UNIVERSITY OF ENGINEERING AND TECHNOLOGY

Nguyễn Lê Việt Hoàng

CẢI THIỆN HIỆU NĂNG THUẬT TOÁN PHÁT HIỆN BẤT THƯỜNG MIDAS-R TRONG AN NINH MẠNG

KHÓA LUẬN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY

Ngành : Điện Tử Viễn Thông

Hanoi, 2023

UNIVERSITY OF ENGINEERING AND TECHNOLOGY

Nguyễn Lê Việt Hoàng

CẢI THIỆN HIỆU NĂNG THUẬT TOÁN PHÁT HIỆN BẤT THƯỜNG MIDAS-R TRONG AN NINH MẠNG

KHÓA LUẬN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY

Ngành : Điện Tử Viễn Thông

Cán bộ hướng dẫn:

Cán bộ đồng hướng dẫn:

Hanoi, 2023

Tuyên thệ của tác giả

"I hereby declare that the work contained in this thesis is of my own and has not been previously submitted for a degree or diploma at this or any other higher education institution. To the best of my knowledge and belief, the thesis contains no materials previously published or written by another person except where due reference or acknowledgement is made."

Signature:……

Xác nhận của cán bộ hướng dẫn

"I hereby approve that the thesis in its current form is ready for committee examination as a requirement for the Bachelor of Computer Science degree at the University of Engineering and Technology."

Signature:……

Lời cảm ơn

And above all, this thesis is for Huong, the one that completes me.

Cải thiện hiệu năng thuật toán phát hiện bất thường MIDAS-R trong an ninh mạng

Tóm tắt đồ án

TÓM TẮT

Mục lục

Danh sách hình vẽ

Danh sách bảng

Chương 1

TỔNG QUAN

Chương này nêu định nghĩa về ngành an ninh mạng, đánh giá thực trạng về an toàn không gian mạng hiện nay và khảo sát các phương pháp được sử dụng trong IDS(Intrusion Detection System) để phát hiện các cuộc tấn công mạng.

1.1 Giới thiệu về an ninh mạng

An ninh mạng là một tập hợp các quy tắc và cấu hình được thiết kế để bảo vệ tính toàn vẹn, bảo mật và khả năng truy cập của mạng máy tính và dữ liệu bằng cả công nghệ phần mềm và phần cứng. Ngành an ninh mạng đang phát triển nhanh chóng do nhu cầu ngày càng tăng đối với các giải pháp và dịch vụ có thể ngăn chặn, phát hiện và giảm thiểu các cuộc tấn công mạng vào các doanh nghiệp, cơ quan chính phủ và cá nhân. Theo nghiên cứu — Các yếu tố chính thúc đẩy sự tăng trưởng của thị trường bao gồm số lượng các mối đe dọa mạng ngày càng tăng, việc áp dụng công nghệ đám mây và di động và nhu cầu làm việc từ xa an toàn.

1.2 Thực trạng an ninh mạng hiện nay

Theo báo cáo của AAG[6] năm 2023, bối cảnh an ninh mạng toàn cầu đã chứng kiến các mối đe dọa gia tăng trong những năm gần đây. Trong đại dịch, tội phạm mạng đã lợi dụng các mạng bị sai lệch khi các doanh nghiệp chuyển sang môi trường làm việc từ xa. Năm 2020, các cuộc tấn công bằng phần mềm độc hại tăng

358% so với năm 2019.

Từ đây, các cuộc tấn công mạng trên toàn cầu đã tăng 125% cho đến năm 2021 và số lượng các cuộc tấn công mạng ngày càng tăng tiếp tục đe dọa các doanh nghiệp và cá nhân vào năm 2022.

Dưới đây là một số thông tin đáng chú ý:

- Lừa đảo(Phishing) vẫn là hình thức phổ biến nhất của tội phạm trực tuyến. Vào năm 2021, 323.972 người dùng internet được cho là nạn nhân của các cuộc tấn công lừa đảo. Điều này có nghĩa là một nửa số người dùng bị vi phạm dữ liệu đã rơi vào một cuộc tấn công lừa đảo.
- Gần 1 tỷ email đã bị lộ trong một năm, ảnh hưởng đến 1/5 người dùng internet.
- Vi phạm dữ liệu khiến các doanh nghiệp thiệt hại trung bình 4,35 triệu đô la vào năm 2022.
- Khoảng 236,1 triệu cuộc tấn công ransomware đã xảy ra trên toàn cầu trong nửa đầu năm 2022.
- Cứ 2 người dùng internet ở Mỹ thì có 1 người bị xâm phạm tài khoản vào năm 2021.
- 39% doanh nghiệp ở Vương quốc Anh cho biết đã bị tấn công mạng vào năm 2022.
- Khoảng 1 trong 10 tổ chức của Hoa Kỳ không có bảo hiểm chống lại các cuộc tấn công mạng.
- 53,35 Công dân Hoa Kỳ bị ảnh hưởng bởi tội phạm mạng trong nửa đầu năm 2022.
- Tội phạm mạng khiến các doanh nghiệp ở Vương quốc Anh thiệt hại trung bình £4200 vào năm 2022.
- Năm 2020, các cuộc tấn công bằng phần mềm độc hại đã tăng 358% so với năm 2019.

