Bộ môn An toàn Thông tin – Khoa MMT&TT
Trường Đại học Công nghệ Thông tin (UIT)

BÁO CÁO TỔNG KẾT SEMINAR

Môn học: Cơ chế hoạt động của mã độc

Tên chủ đề: Cryptojacking Detection System

GVHD: Phan Thế Duy

Ngày báo cáo: 30/5/2022

Team: NBG

THÔNG TIN CHUNG:

(Liệt kê tất cả các thành viên trong nhóm)

Lớp: NT230.M21.ANTN

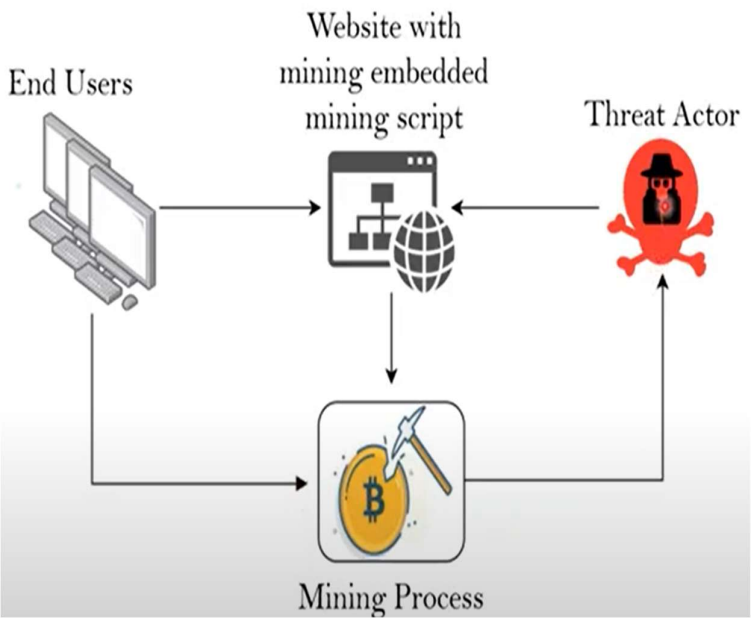
STT	Họ và tên	MSSV	Email
1	Lâm Thanh Ngân	19521884	19521884@gm.uit.edu.vn
2	Lê Hồng Bằng	19520396	19520396@gm.uit.edu.vn
3	Nguyễn Ngọc Quỳnh Giang	19520500	19520500@gm.uit.edu.vn

Phần bên dưới của báo cáo này là tài liệu báo cáo chi tiết của nhóm thực hiện.

BÁO CÁO CHI TIẾT

Nội dung báo cáo

1. Ngữ cảnh



- Sự thay đổi của tiền điện tử mới ra đời: CPU-bound PoW (Bitcoin or Ether) -> Memory-bound PoW schemes (Monero) thuận tiện cho việc đào coin trên máy tính thông thường
- Lợi ích lớn đến từ tiền điện tử, và lợi ích của việc khai thác trộm.
- WSAM: các công nghệ này đã phục vụ để tạo điều kiện cho các ứng dụng web có hiệu suất cao, có thể mở rộng chạy trên các trình duyệt.
- Dịch vụ khai thác Coinhive đã cung cấp các tập lệnh khai thác Monero dựa trên WebAssembly cho chủ sở hữu trang web khai thác trong quá trình khách truy cập vào website của họ, lén lút đào tiền mà k có sự cho phép của ng dùng, lạm dụng khả năng xử lí (processing power) của nạn nhân và lấy doanh thu => Đây là lý do cryptojacking ra đời.

⇨ Do cryptojecking malware ngày càng phổ biến và những ảnh hưởng xấu mà nó mang lại nên cần có những hệ thống phát hiện loại mã độc này để bảo vệ người dung khi truy cập các trang web được an toàn.

