Teach Me Skills

Защита инфраструктуры приложений

Часть 2

Teach Me Skills

Вопросы по предыдущим темам или ДЗ

Mini-quize по прошлым темам:

1. **Как осуществляется безопасность облачных сред?**
2. **Какой самый распространенный формат хранения в Docker?**
3. **Как вы бы организовали backup серверов в компании?**
4. **Как можно настроить процесс инвентаризации активов?**
5. **С чего вы начнете, для внедрения процесса управления уязвимостями?**
6. **Чем Windows отличается от Linux?**
7. **Как в компаниях хранят электронные подписи?**

Mini-quize по новой теме:

1. **Что такое HoneyPot и Deception?**
2. **Для чего предназначен SandBox?**
3. **Есть ли бесплатный SandBox и где его можно найти?**
4. **Для чего мультифакторная авторизация и как она работает?**
5. **Как работает Dynamic access control в Windows?**

Teach Me Skills

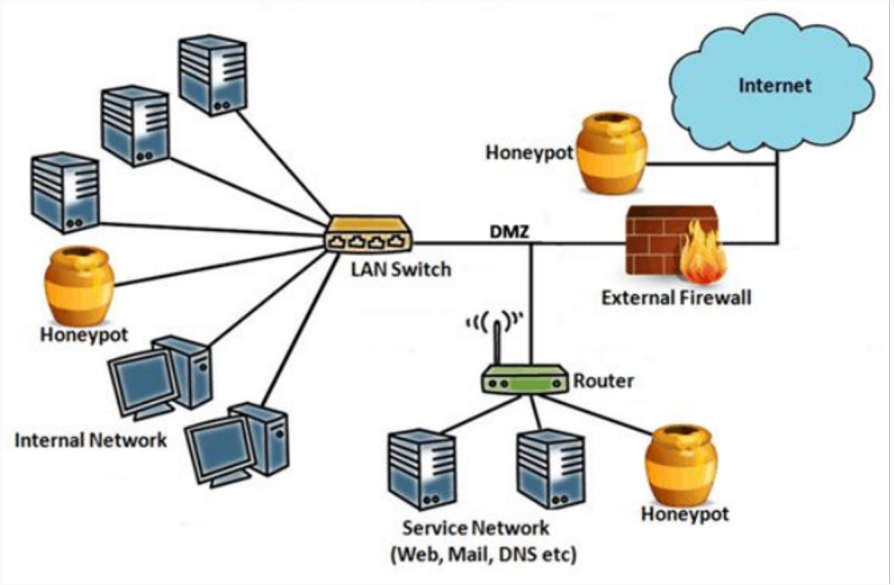
План занятия

1. Динамический контроль доступа
2. Honeypot
3. Песочницы
4. Практика включения Windows Sandbox
5. Мультифакторная аутентификация, разновидности

Защита инфраструктуры приложений

Honeypot

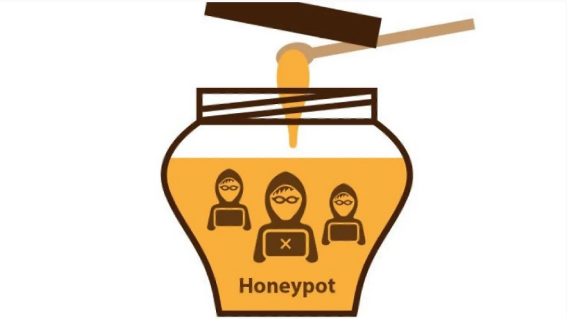
Honeypot (приманка, ловушка) - оборудование или ПО, которые специально развертываются отделами ИБ и служит приманкой для сбора информации о злоумышленнике и защиты реальной целевой системы.



Защита инфраструктуры приложений

Honeypot - типы

- Полная эмуляция системы (Pure honeypot)
- Высокоинтерактивные (High-interaction honeypot)
- Низкоинтерактивные (Low-interaction honeypot)

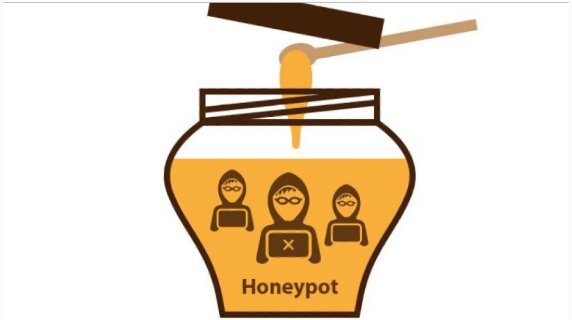


Защита инфраструктуры приложений

Honeypot - типы

- Полная эмуляция системы (Pure honeypot)

Представляет собой подлинно уязвимое ПО, которое обычно имеется в производственной системе, где взаимодействует с различными приложениями. Такие ханипоты сложнее поддерживать, но они предоставляют более полезную информацию.

