Методы построения моделей штатной работы ПО и алгоритмы выявления аномального поведения ПО

Жилкин Сергей Дмитриевич
МИФИ, факультет Информационной Безопасности

Предпосылки

Хорошо поддаются обнаружению:

- классические вирусы
- вирусы-трояны
- spyware/adware
- прочие вирусы, содержащие вредоносный код

Меньше внимания уделено:

- недекларированным возможностям (НДВ)
- обнаружению кода ПО, не являющимся вредоносным, но приводящим к несанкционированному доступу (НСД)

Задачи

Создание программного комплекса для:

- моделирования работы ПО в режиме, который заведомо считается доверенным
- использование построенной модели для анализа работы ПО на предмет нахождения аномалий поведения

Особенности:

- отсутствие требований экспертных знаний о ПО
- независимость от платформы и ОС моделируемого ПО
- возможность тиражирования

Целевое ПО

ПО пользовательского уровня можно поделить на:

1. специализированное алгоритмическое ПО
   - системные службы
   - службы, работающие в фоновом режиме
2. ПО с пользовательским интерфейсом
   - прикладное ПО
   - офисные пакеты Microsoft Office, StarOffice

Решения для моделирования ПО 1-го класса:

- ряд программ: ps-watcher, pwatch, pScan (для ОС Linux)

Для моделирования ПО 2-го класса:

- пока нет законченного продукта

Моделирование ПО

Предлагается отслеживать поведение по взаимодействию с ресурсами операционной системы:

- работа с файловой системой
- обращение к сетевым ресурсам
- вызовы функций драйверов
- прочее (реестр, принтеры, ОЗУ, …)

Система аудита, основывающаяся на:

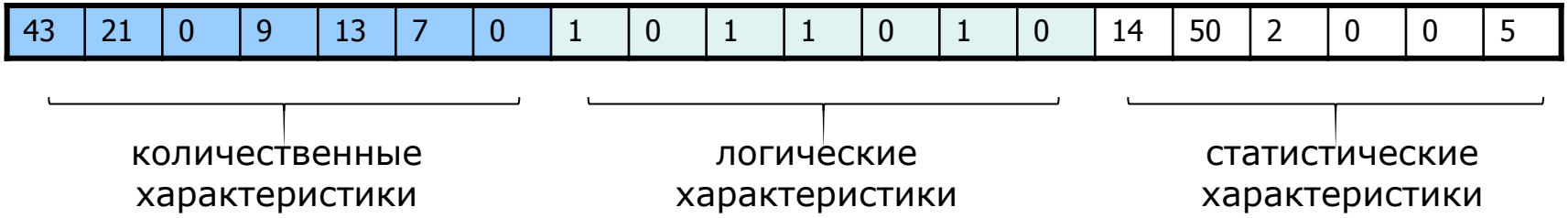
- журнальных файлах и внутреннем аудите ОС (например, «Snare LogAgent» от *InterSect*)
- системных вызовах на уровне драйверов ОС (например, «Инсайдер» от *ООО Праймтек*)

Описание поведения ПО

Предлагается описывать поведение процесса характеристиками трёх типов:

Количественные	число файловых операций, число операций с реестром ОС, число сетевых соединений, прочее
Логические	были ли порождены другие процессы от имени ПО, является ли ПО системным приложением, прочее
Статистические	выделения оперативной памяти, обращения к жёсткому диску за промежутки времени, прочее

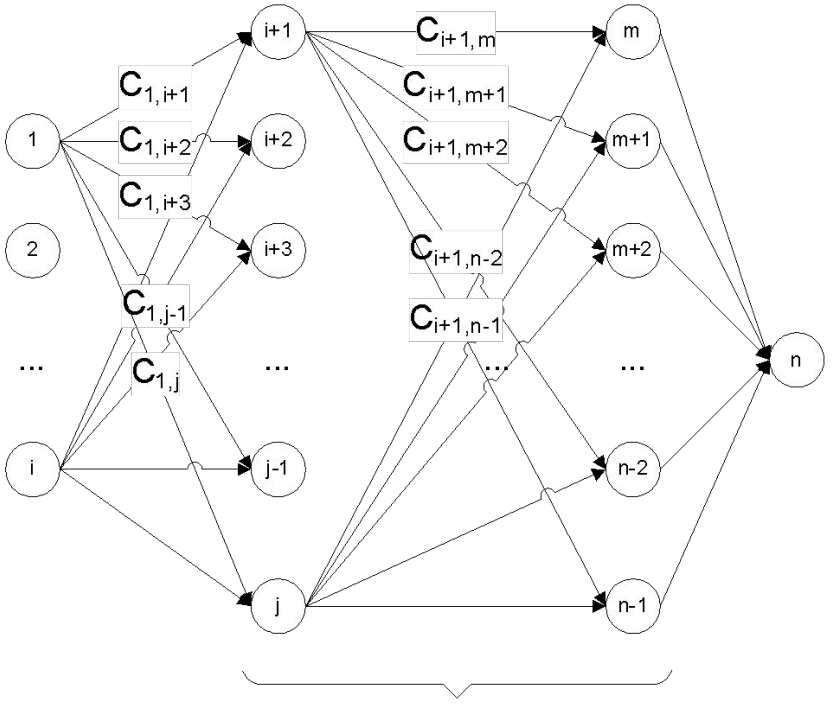
Профиль поведения ПО



- поведение ПО описывается набором чисел (вектором), в дальнейшем называемым профилем ПО
- профиль описывает поведение ПО за некоторое время работы

