Применение технологии Process mining для выявления аномальных ситуаций в работе наукоемкого оборудования

А.М. Хасанова, М.Е. Дунаев

Аннотация **- В современном мире все крупные компании используют IT инфраструктуру для организации своей деятельности. И задача выявления и устранения различных аномальных событий (включая угрозы безопасности) в деятельности технологических платформ становится крайне актуальной.**

Такие платформы становятся основной частью IT-индустрии, поддерживая широкий спектр онлайн-сервисов (таких как поисковые системы, социальные сети, различные типы ассистентов) и интеллектуальных приложений (прогноз погоды, биомедицинская инженерия и т. д.). Большинство подобных систем обеспечивают работу сложного оборудования в различных отраслях: горнодобывающей промышленности, атомной индустрии при проектировании и эксплуатации АЭС, транспортной сфере городов и т.д. Поскольку почти все эти системы предназначены для круглосуточной работы, обслуживая тысячи компьютеров одновременно, высокая их доступность и надежность становятся обязательными.

Любые инциденты с подобными системами, включая перебои или снижения качества в обслуживании, приведут к выходу из строя отдельных приложений и, соответственно к финансовым издержкам. Помимо этого, нарушения работоспособности цифрового оборудования может привести к авариям и несчастным случаям на производстве.

Одним из инструментов решения вышеупомянутых проблем является Process mining, позволяющий анализировать конкретные процессы, обнаружить аномальные события, прогнозировать узкие места и т.п.

Целью настоящей работы является исследование и реализация эффективных технологий интеллектуального анализа процессов (Process mining) для выявления аномальных ситуаций в работе наукоемкого цифрового оборудования по его журналам событий (на примере ОС Windows).

Ключевые слова— безопасность, журналы событий, Process mining, ОС Windows, аномальные ситуации

I. Введение

В настоящее время практически каждая крупная компания использует IT инфраструктуру для организации своей деятельности. Рост современных компаний достигается путем масштабирования распределенных систем, в которых могут участвовать тысячи разных видов компьютеров и сотни одновременно работающих приложений. Очевидно, при таком[i] значительном количестве используемых вычислительных ресурсов компании сталкиваются с угрозами безопасности своей инфраструктуры, поэтому крайне актуальна задача выявления и устранения различных аномалий (включая угрозы безопасности) в деятельности технологических платформ.

Любые инциденты с подобными системами, включая перебои или снижения качества в обслуживании, приведут к выходу из строя отдельных приложений и, соответственно к финансовым издержкам. Помимо этого, нарушения работоспособности цифрового оборудования может привести к авариям и несчастным случаям на производстве.

Эффективным подходом для обнаружения неисправностей функционирования и/или атак на информационные ресурсы компьютерных систем является интеллектуальный анализ выполняемых процессов (Process mining) по информации журналов событий каждой из подсистем.

Целью данной работы является исследование и реализация эффективных технологий интеллектуального анализа процессов для выявления аномальных ситуаций в работе наукоемкого цифрового оборудования по его журналам событий (на примере процессов ОС Windows).

II. Получение и обработка журналов событий ОС Windows

Безопасность - одна из самых значимых и актуальных проблем любой компании. Журналы событий ОС Windows - очень полезный источник данных для получения информации о безопасности, но иногда их почти невозможно использовать из-за сложности представления данных журнала или из-за очень большого количества событий, генерируемых за минуту. В связи с этим анализ журнала событий операционной системы оказывается очень трудоемкой, практически неподъемной задачей при ручной обработке и анализе. Поэтому необходимо разрабатывать различные компьютерные системы, анализирующие процессы, идущие на платформах, умном оборудовании и прикладных системах (Process mining) по информации журналов событий этих программ и устройств.

Журнал событий ОС Windows – это набор специальных лог-файлов, в которые ОС и выполняемые приложения записывают все значимые для ОС события, такие как установка нового устройства; ошибки в работе приложений; вход пользователей в систему; незапустившиеся службы и т.д.

Журналы событий ОС Windows включает три основные и две дополнительные категории событий: основные – это Приложение, Система, Безопасность; дополнительные – Установка и Перенаправленные события [1]. Рассмотрим их назначение.

- Приложение – хранит важные события, связанные с конкретным приложением. Эти данные помогут системному администратору установить причину отказа той или иной программы.

- Система – хранит события операционной системы или ее компонентов (например, неудачи при запусках служб или инициализации драйверов; общесистемные сообщения и прочие сообщения, относящиеся к системе в целом).

- Безопасность – хранит события, связанные с безопасностью (такие как: вход/выход из системы, управление учётными записями, изменение разрешений и прав доступа к файлам и папкам).

