***XBA.***

***Tóm tắt:***

* Phân tích Nhị phân Đa nền tảng yêu cầu một biểu diễn chung của các tập tin nhị phân trên các nền tảng khác nhau, dựa vào đó có thể thực hiện phân tích cụ thể.
* Tuy nhiên, không may, các kỹ thuật hiện có đều không đạt được yêu cầu vì:
* Chúng chỉ đối với một nền tảng cụ thể và tạo ra các biểu diễn không được căn chỉnh trên các nền tảng khác nhau.
* Hoặc không được thiết kế để thu thập thông tin ngữ cảnh phong phú có sẵn trong mã nhị phân đã tháo dỡ.
* Chúng tôi giới thiệu một phương pháp mới dựa trên học sâu mang tên XBA, giải quyết các vấn đề nêu trên.

1. ***Giới thiệu:***

* Mục tiêu của bài báo này là học mô hình biểu diễn có thể tạo ra các lời giải của tập tin nhị phân, có độ giàu ý nghĩa ngữ nghĩa và căn chỉnh trên các nền tảng khác nhau - trên các hệ điều hành (OS) hoặc kiến trúc bộ chỉ thị (ISAs) khác nhau.

A screenshot of a computer code

Description automatically generated

***Hình 1: Mã tương đương về mặt ngữ nghĩa trong OpenSSL có cài đặt khác nhau trên các hệ điều hành hoặc kiến trúc máy tính.***

* Rất tiếc, các phương pháp hiện có tạo ra các biểu diễn mã nhị phân không giàu ý nghĩa và không được căn chỉnh trên các nền tảng khác nhau để hỗ trợ phân tích nhị phân đa nền tảng.
* Cùng xem xét hai cặp mã nguồn ví dụ được hiển thị trong Hình 1, trong đó mỗi cặp có cùng ngữ nghĩa, nhưng ở dạng cụ thể của một hệ điều hành (OS) hoặc kiến trúc bộ chỉ thị (ISA) khác nhau.
* Các phương pháp học không giám sát hiện có chỉ huấn luyện mô hình của họ cho một nền tảng duy nhất, không thể tạo ra các biểu diễn được căn chỉnh trên các nền tảng.
* Trong khi đó, các phương pháp học có giám sát hiện có lại huấn luyện mô hình của họ trực tiếp tối ưu hóa cho sự căn chỉnh giữa các khối cơ bản và các hàm trên các nền tảng khác nhau.
* Cả hai phương pháp đề chỉ sử dụng các đặc trưng nội bộ của từng khối cơ bản hoặc từng hàm, và không sử dụng thông tin ngữ cảnh của chúng (Ví dụ: chuyển dòng điều khiển nội hoặc ngoại hoặc tham chiếu đến chuỗi chữ) có sẵn trong mã nhị phân đã tháo dỡ.
* Bỏ qua thông tin ngữ cảnh này có thể vô tình dẫn đến việc tạo ra các mô hình có khả năng ý thức hạn chế.
* Do đó, chúng tôi đề xuất một phương pháp tiếp cận bán giám sát dựa trên đồ thị để học mô hình biểu diễn, giúp tạo ra các biểu diễn mã nhị phân có ý nghĩa ngữ nghĩa và được căn chỉnh trên các nền tảng khác nhau - trên các ISA hoặc OSs.

***Kỹ thuật được sử dụng:***

* Kỹ thuật chính của chúng tôi là căn chỉnh cấp đồ thị được thực hiện trên mã nhị phân theo cấu trúc đồ thị trên các nền tảng.
* Một cặp tập tin nhị phân biên dịch cho các nền tảng khác nhau được trừu tượng thành các đồ thị có kiểu, gọi là đồ thị giải mã nhị phân (BDGs), mã hoá thông tin ngữ cảnh phong phú.
* Sau đó, hai mạng neural đồ thị chia sẻ tham số của chúng được huấn luyện, tối ưu trực tiếp cho sự căn chỉnh giữa hai đồ thị đó, sử dụng (i) thông tin căn chỉnh giữa các nền tảng một phần (do đó, hình thành một quá trình học bán giám sát), và (ii) thông tin ngữ cảnh phong phú được mã hóa trong mỗi BGD.
* Sử dụng mạng neural đồ thị (GCNs) trong lĩnh vực xử lý ngôn ngữ tự nhiên.

