Автоматизация процесса выявления причин возникновения событий в ИТ-инфраструктуре

Овчинников Александр Алексеевич
Заведующий Учебно-исследовательской лабораторией Интернет технологий и сервисов
Национальный исследовательский университет Высшая школа экономики
123458, Москва, Таллинская ул. 34
aaovchinnikov@hse.ru

Старых Владимир Александрович
к.т.н., доцент, профессор
Национальный исследовательский университет Высшая школа экономики
123458, Москва, Таллинская ул. 34
vstarykh@hse.ru

Аннотация: Данная работа посвящена созданию информационно-аналитической системы (ИАС) выявления причинно-следственных отношений между событиями, фиксируемыми в ИТ-инфраструктуре. Назначением такой системы является повышение эффективности работы инженеров поддержки за счёт определения возможных причин возникновения событий, тем самым обоснованно направляя поиск и расследование событий, инцидентов и проблем. В работе показано использование шаблонов как способ первичной классификации событий, без выявления семантики/смысла события. Извлечение шаблонов из «сырых» записей является этапом предобработки. Для целей извлечения шаблонов был разработан шаблонизатор, автоматизирующий этот процесс. Заявленная система реализована при помощи открытых бесплатных, хорошо себя зарекомендовавших средств, таких как сервер журналирования Rsyslog и разработанное прикладное ПО -Шаблонизатор.

Ключевые слова: ИТ–инфраструктура, системный журнал, анализ, причинно-следственная связь, мониторинг, шаблон, журналирование, извлечение данных, большие данные.

Automation of the process of identifying the causes of events in the IT-infrastructure

Alexander A. Ovchinnikov
Head of the Research Laboratory of Internet Technologies and Services
National Research University Higher School of Economics
123458, Moscow, Tallinskaya Ul., 34
aaovchinnikov@hse.ru

Vladimir A. Starykh

Ph.D., Associate Professor, Professor
National Research University Higher School of Economics
123458, Moscow, Tallinskaya Ul., 34
vstarykh@hse.ru

Abstaract: This work is devoted to the creation of an information-analytical system for identifying causal relationships between events recorded in the IT infrastructure. The purpose of such a system is to increase the efficiency of support engineers by identifying possible causes of events, thereby reasonably directing the search and investigation of events, incidents and problems. The paper shows the use of templates as a way of primary classification of events, without revealing the semantics / meaning of the event. Extracting templates from raw records is a preprocessing step. For the purpose of extracting templates, a template engine was developed that automates this process. The claimed system is implemented using open, free, well-established tools, such as the Rsyslog logging server and the developed application software - Tabler.

Keywords: IT-infrastructure, syslog, analysis, causation, monitoring, pattern, journaling, Data mining, data extraction, big data.

1 Введение

В настоящее время деятельность большинства предприятий полагается на средства вычислительной техники и автоматизации, то есть на информационные системы. Современные информационные системы, предоставляющие необходимые бизнесу ИТ-сервисы, отличаются большим уровнем сложности: они многокомпонентны, зачастую компоненты распределены в вычислительной сети и взаимодействуют друг с другом при помощи различных технологий и протоколов. Кроме того, работа одних систем, как правило, опирается на работу других систем, поэтому, несмотря на определённую автономность и модульность, информационные системы очень тесно связаны между собой, что делает поиск причин неисправностей, поиск решений инцидентов и проблем, а также другие задачи достаточно трудными, под силу лишь высококвалифицированным инженерам с обширным опытом.

Любому администратору ИТ-инфраструктуры в рамках своей деятельности приходится постоянно отвечать на 3 вопроса:

1. Что случилось в ИТ-инфраструктуре?
2. Почему это случилось?
3. Что в связи с этим следует предпринять?

В общем смысле ответом на первый вопрос будет «события».

Согласно словарю ITILv3 [1]:

- ИТ-инфраструктура - все аппаратное и программное обеспечение, сети, инженерное обеспечение и т.п., необходимые для разработки, тестирования, предоставления, мониторинга, контроля и поддержки ИТ-услуг.
- Термин ИТ-инфраструктура включает в себя все компоненты информационных технологий, но не включает связанные с ними персонал, процессы и документацию.
- Событие - изменение состояния, которое имеет значение для управления ИТ-услугой или другой конфигурационной единицей.
- Этот термин также используется для обозначения оповещения или уведомления, созданного любой ИТ-услугой, конфигурационной единицей или средством мониторинга.

События обычно требуют от персонала эксплуатации ИТ выполнения действий и часто приводят к регистрации инцидентов.

При этом следует отличать термин «инцидент» от термина «событие». Согласно словарю ITILv3 [1], инцидент – незапланированное прерывание или снижение качества ИТ-услуги. Сбой конфигурационной единицы, который

еще не повлиял на услугу, также является инцидентом, например, сбой одного диска из RAID-массива. Как следует из определения, термин «событие» является более общим по отношению к термину «инцидент».

