Программная реализация имитации социоинженерных атак с помощью марковских полей [1]

Абдурахманова К.Ф., студентка кафедры информатики СПбГУ, camilla.si@yandex.ru;
Абрамов М.В., м.н.с. лаб. ТиМПИ СПИИРАН, м.н.с. лаб. социального компьютинга ИПИ МПГУ, аспирант каф. информатики СПбГУ, mva16@list.ru;
Азаров А.А., к.т.н, консультант ректората СПбГУ, artur-azarov@yandex.ru;
Тулупьев А.Л., д.ф.-м.н., доц., зав. лаб. ТиМПИ СПИИРАН, проф. каф. информатики СПбГУ, alexander.tulupyev@gmail.com;
Тулупьева Т.В., к.пс.н, доц., доц. СЗИУ РАНХиГС, ст.н.с. СПИИРАН, доц. СПбГУ, tvt100a@mail.ru

Санкт-Петербургский Государственный Университет
Санкт-Петербург, Россия

Аннотация

В данной статье рассмотрен существующий подход к моделированию социоинженерных атак. Предложен алгоритм, позволяющий находить вероятности доступа атакующего к элементам модели. Рассмотрены аспекты программной реализации алгоритма и модели.

Введение

В настоящее время информационные технологии используются повсеместно, ежедневно растёт объём информации, вместе с тем растёт и её ценность. Всё большее количество компаний сталкивается с проблемами информационной безопасности.

В 2014 году компанией Arbor Networks было проведено исследование, в рамках которого произвели анализ динамики количества DDOS-атак на информационную безопасность. Полученные результаты констатировали, что за год этот показатель увеличился вдвое [1]. Данная статистика подчёркивает актуальность проблем информационной безопасности.

[1]Статья содержит материалы исследований, частично поддержанных грантами РФФИ 14-01-00580, 15-01-09001-а.

Сегодня, наряду с интенсификацией программно-технических атак, средства защиты от которых становятся всё более сложными, для нарушения информационной безопасности компаний активно применяются социоинженерные атаки, в основе которых лежат манипулятивные воздействия на пользователя. Злоумышленники в этом случае воздействуют не на систему, а на пользователя системы, имеющего определённые психологические особенности и основанные на них уязвимости. Исследователи склоняются к тому, что технической безопасности обычно уделяется достаточно внимания, и сейчас наиболее уязвимым элементом информационных систем остается человек [2].

Социальная инженерия становится все более популярной в связи с повышением роли социальных сетей, электронной почты или других видов онлайн-коммуникации в нашей жизни. В то время как на рынке доступно огромное количество продуктов для обеспечения безопасности, человек остается слабым звеном [3].

Следует учитывать, что при социоинженерных атаках обнаружение утечки данных для компании происходит после нанесения урона. Поэтому остро стоит проблема анализа поражаемости документов в системе, а также ее пользователей, на предмет социоинженерной атаки [4]. Таким образом, одним из аспектов анализа защищенности предприятия является анализ его защищенности от социоинженерных атак

Классификация угроз информационной безопасности

Существует различные подходы к классификации угроз информационной безопасности. В [5] описана наиболее подробная классификация угроз, основанная на источниках угроз, его положении, привлекательности атаки, степени возможного повреждения, вероятности успеха, характере атаки, и другим параметрам. В [6] приведена классификация источников угроз. Обобщая эти классификации, мы сформировать классификацию атак по источнику угроз:

1. Стихийные бедствия
2. Обусловленные техническими средствами
   (a) Внутренние
   (b) Внешние

3. Обусловленные действиями субъектов
   (a) Внутренние
   (b) Внешние

Согласно [5], внутренние источники атак наиболее опасны, так как пользователи имеют доступ к техническим средствам и конфиденциальной информации, и их действия могут принести значительный урон организации. В [2] отмечается, что высокая ответственность за безопасность и конфиденциальность данных в информационных системах (ИС) несет сам пользователь. Однако, авторы этих статей не выделяют отдельную категорию угроз, когда третья лица воздействуют на пользователей ИС с целью получить доступ к конфиденциальным данным, извлечь выгоду, нарушить работу предприятия или нарушить нормальную работу ИС.