***Các đóng góp của đồ án bao gồm:***

* Giới thiệu một phương pháp mới dựa trên học sâu mang tên XBA, giúp học biểu diễn mã nhị phân bằng mục tiêu căn chỉnh đồ thị.
* Để đánh giá XBA. Chúng tôi chỉ ra rằng XBA cải thiện các phương pháp cơ sở một cách đáng kể, và XBA có thể nắm bắt ngữ nghĩa của mã nhị phân từ một tập con nhỏ của các mẫu được đánh nhãn.

1. ***Nền tảng:***
2. ***Học biểu diễn mã nhị phân:***

* Các công trình gần đây đã cho thấy rằng học biểu diễn vượt trội hơn so với các phương pháp truyền thống dựa trên heuristic trong nhiều nhiệm vụ phân tích nhị phân, chẳng hạn như phát hiện sự tương đồng nhị phân, dự đoán tên hàm và xác định ranh giới hàm.
* Chìa khóa thành công trong việc này là học các biểu diễn phân tán của mã nhị phân - ở cấp độ của các chỉ thị, khối cơ bản hoặc hàm - trong không gian vector, còn được gọi là nhúng, có thể được sử dụng cho các nhiệm vụ hậu xử lý khác nhau.
* Chúng tôi phân loại công trình trước đó thành hai phương pháp dựa trên cách học nhúng.

***Học Biểu Diễn Không Giám Sát:***

* Các phương pháp không giám sát có lợi thế không đòi hỏi bất kỳ dữ liệu huấn luyện có nhãn nào, nó chỉ yêu cầu các mã nhị phân được tháo dỡ (tức là có thể xác định được các chỉ thị và ranh giới hàm, và có thể xây dựng đồ thị dòng điều khiển nội hoặc ngoại hoặc của cùng một chương trình).
* DeepBinDiff và Asm2Vec là các nghiên cứu tiêu biểu trong loại học không giám sát này.
* ***DeepBinDiff:*** Sử dụng Word2Vec để nhúng các chỉ thị và các khối cơ bản, và tiếp tục điều chỉnh các nhúng của khối cơ bản với Text-Associated DeepWalk (TADW) để mã hóa vị trí ngữ cảnh của khối cơ bản trong đồ thị dòng điều khiển nội chương trình trên cấp toàn bộ chương trình.
* ***Asm2Vec:*** Thực hiện các bước đi ngẫu nhiên trên đồ thị dòng điều khiển nội của mỗi hàm. Các chuỗi chỉ thị được tạo ra từ các bước đi ngẫu nhiên được sử dụng để huấn luyện mô hình nhúng mã nhị phân cấp hàm dựa trên mô hình PV-DM.

***Học Biểu Diễn Có Giám Sát:***

* Hai mô hình Structure2Vec giống nhau chia sẻ tham số xây dựng mạng Siamese, và mạng được huấn luyện trên các cặp ACFGs và nhãn tương đồng của chúng.
* Sau đó sử dụng các đầu ra của mô hình Structure2Vec đã được huấn luyện làm nhúng cho các hàm.