2. Web Assembly và Cryptojacking Malware

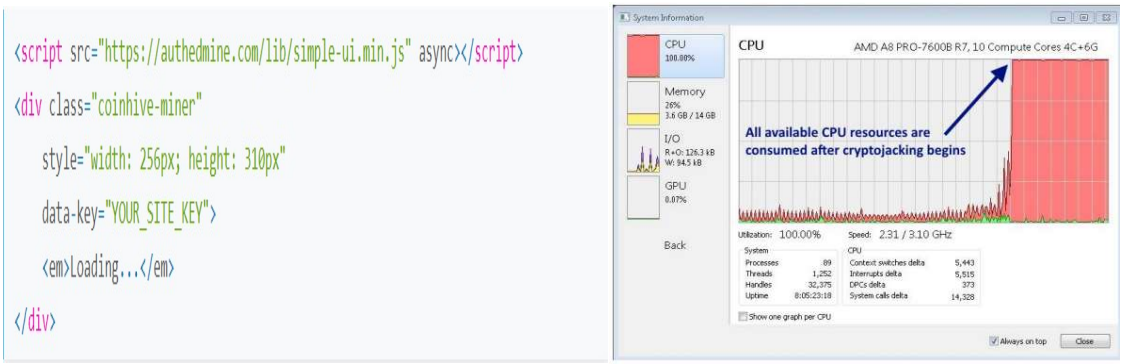
a) Cryptojacking

Là một loại fileless malware khai thác tiền điện tử mới hoạt động bí mật trên các trình duyệt của người dùng cuối dựa trên công nghệ web mới nhất và dễ dàng tiếp cận nạn nhân của nó thông qua các trang web mà không yêu cầu bất kỳ cài đặt phần mềm nào. Nó chạy ẩn trong nền, yên lặng đào tiền ảo trên thiết bị của bạn và sau đó gửi cho những kẻ tấn công. Nếu bạn không chú ý thấy máy tính chạy chậm hoặc tiến trình sử dụng 100% CPU, bạn thậm chí còn không nhận ra có phần mềm độc hại trên thiết bị của mình.

(fileless malware là loại phần mềm độc hại không dựa vào các tệp nào, không để lại dấu vết, khiến việc loại bỏ trở nên khó khăn. Nó sử dụng các chương trình hợp pháp, như RAM của PC, RAM dùng để lưu trữ phần mềm trong khi chạy)

Các tác giả của phần mềm độc hại cryptojacking sử dụng các kĩ thuật CPU throttling, obfucation, dynamin domain name, mã hóa giao tiếp để chống lại các kĩ thuật, cơ chế phát hiện cryptojacking

- CPU Throttling là một cơ chế điều chỉnh xung nhịp của CPU, làm giảm hiệu suất của CPU để tiết kiệm pin hoặc để tránh tình trạng CPU bị quá nhiệt (do sử dụng nhiều hoặc chạy các phần mềm, các game nặng…) và làm hỏng các bộ phận bên trong máy.
- Obfuscation là hành động cố tình tạo mã bị xáo trộn, tức là mã nguồn hoặc mã máy rất khó hiểu đối với con người. Nó là một cái gì đó tương tự như mã hóa, nhưng một máy có thể hiểu mã và có thể thực thi



(Script độc hại và ảnh hưởng đến hiệu năng CPU của nạn nhân)

b) ***WebAssembly (Wasm)***

Wasm là một định dạng lệnh nhị phân cấp thấp chạy mã gần với tốc độ gốc trong máy ảo dựa trên stack trong trình duyệt, được hỗ trợ bởi 4 trình duyệt chính gồm gg chrome, Mozilla Firefox, Microsoft Edge và Safari

- Hiệu quả về kích thước, thời gian tải, tốc độ thực thi
- Dễ giải mã, k phụ thuộc vào phần cứng và nền tảng, nhỏ gọn
- Bổ sung và chạy song song JS (không bỏ JS), được biên dịch trong môi trường Sandbox, sử dụng các Web APIs có sẵn giống JS, các Wasm module có khả năng call in and out of the JS context và truy cập chức năng của trình duyệt
- Toolchain đc sử dụng rộng rãi để biên dịch C/C++ thành Wasm là LLVM complier
- Tốc độ gần như nguyên bản của Wasm do các modules đc tối ưu hóa trong quá trình biên dịch và việc quản lí bộ nhớ đc thực hiện mà k cần sử dụng garbage collector

