertas vem da mesma origem.

h) o campo PORTA_O de um alerta preparatório está dentro de uma faixa que corresponda a uma faixa do campo PORTA_D de um alerta resultante.

Acréscimo de ligações: há também a possibilidade de aparecerem ligações entre nós que não estavam ligados inicialmente. Estas ligações são feitas manualmente dentro dos mesmos critérios do corte de ligações e adicionalmente com base em observações do possível relacionamento do conteúdo dos campos.

Identificação de isolados-FP1: o critério usado para identificação dos alertas isolados-FP1 é a sua não ligação a outro alerta. No grafo (c1) os alertas isolados aparecem como nós sem setas. Como resultado, os alertas isolados-FP1 são eliminados do grafo.

Identificação de alertas FP-FP1: o critério usado para identificar os alertas FP-FP1 é a eliminação de sequências de nós que representem operações nitidamente normais do sistema, como por exemplo requisições ICMP. Como resultado os alertas FP-FP1 são eliminados e um novo grafo é gerado: grafo (FP1).

D. Atividades do Sub-Módulo FP2

O sub-módulo FP2 executa apenas a atividade de identificação de isolados-FP2.

Identificação de isolados-FP2: o critério para um alerta ser considerado isolado continua o mesmo: a sua não ligação a outro alerta. Eventuais alertas isolados remanescentes da etapa FP1 são eliminados e um novo grafo é gerado: grafo (FP2).

E. Atividades do Sub-Módulo FP3

O sub-módulo FP3 executa as atividades de mapeamento, análise ascendente/descendente, validação de nós, validação de ligações, identificação de FP-FP3, identificação de isolados-FP3 e mapeamento dos isolados FP1-FP2-FP3.

Mapeamento: é o correlacionamento dos alertas de SDI com os *logs* do sistema operacional por meio de uma tabela denominada tabela de mapeamento, que contém os alertas de SDI ligados aos *logs* do sistema operacional pelo *timestamp*.

Análise ascendente/descendente: para complementar o processo de mapeamento, são utilizadas técnicas de análise ascendente e descendente, que basicamente comparam os serviços e operações oriundas das assinaturas do SDI com as informações vindas da detecção dos *logs* do sistema operacional.

A Tabela 5 descreve os campos que são utilizados na tabela de mapeamento para realização da análise ascendente / descendente.

Para implementação da análise ascendente e descendente, dois modos são utilizados: validação de nós e validação de ligações.

Validação de nós: verifica-se se os alertas de SDI possuem eventos (processos, arquivos ou operações de registro) que tenham confirmação no sistema operacional.

Os critérios adotados para validar os alertas são os seguintes:

a) 1.1: quando a origem e/ou destino das ligações é a máquina *gateway.*

b) 1.2: quando o serviço principal do alerta está inserido nos campos PROCESS_NA, IMAGE_PATH ou COMMAND_LI de mesmo *timestamp.*

c) 1.3: quando a operação principal do alerta está inserida nos campos OPERATION ou DETAIL de mesmo *timestamp*;

d) 1.4: quando o serviço secundário do alerta está inserido nos campos PROCESS_NA, IMAGE_PATH ou COMMAND_LI de mesmo *timestamp.*

e) 1.5: quando a operação secundária do alerta está inserida nos campos OPERATION ou DETAIL de mesmo *timestamp.*

f) 1.6: quando a operação principal do alerta está inserido nos campos PROCESS_NA, IMAGE_PATH ou COMMAND_LI de mesmo *timestamp.*

g) 1.7: quando o serviço principal do alerta está inserido nos campos OPERATION ou DETAIL de mesmo *timestamp.*

h) 1.8: quando a operação secundária do alerta está inserido nos campos PROCESS_NA, IMAGE_PATH ou COMMAND_LI de mesmo *timestamp.*

i) 1.9: quando o serviço secundário do alerta está inserido nos campos OPERATION ou DETAIL de mesmo *timestamp.*

j) 1.10: quando o serviço principal ou secundário do alerta está presente nos campos PROCESS_NA, IMAGE_PATH, COMMAND_LI, DETAIL ou PATH de *timestamp* diferente.

