Tratamento Automatizado de Incidentes de Segurança da Informação em Redes de Campus

Italo Valcy[1,2], **Luciano Porto Barreto**[1,2], **Jerônimo Bezerra**[1,2]

[1] Universidade Federal da Bahia (UFBA)
Salvador, BA – Brasil

[2]Grupo de Resposta a Incidentes de Segurança - Bahia/Brasil (CERT.Bahia)
Salvador, BA – Brasil

{italovalcy, lportoba, jab}@ufba.br

Resumo. *O crescimento atual da Internet tem alavancado o número de incidentes de segurança da informação em diversas instituições. Os prejuízos causados por tais incidentes e sua dificuldade de prevenção requerem o estabelecimento de políticas e mecanismos eficientes de tratamento e resposta a incidentes de segurança. Entretanto, a correta identificação de equipamentos comprometidos ou participantes em um incidente de segurança é severamente prejudicada pela ampla existência de redes que utilizam técnicas de tradução ou atribuição dinâmica de endereços IP (como o NAT ou DHCP), as quais dificultam a identificação precisa dos equipamentos internos. Este trabalho descreve o projeto, a implementação e avaliação da ferramenta TRAIRA, a qual automatiza o procedimento de detecção, identificação e isolamento dos equipamentos geradores de incidentes de segurança em redes com estas características. A ferramenta está atualmente em produção e uso efetivo em uma rede de campus com cerca de 12.000 equipamentos conectados.*

1. Introdução

A popularização da Internet tem proporcionado relevante democratização do acesso às informações em rede, porém, paralelo a esse fenômeno, é notório o aumento substancial na ocorrência de incidentes relacionados à segurança da informação. Tais incidentes são diversos e variam sobremaneira em gravidade, impacto e escala. Exemplos abrangem desde a infecção e disseminação de software malicioso, apropriação de senhas e informações privadas até fraudes bancárias, violação de sigilo e propriedade intelectual, ataque a sites comerciais e ciberterrorismo. No âmbito das empresas e instituições governamentais, torna-se fundamental atuação efetiva da alta administração visando à prevenção, detecção e tratamento adequado de tais eventos adversos com o intuito de proteger os ativos organizacionais (patrimônio, imagem, pessoas).

Instituições lidam com a questão geralmente através da atuação em dois eixos. O primeiro estabelece uma Política de Segurança da Informação (PSI) que normatiza regras, condutas e planos de ação, bem como possíveis sanções aos usuários em caso do descumprimento das regras estatuídas. O segundo eixo perpassa a constituição de um grupo especializado de resposta a incidentes de segurança (CSIRT, do inglês, *Computer Security Information Response Team*) responsável pelas questões operacionais desde a fase de identificação até a resolução dos incidentes de segurança. Seguindo essa linha de

atuação em segurança da informação, nossa instituição contempla a administração de uma rede de campus que interliga diversas instituições acadêmicas e de pesquisa perfazendo aproximadamente 12.000 equipamentos (desconsiderando equipamentos conectados em redes sem fio). Ademais, no período compreendido entre janeiro e julho de 2011, foram reportados aproximadamente 2.000 incidentes de segurança da informação referentes às instituições de pesquisa e ensino monitoradas pelo nosso CSIRT [CERT.Bahia 2010], o que atesta a necessidade de tratamento efetivo e eficaz de tais incidentes.

Os prejuízos causados por tais incidentes aliados à sua dificuldade de prevenção [Scarfone et al. 2008] demandam o estabelecimento de políticas e mecanismos eficientes de tratamento e resposta. A grande maioria desses incidentes são reportados por CSIRTs externos à instituição, a exemplo do CERT.br e do CAIS/RNP. Lentidão, leniência no tratamento do incidente, bem como a reincidência podem ensejar sanções severas e indesejáveis, tais como o bloqueio de acesso e a recusa de e-mails originários da instituição comprometida. A infecção e disseminação de *malware*, a exemplo do vírus Conficker, ou a participação (ainda que involuntária) de máquinas em *botnets* (redes de ataques em larga escala) são exemplos dos tipos de incidentes mais frequentes na geração de notificações externas. Portanto, os prejuízos institucionais decorrentes de tais sanções são consideráveis em nosso contexto (instituições acadêmicas e de pesquisa), o que requer atuação rápida e efetiva do time de resposta a incidentes.

Entretanto, um dos principais entraves à correta identificação de equipamentos comprometidos ou participantes em um incidente de segurança consiste na ampla existência de redes configuradas com faixas de endereços IP "falsos" (i.e., não-roteáveis na Internet [Rekhter et al. 1996]) e NAT (do inglês, *Network Address Translation*) [Egevang and Francis 1994]. Estas redes geralmente utilizam o serviço de DHCP (do inglês, *Dynamic Host Configuration Protocol*) [Droms 1997], o qual pode atribuir temporariamente e aleatoriamente endereços às máquinas da rede. Essa configuração, adotada em grande parte das instituições, oculta a identidade precisa das máquinas internas comprometidas, o que dificulta sobremaneira o tratamento adequado de incidentes.

Outros fatores complicadores, nesse contexto, incluem o elevado volume de notificações recebidas e a heterogeneidade dos elementos da rede. O uso de roteadores e *switches* de fabricantes diversos (caso geral nas instituições) compromete ou limita a utilização de soluções proprietárias. Ainda que o parque organizacional de equipamentos seja o mais homogêneo possível, as soluções proprietárias existentes são significativamente onerosas, o que pode inviabilizar sua adoção. Por fim, mesmo instituições de médio e grande porte carecem de equipe e ferramentas de segurança especializadas para tratar os principais tipos de incidentes.