Для получения полного журнала событий из всех источников ОС Windows в данной работе была использована утилита Sysmon [2]. С помощью power-shell можно получить все события, которые агрегирует Sysmon в формате XML:

Get-WinEvent -LogName "Microsoft-Windows-Sysmon/Operational" | Export-Clixml sysmon_logs.xml"

Таким образом с помощью Sysmon и power-shell для работы был получен фрагмент полных журналов событий ОС Windows (54958 событий).

Каждое событие имеет свой набор полей. Основными полями для всех событий являются следующие (см. рис. 1).

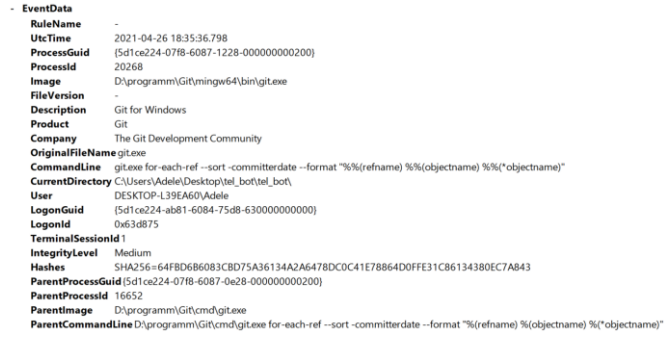

Рис. 1 - Структура событий Sysmon.

UtcTime – дата и время события.

ProcessGuid – GUID события, чтобы обеспечить корреляцию событий, даже когда Windows повторно использует идентификаторы процессов (формат xxxxxxxx-xxxx-Mxxx-Nxxx-xxxxxxxxxxxx).

ProcessId – ID события, чтобы обеспечить корреляцию событий, даже когда Windows повторно использует идентификаторы процессов (формат: xxxxx в десятичном формате).

Image – Путь к файлу порождаемого / создаваемого процесса.

ParentProcessGuid – GUID родительского события

ParentProcessId – ID родительского события

ParentImage – Путь к файлу родительского процесса.

III. Интеллектуальный анализ журналов событий

Согласно манифесту [3] Process mining, журналы событий могут быть использованы для осуществления трех форм Process mining:

- Извлечение - метод обнаружения использует журнал событий и создает модель, которая описывает поведение системы, записанное в данном журнале.

- Соответствие - здесь существующая модель процесса сравнивается с журналом событий того же процесса. Проверка соответствия может использоваться для проверки того, соответствует ли реальность, записанная в журнале, модели и наоборот.

- Усовершенствование - На данном этапе происходит расширение или улучшение существующей модель процесса, используя информацию о фактическом процессе, записанную в некотором журнале событий. В отличии от второго типа, который измеряет соответствие между моделью и реальностью, третий тип анализа процессов направлен на изменение или расширение априорной модели.

Таким образом по журналам событий можно построить и визуализировать модели, которые будут отражать поведение системы, по которой были получены эти журналы событий.

Для полноценного анализа безопасности ОС Windows с помощью интеллектуального анализа журналов событий необходимо выполнить следующие этапы:

- сбор журналов с помощью Sysmon и экспорт всех событий в XML формат;

- преобразование журнала событий из формата XML в CSV;

- обработка и подготовка журналов;

- извлечение признаков и необходимых данных с помощью алгоритмов обнаружения процессов (Alpha-алгоритм, индуктивный и эвристический алгоритмы), выбрать наиболее подходящий алгоритм и формализовать его с помощью сетей Петри;

- обнаружение аномалий или вредоносных событий.

A. *Преобразование журнала событий из формата XML в CSV*

Для применения алгоритмов интеллектуального анализа данных логи ОС Windows, получаемые в формате XML, необходимо преобразовать в формат CSV..

В настоящей работе была разработана функция, которая парсит (разбирает, англ. parse) XML файл с помощью библиотеки lxml, преобразовывает данные в DataFrame и сохраняет их в CSV формат с помощью библиотеки Pandas.

С помощью цикла осуществляется проход по всему файлу и по тэгам в XML файле выделяются необходимые

поля. Количество и наименование полей для каждого события может отличаться в зависимости от типа данного события. Всего в Sysmon существует 23 типа событий (идентификатор типа события хранится в поле EventID), и каждый тип событий обладает своим набором параметров.

После преобразования журнала событий в формат CSV, он выглядит так, как изображено на рис. 2.

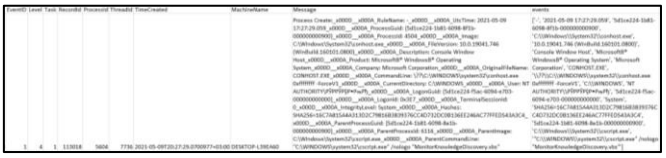

Рис. 2. Преобразованный журнал событий в CSV формат.

В свойствах и сообщении события содержится также необходимая информация о произошедших событиях: дата и время создания события (UtcTime), ID события (ProcessGuid и ProcessId), путь к файлу, который вызвал событие (Image), а также сведения о родительском событии (ParentProcessGuid, ParentProcessId, ParentImage), имя пользователя (User).

