Министерство образования Республики Беларусь

Учреждение образования

«Брестский государственный технический университет»

Кафедра интеллектуально-информационных технологий

РЕФЕРАТ

по дисциплине

«Современные методы защиты компьютерных систем»

Выполнила:
студент 4 курса,
ФЭИС,
группы ИИ-22
Дубина Н.С.

Брест-2024

PowerShell и Bash

PowerShell и **Bash** — это мощные скриптовые оболочки, которые используются для автоматизации задач, управления системами и взаимодействия с операционной системой. Однако они разработаны для разных целей и платформ, поэтому имеют свои особенности.

PowerShell — это скриптовый язык и оболочка, разработанная компанией Microsoft. Он был создан для автоматизации задач в операционных системах Windows и интеграции с экосистемой Microsoft. Основная особенность PowerShell — работа с объектами. В отличие от других оболочек, PowerShell использует объектно-ориентированный подход: команды (их называют "cmdlet") возвращают не текст, а объекты .NET, которые легко передавать из одной команды в другую. Это делает обработку данных чрезвычайно мощной и гибкой. PowerShell также предоставляет встроенные возможности для управления процессами, файлами, реестром, службами и даже облачными сервисами, такими как Azure. В более современных версиях, начиная с PowerShell Core, появилась кроссплатформенность, что позволяет использовать его на Windows, macOS и Linux.

Bash (Bourne Again Shell) — это стандартная оболочка Unix-систем, которая появилась как замена оригинальной оболочки Bourne Shell. Bash стал стандартом для большинства дистрибутивов Linux и широко используется для автоматизации задач в этих системах. Bash работает с текстом и строками, что делает его чрезвычайно эффективным для обработки файлов, создания скриптов и управления конфигурацией систем. Он поддерживает большое количество встроенных команд и утилит, что позволяет решать практически любые задачи. Bash отличается простой синтаксической структурой, а благодаря возможности использования потоков ввода/вывода и перенаправления данных пользователи могут эффективно соединять команды в цепочки.

Основные характеристики PowerShell:

1. **Объектно-ориентированная архитектура**:
   В отличие от традиционных оболочек, таких как Bash, которые работают преимущественно с текстом, PowerShell работает с объектами. Каждая команда возвращает объекты .NET, которые содержат свойства и методы. Это позволяет легко фильтровать, изменять и передавать данные между командами.
2. **Командлеты (cmdlets)**:
   PowerShell использует специальные команды, называемые *cmdlet*, которые представляют собой небольшие функции, предназначенные для выполнения определённых задач. Все cmdlet имеют стандартный синтаксис, состоящий из пары "глагол-сущность" (например, Get-Process, Set-Item, Remove-File). Этот подход делает команды интуитивно понятными и легко читаемыми.
3. **Пайплайны**:
   PowerShell поддерживает передачу объектов из одной команды в другую через пайплайны. Это позволяет создавать мощные цепочки обработки данных, где выход одной команды становится входом для другой.

4. **Кроссплатформенность**:
   С появлением PowerShell Core (начиная с версии 6) оболочка стала доступна не только на Windows, но и на macOS и Linux. Это позволило использовать PowerShell для управления смешанными инфраструктурами.
5. **Интеграция с Windows**:
   PowerShell идеально подходит для управления Windows, предоставляя доступ к таким компонентам, как Active Directory, реестр, службы, файлы, процессы и события. Также через PowerShell можно управлять различными продуктами Microsoft, включая Exchange Server, Azure и SharePoint.
6. **Расширяемость**:
   Пользователи могут создавать собственные функции, модули и даже cmdlet. Кроме того, PowerShell поддерживает использование .NET библиотек, что позволяет писать сложные сценарии с использованием стандартных и сторонних API.
7. **Скрипты и автоматизация**:
   Скрипты PowerShell (*.ps1 файлы) используются для выполнения повторяющихся задач. Они могут включать логические конструкции (if/else, циклы), вызовы функций и обработку ошибок, что делает их полноценным инструментом для DevOps и системного администрирования.