De fato, a automatização adequada do processo de tratamento de incidentes pode reduzir substancialmente os custos financeiros do tratamento de incidentes (especialmente o de pessoal alocado para esta tarefa) além de resolução mais célere dos problemas. No cenário ideal, concretamente, uma ferramenta pode automaticamente detectar e isolar as máquinas comprometidas para, em seguida, acionar a equipe de apoio (*helpdesk*) a fim de proceder a desinfecção das máquinas. Assim, os analistas de segurança, cujo custo de contratação é comumente alto, podem se ater ao tratamento de incidentes mais importantes ou complexos.

Diante deste cenário de problemas reais, este trabalho descreve o desenvolvimento e avaliação de uma ferramenta, chamada TRAIRA (*Tratamento de Incidentes de Rede Automatizado*), que automatiza as principais etapas do processo de tratamento de incidentes de segurança (*detecção* e *contenção* da máquina interna causadora do incidente). A ferramenta desenvolvida foi avaliada em um ambiente real, na rede de campus da Universidade Federal da Bahia (UFBA), onde é utilizada como base do processo de tratamento de incidentes de segurança gerados pela instituição há aproximadamente um ano, ajudando a equipe de segurança a tratar e responder uma média de 30 notificações de incidentes por semana. O baixo custo de implantação e execução da ferramenta indica a viabilidade de sua aplicação prática em outros ambientes corporativos.

O restante deste artigo está estruturado da seguinte maneira. A seção 2 destaca os principais desafios acerca do processo de tratamento de incidentes de segurança; fatores motivadores para desenvolvimento da ferramenta . Em seguida, descrevemos a arquitetura e funcionamento da ferramenta TRAIRA (seção 3), bem como os resultados obtidos mediante sua utilização em ambientes reais (seção 4). Por fim, discutimos trabalhos correlatos quanto à automatização do processo de tratamento de incidentes de segurança na seção 5 e apresentamos as considerações finais e trabalhos futuros na seção 6.

2. Fases e desafios no tratamento de incidentes de segurança

O guia para tratamento de incidentes de segurança do NIST (do inglês, *National Institute of Standards and Technology*) [Scarfone et al. 2008] decompõe o processo de resposta a incidentes em quatro fases: *Preparação*, *Detecção e Análise*, *Contenção, Mitigação e Recuperação* e *Ações Pós-Incidente*. A fase de *Preparação* envolve o estabelecimento e treinamento de um grupo de resposta a incidentes, aquisição de ferramentas e recursos necessários, armazenamento dos registros de atividades dos sistemas para futuras auditorias etc. A fase de *Detecção e Análise* visa detectar ou identificar a existência de um incidente. A fase *Contenção, Mitigação e Recuperação*, por sua vez, objetiva evitar que os efeitos do incidente se propaguem ou afetem outros recursos da rede, e restaurar o funcionamento normal dos serviços afetados. Por fim, a etapa *Ações Pós-Incidente* consiste em avaliar o processo de tratamento de incidentes e verificar a eficácia das soluções adotadas.

Cada uma destas fases requer ações específicas de mitigação ou controle. Por exemplo, na fase de detecção e análise, deve-se registrar os recursos afetados (no caso de incidentes contra a organização) ou a origem do incidente (no caso de incidentes originados na organização). Na fase de contenção e mitigação deve-se isolar os sistemas diretamente relacionados ao incidente e efetuar o tratamento do recurso em questão (desinfecção de uma máquina contaminada com *vírus/worm*, remoção de um artefato malicioso, recuperação de uma página web modificada, etc). No entanto, alguns serviços comumente utilizados na configuração de redes, a exemplo do NAT e DHCP, podem dificultar consideravelmente a consecução dessas ações corretivas.

A técnica de NAT visa traduzir os endereços IP utilizados na rede interna em um endereço IP (ou faixa de endereços) utilizado na rede externa (Internet). Os endereços IP internos são, portanto, desconhecidos dos equipamentos externos, tal qual o número de um ramal de um telefone é escondido quando efetuada uma ligação via PABX. Assim, a rede externa desconhece o verdadeiro emissor do pacote. No que tange ao tratamento de incidentes de segurança, a principal dificuldade adicionada pelo NAT consiste em deter-

minar com precisão o endereço IP interno que foi traduzido no endereço IP externo, uma vez que as notificações de incidentes recebidas de fontes externas (*e.g.* outros CSIRTs) contêm apenas o endereço IP externo.

Outro agravante reside no uso disseminado do *Protocolo de Configuração Dinâmica de Hosts* (DHCP) [Droms 1997], que permite a um *host* obter endereço(s) IP, e outros parâmetros de configuração da rede, automaticamente. Em uma rede com DHCP, é possível que um mesmo dispositivo possua diferentes endereços IP ao longo do dia, da semana ou do mês, a depender da política de tempo de concessão (*lease time*) utilizada. Por isso, limitar a identificação da máquina a um endereço IP pode ser ineficaz ou produzir resultados errôneos (o que seria bastante prejudicial em caso de bloqueio automatizado da máquina comprometida). Portanto, atualmente considera-se mais adequada a utilização do endereço MAC (*Media Access Control*) como identificador único do *host*.

Um terceiro desafio para o tratamento de incidentes é a falta de gerenciamento dos registros de atividades (*logs*) de dispositivos. Esses registros são de grande valia quando da ocorrência de um incidente, pois auxiliam a auditoria dos sistemas afetados. Não obstante, a quantidade, volume e variedade dos *logs* de segurança dos sistemas têm crescido bastante, comprometendo e, por vezes, até inviabilizando, a investigação de incidentes de segurança gerados por uma instituição. Essa investigação consiste geralmente em efetuar buscas nos *logs* do dispositivo de NAT por uma ocorrência do IP e porta listados na notificação e cuja data e hora estejam em concordância com os dados. Vale salientar que, considerando os entraves supracitados, o processo de tratamento de incidentes de segurança, em muitos casos, tende a ser interrompido nessa etapa. Portanto, a automatização adequada dessa etapa é de fundamental importância para o tratamento efetivo de incidentes de segurança em tempo hábil.

