МОДЕЛИРОВАНИЕ ПРЕОБРАЗОВАНИЙ ИНФОРМАЦИОННЫХ СВЯЗЕЙ В АРХИТЕКТУРЕ КИБЕРБЕЗОПАСНОСТИ СИСТЕМ С ИСПОЛЬЗОВАНИЕМ АЛГОРИТМОВ НА ГРАФАХ И ФОРМАЛЬНОЙ МОДЕЛИ "TAKE-GRANT"

Шумов А.С.

Институт проблем управления им. В.А. Трапезникова РАН,
Россия, г. Москва, ул. Профсоюзная д.65
mau17@list.ru

Аннотация: Рассмотрен вопрос моделирования преобразований информационных связей в архитектуре кибербезопасности систем на графах, с использованием формальной модели "take-grant". Предложены алгоритмы моделирования для различных исходных условий задачи. Создан программный модуль, реализующий предложенные алгоритмы.

Ключевые слова: архитектура кибербезопасности, моделирование, графы, модель "take-grant".

Введение

При проектировании, испытании и эксплуатации систем, к которым предъявляются требования безопасности (в частности, к таким системам относятся объекты атомной энергетики), иногда возникает необходимость изменения архитектуры кибербезопасности системы. Эта необходимость может быть вызвана, например, появлением новых угроз безопасности, выявлением слабых мест в текущей архитектуре, и т. д. В системах с дискреционным разграничением доступа эти преобразования сводятся в основном либо к исключению некоторых путей распространения информации, либо к исключению некоторых элементов системы, при этом система должна сохранять функциональность и соответствие назначенным для нее требованиям безопасности.

В данной работе рассматривается случай с исключением путей распространения информации, таким образом, целью преобразования архитектуры кибербезопасности системы является здесь обеспечение невозможности распространения информации в системе от некоторого элемента A к некоторому элементу B.

1 Постановка задачи

Для достижения обозначенной цели требуется разорвать все пути, по которым информация может распространяться от элемента A к элементу B, что означает необходимость разорвать некоторые информационные связи между некоторыми элементами системы. При этом, в общем случае, следует стремиться к тому, чтобы вносимые в систему изменения были минимальными. Оценка степени вносимых изменений в данном случае должна учитывать как количество разрываемых связей, так и их важность. Таким образом, возникает задача - определить, какие именно связи должны быть разорваны для достижения поставленной цели, с учетом указанных выше требований.

При этом, задача может быть поставлена как без ограничений, так и с введением в нее дополнительных условий. Рассмотрим три основных варианта постановки задачи:

1) Без ограничений. Требуется найти такие информационные связи, при разрыве которых все пути от элемента A к элементу B окажутся разорванными. Дополнительных условий не накладывается.

2) В задачу из п.1 вводится дополнительное условие — некоторые (заданные) информационные связи в системе должны быть сохранены, их разрыв является недопустимым.

3) В задачу из п.1 вводится дополнительное условие — по итогам преобразований, некоторые (заданные) элементы C и D системы должны сохранить информационную связь между собой, т. е. должен существовать как минимум один путь, по которому информация от элемента C может поступать к элементу D.

Данную задачу можно решать, моделируя систему с помощью графов, например, используя модель «take-grant» («брать-давать»). В таком случае, система представляется в виде ориентированного взвешенного графа, где вершинами служат элементы системы, а ребрами — информационные связи между ними. Такой граф будем называть графом безопасности системы. Типы информационных связей в нем могут быть различными (для модели «take-grant» основные типы связей — r (read), w (write), t (take), g (grant)) [1], но в рамках данной задачи поиск решения для каждого типа связей должен производиться отдельно. Пример графа безопасности системы приведен на рис. 1, Ri - веса соответствующих связей.

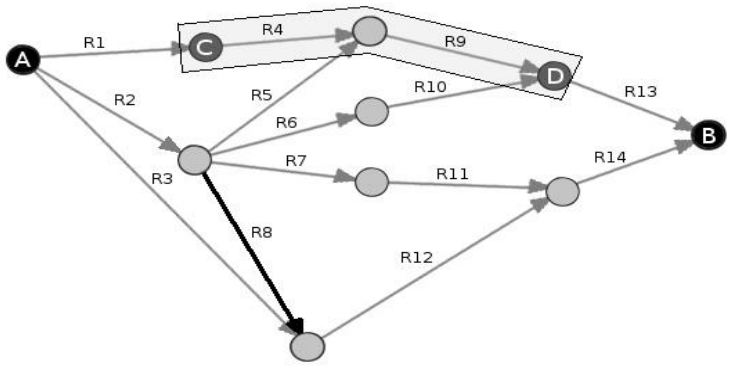

Рис. 1. Пример графа безопасности

В таком случае, задачу можно математически сформулировать следующим образом. Пусть граф безопасности G = (V, E) — взвешенный орграф, отражающий архитектуру системы, где V — множество его вершин, соответствующих элементам этой системы, а E — множество его ребер, представляющих информационные связи между элементами. Вес каждого ребра определяется важностью соответствующей информационной связи, направленность ребра соответствует направлению связи. В качестве критерия, отражающего степень изменений, вносимых в систему, примем суммарный вес изменяемых связей. Таким образом, для решения задачи требуется найти реберное разделяющее множество наименьшего веса между A, B ∈ V.

