**Fuzz Testing**

**1 Суть метода**

Fuzz Testing(fuzzing, фаззинг) – методика тестирования, при которой на вход программы подаются невалидные, непредусмотренные или случайные данные. Это своего рода стрессовое тестирование программы.

Смысл такой проверки сводится к тому, что тестировщик не знает, какие данные будут переданы приложению/протоколу/функции, поэтому его задача предусмотреть и проверить как можно больше вариантов. Если происходит зависание, крах, утечка памяти или другое ненормальное поведение, то можно говорить об обнаружении проблемы, и высока вероятность, что это проблема в безопасности.

По способу манипуляции с данными фаззинг распределяется на генерацию и мутацию.

1. Генерация — это создание условий для тестирования с чистого листа. Например, это генерация набора байтов, которые подсовывается, программе для просмотра картинок со словами: "Это на самом деле JPEG-файл, читай его".

Генерация потенциально ошибочных данных не ограничена псевдослучайными последовательностями. Она может быть проведена по определенным шаблонам или имеющимся входным данным.

1. Мутация — прием намного более изящный и подразумевает внесение изменений в "хороший", то есть вполне корректный файл.

Все фаззеры(программы, осуществляющие фаззинг) работают примерно одинаковым образом, позволяя задавать некоторые ограничения на мутирование входных данных определенными байтами или последовательностью байтов.

**2 Решаемые задачи**

Фаззинг повышает безопасность программного обеспечения, т.к. фаззеры обнаруживают критические проблемы уязвимости в короткий срок. Ошибки, найденные фаззером, могли бы быть использованы злоумышленниками, если их вовремя не исправить

Метод позволяет обнаружить аварийные ситуации, ошибки, сбои и утечки памяти. Он может быть пригоден для тестирования больших приложений, где любая ошибка памяти может привести к серьёзной уязвимости.

Fuzz Testing также позволяет найти непредусмотренные различия между двумя версиями программы или между двумя реализациями из одной и той же спецификации.

Обычно данный метод применяется для тестирования программ, для которых работоспособность и безопасность при любых входных данных являются наиважнейшими приоритетами/ веб-сервер, клиент/сервер SSH, sandboxing, сетевые протоколы.

Сам по себе Fuzz Testing является простым методом и дает возможность проведения тестирования в автоматическом режиме.

**3 Недостатки метода**

Fuzz Testing проверяет устойчивость программы на поступление некорректных входных данных, однако не проверяет правильность обработки верных, что требует дополнительных тестов.

Используя фаззинг, генерируются случайные данные, а значит нельзя точно быть уверенным, что будут протестированы какие-либо значения, ломающие программу(например, могут быть упущены очевидные значения). Сложно обеспечить проверку всех критических ветвей кода. Некоторые тесты всё равно придется проводить вручную.

Сам фаззер ничего не знает о внутреннем состоянии программы, что может затруднить проведение автоматического тестирования. Также он вынужден перебирать огромное количество вариантов, которые растет экспоненциально в зависимости от входных данных.

**Ответы по технике:**

1. Преимущества:

* Простота метода.
* Легко автоматизируется процесс тестирования
* Находит много ошибок, которые трудно найти при ручном тестировании
* Возможность быстрого получения удовлетворительного результата(в нашем случае – ошибки в программе)
* Очень сильно помогает выявить на раннем этапе то, что может “завалить” программу в боевом режиме.
* Позволяет выявить нестандартное поведение программы, вплоть до вредоносного воздействия на программу или даже систему, в которую она входит.

Данная техника применяется, когда необходимо проверить, что система без сбоев обрабатывает ошибочные ситуации, а не ведет себя неправильно.

Методика получила довольно широкое развитие и сегодня используется для тестирования широкого типа приложений, выполняющих обработку файловых структур, взаимодействующих с пользователями через интерфейс.

1. Фаззинг как методика тестирования в основном относится к областям серого и черного ящиков.

При использовании метода черного ящика исходный код недоступен. Фаззинг, проводимый в ходе этого метода, называют неуправляемым. При этом вводятся данные и отслеживается реакция системы, однако неизвестны детали, касающиеся внутреннего состояния объекта. Фаззинг посредством метода серого ящика считается наилучшим, т.к. можно сгенерировать такие данные, при которых приложение не сломается, а выдаст ошибку.

Управляемым фаззинг будет в том случае, если программы – белый ящик, подбор данных для которой осуществляется на основе определенной модели. Однако не рекомендуется использовать метод белого ящика, так как то, что мы видим(исходный код) – это может быть совсем не то, что необходимо выполнить.

1. Фаззинг чаще всего применяется как автоматическое и полуавтоматическое тестирование(когда на данные генерируются программой, а тесты проводит тестировщик). Однако следует учитывать, что не все данные могут быть проверены. Следовательно, фаззинг и ручное тестирование должны дополнять друг друга.