➔ Các tính năng thuận lợi của Wasm phù hợp với việc triển khai và thực thi các chức năng đào tiền điện tử bằng trình duyệt yêu cầu năng lượng tính toán đáng kể như cryptonight_hash => đa số các phần mềm độc hại cryptojacking dựa trên trình duyệt triển khai Wasm để thực thi các payload đào tiền ảo (theo nghiên cứu của Konoth thì 100% trong 1735 website đào tiền ảo sử dụng Wasm. Hay nghiên cứu của Musch phân tích có 0,16% trong 1 triệu trang web hàng đầu của Alexa sử dụng Wasm và hơn một nữa số đó có mục

đích xấu, cryptojacking là ứng dụng chính trong số các trường hợp sử dụng các phần mềm độc hại

c) *Ví dụ các bước thực hiện cơ bản của cryptojacking malware*

- Tác giả của cryptojacking malware viết mã bằng c/c++ thực hiện nhiu chức năng khai thác bao gồm
  - Cryptonight_create
  - Cryptonight_destroy
  - Cryptonight_hash
- Sau đó compile nó thành 1 Wasm module bằng toolchain Emscripten, Wasm module sau đó được truy cập thông qua hàm JS (WebAssembly.instantiateStreaming)
- Phương thức fetch() của Fetch JS API sử dụng đối số đầu tiên của hàm. Phương thức này biên dịch và khởi tạo module và cho phép truy cập vào raw bytecode. Trong giai đoạn biên dịch, mã nhị phân Wasm đã được tối ưu hóa và có thể kết nối trực tiếp với backend nơi mã máy đc tạo và thực thi. Mã này thực hiện các phép toán để tạo điều kiện thuận lợi cho việc giải các hash puzzle có thể biến đổi, tức là khai thác tiền điện tử

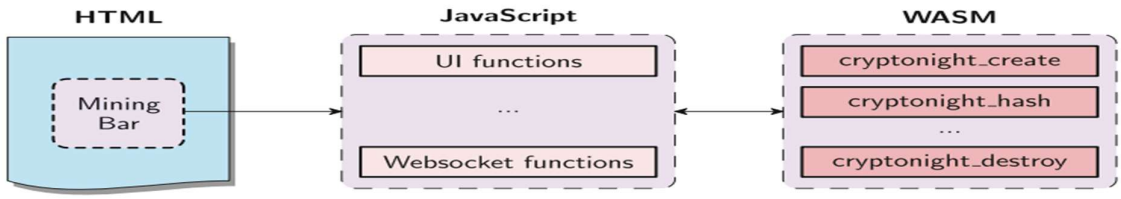
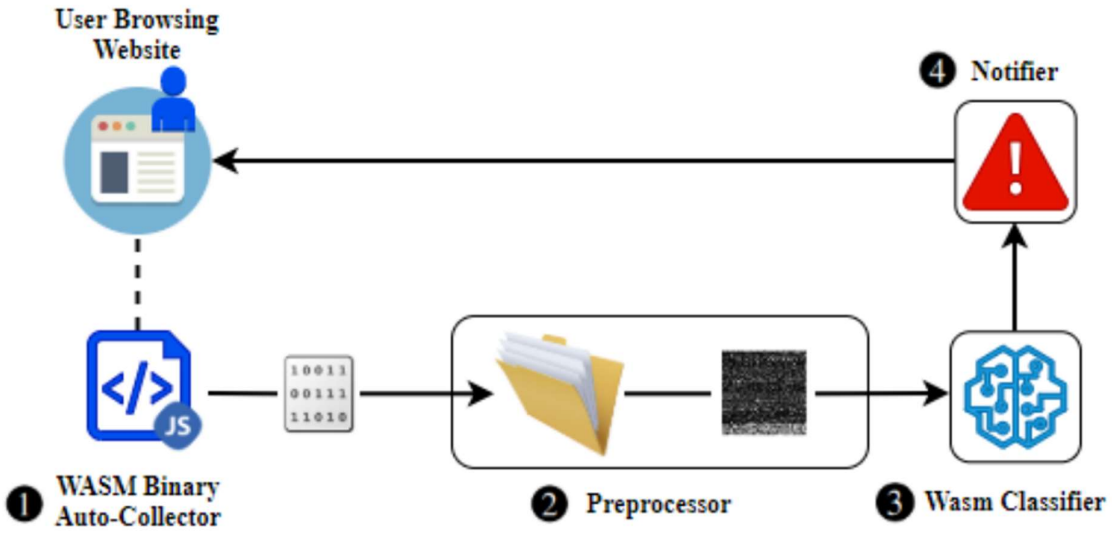