2 Методы решения задачи

Задача о наименьшем разрезе является одной из стандартных в теории графов и имеет несколько стандартных методов решения. Обычно при решении задач о реберном разрезе с заданными источником и стоком исходят из теоремы Форда-Фалкерсона, в соответствие с которой величина максимального потока в графе равна величине пропускной способности его минимального разреза. [2] В таком случае, за значения пропускных способностей связей принимаются значения их веса, а для решения задачи используются алгоритмы поиска максимального потока.

Основными методами решения таких задач являются метод Форда-Фалкерсона и его частные случаи, такие как алгоритм Эдмондса-Карпа (вычислительная сложность — $O(ve^2)$, где v — количество вершин в графе, e — количество ребер) и алгоритм Дийница (вычислительная сложность — $O(v^2e)$). [2] Еще один метод — алгоритм проталкивания предпотока и его частные случаи. Вычислительная сложность этого алгоритма без усовершенствований — $O(v^2e)$. [2]

3 Решение задачи

Предлагаются следующие алгоритмы решения задачи для каждого из трех ее вариантов, обозначенных выше.

3.1 Задача без ограничений

Этот случай является наиболее простым — здесь требуется применить на графе безопасности системы один из алгоритмов поиска минимального разреза, приняв в качестве источника вершину, соответствующую элементу системы A, а в качестве стока — вершину, соответствующую элементу

системы B (см. рис. 1). В результате работы алгоритма будет получен список ребер, составляющих минимальный разрез, и соответствующих тем информационным связям, которые требуется разорвать для достижения поставленной цели.

3.2 Задача с дополнительным условием, требующим сохранения определенных связей

В этом случае, прежде чем применять стандартные методы поиска минимального разреза, в граф безопасности нужно внести изменения — а именно, в качестве веса каждого из ребер, соответствующих каждой информационной связи, которую необходимо сохранить (в примере выделена жирным, см. рис. 1), ввести значения, бесконечно большие по сравнению со значениями веса остальных ребер графа безопасности. После данных преобразований, следует произвести действия, описанные в пункте 3.1. Следует отметить, что в данном случае задача может не иметь решения.

3.3 Задача с дополнительным условием, требующим сохранения связи между определенными элементами системы

Это наиболее сложный случай, также требующий предварительных преобразований графа безопасности. По условию задачи, нам нужно сохранить как минимум один путь между некоторыми заданными элементами — и, соответственно, между соответствующими им вершинами графа безопасности (в примере - вершины C и D, см. рис. 1). Для этого можно действовать по аналогии с предыдущим пунктом — присвоить бесконечно большие значения весам всех ребер, составляющих этот путь. Но если такой путь не один, возникает задача выбора того пути, который будет сохранен. Критерии выбора такого пути могут быть различными. В качестве одного из вариантов критерия предлагается наибольший средний вес ребра, являющегося элементом данного пути.

В данном случае задача также может не иметь решений. Необходимым условием для существования решений задачи в данном и предыдущем вариантах является следующее - после всех проведенных предварительных преобразований графа безопасности, в составе каждого пути, соединяющего вершины A и B, должны существовать связи с небесконечным значением веса.

3.4 Программная реализация

Для реализации описанных выше алгоритмов был создан программный модуль, написанный на языке Python, использующий библиотеку NetworkX. NetworkX — библиотека Python-a, предназначенная для работы с графами и сетями. В составе этой библиотеки существуют готовые функции для решения различных задач на графах — в том числе, задачи поиска наименьшего реберного разреза. В данном случае, была использована функция minimum_cut, основанная на теореме Форда-Фалкерсона. В рамках данной задачи, за величину пропускной способности каждой связи был принят ее вес. В качестве алгоритмов поиска максимального потока функция может использовать алгоритм проталкивания предпотока либо алгоритм Эдмондса-Карпа. Использование того или иного алгоритма определяется ключом, переданным функции. [3] В данном случае был применен алгоритм проталкивания предпотока, так как для большинства графов, которые могут исследоваться в рамках данной задачи, этот алгоритм имеет меньшую вычислительную сложность — v^2e против ve^2 [2], учитывая, что в графах, отображающих реальные системы со сложной архитектурой в большинстве случаев количество вершин v будет меньше количества связей e.

Исходные данные о графах безопасности хранятся в базе данных, в формате json. Реализованный программный модуль принимает в качестве входных данных идентификаторы, соответствующие в базе данных исследуемому графу и вершинам, относительно которых нужно найти разрез, осуществляет чтение информации о них из базы данных, рассчитывает минимальный разрез и выводит информацию о нем пользователю, а также сохраняет ее в отдельный файл. Работа модуля была успешно протестирована на различных графах безопасности, с различными наборами входных данных.

Литература

1. *Bishop, M.* Computer Security: Art and Science. — Boston: Addison Wesley. — 2003. — 1136 p.
2. *Кормен, Т., Лейзерсон, Ч., Ривест, Р., Штайн, К.* Алгоритмы: построение и анализ = Introduction to Algorithms / Под ред. И. В. Красикова. — 2-е изд. — М.: Вильямс, 2005. — 1296 с. — Глава 26.
3. *Hagberg A., Schult D., Swart P.* NetworkX Reference, Release 1.11. Jul 05, 2017. — 552 p.