* Cách cài đặt: cài đặt tại nguồn chính của AFL trên github <https://github.com/google/AFL>
* Hệ điều hành: Linux 64 bit 4GB ram, 6 core CPU
* Tham khảo cách sử dụng AFL: <http://fuzzing.in/codelabs/fuzzing_opensource/index.html?index=../..index#0>
* Thử thách CTF dùng để áp dụng: CSAW challenge wyvern (rev-500)
* Chương trình dùng để thử nghiệm fuzzing

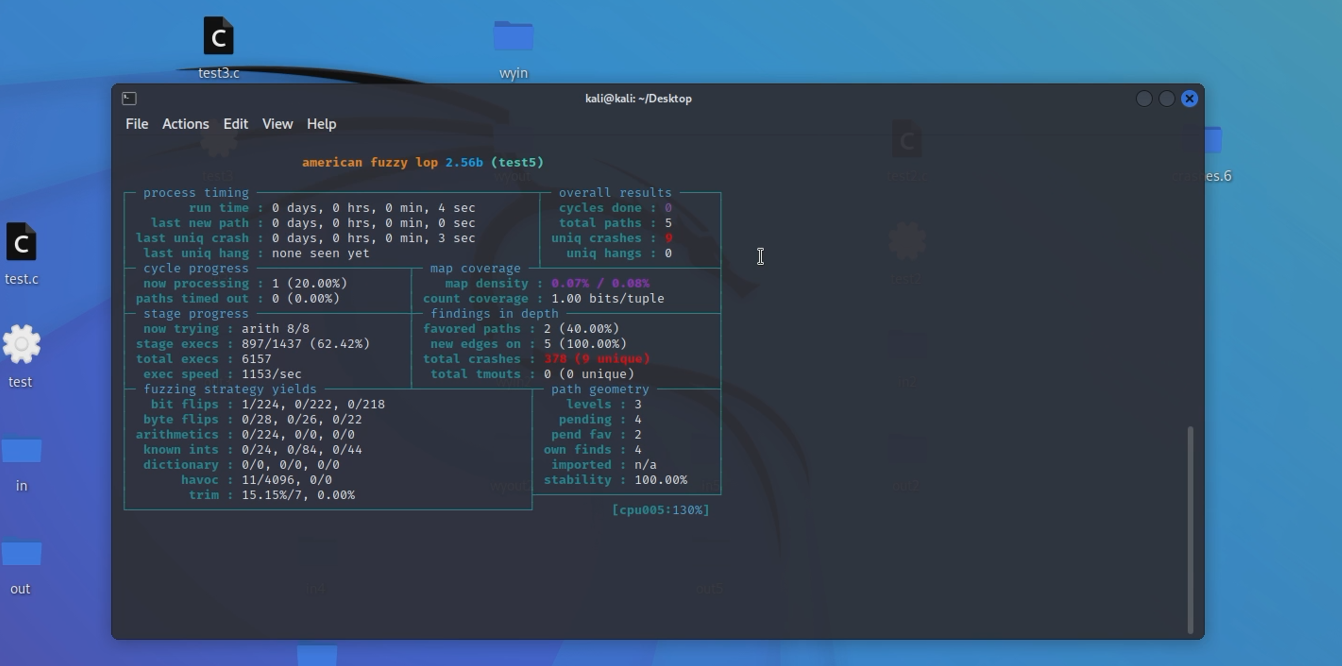
Ảnh có chứa văn bản

Mô tả được tạo tự động

Ảnh có chứa văn bản

Mô tả được tạo tự động

+ Kết quả thực nghiệm