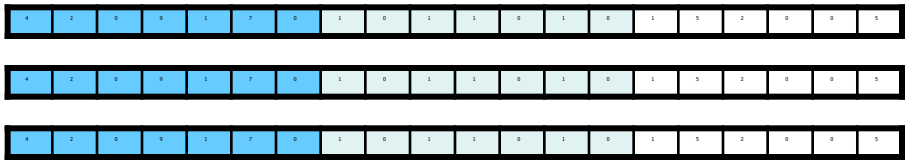
Распознавание с помощью нейронных сетей

- Используются для распознавания образов
- Выходом является результат соответствия поданного на вход образа эталонному, на который обучена сеть
- Обучение методом обратного распростра-нения ошибки

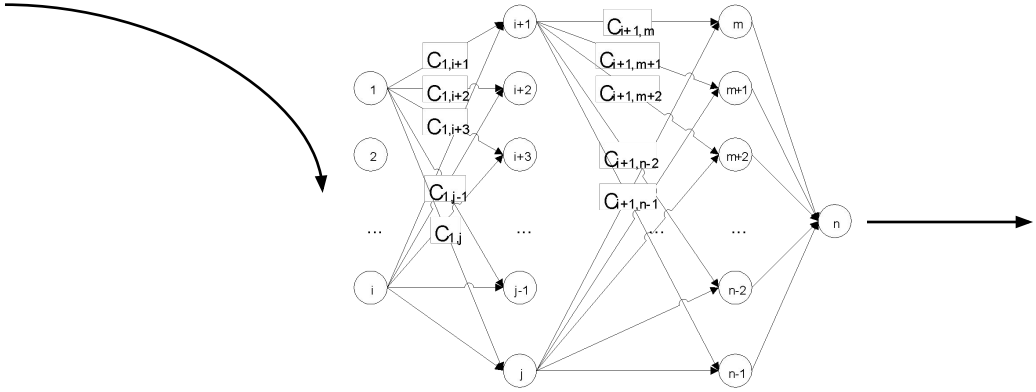


Итерации обучения нейронной сети

- Множество профилей для различных запусков моделируемого ПО
- Корректировка нейронной сети, чтобы на её выходе для каждого профиля был положительный результат (близкий к 1)



профили моделируемого ПО, описывающие работу в штатном режиме



$p \approx 1$

вероятность соответствия

Разделение работы на фазы

Трудности:

- заранее не определённая последовательность выполнения функций ПО
- заранее не определённое время работы ПО

Решение:

- разделение данных о поведении на фазы следующих типов:
  - запуск ПО
  - специфические действия ПО
  - завершение работы ПО

Специфические действия ПО

Автоматическое выявление специфических действий в поведении ПО

Может быть несколько специфических действий

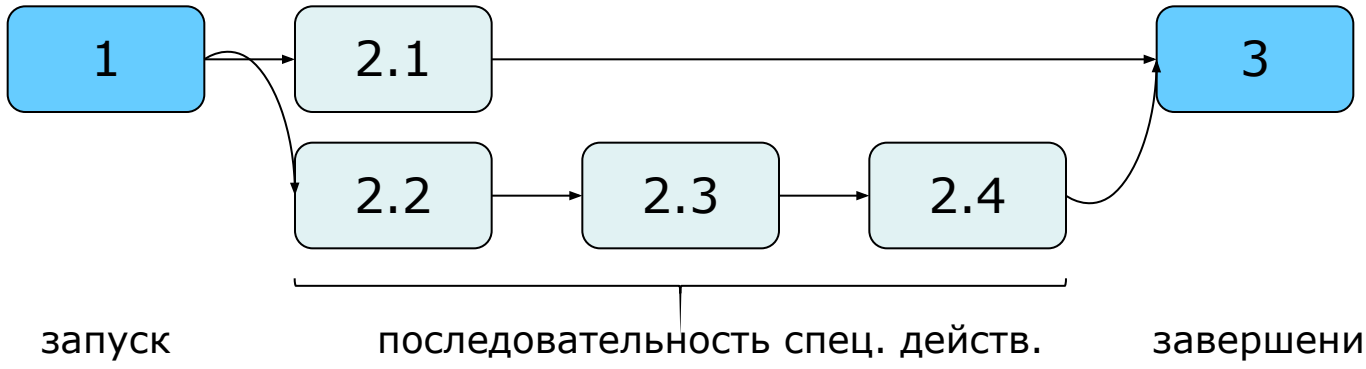
Для каждого из них обучается специальная нейронная сеть