1.3 IDS và các phương pháp phát hiện tấn công

1.3.1 IDS - Intrusion Detection System

Hệ thống phát hiện xâm nhập (IDS) là một ứng dụng phần mềm giám sát các hoạt động của mạng hoặc hệ thống và phân tích chúng để tìm các dấu hiệu vi phạm chính sách, cách sử dụng được chấp nhận hoặc các biện pháp bảo mật tiêu chuẩn. Sau đó, nó sẽ báo cáo mọi hoạt động độc hại hoặc vi phạm chính sách cho quản trị viên hệ thống. IDS có thể giúp bảo vệ một tổ chức khỏi các cuộc tấn công mạng bằng cách phát hiện và cảnh báo về các mối đe dọa tiềm ẩn trước khi chúng gây ra thiệt hại.

1.3.2 Các phương pháp phát hiện tấn công trong IDS

Có thể phân loại các IDS theo phương pháp phát hiện như sau:

- IDS dựa trên chữ ký (SIDS) - Signature-based IDS: Hệ thống phát hiện xâm nhập chữ ký (SIDS) dựa trên về các kỹ thuật khớp mẫu để tìm ra một cuộc tấn công đã biết; chúng còn được gọi là Phát hiện dựa trên tri thức - Knowledge-based Detection([1]). Trong SIDS, các phương thức đối sánh được sử dụng để tìm ra sự xâm nhập trước đó. Nói cách khác, khi một chữ ký xâm nhập phù hợp với chữ ký của một lần xâm nhập trước đó đã tồn tại trong cơ sở dữ liệu chữ ký, tín hiệu báo động được kích hoạt. Đối với SIDS, nhật ký của máy chủ được kiểm tra để tìm chuỗi các lệnh hoặc hành động trước đây đã được xác định là phần mềm độc hại.
- IDS dựa trên sự bất thường (AIDS) - Anomaly-based IDS: AIDS đã thu hút sự quan tâm của rất nhiều học giả do nó khả năng khắc phục hạn chế của SIDS. Trong AIDS, một mô hình bình thường của hành vi của một hệ thống máy tính là được tạo bằng cách sử dụng máy học(machine learing), dựa trên thống kê(statistical-based) hoặc phương pháp dựa trên tri thức(knowledge-based). Bất kỳ sai lệch đáng kể nào giữa hành vi được quan sát và mô hình đều được coi là như một sự bất thường, có thể được hiểu là một sự xâm nhập(intrusion). Giả định cho nhóm kỹ thuật này là hành vi ác ý khác với hành vi thông thường của người dùng. Các hành vi của người dùng bất thường không giống với các hành vi tiêu chuẩn được phân loại là xâm nhập. Sự phát triển của AIDS bao

gồm hai giai đoạn: giai đoạn đào tạo và giai đoạn thử nghiệm. Trong giai đoạn đào tạo, bình thường hồ sơ lưu lượng truy cập được sử dụng để tìm hiểu một mô hình hành vi bình thường và sau đó trong giai đoạn thử nghiệm, một bộ dữ liệu mới được sử dụng để thiết lập khả năng khái quát hóa của hệ thống đối với các cuộc xâm nhập chưa từng thấy trước đây. AIDS có thể được phân loại thành một số loại dựa trên phương pháp được sử dụng cho đào tạo, ví dụ, dựa trên thống kê, dựa trên tri thức và dựa trên máy học[7]

Mỗi phương pháp đều có những ưu và nhược điểm riêng, điều này được tóm gọn trong bảng dưới đây:

