MỘT CÁCH TIẾP CẬN SỬ DỤNG HỌC SÂU TRONG PHÁT HIỆN MÃ ĐỘC POWERSHELL

Ngô Đức Hoàng Sơn - 230202030

Tóm tắt

- Lớp: CS2205.CH181
- Link Github: https://github.com/pinanek/CS2205.CH181
- Link YouTube video: https://youtu.be/K8I6yYSI_DY
- Họ và Tên: Ngô Đức Hoàng Sơn

Giới thiệu

- Xu hướng phát triển mã độc mới: công cụ có sẵn trên hệ thống.
  - Nổi bật nhất là **PowerShell**.
- Mã độc PowerShell:
  - Được xem nhưng là một chương trình hợp lệ
  - Không để lại dấu vết.

Giới thiệu

- Các phương pháp hiện tại vẫn còn tồn tại nhiều vấn đề:
  - Phương pháp dựa trên quy tắc: Nhanh nhưng bị động.
  - Phương pháp học sâu: Chỉ thực hiện nhúng từ ở mức độ token.
- Phương pháp transformer-based phát triển: DeBERTa.

⇒ Đề xuất phương pháp sử dụng mô hình ngôn ngữ DeBERTa trong phát hiện mã độc PowerShell.

Mục tiêu

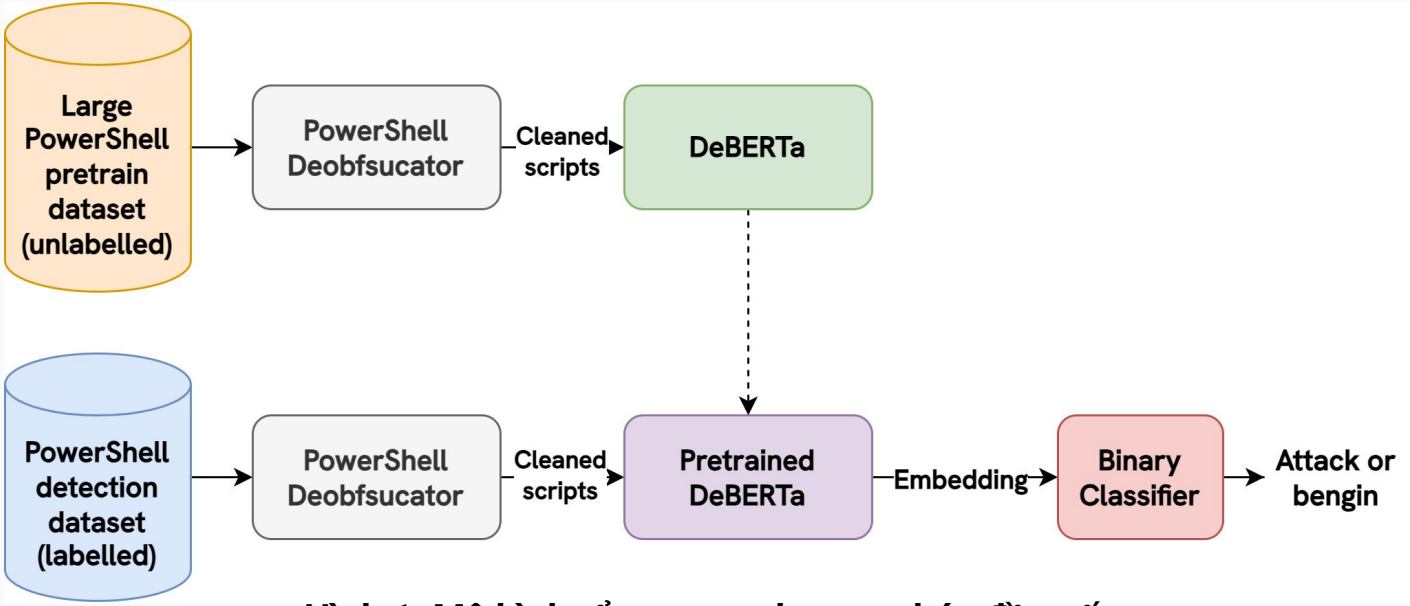
1. Xây dựng được bộ dữ liệu không nhãn và có nhãn về mã nguồn PowerShell.
2. Xây dựng mô hình phân loại dựa trên mô hình ngôn ngữ DeBERTa nhằm phát hiện mã độc PowerShell một cách hiệu quả.
3. Cung cấp một mô hình được tiền huấn luyện nhằm dễ dàng tinh chỉnh cho các tác vụ khác.

Nội dung và Phương pháp

1. Tìm hiểu về mã độc phi mã và các phương pháp phát hiện mã độc PowerShell.
2. Tìm hiểu và xây dựng bộ dữ liệu về mã nguồn PowerShell.
3. Tìm hiểu về các phương pháp gỡ rối mã nguồn PowerShell.

Nội dung và Phương pháp

4. Nghiên cứu và xây dựng mô hình phân loại dựa trên mô hình ngôn ngữ DeBERTa nhằm phát hiện mã độc PowerShell.



Hình 1: Mô hình tổng quan phương pháp đề xuất

Kết quả dự kiến

- Xây dựng được bộ dữ liệu mã nguồn PowerShell.
- Phương pháp gỡ rối có tỉ lệ chính xác từ 90% - 95%.
- Tiền huấn luyện thành công mô hình DeBERTa.
- Mô hình phát hiện mã độc có tỉ lệ phát hiện trên 95%.

Tài liệu tham khảo

[1]. Jannatul Ferdous, Md. Rafiqul Islam, Arash Mahboubi, Md Zahidul Islam:
A Review of State-of-the-Art Malware Attack Trends and Defense Mechanisms. IEEE Access 11: 121118-121141 (2023).

[2]. Side Liu, Guojun Peng, Haitao Zeng, Jianming Fu: A survey on the evolution of fileless attacks and detection techniques. Comput. Secur. 137: 103653 (2024).

[3]. Victor M. Alvarez: Yara: The pattern matching swiss knife. GitHub: VirusTotal/yara. URL: https://github.com/VirusTotal/yara. (2014).

[4]. SigmaHQ: Sigma: The shareable detection format for security professionals. GitHub: SigmaHQ/sigma. URL: https://github.com/SigmaHQ/sigma (2019).

[5]. Yong Fang, Xiangyu Zhou, Cheng Huang: Effective method for detecting malicious PowerShell scripts based on hybrid features☆. Neurocomputing 448: 30-39 (2021).

Tài liệu tham khảo

[6]. Mamoru Mimura, Yui Tajiri: Static detection of malicious PowerShell based on word embeddings. Internet Things 15: 100404 (2021).

[7]. Diksha Khurana, Aditya Koli, Kiran Khatter, Sukhdev Singh: Natural language processing: state of the art, current trends and challenges. Multim. Tools Appl. 82(3): 3713-3744 (2023).

[8]. Zhifeng Xu, Xianjin Fang, Gaoming Yang: Malbert: A novel pre-training method for malware detection. Comput. Secur. 111: 102458 (2021).

[9]. Ferhat Demirkiran, Aykut Çayir, Ugur Ünal, Hasan Dag: An ensemble of pre-trained transformer models for imbalanced multiclass malware classification. Comput. Secur. 121: 102846 (2022).

[10]. Pengcheng He, Jianfeng Gao, Weizhu Chen: DeBERTaV3: Improving DeBERTa using ELECTRA-Style Pre-Training with Gradient-Disentangled Embedding Sharing. ICLR 2023.