FACULDADE DE ENGENHARIA DE COMPUTAÇÃO

PROJETO FINAL I e II

PLANO DE TRABALHO

SFAnalytics

Lucas Carvalho Roncoroni

Edmar Roberto Santana de Rezende

15/04/2017

INTRODUÇÃO

Quase um milhão de malwares são criados todos os dias (CNN, 2015), hackers estão custando entre U$345 e U$545 bilhões anualmente para usuários e empresas (U.S. News, 2014). Por isso o desenvolvimento de ferramentas que ajudem um analista a identificar novas ameaças é de extrema importância.

A identificação de um malware, ou programa malicioso, na maioria dos casos, é feita por um antivírus e segundo SBseg, “O grande problema dos antivírus é o surgimento frequente e crescente de variantes de malware” (2011).

Para a identificação de novas variantes por programas, é necessário o uso de técnicas de aprendizagem de máquina, mas o emprego dessas técnicas no âmbito da segurança não é tão simples, existe uma grande preocupação com falsos positivos.

CARACTERIZAÇÃO DE PROBLEMAS E OBJETIVO (S)

Segundo Gates e Taylor, o uso da aprendizagem de máquina na detecção de anomalias é diferente de outros domínios de aplicação (2007), e segundo SBSeg, “detectores tendem a gerar grande quantidade de falsos positivos” (2011).

Devido à grande quantidade de malwares e suas variantes que surgem todos os dias é necessário que a aprendizagem de máquina seja utilizada como um parâmetro de análise de um especialista, em conjunto com outros parâmetros que o ajudem a identificar se a classificação do algoritmo se trata de um falso positivo ou não.

Sendo assim este trabalho tem como objetivos: a classificação de programas em maliciosos ou não utilizando aprendizagem de máquina supervisionada; a extração de regras de classificação adquiridas durante o processo de aprendizagem; a apresentação dos dados utilizados na classificação além das regras extraídas e o resultado do algoritmo através de uma interface gráfica aprovada pelo cliente.

PLANO DE AVALIAÇÃO DO TRABALHO

Será feita a aprendizagem com malwares coletados na internet pelo próprio autor. Para fins acadêmicos, a base está disponível em: https://github.com/lucascarvalhoroncoroni/MalwareAnalysis/tree/master/Malwares. A base contém malwares de diversos tipos, todos eles são arquivos executáveis. Também será feita a aprendizagem com arquivo não maliciosos, para fins de aprendizagem. Para fins acadêmicos, a base com programas não maliciosos, está disponível no link: https://github.com/lucascarvalhoroncoroni/MalwareAnalysis/tree/master/Softwares. Isso permite que outros projetos possam usar essas bases para comparar resultados com este artefato.

Após a aprendizagem, será feito um comparativo com a ferramenta malwr, com uma seleção de programas disponíveis em: https://github.com/lucascarvalhoroncoroni/MalwareAnalysis/tree/master/LearningTest. Esta ferramenta foi escolhida por ser uma ferramenta que se assemelha ao projeto apresentando dados extraídos do executável, além de apenas classificar o arquivo em malicioso ou não. Para a comparação será considerada a classificação da maioria dos antivírus.

O projeto será considerado bem-sucedido se obtiver uma taxa de acerto maior que a ferramenta.

PROPOSTA DO ARTEFATO

O artefato deste trabalho consiste em um classificador de arquivos em maliciosos ou não através de aprendizado supervisionado de máquina com a frequência de cada instrução do código objeto, dlls utilizadas pelo programa e strings dentro do programa.

O diagrama de arquitetura do artefato é apresentado na Figura 1. O diagrama mostra o browser fazendo um Request a todas as views, cada view responde com um Response.