Эти параметры были получены из свойств и сообщения события (рис. 3) с помощью регулярных выражения с использованием библиотеки re.

EventID	Level	Task	RecordId	ProcessId	ThreadId	TimeCreated	MachineName	UtcTime	CommandLine
1	4	1	127887	16640	5320	2021-05-29T12:49:31.669714+03:00	DESKTOP-BI7J1U4	2021-05-29 09:49:31.662	"C:\Program Files (x86)\CCleaner Browser\Appli...
1	4	1	127886	7908	5320	2021-05-29T12:49:31.656054+03:00	DESKTOP-BI7J1U4	2021-05-29 09:49:31.646	"C:\Program Files (x86)\CCleaner Browser\Appli...
1	4	1	127885	17572	5320	2021-05-29T12:49:31.641474+03:00	DESKTOP-BI7J1U4	2021-05-29 09:49:31.635	"C:\Program Files (x86)\CCleaner Browser\Appli...
1	4	1	127884	16864	5320	2021-05-29T12:49:31.631366+03:00	DESKTOP-BI7J1U4	2021-05-29 09:49:31.622	"C:\Program Files (x86)\CCleaner Browser\Appli...

ProcessGuid	Image	User	ParentProcessGuid:	ParentProcessId:	ParentImage:
{e3ab61c8-0e2b-60b2-70bd-000000000900}	C:\Program Files (x86)\CCleaner Browser\Applic...	DESKTOP-BI7J1U4\adela	{e3ab61c8-0e2b-60b2-730d-000000000900}	10696	C:\Program Files (x86)\CCleaner Browser\Applic...
{e3ab61c8-0e2b-60b2-7e0d-000000000900}	C:\Program Files (x86)\CCleaner Browser\Applic...	DESKTOP-BI7J1U4\adela	{e3ab61c8-0e2b-60b2-730d-000000000900}	10696	C:\Program Files (x86)\CCleaner Browser\Applic...
{e3ab61c8-0e2b-60b2-7d0d-000000000900}	C:\Program Files (x86)\CCleaner Browser\Applic...	DESKTOP-BI7J1U4\adela	{e3ab61c8-0e2b-60b2-730d-000000000900}	10696	C:\Program Files (x86)\CCleaner Browser\Applic...
{e3ab61c8-0e2b-60b2-7c0d-000000000900}	C:\Program Files (x86)\CCleaner Browser\Applic...	DESKTOP-BI7J1U4\adela	{e3ab61c8-0e2b-60b2-730d-000000000900}	10696	C:\Program Files (x86)\CCleaner Browser\Applic...

Рис. 3 - Преобразованные логи событий ОС Windows.

B. Обработка и подготовка журналов событий

После получения и преобразования журналов событий из ОС Windows 10 необходимо преобразовать журнал событий по умолчанию в форму, удобную для использования алгоритмов Process mining с помощью библиотеки pm4py [7].

Минимальные требования для преобразования журнала событий в необходимый формат можно посмотреть, например, в Руководстве пользователя Disco [4]. В журнале событий должны быть как минимум три элемента для обеспечения анализа интеллектуального анализа процессов:

- отметка времени (Timestamp) – время создания процесса;
- идентификатор случая (Case ID) – уникальное Id события, которое объединяет в себе все процессы, относящие к данному событию;
- действие (Activity) – данные о самом событии, пусть к файлу, который запустил данный процесс.

Эти три элемента позволяют взглянуть на данные с точки зрения процесса. Также дополнительно можно использовать другие элементы, такие как ресурсы, состояние, приоритет и т. д.

Исходя из полученного журнала событий, необходимо преобразовать существующие колонки для последующего применения алгоритмов интеллектуального анализа в соответствии с Таблицей 1.

Таблица 1. Переименование параметров (колонок) журнала событий

Название параметра для применения алгоритмов Process mining с помощью библиотеки Pm4py	Название параметра для использования журнала событий в программе Disco	Текущее название параметра
case:concept:name	Case ID	Case_id
concept:name	Activity	Image
time:timestamp	Timestamp	UtcTime

Журналы событий, полученные из ОС Windows, не имеют иерархии, что не позволяет сгруппировать отдельные процесс в одно событие (Case ID). Для дальнейшего анализа и построение сетей Петри, необходимо восстановить иерархию и создать цепочки процессов для каждого события. Для этого был разработан алгоритм на языке Python.

