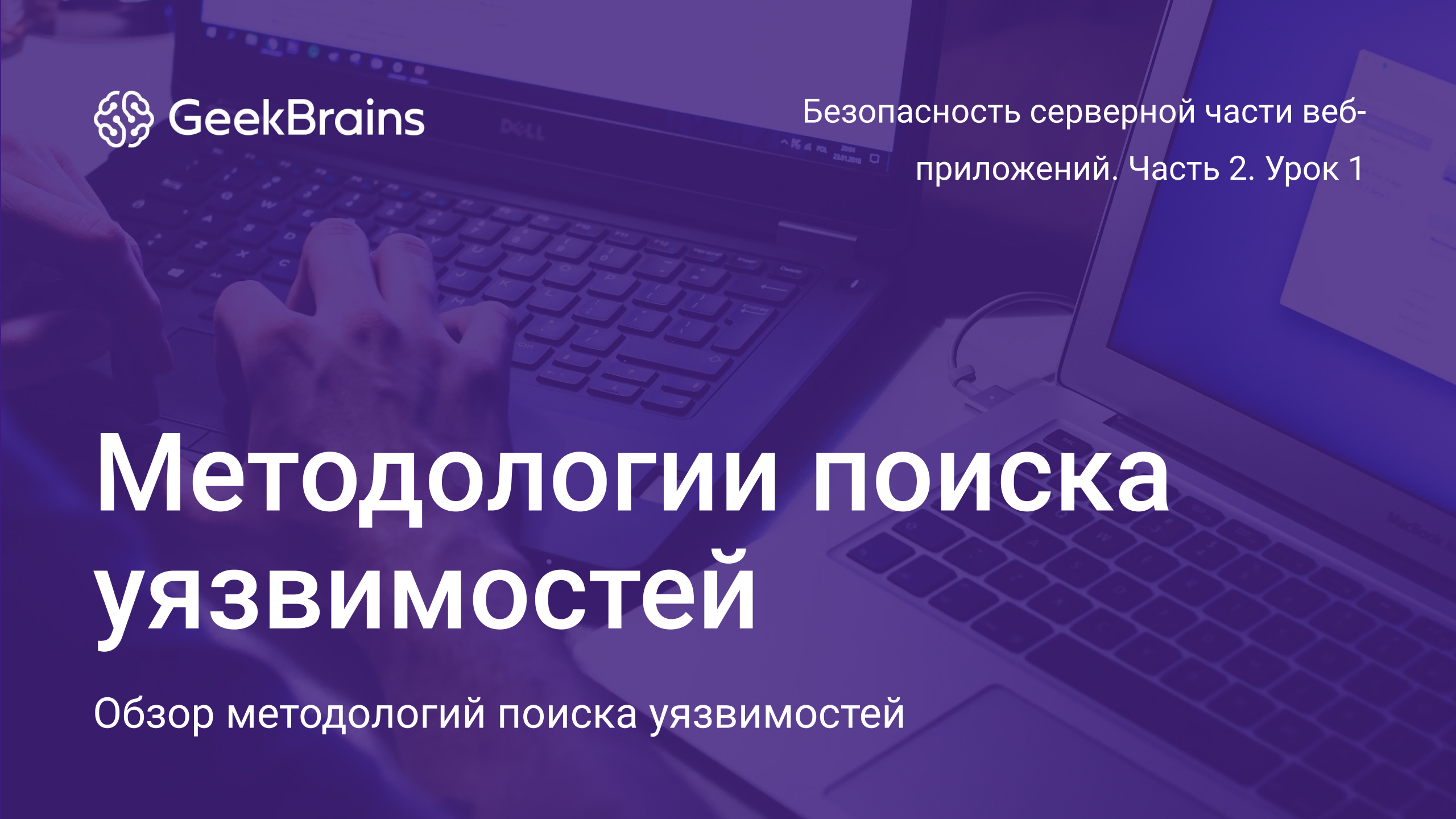


GeekBrains

Безопасность серверной части веб-приложений. Часть 2. Урок 1

Методологии поиска уязвимостей

Обзор методологий поиска уязвимостей

Регламент курса

1. 8 уроков по 2 часа
2. Практические задания
3. Виртуальная машина для практики
4. Видеозаписи уроков будут выкладываться
5. Задавайте вопросы.