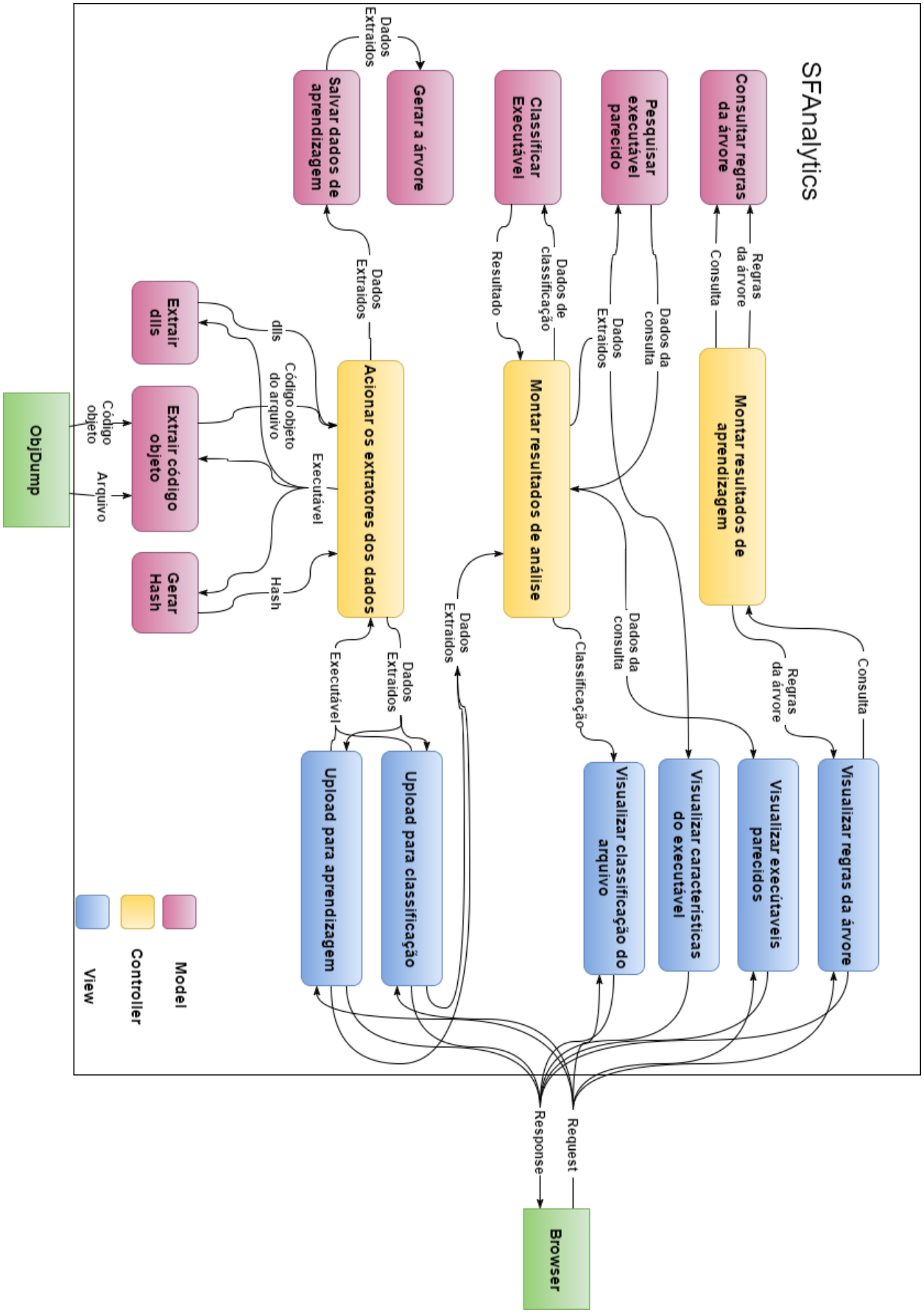

Figura 1 – Diagrama de Arquitetura. Fonte: O próprio autor.

TRABALHOS RELACIONADOS

Trabalho	Análise estática	Análise dinâmica	Interface gráfica	Aprendizado de máquina	Descrições em alto nível
VxStream	X	X	X		X
Malwr	X	X	X		
SFAnalytics	X		X	X	X

MÉTODO DE DESENVOLVIMENTO

O método de desenvolvimento escolhido foi o Scrum. O Scrum consiste em Times Scrum e seus papéis eventos e artefatos (SCRUMGUIDE, 2016). O Time Scrum é composto por Product Owner, Development Team e Scrum Master (SCRUMGUIDE, 2016). O Scrum prescreve quatro eventos formais: Sprint Planning, Daily Scrum, Sprint Review e Sprint Retrospective (SCRUMGUIDE, 2016), como mostra a figura 2. O Scrum define três artefatos, o Product Backlog, Sprint Backlog e o increment (SCRUMGUIDE, 2016), mostrados também na figura 2, sendo increment o incremento.