Таким образом, после применения алгоритмов, разработанных на языке Python с использованием библиотек Pandas, re, lmxl, pm4py были получены преобразованные и обработанные журналы событий ОС Windows для применения алгоритмов Process mining (рис. 4).

case:concept:name	concept:name	time:timestamp	ProcessId	ParentProcessId	ParentImage
0	1 C:\Users\Adele\AppData\Roaming\Zoom\bin\CptHos...	2021-05-07 09:57:44.216	14648	7568	C:\Users\Adele\AppData\Roaming\Zoom\bin\Zoom.exe
1	1 C:\Users\Adele\AppData\Roaming\Zoom\bin\Zoom.exe	2021-05-07 09:57:38.545	7568	10500	C:\Users\Adele\AppData\Roaming\Zoom\bin\Zoom.exe
2	1 C:\Users\Adele\AppData\Roaming\Zoom\bin\Zoom.exe	2021-05-07 09:57:37.097	10500	11136	C:\Program Files (x86)\Google\Chrome\Applicati...
3	1 C:\Program Files (x86)\Google\Chrome\Applicati...	2021-05-07 08:09:57.806	11136	5700	C:\Windows\explorer.exe
4	1 C:\Windows\explorer.exe	2021-05-07 08:09:25.552	5700	5440	C:\Windows\System32\userinit.exe
5	1 C:\Windows\System32\userinit.exe	2021-05-07 08:09:25.071	5440	780	C:\Windows\System32\winlogon.exe
6	1 C:\Windows\System32\winlogon.exe	2021-05-07 08:09:16.637	780	700	C:\Windows\System32\smss.exe

Рис. 4 - Преобразованные журналы событий ОС Windows.

C. Построение модели процессов с помощью алгоритмов Process mining

Извлечение процессов – второй тип Process mining, это собирательное название различных методов, предназначенных для синтеза, анализа и усовершенствования моделей процессов в результате работы с журналами событий информационных (и других) систем. Извлечение процессов – одна из наиболее важных частей интеллектуального анализа данных, так как после этого этапа, построенная модель будет отражать поведение системы.

На сегодняшний день разработано большое количество алгоритмов, предназначенных для извлечения процессов из журналов событий. Среди алгоритмов извлечения обычно выделяют три основные, задающие целые семейства подходов [9,10,11]:

1) Alpha-алгоритм и Alpha+ алгоритм;
2) эвристические подходы;
3) индуктивные алгоритмы.

Вторая форма Process mining – это проверка соответствия. На данном этапе проводится сопоставление существующей модели процесса с журналом событий этого же процесса и оценка полученной модели. В библиотеке Pm4py для построения алгоритмов Process mining представлены все вышеописанные алгоритмы, также библиотека предоставляет методы оценки полученной модели с помощью различных метрик [9,10,11]:

- соответствие модели журналам событий (Fitness);
- точность (Precision);

- обобщенность (Generalization).

Каждый из алгоритмов был протестирован на журналах событий ОС Windows и проведен сравнительный анализ данных алгоритмов по различным метрикам. Результат сравнения алгоритмов представлен в Таблице 2.

Таблица 2. Сравнение алгоритмов обнаружения процессов

Алгоритм	Соответствие модели журналам событий (Fitness)	Точность (Precision)	Обобщенность (Generalization)
Alpha Miner	0.491	0.197	0.422
Heuristic Miner	0.765	0.521	0.487
Inductive Miner	0.96	0.709	0.957

Исходя из полученных результатов сравнения, наилучшим алгоритмом для обнаружения процессов и построения модели, оказался индуктивный алгоритм, на основе которого модель была визуализирована с помощью DFG графа и сети Петри. Для удобства формализации полученной модели с помощью сетей Петри в качестве примера были взяты два события (рис. 3 и 4): запуск Zoom конференции и запуск Telegram Desktop.

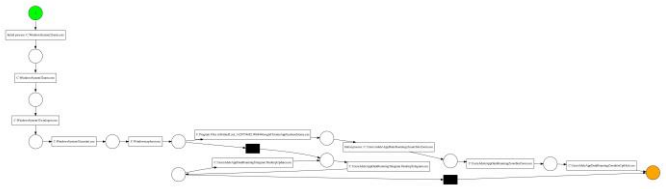

Рис. 3 - Пример сетей Петри для двух событий.

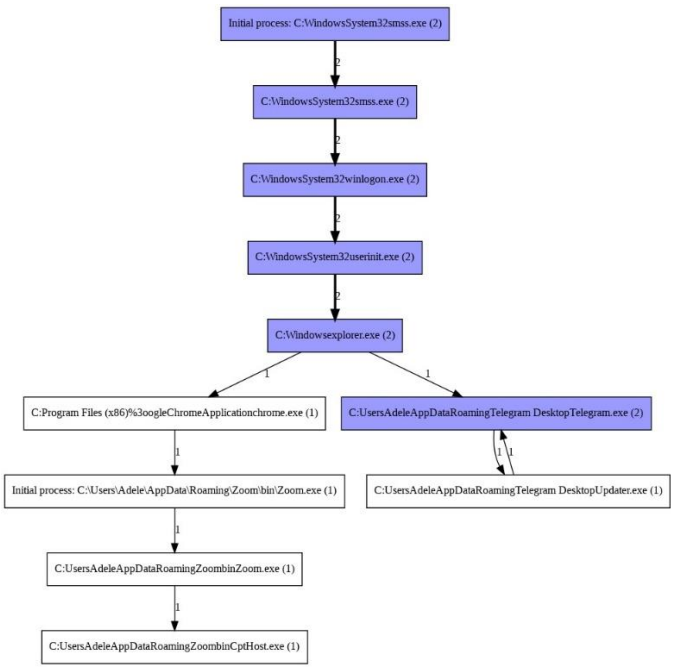

Рис. 4 - Пример DFG графа (Directly-Follows Graph) для двух событий.

