Онтологический подход к анализу событий сетевой инфраструктуры

М. С. Михайлов, С. А. Кузьмин, И. В. Герасимов

Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им. В.И. Ульянова (Ленина)

MSMikhaylov@stud.etu.ru, SAKuzmin@etu.ru, IVGerasimov@etu.ru

Аннотация. **В работе представлена онтология для анализа инцидентов, происходящих в сетевой инфраструктуре предприятия, которая обеспечивает сбор и представление данных о событиях и действиях в информационной системе. Информация поступает из различных мониторинговых систем в режиме реального времени, что позволяет оперативно выявлять и классифицировать инциденты. Разработанные правила логического вывода автоматизируют анализ происшествий и формирование рекомендаций по их устранению. Для хранения и обработки знаний используется графовая СУБД GraphDB, обеспечивающая эффективную работу с онтологическими данными.**

Ключевые слова: моделирование сетевой инфраструктуры; анализ инцидентов; система поддержки принятия решений; онтология; правила логического вывода; GraphDB

I. Введение

В современном мире большинству организаций необходима собственная сетевая инфраструктура, которая позволяет эффективно взаимодействовать с клиентами, а также сотрудникам между собой. При этом, с развитием Интернета сетевое взаимодействие постоянно изменяется – что приводит к усложнению сетевой архитектуры и появлению новых угроз [1]. Для полноценного анализа инцидентов сетевой инфраструктуры необходимо отслеживать сложное взаимодействие интеллектуальных агентов между собой.

В данной работе рассматривается разработка системы, которая позволит упростить анализ инцидентов, а также предоставить поддержку в принятии решений системным администраторам. В ней имеется база данных уязвимостей сервисов, а также онтология, в которой аккумулируются данные о событиях и действиях, происходящих в сетевой инфраструктуре компании. Причём, данные поступают в неё из различных мониторинговых сервисов в режиме реального времени. Онтология содержит правила логического вывода, в которых рассматриваются конкретные произошедшие инциденты, они классифицируются по степени угрозы, а также вырабатываются рекомендации по их разрешению.

II. Проектирование онтологии

A. Описание онтологии на естественном языке

Ключевыми элементами разработанной онтологии являются классы инцидента (Incident) и расследования (Investigation).

Инциденты представляют собой зарегистрированные события, произошедшие в сетевой инфраструктуре. Каждый инцидент содержит: описание проблемы, время происшествия, IP-адрес источника атаки, целевой IP-адрес и порт, название атакованного сервиса, а также вовлечённую компанию и систему, которая зарегистрировала данный инцидент.

Возникновение инцидента инициирует процесс расследования (Investigation), цель которого – установить причины, участников и потенциальные последствия атаки. В расследовании фиксируются его статус, приоритет, результаты анализа, а также уязвимости, которые могли быть использованы злоумышленником. Оно связывается с конкретным инцидентом через атрибут «начато по инциденту» (started_for). В ходе расследования определяются атакующие субъекты, а также назначается аналитик, ответственный за разбор инцидента.

В онтологии представлены классы, моделирующие сетевую инфраструктуру компании. Ключевыми из них являются серверы (Server) и сервисы (Service). Серверы идентифицируются по IP-адресу и имени хоста. На них запускаются различные сервисы, которые характеризуются версией и портом.

Для мониторинга событий, происходящих в сетевой инфраструктуре, предусмотрен класс мониторинговой системы (Monitoring_System). При помощи него осуществляется фиксирование всех произошедших инцидентов в реальном времени.

В онтологии предусмотрены классы для различного программного обеспечения, которое может сообщать о происходящих событиях и действиях. Всё такое программное обеспечение в онтологии представлено обобщённым классом «Monitoring_System». В онтологии имеется несколько подклассов для мониторинговых систем, которые содержат специфичные для каждого типа программного обеспечения свойства.

Также в онтологии хранятся сведения о различных уязвимостях, которые злоумышленники использовали для осуществления атак, а также об уязвимостях, присущих конкретным сервисам.

На рис. 1 представлена семантическая сеть онтологии, которая позволяет рассмотреть ключевые взаимосвязи между классами.

B. Семантическая сеть онтологии

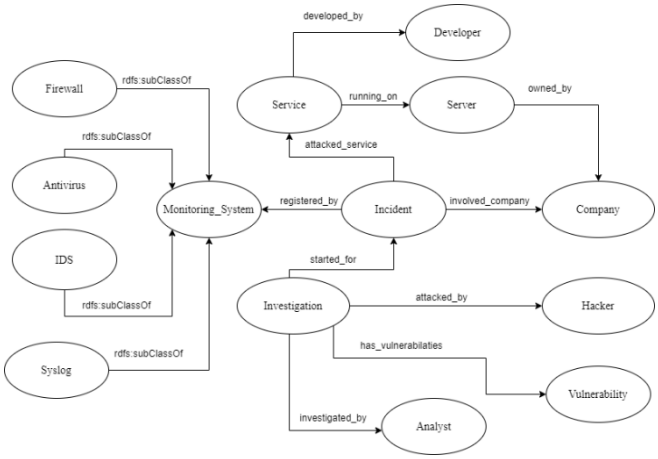

Рис. 1. Семантическая сеть онтологии для анализа событий сетевой инфраструктуры

C. Правила логического выводв в онтологии

Также были разработаны различные правила для автоматизации процессов обнаружения и реагирования на инциденты. Эти правила направлены на повышение эффективности защиты сетевой инфраструктуры и минимизацию ущерба от атак. Далее представлены примеры созданных правил.