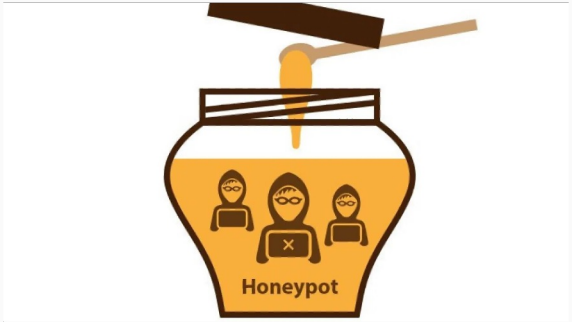


Защита инфраструктуры приложений

Honeypot - типы

- Высокоинтерактивные (High-interaction honeypot)

Основаны на имитации операционных систем реального времени и имеют все свои приложения и сервисы, как и у целевой сети. Обычно собирают больше информации, поскольку их цель — остановить злоумышленника, что дает больше времени для адекватной реакции на угрозу.

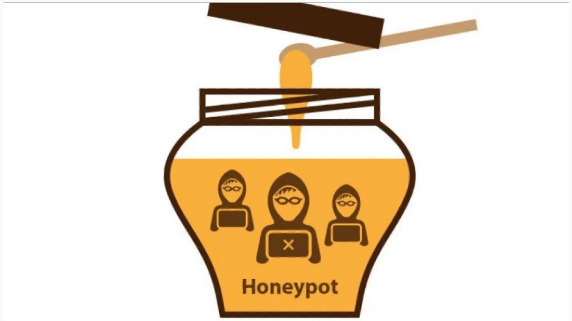


Защита инфраструктуры приложений

Honeypot - типы

- Низкоинтерактивные (Low-interaction honeypot)

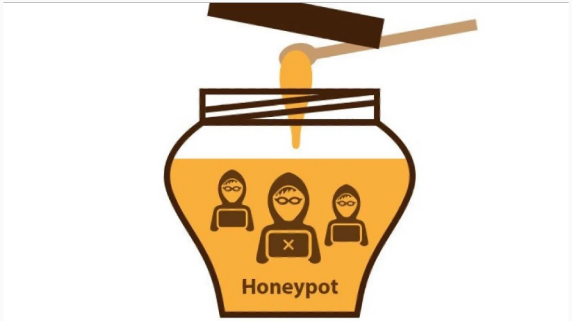
Соответствуют очень ограниченному количеству сервисов и приложений, как в системе, так и в сети. Этот тип приманки можно использовать для отслеживания портов и служб



Защита инфраструктуры приложений

Honeypot недостатки

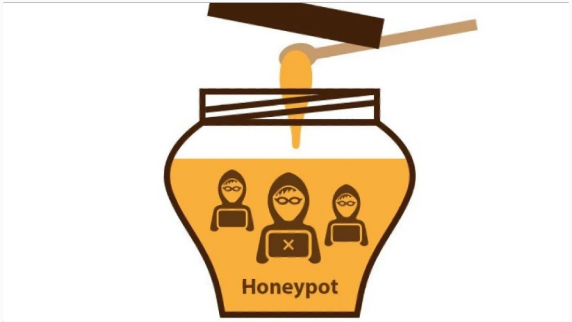
- Если ловушка плохо настроена, её легко распознать.
- Ловушка может обнаружить только атаку на саму ловушку.
- Если ловушка распознана, на неё можно совершить “атаку”, чтобы отвлечь от реальных действий в системе.
- Если ловушка содержит уязвимости, через неё можно атаковать систему.



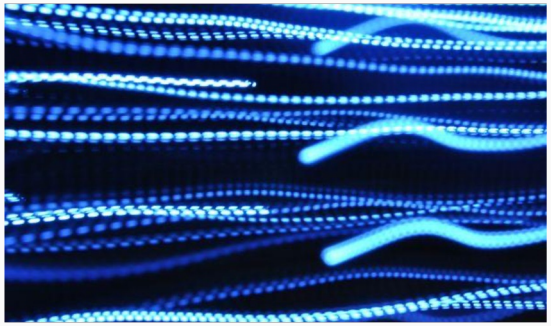
Защита инфраструктуры приложений

Honeypot

[Бесплатные приманки для ловли хакеров]



Защита инфраструктуры приложений



Deception Technology или **Технология обмана** – это усовершенствованный тип кибербезопасности, следующая ступень развития имитирующих систем HoneyPot.