Что будем изучать на курсе?

- Мы будем рассматривать уязвимости веб-серверов и методы их обнаружения.
- Научимся проводить разведку ресурсов веб-сервера.
- Изучим особенности таких процедур, как Pentest, BugBounty, CTF.

Что получим по окончании курса?

1. Изучим методы сбора информации о сервере.
2. Научимся находить уязвимости серверной части веб-приложения.
3. Научимся настраивать защиту от атак, использующих уязвимости серверной части веб-приложения.

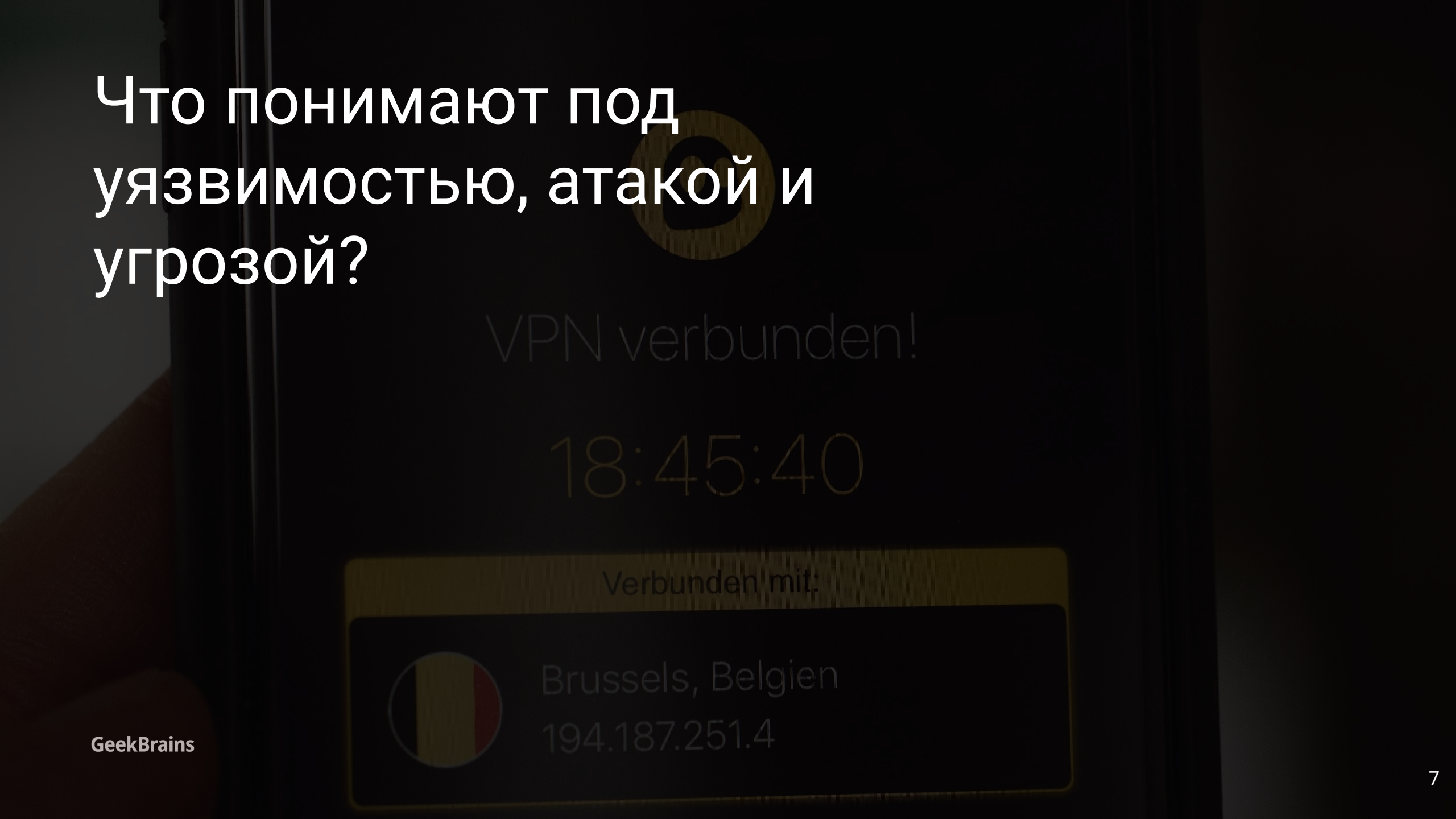