1) Правило расследования инцидента при попытке проникновения в систему:

Если есть инцидент (Incident) и начато его расследование (Investigation). В описании инцидента содержится «system penetration» и выявлен источник атаки (source_ip). В ходе расследования обнаружен злоумышленник (Hacker). Тогда со злоумышленником становится ассоциирован IP-адрес атаки, статус расследования становится завершённым, а результатом расследования является строка «hacker ip added to block list».

2) Правило блокировки IP-адреса инцидента при DDOS-атаке:

Если есть инцидент (Incident) и начато его расследование (Investigation). В описании инцидента содержится «DDOS attack», выявлен источник атаки (source_ip), а число запросов от него к сервису (Service) превышает установленный порог (threshold). В ходе расследования обнаружен злоумышленник (Hacker). Тогда со злоумышленником становится ассоциирован IP-адрес атаки, статус расследования становится завершённым, а результатом расследования является строка «hacker ip added to block list».

3) Правило расследования инцидента при обнаружении вируса:

Если есть инцидент (Incident) и начато его расследование (Investigation). У Incident есть описание «malware» и существует связанная с ним система отслеживания (Monitoring_System), которая является антивирусом (Antivirus). Тогда статус расследования (Investigation) становится завершённым, а результатом расследования является строка «provide training to employees about online safety».

III. РАЗРАБОТКА ОНТОЛОГИИ В GRAPHDB

Разработка онтологии осуществлялась при помощи графовой СУБД GraphDB [2]. В неё были добавлены ранее описанные классы, свойства и отношения. На рис. 2–4 представлена работа с экземплярами и правилами онтологии в GraphDB [3]. Для внесения знаний в GraphDB использовался язык запросов SPARQL 1.1. Update [4].

A. Добавление новых знаний через SPARQL 1.1 Update

Unnamed

```
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX f: <http://cybersecurity.com/cybersec#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
INSERT DATA {
    GRAPH  <http://cybersecurity.com/cybersec#> {
        f:Incident1 rfd:type owl:NamedIndividual .
        f:Incident1 rdf:type f:Incident .
        f:Incident1 f:incident_time "2024-02-15T14:32:00Z"^^xsd:dateTime .
        f:Incident1 f:incident_description "system penetration" .
        f:Incident1 f:incident_source_ip "178.45.13.119" .
        f:Incident1 f:incident_target_ip "192.168.2.50" .
        f:Incident1 f:incident_target_port 443 .
        f:Incident1 f:attacked_service f:NGINX .
        f:Incident1 f:involved_company f:SecurityLab .
        f:Incident1 f:registered_by f:Snort .
    }
}
```

Рис. 2. SPARQL-запрос на внесение значений экземпляров в GraphDB

Unnamed

```
INSERT {
    GRAPH <http://cybersecurity.com/cybersec#> {
        ?hacker f:assigned_ips ?sourceIP .
        ?investigation f:investigation_status "resolved".
        ?investigation f:investigation_result "hacker ip added to blocklist"
    }
} WHERE {
    GRAPH <http://cybersecurity.com/cybersec#> {
        ?incident rdf:type f:Incident.
        ?incident f:incident_description "system penetration" .
        ?incident f:incident_source_ip ?sourceIP .
        ?investigation f:started_for ?incident.
        ?investigation f:attacked_by ?hacker .
    }
```

Added 4 statements. Update took 0.1s, moments ago.

Рис. 3. SPARQL-запрос на внесение правила логического вывода в GraphDB

B. Демонстрация невных знаний в GraphDB

	subject	predicate	object
1	f:Investigation1	f:attacked_by	f:Fancy_bear
2	f:Investigation1	f:has_vulnerabilities	f:NGINX_CVE_2022_30503
3	f:Investigation1	f:investigated_by	f:Mike
4	f:Investigation1	f:investigation_priority	"low"
5	f:Investigation1	f:investigation_result	"hacker ip added to blocklist"
6	f:Investigation1	f:investigation_status	"resolved"
7	f:Investigation1	f:started_for	f:Incident_1
8	f:Investigation1	rdf:type	f:Investigation
9	f:Investigation1	rdf:type	owl:NamedIndividual

Рис. 4. Демонстрация полученных неявных знаний в GraphDB

IV. Веб-приложение для работы с онтологией

Для взаимодействия специалистов с онтологией было разработано веб-приложение. Оно позволяет авторизованным сотрудникам расследовать инциденты в рамках онтологии предприятия. Причём, они могут одновременно работать как с одной онтологией, так и с несколькими. На рис. 5–7 представлено взаимодействие пользователя с онтологией, а именно: добавление новых правил логического вывода, а также извлечение явных и неявных знаний.

