|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ

КАФЕДРА ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ

**РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

***К КУРСОВОМУ ПРОЕКТУ***

***НА ТЕМУ:***

***Утилита для автоматизации этапа проверки гипотез в процессе Threat Hunting***

Студент ИУ8-33 **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  И. Д. Сушин

(Группа) (Подпись,дата) (И.О.Фамилия)

Руководитель курсового проекта **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  Е. В. Глинская

(Подпись, дата) (И.О.Фамилия)

Консультант **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

(Подпись, дата) (И.О.Фамилия)

*2023 г.*

**Министерство науки и высшего образования Российской Федерации**

**Федеральное государственное бюджетное образовательное учреждение**

**высшего образования**

**«Московский государственный технический университет имени Н.Э. Баумана**

**(национальный исследовательский университет)»**

**(МГТУ им. Н.Э. Баумана)**

УТВЕРЖДАЮ

Заведующий кафедрой ИУ8

(Индекс)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Басараб М.А.

(И.О.Фамилия)

« \_\_\_\_\_ » \_\_\_\_\_\_\_\_\_\_\_\_ 2023 г.

**ЗАДАНИЕ**

**на выполнение курсового проекта**

по дисциплине Технологии и методы программирования

Студент группы ИУ8-33

Сушин Иван Дмитриевич

(Фамилия, имя, отчество)

Тема курсового проекта Утилита для автоматизации этапа проверки гипотез в процессе Threat Hunting \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Направленность КП (учебный, исследовательский, практический, производственный, др.)

учебный

Источник тематики (кафедра, предприятие, НИР) кафедра

График выполнения проекта: 25% к 3 нед., 50% к 9 нед., 75% к 12 нед., 100% к 15 нед.

***Задание:*** Сформулировать по техникам из разных тактик MITRE ATT&CK гипотезы о том, что используют атакующие. Составить список событий, которые смогут определить выполнение определённой техники. Разработать программу, которая собирает и анализирует логи с хостов в организации в соответствии с составленным файлом конфигурации.

***Оформление курсового проекта:***

Расчетно-пояснительная записка на \_\_\_\_\_ листах формата А4.

Перечень графического (иллюстративного) материала (чертежи, плакаты, слайды и т.п.)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата выдачи задания « \_\_\_ » сентября 2023 г.

**Руководитель курсового проекта**  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Е. В. Глинская

(Подпись, дата) (И.О.Фамилия)

**Студент** **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  И. Д. Сушин

(Подпись, дата) (И.О.Фамилия)

**СОДЕРЖАНИЕ**

Введение……………………………………………………………………………….….……..4

1. Техническое задание……………………………………………………………………………5

2. Описание программы……………………………………………………………………...…...11

3. Программа и методика испытаний……………............…….……...........................................19

4. Заключение……………………………………………………………...……………...…....…28

5. Приложение А. Исходные коды программы…...…………………...………………………..28

Введение  
 Киберпреступники и кибершпионы непрерывно развивают новые и продвинутые методы для вторжения в сети и компьютерные системы организаций. Некоторые меры безопасности остаются неэффективными против продвинутых и целенаправленных атак. Поэтому принять активные меры по обнаружению современных угроз помогает Threat Hunting и является необходимым условием для эффективной защиты информации. С помощью журнала событий можно выявить активность процессов, по которым могут быть обнаружены многие действия атакующих.  
 Цель проекта заключается в сравнении событий, которые записываются в журнале с событиями, которые я предположил в файле конфигурации.  
 Для достижения цели необходимо решить следующие задачи:  
1. Изучить предметную область посвящённую Threat Hunting  
2. Выбрать и проанализировать технику из MITRE ATT&CK  
3. Составить список событий, по которым можно определить, что данная техника была использована (идентификатор, название журнала, поле и часть, отвечающая за детектирующую логику, связанную с полем)  
4. Изучить синтаксис языка программирования powershell  
5. Разработать файл конфигурации  
6. Научиться выборочно собирать события с удаленного хоста  
7. разработать программу и её описание

1. Техническое задание

**СОДЕРЖАНИЕ**

ВВЕДЕНИЕ 6

[1. ОСНОВАНИЕ ДЛЯ РАЗРАБОТКИ 6](#_Toc120660547)

[2. НАЗНАЧЕНИЕ РАЗРАБОТКИ 6](#_Toc120660548)

[3. ТРЕБОВАНИЯ К ПРОГРАММЕ 6](#_Toc120660549)

[3.1. Требования к функциональным характеристикам 6](#_Toc120660550)

[3.2. Требования к надежности 7](#_Toc120660551)