Такое воздействие называется «социоинженерными атаками». В стандартах от Microsoft [7] сообщается, что в таких атаках злоумышленник использует приемы социальной инженерии с целью получения доступа к закрытой информации и ресурсам, и приводится классификация таких атак.

Подход Н.В. Хованова

Н.В. Хованов, обобщая и адаптируя в [8] результаты и положения ряда общетеоретических, частных и учебных публикаций, предложил моделировать комплекс «товар – посредник – потребитель» на основе реляционного подхода.

Применяя эту модель к социоинженерным атакам, авторы в [9] предложили модель комплекса «информационная система – персонал – злоумышленник». Между элементами модели предприятия существуют связи. В обобщенном, стохастическо-реляционном случае эти связи носят вероятностный характер, имитируя неопределенность и силу этих связей. Следует отметить, что этот подход включает в себя реляционный детерминированный подход к задаче: всем связям в таком случае назначается вероятность 1.

Такой подход основан на Марковском случайном поле [9]. Так, если атакующий X будет распространять своё воздействие через объекты $(A_1, A_2, A_3, ..., A_n)$, вплоть до целевого документа D , то вероятность доступа будет вычисляться следующим образом:

$$P_{X,D} = P_{X,A_1} * P_{A_1,A_2} * P_{A_2,A_3} * ... * P_{A_{n-1},A_n} * P_{A_n,D} \qquad (1)$$

, где $P_{A,B}$ означает вероятность перехода от узла графа A к узлу графа B. Однако, следует отметить, что в [9] вероятностный подход рассмотрен не полностью. Не указаны методы поиска наиболее уязвимых элементов, и аспекты конкретной реализации алгоритма нахождения вероятности уязвимости.

Описание модели

Элементы модели комплекса [9]:

- Документ – объект, имеющую материальную ценность; конечная цель атакующего.
- Хост – компьютер, с помощью которого пользователь осуществляет взаимодействие с документами.
- Пользователь – человек, работающий на предприятии. Именно с помощью взаимодействия с пользователями атакующий добивается своих целей.
- Атакующий – элемент, лежащий вне предприятия, но необходимый в модели для описания атаки. Рассмотрим имеющиеся связи предприятия:
- Атакующий-пользователь. Вероятность, присвоенная этой связи, означает то, что атакующему удалось с помощью пользователя воздействовать на предприятие. Эта вероятность зависит от способностей атакующего и пользователя, выраженных в числах – компетенциях [10]. Вероятность воздействия вычисляется как нормированный максимум разницы компетенций атакующего к компетенции пользователя. Связи между атакующими не существует, что позволяет одновременно моделировать несколько независимых атак.
- Пользователь-пользователь. Атакующий может распространить своё влияние с одного пользователя на другого; при этом успешность этого будет зависеть от личных отношений между пользователями. Эта вероятность задается экспертно.
- Пользователь – хост. Для получения доступа к документам атакующий должен попросить пользователя совершить определенные с определенным компьютером. Однако, есть случайные (с точки

зрения атакующего) факторы, препятствующие этому: у пользователя может не быть доступа к нему, или компьютер может быть занят. Мы не разделяем эти случаи, а указываем общую вероятность перехода в связи. Вероятность задается экспертно.

- Хост – хост. Так как компьютеры предприятия объединены в локальную сеть, можно с одного компьютера получить доступ к другому. Однако, из-за разграничений прав доступа или из-за особенностей сети этот доступ может быть неосуществим; это выражается в вероятности связи. Вероятность задается экспертно.