Os critérios 1.1 a 1.9 são enquadrados como análise descendente. Já o critério 1.10 é enquadrado como análise ascendente e é o único com um caráter excludente, ou seja, mostra os alertas que devem ser descartados do grafo já que evidencia os serviços que não fazem parte das configurações da máquina usada como *gateway.*

Validação de ligações: verifica se os eventos presentes no sistema operacional confirmam as eventuais ligações entre os nós. Os critérios adotados para validar as ligações estão listados abaixo:

a) 2.1: quando o Número Identificador do Processo (PID) do nó resultante é um processo filho do processo do nó preparatório.

b) 2.2: quando alguma informação listada no conteúdo dos campos DETAIL dos nós preparatórios e resultantes tem alguma informação coincidente.

c) 2.3: quando o conteúdo do campo PATH do nó resultante é igual ou complementar ao do nó preparatório.

Ao final da etapa de validação de nós e validação de ligações, um novo grafo (FP3) é emitido. Neste grafo estão os alertas de SDI já mapeados aos *logs* do sistema operacional, com os alertas que não se enquadraram no critério 1.10 excluídos.

Identificação FP-FP3: os alertas não validados no modo de validação de nós e as ligações não validadas no modo de validação de ligações são marcadas como FP-FP3 e eliminadas do grafo (FP3).

Identificação isolados-FP3: com a eliminação dos alertas FP-FP3, abre-se a possibilidade da identificação e eliminação de novos alertas isolados do grafo (FP3).

Mapeamento dos isolados FP1, FP2 e FP3: a próxima etapa envolve os alertas isolados remanescentes das etapas FP1, FP2 e FP3, que são analisados apenas no modo de validação dos nós.

TABELA 5
Descrição dos Campos da Tabela de Mapeamento

Campo	Descrição	Local
process_na	nome do processo executado	*logs*
image_path	caminho completo do processo executado	*logs*
event_class	a classe (arquivo, registro ou processo) do evento	*logs*
operation	a operação específica do evento (leitura, gravação, criação, etc)	*logs*
path	o caminho do recurso ou do registro do sistema a que um evento faz referência	*logs*
detail	informações específicas adicionais de um evento	*logs*
pid	o número identificador do processo executado	*logs*
parent_id	o número identificador do processo gerador do processo atual	*logs*
command_li	linha de comando do processo executado	*logs*
process1	serviço principal do alerta	SDI
process2	serviço secundário do alerta	SDI
operation1	operação principal do alerta	SDI
operation2	operação secundária do alerta	SDI

Como resultado o módulo Correlacionador (c) envia todos os dados para o módulo Atualizador (b) que os envia para o controle do módulo Calculador (d).

V. Validação da Proposta e Ambiente de Testes

Para validação do SAE e obtenção de resultados, um ambiente de testes, descrito na sequência, é utilizado.

A rede local é cabeada e sua estrutura é composta por uma máquina onde está instalado o SAE, funciona como *gateway* da rede local permitindo o acesso à *Internet* e registra exclusivamente os alertas de SDI e os *logs* do sistema operacional. A Tabela 6 e a Tabela 7 listam detalhes do experimento.

A representatividade da rede montada no experimento, ilustrada na Fig. 4, está baseada em sua diversidade: várias máquinas, configurações, protocolos e serviços diferentes em execução, vários sistemas operacionais e acesso livre à *Internet*.

Os resultados do experimento são conseguidos com a análise dos sucessivos grafos e tabelas emitidos a cada etapa. As quantidades de alertas e de correlações são contabilizadas separadamente entre as que tem como origem ou destino o *gateway* (GAT) e as que não tem (N-GAT). Todas estas informações, compiladas na Fig. 5 e na Tabela 8, levam às seguintes constatações:

a) a concentração de correlações no *gateway* (coluna GAT) e o alto índice de FP entre os isolados (P_FP_isolados=72,41%) são tendências observadas também nos outros trabalhos pesquisados.

b) a repetição de setas nas correlações (C_bruto_repetido, Q_CF_repetido e Q_CVP_repetido) reflete apenas a redundância de alguns ataques e não influi nos resultados apresentados.

c) a não ocorrência de alertas isolados após a etapa FP1 (Q_isolados=0 nas etapas FP1 e FP2) é normal visto que os alertas isolados são eventuais.