Исходя из приведённого определения термина “событие”, для фиксации событий ИТ-инфраструктуры следует отслеживать:

- состояние аппаратных средств;
- состояние программных средств;
- состояние сети.

2 Оценка количества событий ИТ-инфраструктуры

Основным аппаратным компонентом любой вычислительной системы является вычислитель – процессор. Современные процессоры работают с тактовой частотой в примерном диапазоне от 400 МГц до 3800 МГц. При этом за один такт происходит изменение значений параметров, в частности значений регистров памяти процессора, т.е. происходит событие. Это значит, количество происходящих событий в процессоре оценивается значениями порядка триллионов в секунду (10^9 Гц). Для регистрации такого количества событий для каждого процессора ИТ-инфраструктуры необходим как минимум такой же по производительности вычислитель, что, очевидно, не представляется возможным на практике. Из этого следует невозможность 100%-ного отслеживания всех событий ИТ-инфраструктуры.

При этом следует обратить внимание на то, что в ИТ-инфраструктуре большая часть значимых событий так или иначе журналируется как самим оборудованием и информационными системами, так и системами мониторинга, предоставляя огромное количество информации для анализа. Однако форматы сообщений журналов разнятся не только от системы к системе, но и порой от версии к версии одной и той же системы, в связи с чем вполне справедливо можно утверждать, что в общем случае записи о событиях в системных журналах всей ИТ-инфраструктуры являются слабо структурированным текстом.

Оценим снизу количество журналируемых событий ИТ-инфраструктуры. Для этого вместо оценки количества событий в любых информационных системах будем считать максимальные производительности их баз данных, так как все запросы к ним журналируются. Также не будем учитывать рядовые компоненты ИТ-инфраструктуры, такие как рабочие станции пользователей, в оценку пусть входят лишь важные для бизнеса информационные системы и ключевые компоненты ИТ-инфраструктуры, за функционированием которых нужно следить.

В работе [2] выполнена оценка производительности популярных СУБД, согласно которой наименьший показанный результат составляет 54 операции в секунду (СУБД Oracle на тесте Workload A), его и примем за основу и обозначим $T_{СУБД}$.

Тогда формула оценки будет иметь вид:

$$P = T_{СУБД} \times N = 54 \times N \text{ Гц}$$

Где P - оценочная плотность потока событий для ИТ-инфраструктуры,

N - количество вычислительных узлов в ИТ-инфраструктуре.

Оценим N. Как упомянуто выше, в N входят ключевые для бизнеса системы, а также ключевые компоненты ИТ-инфраструктуры, а именно сетевая составляющая, т.е. управляемые коммутаторы, маршрутизаторы и инфраструктурные сервера: DHCP, DNS и сервер каталога (например на базе LDAP). Отсюда имеем:

$$N = N_{БИЗ} + N_{СЕТЬ} + N_{DHCP} + N_{DNS} + N_{DS}$$

Где $N_{БИЗ}$ - количество ключевых для бизнеса систем,

$N_{СЕТЬ}$ - количество управляемых сетевых устройств,

N_{DHCP} - количество серверов DHCP,

N_{DNS} - количество серверов DNS,

N_{DS} - количество серверов каталога.

В самой маленькой инфраструктуре каждый компонент будет представлен хотя бы в количестве 1, а потому получаем нижнюю оценку N = 5. Для средних и крупных ИТ-инфраструктур ключевые сетевые серверы обычно просто дублируют для обеспечения отказоустойчивости, что даёт:

$$N = N_{БИЗ} + N_{СЕТЬ} + 2 + 2 + 2 = 6 + N_{БИЗ} + N_{СЕТЬ}$$

То есть N растёт при увеличении количества бизнес-систем и размеров сети.

С учётом нижних оценок N получаем:

$$P = 54 \times N = 54 \times 5 = 270 \text{ Гц}$$

То есть в минимальной инфраструктуре журналируется около 270 событий в секунду.

Для целей оценивания будем считать, что события происходят (и журналируются) лишь в рабочее время, т.е. на протяжении 8 часов. Это даёт:

$$M_{ДЕНЬ} = P \times 8 \times 3600 = 270 \times 8 \times 3600 = 7\,776\,000 \text{ событий/день}$$

На это значение и будем ориентироваться.

3 Задачи

Обрабатывать вручную такое большое количество записей не представляется возможным. С другой стороны записи журналов слабо структурированы, что делает методы стандартного программирования малопригодными для организации процедуры анализа и принятия решения. В связи с этими фактами для получения ответа на вопрос “Почему это случилось?” перспективным видится применение методов Data Mining.

В соответствии с [3] Data Mining это полуавтоматический процесс обнаружения паттернов с помощью методов анализа данных в преимущественно очень больших многомерных базах данных. Схематично процедура применения методов Data Mining, использованная при разработке данной ИАС, показана на рисунке 1.