[3.3. Условия эксплуатации 7](#_Toc120660552)

[3.4. Требования к составу и параметрам технических средств 7](#_Toc120660553)

[3.5. Требования к информационной и программной совместимости 8](#_Toc120660554)

[3.6. Требования к маркировке и упаковке 8](#_Toc120660555)

[3.7. Требования к транспортированию и хранению 8](#_Toc120660556)

[3.8. Специальные требования 8](#_Toc120660557)

[4. ТРЕБОВАНИЯ К ПРОГРАММНОЙ ДОКУМЕНТАЦИИ 8](#_Toc120660558)

[4.1. Предварительный состав программной документации 8](#_Toc120660559)

[4.2. Специальные требования к программной документации 8](#_Toc120660560)

[5. ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ 9](#_Toc120660561)

[5.1. Ориентировочная экономическая эффективность 9](#_Toc120660562)

[5.2. Предполагаемая годовая потребность 9](#_Toc120660563)

[5.3. Экономические преимущества разработки 9](#_Toc120660564)

[6. СТАДИИ И ЭТАПЫ РАЗРАБОТКИ 9](#_Toc120660565)

[6.1. Стадии разработки 9](#_Toc120660566)

[6.2. Этапы разработки 9](#_Toc120660567)

[6.3. Содержание работ по этапам 9](#_Toc120660568)

[6.4. Исполнители 10](#_Toc120660569)

[7. ПОРЯДОК КОНТРОЛЯ И ПРИЕМКИ 10](#_Toc120660570)

[7.1. Виды испытаний 10](#_Toc120660571)

[7.2. Общие требования к приемке работы 10](#_Toc120660572)

ВВЕДЕНИЕ

Наименование программы - «Утилита для автоматизации этапа проверки гипотез в процессе Threat Hunting».

Программа предназначена для применения в учебных целях.

## 1. ОСНОВАНИЕ ДЛЯ РАЗРАБОТКИ

Основанием для проведения разработки является учебный план кафедры ИУ8 и рабочая программа дисциплины «Технологии и методы программирования».

Наименование темы разработки - «Разработка утилиты для автоматизации этапа проверки гипотез в процессе Threat Hunting».

Условное обозначение темы разработки (шифр темы) «КП00001».

# 2. НАЗНАЧЕНИЕ РАЗРАБОТКИ

Функциональное назначение программы:

Функциональным назначением программы является проверка гипотез.

Эксплуатационное назначение программы:

Программа должна эксплуатироваться в учебных учреждениях.

Конечными пользователями программы должны являться преподаватели или студенты учебного учреждения.

# 3. ТРЕБОВАНИЯ К ПРОГРАММЕ

## 3.1. Требования к функциональным характеристикам

*Требования к составу выполняемых функций*

Программа должна обеспечивать возможность выполнения перечисленных ниже функций:

а) ввод пользователем данных для совершения проверки;

б) вывод данных.

*Требования к организации входных данных*

Входные данные должны быть отображены в графическом интерфейсе приложения.

*Требования к организации выходных данных*

Выходные должны быть отображены в графическом интерфейсе приложения.

*Требования к временным характеристикам*

Требования к временным характеристикам программы не предъявляются.

## 3.2. Требования к надежности

Требования к обеспечению надежного (устойчивого) функционирования программы.

Надежное (устойчивое) функционирование программы должно быть обеспечено выполнением совокупности организационно-технических мероприятий, перечень которых приведен ниже:

а) организацией бесперебойного питания технических средств;

б) регулярным выполнением рекомендаций Министерства труда и социального развития РФ, изложенных в Постановлении от 23 июля 1998 г. «Об утверждении межотраслевых типовых норм времени на работы по сервисному обслуживанию ПЭВМ и оргтехники и сопровождению программных средств»;

в) регулярным выполнением требований ГОСТ 51188-98. Защита информации. Испытания программных средств на наличие компьютерных вирусов.

*Время восстановления после отказа*

Время восстановления после отказа, вызванного сбоем электропитания технических средств (иными внешними факторами), не фатальным сбоем (не крахом) операционной системы, не должно превышать времени, необходимого на перезагрузку операционной системы и запуск программы, при условии соблюдения условий эксплуатации технических и программных средств.

Время восстановления после отказа, вызванного неисправностью технических средств, фатальным сбоем (крахом) операционной системы, не должно превышать времени, требуемого на устранение неисправностей технических средств и переустановки программных средств.

## 3.3. Условия эксплуатации

*Климатические условия эксплуатации*

Климатические условия эксплуатации, при которых должны обеспечиваться заданные характеристики, должны удовлетворять требованиям, предъявляемым к техническим средствам в части условий их эксплуатации.