1. ***Phân Tích Nhị Phân Đa Nền Tảng:***

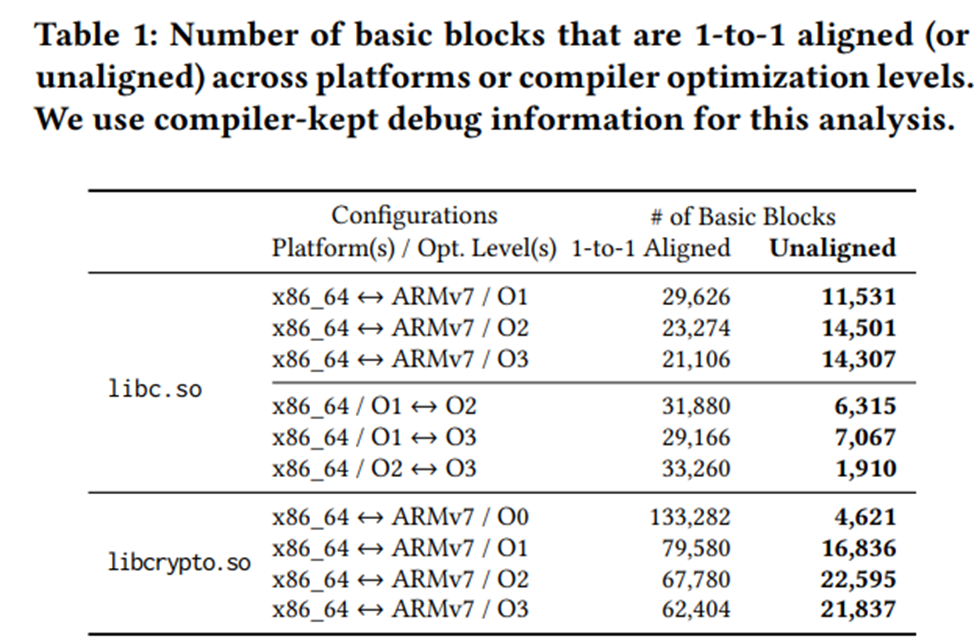
* Lợi ích chính của phân tích nhị phân đa nền tảng là những nỗ lực phân tích được thực hiện trên một nền tảng có thể dễ dàng được chuyển sang các nền tảng khác; ví dụ, chữ ký lỗi chỉ định được định nghĩa một lần duy nhất và việc tìm kiếm lỗi có thể được thực hiện cho các mã nhị phân có sẵn trong các ISA khác nhau.

1. ***Nhúng cho phân tích nhị phân đa nền tảng:***

* Một mô hình nhúng hiểu quan hệ ngữ nghĩa giữa các mã nhị phân biên dịch cho các nền tảng khác nhau sẽ tạo ra các vector nhúng phân biệt.
* Các cặp mã nhị phân có cùng ý nghĩa trên các nền tảng sẽ gần nhau, mặc dù chúng có vẻ khác nhau trên các nền tảng.
* Các cặp không có cùng ý nghĩa sẽ xa nhau.
* Phương pháp học không giám sát có lợi thế không đòi hỏi bất kỳ dữ liệu đánh nhãn nào, nhưng có thể không tạo ra các nhúng nhúng bắt được mối quan hệ ngữ nghĩa giữa mã nhị phân trên các nền tảng.
* Các phương pháp nhúng không giám sát hiện có dựa hoàn toàn vào các đặc trưng có sẵn trong một nền tảng duy nhất trong quá trình huấn luyện, mà không tích hợp bất kỳ thông tin căn chỉnh ngữ nghĩa đa nền tảng nào.
* Cách tiếp cận học có giám sát được sử dụng trong các nghiên cứu trước đây về nhúng mã code qua các kiến trúc tập lệnh đã được chứng minh hiệu quả trong việc huấn luyện mô hình nhúng phân biệt. Phương pháp này yêu cầu một tập dữ liệu huấn luyện có nhãn, trong đó các cặp nhãn mức bộ tự tương đồng.

***Hạn chế của các phương pháp học có giám sát hiện có.***

* Các phương pháp học có giám sát, chẳng hạn như việc huấn luyện một mô hình nhúng sử dụng kiến trúc Siamese, có thể căn chỉnh các cặp nhãn, nhưng các phương pháp hiện có chỉ xem xét việc ghép các khối cơ bản và không thể mã hóa thông tin ngữ cảnh.



* Thứ nhất, như được minh họa trong Hình 1, các phiên bản của cùng một chức năng có thể không giống nhau trên các nền tảng. Mã nguồn cụ thể cho từng nền tảng không thể biên dịch cho các nền tảng khác nhau, do đó không thể đồng nghĩa về mặt ngữ nghĩa giữa các nền tảng mà không cần cố gắng thủ công.
* Thứ hai, trình biên dịch thông thường chỉ tạo ra một phần của tất cả các chỉ thị có sẵn trong một kiến trúc tập lệnh (ISA). Một ISA có thể hỗ trợ quá nhiều chỉ thị (ví dụ: hàng nghìn chỉ thị trong Intel x86\_64), bao gồm cả các chỉ thị không còn được sử dụng cho tương thích ngược (backward compatibility) và / hoặc các chỉ thị được xây dựng cho mục đích cụ thể (ví dụ: chỉ thị AES cho Intel x86\_64 [27] và ARM [3]) thường được sử dụng trong mã assembly viết tay cho mỗi ISA. Mã lắp ráp viết tay, hiển nhiên, không thể biên dịch cho các ISA khác nhau.