3. TRAIRA: uma ferramenta para Tratamento Automatizado de Incidentes de Rede

O TRAIRA é um programa que atua em todas as fases do tratamento de incidentes (a saber, *preparação*, *detecção e análise*, *contenção, mitigação e recuperação* e *ações pós-incidente* [Scarfone et al. 2008]) de forma que a detecção e contenção da máquina interna causadora do incidente são totalmente automatizadas. Na fase de preparação destacam-se dois recursos requeridos pelo TRAIRA: i) a configuração do serviço de *logging* remoto do equipamento de NAT e ii) a utilização de um sistema de registro sobre a atribuição de IPs, associando-os aos endereços físicos dos dispositivos de rede: os endereços MAC.

Já na fase de detecção e análise, o TRAIRA utiliza os recursos configurados anteriormente para automaticamente extrair as informações relevantes de uma notificação; buscar por evidências nos *logs* do dispositivo de NAT que informem o(s) IP(s) interno(s) responsável(veis) pela notificação recebida; associar o endereço IP interno a um endereço MAC da máquina, de forma que sua identificação seja única; gerar relatórios e estatísticas sobre os incidentes recebidos; e responder à organização que reportou o incidente. A fase de contenção implementa políticas de cessação da atividade maliciosa através do isolamento da máquina comprometida como, por exemplo, bloqueio no *switch* gerenciável ou roteador mais próximo. Ao final do tratamento de uma notificação, o TRAIRA gera uma resposta automática à organização que reportou o incidente e também um relatório detalhado para a equipe de apoio a fim de que as medidas cabíveis sejam aplicadas. No

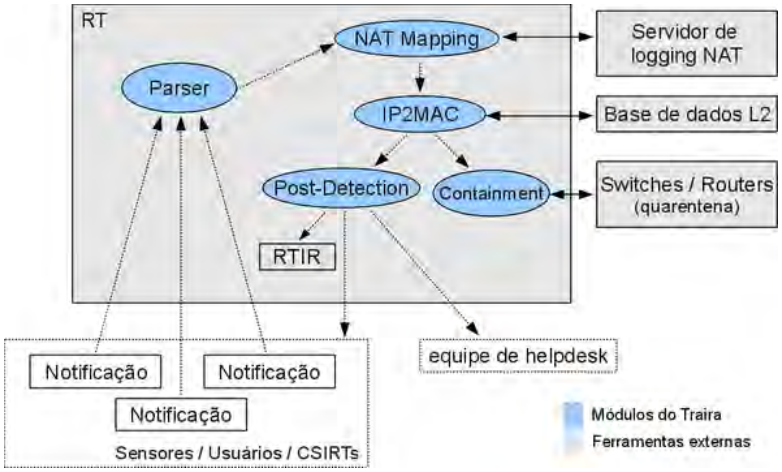

Figura 1. Visão geral da arquitetura do TRAIRA

âmbito desse artigo, serão enfatizadas as fases de preparação e detecção e analise e alguns aspectos relacionados à contenção.

No nosso contexto e em diversas outras instituições, há utilização disseminada do *Request Tracker* (RT) [BestPractical 2011] e como sistema de *helpdesk* e tratamento inicial de incidentes. A fim de preservar o conhecimento e a utilização dessas ferramentas, o TRAIRA foi idealizado como um aprimoramento do RT através de extensões específicas para o tratamento automatizado de incidentes em redes. Tal decisão de projeto reduziu o tempo e custo de desenvolvimento, permitindo a reutilização de diversas funcionalidades existentes nessas ferramentas (autenticação, *backup*, atualização) e da própria base de incidentes de segurança pré-existente. Além disso, isso facilita a adoção do TRAIRA por outras instituições interessadas em incrementar a automatização dos procedimentos de tratamento de incidentes.

Na subseção seguinte, a arquitetura e funcionamento do TRAIRA será apresentada em maiores detalhes.

3.1. Arquitetura do TRAIRA

A concepção do TRAIRA foi estruturada em uma arquitetura modular, apresentada na Figura 1. Nessa figura, os componentes em formato de elipse representam os módulos que foram desenvolvidos como parte do TRAIRA e os componentes em formato de retângulo representam programas ou recursos externos necessários ao funcionamento do TRAIRA. Os cinco módulos do TRAIRA são: *Parser*, *NAT Mapping*, *IP2MAC*, *Post-Detection* e *Containment*. A seguir, apresentamos uma breve descrição das funcionalidades desses módulos.

- *Parser:* Responsável pelo recebimento da notificação e pela extração das informações essenciais ao tratamento do incidente: endereço IP e porta de origem, data e horário.
- *NAT Mapping:* Utiliza as informações extraídas pelo *parser* e realiza uma busca nos *logs* do dispositivo de NAT para associar a tupla `<data, hora, IP, porta>` a um endereço IP interno, responsável de fato pelo incidente.

- *IP2MAC:* aqui é feita a associação do IP interno ao endereço MAC da máquina. Esse passo é importante em instituições que utilizam o DHCP, pois um IP pode ter sido usado por mais de uma máquina ao longo do tempo.
- *Post-Detection:* Responsável pela extração de dados da notificação e do tratamento realizado a fim de gerar estatísticas sobre os incidentes, gerar relatórios à equipe de *helpdesk* (para, por exemplo, efetuar o isolamento e desinfecção da máquina) e responder à instituição que reportou o incidente.
- *Containment:* Neste módulo é feito o isolamento do *host* que causou o incidente para evitar que ele continue com a atividade maliciosa na rede ou afete outros recursos.