Bảng 1.3.1: So sánh các phương pháp phát hiện tấn công trong IDS

		Ưu điểm	**Nhược điểm**
Phương pháp phát hiện	SIDS	- Rất hiệu quả trong việc xác định xâm nhập với báo động sai tối thiểu (FA). - Phát hiện kịp thời các hành vi xâm nhập. - Vượt trội để phát hiện các cuộc tấn công đã biết. - Thiết kế đơn giản.	- Cần cập nhật chữ ký mới thường xuyên. - SIDS được thiết kế để phát hiện các cuộc tấn công đối với các chữ ký đã biết. Khi trước đó xâm nhập đã được thay đổi một chút thành một biến thể mới, thì hệ thống sẽ không thể xác định độ lệch mới này của cuộc tấn công tương tự. - Không thể phát hiện cuộc tấn công zero-day. - Không phù hợp để phát hiện các cuộc tấn công nhiều bước. - Ít hiểu biết sâu sắc về các cuộc tấn công
	AIDS	- Có thể được sử dụng để phát hiện các cuộc tấn công mới. - Có thể được sử dụng để tạo chữ ký xâm nhập.	- AIDS không thể xử lý các gói được mã hóa, vì vậy cuộc tấn công có thể không bị phát hiện và có thể đưa ra một mối đe dọa. - Báo động dương tính giả cao. - Khó xây dựng một hồ sơ bình thường cho một hệ thống máy tính rất năng động. - Cảnh báo chưa được phân loại. - Cần đào tạo ban đầu.

Gần đây, một thuật toán phát hiện bất thường mới được đề xuất được gọi là

MIDAS hứa hẹn mang ưu điểm của cả hai nhóm trên đồng thời hạn chế nhược điểm của các phương pháp được sử dụng trong AIDS.

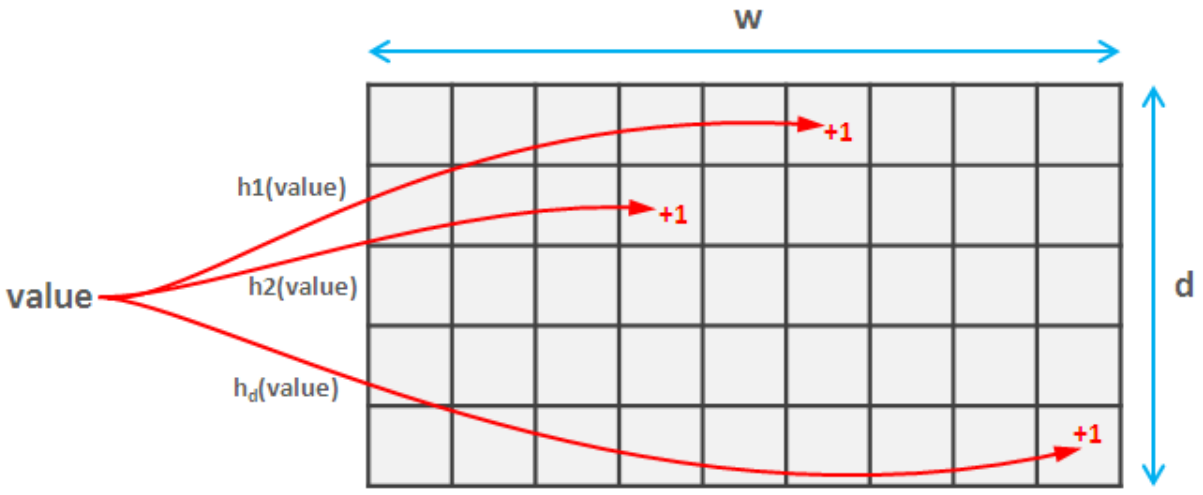
1.4 MIDAS - Microcluster-Based Detector of Anomalies in Edge Streams

Phần này sẽ giới thiệu về CMS, cấu trúc dữ liệu phác thảo được sử dụng trong MIDAS. Tiếp đến là tổng quan về MIDAS và biến thể của nó, MIDAS-R. Cuối cùng là đánh giá hiệu quả của chúng.

1.4.1 Count-Min Sketch - Bản phác thảo đếm tối thiểu

Câu hỏi đặt ra là: Cho một chuỗi dữ liệu liên tục, làm thế nào để biết số lần xuất hiện của một giá trị bất kỳ trong chuỗi đó?

CMS - Count-Min Sketch[4] là một cấu trúc dữ liệu trả lời câu hỏi trên với kích thước bộ nhớ cố định, thời gian truy vấn không đổi và lưu trữ gần đúng số lần xuất hiện của các giá trị trong chuỗi dữ liệu. Ý tưởng cơ bản của Count-Min Sketch khá đơn giản, nó chỉ là một mảng hai chiều $(d \times w)$ của các bộ đếm số nguyên. Khi một giá trị đến, nó được ánh xạ tới một vị trí tại mỗi d hàng bằng cách sử dụng d hàm băm khác nhau. Bộ đếm trên mỗi vị trí được tăng lên. Quá trình này được thể hiện trong hình dưới đây:



Hình 1.4.1: Minh họa Count-Min Sketch

Để truy vấn, ta chỉ cần trả về giá trị nhỏ nhất của các bộ đếm tại các vị trí được ánh xạ tới giá trị đó.