Данная технология использует тактику обмана – от поддельных сетевых сред до искусственных учетных данных. С целью поймать злоумышленников и узнать побольше об их методиках и действиях.

В отличие от традиционной инфраструктуры безопасности, например брандмауэров, технология обмана не стремится защищать просто периметр. Система Deception обнаруживает любую незаконную деятельность, даже если она происходит внутри организации.

Защита инфраструктуры приложений

Honeypot или Deception

HoneyPot имитирует, будто в системах компании есть точки входа, не защищенные должным образом, тем самым привлекая злоумышленников. Благодаря приманке, компания предотвращает атаки и может отследить и проанализировать все действия хакеров. Это потенциально полезная стратегия, прежде всего для крупных компаний, которые часто располагают большим количеством конфиденциальных данных и являются привлекательной целью для злоумышленников.

Система Deception обнаруживает злоумышленников, когда они делают свой первый шаг в сети организации. Киберпреступники используют определенные схемы после получения доступа к сети. Больше они не могут свободно перемещаться по сети и возвращаться несколько раз с использованием эксплойтов.

Защита инфраструктуры приложений

Преимущества Deception

Адаптируемость

Адаптируемость для работы во многих типах систем, в том числе устаревших и стабильную работу сети без нарушений в сетевых системах и бизнес-процессах.

Масштабируемость

Масштабируемость для достижения максимальной эффективности при росте и дальнейшей автоматизации процессов.

Обнаружение

Обнаружение злоумышленников до взлома с помощью обмана вы можете найти злоумышленников и наблюдать за их перемещениями еще до того, как они войдут в вашу сеть.

Получение данных

Получение данных об угрозах в режиме реального времени создается информация об угрозах, привлекая злоумышленников и доставляя ее вашей группе безопасности в режиме реального времени.

Решения Deception и HoneyPot

DejaVu

TrapX Security

Решение DeceptionGrid обнаруживает и предотвращает атаки в реальном времени. Используется действенный интеллект для создания «минного поля» из ловушек и предупреждения вас о подозрительной деятельности.

Xello

Российский разработчик программного обеспечения (ПО) в сфере информационной безопасности. Недавно компания объявила о внесении их платформы «Xello-Deception» в Единый реестр российских программ ЭВМ и баз данных Министерства цифрового развития, связи и коммуникации РФ.

Open Source Deception Platform

Защита инфраструктуры приложений

Sandbox (Песочница) — это среда безопасного тестирования. Решение изолирует непроверенные изменения в коде и эксперименты от производственной среды и хранилища в контексте разработки программного обеспечения. Включая веб-разработку и контроль версий.

Песочница обнаруживает угрозы в файлах, передаваемых по сети (почтовые сообщения, загрузка файлов из Интернет и т. д.) с помощью продвинутых технологий поведенческого анализа. Система помогает обнаруживать и предотвращать APT-угрозы до их проникновения на конкретный хост. Далее мы разберем как работает система, как запускать «sandboxie», что позволяет делать песочница и какое решение выбрать.

Что можно сделать в среде песочницы?

1. Запустить код и оценивать его на основе деятельности, а не атрибутов.
2. Запустить исполняемые файлы и другие скрытые вредоносные программы.
3. Разрешить и наблюдать за сетевым трафиком.
4. Безопасно выполнять вредоносный код или операции с диском.
5. Безопасно изменять реестры / систему / конфигурацию и т. п.

Как работает Sandbox?

Эмуляция реального устройства

Песочница имитирует физическое оборудование, обеспечивая глубокое представление о поведении и воздействии программы. Приложение для теста имеет доступ к тем же ресурсам, что и анализируемый код, включая ЦП, память и хранилище.

Эмуляция операционных систем

Имитация операционной системы (ОС) конечного пользователя, но не аппаратного обеспечения машины. В случае виртуальной машины песочница изолирована от базового физического оборудования, но имеет доступ к установленной ОС.

Виртуализация сред

Использование изолированной программной среды на основе виртуальной машины для хранения и проверки подозрительных программ. Не имеет доступа к физическим ресурсам, но может получить доступ к виртуализированному оборудованию.