Fig. 1. Browser-based mining workflow

```
const obj = {
    imports: {
        imported_func: function(arg) { console.log(arg); }
    }
};
const wasm = await WebAssembly.instantiateStreaming(
    fetch('example.wasm'), obj
);
let result = wasm.instance.exports.factorial(13);
```

Fig. 2. Instantiating a WebAssembly module and calling an exported function.

3. MINOS Framework



System model gồm 4 thành phần chính:

- Wasm module auto-collector: bộ thu module tự động, khi ng dùng đang sd trình duyệt web thì nó sẽ chạy liên tục trong nền, kiểm tra xem các web đang đc truy cập này có đang tạo ra bất kì tệp nhị phân Wasm nào không, nếu có thì tự động tải và trích xuất tệp nhị phân Wasm đc liên kết vào 1 folder đc chỉ định (chỉ tải Wasm binaries và không tải bất kì thành phần trang web khác, nếu có nhiều hơn 1 module Wasm thì tải xuống toàn bộ)
- Preprocessor: bộ tiền xử lí, nó sẽ đọc các folder đc tải xuống trc đó và chuyển đổi từng tệp nhị phân trong folder thành hình ảnh gray-scale và tiếp tục xử lí trước hình ảnh này thành 1 dạng mà mạng nơ-ron có thể sử dụng làm đầu vào (thay đổi kích thước nó thành kích thước chung). Preprocessor chuyển đổi nhị phân thành 1 mảng các số nguyên với mỗi số nguyên đại diện cho một pixel của gray-scale, sau đó chuẩn hóa và định hình (normalizes and reshapes) lại mảng kết quả rồi chuyển qua Wasm classifier

Algorithm 1: Preprocessor

```
def preprocess():
    while len(Wasm_directory) == 0 do
        time.sleep(1)
    if len(Wasm_directory)! = 0 then
        Wasm_images → []
        for file in Wasm_directory do
            f → open(file)
            ln → getSize(file)
            width → math.pow(ln, 0.5)
            rem → ln%width
            a → array('B')
            a.fromfile(f, ln − rem)
            f.close()
            os.remove(file)
            g → reshape(a, (len(a)/width), width)
            g → uint8(g)
            h → resize(g, size, size)
            h → h/255
            h → h.reshape(−1, 100, 100, 1)
            Wasm_images(h)
    classify(Wasm_images)
return preprocess()
```

- Wasm classifier: bộ phân loại wasm, các mã nhị phân đã chuyển đổi đc đưa vào đây, một CNN đc đào tạo trước sẽ phân loại từng tệp nhị phân đc xử lí trc đó là độc hại hay lành tính
- Notifier: bộ thông báo, dựa vào kết quả phân loại trên sẽ quyết định có thông báo ng dùng về hd khai thác độc hại hay không. Nếu phát hiện là độc hại thì sẽ thông báo ng dùng trang web mà họ đang truy cập sử dụng tài nguyên tính toán của họ để khai thác tiền điện tử và họ nên đóng nó và chấm dứt mọi quy trình khai thác chạy trong nền. Nếu đó là tệp nhị phân lành tính thì sẽ ko thông báo làm gián đoạn ng dùng, và Wasm Module Auto-collector sẽ tiếp tục kiểm tra việc khởi tạo Wasm module

Chống lại việc làm rối:

Listing 1: An example code snippet from the source code of a browser-based cryptocurrency miner.

```
1
2 static void copy_block(uint8_t *dst, const uint8_t *src)
3 {
4   ((uint64_t *)dst)[0] = ((uint64_t *)src)[0];
5   ((uint64_t *)dst)[1] = ((uint64_t *)src)[1];
6 }
```