Для дальнейшего поиска аномалий или нарушений ОС Windows с помощью алгоритмов Process mining используется построенная модель для журналов событий ОС Windows.

IV. Поиск аномальных и вредоносных событий

Каждый час в системе генерируется порядка 1000 событий. Для дальнейшего анализа новых событий в системе, необходимо получить новые события системы. Для их получения была разработана функция на языке Python с использованием библиотек subprocess, sys.

- Модуль subprocess дает возможность запускать процессы программ из Python. Используя данный модуль, можно запускать скрипт powershell, который будет возвращать событий ОС Windows за последний час в формате XML:

Get-WinEvent -FilterHashtable @{LogName = "Microsoft-Windows-Sysmon/Operational"; StartTime = ((Get-Date).AddHours(-1))}

- Модуль sys обеспечивает доступ к некоторым переменным и функциям, взаимодействующим с интерпретатором Python.

Получаемые новые события необходимо по описанной выше схеме преобразовать, структурировать и восстановить их иерархию для дальнейшего анализа.

A. Пример инцидента

Для проверки разработанных алгоритмов была проведена симуляция вредоносного события в ОС Windows.

Пользователь ОС Windows в результате фишинговой атаки открывает вложение к письму — документ Microsoft Excel с завлекающим названием (например, «Данные по зарплате за май.xlsx»). Пользователь скачивает данный файл, который содержит вредоносный макрос. При открытии пользователь фактически дает разрешение на запуск вредоносного макроса, который выполняет следующую последовательность действий.

1. Подключение к серверу управления атакующего и скачивание файла viewpage.php, содержащего полезную нагрузку — meterpter reverse shell;
2. Переименование загруженного файла с сохранением его в каталоге %TEMP% под именем sysprov32.dll;
3. Прописывание в ключ реестра HKCU\Software\Microsoft\Windows\CurrentVersion\Run значения userprep со следующим содержимым: rundll32 C:\Users\Adele\AppData\Local\Temp\sysprov32.dll;
4. Запуск полезной нагрузки, которая содержится в файле sysprov32.dll, командой rundll32 C:\Users\vadmin\AppData\Local\Temp\sysprov32.dll;
5. Прописывание полезной нагрузки в ключ реестра;
6. Выполнение дампа учетных данных и кража учетных данных.

B. Обнаружение инцидента

На первом этапе работы вредоносного макроса-скрипта из файла MS EXCEL атакующий создает сетевое подключение к удаленному серверу. Такие события в Sysmon будут иметь Event ID = 3.

Используя данные из Threat Intelligence, можно получить IP-адреса сервера, которые считаются вредоносными. Согласно данным Threat Intelligence IP-адрес 31.179.135.186 используется вредоносным программным обеспечением. Таким образом при

открытии вредоносного файла в операционной системе появляется скомпрометированный хост или группа хостов, которые осуществляли или продолжают осуществлять подключения к вредоносному серверу управления с IP-адресом 31.179.135.186.

По специфичному event_id для сетевых подключений и IP-адрес на начальном этапе можно обнаружить аномальное событие.

Так как атака начинается с открытия MS Excel, то мы в журнале событий увидим новый процесс, связанный с открытием программы:

C:\Program Files (x86) \ Microsoft Office \ Office16 \ excel.exe

Даже если бы в базе IT-платформы не оказалось IP-адреса атакующего, мы бы заметили, что офисное приложение подключается к внешнему IP.

Атакующие зачастую используют вредоносные макросы в составе документов в качестве легковесного кода, предназначенного для доставки основной полезной нагрузки с сервера управления. В таком случае процесс офисного приложения может сохранить в файловой системе файл с полезной нагрузкой для его последующего запуска. Если атакующий не замаскировал расширение файла под более безобидное, то можно использовать событие FileCreate Sysmon (Event ID = 1) для обнаружения подобной активности.

Результаты запроса показывают, что на хосте DESKTOP-L39EA60 процессом C:\Program Files (x86)\Microsoft Office\Office16\excel.exe был создан исполняемый файл C:\Users\vadmin\AppData\Local\Temp\sysprov32.dll (рис. 5).