O TRAIRA foi desenvolvido como uma extensão do RT, permitindo que o tratamento dos incidentes de segurança seja feito tanto pela interface web, onde o usuário fornece manualmente a notificação do incidente, quanto via e-mail quando a organização que reporta o incidente envia uma mensagem para um endereço de e-mail especialmente designado para este fim. Foi utilizada a linguagem de programação *Perl*, com a qual o próprio RT foi implementado. Em sua primeira versão, possui aproximadamente 2.500 linhas de código entre interfaces de usuário, núcleo da aplicação, módulos de interface com recursos externos (*logs*, tabela de endereços MAC, etc) e demais componentes. O TRAIRA é distribuído sob a licença GPLv2 ou superior[1] e encontra-se disponível para download em [TRAIRA 2011].

Tratamento de Incidentes no TRAIRA

O tratamento de incidentes automatizados pelo TRAIRA segue um fluxo de trabalho composto pelas etapas a seguir.

1. Uma entidade (interna ou externa) submete uma notificação ao TRAIRA reportando um incidente de segurança. Essa notificação deve conter evidências suficientes para comprovar a atividade maliciosa e incluir, no mínimo, o endereço IP, porta de origem, data, hora e *timezone* da ocorrência. A entidade que reporta incidentes pode ser materializada nos CSIRTs (e.g., CAIS, CERT.br), em sensores de monitoramento de atividades maliciosas (IDSs, *honeypots* etc) ou até mesmo em usuários que submetem incidentes através da interface web;
2. O TRAIRA verifica se existe um *parser* específico para o tipo de notificação recebido. Em caso afirmativo, o *parser* extrai os dados importantes da notificação e o tratamento avança para a detecção da máquina interna. Caso inexista um *parser* apropriado, a notificação permanece em aberto no RT aguardando pelo tratamento manual da equipe de segurança;
3. A partir dos dados extraídos da notificação (tupla `<data, hora, IP, porta>`) é feita uma busca nos *logs* do dispositivo de NAT para determinar o respectivo endereço IP da máquina da rede interna;
4. De posse do endereço IP da máquina causadora do incidente, é realizada uma busca para descobrir o respectivo endereço MAC;

[1] GPL é uma sigla usada para *GNU Public License*, uma licença de software livre especificada pela *Free Software Foundation*.

5. Caso o módulo de Contenção esteja habilitado para executar automaticamente, a máquina em questão (representado pelo MAC obtido na etapa anterior) é bloqueado ou movido para uma Rede Virtual (VLAN) de quarentena;
6. De posse do endereço MAC e do resultado do módulo de Contenção, o TRAIRA notifica a equipe de *helpdesk* para tomada das medidas cabíveis;
7. Uma resposta automática (e-mail) é enviada à instituição produtora da notificação para informar a identificação da máquina causadora do incidente e o início dos procedimentos de tratamento e recuperação.

Diante do exposto, o TRAIRA automatiza completamente o processo inicial de tratamento de incidentes de segurança. Cabe ainda ao administrador executar as providências necessárias para resolução do incidente, a exemplo da desinfecção de máquinas contaminadas com *vírus/worm* ou aplicar as medidas administrativas cabíveis à uma violação de *copyright*. Vale salientar, entretanto, que, em virtude do considerável volume de notificações recebidos pelas instituições e a carência de pessoal especializado e em número suficiente para responder às notificações, a etapa de detecção tende, muitas vezes, a nem ser iniciada. As etapas descritas acima são executadas de forma *on-line*. Portanto, assim que um incidente é reportado ao TRAIRA, seu tratamento tem início imediato. Assim, o TRAIRA proporciona uma importante contribuição para o processo de tratamento e resposta aos incidentes de segurança de uma instituição.

As subseções seguintes visam detalhar duas etapas importantes do tratamento do incidente pelo TRAIRA: detecção e isolamento do *host* responsável pelo incidente.

3.2. Detecção do *host* responsável pelo incidente

Apesar de suas desvantagens [Egevang and Francis 1994, Seção 4], o NAT é uma técnica bastante utilizada pelas instituições de ensino e pesquisa conectadas à Internet, principalmente pela possibilidade de economia de endereços IPv4 e ocultação da topologia da rede interna. Por outro lado, sua utilização traz implicações no tratamento de incidentes de segurança, uma vez que o endereço listado na notificação não representa diretamente a máquina da rede interna que realmente causou o incidente. Nesse caso, faz-se necessário um mapeamento entre o IP e porta listados na notificação e o IP interno que causou o incidente. Para realizar esse mapeamento, o módulo *NATMapping* utiliza as informações extraídas pelo *Parser* e as correlaciona aos *logs* do(s) dispositivo(s) de NAT, retornando o(s) IP(s) internos responsáveis pelo incidente.

O processo de correlacionar essas duas entradas, no entanto, não é uma tarefa trivial. É necessário considerar a grande diversidade de dispositivos de NAT disponíveis (cada um deles pode produzir *logs* de forma específica), o grande volume de dados a serem processados e, ainda pior, a correspondência entre a data/horário que é listado na notificação e aqueles listados nos *logs*. Para lidar com este último problema, a ferramenta incorpora a definição de um valor, configurável, de tolerância temporal entre os horários da notificação e dos *logs*. O valor mais adequado para uma instituição depende das características da rede. Um fator obrigatório a ser observado nessa definição é a relação de máquinas da rede interna por IP de NAT: quanto mais máquinas são associadas a um único NAT, maior será a taxa de reaproveitamento de portas e, consequentemente, menor poderá ser a tolerância à diferença nos relógios.

Para tratar da diversidade na utilização de dispositivos de NAT nas instituições,

e, até mesmo internamente à uma instituição (com diferentes dispositivos de NAT por segmento de rede), o módulo *NATMapping* foi desenvolvido de forma que seja possível definir um dispositivo de NAT para cada segmento de rede (levando-se em consideração a sobreposição entre segmentos) e um dispositivo padrão a ser usado caso não haja uma definição específica para determinado segmento de rede. Por exemplo, o administrador pode definir que a rede `200.128.99.0/24` utiliza o *ASA/Cisco*, já a rede `200.128.196.0/23` utiliza *IPTables/Netfilter* com exceção da sub-rede `200.128.197.0/28` que também utiliza *ASA/Cisco* e, finalmente, a rede `200.128.199.0/24` não utiliza NAT. Note que o mapeamento acima é sobre uma visão externa ou, mais especificamente, considerando os dados da notificação. Essa flexibilidade de configuração permite, por exemplo, definir as redes privadas [Rekhter et al. 1996] como "sem NAT", o que viabiliza também o tratamento de incidentes internos (detectados a partir de sensores posicionados na rede interna).