Чем мы будем пользоваться

1. Kali Linux
2. VirtualBox
3. BM metasploitable 3 на основе Ubuntu 14.
4. Mutillidae, Dvwa, Bwapp, XVWA в составе BM.

План урока

1. Что понимают под уязвимостью, атакой и угрозой?
2. Оценка опасности уязвимости.
3. Практическая часть.

Что понимают под уязвимостью, атакой и угрозой?



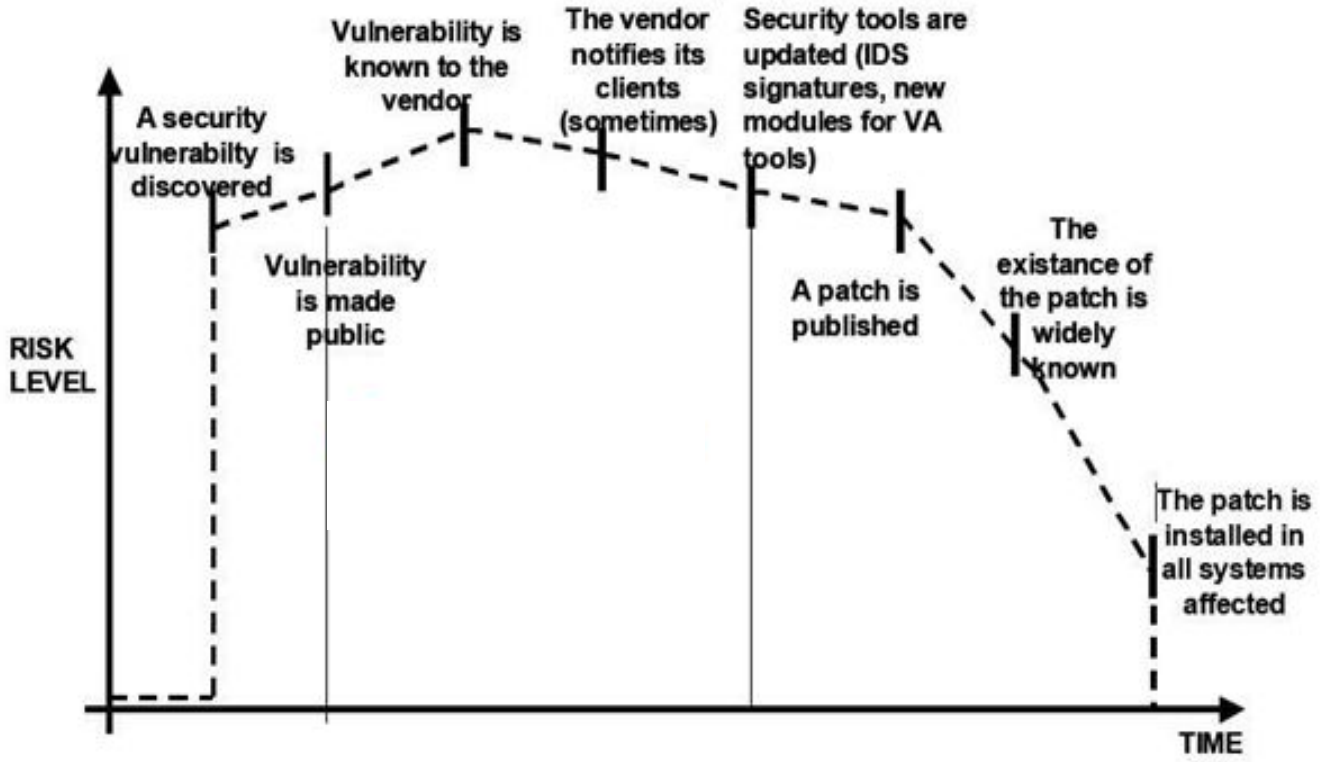
Что понимают под уязвимостью

Уязвимость — недостаток в системе, использование которого может привести к нарушениям в ее работе.