Sự phụ thuộc giữa kích thước bản phác thảo và độ chính xác[5] được thể hiện trong công thức bên dưới với chiều rộng w và độ sâu d:

$$w = \left\lceil \frac{2}{\epsilon} \right\rceil \qquad \epsilon \text{ :Lỗi ước tính}$$
$$d = \left\lceil \frac{\ln(1-\delta)}{\ln \frac{1}{2}} \right\rceil \qquad \delta \text{ :độ tin cậy}$$

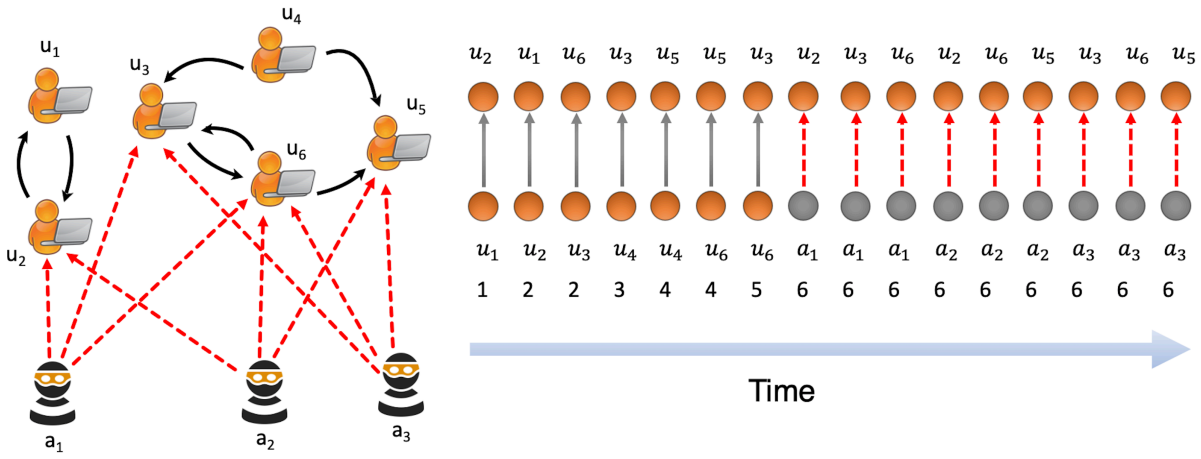
Tóm lại hoạt động của CMS gồm:

- **Khởi tạo(Init):**
  - Bước 1: Tạo một bản phác thảo CMS với kích thước $d \times w$.
  - Bước 2: Đặt tất cả các bộ đếm(hay phần tử trong mảng) bằng 0.
- **Thêm giá trị(Update):**
  - Bước 1: Tạo các giá trị băm sử dụng d hàm băm khác nhau.
  - Bước 2: Ánh xạ các giá trị băm đó tới một vị trí tại mỗi hàng.
  - Bước 3: Tăng bộ đếm tại các vị trí đó lên.
- **Truy vấn giá trị(Query):**
  - Bước 1: Tạo các giá trị băm sử dụng d hàm băm khác nhau.
  - Bước 2: Ánh xạ các giá trị băm đó tới một vị trí tại mỗi hàng.
  - Bước 3: Trả về giá trị nhỏ nhất của các bộ đếm tại các vị trí đó.

1.4.2 Sơ lược về thuật toán MIDAS

MIDAS được thiết kế nhằm trả lời câu hỏi sau: Cho một luồng các kết nối liên tục, làm thế nào để biết các sự kiện đó bất thường hay không?

Trong trường hợp này, các tác giả lập mô hình các nút(Node) như máy tính xách tay, máy tính để bàn là đỉnh của đồ thị, và một kết nối giữa chúng như là một cạnh(Edge).