Основные характеристики Bash:

1. **Текстовый и строковый подход**:
   Bash работает преимущественно с текстовыми данными, а не объектами. Это делает его простым для выполнения задач, связанных с обработкой файлов, текстов и вывода команд.
2. **Универсальность**:
   Bash является стандартной оболочкой для большинства дистрибутивов Linux и macOS. Он совместим с огромным количеством системных утилит и программ, что делает его основным инструментом системного администрирования.
3. **Потоки ввода/вывода и перенаправление данных**:
   Bash позволяет легко перенаправлять данные между командами, файлами и устройствами ввода/вывода. Это делает его удобным для создания сложных цепочек обработки данных.
4. **Масштабируемость**:
   Bash-скрипты могут быть простыми однострочными командами или сложными программами с использованием циклов, функций и логических конструкций.
5. **Поддержка внешних утилит**:
   Bash интегрируется с мощными инструментами, такими как awk, sed, grep, которые позволяют выполнять сложные операции по анализу и обработке данных.
6. **Кроссплатформенность**:
   Хотя Bash изначально разработан для Unix-подобных систем, он может быть установлен и использоваться на Windows через WSL (Windows Subsystem for Linux).

Примеры возможностей PowerShell:

1. **Управление файлами и папками**:
   PowerShell позволяет искать файлы в заданной папке и её подкаталогах, фильтруя их по различным параметрам, таким как размер, дата создания или расширение. Например, можно найти все текстовые файлы в конкретной директории или удалить устаревшие файлы.
2. **Управление процессами**:
   С помощью PowerShell можно отслеживать процессы, работающие на компьютере, сортировать их по нагрузке на процессор или память, завершать ненужные процессы или анализировать их характеристики, такие как использование ресурсов.
3. **Работа с Active Directory**:
   PowerShell позволяет управлять учетными записями пользователей и групп в Active Directory. Например, можно получить список всех пользователей, обновить их контактные данные, сбросить пароли или назначить новые права доступа.
4. **Подключение к удалённым системам**:
   PowerShell поддерживает выполнение команд на удалённых компьютерах. Это позволяет управлять серверами или рабочими станциями из одной центральной консоли, например, перезагружать службы или собирать системную информацию.
5. **Облачные технологии**:
   PowerShell интегрируется с облачными платформами, такими как Microsoft Azure. Это позволяет автоматизировать создание виртуальных машин, управление хранилищами данных и настройку сетевых параметров.
6. **Мониторинг и аудит**:
   PowerShell может собирать данные о состоянии системы, например, информацию о состоянии службы, журналах событий или доступных обновлениях. Это полезно для создания отчётов или настройки автоматических уведомлений.
7. **Интеграция с API**:
   PowerShell может взаимодействовать с внешними веб-сервисами, получая данные из API. Это используется для интеграции разных систем или автоматической обработки данных из интернета, таких как загрузка и анализ JSON-ответов.

Примеры возможностей Bash:

1. **Управление файлами и каталогами**:
   Bash позволяет создавать, перемещать, копировать или удалять файлы и папки. Например, можно массово переименовать файлы в директории, сортировать их по размеру или времени создания, а также организовывать резервное копирование данных.
2. **Работа с текстовыми данными**:
   Bash часто используется для обработки текстовых файлов. Он может фильтровать строки, находить и заменять текст, объединять содержимое нескольких файлов или анализировать логи. Например, можно выделить строки с ошибками из системного журнала.
3. **Автоматизация задач**:
   Bash-скрипты помогают автоматизировать рутинные процессы, такие как регулярное удаление временных файлов, запуск приложений по расписанию или выполнение резервного копирования.