Защита инфраструктуры приложений

Три модели интеграции SandBox в инфраструктуру компании

1. **Самостоятельно**. Купить готовый продукт и своими силами интегрировать его в сетевую инфраструктуру. Такой подход актуален для крупных IT-ориентированных компаний, у которых есть достаточное количество специалистов соответствующей компетенции.
2. **Опосредованно**. От предыдущей модели этот вариант отличается тем, что интеграцией и настройкой занимаются внешние специалисты, представители поставщика ПО или другая компетентная компания. Такой метод, с позиции бизнеса, проще, но не решает вопрос того, кто будет работать с песочницей «на местах».
3. **Делегировано**. То есть по SaaS-модели, где компания оплачивает подписку, а все операции с SandBox удаленно выполняет поставщик. Такой подход позволяет оптимизировать расходы (нет необходимости «разово» платить большую сумму за приобретение самой программы песочницы) и снижает нагрузку на штат ИБ компании.

Защита инфраструктуры приложений

Практика SandBox

ссылка для проверки

в ANY.RUN

в Triage

https://clck.ru/JhxDz/

Для работы в ней необходима корпоративная почта

Защита инфраструктуры приложений

Лучшие решения Sandbox

Kaspersky Sandbox

Автоматизированная защита от продвинутых угроз. В решение входит динамическое обнаружение, простая управляемость и масштабируемость. Также имеется возможность интеграции с другими решениями Kaspersky Security.

Kaspersky Anti Targeted Attack

Решение класса XDR с надежной защитой от сложных кибератак. Автоматическое реагирование на сложные угрозы, сбор и хранение данных, всесторонний анализ и своевременное обнаружение. Расширенные возможности защиты от сложнейших

Positive Technologies Sandbox

Первая песочница, которая защищает именно вашу инфраструктуру. Гибкая настройка виртуальных сред. Ретроспективный анализ. Поддерживает гибкую удобную кастомизацию виртуальных сред для анализа и обнаруживает угрозы не только в файлах, но

Check Point Sandbox

Технология Check Point SandBlast Zero-Day Protection для предотвращения ранее неизвестных и целевых атак. SandBlast Threat Emulation — компонент, являющийся новым видом организации «песочницы» от Check Point (Check Point Sandbox).