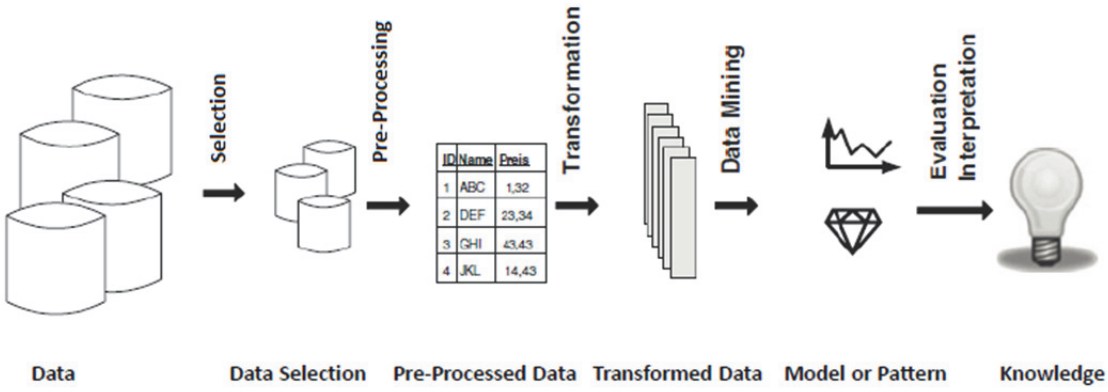

Рисунок 1 – Схема процесса Data Mining

Отбор данных (Selection) - определение контекста решаемой задачи, выбор источников данных и самих подходящих данных для анализа.

Предобработка (pre-processing) - над исходными “сырыми” данными выполняются операции по поиску и устранению различных проблем данных, ухудшающих применение методов Data Mining, например удаление дубликатов, обнаружение и удаление так называемых “выбросов” - сильных отклонений значений показателей от соседних окружающих, - дополнение отсутствующих данных, поиск и исправление некорректных значений в соответствиями с правилами предметной области.

Трансформация (transformation) - данные преобразуются в доходящую для анализа форму, например нормируются, строковые значения заменяются на числовые и т.п. Конечная форма определяется тем методом Data Mining-а, который планируется применять к подготовленным данным.

Data-Mining - применение выбранного метода к трансформированным данным, в результате чего получаем некую модель, которая описывает/характеризует данные или обнаруживает в них искомые паттерны, аномалии или структурные зависимости.

Оценка и интерпретация (Interpretation and Evaluation) - полученную модель пытаются интерпретировать, т.е. объяснить на основе её структуры или результатов тестирования, а также выполняется оценка её применимость/пригодность. Обнаруженные паттерны можно подредактировать или визуализировать для процедуры дальнейшего принятия решений.

После оценки следует реализовать информационно-аналитическую систему, заложив в её основу полученную аналитическую модель.

4 Источники данных и их сбор

Применительно к анализу системных журналов источниками данных являются информационные системы и подсистемы, а также сетевое оборудование. Записи системных журналов передаются на центральный сервер по протоколу Syslog [4] для централизованного хранения и обработки. При этом системы, отсылающие записи журналов по протоколу Syslog, добавляют к ним информацию об уровне важности события и название/идентификатор приложения, в которой событие произошло. Syslog-сервер в свою очередь доукомплектовывает записи информацией об источнике сообщения и временными метками, после чего сохраняет полученную структуру данных. Если настроить вывод таких структур в формат JSON [5], то они будут иметь следующий вид:


```
{
  "timegenerated-rfc-3339":"2016-01-09T03:36:03.637195+03:00"
  "timegenerated":"Jan 9 03:36:03"
  "timereported-rfc-3339":"2016-01-09T00:36:04+03:00"
  "timereported":"Jan 9 00:36:04"
  "hostname":"frontend1"
  "fromhost":"192.168.0.129"
  "fromhost-ip":"192.168.0.129"
  "syslogtag":"postfix\/smtpd[86372]:"
  "programname":"postfix"
```

```
  "procid":"86372"
  "app-name":"postfix"
  "pri-text":"mail.info"
  "pri":"22"
  "syslogfacility-text":"mail"
  "syslogfacility":"2"
  "syslogseverity-text":"info"
  "syslogseverity":"6"
  "uuid":"76DFD6B0820849F195D84E86640D76E9"
  "msgid":"-"
  "msg":" 2D3EE184B130_69055F4F: client=unknown[192.166.219.9]"
  "structured-data":"-"
}
```


Детальное описание всех полей можно найти в документации к Syslog-серверу [6].

5 Предобработка данных

При рассмотрении множества сообщений о событиях можно заметить, что сообщения об одних и тех же типах событий имеют схожую структуру, так как создатели подсистемы журналирования закладывают на этапе её создания набор шаблонов сообщений в самом программном коде. Такие шаблоны можно использовать как способ первичной классификации событий, без выявления семантики/смысла события. Но чтобы выполнять такую классификацию эти шаблоны из собранных записей системных журналов нужно извлечь. Извлечение шаблонов из «сырых» записей будет является этапом предобработки, в соответствии с рисунком 1.