EventID	UtcTime	Image	ProcessId	ParentProcessId	ParentImage:
13	23.05.2021 17:54	HKU\S-1-5-21-3921924719-2751751025-4067464375-1003\Software\Microsoft\Windows\CurrentVersion\Run	1564	5048	C:\Users\vadmin\AppData\Local\Temp\sysprov32.dll
1	23.05.2021 17:52	C:\Users\vadmin\AppData\Local\Temp\sysprov32.dll	5048	10380	C:\Windows\SysWOW64\rundll32.exe
3	23.05.2021 17:52	C:\Windows\SysWOW64\rundll32.exe	10380	7308	C:\Windows\System32\cmd.exe
1	23.05.2021 17:51	C:\Windows\System32\cmd.exe	7308	13700	C:\Program Files (x86)\Microsoft Office\Office16\EXCEL.EXE
3	23.05.2021 17:49	C:\Program Files (x86)\Microsoft Office\Office16\EXCEL.EXE	13700	13712	C:\Program Files (x86)\Microsoft Office\Office16\EXCEL.EXE
1	23.05.2021 17:48	C:\Program Files (x86)\Microsoft Office\Office16\EXCEL.EXE	13712		

Рис. 5 – Пример вредоносного события.

Т. е. пользователь открыл документ и разрешил выполнение макроса. В свою очередь макрос загрузил с подконтрольного атакующему командного центра исполняемый файл с основной полезной нагрузкой (DLL-библиотека Reverse Shell-а) и сохранил его в каталоге временных файлов под именем sysprov32.dll. При этом адрес командного центра, с которого была выполнена загрузка файла, на момент инцидента уже был известен сообществу специалистов по кибербезопасности и фигурировал во множестве источников Threat Intelligence как вредоносный.

После того как файл с полезной нагрузкой скачен, он будет запущен через командную строку. Здесь как раз будет важным знание об иерархии процессов «родительский процесс — дочерний процесс», а с помощью построенной модели с использованием алгоритмов Process mining есть знание о том, какие пары «родительский процесс — дочерний процесс» являются для ОС Windows нормальными. Запуск процессом офисного приложения командного интерпретатора cmd — аномальное событие. Оно может свидетельствовать об исполнении вредоносного кода, встроенного в документ, например, макроса или DDE.

Из журнала событий видно, что на хосте DESKTOP-L39EA60 процессом C:\Program Files (x86)\Microsoft Office\Office16\excel.exe был запущен командный интерпретатор cmd с командной строкой C:\Windows\System32\cmd.exe /c rundll32 C:\Users\vadmin\AppData\Local\Temp\sysprov32.dll

Т. е. для исполнения полезной нагрузки, содержащейся в файле sysprov32.dll, атакующий использовал программу rundll32.

Rundll32.exe — легитимная программа ОС Windows, которая позволяет запускать код из произвольной dll-библиотеки. С помощью rundll32 злоумышленники могут запускать исполнение вредоносного кода.

Когда пользователь открыл вредоносный документ, атакующий получил удаленный доступ к компьютеру пользователя. Однако такой доступ не вечен и будет потерян при первой перезагрузке системы.

Для сохранения доступа после перезагрузки атакующие используют различные техники закрепления в системе (Persistence). Одна из таких техник — прописывание полезной нагрузки в ключи реестра, отвечающие за автозагрузку и выполнение кода при входе пользователя в систему.

В случае данного инцидента полезная нагрузка была сохранена в ключе реестра HKCU: \SOFTWARE\Microsoft\Windows\CurrentVersion\Run. Прописывание офисными приложениями каких-либо значений в ключи реестра, отвечающие за автозагрузку, является аномальным и может свидетельствовать о попытках вредоносного кода закрепиться в системе.

Из журнала событий видно, что процессом C:\Program Files (x86)\ Microsoft Office \Office16\excel.exe в ключе реестра HKU\ S-1-5-21-3921924719-2751751025-4067464375-1003\ Software\ Microsoft\ Windows\ CurrentVersion\ Run было создано значение userprep с содержимым rundll32 C:\Users\vadmin\AppData\Local\Temp\sysprov32.dll.

Для обнаружения подобной активности может быть использовано как событие Sysmon Event ID = 13, так и аудит событий безопасности ОС Windows (Event ID = 4657). Чтобы событие ОС Windows начало генерироваться, предварительно необходимо установить SACL (System Access Control List) на соответствующие ключи реестра.

Для детектирования данного инцидента были использованы следующие поведенческие признаки, на базе которых были разработаны соответствующие правила:

- обращение процесса приложения Microsoft Office к адресу, который фигурирует в используемых источниках Threat Intelligence как вредоносный;
- взаимодействие процесса приложения Microsoft Office с внешними адресами;
- создание процессом приложения Microsoft Office файла с исполняемым расширением;
- прописывание полезной нагрузки в ключ реестра.