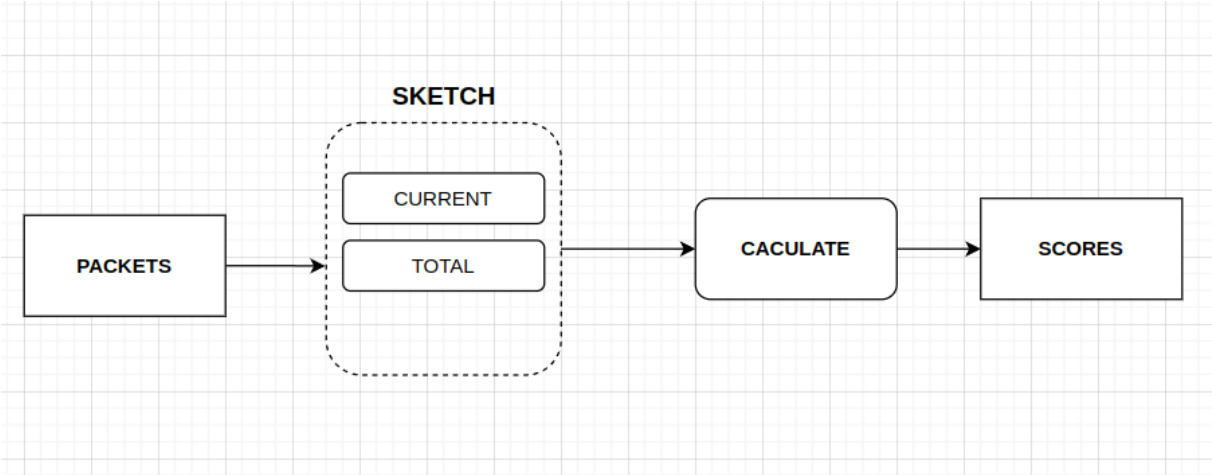


Hình 1.4.2: Minh họa cuộc tấn công mạng

Các sự kiện gian lận hoặc bất thường trong nhiều ứng dụng xảy ra trong các cụm vi mô(Microcluster) hoặc đột nhiên đến các nhóm có cạnh giống nhau đáng ngờ, ví dụ: tấn công từ chối dịch vụ(Dos) trong dữ liệu lưu lượng mạng. Các phương thức hiện có xử lý các luồng cạnh trong một cách trực tuyến nhằm mục đích phát hiện các cạnh bất ngờ riêng lẻ, không phải các cụm vi mô và do đó có thể bỏ lỡ một lượng lớn hoạt động đáng ngờ.

Thuật toán MIDAS sử dụng bản phác thảo đếm tối thiểu (CMS) để đếm số lần xuất hiện trong mỗi dấu thời gian, sau đó sử dụng kiểm tra chi bình phương để đánh giá mức độ sai lệch và tạo ra một số điểm đại diện cho sự bất thường. Điểm càng cao thì độ bất thường của cạnh càng lớn. Phương pháp được đề xuất sử dụng bộ nhớ không đổi và có độ phức tạp thời gian không đổi khi xử lý từng cạnh. Ngoài ra, bằng cách sử dụng một nguyên tắc khuôn khổ thử nghiệm giả thuyết, MIDAS cung cấp các giới hạn lý thuyết về giả thuyết xác suất dương mà những phương pháp tương tự khác ([3],[2]) không cung cấp.

Hình 1.4.2 minh họa cách MIDAS tính điểm bất thường:



Hình 1.4.3: Sơ đồ hoạt động MIDAS

Trong đó:

- PACKETS: danh sách các gói tin mô phỏng một mạng thực.
- SKETCH: duyệt qua các gói tin và sử dụng CMS để ước lượng số lần xuất hiện của từng cạnh trong cùng thời điểm và trong toàn bộ thời gian chạy.
  - CURRENT: đếm số lượng cạnh trong thời điểm hiện tại.
  - TOTAL : đếm tổng số cạnh từ khi băt đầu theo dõi.
- CALCULATE: tính toán điểm bất thường cho từng gói tin bằng kiểm định giả thiết Chi-Square để xác định sự tương quan giữa các mạng.
- SCORES: điểm bất thường của từng gói tin.

Algorithm 1: MIDAS: Streaming Anomaly Scoring

Data: Stream of graph edges over time

Output: Anomaly scores per edge

▷ **Initialize CMS data structures:**

Initialize CMS for total count s_{uv} and current count a_{av}

while *new edge* $e = (u, v, t)$ *is received* : **do**

  ▷ **Update Counts:**

  Update CMS data structures for the new edge uv

  ▷ **Query Counts:**

  Retrieve updated counts $\hat{s}_{uv}$ and $\hat{a}_{uv}$

  ▷ **Anomaly Score:**

  output score$(u, v, t) = (\hat{a}_{uv} - \frac{\hat{s}_{uv}}{t})^2(\frac{t^2}{\hat{s}_{uv}(t-1)})$

end

Algorithm 1 mô tả cách hoạt động của MIDAS, CMS được xóa sau mỗi lần thay đổi dấu thời gian. Tuy nhiên, một số bất thường vẫn tồn tại trong nhiều dấu thời gian. Các tác giả của MIDAS đã đề xuất một biến thể được gọi là MIDAS-R có thể duy trì số lượng một phần của dấu thời gian trước đó cho phép tiếp theo thuật toán để nhanh chóng tạo ra điểm cao khi cạnh xuất hiện trở lại. Nó cũng cũng coi các nút nguồn và đích là thông tin bổ sung giúp xác định các cạnh bất thường.