TABELA 6
Serviços Executados no Experimento

Serviços	Máquina origem	Destino
Navegação na *Internet* (*browser)*	14 e 23	Portais
Acesso remoto (VNC)	59 e 106	Portao
Peer-to-peer (Bitcomet)	59	*Internet*
E-mail recebimento e envio (Winconnection)	59	*Internet*
Teste de ataque completo (Fast-Track)	65	Portao
Teste de ataque completo (Nessus)	204	*Portao*

TABELA 7
Aplicativos Usados no Experimento

Componentes	Aplicativo	Tempo (m)	Detalhes
SAE	Visual FoxPro		
SDI	Snort	19	13113 assinaturas habilitadas
Detecção *logs*	Procmon	19	752851 *logs* gerados
Grafos	Graphviz		

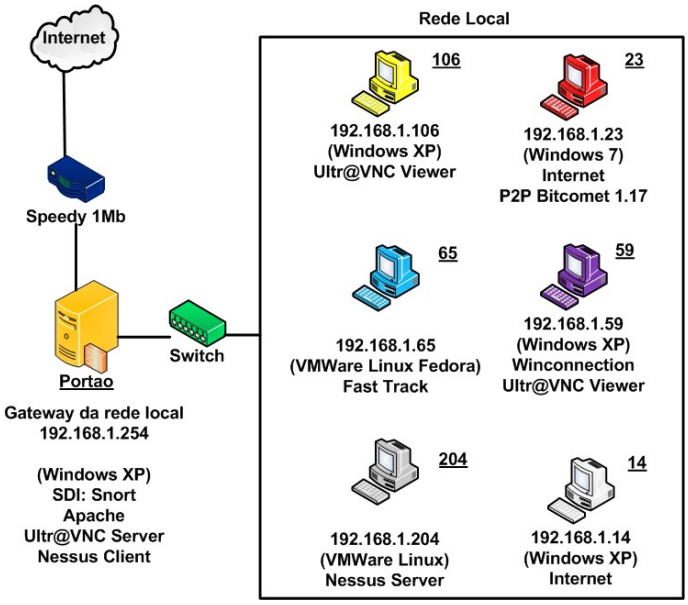

Fig. 4 – Detalhamento da rede local com os endereços IP e os serviços

VI. Conclusão

Este trabalho apresentou uma proposta para levantamento padronizado de causas e consequências dentro do método de correlação PC em conjunto com critérios de multi-correlação e de análise (ascendente/descendente) de correlacionamentos. para identificação de alertas FP por meio de tabelas e grafos.

Quatro conclusões são levantadas dos resultados do experimento:

a) a concentração de alertas com origem ou destino ao *gateway* (coluna GAT) reforça a escolha do local dos sensores de detecção (a máquina *gateway*).

b) a não ocorrência de alertas isolados após a etapa FP1, apesar de normal, confirma que o resultado do processo de correlação foi satisfatório, ratificando a adoção de tuplas padronizadas e de uma tabela de correlação no SAE. Chega-se a esta conclusão verificando-se que apesar das altas taxas FP das etapas FP1 (P_FP=21,08%) e FP3 (P_FP=44,72%), durante todo o experimento, nenhum alerta VP foi correlacionado exclusivamente como resultante de um FP.

c) a ocorrência de alertas VP entre os alertas isolados (Q_VP_isolados=8 o que indica que os alertas isolados são 10,52% de todos os isolados) confirma a hipótese de que o descarte destes alertas é prejudicial.

d) o incremento no percentual de alertas FP que acontece da etapa FP1 (P_FP=21,08%) para a etapa FP3 (P_FP=44,72%) tem como motivo a multi-correlação, fato que ratifica o critério criado para o mapeamento.

Como sugestão de aprimoramento do trabalho, há necessidade de transformar algumas atividades de análise que requerem interpretação do usuário (tabela de correlação e mapeamento) em atividades automáticas e pesquisar a possibilidade de uso do SAE em tempo real. A precisão dos resultados também pode ser aprimorada se a multi-correlação for estendida à rede local.