3.3. Isolamento do *host* responsável pelo incidente

A etapa de isolamento efetua a contenção do *host* detectado anteriormente para evitar que ele continue com a atividade maliciosa na rede ou comprometa outros recursos. Esta contenção pode acontecer por diversas vias: desligamento da máquina, remoção do cabo de rede, bloqueio no dispositivo de rede gerenciável mais próximo etc. Essa última alternativa é mais interessante do ponto de vista de automatização. Em contrapartida, o inconveniente é que o usuário desconhece a razão pela qual sua estação está sem comunicação na rede. Nesse sentido, uma técnica mais apropriada, proposta em [Kaiser et al. 2006], consiste em direcionar o tráfego de rede do *host* comprometido para uma VLAN de quarentena. Nesta VLAN, as requisições do usuário seriam encaminhadas a uma página web da instituição que informa sobre o bloqueio preventivo da máquina e ações que este deve tomar (e.g., contactar imediatamente o *helpdesk*).

Tal abordagem de contenção tem sido reiteradamente considerada na literatura, principalmente, através do uso de ferramentas como *firewall* e IDS. Não obstante, essa abordagem mostra-se ineficiente do ponto de vista de propagação da atividade maliciosa na rede local do *host* detectado, pois, para os segmentos diretamente conectados, os pacotes não trafegam pelo *firewall* ou IDS. De fato, o bloqueio mais eficaz pode ser realizado na camada 2 (*Layer 2*), por exemplo, nos *switches* gerenciáveis ao qual o *host* comprometido está conectado. Devido à possível variação no ambiente de rede das instituições, o TRAIRA considera três estratégias possíveis para etapa de isolamento do processo de tratamento de um incidente: i) bloqueio do *host* no equipamento de *firewall*, ii) bloqueio no *switch* gerenciável mais próximo ao *host* detectado, iii) direcionamento do *host* para uma VLAN de quarentena.

Cada uma dessas estratégias possui vantagens, desvantagens e requisitos de implantação específicos. Portanto, a decisão final acerca da política mais apropriada depende das características de cada instituição. A opção mais simples consiste no bloqueio do *host* no equipamento de *firewall*. Essa estratégia mostra-se eficaz para contenção de ataques cujo destino seja outra instituição ou esteja em outro segmento de rede ao qual *host* detectado pertence. Também possibilita a criação de regras para redirecionar de forma transparente o tráfego oriundo do *host* comprometido para uma página web, a qual informa ao usuário o bloqueio de sua máquina e explica a razão para tal ação. Contudo, não atende ao tratamento da propagação de atividade maliciosa na rede local. O requisito

de implantação consiste basicamente no suporte à criação de filtros de pacotes e regras de redirecionamento baseadas em endereço MAC no *firewall* da instituição (característica comumente encontrada nos *firewalls* mais usados).

Outra variante mais completa consiste em direcionar o tráfego do *host* comprometido para uma VLAN de quarentena no nível da camada de enlace, o que evita o problema supracitado de atividade maliciosa na rede local e simplifica sobremaneira o procedimento de contenção. Entretanto, esta opção tem requisitos de implantação mais complexos. É necessário que a máquina esteja conectada em algum *switch* que possibilite a criação de filtros (ACLs) para modificar a VLAN baseado no endereço MAC de origem dos pacotes. Tal funcionalidade é também conhecida como *MAC-based VLAN*. Todavia, em pesquisas realizadas, constatamos que apenas um fabricante de equipamentos de rede implementa essa funcionalidade. Considerando a efetividade dessa solução, decidimos reorientar nossos procedimentos de compra para a aquisição de novos equipamentos com esta funcionalidade.

4. Avaliação experimental e resultados obtidos

Desde a implantação do TRAIRA na rede da UFBA em setembro de 2010, todas as notificações de incidentes de segurança recebidas pela equipe (sejam aquelas enviadas por ferramentas de monitoramento interno, tais como *honeypots*, IDSs, ou as enviadas pelos grupos de segurança, tais como CAIS e CERT.br) tem sido tratados automaticamente. Por exemplo, as notificações de incidente relacionadas ao projeto Honeypot.BR do CERT.br [Honeynet.BR 2011], do qual a UFBA participa, correspondem a uma média diária de 5 a 10 notificações, e cada notificação contém cerca de 20 incidentes.

Um recurso fundamental aos grupos de resposta a incidentes de segurança (CSIRTs) consiste na produção de estatísticas relevantes ao contexto de tratamento de incidentes e tomada de decisão para a alta administração [Arvidsson et al. 2001]. A obtenção de tais dados auxiliam os CSIRTs a detectar tendências, prever futuros ataques em grande escala e direcionar atividades, dentre outros. A implementação atual do TRAIRA fornece estatísticas importantes geradas automaticamente a partir de informações retiradas da notificação recebida e do tratamento efetuado. A seguir, discutiremos os resultados obtidos a partir dessas estatísticas.

Situação e taxa de tratamento dos incidentes. O gráfico quantitativo (Figura 2) pode ser utilizado para aferir a efetividade do tratamento de incidentes de segurança na instituição. Neste âmbito, são listados os incidentes reportados *versus* os que foram resolvidos. No caso ideal, espera-se que a linha de incidentes resolvidos esteja o mais próximo possível da linha dos incidentes reportados.

