XIX Congresso de Iniciação Científica da Unicamp
Outubro de 2011 – Campinas – São Paulo

Uma Ferramenta Interativa para Visualização de Programas Maliciosos

Bolsista: Alexandre Or Cansian Baruque* (Unicamp)
Orientador: André Ricardo Abed Grégio (CTI/MCT)

RESUMO

Uma das mais graves ameaças à segurança de computadores é o ataque por meio de programas maliciosos. Estes podem ser utilizados no roubo de informações sensíveis (cartão de crédito, contas, senhas e documentos) e na destruição de dados. A execução destes programas em ambientes controlados permite a obtenção de seu comportamento, que pode ser extensa e de difícil análise. Técnicas de visualização podem ser utilizadas para facilitar a análise e para a identificação de padrões que permitam a criação de contra-medidas. Para isto, foi criada uma ferramenta interativa que permite a análise visual (tridimensional) de comportamentos maliciosos.

Introdução. Programas maliciosos (*malware*) constituem uma grande ameaça aos usuários de sistemas computacionais. A monitoração da execução deste tipo de programa provê uma grande quantidade de informações, que devem ser analisadas de forma a produzir resultados úteis que possam auxiliar na tomada de contra-medidas. Entretanto, muitas variantes de *malware* surgem a cada dia, causando uma sobrecarga para os mecanismos de defesa e para os analistas de segurança. Para facilitar a análise das ações nocivas executadas por *malware*, é possível se aplicar técnicas de visualização, as quais permitem a observação de padrões comportamentais de ataques. Neste trabalho, foi desenvolvida uma ferramenta interativa tridimensional para ajudar na análise das atividades que um *malware* efetua durante a infecção de uma máquina-alvo.

Descrição da Ferramenta. A ferramenta proposta neste trabalho foi desenvolvida em linguagem Java, utilizando a biblioteca *j3d.* Sua finalidade é interpretar um arquivo de texto passado como entrada contendo o comportamento de um código malicioso é disponibilizá-lo visualmente, em um gráfico interativo e tridimensional, apresentado sob a forma de uma espiral. A ferramenta é composta por dois módulos: o da interface gráfica de gerenciamento e o da apresentação e manuseio da espiral gerada.

Arquivo de Entrada. O arquivo texto utilizado como entrada para a ferramenta representa o comportamento de um *malware* obtido de sua execução. Cada atividade deste comportamento localiza-se em uma linha e contém os seguintes campos, separados por ";": EXECUTOR, AÇÃO, TIPO e ALVO. Um exemplo de comportamento que pode ser utilizado na ferramenta é mostrado na Figura 1 a seguir, no qual são criados um arquivo e processo, e um arquivo é excluído.

```
C:\Malware.exe;write;file;C:\Meus Documentos\temp.txt
C:\Malware.exe;open;process;C:\Windows\System32\calc.exe
C:\Malware.exe;delete;file;C:\Meus Documentos\temp.txt
```

Fig. 1: Exemplo de comportamento de *malware*.

Módulo "GUI". O componente que permite a configuração da espiral a ser produzida é chamado de "módulo GUI", por se tratar de uma interface gráfica provida ao usuário. Nela, podem ser escolhidas as cores que representam cada tipo (FILE, REGISTRY, PROCESS, NET e MUTEX), os quais podem ser modificados. É possível também escolher as formas geométricas que para representar as ações, bem como as ações, como por exemplo, de escrita, de remoção, de criação etc. Por fim, pode-se modificar parâmetros como o caracter separador (*default: ";"*) e propriedades da espiral a ser gerada para visualização, tais como o raio e o tamanho dos pontos. Na Figura 2, é apresentado o módulo GUI e seus campos.

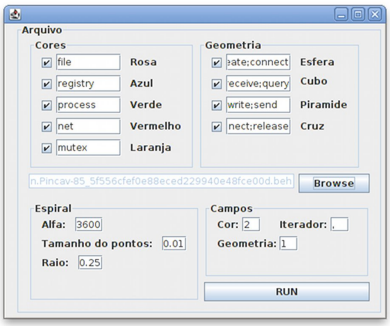

Fig. 2: Módulo GUI (interface) e suas opções de configuração.

Módulo "Visualização". Ao ser inicializado, este módulo executa as tarefas de receber e aplicar os parâmetros do módulo GUI, criar a cena geral, renderizá-la e exibir a imagem do gráfico. A definição da posição (x, y, z) na qual um ponto estará situado obedece às equações

$$x = cos\ (\alpha y)\ \text{e}\ z = sin\ (\alpha y),$$

onde "y" depende da linha correspondida pelo ponto e do valor passado de "α".

A **criação** de uma cena envolve percorrer o arquivo de comportamento e calcular seus pontos, seguido da adição dos eixos e da curva espiral. A **renderização** produz uma imagem visível para o usuário e utiliza métodos nativos da *j3d*, a classe *CanvasOverlay* e a classe *MouseBehavior*, permitindo a interação. *MouseBehavior* possibilita o reconhecimento de comandos do *mouse* (e.g., selecionar uma objeto) e *CanvasOverlay* cria uma camada de texto informativa a respeito da seleção anterior (e.g., a atividade maliciosa realizada). Isto pode ser visto na Figura 3.

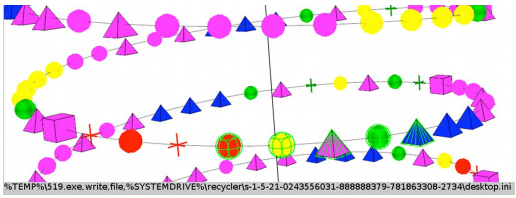

Fig. 3: Parte de uma cena criada e renderizada, seleção de objeto e informação textual provida.

Testes e Resultados. Para fins de teste, foram obtidos os comportamentos de mais de 400 exemplares de *malware*, dos quais gerou-se as espirais através da ferramenta desenvolvida. É interessante notar que exemplares identificados pelo antivírus ClamAV como da família "*Allaple*" apresentam padrões similares, mesmo quando o comportamento é incompleto (Figura 4). Em um outro caso, foi possível classificar um exemplar não identificado pela semelhança com a espiral de um cavalo-de-tróia conhecido (identificado pelo ClamAV), conforme mostrado na Figura 5.

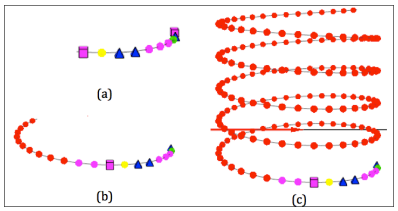

Fig. 4: Espirais de comportamento de exemplares da família de worms "Allaple".

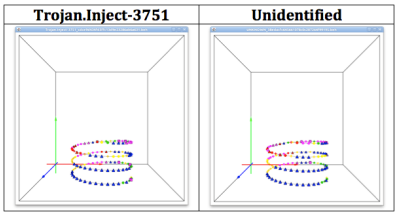

Fig. 5: Classificação de malware não identificado, por semelhança visual.

* Alexandre Or Cansian Baruque foi bolsista PIBIC/CNPq no Centro de Tecnologia da Informação Renato Archer (CTI/MCT), sob Processo nº 104693/2010-0