Лучшие решения Sandbox

Open Source Projects

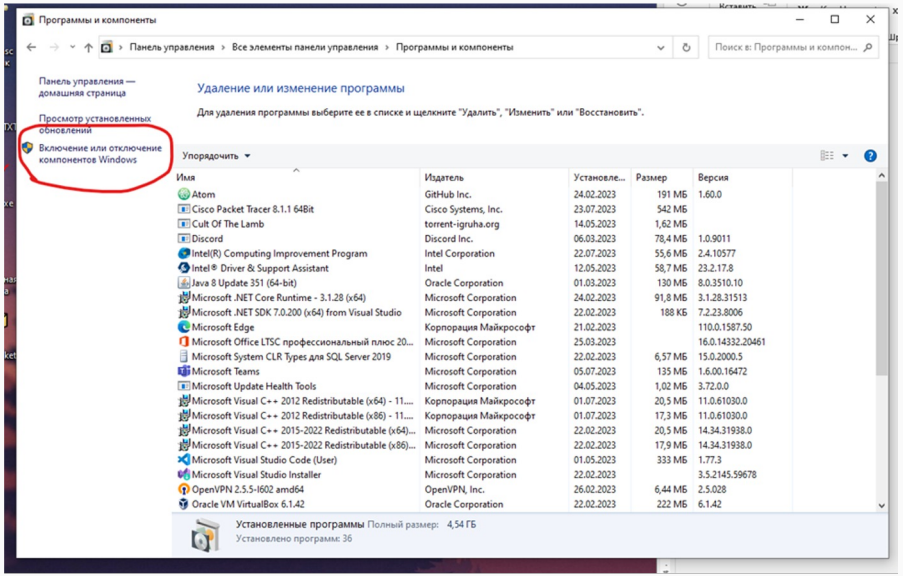
Установка Cuckoo

Установка SandboxIE

Как включить и использовать песочницу Windows Sandbox в Windows 10

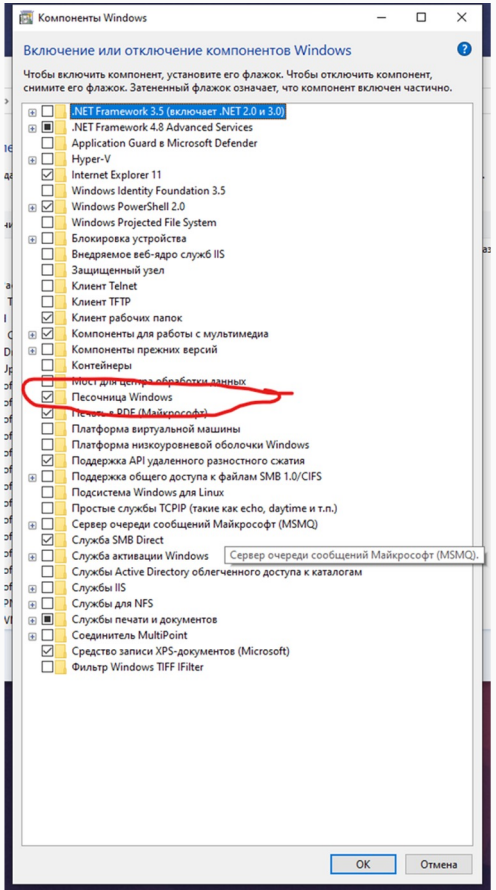
1. Проверьте требования:
- Ваша версия должна быть Windows 10 Pro, Enterprise или Education. К сожалению, эта функция недоступна в домашних версиях Windows 10.

2. Включите песочницу Windows Sandbox:
Шаг 1: В "Панели управления" выберите "Программы и компоненты" и затем "Включение или отключение компонентов Windows".



Как включить и использовать песочницу Windows Sandbox в Windows 10

Шаг 3: Прокрутите вниз и найдите "Песочница Windows", убедитесь, что он отмечен, и нажмите "OK". Это может потребовать перезагрузки системы.

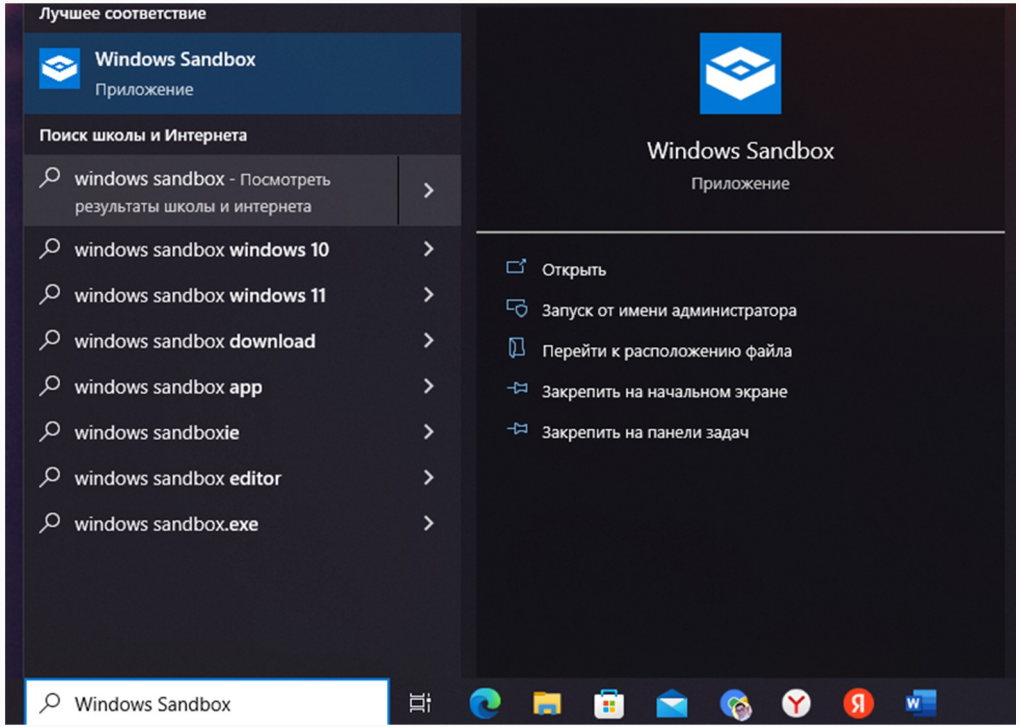


Как включить и использовать песочницу Windows Sandbox в Windows 10

3. Запустите Windows Sandbox:

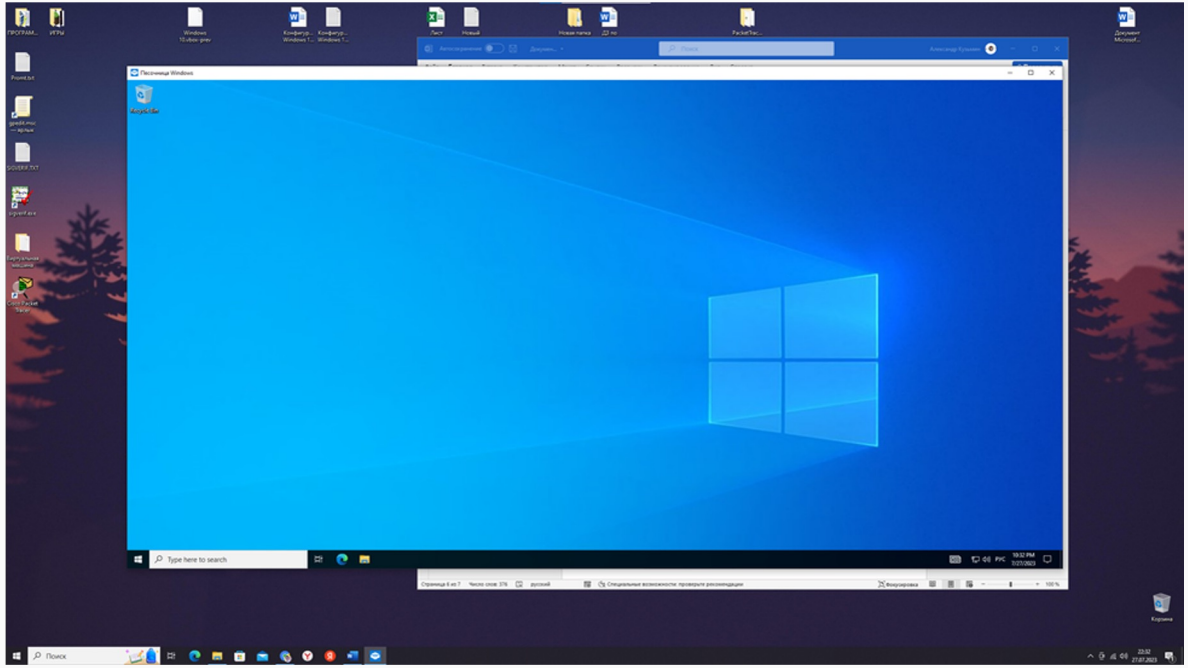
Шаг 1: Нажмите левой кнопкой мыши на кнопке "Поиск".

Шаг 2: Введите "Windows Sandbox" и запустите приложение.



Как включить и использовать песочницу Windows Sandbox в Windows 10

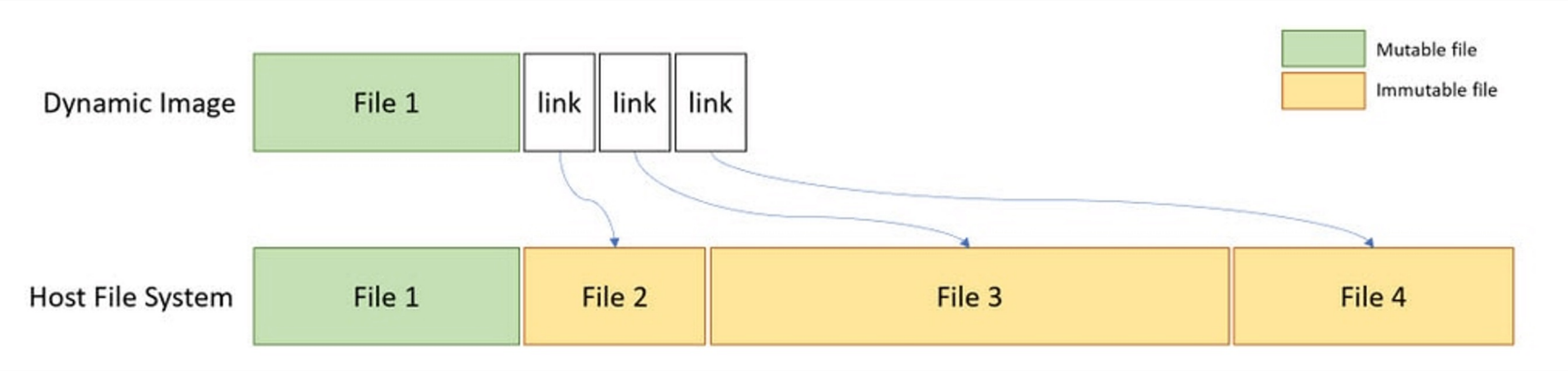
Шаг 3: В открывшемся окне Windows Sandbox вам будет предложено подождать несколько мгновений, пока будет создана изолированная среда. После завершения этого процесса вы увидите окно, похожее на обычную копию Windows.