Tendo em vista que a rede da UFBA conta mais de 12.000 equipamentos, o tratamento de todas essas notificações era extremamente custoso, do ponto de vista de alocação de pessoal qualificado. Conforme pode ser visto na Figura 2 (extraída do TRAIRA), desde o início de 2011, nossa instituição recebeu mais de 550 notificações, sendo que a grande maioria delas foi tratada automaticamente pelo TRAIRA. Nesta figura, uma linha representa os incidentes reportados, enquanto a outra indica quais destes incidentes foram tratados automaticamente pelo TRAIRA. Portanto, a proximidade das linhas indica a eficácia do uso da ferramenta no tratamento de incidentes de segurança.

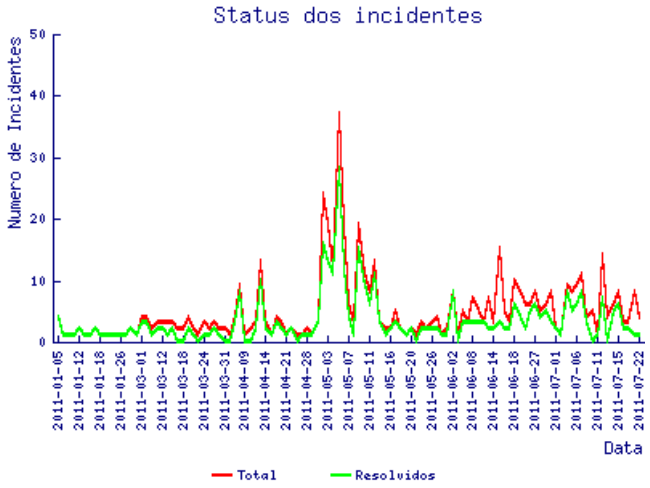

Figura 2. Situação do tratamento de incidentes reportados a UFBA entre janeiro a julho de 2011

Segmentação de incidentes por Rede Virtual (VLAN). Nossa rede institucional é estruturada em VLANs a fim de estabelecer políticas de controle de acesso e segmentação de tráfego. Atualmente, diversas instituições têm adotado esse modelo como facilitador da administração de grupos de usuários e recursos em redes de campus e de larga escala. A Figura 3 apresenta tal gráfico para a UFBA no período de janeiro a julho de 2011. Esse gráfico ressalta as VLANs (cujos nomes reais foram sombreados por questões de segurança e privacidade) que mais impactam na geração de incidentes de segurança, o que permite direcionar medidas de prevenção específicas. Tais dados são fundamentais para redes de campus ou extensas, visto que há forte tendência nas instituições em administrar redes por meio de uma estruturada combinada de VLANs.

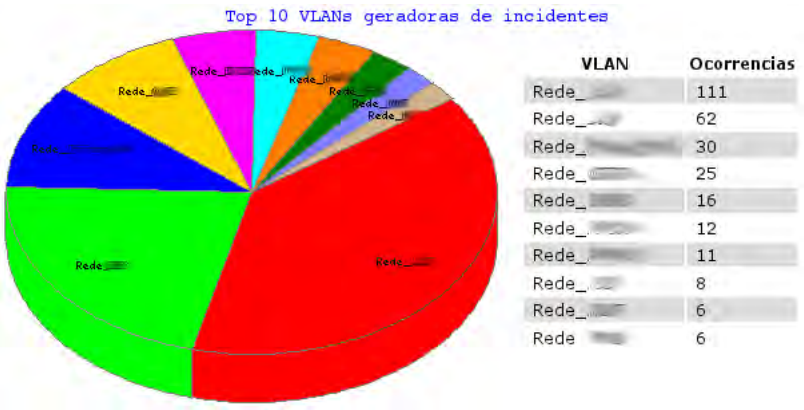

Figura 3. As dez principais VLANs geradoras de incidentes na rede UFBA (janeiro a julho de 2011)

Em nosso ambiente institucional, a análise das estatísticas produzidas pelo TRAIRA permitiram a identificação precisa das principais sub-redes geradoras de incidentes. Com base nesse resultado, pôde-se iniciar um trabalho específico e direcionado aos usuários, dirigentes e administradores destas sub-redes visando identificar o motivo da atividade maliciosa e implantar estratégias de controle e mitigação.

Identificação de máquinas reincidentes. Esta métrica indica a taxa de reincidência na geração de incidentes em determinado *host*. Pode ser usada como indicador qualitativo do tratamento e pós-tratamento de incidentes. A interpretação desse dado pode levantar diversas hipóteses, tais como: a fase de isolamento e desinfecção está sendo ineficaz; no caso dos incidentes de *vírus/worm* pode indicar inexperiência ou desleixo do usuário no uso do recurso, propiciando novas infecções com facilidade; dentre outros.

A automatização proporcionada pelo TRAIRA simplifica o procedimento de tratamento dos principais incidentes, pois a equipe de *helpdesk* apenas recebe o endereço MAC dos dispositivos suspeitos identificados pelo sistema e realiza o tratamento das máquinas. A resposta às notificações que envolvem contenção automática é praticamente instantânea, quando comparada à abordagem manual em que cada incidente era resolvido em cerca de 30 minutos. Tal demora ensejava, por vezes, o não atendimento de algumas notificações por restrições de tempo da equipe. A economia de tempo na identificação e contenção da máquina comprometida representa um ganho qualitativo fundamental frente às instituições externas que reportam incidentes, bem como em celeridade e precisão em relação ao cessamento da propagação de atividades maliciosas na rede interna.

5. Trabalhos correlatos

A literatura apresenta uma série de trabalhos que versam sobre a definição de políticas de segurança e tratamento dos incidentes [Ceron et al. 2009, Scarfone et al. 2008, Werlinger et al. 2007, Lundell 2009], porém, no melhor de nosso conhecimento, poucos deles têm se preocupado com a automatização do procedimento a fim de minorar custos e reduzir o tempo de tratamento dos incidentes.