*Требования к численности и квалификации персонала*

Минимальное количество персонала, требуемого для работы программы, должно составлять не менее 1 штатной единицы - конечный пользователь программы - оператор.

Конечный пользователь программы (оператор) должен обладать практическими навыками работы с графическим пользовательским интерфейсом операционной системы.

## 3.4. Требования к составу и параметрам технических средств

В состав технических средств должен входить IBM-совместимый персональный компьютер (ПЭВМ), работающий на ОС Windows.

## 3.5. Требования к информационной и программной совместимости

*Требования к информационным структурам и методам решения*

Требования к информационным структурам на входе и выходе, а также к методам решения не предъявляются.

*Требования к исходным кодам и языкам программирования*

Исходные коды программы должны быть реализованы на языке PowerShell. В качестве интегрированной среды разработки программы может быть использована среда Windows PowerShell ISE.

*Требования к защите информации и программ*

Требования к защите информации и программ не предъявляются.

## 3.6. Требования к маркировке и упаковке

Исполняемые и исходные коды программы могут храниться на CD или DVD диске или флеш-носителе.

## 3.7. Требования к транспортированию и хранению

Не предъявляются.

## 3.8. Специальные требования

Специальные требования к программе не предъявляются.

# 4. ТРЕБОВАНИЯ К ПРОГРАММНОЙ ДОКУМЕНТАЦИИ

## 4.1. Предварительный состав программной документации

Состав программной документации должен включать в себя:

1. техническое задание;
2. спецификацию;
3. текст программы;
4. описание программы;
5. программу и методики испытаний;
6. пояснительную записку;
7. ведомость эксплуатационных документов;
8. формуляр;
9. описание применения;
10. руководство программиста;
11. руководство пользователя.

## 4.2. Специальные требования к программной документации

Специальные требования к программной документации не предъявляются.

# 5. ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ

## 5.1. Ориентировочная экономическая эффективность

Ориентировочная экономическая эффективность не рассчитывается.

## 5.2. Предполагаемая годовая потребность

Предполагаемая годовая потребность не рассчитывается.

## 5.3. Экономические преимущества разработки

Экономические преимущества разработки не рассчитываются.

# 6. СТАДИИ И ЭТАПЫ РАЗРАБОТКИ

## 6.1. Стадии разработки

Разработка должна быть проведена в три стадии:

1) разработка технического задания;

2) рабочее проектирование;

3) внедрение.

## 6.2. Этапы разработки

На стадии разработки технического задания должен быть выполнен этап разработки, согласования и утверждения настоящего технического задания.

На стадии рабочего проектирования должны быть выполнены перечисленные ниже этапы работ:

1) разработка программы;

2) разработка программной документации;

3) испытания программы.

На стадии внедрения должен быть выполнен этап разработки - подготовка и передача программы.

## 6.3. Содержание работ по этапам

На этапе разработки технического задания должны быть выполнены перечисленные ниже работы:

1) постановка задачи;

2) определение и уточнение требований к техническим средствам;

3) определение требований к программе;

4) определение стадий, этапов и сроков разработки программы и документации на неё;

5) выбор языков программирования;

6) согласование и утверждение технического задания.

На этапе разработки программы должна быть выполнена работа по программированию и отладке программы.

На этапе разработки программной документации должна быть выполнена разработка программных документов в соответствии с требованиями ГОСТ 19.101-77 и требованием п. «Предварительный состав программной документации» настоящего технического задания.

На этапе испытаний программы должны быть выполнены перечисленные ниже виды работ:

1) разработка, согласование и утверждение программы и методики испытаний;

2) проведение испытаний;

3) корректировка программы и программной документации по результатам испытаний.

На этапе подготовки и передачи программы должна быть выполнена работа по подготовке и передаче программы и программной документации в эксплуатацию.

## 6.4. Исполнители

Исполнитель:

Студент гр. ИУ8-33 Сушин И. Д.

# 7. ПОРЯДОК КОНТРОЛЯ И ПРИЕМКИ

## 7.1. Виды испытаний

Испытания программы должны проводиться согласно разработанной и согласованной «Программы и методики испытаний».

Ход проведения испытаний документируется в Протоколе проведения испытаний.

## 7.2. Общие требования к приемке работы

После проведения испытаний в полном объеме, на основании «Протокола испытаний» утверждают «Свидетельство о приемке» и производят запись в программном документе «Формуляр».