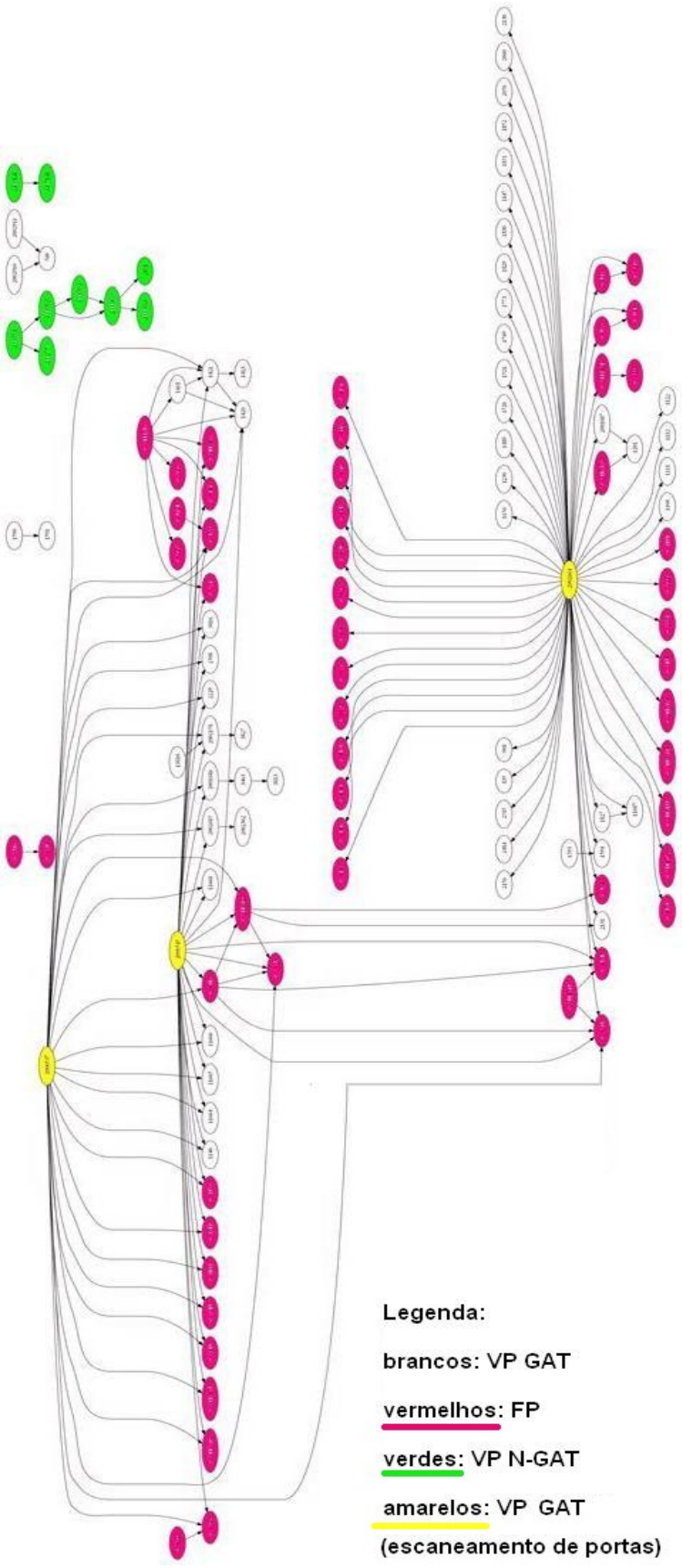

Fig. 5 – Grafo final FP3

TABELA 8
RESULTADOS DO EXPERIMENTO

Siglas	Descrição	FP1			FP2			FP3			TOTAL
		GAT	N-GAT	Total	GAT	N-GAT	Total	GAT	N-GAT	Total	
Q_AD	Quantidade alertas detectados	2554	1588	4142							
C_bruto	Correlações brutas	215	26	241	190	12	202	190	12	202	
C_bruto_repetido	Correlações brutas repetidas			18			6			39	
Q_C_bruto	Quantidade de tipos de alertas (grupo de alertas sem isolados)			137			123			123	
Q_isolados	Quantidade de alertas isolados	28	1	29	0	0	0	0	0	0	
Q_VP_isolados	Quantidade de alertas isolados VP	8	0	8	0	0	0	0	0	0	
Q_FP_isolados	Quantidade de alertas isolados FP	20	1	21	0	0	0	0	0	0	
P_FP_isolados	% alertas FP entre os isolados			72,41							72,41
Q_FP	Quantidade de alertas FP			14						55	
Q_CFP	Correlações entre os alertas marcados como FP	25	14	39				86	0	86	
Q_CFP_repetido	Correlações repetidas entre os alertas marcados como FP			12						1	
P_FP	% alertas FP			21,08						44,72	54,22
Q_CVP	Correlações entre os alertas marcados como VP	190	12	202	190	12	202	73	8	81	
Q_CVP_repetido	Correlações repetidas entre os alertas marcados como VP			6			6			3	
Q_VP	Quantidade de alertas VP			123			123			68	
Q_C_TIMESTAMP	Quantidade de tipos de alertas (grupo de *timestamp* sem isolados)			152			130			122	

REFERÊNCIAS

[1] Emanoel C. C. Silva. Gerenciamento e Integração das Bases de Dados de Sistemas de Detecção de Intrusões. Universidade Federal do Maranhão – Departamento de Engenharia de Eletricidade; São Luis, MA, Brasil, 2006, 151 f. Dissertação (Mestrado em Engenharia de Eletricidade).