De maneira geral, a maioria dos CSIRTs usa sistemas de *helpdesk* customizados (também conhecidos como sistemas de chamados) para tratar seus incidentes, a fim de melhor atender às demandas do processo de tratamento de incidentes [Kaiser et al. 2006]. Dois sistemas bem conhecidos são o *Request Tracker* (RT) [BestPractical 2011] e o *Open Source Ticket Request System* (OTRS) [OTRS 2011]. Existe ainda uma extensão para o RT chamada *Request Tracker for Incident Response* (RTIR), que se concentra na resposta aos incidentes de segurança (classificação de incidentes, geração de estatísticas, etc.). Até nosso conhecimento, nenhuma dessas ferramentas, no entanto, atua especificamente na automatização do processo de tratamento e resposta a incidentes. Outros frameworks e ferramentas específicos incluem o SIRIOS [KLINGMÜLLER 2005], que apresenta algumas funcionalidades interessantes, como a de gerenciamento de contatos de segurança de uma instituição e a possibilidade de troca de informações de segurança com outros CSIRTs. O SANS Institute desenvolveu o Orion [Jarocki 2010], uma distribuição em Live-CD baseada no BackTrack para o tratamento de incidentes de segurança. Apesar de prover boas ferramentas para tratamento, o Orion ainda lida precariamente com a contenção de incidentes em redes.

Em [Kaiser et al. 2006] os autores propõem um gerenciamento semi-automatizado dos incidentes de segurança, onde os incidentes menos importantes são tratados pelo próprio usuário envolvido, ao passo que os incidentes mais sérios são encaminhados para uma equipe de segurança qualificada. Para possibilitar ao usuário não-especializado tratar um incidente, a instituição deve prover documentação suficiente e compreensível sobre as questões técnicas e organizacionais relacionadas. Os autores

propõem um sistema de gerenciamento de incidentes, o PRISM (*Portal for Reporting Incidents and Solution Management*), que consiste em três componentes. O primeiro recebe as notificações no formato IDMEF[2]. O segundo componente contém a lógica para tratamento de incidentes e medidas corretivas relacionadas. Por fim, o terceiro componente é responsável pela geração de páginas web dinâmicas que informam aos usuários as soluções indicadas para o incidente reportado. Entretanto, essa solução não contempla o tratamento de notificações externas, as quais requerem, por exemplo, a resolução de mapeamento entre o NAT efetuado e as máquinas internas.

Farnham [Farnham 2009] apresenta uma solução proprietária da Cisco, chamada *Cisco Security Agent* (CSA), e seu impacto nas várias fases do tratamento de incidentes. O CSA é um sistema de prevenção de intrusão baseado em *hosts* (HIPS, do inglês *Host Intrusion Prevention System*) que pode ser usado tanto em servidores quanto em máquinas clientes. No CSA, cada *host* possui um agente e um centro de gerenciamento, que define as políticas de segurança do *host* e centraliza o registro de eventos (*logging*). O software é baseado em regras e examina as atividades do sistema (no agente) e o tráfego de rede a fim de diferenciar comportamentos normais daqueles indicadores de uma anomalia ou ataque. O autor analisa a adequação do CSA nas etapas de tratamento de um incidente, usando como estudo de caso o software malicioso *Conficker*[3]. As desvantagens dessa solução estão relacionados, principalmente, ao alto custo de implantação e de sua inadequação a ambientes heterogêneos de rede.

Em [Ceron et al. 2010], propõe-se uma arquitetura voltada para detecção e contenção automatizada de máquinas participantes de *botnets*. A arquitetura i) coleta arquivos binários maliciosos (e.g., através de *honeypots*), ii) extrai informações dos binários coletados (tais como tentativas de acesso a endereços IP suspeitos), iii) utiliza essas informações na geração de assinaturas e monitoramento do tráfego de rede da instituição, e iv) prevê a contenção automatizada dessas máquinas por meio, por exemplo, do bloqueio no *firewall* daqueles endereços IPs identificados. Até nosso conhecimento, o trabalho não considera a tradução automática de endereços via NAT e DHCP, enfatizando o tratamento de um tipo específico de incidente: máquinas participantes de *botnets*. Outra limitação reside no fato da contenção basear-se apenas no endereço IP do *host* detectado e ser realizada no *firewall* da instituição. Tal bloqueio, infelizmente, não previne a propagação de atividade maliciosa na rede local. Por essas razões, o TRAIRA utiliza o endereço MAC como identificador de *host* (que, apesar da possibilidade de alteração, requer conhecimentos avançados para efetuá-la) e permite a escolha da estratégia de contenção: bloqueio no equipamento gerenciável mais próximo ou direcionamento para VLAN de quarentena.

[2]Motivado pela necessidade de se definir um padrão de arquitetura e comunicação para *Sistemas de Detecção de Intrusão* (IDS, do inglês *Intrusion Detection System*), o IETF, através do grupo de trabalho IDWG (*Intrusion Detection Working Group*) especificou o protocolo IDXP (*Intrusion Detection Exchange Protocol*) [Feinstein and Matthews 2007] e um formato para troca de alertas entre IDSs, denominado IDMEF (*Intrusion Detection Message Exchange Format*) [Debar et al. 2007]. Apesar de originalmente concebidos para uso em sistemas IDS, esses padrões também são bastante utilizados para notificações de incidentes de segurança.

[3]O *Conficker*, também conhecido como *Downadup* ou *Kido*, é um *worm* que ficou bastante conhecido pelo número de sistemas que conseguiu infectar ao redor do mundo. Ele explora uma vulnerabilidade conhecida do MS Windows Server (MS08-067) e pode comprometer uma série de versões do Windows [CWG 2011].