4. **Управление процессами**:
   В Bash можно управлять процессами на уровне операционной системы: запускать программы в фоновом режиме, завершать ненужные процессы, отслеживать их статус или изменять приоритет выполнения.
5. **Обработка потоков данных**:
   Bash поддерживает перенаправление ввода и вывода, что позволяет соединять команды в цепочки. Например, можно передать вывод одной команды на вход другой для фильтрации или дальнейшей обработки данных.
6. **Сетевые операции**:
   Bash активно используется для работы с сетью. Это может быть автоматизация загрузки файлов через протоколы FTP или HTTP, пингование серверов для проверки их доступности, настройка сетевых интерфейсов или мониторинг сетевого трафика.
7. **Администрирование системы**:
   В Linux с помощью Bash можно управлять системными настройками: изменять разрешения файлов, добавлять или удалять пользователей, настраивать службы, устанавливать обновления и управлять конфигурацией системы.
8. **Мониторинг и аудит**:
   Bash позволяет собирать информацию о системе, например, данные о дисковом пространстве, использовании памяти или загруженности процессора. Эти данные можно использовать для анализа производительности или выявления проблем.
9. **Работа с другими утилитами**:
   Bash интегрируется с большим количеством инструментов, таких как awk, sed и grep. Это позволяет выполнять сложные текстовые преобразования, анализировать данные и создавать мощные сценарии для обработки информации.
10. **Управление пакетами и обновлениями**:
    Bash используется для установки, обновления и удаления программных пакетов в системах на базе Linux. Это упрощает управление программным обеспечением через пакетные менеджеры, такие как apt или yum.

Области применения PowerShell:

1. **Системное администрирование:**
   Автоматизация рутинных задач, таких как управление пользователями, настройка сетевых параметров, управление сервисами и конфигурациями систем.
2. **Управление облачными сервисами:**
   PowerShell активно используется для работы с Microsoft Azure. Например, через Azure PowerShell можно развертывать ресурсы, управлять виртуальными машинами и настраивать сетевые подключения.
3. **Интеграция с DevOps:**
   PowerShell используется для настройки CI/CD процессов, управления конфигурациями с помощью таких инструментов, как DSC (Desired State Configuration), и написания скриптов для работы с системами контроля версий (например, Git).
4. **Мониторинг и аудит:**

PowerShell может использоваться для сбора данных о состоянии системы, анализировать журналы событий и автоматизировать отчётность.

5. **Работа с API и JSON:**
   Встроенные команды для работы с REST API и данными в формате JSON позволяют PowerShell быть инструментом для интеграции различных систем.

Области применения Bash:

1. **Системное администрирование**:
   Bash активно используется для управления Linux-серверами. Он помогает в настройке системы, мониторинге процессов, управлении файлами и конфигурацией.
2. **Автоматизация рутинных задач**:
   Администраторы создают Bash-скрипты для выполнения повторяющихся задач, таких как создание резервных копий, очистка временных файлов или установка обновлений.
3. **Обработка данных**:
   Bash широко используется для анализа текстовых файлов, логов и журналов. Это может включать фильтрацию строк, поиск ошибок или извлечение ключевых данных.
4. **Сетевые операции**:
   В Bash легко автоматизировать работу с сетью: подключение по SSH, загрузку файлов, проверку доступности серверов и настройку сетевых интерфейсов.
5. **Разработка и тестирование**:
   Bash применяется разработчиками для автоматизации сборки проектов, запуска тестов и развертывания приложений.
6. **Интеграция с DevOps**:
   Bash активно используется в инструментах CI/CD для настройки конвейеров сборки, развертывания и тестирования.
7. **Обучение и начальная настройка**:
   Благодаря простоте и популярности Bash является первым инструментом, с которым знакомятся многие начинающие системные администраторы и разработчики.
8. **Контейнеризация и виртуализация**:
   В Bash пишут скрипты для управления контейнерами Docker, оркестраторами Kubernetes и виртуальными машинами, что делает его важным инструментом в современных IT-инфраструктурах.

