Секция «Вычислительная математика и кибернетика»

Иерархическая топология декомпозированных алгоритмов для обнаружения вредоносного исполнимого кода

***Гайворонская С.А.*[1], *Гамаюнов Д.Ю.*[2]**

1 - Московский государственный университет имени М.В. Ломоносова, Факультет вычислительной математики и кибернетики, 2 - Московский государственный университет имени М.В. Ломоносова, Факультет вычислительной математики и кибернетики, Москва, Россия

E-mail: sadie@lvk.cs.msu.su

Темой работы является обнаружение вредоносного исполнимого кода в высокоскоростных каналах передачи данных. В рамках работы рассматривается механизм распространения ботнетов с помощь сетевых червей путем эксплуатации уязвимости переполнения буфера.

Вредоносный код, эксплуатирующий подобную уязвимость, называется шеллкодом. Так как в работе рассматривается не просто удаленная эксплуатация уязвимостей, а строительство (распространение) ботнета, задача имеет ряд важных особенностей. Как любой массовое феномен, распространение червя проще заметить в крупных масштабах (на высокоскоростных каналах), чем на оконечных узлах. Учитывая два эмпирических закона: Мура и Гилдела, гласящих о том, что пропускная способность коммуникационных ресурсов растет быстрее вычислительной сложности ресурсов, доступных за одну и ту же стоимость, алгоритмы обнаружения шеллкодов должны иметь разумную вичислительную сложность. Кроме того, в виду большого объема передаваемой информации, критичным так же является вопрос ложных срабатываний алгоритов.

В результате анализа существующих алгоритмов обнаружения шеллкодов было выявлено, что существующие алгоритмы не применимы к высокоскоростным каналам: либо они имеют приемлимую вычислительную сложность, но характеризуются большим процентом ложных срабатываний, либо методы характеризуются высокой точностью, но не в состоянии обрабатывать входные данные в режиме реального времени.

Целью данной работы является разработка и реализация средства распознавания шеллкодов в высокоскоростных каналах передачи данных. При этом требуется разработать следующий комбинированный алгоритм обнаружения шеллкодов:

- вероятность ложных срабатываний алгоритма минимальна;
- вычислительная сложность алгоритма минимальна по сравнению с простой комбинацией алгоритмов;
- обеспечивается полное покрытие классов шеллкодов;

Разработка такого классификатора подразделяется на следующие подзадачи:

1. Выделение набора признаков ВПО и классификация шеллкодов.
2. Построение библиотеки элементарных классификаторов – построение набора алгоритмов, обнаруживающих специфичные классы шеллкодов.

3. Алгоритм выполнения классификатора – решение оптимизационной задачи генерации графа из элементарных классификаторов.

В результате анализа исследовательских статей, исходных кодов современных эксплойтов (где они доступны) вручную выделен набор признаков вредоносного исполнимого кода – как статических, так и динамических. Статическими являются такие признаки, которые могут проявить свое наличие в исследуемом наборе инструкций без их непосредственного исполнения. Динамические признаки могут быть обнаружены лишь в результате исполнения кода.

На основе выделенных признаков вредоносного кода, пространство шеллкодов было разбито на 19 частично-перекрывающих классов или «семейств» так, что каждый существующий шеллкод по схожести демонстрируемых им вредоносных признаков может быть отнесен к одному из выделенных классов. В частности, класс шеллкодов, содержащих многобайтный НОП-эквивалентный след, характеризуется следующим набором признаков: корректное дизассемблирование входных данных в цепочку инструкций длины, превышающей заданный порог; корректное дизассемблирование входных данных, начиная с каждого смещения; наличие только таких инструкций, которые не влияют на ход выполнения программы, а просто увеличивают программый счетчик. Стоит заметить, что выделенные признаки ВПО могут быть как широкоприменимыми (в частности, корректное дизассемблирование в цепочку инструкций заданной длины), так и узконаправленными. Последние признаки могут характеризовать ВПО, эксплуатирующее конкретную уязвимость (multiply host header, к примеру).

Библиотека элементарных классификаторов частично построена на основе выявленных признаков ВПО (каждый из таких признаков потенциально может являться простейшим классифкатором), а так же путем декомпозиции существующих алгоритмов обнаружения шеллкодов по используемым в них эвристикам.

Используя набор элементарных классификаторов, можно сформулировать задачу автоматической генерации гибридного классификатора как составление ориентированного графа G(V,E) с определенной топологией, где {V} – множество узлов- классификаторов, а дуги {E} представляют из себя пути следования потока данных. Предполагается, что классификаторы не осуществляют передачу потока, который был определен как легитимный, в другие классификаторы. Однако, если какой-либо классификатор определил анализируемый поток как вредоносный, такой поток перенаправляется в другие классификаторы на следующем уровне графа.

Для решения задачи предложен переборный алгоритм, строящий иерархическую топологию их элементарных классификаторов по следующим признакам:

- на каждом уровне обеспечивается покрытие всех доступных классов шеллкодов;
- каждый уровени оптимален в терминах вычислительной сложности и ложных срабатываний.

В проведенных экспериментах предложенный классификатор с иерархической топологией сравнивается с линейной топологией классификатора. Эксперименты показали, что иерархический метод характеризуется схожими значениями доли ложных срабатываний, но в разы эффективнее по своим скоростным характеристикам на смешанных и

легитимных наборах данных. В частности, на легитимном наборе данных предлагаемый подход показал в 45 раз лучшее время.

Литература

1. Y. I. Zhuravlev, Algebraic approach to the solution of recognition or classification problems. Pattern recognition and image analysis, 1998, vol. 8; no.1, 59-100

2. Brett Stone-Gross, Marco Cova, Lorenzo Cavallaro, Bob Gilbert, Martin Szydlowski, Richard Kemmerer, Christopher Kruegel, Giovanni Vigna Your Botnet is My Botnet: Analysis of a Botnet Takeover. Technical report, University of California, May 2009

3. E. Filiol Metamorphism, formal grammars and undecidable code mutation. International Journal of Computer Science,2, 2007

4. Cormen T., Leiserson C., Rivest R., Stein S. Introduction to algorithms, 3rd edition // The MIT Press, Cambridge, Massachu- setts. 2009.