Защита инфраструктуры приложений

Windows Sandbox - это более легкая версия Hyper-V. Поскольку гипервизор работает под управлением ОС следовательно, Sandbox требует наличия собственной ОС для запуска и выполнения различных задач. Ключевое преимущество использования Windows Sandbox по сравнению с виртуальной машиной заключается в том, что новая копия ОС запускается каждый раз при открытии Песочницы.

Копия образа Windows 10 сохраняется как «Базовый динамический образ» и используется, когда включена функция Windows Sandbox.



Защита инфраструктуры приложений

Мультифакторная аутентификация (MFA) — это мера безопасности, в рамках которой пользователям требуется проходить более одной аутентификации для доступа к службе или приложению.

3 фактора аутентификации

При использовании многофакторной аутентификации может быть задействован целый ряд различных способов проверки пользователя на достоверность. Все они делятся на три категории:

1. Фактор знания.
2. Фактор владения.
3. Фактор свойства.

Фактор знания

С его помощью пользователь может подтвердить, что он владеет каким-либо знанием, ранее заведенным в систему. Наибольшее распространение получили, собственно, пароли, а также кодовые слова, ответы на секретные вопросы, личные идентификационные номера PIN для банковских карт и приложений, Seed-фразы для криптокошельков и т.д.

Эти варианты защиты просты в реализации, но считаются наименее эффективными с точки зрения реальной защиты. Многие из них достаточно легко узнаются или вычисляются. Кроме того, невозможно отследить, чтобы для разных сервисов были придуманы разные пароли/ответы.

Фактор владения

Идентификация человека по фактору владения предполагает наличие у него некоторого предмета, который подтверждает право этого человека на доступ. Таким предметом может быть ключ, токен, чип, печать и т.д.

Современный способ реализации фактора владения — подтверждение входа на одном устройстве с помощью другого устройства, например, мобильного телефона. Тем самым проверяется фактор владения другим устройством с выполненным входом.

Реализация аутентификации с использованием фактора владения сложнее, чем в случае фактора знания, но и злоумышленникам получить доступ к такой системе гораздо сложнее.

Фактор свойства

Самым распространенным примером фактора свойства являются биометрические данные человека. Они неповторимы, а современные устройства научились считывать и анализировать их за доли секунды. Именно на этом принципе основана разблокировка телефона по отпечатку пальца или лицу, вход в метро по биометрике и другие решения.

Такие системы наиболее сложны в реализации и связаны с хранением персональных данных, но обладают наибольшей защищенностью, особенно вкупе с другими средствами аутентификации.

Каждый из указанных факторов хоть и обеспечивает определенный уровень защиты, но сильно уступает варианту с использованием комбинации этих факторов.

Примеры многофакторной аутентификации

Существует множество различных методов проверки в рамках четырех вышеперечисленных категорий, но здесь приведены наиболее распространенные методы, с которыми обычный пользователь может столкнуться в своей цифровой жизни.

1. Одноразовый пароль на основе времени (TOTP)

TOTP — это код, обычно 6-значный номер, который действителен только в течение короткого периода времени — часто от тридцати до шестидесяти секунд. С помощью этого метода пользователь может использовать менеджер паролей, который хранит коды TOTP, или скачать приложение для аутентификации для хранения и доступа к этим кодам. После ввода пароля для входа в учетную запись пользователю будет предложено ввести код для подтверждения личности.

Это одна из самых надежных форм MFA, поскольку коды защищены и их трудно перехватить. Единственный способ, которым злоумышленник может украсть код — это скомпрометировать устройство, на котором генерируется код, украсть его или заразить вредоносным ПО.

2. Токен текстовых сообщений SMS

Этот метод требует от пользователя ввести свой номер телефона при создании учетной записи. Когда пользователь входит в систему с помощью учетных данных, ему будет предложено ввести код, отправленный на телефон с помощью SMS. С помощью кода пользователи могут войти в систему.

3. Токен электронной почты

4. Аппаратный ключ безопасности

Аппаратный ключ безопасности — это физический токен. После подключения к учетным записям храните его в надежном месте, где вы не сможете его потерять. Когда вы входите в свою учетную запись, вы обычно вставляете ключ в USB-порт или касаетесь им своего устройства. Ваше устройство почувствует ключ и подтвердит вашу личность.

5. Биометрическая аутентификация

Биометрическая аутентификация проверяет вашу личность с помощью распознавания лиц, сканирования отпечатков пальцев или сканирования радужной оболочки глаза. При первой настройке биометрической аутентификации пользователь регистрирует на устройстве свой отпечаток пальца или скан лица. Затем система сравнит будущие снимки с первым, чтобы подтвердить вашу личность.