Cyber Kill Chain

Cyber Kill Chain — это концептуальная модель, разработанная компанией Lockheed Martin, которая описывает последовательные этапы проведения кибератаки. Она изначально была создана для анализа киберугроз и разработки мер защиты,

позволяющих прервать атаку на различных этапах. Эта модель широко используется в кибербезопасности для понимания, обнаружения и предотвращения угроз.

Этапы Cyber Kill Chain:

1. **Разведка (Reconnaissance)**
   На этом этапе злоумышленник собирает информацию о своей цели. Это может включать изучение открытых данных, таких как социальные сети, публичные веб-ресурсы, доменные имена, IP-адреса и уязвимости в системах.
   Цель — понять, как можно получить доступ к сети или системе.
2. **Вооружение (Weaponization)**
   Злоумышленник создает вредоносное ПО или готовит эксплойты, которые будут использованы для атаки. Это может быть вирус, троян, фишинговое письмо или программный скрипт для эксплуатации уязвимости.
3. **Доставка (Delivery)**
   На этом этапе вредоносное ПО или атака доставляются цели. Это может быть сделано через электронную почту (фишинг), вредоносные ссылки, заражённые USB-устройства или уязвимости веб-приложений.
4. **Эксплуатация (Exploitation)**
   Злоумышленник использует уязвимость в системе или человеческом факторе, чтобы выполнить атаку. Например, запуск вредоносного файла на компьютере жертвы или эксплуатация недостатков в программном обеспечении.
5. **Установка (Installation)**
   Вредоносное ПО устанавливается на целевой системе. Это может быть скрытая программа, которая даёт злоумышленнику возможность удалённого управления или запускает последующие атаки.
6. **Управление (Command and Control, C2)**
   Установленное ПО связывается с сервером злоумышленника, чтобы получать команды или передавать данные. Этот этап позволяет злоумышленнику управлять зараженной системой.
7. **Действия на цели (Actions on Objectives)**
   На завершающем этапе злоумышленник достигает своей цели: похищает данные, разрушает системы, нарушает их работу или выполняет иные действия, в зависимости от намерений атаки.

Использование модели:

Cyber Kill Chain используется для:

- **Обнаружения атак**: Понимание этапов позволяет быстрее распознать угрозу на ранних стадиях, например, во время разведки или доставки.
- **Прерывания атак**: Защита строится таким образом, чтобы на каждом этапе атаки злоумышленник встречал препятствия, такие как фильтрация входящей почты, защита веб-приложений, системы обнаружения вторжений.
- **Анализа инцидентов**: Модель помогает разложить кибератаку на этапы и понять, как она была проведена, чтобы улучшить защиту в будущем.

MITRE ATT&CK

MITRE ATT&CK — это база знаний, разработанная организацией MITRE, которая систематизирует информацию о тактиках, техниках и процедурах (TTPs) злоумышленников. Она используется специалистами по кибербезопасности для анализа атак, разработки защитных мер и построения стратегий противодействия угрозам.

Основные компоненты MITRE ATT&CK

1. **Тактики (Tactics)**
   Тактики — это цели, которых злоумышленник пытается достичь на каждом этапе атаки. Они описывают, *что* хочет сделать злоумышленник (например, получить доступ, установить контроль, украсть данные).
   Примеры тактик:

   1) Первоначальный доступ (Initial Access)
   2) Выполнение (Execution)
   3) Эскалация привилегий (Privilege Escalation)
   4) Сбор данных (Collection)

2. **Техники (Techniques)**
   Техники описывают, *как* злоумышленник достигает своих целей. Это конкретные методы, которые используются для выполнения действий.
   Примеры техник:

   1) Фишинг для первоначального доступа.
   2) Использование скриптов PowerShell для выполнения.
   3) Кража учетных данных из памяти системы.