Уязвимость компонента системы — недостаток в компоненте, использование которого может привести к нарушениям в его работе.

Обычно эксплуатация **уязвимости** «приводит к нарушению конфиденциальности, целостности и доступности информации».

Окно уязвимости



Особенности жизненного цикла уязвимости

1. Уязвимость не несет в себе опасности, пока злоумышленник не подберет для ее эксплуатации средства.

2. Окно уязвимости — временной промежуток между событием опубликования уязвимости и событиями выхода патчей (или иного механизма), сигнатур, модулей и других сущностей для средств защиты. Наиболее опасное время жизненного цикла уязвимости.

Особенности жизненного цикла уязвимости

3. Основная задача — сократить это окно, тем самым снизив вероятность реализации уязвимости.

Типичные причины появления уязвимости

1. Ошибки в реализации компонентов.
2. Некорректная обработка передаваемых в приложение данных.
3. Ошибки во взаимодействии компонентов. Типичный пример — некорректные запросы к БД.
4. Использование заведомо «слабых» сущностей для идентификации и аутентификации.

Оценка опасности уязвимости

1. Почти всегда проводится по последствиям эксплуатации уязвимости.

2. При поиске уязвимости важно выяснить границы реализации уязвимости — какие угрозы можно реализовать при помощи найденной уязвимости.

Оценка опасности уязвимости

3. Универсальный способ оценки — вектор CVSS.

4. Но на практике последствия от эксплуатации уязвимости надо оценивать для конкретной системы — так как злоумышленник стремится на практике эксплуатировать уязвимость.

Какие можно выделить типы уязвимостей?

По принципу эксплуатации:

- Критические
- Опасные
- Неопасные

По наличию для них эксплоита:

- Эксплоит есть
- Эксплоита нет

По компоненту, в котором они встречаются:

- Уязвимости в операционной системе
- Уязвимости компонентов и приложений

По направлению атаки:

- Уязвимости Server Side
- Уязвимости Client Side

С точки зрения обнаружения:

- Уязвимости, обнаруженные в процессе тестирования
- Уязвимости, обнаруженные «in-the wild»

С точки зрения защиты:

- Для уязвимости был выпущен патч
- Для уязвимости нет патча

Краткие выводы

При поиске уязвимости важно выяснить границы ее реализации:

1. Какие дополнительные возможности нужны для эксплуатации уязвимости?

2. Какие угрозы можно реализовать при помощи найденной уязвимости?

3. Можно ли развить уязвимость до более опасной?

Что понимают под угрозой

1. Под угрозой понимаются действия, которые приводят к нарушению информационной безопасности.

2. Угроза обычно носит вероятностный характер и связана с уязвимостями.

2. Это может быть одна или несколько уязвимостей, которые будут повышать вероятность реализации угрозы.

Кратко об оценке угроз

1. Выяснить границы реализации уязвимости = узнать, какие угрозы можно реализовать при помощи найденной уязвимости.
2. Угрозы обычно реализуют злоумышленники (хотя могут и пользователи неумышленно).

Кратко об оценке угроз

3. С точки зрения веб-безопасности угрозы, как правило, являются внешними — их реализуют злоумышленники, которые не имеют непосредственного доступа к внутренней структуре веб-сервера.

4. Именно угрозы проверяются на практике при тестировании на проникновение.

Кратко об атаках

1. Под атакой обычно понимают практическую реализацию угрозы посредством эксплуатации уязвимости.

2. Атака может быть связана с эксплуатацией нескольких угроз.