Все перечисленные события и процессы были обнаружены при анализе журнала событий ОС Windows и сделан вывод о том, что совершены вредоносные действия.

V. Заключение

Данная работа посвящена применению технологии Process mining для выявления аномальных ситуаций в

работе наукоемкого оборудования по журналам событий. В качестве примера была использована ОС Windows, как одна из наиболее популярных современных операционных систем, имеющая много журналов событий.

В ходе выполнения работы были получены следующие основные результаты:

1. На первом этапе работы было проведено исследование журналов событий ОС Windows: формат журналов; структура; информация, которая храниться в данных; типы событий (аудит безопасности, сетевое подключение, установка и другие). Исследование показало, что:

- в ОС Windows существуют разные типы журналов событий: безопасность, система, приложения. Эти журналы хранятся в разных местах и файлах;
- размер журналов событий и скорость их нарастания требует автоматической обработки логов;
- журналы событий хранятся в системе в слабоструктурированном виде;
- в журнале событий все процессы хранятся в виде отдельных записей, т.е. отсутствует иерархия процессов.

2. Учитывая особенности хранения информации в журналах событий ОС Windows, авторами был разработан алгоритм для предобработки и подготовки журналов событий, содержащий 4 этапа.

3. Для построения модели, которая характеризует нормальное поведение ОС Windows были исследованы такие алгоритмы Process mining, как Alpha-алгоритм, эвристический алгоритм и индуктивный алгоритм.

4. Каждая построенная модель была оценена с помощью трех метрик. По полученным оценкам лучшим оказался индуктивный алгоритм.

5. Для выявления аномальных или вредоносных событий в ОС Windows был реализован алгоритм получения новых событий ОС в нужном формате, который сравнивает получаемые события с ранее построенной моделью нормального ее поведения.

6. При тестировании разработанных алгоритмов было проведено моделирование вредоносного события и его обнаружение по логам системы.

Использование результатов данной работы для выявления аномальных или вредоносных событий в работе наукоемкого оборудования по журналам событий позволит автоматизировать процесс получения полных журналов событий системы, ускорить режим просмотра логов, автоматизировать процесс детектирования аномальных событий в системе, что способствует повышению безопасности и эффективности работы системы.

Работа выполнена в рамках выпускной квалификационной работы магистерской диссертации.

Благодарности

Авторы выражают благодарность Высшей инжиниринговой школе НИЯУ МИФИ за помощь в возможности опубликовать результаты выполненной работы.

Список литературы

[1] Khan S., Parkinson S. Discovering and utilising expert knowledge from security event logs //Journal of Information Security and Applications. – 2019. – Т. 48. – С. 102375.

[2] He S. et al. Experience report: System log analysis for anomaly detection //2016 IEEE 27th International Symposium on Software Reliability Engineering (ISSRE). – IEEE, 2016. – С. 207-218.

[3] Van Der Aalst W. et al. Process Mining manifesto //International Conference on Business Process Management. – Springer, Berlin, Heidelberg, 2011. – С. 169-194.

[4] Fluxicon Disco User's Guide, https://fluxicon.com/disco/files/Disco-User-Guide.pdf McGrath, M., Price, M.: Windows 10 in easy steps - Special Edition: To venture further. In Easy Steps Limited, Warwickshire (2015)

[5] Dwyer J., Truta T. M. Finding anomalies in windows event logs using standard deviation //9th IEEE International Conference on Collaborative Computing: Networking, Applications and Worksharing. – IEEE, 2013. – С. 563-570.

[6] Van Der Aalst W. Data science in action //Process Mining. – Springer, Berlin, Heidelberg, 2016. – С. 3-23.

[7] Berti A., van Zelst S. J., van der Aalst W. Process Mining for python (PM4Py): bridging the gap between process-and data science //arXiv preprint arXiv:1905.06169. – 2019.

[8] Van der Aalst W. M. P. Process Mining: discovery, conformance and enhancement of business processes. Springer, 2011.

[9] Van der Aalst W.M.P., Weijters A.J.M.M., Maruster L. Workflow Mining: Discovering Process Models from Event Logs // IEEE Transactions on Knowledge and Data Engineering, 2004. Vol. 16(9). P. 1128–1142.

[10] Van der Aalst W.M.P., Adriansyah A., Van Dongen B.F. Replaying history on process models for conformance checking and performance analysis // Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery. Vol. 2(2). Wiley Online Library. 2012. P. 182–192.

[11] Van der Werf J. M. E. M. et al. Process discovery using integer linear programming // Applications and Theory of Petri Nets. Springer Berlin Heidelberg, 2008. P. 368–387.