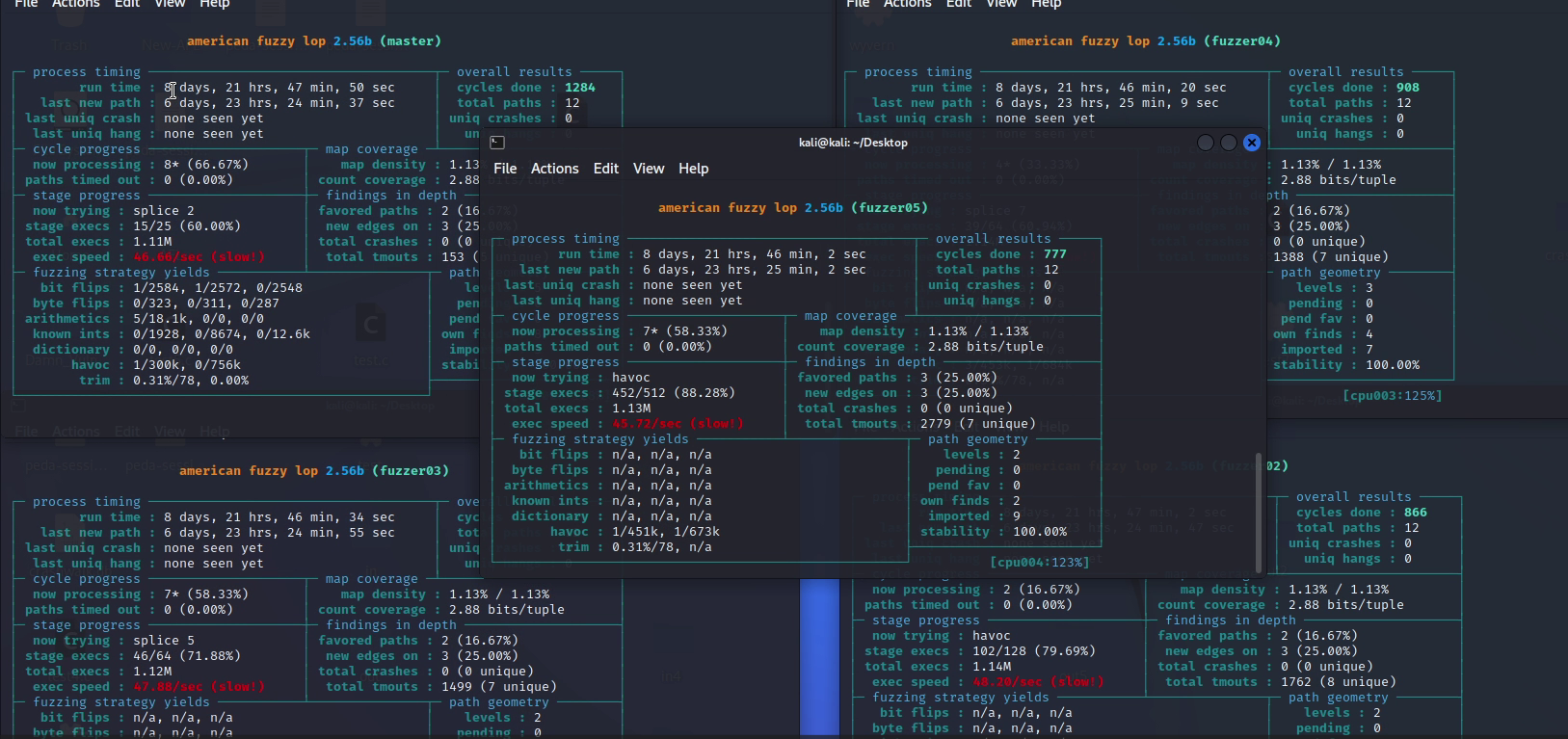


* Hình ảnh khi thực hiện fuzzing 1 chương trình có các lỗi: interger overflow, interger underflow, double free, use after free, out of bound

Ảnh có chứa văn bản

Mô tả được tạo tự động

* Hình ảnh khi sử dụng công cụ crashwalk giúp phân tích và xác định các lỗi crash mà chương trình AFL fuzzing ra được



* Hình ảnh khi dùng công cụ AFL để fuzzing chương trình CTF wyvern sử dụng 6 core, với 1 core để làm master và 5 core fuzzer phụ để giúp quá trình fuzzing tiến hành nhanh hơn