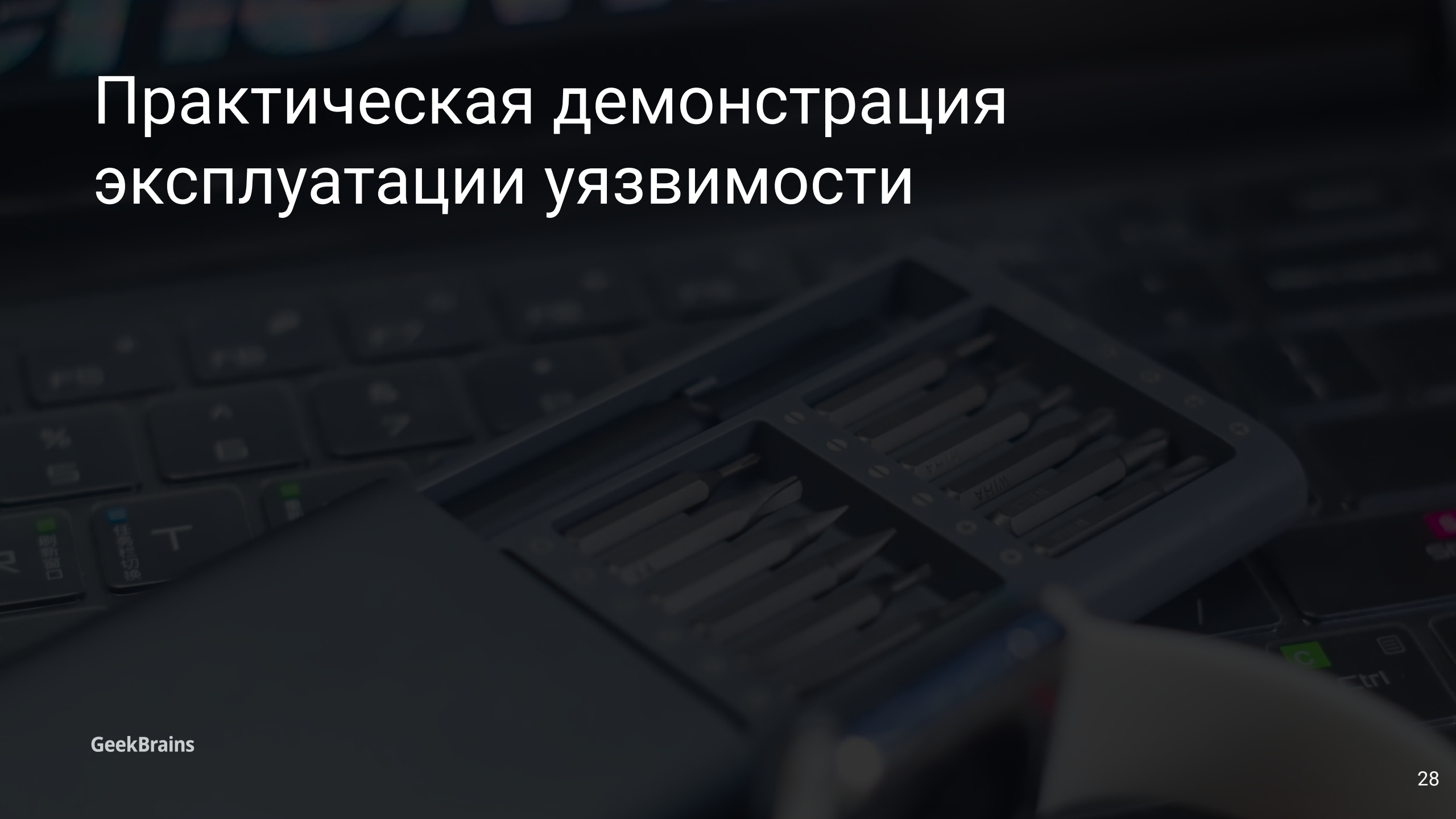
3. Способ реализации атаки называют ее вектором.

Кратко об атаках

4. Полезная часть вектора атаки (та, что нужна злоумышленнику для его целей) называется полезной нагрузкой (Payload).