Для целей извлечения шаблонов был разработан шаблонизатор, автоматизирующий этот процесс. С кодом шаблонизатора и некоторыми связанными артефактами можно ознакомиться на GitHub [7]. Цель шаблонизатора – из исходной последовательности сообщений выявить шаблоны построения этих сообщений. В дальнейшем набор шаблонов (структур сообщений) сообщений можно использовать для разбора новых сообщений, потому далее по тексту такой набор шаблонов ещё будет называться «ядро разбора».

Шаблонизатор основан на идее обучения с учителем: некоторый набор сообщений передаётся в шаблонизатор, шаблонизатор на основе метрики находит похожие строки, выделяет в них постоянные подстроки и изменяющиеся фрагменты и формирует свою версию шаблона, которую инженер-эксперт, как носитель экспертных знаний, может подредактировать. Затем данный шаблон добавляется в текущее ядро разбора и выполняется попытка разобрать все сообщения обучающего набора текущим ядром разбора. Все сообщения, которые удалось разобрать, предлагаются эксперту для оценки. После оценивания экспертом все сообщения, которые были отмечены как разобранные корректно, шаблонизируются. Данная процедура повторяется, пока не кончатся группы похожих сообщений, которые можно шаблонизировать. Схематично описанная процедура показана на рисунке 2, овалами изображены структуры данных, передаваемые от шага к шагу, шаги процедуры изображены прямоугольниками.

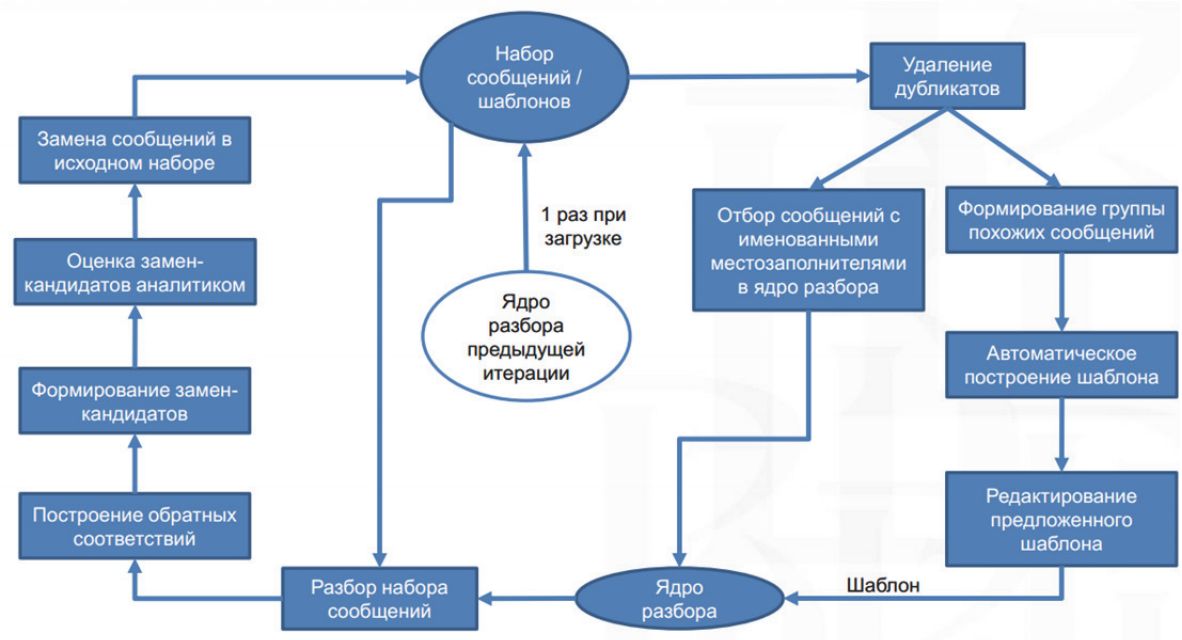

Рисунок 2 – Процесс построения ядра разбора.

Итогом описанной выше процедуры является получение ядра разбора для набора обучающих сообщений. Также предусмотрена процедура пополнения ядра разбора – процедура дообучения – за счёт добавления в обучающий набор новых сообщений. Схематично дообучение изображено на рисунке 3.