3. **Подтехники (Sub-techniques)**
   Подтехники являются уточнениями конкретных техник. Например, если техника "Фишинг" описывает общий способ обмана, подтехники могут включать "Фишинг через электронную почту" или "Фишинг через социальные сети".
4. **Процедуры (Procedures)**
   Процедуры представляют собой детализированные шаги, которые используют реальные злоумышленники или группы при выполнении техник. Это описание реальных сценариев атак.

Примеры тактик и техник MITRE ATT&CK

1. **Первоначальный доступ (Initial Access)**
   Цель: Получить доступ к целевой системе.
   Техники:

   1) Фишинг (Phishing)
   2) Эксплуатация уязвимостей публичных приложений (Exploit Public-Facing Application).

2. **Эскалация привилегий (Privilege Escalation)**
   Цель: Получить более высокий уровень доступа в системе.
   Техники:

   1) Эксплуатация уязвимостей (Exploitation for Privilege Escalation).
   2) Использование доверенных утилит (Abuse Elevation Control Mechanism).

3. **Командование и управление (Command and Control)**
   Цель: Поддерживать связь между зараженной системой и сервером злоумышленника.
   Техники:

   1) Использование скрытых каналов связи (Ingress Tool Transfer).
   2) Шифрованный обмен данными (Encrypted Channel).

Использование MITRE ATT&CK

1. **Анализ атак**:
   MITRE ATT&CK помогает понять, какие методы использовал злоумышленник, чтобы проникнуть в систему, закрепиться и достичь своих целей. Это особенно полезно для ретроспективного анализа инцидентов.
2. **Разработка защитных мер**:
   База знаний предоставляет информацию о том, как защититься от каждой техники. Например, для предотвращения фишинга можно использовать обучение сотрудников, фильтрацию электронной почты и двухфакторную аутентификацию.
3. **Оценка безопасности**:
   Организации используют MITRE ATT&CK для проведения тестов на проникновение (penetration testing) и оценки своей защищенности от известных техник злоумышленников.
4. **Создание систем обнаружения и мониторинга**:
   С помощью MITRE ATT&CK можно настроить системы обнаружения угроз, такие как SIEM и EDR, чтобы они отслеживали подозрительные действия в системе.

Варианты применения

MITRE ATT&CK разделяется на несколько матриц, каждая из которых предназначена для определённой среды:

1. **Enterprise Matrix**: охватывает атаки на корпоративные сети и инфраструктуры, включая Windows, macOS, Linux и облачные платформы.
2. **Mobile Matrix**: описывает атаки на мобильные устройства, такие как iOS и Android.
3. **ICS Matrix**: фокусируется на угрозах для промышленных систем управления (Industrial Control Systems).

MITRE ATT&CK является мощным инструментом для специалистов по кибербезопасности. Его применение позволяет не только лучше понимать угрозы, но и выстраивать эффективные стратегии защиты, опираясь на реальные данные о поведении злоумышленников.

SIEM

SIEM (Security Information and Event Management) — это система управления информацией и событиями безопасности, которая помогает собирать, анализировать и управлять данными о безопасности в реальном времени. SIEM позволяет централизованно собирать данные о событиях и инцидентах, происходящих в различных системах и приложениях организации, для их дальнейшего анализа, обнаружения угроз и обеспечения соответствия нормативным требованиям.