5. Если вектор атаки опубликован в открытом доступе, опасность атаки возрастает.

Практическая демонстрация эксплуатации уязвимости



Методологии поиска и оценки уязвимостей

Способы тестирования

Black box testing. Тестировщик ничего не знает об устройстве или функционирования приложения.

White box testing. В рамках такого тестирования у тестировщика есть знания о том, как работает приложение.

Grey box testing. Это сочетание первых двух техник, призванное компенсировать их недостатки.

Пример Black Box теста

1. **Запрос**
   nc 192.168.56.103 80
   GET / HTTP/1.0

2. **Ответ** HTTP/1.1 400 Bad Request
   Date: Sun, 23 Dec 2018 18:56:31 GMT
   Server: **Apache/2.4.10 (Debian)**
   Content-Length: 301
   Connection: close
   Content-Type: text/html; charset=iso-8859-1

Типичный сценарий поиска уязвимости

Проанализировать логику работы приложения	Black Box
Протестировать работу приложения заведомо неправильными данными	Black Box/Grey box
Проверить, есть ли такие данные, передав которые приложение будет вести себя некорректно	Black Box
Реализация атаки с использованием полученных данных	Grey box
Можно ли «раскрутить» уязвимость до более опасной?	Grey box

Классификация OWASP Top 10

- A1 Внедрение кода.
- A2 Некорректная аутентификация и управление сессией.
- A3 Утечка чувствительных данных.
- A4 Внедрение внешних XML-сущностей (XXE).
- A5 Нарушение контроля доступа.

Классификация OWASP Top 10

- A6 Небезопасная конфигурация.
- A7 Межсайтовый скриптинг.
- A8 Небезопасная десериализация.
- A9 Использование компонентов с известными уязвимостями.
- A10 Отсутствие журналирования и мониторинга.

Особенности OWASP Top 10

1. Часто уязвимости связаны между собой. Например, инъекция (A7) вполне может приводить к утечке данных (A3).

2. Некоторые способы эксплуатации уязвимостей будут подразумевать наличие других уязвимостей. Например, наличие уязвимости A9 может приводить к возникновению многих других уязвимостей. Пример — уязвимости в плагинах для серверных фреймворков.

Особенности OWASP Top 10

3. С данной классификацией тесно связано методология тестирования OWASP Testing Guide.

4. Information Gathering (сбор информации).

5. Configuration and Deployment Management Testing (тестирование конфигурации).

Особенности OWASP Top 10

6. Identity Management Testing (тестирование управлением идентификацией).
7. Authentication Testing (тестирование аутентификационных механизмов).
8. Authorization Testing (тестирование механизмов авторизации).
9. Session Management Testing (тестирование механизмов управления сессиями).

Особенности OWASP Top 10

10. Identity Management Testing (тестирование управлением идентификацией).
11. Testing for Error Handling (тестирование обработки ошибок).
12. Testing for weak Cryptography (оценка слабости криптографических механизмов).
13. Business Logic Testing (тестирование бизнес-логики).