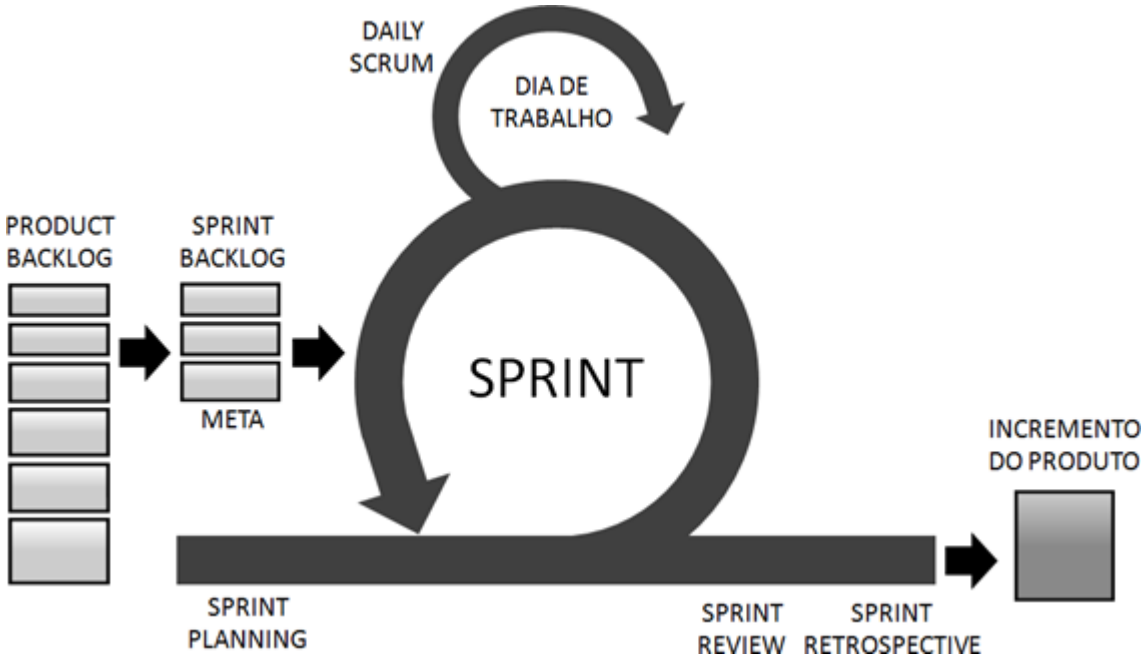

Figura 2 – Mecânica e ciclo do Scrum. Fonte: DEVMEDIA (2015).

O Product Backlog é uma lista que contém tudo o que o produto deverá ter (DEVMEDIA, 2015). No Scrum original o Product Owner é o responsável por fazer essa lista. Neste projeto o cliente fará o papel de Product Owner, mas não será responsável por fazer o Product Backlog, mas o cliente deve aprovar o que está sendo feito no projeto.

O Sprint Backlog é um apanhado de itens selecionados do Product Backlog (SCRUMGUIDE, 2016), ele serve como um guia para o Development Team, ou time de desenvolvimento, para saberem o que

deve ser feito durante o Sprint. Neste projeto o Development Team será somente o autor do projeto.

O Scrum Master, deve garantir o progresso do projeto (DEVMEDIA, 2015), mantendo a comunicação, monitorando o trabalho e organizando as reuniões. Neste projeto, os papéis serão desempenhados como um conjunto entre orientador, coorientador, cliente e autor do projeto. Orientador, coorientador e cliente sempre acompanharão o projeto, auxiliando, para o progresso do projeto, todos eles podem marcar reuniões a qualquer momento se julgarem necessário. O autor do projeto pode marcar reuniões também, quando julgar necessário, para o progresso do mesmo.

Segundo SCRUMGUIDE, o Scrum é fundado na teoria de controle de processos empíricos, e essa teoria é fundada em três pilares, transparência, inspeção e adaptação (2016), a flexibilidade de reuniões a qualquer momento pode ser considerado um item de transparência, e sendo um dos pilares do pilar do Scrum, pode trazer vantagens ao Scrum original, que define o Sprint Planning, a Daily Scrum, o Sprint Review e o Sprint Retrospective como os únicos eventos, ou reuniões, durante o desenvolvimento do projeto. Destes eventos, nesse projeto não haverá a Daily Scrum, que são reuniões diárias para o acompanhamento do time de desenvolvimento. Não haverá também o Sprint Planning, a Sprint Retrospective e a Sprint Review, haverá uma reunião fixa toda a semana, que substitui estas.