Основные функции SIEM

1. **Сбор данных**:
   SIEM собирает логи и события из множества источников, таких как сетевые устройства (например, маршрутизаторы, коммутаторы), серверы, рабочие станции, базы данных, системы безопасности (например, IDS/IPS, файрволы) и приложения.
2. **Нормализация данных**:
   Данные, собранные из разных источников, могут быть представлены в различных форматах. SIEM нормализует эти данные, преобразуя их в унифицированный формат, что упрощает их дальнейший анализ и обработку.
3. **Корреляция событий**:
   SIEM анализирует события и логи в реальном времени, чтобы выявить подозрительные паттерны или аномалии, которые могут указывать на угрозы. Например, системные логи могут быть сопоставлены с сетевыми событиями для выявления потенциальных атак.
4. **Обнаружение угроз**:
   Система использует заранее настроенные правила, алгоритмы и искусственный интеллект для выявления аномалий, атак или других подозрительных действий. Например, если несанкционированный доступ пытается войти в систему с множества IP-адресов за короткий промежуток времени, SIEM может идентифицировать это как потенциальную атаку.
5. **Мониторинг в реальном времени**:
   SIEM позволяет наблюдать за событиями в режиме реального времени, предоставляя оперативные данные о безопасности. Это особенно важно для быстрого реагирования на инциденты, такие как взломы, вирусные атаки или утечки данных.
6. **Анализ инцидентов и расследования**:
   Когда происходит инцидент безопасности, SIEM помогает собрать информацию о событиях, связанных с этим инцидентом, и предоставляет данные для расследования. Это может включать в себя хронологию событий, детали источников атак, а также возможные уязвимости.
7. **Отчётность и соответствие требованиям**:
   SIEM помогает организациям собирать данные для отчетности и соответствия различным стандартам и нормативам (например, GDPR, HIPAA, PCI DSS). Это включает в себя автоматическую генерацию отчётов и мониторинг на предмет соблюдения внутренних политик безопасности.
8. **Хранение и архивация данных**:
   SIEM сохраняет логи и события для их последующего анализа и хранения в

архиве. Важно, чтобы данные были доступны для расследования в случае необходимости, а также для выполнения требований по длительности хранения.

Преимущества использования SIEM

- **Раннее обнаружение угроз**:
  SIEM помогает обнаруживать угрозы на ранних стадиях, что позволяет предотвратить или минимизировать ущерб от кибератак.
- **Централизованное управление**:
  Система собирает данные из различных источников и предоставляет единую точку для их анализа, что упрощает управление безопасностью.
- **Автоматизация реагирования**:
  Многие SIEM-системы поддерживают автоматическое реагирование на угрозы, например, блокировку IP-адреса, отправку уведомлений или даже выполнение скриптов для предотвращения атак.
- **Улучшение расследования инцидентов**:
  SIEM помогает следить за событиями и предоставляет подробную информацию, что ускоряет расследования и анализ инцидентов.
- **Снижение рисков и улучшение соблюдения стандартов**:
  Благодаря возможности анализа и отчетности, SIEM помогает организациям соответствовать требованиям нормативных актов и снижать риски, связанные с безопасностью данных.

Примеры популярных SIEM-систем

1. **Splunk**:
   Одна из самых известных и мощных SIEM-платформ, которая позволяет собирать и анализировать большие объемы данных. Splunk используется для мониторинга безопасности, анализа журналов и построения отчетности.
2. **IBM QRadar**:
   QRadar — это ещё одна популярная SIEM-система, которая предлагает анализ событий безопасности в реальном времени, корреляцию данных и мониторинг инцидентов.
3. **ArcSight (Micro Focus)**:
   Это решение помогает интегрировать различные источники данных безопасности и предоставляет возможности для глубокого анализа и расследования инцидентов.
4. **LogRhythm**:
   Система SIEM с фокусом на централизованном управлении журналами, автоматическом реагировании и обнаружении аномалий.
5. **AlienVault (AT&T Cybersecurity)**:
   AlienVault использует встроенные алгоритмы для обнаружения угроз и поддерживает интеграцию с различными внешними источниками данных для повышения уровня безопасности.

SIEM играет важную роль в современной кибербезопасности, помогая организациям выявлять угрозы в реальном времени, управлять рисками и обеспечивать соблюдение нормативных стандартов. Он предоставляет мощные средства для анализа и корреляции данных, что делает его важным инструментом для защиты от кибератак и эффективного реагирования на инциденты безопасности.