Application of Process mining technology to identify abnormal situations in the operation of high-tech equipment

Adelya Khasanova, Maxim Dunaev

***Abstract*—In the modern world, all companies use IT infrastructure to organize their activities. And an attempt to eliminate various anomalous events (including security threats) in the activities of technology platforms is becoming extremely urgent.**

Such platforms are becoming the mainstream of the IT industry, supporting a wide range of online services and intelligent applications (weather forecast, biomedical engineering, etc.). Most of these systems support the operation of complex equipment in various industries: mining, industrial design and operation of nuclear power plants, transport industry, etc. Serving thousands of computers simultaneously, almost all systems are designed to operate around the clock, serving thousands of computers simultaneously, high availability and reliability.

Any incidents with such systems, including interruptions or reduced quality of service, will lead to the exit from individual applications and, accordingly, to financial costs. In addition, malfunctioning digital equipment can lead to accidents and industrial accidents.

One of the tools for solving the above problems is the development process, which allows you to analyze processes, abnormal events, predict bottlenecks, etc.

The purpose of this work is to study and implement effective technologies for intelligent analysis of processes (Process Mining) for possible operations in event logs (using the example of Windows OS).

***KEY WORDS*—SECURITY, EVENT LOGS, PROCESS MINING, WINDOWS OS, ANOMALOUS SITUATIONS.**

REFERENCES

[1] Khan S., Parkinson S. Discovering and utilising expert knowledge from security event logs //Journal of Information Security and Applications. – 2019. – T. 48. – C. 102375.

[2] He S. et al. Experience report: System log analysis for anomaly detection //2016 IEEE 27th International Symposium on Software Reliability Engineering (ISSRE). – IEEE, 2016. – C. 207-218.

[3] Van Der Aalst W. et al. Process Mining manifesto //International Conference on Business Process Management. – Springer, Berlin, Heidelberg, 2011. – C. 169-194.

[4] Fluxicon Disco User's Guide, https://fluxicon.com/disco/files/Disco-User-Guide.pdf McGrath, M., Price, M.: Windows 10 in easy steps - Special Edition: To venture further. In Easy Steps Limited, Warwickshire (2015)

[5] Dwyer J., Truta T. M. Finding anomalies in windows event logs using standard deviation //9th IEEE International Conference on Collaborative Computing: Networking, Applications and Worksharing. – IEEE, 2013. – C. 563-570.

[6] Van Der Aalst W. Data science in action //Process Mining. – Springer, Berlin, Heidelberg, 2016. – C. 3-23.

[7] Berti A., van Zelst S. J., van der Aalst W. Process Mining for python (PM4Py): bridging the gap between process-and data science //arXiv preprint arXiv:1905.06169. – 2019.

[8] Van der Aalst W. M. P. Process Mining: discovery, conformance and enhancement of business processes. Springer, 2011.

[9] Van der Aalst W.M.P., Weijters A.J.M.M., Maruster L. Workflow Mining: Discovering Process Models from Event Logs // IEEE Transactions on Knowledge and Data Engineering, 2004. Vol. 16(9). P. 1128–1142.

[10] Van der Aalst W.M.P., Adriansyah A., Van Dongen B.F. Replaying history on process models for conformance checking and performance analysis // Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery. Vol. 2(2). Wiley Online Library. 2012. P. 182–192.

[11] Van der Werf J. M. E. M. et al. Process discovery using integer linear programming // Applications and Theory of Petri Nets. Springer Berlin Heidelberg, 2008. P. 368–387.

First A. Khasanova Adelya Marselevna. Date of birth: May 6, 1997. Place of birth: Russia, rep. Bashkortostan, Sterlitamak. Education: NRNU MEPhI, «Informatics and Computer Engineering», Bachelor's Degree (2015-2019); NRNU MEPhI, «Software Engineering», Master's Degree (2019-2021).

She works at The Rosatom State Corporation Engineering Division ASE, Engineer from 2019 to the present. Information concerning previous publications:

Khasanova A. M. Detection of Attacks on Wi-Fi Access Points //2021 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus). – IEEE, 2021. – C. 28-31.

Khasanova A. M., Pasechnik M. O. Social Media Analysis with Machine Learning //2021 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus). – IEEE, 2021. – C. 32-35.

Second A. Dunaev Maxim Evgenievich. Date of birth: March 15, 1996. Place of birth: Russia, Morshansk. Education: NRNU MEPhI, « applied informatics», Bachelor's Degree (2013-2017); NRNU MEPhI, «business Informatics », Master's Degree (2017-2019); NRNU MEPhI «Informatics and Computer Engineering», Postgraduate Degree(2019- present).

M. Dunaev, K. Zaytsev Logs analysis to search for anomalies in the functioning of large technology platforms // Journal of Theoretical and Applied Information Technology, 2019 Vol. 97, No. 11, Q3 pp. 3111-3123;

M. Dunaev, K. Zaytsev, M. Titov A study of sequential pattern mining algorithms for use in detection of user activity patterns // Journal of Theoretical and Applied Information Technology, 2018 Vol. 96, No. 13, Q3 pp. 4306-4315