Ảnh có chứa văn bản

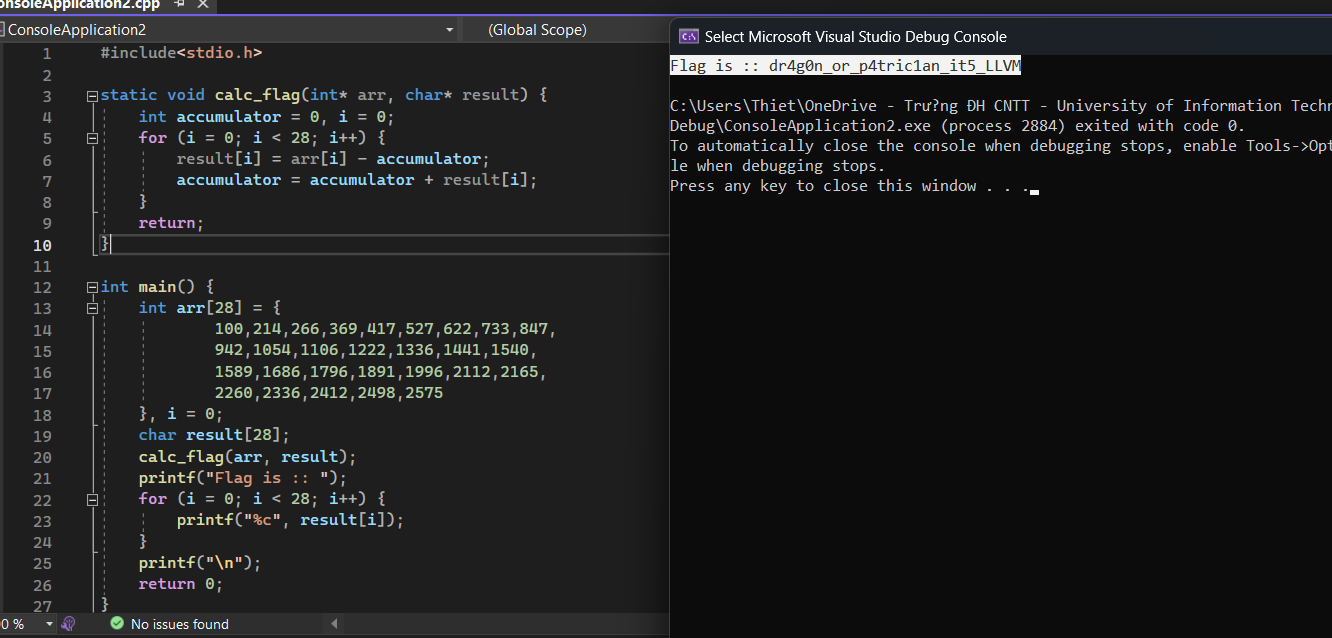
Mô tả được tạo tự động

* Hình ảnh khi ta chưa biết nhập gì hay bao nhiêu ký tự vào chương trình wyvern thì sẽ thất bại

Ảnh có chứa văn bản

Mô tả được tạo tự động

* Queue mà chương trình AFL đưa vào giúp xác định số lượng ký tự cũng như hỗ trợ để giải chương trình CTF



* Chương trình C++ viết để giải ra flag sau khi đã biết được quy luật của input đầu vào

Ảnh có chứa văn bản

Mô tả được tạo tự động

* Kết quả khi ta đưa đúng đầu vào chương trình CTF

\*Nhận xét

+ Chương trình AFL là 1 trình fuzzer giúp ta có thể kiểm thử black box, grey box, nó hỗ trợ ta trong quá trình pentest giúp tốn ít thời gian hơn so với quá trình pentest thông thường

+ Nhờ kỹ thuật fuzzing và ta có thể giải được 1 chương trình CTF nhanh hơn so với quá trình reverse thông thường