6. Секретные вопросы

Секретные вопросы часто используются для устного подтверждения вашей личности, например, при телефонном разговоре с банком, но они применяются и в цифровом виде.

Внедрение MFA в Ubuntu

https://www.linuxbabe.com/ubuntu/two-factor-authentication-ssh-key-ubuntu

Защита инфраструктуры приложений

Обзор 2FA приложений

Критерий / приложение	Twilio Authy 2-Factor Authentication	Duo Mobile	ESET Secure Authentication	Gemalto SafeNet MobilePASS+	Multifactor	Protectimus SMART	RSA SecurID Software Token
Архитектура	Облачная	Облачная	On-premise	Облачная и on-premise	Облачная	Облачная и on-premise	Облачная и on-premise
Поддерживаемые платформы	iOS, Android, macOS, Windows, Linux	iOS, Android	iOS, Android	iOS (10 или выше), Android (4.4 или выше), Windows 10	iOS, Android	iOS, Android	iOS, Android, Windows, BlackBerry OS, macOS
Офлайн-режим	Да	Да	Да	Да	Да	Да	Нет
Пуш-уведомления	Да	Да	Да	Да	Да	Да	Да
Резервное копирование учётных записей в облако	Да	Да	Нет (данные ESA автоматически добавляются в резервные копии Active Directory)	Нет	Да	В разработке	Да
Язык интерфейса	Английский	Английский	Английский	Английский	Русский	Русский, английский	Английский
Защита приложения	PIN-код, отпечаток пальца	PIN-код, отпечаток пальца	PIN-код, отпечаток пальца	PIN-код, отпечаток пальца, FaceID	PIN-код и Биометрия	PIN-код, отпечаток пальца (в разработке)	PIN-код, отпечаток пальца

Защита инфраструктуры приложений

Динамический контроль доступа

Dynamic access control – инновационный механизм управления доступом, предназначенный для Windows Server. Он используется для повышения безопасности и эффективности управления доступом к данным, основываясь на конкретных параметрах пользователей и ресурсов.

Dynamic access control основан на применении условных правил для управления доступом к отдельным ресурсам. Например, данная технология позволяет настроить права доступа на основе пользовательской роли, времени доступа и географического положения пользователя.

Защита инфраструктуры приложений

Динамический контроль доступа

Одной из главных преимуществ dynamic access control является его способность автоматически настраивать права доступа на основе изменения параметров пользователей и ресурсов. Также он обеспечивает более точное управление доступом к данным и уменьшает риски утечки конфиденциальной информации.

Для использования dynamic access control необходимо настроить политики управления доступом (GPO) и использовать файловую службу (File Server Resource Manager) для создания и настройки условных правил.

Динамический контроль доступа

Преимущества dynamic access control

- **Управление доступом на основе контекста**

Dynamic access control позволяет принимать решения об управлении доступом на основе контекста, что означает, что доступ к ресурсам может быть ограничен в зависимости от места, времени и других факторов. Например, вы можете настроить правила доступа для пользователей, имеющих определенный IP-адрес или доступ к ресурсам только в определенное время дня.

- **Безопасность данных**

Dynamic access control позволяет создавать политики безопасности, которые могут гарантировать, что конфиденциальная информация и данные будут защищены, даже если они попадут в неправильные руки. Например, вы можете защитить конфиденциальные документы таким образом, чтобы только определенные пользователи могли иметь доступ к этим документам.

Динамический контроль доступа

Преимущества dynamic access control

- **Упрощение управления доступом**

Dynamic access control упрощает управление доступом к ресурсам, поскольку вы можете создавать правила управления доступом на основе атрибутов без необходимости создавать множество групп для разных пользователей и ресурсов. Это позволяет значительно сократить количество необходимых групп и упростить процесс администрирования.

- **Контроль доступа на уровне файлов**

Dynamic access control позволяет настраивать правила доступа на уровне файлов, что означает, что вы можете управлять доступом к отдельным файлам и папкам, а не только к ресурсам в целом. Это упрощает процесс управления доступом и обеспечивает более гибкую защиту данных.

Защита инфраструктуры приложений

Рекомендации по использованию dynamic access control

1. Планируйте заранее.
2. Не используйте слишком много правил.
3. Проверяйте свойства файлов и папок.
4. Используйте резервные копии.
5. Обучайте пользователей.

Teach
Me
Skills

Спасибо за внимание!