2. Описание программы

**СОДЕРЖАНИЕ**

[1. Общие сведения 12](#_Toc120660192)

[2. Функциональное назначение 13](#_Toc120660193)

[3. Описание логической структуры 14](#_Toc120660194)

[3.1. Алгоритм программы 14](#_Toc120660195)

[3.2. Структура программы с описанием функций составных частей и связи между ними 14](#_Toc120660196)

[4. Используемые технические средства 16](#_Toc120660197)

[5. Вызов и загрузка 17](#_Toc120660198)

[6. Входные данные 18](#_Toc120660199)

[7. Выходные данные 19](#_Toc120660200)

# **1. Общие сведения**

Специальное программное обеспечение защиты информации разрабатывается под наименованием «УиПЛ».

Для функционирования СЗИ необходимо следующее программное обеспечение: 32- разрядные или 64-разрядные операционные системы (ОС) Windows 7 или Windows Server 2008, далее просто ОС Windows.

Для написания комплекса программных средств использовался язык PowerShell

# **2. Функциональное назначение**

Программа предназначена для проверки гипотез в процессе Threat Hunting.

Программа решает задачу подтверждение гипотезы о применении технике атакующего.

# **3. Описание логической структуры**

## 3.1. Алгоритм программы

При запуске программы ожидается в выводе следующее:

• “ Сработала техника t ...“ — это значит подтверждение гипотезы

• Ничего — это значит, что гипотеза не подтвердилась или атаки не было

## 3.2. Структура программы с описанием функций составных частей и связи между ними

Таблица 1, Название функции

|  |  |
| --- | --- |
| Function MyCompare | Функция, которая одновременно парсит и сверяет свойства события |

# **4. Используемые технические средства**

Защищаемые вычислительные средства являются персональными компьютерами на базе 32-разрядных или 64-разрядных (операционная система является 32-разрядной) процессоров архитектуры типа Intel, AMD. ПК функционируют в составе локальной вычислительной сети на базе технологии Ethernet.

# **5. Вызов и загрузка**

В программе подгружается файл с конфигом

$Config = (gc Путь до файла конфигурации\config.txt)

Вызов программы осуществляется путём открытия сценария в самой интегрированной среде Windows PowerShell ISE.

# **6. Входные данные**

Входные данные представляют собой .txt файл, ручной ввод или автоматический ввод данных.

Данные записаны с новой строки. Данные, подающиеся на вход это есть предполагаемые события.

# **7. Выходные данные**

Выходные данные представлены в консоле программы.

3. Программа и методика испытаний

Содержание

[Введение 21](file:///C:\Работа\ТиМП\КП\Программа%20и%20методика%20испытаний.doc#_Toc394177306)

[1. Объект испытаний 22](file:///C:\Работа\ТиМП\КП\Программа%20и%20методика%20испытаний.doc#_Toc394177307)

[2. Цель испытаний 23](file:///C:\Работа\ТиМП\КП\Программа%20и%20методика%20испытаний.doc#_Toc394177308)

[3. Требования к программе 24](file:///C:\Работа\ТиМП\КП\Программа%20и%20методика%20испытаний.doc#_Toc394177309)

[4. Требования к программной документации 24](file:///C:\Работа\ТиМП\КП\Программа%20и%20методика%20испытаний.doc#_Toc394177310)

[5. Средства и порядок испытаний 26](file:///C:\Работа\ТиМП\КП\Программа%20и%20методика%20испытаний.doc#_Toc394177311)

# **Введение**

Программное обеспечение разрабатывается под наименованием «УиПЛ»

Данная программа и методика испытаний разработана как руководство для проведения предварительных (комплексных испытаний) доработанного программного комплекса.

# **1. Объект испытаний**

Объектом испытаний является программа разработка утилита для автоматизации этапа проверки гипотез в процессе Threat Hunting. Разработанная программапредназначена для проверки составленных гипотез. Функционирует в среде 32-разрядных или 64-разрядных операционных систем (ОС) Windows 7, Windows Server 2008, далее просто ОС Windows, и устанавливается как на автономных рабочих местах, так и на рабочих станциях и серверах локальной вычислительной сети.

# **2. Цель испытаний**

Цель испытаний состоит в проверке степени соответствия реальных характеристик разработанной программы определенным требованиям.

# **3. Требования к программе**

К программе предъявляются следующие требования:

* Вывод результата работы в консоль
* Ввод данных из файла

# **4. Требования к программной документации**

В качестве программной документации разрабатываются:

* «Техническое задание»;
* «Спецификация»;
* «Пояснительная записка»;
* «Руководство оператора»;
* «Руководство по комплексу средств защиты»;
* «Руководство системного программиста»;
* «Программа и методика испытаний».

# **5. Средства и порядок испытаний**

В ходе проведения испытаний проверяются соответствие реальных характеристик разработанного программного комплекса требованиям.