1.4.3 Thuật toán MIDAS-R

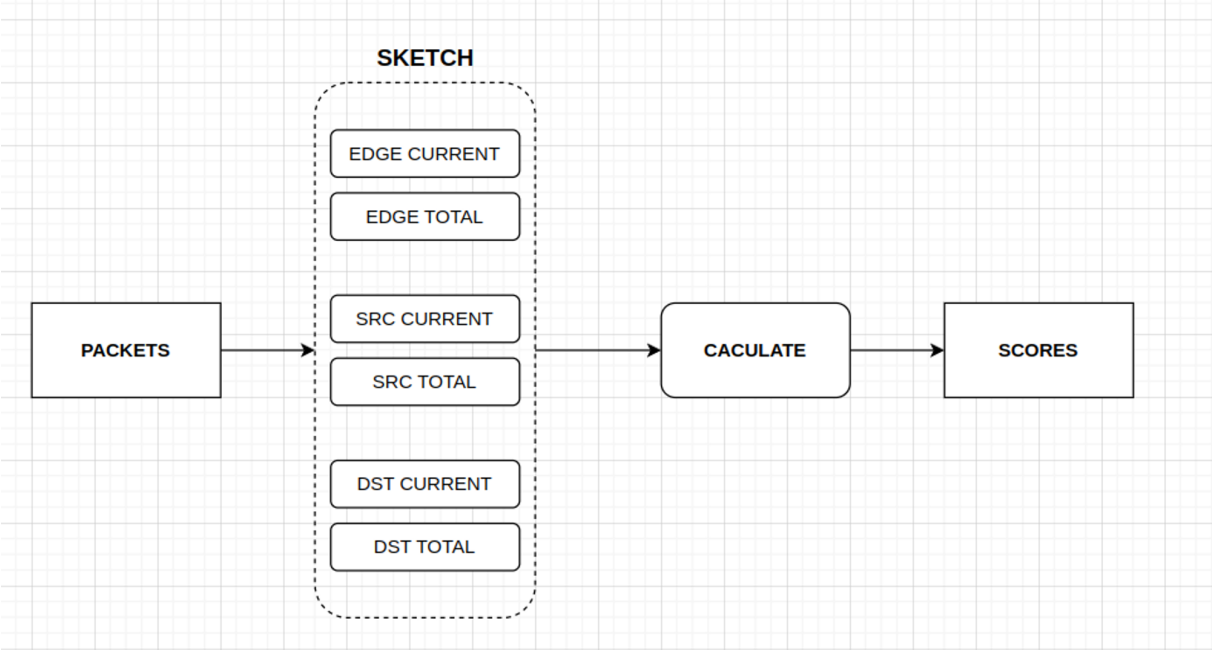
Phần này mô tả cách tiếp cận của MIDAS-R[9], phương pháp này xem xét các cạnh liên quan: nghĩa là, nó nhằm mục đích nhóm các cạnh gần nhau lại với nhau, hoặc theo thời gian hoặc không gian.

Quan hệ tạm thời: Thay vì chỉ đếm các cạnh trong cùng một khoảng thời gian (như đã làm ở MIDAS), các cạnh trong quá khứ gần đây cũng sẽ được tính vào dấu thời gian hiện tại, nhưng được sửa đổi bằng cách giảm cân nặng. Một cách đơn giản và hiệu quả để thực hiện việc này bằng cấu trúc dữ liệu CMS như sau: vào cuối mỗi lần đánh dấu, thay vì đặt lại cấu trúc dữ liệu CMS đối với a_{uv}, ta sẽ chia tỷ lệ tất cả số lượng của nó theo một phân số cố định $\alpha \in (0, 1)$. Điều này cho phép các cạnh trong quá khứ để tính vào dấu tích thời gian hiện tại, với trọng số giảm dần. Lưu ý rằng không xem xét 0 hoặc 1, vì 0 xóa tất cả các giá trị trước đó

khi dấu thời gian thay đổi và do đó không bao gồm bất kỳ hiệu ứng tạm thời nào; và 1 không chia tỷ lệ dữ liệu CMS cấu trúc ở tất cả.