6. Considerações finais

Este trabalho apresentou o projeto, implementação e avaliação de uma ferramenta para automatizar o processo de tratamento de incidentes de segurança em redes de campus e de larga escala. A ferramenta atua em todas etapas do tratamento de um incidente, o que permite melhor aproveitamento dos recursos humanos destinados à gestão e operacionalização da segurança da informação.

Os requisitos de hardware e software necessários à implantação e execução dessa solução são triviais e muito pouco onerosos, o que reforça a viabilidade de sua aplicação prática em ambientes complexos e heterogêneos, tais como instituições acadêmicas de ensino e pesquisa, governamentais ou corporações privadas.

Atualmente, o TRAIRA encontra-se em produção e uso efetivo na rede de campus da UFBA desde setembro de 2010, sendo usado como ferramenta primária no tratamento do ciclo de incidentes de segurança das notificações recebidas pela instituição e produzidas internamente. Em verdade, baseados em nossas demandas e na situação existentes em outras instituições parceiras, consideramos que os problemas solucionados pela ferramenta são úteis a diversas instituições. Assim, nossa intenção é compartilhar nossa experiência no desenvolvimento e uso dessa ferramenta a fim de aprimorar os processos de tratamento de incidentes de segurança em outras instituições brasileiras e, como consequência, estabelecer parcerias de pesquisa e inovação com potenciais usuários e desenvolvedores interessados.

Como trabalhos futuros, destacam-se a necessidade de otimização no armazenamento e consulta dos logs, principalmente das traduções NAT (*e.g.* através de informações resumidas em bancos de dados); adoção de padrões para notificações (*e.g.* IDMEF) no *parser*; extensão para outros mapeamentos de endereço de rede, como no caso do uso de *proxy de cache http*, onde uma requisição HTTP é intermediada pelo proxy e assim o endereço de origem visto no servidor remoto é o endereço do proxy e não do cliente original; adicionar suporte a outros dispositivos de NAT, por exemplo o PFSense/FreeBSD.

7. Agradecimentos

Este trabalho foi parcialmente financiando pela FAPESB.

Referências

Arvidsson, J., Cormack, A., Demchenko, Y., and Meijer, J. (2001). TERENA'S Incident Object Description and Exchange Format Requirements. RFC 3067 (Informational). Disponível em: http://www.ietf.org/rfc/rfc3067.txt. Último acesso em Julho de 2011.

BestPractical (2011). RT: Request Tracker. Disponível em: http://www.bestpractical.com/rt/. Último acesso em Julho de 2011.

Ceron, J., Boos Jr, A., Machado, C., Martins, F., and Rey, L. (2009). O processo de tratamento de incidentes de segurança. *IV Workshop de TI das IFES*.

Ceron, J., Granville, L., and Tarouco, L. (2010). Uma Arquitetura Baseada em Assinaturas para Mitigação de Botnets. In *X Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais (SBSEG'10)*, pages 105–118. SBC.

CERT.Bahia (2010). Estatísticas do CERT.Bahia. Disponível em: http://www.certbahia.pop-ba.rnp.br/Estatisticas. Último acesso em Julho de 2011.

CWG (2011). Conficker Working Group. Disponível em: http://www.confickerworkinggroup.org/wiki/. Último acesso em Julho de 2011.

Debar, H., Curry, D., and Feinstein, B. (2007). The Intrusion Detection Message Exchange Format (IDMEF). RFC 4765 (Experimental). Disponível em: http://www.ietf.org/rfc/rfc4765.txt. Último acesso em Julho de 2011.

Droms, R. (1997). Dynamic Host Configuration Protocol. RFC 2131 (Draft Standard). Updated by RFCs 3396, 4361, 5494.

Egevang, K. and Francis, P. (1994). The IP Network Address Translator (NAT). RFC 1631 (Informational). Obsoleted by RFC 3022.

Farnham, G. (2009). Cisco Security Agent and Incident Handling. *SANS Institute InfoSec Reading Room*.

Feinstein, B. and Matthews, G. (2007). The Intrusion Detection Exchange Protocol (IDXP). RFC 4767 (Experimental). Disponível em: http://www.ietf.org/rfc/rfc4767.txt. Último acesso em Julho de 2011.

Honeynet.BR (2011). Brazilian Honeypots Alliance. Disponível em: http://www.honeypots-alliance.org.br/. Último acesso Julho de 2011.

Jarocki, J. (2010). Orion Incident Response Live CD.

Kaiser, J., Vitzthum, A., Holleczek, P., and Dressler, F. (2006). Automated resolving of security incidents as a key mechanism to fight massive infections of malicious software. In *GI SIDAR International Conference on IT-Incident Management and IT-Forensics (IMF 2006), volume LNI P-97*, pages 92–103.

KLINGMÜLLER, T. (2005). SIRIOS: A Framework for CERTs. *FIRST Conference on Computer Security Incident Handling*.

Lundell, M. (2009). Incident Handling as a Service. *SANS Institute InfoSec Reading Room*.

OTRS (2011). Open source trouble ticket system. Disponível em: http://www.otrs.org/. Último acesso em Julho de 2011.

Rekhter, Y., Moskowitz, B., Karrenberg, D., de Groot, G. J., and Lear, E. (1996). Address Allocation for Private Internets. RFC 1918 (Best Current Practice). Disponível em: http://www.ietf.org/rfc/rfc1918.txt. Último acesso em Julho de 2011.

Scarfone, K., Grance, T., and Masone, K. (2008). Computer Security Incident Handling Guide. *NIST Special Publication*, 800–61.

TRAIRA (2011). TRAIRA – Tratamento de Incidentes de Rede Automatizado. Disponível em: http://www.pop-ba.rnp.br/files/sw/rt-traira.tgz. Último acesso em Julho de 2011.

Werlinger, R., Botta, D., and Beznosov, K. (2007). Detecting, analyzing and responding to security incidents: a qualitative analysis. In *Proceedings of the 3rd symposium on Usable privacy and security*, pages 149–150. ACM.