[2] Cristina Abad et al. Log Correlation for Intrusion Detection A Proof of Concept. University of Illinois as Urbana-Champaign / NCSA / SAIC; 2003. p. 10. In: Proceedings of the 19th Annual Computer Security Applications Conference (ACSAC 2003).

[3] Samuel King, Peter Chen, Z. Morley Mao, Dominic G. Lucchetti Enriching Intrusion Alerts Through Multi-Most Causality. University of Michigan; Michigan, EUA, 2005. p. 13.

[4] Dario Valentino Forte. The Art of Correlation - Part 1. Computer Fraud & Security, Italia, 2004. pp. 7-11.

[5] Peng Ning; Yun Cui; Douglas Reeves S. . Analyzing Intensive Intrusion Alerts via Correlation. North Carolina State University; Raleigh, NC, EUA, 2002. p. 21.

[6] Peng Ning; Yun Cui. An Intrusion Alert Correlator Based on Prerequisites of Intrusions. North Carolina State University; Raleigh, NC, EUA, 2002. p. 16.

[7] Yan Zhay; Peng Ning; XU, JUN. Integrating IDS Alert Correlation and OS-Level Dependency Tracking. North carolina State University , 2006, S. Mehrotra et al. (Eds.): ISI 2006, LNCS 3975, pp. 272–284, 2006.

[8] Douglas Reeves S., Peng Ning; Yun Cui. Constructing Attack Scenarios Through Correlation of Intrusion Alerts. North Carolina State University; CCS'02, Washington, DC, USA., 2002. p. 10.

[9] Yan Zhay, Peng Ning, Purush Iyer, Douglas S. Reeves. Reasoning About Complementary Intrusion Evidence . North Carolina State University ; Raleigh, NC, EUA, 2004. p. 10.

[10] Fumio Mizoguchi. Anomaly Detection using Visualization and Machine Learning. Science University of Tokyo – Information Media Center; Noda, Japan, 2000. p. 6.

[11] Constantine Manikopoulos, Symeon Papavassiliou. Network Intrusion and Fault Detection: A Statistical Anomaly Approach. New Jersey Institute of Technology, NJ, EUA, 2002. p. 7.

[12] Alfonso Valdes, Keith Skinner. Probabilistic Alert Correlation. SRI International; Springer-Verlag Berlin Heidelberg , 2001, W. Lee, L. Me, and A. Wespi (Eds): RAID 2001, LNCS 2212, pp. 54-68.

[13] Tadeusz Pietraszek, Axel Tanner. Data mining and Machine Learning – Towards Reducing False Positives in Intrusion Detection. IBM Zuurich Research Laboratory, Ruschlikon, Suécia, 2005. Information Security Technical Report, Vol. 10, ed. 3, pp 169-183.

[14] Benjamin Morin, Hervé Debar. Correlation of Intrusion Symptoms: An Application of Chronicles. France Télécom R&D; Springer-Verlag - Berlin Heidelberg , 2003, G. Vigna, E. Jonsson, and C. Kruegel (Eds.): RAID 2003, LNCS 2820, pp. 94–112.

[15] Ron Gula. Correlating IDS Alerts with Vulnerability Information. Chief Technology Officer – Tenable Network Security; Columbia, MD, EUA, 2007. p. 10.

Anderson Aparecido Alves da Silva é graduado em Tecnologia de Processamento de Dados (FIEO), tem especialização Lato Sensu em Administração de Empresas com ênfase em Análise de Sistemas (FECAP) e é mestrando em Engenharia da Computação com ênfase em redes (IPT). Atualmente é gerente de desenvolvimento da empresa Donatelli.

Adilson Eduardo Guelfi é graduado em Engenharia Eletrônica, tem especialização em Gestão de Negócios e Projetos (FIA), é mestre em Informática Industrial (UFSC) e doutor em Engenharia Elétrica (LSI-EPUSP). Atualmente, é pesquisador associado ao LSI-EPUSP e professor regular do programa de mestrado do Instituto de Pesquisas Tecnológicas (IPT) de São Paulo.