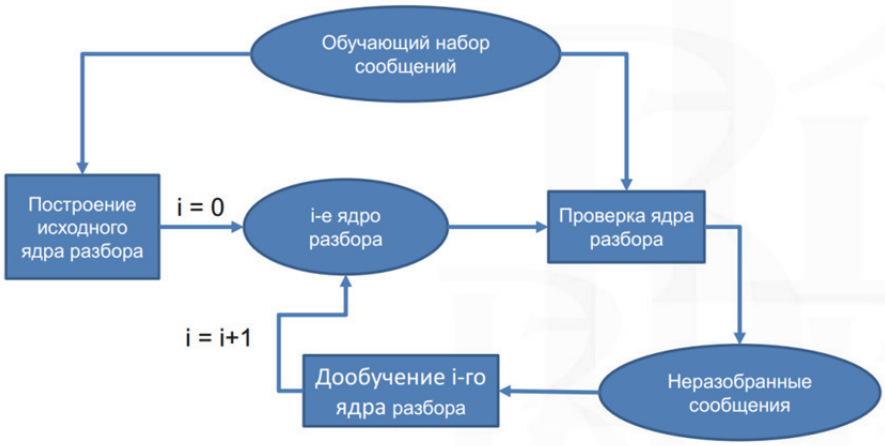

Рисунок 3 – Процесс обучения разборщика

Ключевые шаги получения ядра разбора в разработанном шаблонизаторе: обучающий набор сообщений, построение шаблона для группы похожих сообщений, текущее ядро разбора, оценка разбора экспертом - показаны на рисунках 4-7 соответственно.

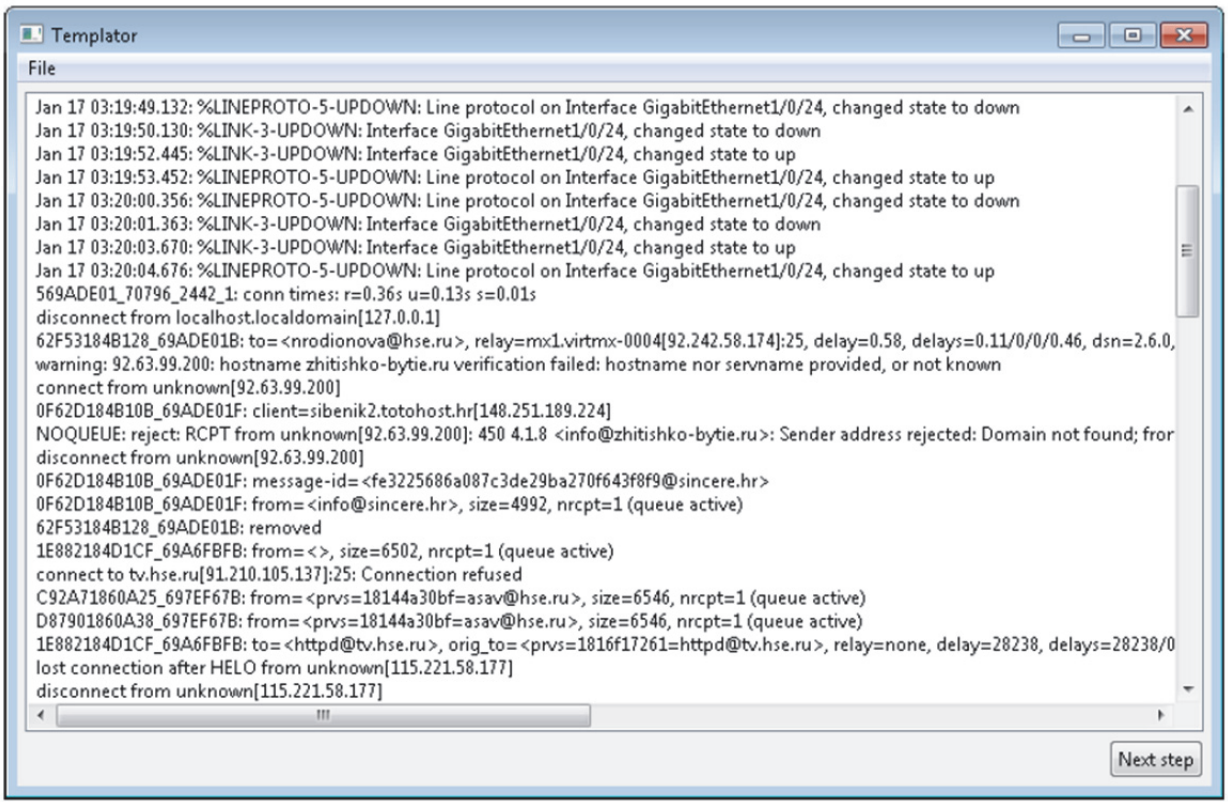

Рисунок 4 – Сообщения, загруженные в шаблонизатор.