- Хост – документ. Получив доступ к нужному хосту, атакующий может с помощью пользователя совершить необходимые действия. Однако, у пользователя может не хватить прав, документ может быть перемещен, и так далее – мы все эти случаи обобщаем под вероятностью перехода. Вероятность задается экспертно.

Описание модели

Алгоритм должен для каждого элемента модели находить максимальную вероятность, с которой атакующий может получить к нему доступ. Предполагается, что атакующий действует оптимально. Для оценки этой вероятности возможно использовать адаптированный алгоритм поиска кратчайшего пути, который будет учитывать то, что мы оперируем вероятностями, при переходе вероятности должны перемножаться, и задача – найти максимальную вероятность. Необходимо выбирать такой алгоритм, который рассчитывает расстояния от атакующего до всех узлов.

Этим требованиям удовлетворяет алгоритм Дейкстры [11]. Поэтому мы будем использовать модификацию этого алгоритма, учитывающую, что мы работаем с вероятностями.

Псевдокод алгоритма:

```
dijkstra(s) =
for v in V
d[v] = 0
used[v] = false
d[s] = 1
for i in V
v = null
for j in C
```

```
if !used[j] and (v == null or d[j] > d[v])
v = j
if d[v] == 0
break
used[v] = true
for e : исходящие из v рёбра
if d[v] * e.len > d[e.to]
d[e.to] = d[v] * e.len
```

Создание программной библиотеки

Рисунок 1 содержит диаграмму классов, описывающих атакуемую организацию. Атакуемый комплекс можно представить в виде направленного графа, где узел — это атакующий, пользователь, хост, или документ. Наличие связи между парой узлов и подразумевает, что действие атакующего может оказать влияние на , если будет оказано влияние на . Связи могут быть нагружены такой дополнительной информацией, как вероятность успешного влияния. Таким образом, создадим класс Edge, который будет хранить информацию о связи между двумя объектами. Чтобы использовать все преимущества языка со строгой типизацией, Edge должен быть шаблонным классом, и содержать три поля: from, to и weight (вес связи).

Следует отметить, что указанные типы узлов формируют иерархию, где связи существуют только в пределах одного уровня, и на уровень ниже (поскольку, например, нет прямых связей между пользователем и документом). Поэтому класс Node для удобства реализации алгоритма мы тоже можем сделать шаблонным, с двумя параметрами: T и T2. T будет указывать на тип узлов на этом же уровне, T2 — на уровне ниже. Также, добавим методы, которые возвращают списки связей с узлами этого же уровня, и уровня ниже.

Создадим классы Attacker, User, Host и Document, являющиеся расширениями класса Node. Так как для Document не существует класса ниже, шаблонный параметр T2 укажем как Document.

Самый крупный класс — это AttackedOrganization, контейнер, вмещающий в себя все необходимые элементы модели атакуемого комплекса.

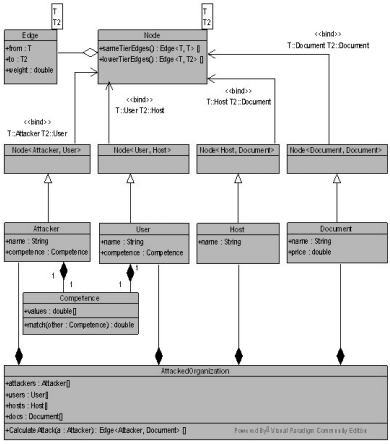

Рис. 1: Архитектура системы

AttackerOrganization

Этот класс содержит функцию CalculateAttack, принимающий объект-атакующего, рассчитывающий возможные атаки, и возвращающий список ребер, содержащих возможные атакуемые документы, и вероятность их успешной атаки. Также, этот класс предоставляет открытый доступ к спискам атакующих, пользователей, нападающих, хостов и документов, позволяя удалять и добавлять элементы модели предприятия. Кроме того, именно этот класс содержит необходимые вспомогательные функции для моделирования атаки.

Attacker

Класс атакующего предоставляет открытый доступ к своему имени. Также, методы AddLink и UpdateLinks позволяют добавлять одну связь и изменять все связи разом.