Испытания проводятся на реальных рабочих местах, на которых должен устанавливаться программный комплекс, в обычных условиях эксплуатации. В ходе проведения испытаний должны выполняться следующие тесты.

Порядок испытаний:

1. Реализация методики проверки инсталляции.
2. Реализация методики проверки наличия и качества программной документации.
3. Реализация проверки функционирования программного комплекса и соответствия функциональных характеристик комплекса требованиям ТЗ.

Заключение

В результате работы над проектом была создана разработка программа для автоматизации этапа проверки гипотез в процессе Threat Hunting. В ходе проведенных испытаний тесты показали, что разработанная программа соответствует требованиям технического задания.

Все поставленные задачи в ходе выполнения проекта были полностью решены.

Приложение.   
Исходный код программы

# Загрузка файла с конфигом

$Config = (gc C:\config.txt)

# Создаю объект со свойсвами и переменную, хранящую номер техники

$event\_cfg = [PSCustomObject]@{Id = $null; Journal = $null; Field = $null; Pattern = $null}

$techniq = $null

# Функция, которая одновременно парсит и сверяет свойства события

Function MyCompare

{

Param ($event)

ForEach ($one in $Config)

{

# Первым парсим id конфига и сверяем его с id события

if ($one -like "\*id\*")

{

$event\_cfg.Id = $one.Substring($one.IndexOf(":")+2)

}

# Парсим поле

if ($one -like "\*field\*")

{

$event\_cfg.Field = $one.Remove($one.Length-1).Substring($one.IndexOf(":")+3)

}

# Парсим паттерн - проделанная техника

if ($one -like "\*pattern\*")

{

$event\_cfg.Pattern = $one.Remove($one.Length-1).Substring($one.IndexOf(":")+3)

}

# Последним парсим название лога в конфиге, где выполняется главная проверка проделанной техники

if ($one -like "\*journal\*")

{

$event\_cfg.Journal = $one.Substring($one.IndexOf(":")+2)

# Сравниваем по названию лога и полю

if(

($event.Id -eq $event\_cfg.Id) -and

($event.LogName -eq $event\_cfg.Journal) -and

(($event\_cfg.Field -like $event.Message) -or -not[bool]$event\_cfg.Field))

{ Write-Host("Сработала техника: ", $techniq) }

}

# Если наткнулись на "event" в конфиге, то обнуляем все свойства объекта (нужно например в отсутствии какого-то свойства в конфиге)

if ($one -like "\*event\*")

{

$event\_cfg.Field = $null

$event\_cfg.Id = $null

$event\_cfg.Journal = $null

$event\_cfg.Pattern = $null

}

# Парсим номер техники

if ($one -like "t\*")

{

$techniq = $one.Remove($one.Length-1).Substring(1)

}

}

}

$startTime = (Get-Date).AddHours(-1)

$endTime = Get-Date

$cred = Get-Credential -Credential admin

$compname = "192.168.43.230"

$logs = Get-WinEvent -ComputerName $compname -Credential $cred -ListLog \* 2>$null

foreach ($log in $logs)

{

$events = Get-WinEvent -ComputerName $compname -Credential $cred -FilterHashtable @{LogName = $log.LogName; StartTime=$startTime; EndTime=$endTime} 2> $null

foreach($event in $events)

{

MyCompare -event $event

}

}

Конфигурация программы  
t1070.001:

event\_1:

id: 1102

journal: Security

event\_2:

id: 104

journal: System

event\_3:

id: 1

field: "CommandLine"

pattern: "\*wevtutil cl\*"

journal: Microsoft-Windows-Sysmon/Operational

event\_4:

id: 4104

field: "Scriptblock"

pattern: "\*Remove-Event\*"

journal: Microsoft-Windows-PowerShell/Operational

t1547.004:

event\_1:

id: 4657

field: "modified"

pattern: "\*Set-Item\*"

journal: Security

event\_2:

id: 4660

field: "deleted"

pattern: "\*Remove-Item\*"

journal: Security

t1053.005:

event\_1:

id: 4698

field: "created"

pattern: "\*schtasks\*/create\*"

journal: Security

event\_2:

id: 4699

field: "deleted"

pattern: "\*schtasks\*/delete\*"

journal: Security

event\_3:

id: 102

field: "TaskCompletion"

pattern: "\*Register-ScheduledTask\*"

journal: Microsoft-Windows-TaskScheduler

event\_4:

id: 201

field: "TaskFailed"

pattern: "\*Unregister-ScheduledTask\*"

journal: Microsoft-Windows-TaskScheduler