1. ***Phương pháp đề xuất: XBA.***

* Chúng tôi giới thiệu XBA, một phương pháp mới dựa trên học sâu giải quyết những hạn chế của các nghiên cứu trước đây, từ đó tạo điều kiện cho phân tích nhị phân đa nền tảng.

***Định dạng đầu vào:***

* Chúng tôi phát triển một định dạng chứa thông tin ngữ cảnh phong phú của các khối cơ bản.
* Công việc trước đó xem xét một khối cơ bản chỉ là một chuỗi các chỉ thị mà không xem xét ngữ cảnh cục bộ của nó.
* Để mã hóa ngữ cảnh của các khối cơ bản vượt xa dòng điều khiển chương trình, chúng tôi đề xuất một định dạng mới, đồ thị disassembly nhị phân (BDG), để biểu diễn các mã nhị phân (Phần 3.1).

***Phương pháp:***

* XBA, sử dụng các mạng gấp đồ thị (GCN), được biết đến để bắt lấy thông tin ngữ cảnh được mã hóa trong các đồ thị (BDGs).

***Huấn luyện:***

* Chúng tôi sử dụng một mục tiêu căn chỉnh đồ thị, một phương pháp học bán giám sát, để huấn luyện các mạng GCN của chúng tôi.

A diagram of a complex network

Description automatically generated

* Hình 2 mô tả tổng quan của XBA.
* Ở phía bên trái, chúng ta thể hiện một ví dụ về mã nguồn có cùng ý nghĩa được triển khai cho Linux và Windows trong cùng một phần mềm, OpenSSL, được viết để có thể di động trên các hệ điều hành.
* XBA nhận hai mã nhị phân được biên dịch cho các nền tảng khác nhau và học (hoặc suy ra) sự tương đồng giữa chúng.
* Chúng ta trước tiên disassemble các mã nhị phân đó và xây dựng các đồ thị disassembly nhị phân (BDG). Các đồ thị này chứa các khối cơ bản, các hàm ngoại vi, các chuỗi và các mối quan hệ giữa chúng.
* Sau đó, chúng ta xử lý các BDG này bằng các mạng gấp đồ thị (GCNs) và tạo ra các embedding cho các thực thể (ví dụ: các khối cơ bản, các hàm ngoại vi hoặc các chuỗi).
* Trong quá trình huấn luyện, chúng ta đảm bảo rằng các GCN tạo ra các embedding cho các thực thể được căn chỉnh gần nhau và các thực thể không được căn chỉnh (được tạo nhân tạo thông qua mẫu) được tách xa.
* Khi được huấn luyện, các GCN của chúng ta tạo ra các embedding cho các thực thể trong BDG (chứa thông tin ngữ cảnh phong phú), hữu ích cho các tác vụ phụ thuộc đa dạng.

1. ***Đồ thị phân tách nhị phân:***

* Phân tích nhị phân trên một mức trừu tượng là phân tích một chuỗi các mã thông báo nhận được từ một mã nhị phân đã được disassemble.
* Trong phân tích đa nền tảng, chúng ta gặp nhiều mã thông báo cụ thể cho từng nền tảng và mối quan hệ ngữ nghĩa của chúng không dễ dàng xác định qua các nền tảng khác nhau.
* Để giải quyết vấn đề này, chúng tôi trừu tượng một mã nhị phân đã disassemble thành một định dạng mới, đồ thị phân tách nhị phân (BDG).
* Từ đó chúng ta có thể rút ra ngữ nghĩa của các mã thông báo cụ thể cho từng nền tảng.
* Một BDG là một đồ thị có loại được gán cho các mối quan hệ của các mã thông báo cụ thể trong một mã nhị phân đã disassemble.
* Một nút trong BDG có thể thuộc một trong các loại sau: Khối cơ bản, hàm ngoại vi và chuỗi chữ.