A reunião fixa a cada semana desempenha o mesmo papel que desempenhariam o Sprint Planning, planejamento dos itens do Product Backlog que passam para o Sprint Backlog, uma análise do incremento para efetuar modificações no Product Backlog se necessário, e a Sprint Retrospective, uma análise do time de desenvolvimento para autocrítica. A reunião será um acompanhamento do projeto, onde serão definidos os itens do Product Backlog que farão parte do Sprint Backlog, a reunião serve de acompanhamento do projeto, onde os envolvidos podem dar sugestões de melhorias, e tudo fica conforme a aprovação do cliente.

CRONOGRAMA

Identificação da Atividade	Descrição	Duração	
		Início	Fim
A1	Gerenciar o TCC	23/2/15	07/12/15
A2	Definição do projeto	23/02/17	24/03/17
A3	Definição das tecnologias utilizadas no projeto	07/03/17	31/03/17
A4	Definição dos dados utilizados para AM	07/03/17	31/03/17
A5	Realização do diagrama de arquitetura	22/03/17	24/05/17
A6	Realização do plano de trabalho	22/03/17	16/06/17
A7	Realização do diagrama de fluxo de dados	22/03/17	10/04/17
A8	Realização do diagrama de classes	25/03/17	19/04/17
A9	Implementação do artefato	03/04/17	13/10/17
A10	Validação com o cliente da GUI	12/04/17	29/05/17
A11	Montagem da Base de Dados	12/06/17	30/06/17
A12	Definição do algoritmo de aprendizagem	03/07/17	16/08/17

A13	Inserção do algoritmo de aprendizagem	03/07/17	16/08/17
A14	Avaliar e Validar o Trabalho	18/08/17	18/10/17
A15	Escrever monografia	10/10/17	27/11/17
A16	Preparar defesa do TCC	10/10/17	27/11/17

DISTRIBUIÇÃO DE ATIVIDADES

Identificação da Atividade	Primeiro Semestre Mês/Semana																				
	Fev				Mar					Abr				Mai				Jun			
				1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
A1				X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
A2				X	X	X	X	X														
A3						X	X	X	X													
A4								X	X	X												
A5								X	X	X	X	X	X	X	X	X						
A6								X	X	X	X	X	X	X	X	X	X	X	X			
A7								X	X	X	X											
A8								X	X	X	X	X										
A9										X	X	X	X	X	X	X	X	X	X	X	X	X
A10											X	X	X	X	X	X	X					
A11																		X	X	X	X	X
A12																		X	X	X	X	X
A13																						
A14																						
A15																						
A16																						

Identificação da Atividade	Segundo Semestre Mês/Semana																				
	Jul				Ago				Set					Out				Nov			
	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
A1	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
A2																					
A3																					
A4																					
A5																					
A6																					
A7																					
A8																					
A9	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X						
A10																					
A11																					
A12	X	X	X	X																	
A13	X	X	X	X	X	X	X														
A14							X	X	X	X	X	X	X	X	X	X					
A15														X	X	X	X	X	X	X	X
A16														X	X	X	X	X	X	X	X

RESULTADOS ESPERADOS

Identificação do	Descrição	Identificação

Resultado		da Atividade
R1	Plano de trabalho	A1
R2	Relatório de Atividades	A2
R3	Diagrama de fluxo de dados	A3
R4	Diagrama de classes	A4
R5	Artefato computacional	A5
R6	Base de dados	A6

RECURSOS MATERIAIS

Recursos de hardware:

Notebook.

Recursos de software:

Objdump;

Strings;

Visual Studio.

UTILIZAÇÃO DOS RECURSOS MATERIAIS

Dia	Segunda-feira	Terça-feira	Quarta-feira	Quinta-feira	Sexta-feira	Sábado	Domingo
Horário	17h-23h	17h-23h	20h-23h	17h-23h	17h-23h	14h-18h	13h-17h
Recurso	Notebook	Notebook	Notebook	Notebook	Notebook	Notebook	Notebook

GRAU DE DIFICULDADE – ASPECTOS DE INOVAÇÃO E APRIMORAMENTO

Inovação	Grau de dificuldade
Mostrar como foi feita a classificação	Alto

Mostrar como foi feita a classificação – O artefato deve mostrar o que o algoritmo de aprendizado de máquina aprendeu para fazer a classificação. O grau é alto devido à falta de conhecimento do autor sobre como fazer isso e também interfere na decisão do algoritmo utilizado no projeto.