Quan hệ không gian: Nắm bắt các nhóm lớn không gian gần đó các cạnh: ví dụ: một địa chỉ IP nguồn duy nhất đột nhiên tạo ra một số lượng lớn các cạnh đến nhiều đích hoặc một nhóm nhỏ các nút đột nhiên tạo ra một số lượng lớn các cạnh giữa chúng. Một trực giác đơn giản là ở một trong hai hai trường hợp này, quan sát các nút có sự xuất hiện đột ngột của một lượng lớn số cạnh. Do đó, chúng ta có thể sử dụng cấu trúc dữ liệu CMS để theo dõi cạnh đếm như trước, ngoại trừ đếm tất cả các cạnh liền kề với bất kỳ nút u. Cụ thể, tạo bộ đếm CMS cho $\hat{a}_u$ và $\hat{s}_u$ để tính gần đúng số lượng cạnh hiện tại và tổng số liền kề với nút u. Với mỗi cạnh tới (u, v), chúng ta có thể tính toán ba điểm bất thường: một cho cạnh (u, v), như trong thuật toán trước đây; một cho nút nguồn u và một cho nút đích v. Cuối cùng là kết hợp ba điểm bằng cách lấy giá trị lớn nhất của chúng.

Hình 1.4.2 minh họa cách MIDAS-R tính điểm bất thường:



Hình 1.4.4: Sơ đồ hoạt động của MIDAS-R

Các khối trong hình 1.4.4 hoàn toàn tương tự như trong MIDAS được đề cập ở hình 1.4.3, nhưng bổ sung thêm 4 khối:

- SRC CURRENT: đếm số lượng nút nguồn hiện tại.
- SRC TOTAL: đếm tổng số nút nguồn từ khi bắt đầu theo dõi.

- DST CURRENT: đếm số lượng nút đích hiện tại.
- DST TOTAL: đếm tổng số nút đích từ khi bắt đầu theo dõi.

Hoạt động của MIDAS-R được mô tả trong **Algorithm 2** dưới đây:

Algorithm 2: MIDAS-R: Incorporating Relations

Data: Stream of graph edges over time

Output: Anomaly scores per edge

▷ **Initialize CMS data structures:**

Initialize CMS for total count s_{uv} and current count a_{uv}

Initialize CMS for total count s_u, s_v and current count a_u, a_v

while *new edge* $e = (u, v, t)$ *is received* : **do**

  ▷ **Update Counts:**

  Update CMS data structures for the new edge uv, source node u and destination node v

  ▷ **Query Counts:**

  Retrieve updated counts $\hat{s}_{uv}$ and $\hat{a}_{uv}$

  Retrieve updated counts $\hat{s}_u, \hat{s}_v$ and $\hat{a}_u, \hat{a}_v$

  ▷ **Compute Edge Score:**

  $\text{score}((u, v, t)) = (\hat{a}_{uv} - \frac{\hat{s}_{uv}}{t})^2(\frac{t^2}{\hat{s}_{uv}(t-1)})$

  ▷ **Compute Node Scores:**

  $\text{score}((u, t)) = (\hat{a}_u - \frac{\hat{s}_u}{t})^2(\frac{t^2}{\hat{s}_{uv}(t-1)})$

  $\text{score}((v, t)) = (\hat{a}_v - \frac{\hat{s}_v}{t})^2(\frac{t^2}{\hat{s}_{uv}(t-1)})$

  ▷ **Final Score:**

  output $\max\{\text{score}(u, v, t), \text{score}(u, t), \text{score}(v, t)\}$

end

1.4.4 Đánh giá và nêu vấn đề

Các tác giả của MIDAS đã tiến hành đo lường[8], kết quả được mô tả trong bảng sau:

Bảng 1.4.1: Độ chính xác(độ lệch chuẩn)

Dataset	PEN miner	F-FADE	SEDAN SPOT	MIDAS	MIDAS-R
DARPA	0.8267	0.8451	0.6442	0.9042(0.0032)	**0.9514**(0.0012)
CTU-13	0.6041	0.8028	0.6397	0.9079(0.0049)	**0.9703**(0.0009)
UNSW -NB15	0.7028	0.6858	0.7575	0.8843(0.0079)	**0.8952**(0.0028)

Bảng 1.4.2: Thời gian chạy

Dataset	PEN miner	F-FADE	SEDAN SPOT	MIDAS	MIDAS-R
DARPA	20423s	325.1s	67.54s	**0.09**s	0.30s
CTU-13	10065s	844.2s	38.73s	**0.05**s	0.21s
UNSW -NB15	12857s	2267s	48.03s	**0.06**s	0.15s

Từ các khái niệm và hai bảng nêu trên, có thể đưa ra nhưng ưu diểm sau của MIDAS-R:

- Có độ chính xác vượt trội so với các thuật toán khác.
- Chỉ cần sử dụng dữ liệu địa chỉ IP và dấu thời gian để phát hiện các bất thường thay vì cần rất nhiều trường như trong các phương pháp dựa trên học máy, thống kê, tri thức...
- Thời gian chạy nhanh hơn nhiều so với những thuật toán còn lại(trừ MIDAS).
- Không tiêu thụ thêm bộ nhớ trong thời gian chạy.