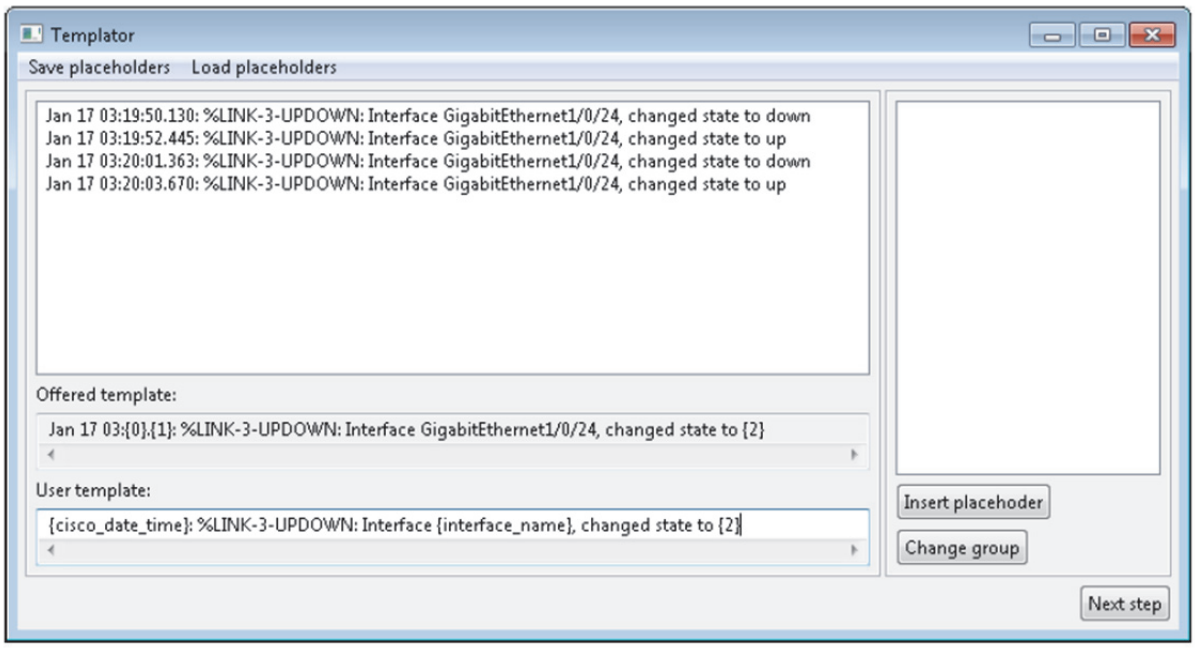

Рисунок 5 – Шаблон для группы похожих сообщений.

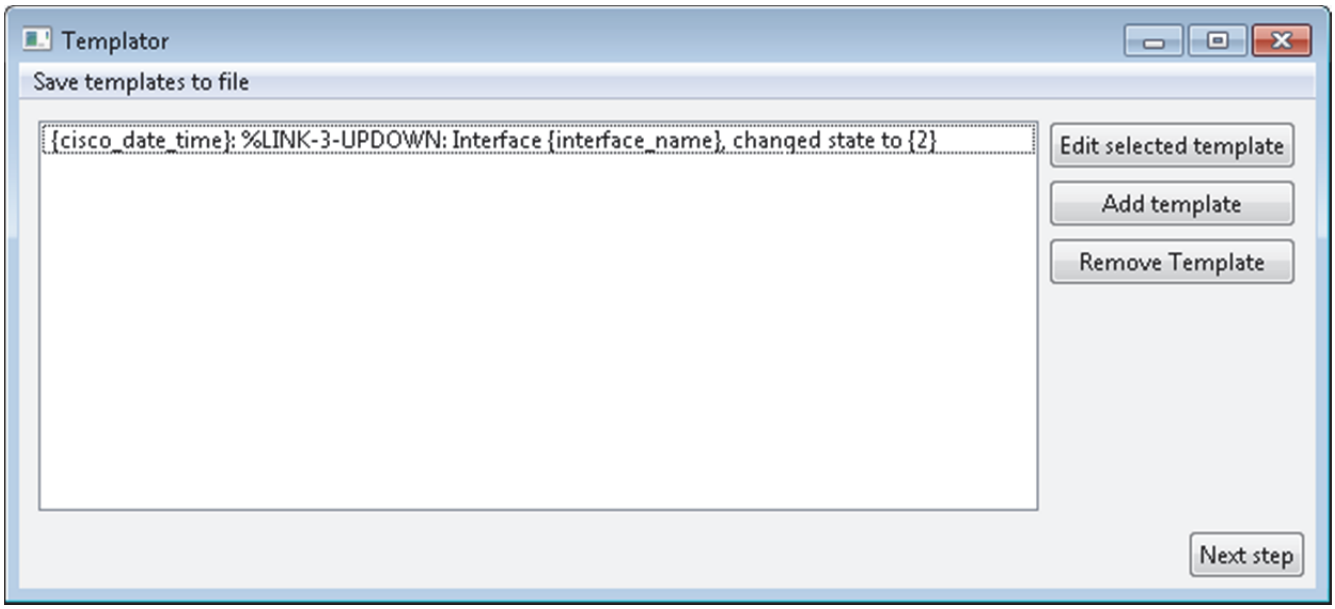

Рисунок 6 – Список шаблонов, используемых на текущей итерации.