User

Класс пользователя предоставляет открытый доступ к своему имени. Также, методы AddLink и UpdateLinks позволяют добавлять одну связь и изменять все связи разом. Методы перегружены, позволяя либо добавлять связь с другим пользователем, либо с хостом. В случае

добавления связи с пользователем эта связь автоматически добавляется и в список связей другого пользователя. При вызове UpdateLinks все связи других пользователей с текущим удаляются, и заменяются на новые.

Host

Класс хоста предоставляет открытый доступ к своему имени. Также, методы AddLink и UpdateLinks позволяют добавлять одну связь и изменять все связи разом. Методы перегружены, позволяя либо добавлять связь с другим хостом, либо с документом. В случае добавления связи с хостом эта связь автоматически добавляется и в список связей другого хоста. При вызове UpdateLinks все связи других хостов с текущим удаляются, и заменяются на новые.

Document

Класс предоставляет открытый доступ к полям, обозначающим его имя и стоимость. Документы не имеют связей ни между собой, ни с низлежащим уровнем иерархии, поэтому вызов методов getSameTierEdges и getLowerTierEdges вызывают исключение.

Заключение

В данной статье был рассмотрен существующий подход к моделированию социоинженерных атак. Предложен алгоритм, позволяющий находить вероятности доступа атакующего к элементам модели. Рассмотрены аспекты программной реализации алгоритма и модели.

Список литературы

[1] А. Батогов, «В 2014 ГОДУ ЗНАЧИТЕЛЬНО УВЕЛИЧИЛОСЬ КОЛИЧЕСТВО DDOS-АТАК,» [В Интернете]. Available: http://hi-news.ru/internet/v-2014-godu-znachitelno-uvelichilos-kolichestvo-ddos-atak.html. [Дата обращения: 15 март 2016].

[2] C. W. Flink, «Weakest Link in Information System Security,» WAEPSSD, 2002.

[3] «Социальная инженерия, или Как «взломать» человека,» [В Интернете]. Available: https://blog.kaspersky.ru/socialnaya-inzheneriya-ili-kak-vzlomat-cheloveka/2559/. [Дата обращения: 22 03 2016].

[4] «Социальная инженерия,» [В Интернете]. Available: http://www.kaspersky.ru/internet-security-center/threats/malware-social-engineering. [Дата обращения: 22 03 2016].

[5] С. В. Вихорев, «Угрозы информационной безопасности,» Сетевые атаки и угрозы информационной безопасности, 2001.

[6] ВОЗМОЖНЫЕ УГРОЗЫ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ И ИХ СПЕЦИФИКА, (В Интернете). http://www.e-nigma.ru/stat/dip_2/. [Дата обращения: 2016 03 15].

[7] «Как защитить внутреннюю сеть и сотрудников компании от атак, основанных на использовании социотехники,» [В Интернете]. Available: https://technet.microsoft.com/ru-ru/library/cc875841.aspx. [Дата обращения: 15 3 2016].

[8] Н. В. Хованов, «Общая модель измерения ценности экономических благ,» Применение математики в экономике. Вып. 18, pp. 108-134, 2009.

[9] А. А. Азаров, Т. В. Тулупьева и А. Л. Тулупьев , «Агентоориентированный подход к моделированию комплекса «Информационная Система –Персонал – Злоумышленник»в задачах оценки защищенности от социоинженерных атак.,» Список-2012: Материалы всероссийской научной конференции по проблемам информатики, p. 374–377, 2012.

[10] А. А. Азаров, Т. В. Тулупьева и А. Л. Тулупьев, «Прототип комплекса программ для анализа защищенности персонала информационных систем построенный на основе фрагмента профиля уязвимостей пользователя,» Труды СПИИРАН, p. 21–40, 2012.

[11] А. В. Левитин, «Алгоритмы. Введение в разработку и анализ,» 2006.