Tuy nhiên, MIDAS-R còn tồn tại các vấn đề sau:

- Thời gian xử lý chậm hơn MIDAS tới 3 lần.

- Chỉ có thể xử lý tuần tự tất cả các gói tin, điều này là không khả thi trong môi trường có lưu lượng mạng lớn và thay đổi đột ngột, nhất là trong giai đoạn bùng nổ truy cập Internet như hiện nay.

1.5 Nội dung đồ án

1.5.1 Mục tiêu và phương pháp

- Mục tiêu:
  - Cải thiện hiệu năng thuật toán MIDAS-R ít nhất 40% so với thuật toán ban đầu trong khi vẫn đảm bảo độ chính xác.
  - Bổ sung khả năng thích ứng trong điều kiện lưu lượng mạng thay đổi theo thời gian.
- Phương pháp: Để giải quyết vấn đề này, đầu tiên cần tiến hành phân tích điểm nghẽn hiệu năng tại các bước xử lý trong MIDAS-R và đưa ra các cách cải tiến phù hợp.

Note: Trong khuôn khổ đồ án này, em chỉ tập trung cải tiến thuật toán MIDAS-R. Tuy nhiên các phương pháp cải thiện được đề xuất hoàn toàn có thể áp dụng cho các thuật toán MIDAS và các biến thể của nó.

1.5.2 Phạm vi của đồ án

- Toàn bộ việc triển khai thuật toán MIDAS-R được thực hiện hoàn toàn trên máy tính.
- Dữ liệu tấn công đã được tiền xử lý dưới dạng file CSV thu thập từ các các tổ chức an ninh mạng.

1.5.3 Phần mềm và công cụ

- Ngôn ngữ lập trình C được sử dụng để triển khai thuật toán.
- Ngôn ngữ script Python dùng cho tổng hợp và trực quan hóa số liệu.

- Công cụ Make để tự động hóa quá trình biên dịch và liên kết.
- Đo lường hiệu suất: perf, flamegraph.
- Môi trường triển khai trên hệ điều hành Linux.

Chương 2

PHÂN TÍCH VÀ CẢI THIỆN THUẬT TOÁN

2.1 Đo lường, phân tích các nút thắt cổ chai và đề xuất cải tiến

2.2 Bản phác thảo NitroSketch

2.3 Tạo hàm băm hiệu quả

2.4 Các phương pháp khác

Chương 3

ĐO LƯỜNG, ĐÁNH GIÁ VÀ KẾT LUẬN

3.1 Chuẩn bị

3.2 Độ chính xác

3.3 Hiệu năng

3.4 Kết luận

Tài liệu tham khảo

[1] Peter Vamplew Ansam Khraisat, Igbal Gondal. An anomaly intrusion detection system using c5 decision tree classifier. in: Trends and applications in knowledge discovery and data mining. *Springer*, 2018.

[2] Matthieu Latapy Audrey Wilmet, Tiphaine Viard. Degree-based outliers detection within ip traffic modelled as a link stream. *Network Traffic Measurement and Analysis Conference(TMA)*, 2018.

[3] Sourav Sikdar Dimitrije Jankov. Real-time high performance anomaly detection over data streams: Grand challenge. *DEBS*, 2017.

[4] S. Muthukrishnan Graham Cormode. An improved data stream summary:the count-min sketch and its applications. *Journal of Algorithms*, 2005.

[5] S. Muthukrishnan Graham Cormode. Approximating data with the count-min data structure. *abc*, 2011.

[6] Charles Griffiths. The latest 2023 cyber crime statistics (updated march 2023). Technical report, AAG, 2023.

[7] Ravi Sankar Ismail Butun, Salvatore D. Morgera. A survey of intrusion detection systems in wireless sensor networks. *IEEE*, 16, 2014.

[8] Minji Yoon Siddharth Bhatia, Bryan Hooi. Midas: Microcluster-based detector of anomalies in edge streams. *abc*, 2020.

[9] Minji Yoon Siddharth Bhatia, Bryan Hooi. Real-time anomaly detection in edge streams. *ACM*, 2022.

Phụ lục A

This is title of appendix A

Phụ lục B

This is title of appendix B