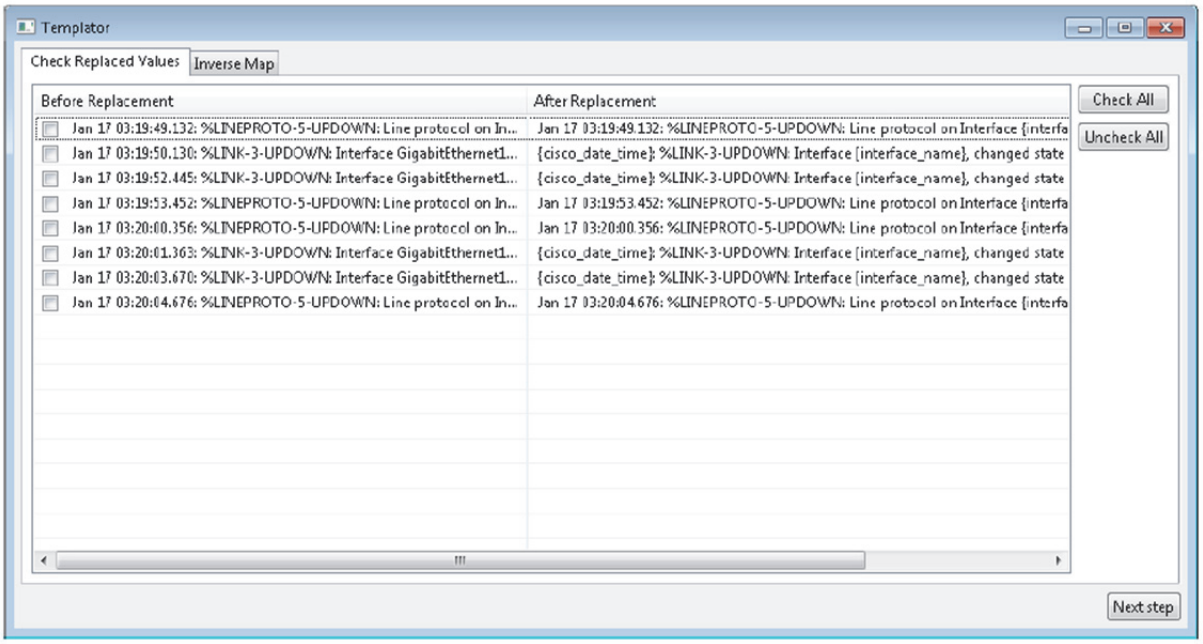

Рисунок 7 – Окно корректировки разбора.

Эффективность процедуры была оценена экспериментально. Для этого в качестве исходных данных был взят файл с записями событий в формате JSON, содержащий 218587 сообщений. Чтобы не перегружать инженера-эксперта в шаблонизатор подгружались лишь 100 первых сообщений из выборки и выполнялась процедура шаблонизации до тех пор пока новых шаблонов найти не удастся. Затем полученные шаблоны применялись к исходному набору записей, успешно разобранные записи удалялись из выборки, после чего данная процедура повторялась. Результаты 10 таких итераций показаны в таблице 1 и на графике (рисунок 6).

Таблица 1. Результаты итераций разбора

№	Шаблонов	Разобрано сообщений	Не разобрано сообщений
0	37	116072	52515
1	41	203295	14692
2	61	208797	9790
3	67	210632	7955
4	75	214916	3671
5	91	215299	3288
6	91	215759	2828
7	106	216116	2471
8	110	216152	2435
9	113	216166	2421
10	127	216234	2353

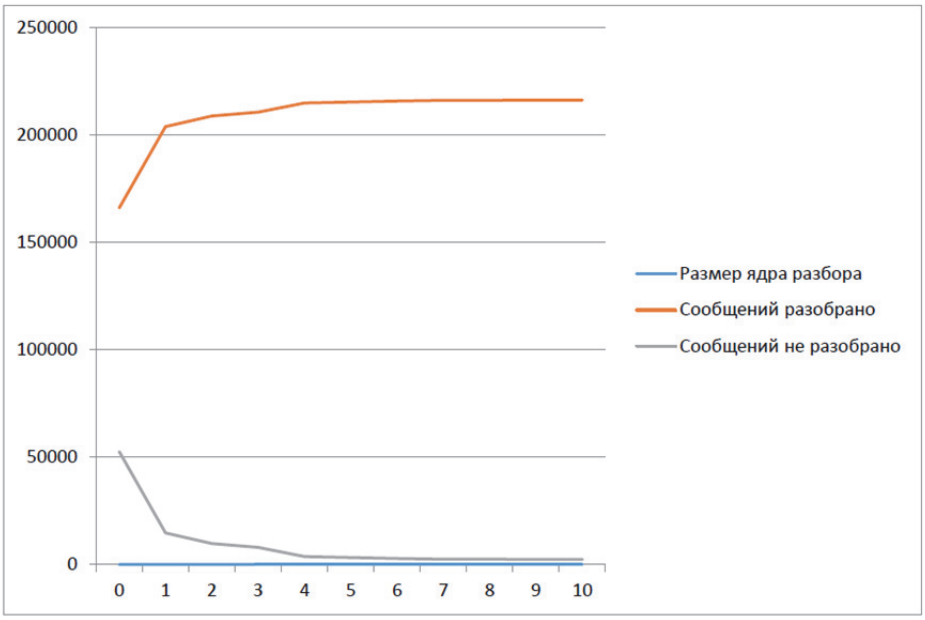

Рисунок 8 – Результаты итераций разбора.