Модель поведения

Модель поведения ПО состоит из:

1. нейронная сеть для запуска
2. набор нейросетей для специфических действий
3. нейронная сеть для завершения работы

Имея экспертные данные о поведении ПО можно задать гибкую/жёсткую последовательность выполнения специфических действий:



Анализ работы ПО

Динамическое выделение фаз работы в данных, поступающих о поведении ПО

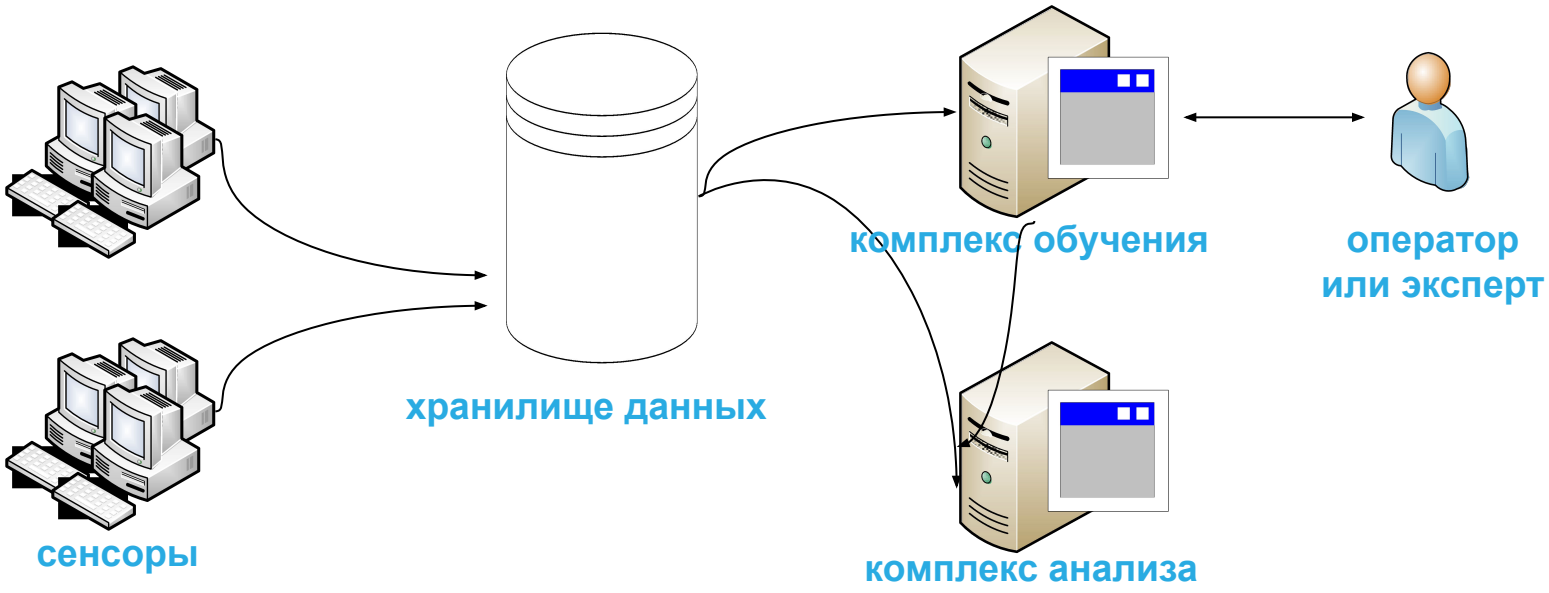
Профиль каждой фазы подаётся на вход соответствующей нейронной сети

В случае низкой вероятности соответствия профиля:

- можно говорить об аномалии поведения
- возможно, нарушен порядок выполнения специфических действий
- известна фаза, в которой выявлена аномалия

Структура комплекса

- программы-сенсоры, поставляющие информацию о действиях ПО
- автоматизированные средства обучения
- автоматические средства обнаружения аномалий



Применение на практике

Комплекс испытывался на: Microsoft Office, Adobe Acrobat, Internet Explorer, системных службах и ряде тестовых программ.

Microsoft Word:

- все из более 30 макро-вирусов обнаружены (в том числе Fries.a, Over.a, Want, Nail.a, Antiavs)
- обнаружена подмена исполняемого файла services.exe
- обнаружено заражение вирусом Downadup

Adobe Acrobat 7.0:

- обнаружена НДВ исполнения произвольного кода в специально подготовленном PDF файле

Заключение

В представленном комплексе реализовано:

- Моделирование и анализ работы ПО с заранее неизвестным поведением с целью выявления НДВ
- Автоматизированное обучение, не требующее экспертных данных о ПО
- Нахождение фазы работы, в которой произошла аномалия поведения
- Независимость от ОС, под которой работает моделируемое ПО