# Моя тема и её описание.

Применение фаззинга для тестирования различных ПО. Обзор на известные методы фаззинга.

Методы анализа программного обеспечения, методика исследования программного обеспечения на наличие уязвимостей - фаззинг, различные методы и применения фаззинга. Как фаззинг помогает нам в тестировании. Различные инструменты фаззинга и его трюки и фишки. Самые главные проблемы, которые могут возникнуть при использовании фаззинга. Структура фаззинга и полное описание всех этапов фаззинга для тестирования ПО разных типов и объёмов. Перспектива фаззинга в будущем и какие методы тестирований ПО могут заменить фаззинг, а какие ему сильно проигрывают. Каким образом фаззинг можно усовершенствовать и сделать его более удобным и ещё более автоматизированным.

1) Как применять фаззинг для тестирования ваших проектов.

2) Разные виды тестирований (различия и сходства).

3) Чем фаззинга может быть полезен, а чем наоборот мешать в тестировании.

4) Все тонкости фаззинга при тестировании и его полная структура.

5) Разновидности фаззинга и краткий обзор по применению для каждого их них.

Статьи на хабре и различные интернет ресурсы. Книга "Fuzzing: исследование уязвимостей методом грубой силы" от авторов Майкла Саттона и Адама Грина.

Методы фаззинга и в чём их основные идеи?

# План выступления

* Что такое фаззинг и вступление – **1.5 минута.**
* Как фаззинг работает в общих чертах – **4 минуты.**
* Главные отличия от других методов тестирования. – **1 минута**
* Разные виды фаззинга. – **2 минуты**
* Главные проблемы фаззинга и их решения. – **3 минуты**
* Краткий пример работы фаззинга. – **2 минуты.**
* Заключение – **1 минута**

# Что такое фаззинг и вступление

Всем привет дорогие друзья! Я рад представить вам доклад на тему “Применение фаззинга для тестирования различных ПО. Обзор на известные методы фаззинга”.

Все мы знаем, что тестирование – это неотъемлемый этап процесса разработки программного обеспечения. Именно поэтому сегодня мы погрузимся в один из самых интригующих и перспективных подходов к тестированию – фаззинг тестирование.

Фаззинг-тестирование, подобно хакеру, использует автоматизированные методы для генерации входных данных и обнаружения ошибок и уязвимостей. Раньше такой подход использовался для тестирования ПО безопасности, но сейчас становится всё более популярным во многих дургих сферах.

В этом докладе мы рассмотрим основные принципы фаззинг-тестирования, его преимущества и недостатки перед другими методами тестирования, а также обсудим, как этот подход может быть интегрирован в общий процесс тестирования ПО.

# Как фаззинг работает в общих чертах

Фаззинг-тестирование – это попытка нагрузить программу случайными данными и проверить не происходит ли ничего предосудительного.

В SDL (то бишь в жизненном цикле безопасной разработки) имеется пункт верификации. Это этап, на котором мы осматриваем программу на предмет отсутствия неприятных инцидентов. И в этом пункте есть подпункт с динамическим анализом. Именно в этом месте и применяется фаззинг-тестирование.

Почему же фаззинг тестирование кардинально отличается от обычного юнит-тестирования? А тем, что при классическом тестировании нереально покрыть все возможные варианты входных данных. Фаззинг же позволяет разнообразить юнит-тесты случайным вводом и таким образом покрыть больше вариантов входных данных.

В то же время максимально нерационально использовать абсолютно все данные мира. Значит, попробуем сузить входные данные для нашего фаззера. Разработчик должен перед тестирование придумать набор сэмплов (шаблонов), который будет покрывать полностью весь функционал нашей программы. Уточним, что все данные из этого набора являются верными и валидными и мы знаем, как программа на этих данных себя покажет. Затем, мы запустим наш фаззер, но внедрим в его работу оператор мутации (искажение), который будет образовывать из наших сэмлов новые сэмплы, отличающиеся от исходных. **Вот некоторый список мутации, на котором мы долго останавливаться не будем.**

И теперь фаззер будет вкидывать мутировавшие сэмлпы пока не получит что-то новое. Но что вообще значит что-то новое?

Введём понятие базового блока и трассового выполнения.

* Базовый блок – набор процессорных инструкций (выполняются последовательно; отсутствуют ветвления и условные переходы; передачи управления извне только в начало блока)
* Трасса выполнения – последовательность базовых блоков, которые были выполнены при конкретном запуске программы.

**(Сколько фаззить – время с момента нахождения последней новой трассы!) Смотрит на трассу выполнения и определяет выкинуть её или добавить в исходный набор сэмплов.**

# Главные различия от других методов (техник) тестирования

* **Автоматизация:** Фаззинг-тестирование является высокоавтоматизированным методом, который использует специальные инструменты для генерации входных данных и обнаружения ошибок. Это позволяет проводить тестирование быстрее и эффективнее, чем при других методах тестирования.
* **Неопределённость:** В отличие, от других методов тестирования, который основываются на определённых тестовых случаях, фаззинг тестирование использует случайные (то бишь мутированные) входные данные, чтобы выявить неожиданные ошибки. Это делает его особо эффективным для тестирования ПО, которое подвергается высокому риску атак.
* **Предназначено для безопасности:** Фаззинг-тестирование изначально был разработан для тестирования ПО безопасности, и поэтому он имеет сильный фокус на обнаружении уязвимостей и ошибок, которые могут быть использованы злоумышленниками.

# Разные виды фаззинга и их реализации