Как следует из полученных данных, неразобранными остались около 1,08 % сообщений, что говорит о высокой эффективности описанной процедуры. Сами полученные шаблоны выглядят следующим образом:

```
conn times: r={conn_times_r}s u=0 s=0
connect from {hostname}[{ip_address}]
connect to {hostname}[{ip_address}]:25: Connection refused
connect to {hostname}[{ip_address}]:25: Operation timed out
disconnect from {hostname}[{ip_address}]
finished {cron_job_name}
improper command pipelining after HELO from {hostname}[{ip_address}]
improper command pipelining after MAIL from {hostname}[{ip_address}]
lost connection after AUTH from {hostname}[{ip_address}]
lost connection after CONNECT from {hostname}[{ip_address}]
lost connection after DATA ({size} bytes) from {hostname}[{ip_address}]
lost connection after DATA from {hostname}[{ip_address}]
lost connection after EHLO from {hostname}[{ip_address}]
lost connection after HELO from {hostname}[{ip_address}]
lost connection after MAIL from {hostname}[{ip_address}]
lost connection after RCPT from {hostname}[{ip_address}]
lost connection after RSET from {hostname}[{ip_address}]
{sophos_internal_id_2}: conn times: r={conn_times_r}s u={conn_times_u}s s=0
{sophos_internal_id_2}: conn times: r={conn_times_r}s u={conn_times_u}s s={conn_times_s}s
{sophos_internal_id_2}: delayed: Suspect Spam
{sophos_internal_id_2}: discarded
{sophos_internal_id_2}: msg times: r={msg_times_r}s u={msg_times_u}s s=0
{sophos_internal_id_2}: msg times: r={msg_times_r}s u={msg_times_u}s s={msg_times_s}s
{sophos_internal_id_2}: pmx_spam: loading new data version {sophos_data_version}
{sophos_internal_id}: client={hostname}[{ip_address}]
```

6 Заключение

На текущий момент решены задачи сбора и предобработки данных для последующей процедуры построения аналитической модели методами Data Mining.

Предстоит выполнить построение аналитической модели выявления причинно-следственных связей между событиями, предварительно выполнив подготовительные преобразования (transformation) данных, выполнить оценку модели, и реализовать оценённую и принятую модель как ИАС на зарекомендовавших себя технологиях работы с Big Data: Apache Hadoop, Apache Lucene и Apache Spark.

Создаваемая информационно-аналитическая система повысит диагностические способности инженеров ИТ-инфраструктуры, сократит время на сбор и анализ информации о событиях, а также позволит снизить требования к квалификации инженеров.

Список использованной литературы

[1] Словарь терминов ITIL® на русском языке, версия 2.0, 29 июля 2011 г. на основе английской версии 1.0, 29 июля 2011.

[2] Performance comparison of the most popular relational and non-relational database management systems. Kamil Kolonko [Электронный ресурс]/ Master of Science in Software Engineering / Faculty of Computing Blekinge Institute of Technology / SE-371 79 Karlskrona Sweden / URL: http://www.diva-portal.org/smash/get/diva2:1199667/FULLTEXT02 – (Дата обращения: 10.07.2019)

[3] Business Intelligence and Business Analytics Systems Prof. Dr. Christoph Sandbrink / University of Mannheim / Master Programs (Course No. IS 602)

[4] RFC 5424 The Syslog Protocol [Электронный ресурс] / Internet Engineering Task Force, March 2009. URL: https://tools.ietf.org/html/rfc5424 – (Дата обращения: 10.07.2019)

[5] RFC 7159 The JavaScript Object Notation (JSON) Data Interchange Format [Электронный ресурс] / Internet Engineering Task Force, March 2014. URL: https://tools.ietf.org/html/rfc7159 – (Дата обращения: 10.07.2019)

[6] rsyslog Properties / Adiscon GmbH / URL: https://www.rsyslog.com/doc/master/configuration/properties.html – (Дата обращения: 10.07.2019)

[7] GitHub – aaovchinnikov/LogAnalizer Log template and cause-effect revealing software [Электронный ресурс] / URL: https://github.com/aaovchinnikov/LogAnalizer – (Дата обращения: 10.07.2019)