Aprimoramento	Grau de dificuldade
Django	Médio
Objdump	Baixo
Python	Médio
Strings	Baixo

Django – Framework para a realização do sistema. O grau é médio devido a falta de conhecimento do autor do framework.

Objdump – Ferramenta para extração do código objeto de um executável. O grau é baixo pois a ferramenta não é difícil de ser utilizada, a dificuldade é na criação de uma interface dela com o artefato.

Python – Linguagem de programação utilizada no projeto. O grau é médio pela falta de conhecimento aprofundado do autor nessa linguagem.

Strings – Ferramenta para extração do código objeto de um executável. O grau é baixo pois a ferramenta não é difícil de ser utilizada, a dificuldade é na criação de uma interface dela com o artefato.

ANÁLISE DE RISCOS

Entrada	Probabilidade	Risco	Alternativa
Base de dados	Baixa	Médio	Gerar base a partir de Malwares de forma automatizada
Objdump	Baixa	Leve	Existem outras ferramentas que fazem a mesma coisa.
Django	Alta	Médio	Pedir ajuda ao orientador, por ter mais conhecimento sobre o framework.
Notebook do autor	Média	Médio	Usar computadores da faculdade para implementar o projeto.

OUTRAS OBSERVAÇÕES

Para o controle de versionamento e backup estão sendo usados o github para a codificação e o Google Drive para documentos e

diagramas. Os dois tem suporte a versionamento e funcionam como backup.

REFERÊNCIAS

CNN, *Nearly 1 million new malware threats released every day.* Disponível em: <https://www.usnews.com/news/articles/2014/06/09/study-hackers-cost-more-than-445-billion-annually>. Acesso em 8 de abril de 2017.

U.S. News, *Study: Hackers Cost More Than $445 Billion Annually.* Disponível em: <http://money.cnn.com/2015/04/14/technology/security/cyber-attack-hacks-security/>. Acesso em 8 de abril de 2017.

Simpósio Brasileiro de Segurança da Informação e de Sistemas Computacionais, 11, 2011, Brasília. SBSeg 2011 Brasília: Sociedade Brasileira de Computação, 2011. 280 p.

Gates, C., Taylor, C. (2007) “Challenging the Anomaly Detection Paradigm: A Provocative Discussion” Em Proc. Workshop on New Security Paradigms.

SCRUMGUIDES. *The Scrum Guide*. Disponível em: <http://www.scrumguides.org/docs/scrumguide/v2016/2016-Scrum-Guide-US.pdf#zoom=100>. Acesso em 18 de abril de 2017.

DEVMEDIA. *Introdução ao Scrum*. Disponível em: <http://www.devmedia.com.br/introducao-ao-scrum/33724>. Acesso em 18 de abril de 2017.

DEFINIÇÕES E ABREVIATURAS

Artefato Computacional – sistema de *software* ou de *hardware*, ou ainda uma combinação dos dois, que será desenvolvido com vistas à solução de um ou mais problemas identificados em um ambiente de interesse.

Dll – Biblioteca de linkagem dinâmica.

Malware – Programa que danifica ou faz ações indesejadas no computador.

Relatório de Atividades – conjunto de lançamentos de eventos que ocorrem no decorrer do TCC, sempre que ocorrer: término previsto, atraso, antecipação ou cancelamento, considerando o início e o fim de uma atividade. Um lançamento é constituído: da identificação da atividade, sua descrição, sua data de início e sua data de fim, conforme proposto no Cronograma. Segue o status (término conforme cronograma, atraso, antecipação ou cancelamento). Caso o término não seja o esperado, devem ser incluídos: justificativa (o porquê do evento); encaminhamento (alteração do cronograma – pode ser apenas a proposta de uma nova data de fim, por conta de um atraso, ou o cancelamento da

atividade); e consequência (análise e alteração das atividades ainda não encerradas por conta do encaminhamento decidido). Esses lançamentos serão úteis para a escrita da monografia.