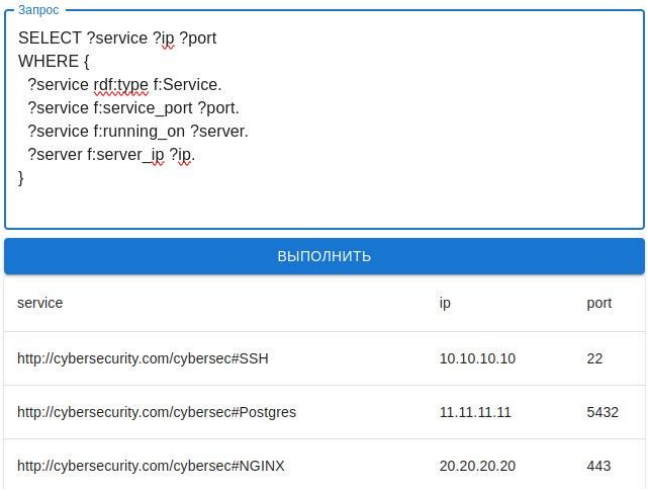

Рис. 5. Окно приложения для выполнения SPARQL-запроса на чтение явных знаний в онтологии

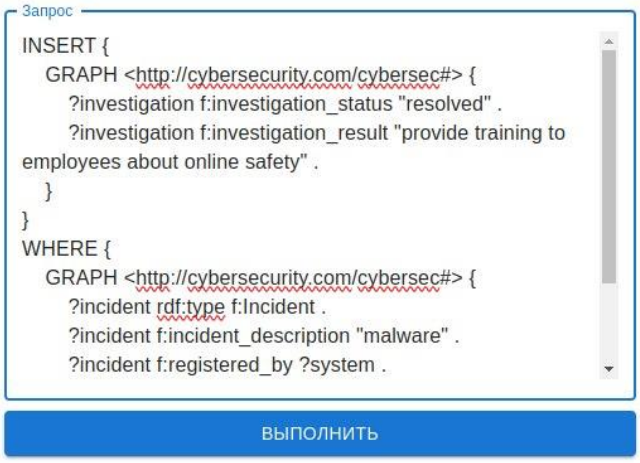

Рис. 6. Окно приложения для выполнения SPARQL-запроса на добавление правила логического вывода в онтологию

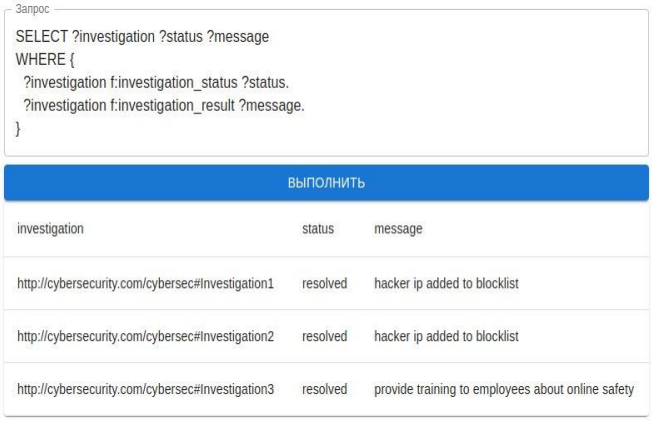

Рис. 7. Окно приложения для выполнения SPARQL-запроса на получение неявных знаний из онтологии

V. Заключение

Разработанная онтология и веб-приложение позволяют получать информацию обо всех событиях в сетевой инфраструктуре в режиме реального времени из различных источников. Она позволяет автоматизировать расследование тривиальных инцидентов, осуществлять поддержку системному администратору в принятии решений, а также обеспечивать одновременную работу множества сотрудников компании.

На текущий момент проект ещё находится в разработке. Возможно расширение онтологии за счёт появления новых классов, описывающих новые виды информационных систем и угроз, а также новых правил. Планируется дальнейшая интеграция сервисов с GraphDB за счёт добавления новых видов ограничений и возможности написания собственных правил в GraphDB.

Список литературы

[1] Загорский А.В., Ромашкина Н.П. Угрозы информационной безопасности в кризисах и конфликтах XXI века. Москва: ИМЭМО РАН, 2015. 151 с.

[2] What is GraphDB [Электронный ресурс]. URL: https://graphdb.ontotext.com/documentation/10.5/ (дата обращения: 10.02.2025 г.).

[3] Working with Workbench [Электронный ресурс]. URL: https://graphdb.ontotext.com/documentation/10.6/working-with-workbench.html (дата обращения: 27.02.2025 г.).

[4] SPARQL 1.1 Query Language [Электронный ресурс]. URL: https://www.w3.org/TR/sparql11-query/ (дата обращения: 03.03.2025 г.).