Listing 2: An example code snippet from the source code of a browser-based cryptocurrency miner that has been obfuscated.

```
1
2 static void fun55(uint8_t *dst_new, const uint8_t *src)
3 {
4   ((uint64_t *)dst_new)[0] = ((uint64_t *)src)[0];
5   ((uint64_t *)dst_new)[1] = ((uint64_t *)src)[1];
6 }
```

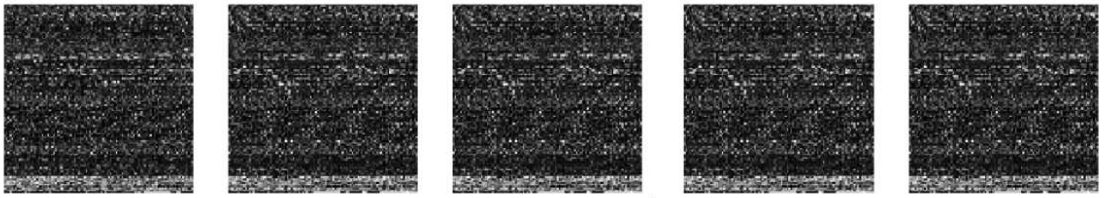
Như một nỗ lực để đánh giá mức độ mạnh mẽ của MINOS trước sự trốn tránh của các tác giả cryptojaking malware

Sử dụng một công cụ khai thác tiền điện tử dựa trên trình duyệt mã nguồn mở, cụ thể là Webminerpool làm mã khai thác cơ bản để làm xáo trộn. Webminerpool cung cấp các triển khai Cryptonight PoW khác nhau được viết bằng C và tạo một trình khai thác dựa trên Wasm sau khi biên dịch.

Mặc dù obfuscation đã là một chủ đề nghiên cứu tích cực trong lĩnh vực phần mềm độc hại, nhưng nó là một lĩnh vực chưa được khám phá trong phần mềm độc hại cryptojacking.

Áp dụng obfuscation trước quá trình biên dịch các mô-đun Wasm, trong mã nguồn C cấp cao của Webminerpool. Để thực hiện phương pháp xáo trộn phổ biến nhất trong phương pháp xáo trộn tên hàm,

Như thể hiện trong hình, mặc dù số lượng tên hàm khác nhau bị xáo trộn trong mã trình khai thác, các tệp nhị phân kết quả của trình khai thác có các biểu diễn hình ảnh thang xám rất giống nhau. Để đánh giá mức độ mạnh mẽ của MINOS so với các mẫu công cụ khai thác bị xáo trộn được hiển thị trong Hình 7, đã cung cấp các mẫu cho MINOS và quan sát thấy rằng MINOS có thể phát hiện các mã nhị phân bị xáo trộn bất kể mức độ hoặc mức độ xáo trộn.



(a) Original miner (b) 25% obfuscated (c) 50% obfuscated (d) 75% obfuscated (e) 100% obfuscated

- Cốt lõi:

MINOS nắm bắt bản chất của phần mềm độc hại mã hóa do một tính năng độc đáo của phần mềm độc hại mã hóa (tức là triển khai các lược đồ PoW cụ thể dựa trên bộ nhớ hoặc các câu đố băm ràng buộc CPU) khiến các triển khai tương tự về mặt cú pháp và

ngữ nghĩa với nhau. Điều này đã được xác minh bởi cả phân tích ban đầu của chúng tôi trong Phần V-A và nghiên cứu của Wang et al. [17].

- Ưu điểm:
  - Thời gian chạy nhanh
  - Khả năng phát hiện tức thời
  - Không cần quyền admin
  - Nền tảng độc lập
  - Chất lượng về trải nghiệm lướt web của ng dùng
  - Chống lại các nỗ lực trốn tránh thông thường

4. Demo

- Thực hiện demo 2 trên 4 phần của mô hình phát hiện cryptojecking malware của hệ thống MINOS.
- Phần đầu là WASM Collector
  - Thực hiện thu gom các wasm binaries trong các trang web
- Phần tiếp theo là Preprocessor
  - Preprocessor chuyển đổi nhị phân thành 1 mảng các số nguyên với mỗi số nguyên đại diện cho một pixel của gray-scale, sau đó chuẩn hóa và định hình (normalizes and reshapes) lại ảnh gray-scale