***Cách xác định các thuộc tính nút như sau:***

* Đối với mỗi khối cơ bản, chúng tôi sử dụng các chuỗi chỉ thị cho thuộc tính nút. Cụ thể hơn, mã ngữ và các toán hạng của mỗi chỉ thị được tìm thấy trong một khối cơ bản cụ thể được chuẩn hóa theo quy tắc được đề ra trong Bảng 4, và chuỗi các mã thông báo đã được chuẩn hóa được sử dụng làm thuộc tính nút của khối cơ bản.

A table with text and symbols

Description automatically generated

* Chúng tôi sử dụng tên ký hiệu của chúng làm thuộc tính nút cho các hàm ngoại vi, các chuỗi chỉ thị của chúng tồn tại bên ngoài một mã nhị phân đã disassemble.
* Chúng tôi sử dụng các chuỗi chữ thông thường được tìm thấy trong một mã nhị phân đã disassemble làm thuộc tính nút.

***Mối quan hệ:***

* Có bốn loại mối quan hệ có thể được xác định giữa các nút của BDG như sau:
* Một truyền dẫn dòng điều khiển trong nội thư hướng, được xác định giữa hai khối cơ bản, chỉ ra rằng điều khiển chương trình chuyển từ một khối cơ bản này sang khối cơ bản khác.
* Cuộc gọi trực tiếp (tức là các truyền dẫn dòng điều khiển giữa các hàm), được xác định giữa một khối cơ bản chứa một hoặc nhiều điểm gọi và các hàm được gọi.
* Chúng tôi cũng xác định 3 các tham chiếu mã-ngôn-ngữ lấy địa chỉ mã-ngôn-ngữ, khi một khối cơ bản chứa một hoặc nhiều chỉ thị lấy địa chỉ của một hàm khác.
* Cuối cùng, 4 các tham chiếu mã-chuỗi được xác định giữa một khối cơ bản và một nút chuỗi, khi khối cơ bản chứa một chỉ thị tham chiếu đến chuỗi chữ.

1. Phân tích mã nhị phân đa nền tảng và Vấn đề canh chỉnh đồ thị:

* Bây giờ chúng tôi trình bày cách phân tích mã nhị phân đa nền tảng có thể được hình thành thành một vấn đề canh chỉnh đồ thị.
* Trong phân tích mã nhị phân đa nền tảng, thách thức chính là xác định các khối cơ bản có chức năng tương tự (hoặc không tương tự) với thông tin không đầy đủ.
* Để ước tính chức năng của chúng, chúng ta cần thông tin ngữ cảnh, chẳng hạn như cùng một khối cơ bản gọi các hàm đó hoặc cùng một đối số được truyền vào chúng.

Vấn đề Canh chỉnh đồ thị:

* Trên các BDG được xây dựng từ các mã nhị phân được biên dịch cho các nền tảng khác nhau, chúng ta nhằm tìm sự tương ứng giữa các nút trong BDG dựa trên một tập con nhỏ của các cặp nút đã được canh chỉnh trước đó.
* Chúng tôi ký hiệu một BDG là 𝐺 = (𝐸, 𝑅, 𝐴, 𝑇 𝑅, 𝑇 𝐴), trong đó 𝐸, 𝑅, 𝐴 là các tập hợp của các thực thể, mối quan hệ và thuộc tính, tương ứng. 𝑇 𝑅 ⊂ 𝐸 × 𝑅 × 𝐸 là tập các ba từ liên quan,𝑇 𝐴 ⊂ 𝐸×𝐴×𝑉 là tập các ba từ thuộc tính và 𝑉 đại diện cho tập giá trị thuộc tính.
* Giả sử rằng hai mã nhị phân được biên dịch từ cùng một nguồn mã cho các nền tảng khác nhau 𝑃1 và 𝑃2. Chúng ta trước tiên xây dựng BDG của chúng 𝐺𝑃1 và 𝐺𝑃2, trong đó 𝐺𝑃𝑖 = (𝐸𝑖, 𝑅𝑖, 𝐴𝑖,𝑇 𝑅 𝑖,𝑇 𝐴 𝑖) cho 𝑖 ∈ {1, 2}.