Методы поиска уязвимостей

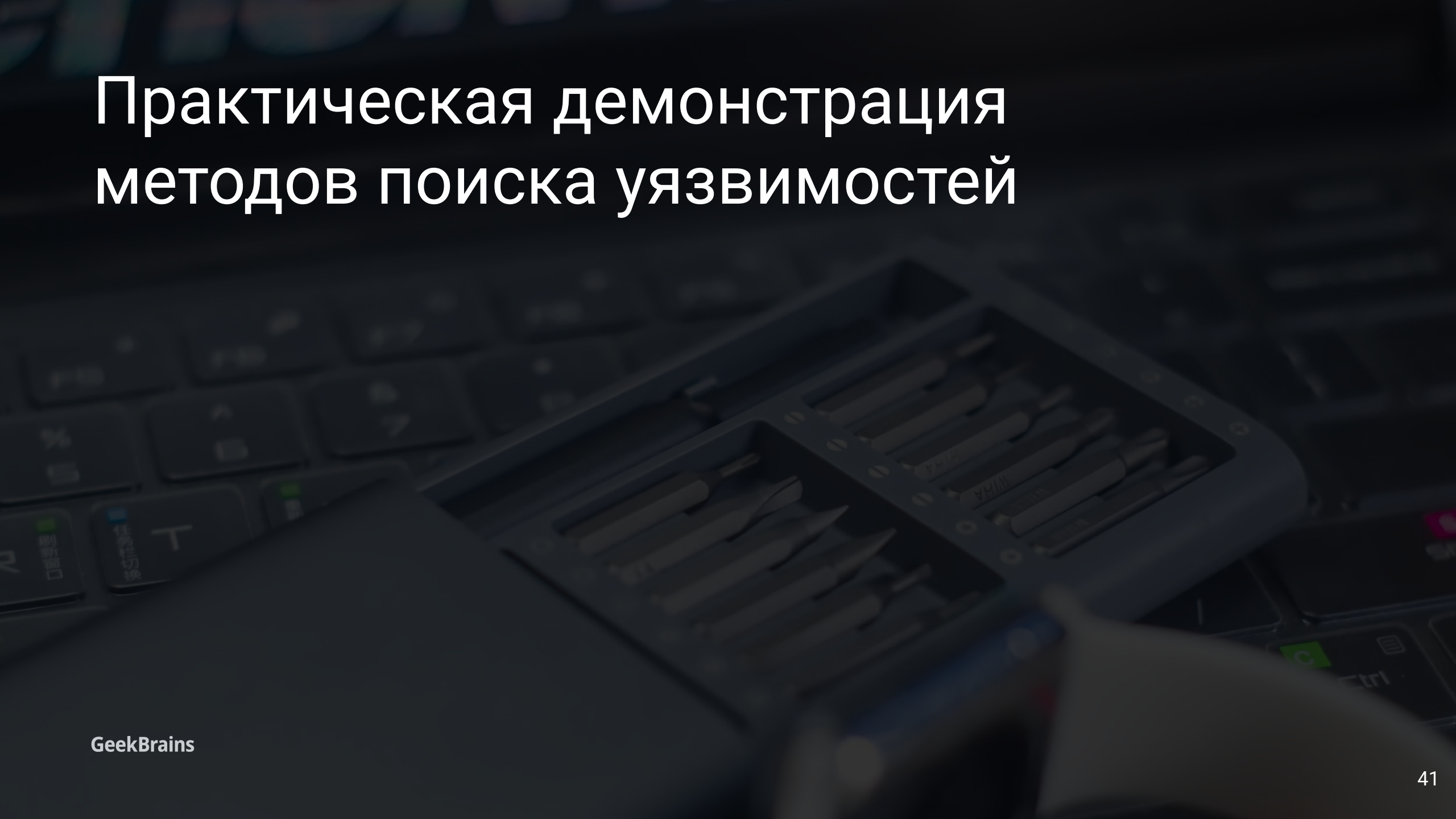
1. Поиск уязвимостей сканерами по базам и сигнатурам.
2. Recursive fuzzing (рекурсивный фаззинг) — идет подбор всех возможных данных (всего возможного алфавита),

Методы поиска уязвимостей

3. Replacive fuzzing (заменяющий фаззинг) – идет подстановка всех возможных параметров, которые задаются из какого-либо источника (например, из файла).

4. Bruteforce (подбор параметров). Идея заключается в передаче параметров (к примеру, логинов и паролей), которые могут «подойти».

Практическая демонстрация методов поиска уязвимостей



Практическое задание

1. Имеется логин admin и пароль yo30E#jb, которые были заданы администратором для входа в систему с использованием веб-формы. Можно ли считать такую комбинацию логина и пароля безопасной для защиты от брутфорса? Ответ обоснуйте.

2. Подберите логин и пароль к странице bruteforce в сервисе DVWA на уровне сложности LOW.

Практическое задание (повышенная сложность)

1. Решите задание http://challenge01.root-me.org/web-serveur/ch3/ методом брутфорса.

Ваши вопросы