**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

**KHOA CÔNG NGHỆ THÔNG TIN 2**

**NGÀNH AN TOÀN THÔNG TIN**

****

**Môn: Kỹ thuật theo dõi, giám sát an toàn mạng**

**Đề tài: Tiến hành tấn công từ chối dịch vụ (DOS) UDP Flood và phòng thủ bằng Proxy và Snort & tường lửa Pfsense**

**GVHD: Nguyễn Xuân Sâm**

**Nhóm sinh viên thực hiện:**

|  |  |  |
| --- | --- | --- |
| Họ và tên | Mã số sinh viên | Lớp |
| Nguyễn Minh Hoàng | N17DCAT029 | D17CQAT01-N |
| Bùi Minh Thuận | N17DCAT069 | D17CQAT01-N |

1. **UDP FLOOD DOS ATTACK**
2. ***UDP Flood DOS là gì?***

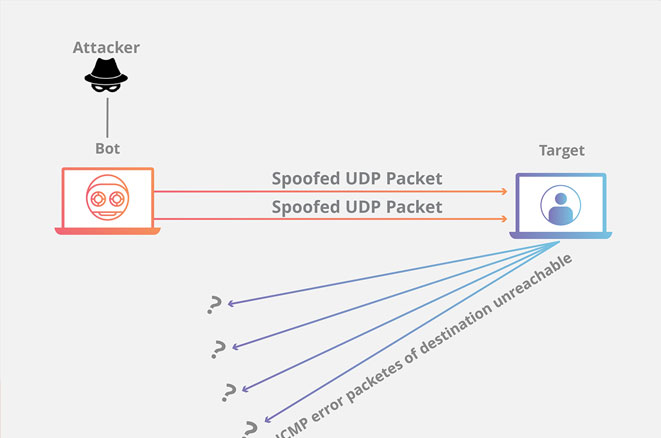
UDP flood là một dạng tấn công từ chối dịch vụ (denial-of-service) với một số lượng lớn gói tin User Datagram Protocol (UDP) được gửi về máy chủ với mục đích làm quá tải khả năng xử lý và phản hồi của thiết bị. Cho dù máy chủ có được bảo vệ bởi một tường lửa thì không có nghĩa máy chủ sẽ có khả năng chống lại một cuộc tấn công DOS UDP flood đủ lớn, thiết bị tường lửa có thể bị cạn tài nguyên bởi loại tấn công này, dẫn tới việc không thể phân biệt được những gói tin hợp lệ.

1. ***UDP Flood DOS attack hoạt động như thế nào?***

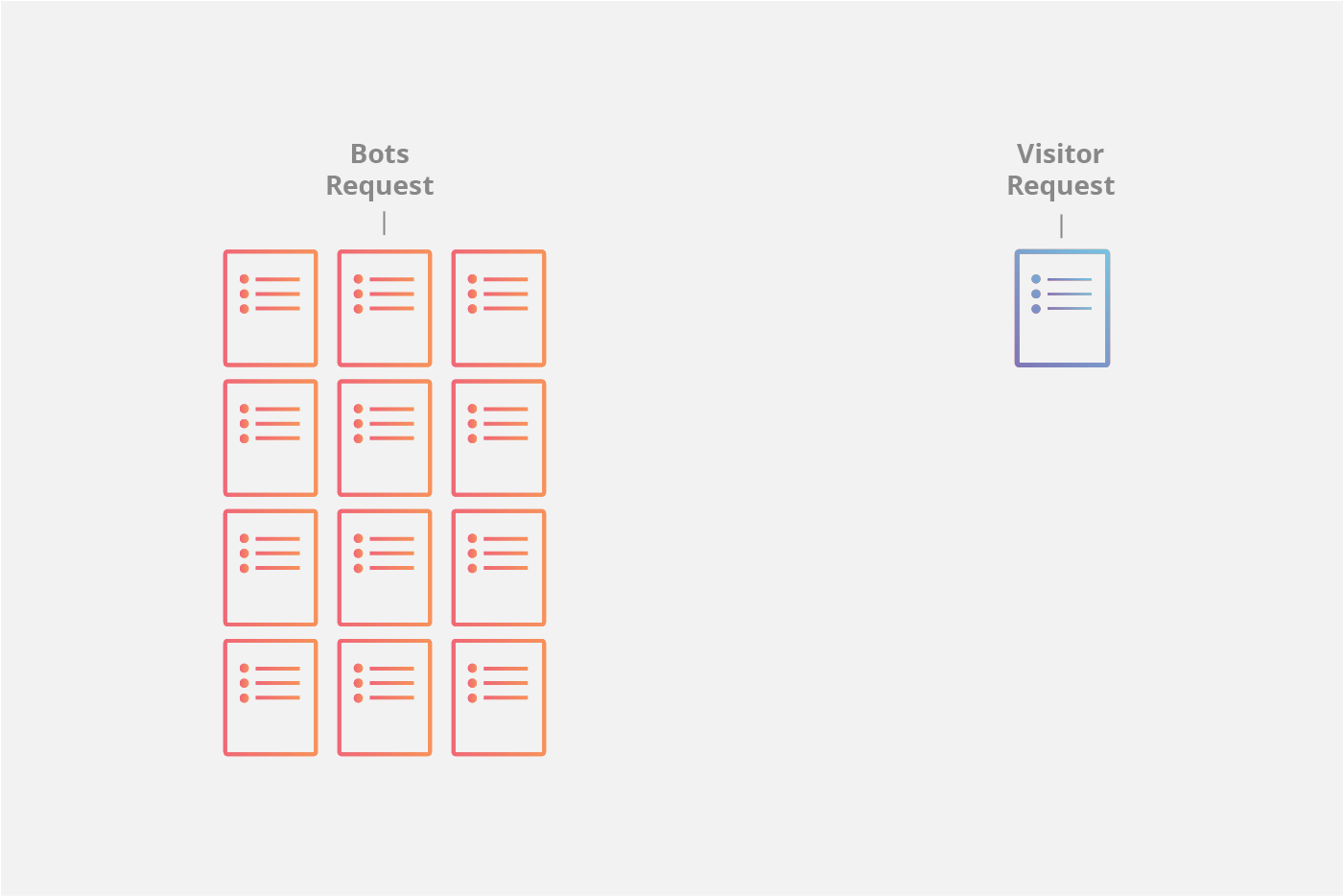
Tấn công UDP bằng cách khái thác các bước mà server thực hiện khi nó phản hồi packet UDP được gửi đến một trong số các port của Client. Trong điều kiện bình thường server nhận packet UDP tại 1 port cụ thể, phản hồi qua 2 bước như sau:

**Bước 1**: Trước tiên, server kiểm tra xem có các chương trình nào đang chạy hay không, hiện tại đang lắng nghe các port nào được chỉ định của chương trình .

**Bước 2**: Nếu không có chương trình nào nhận packet tại port, thì server sẽ phản hồi với packet ICMP (ping) để thông báo cho người gửi rằng đích không thể truy cập được.



Tấn công UDP flood có thể được hình dung đến trong bối cảnh các cuộc gọi định tuyến của nhân viên lễ tân khách sạn. Đầu tiên, nhân viên tiếp tân nhận được một cuộc gọi điện thoại trong đó người gọi yêu cầu được kết nối cuộc gọi với một phòng cụ thể. Sau đó nhân viên tiếp tân cần xem qua danh sách tất cả các phòng để đảm bảo rằng khách có mặt trong phòng và sẵn sàng nhận cuộc gọi. Khi nhân viên lễ tân biết rằng không có khách trong phòng để nhận cuộc gọi, họ phải gọi lại cho người gọi , nói rằng khách không có ở phòng để nhận cuộc gọi. Nếu các line điện thoại sáng lên cùng lúc với các yêu cầu tương tự thì chúng sẽ nhanh chóng trở nên quá tải.



Khi mỗi packet UDP được máy chủ tiếp nhận, nó phải trải qua các bước để xử lý yêu cầu, sử dụng nguồn tài nguyên của server cho quá trình xử lý. Khi các packets UDP được gửi đi , mỗi packets sẽ bao gồm địa chỉ IP của thiết bị nguồn. Trong kiểu tấn công DoS này, kẻ tấn công thường không sử dụng địa chỉ IP thực của họ mà thay vào đó sẽ dùng địa chỉ IP giả mạo nguồn của các packets UDP. Ngăn chặn vị trí thật của kẻ tấn công bị lộ và có khả năng hòa lẫn các gói phản hồi từ máy chủ mục tiêu.

Do server mục tiêu sử dụng tài nguyên để kiểm tra và phản hồi từng packets UDP đã nhận , tài nguyên của server mục tiêu có thể nhanh chóng cạn kiệt khi nhận được một lượng lớn các packets UDP, kết quả việc tấn công Ddos đối với lưu lượng bình thường.

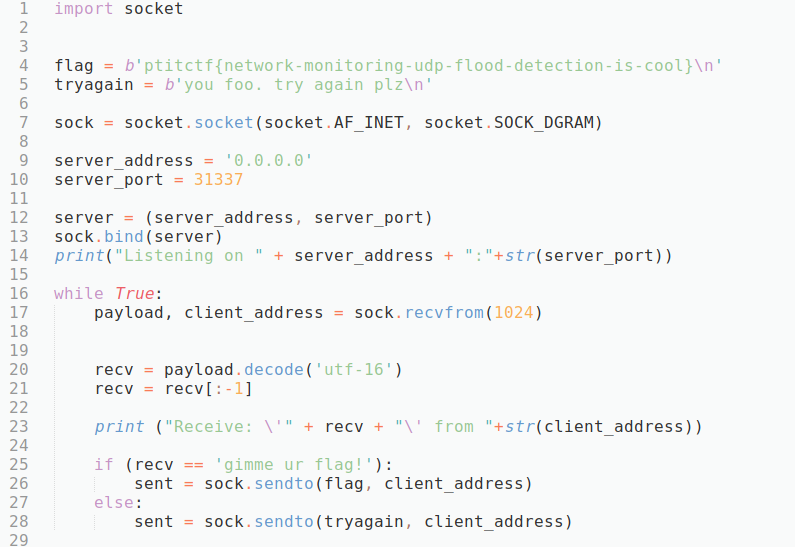
1. **Làm thế nào để giảm thiểu nguy cơ bị tấn công DOS UDP Flood?**

Hầu hết các hệ điều hành có giới hạn tốc độ phản hồi của các packets ICMP một phần để phá vỡ các cuộc tấn công DDoS yêu cầu phản hồi ICMP. Một nhược điểm của kiểu giảm thiểu này là trong một cuộc tấn công, các packets hợp pháp cũng có thể được lọc trong qúa trình này. Nếu cuộc tấn công UDP flood với khối lượng đủ lớn để cân bằng trạng thái firewall của server mục tiêu. Mọi sự giảm thiểu xảy quá mức của server không đủ xảy ra tình trạng nghẽn nút cổ chai từ server mục tiêu.

1. **Phân tích gói tin nhận được trong quá trình máy chủ bị tấn công UDP Flood**

Đặt trường hợp máy chủ đang chạy một chương trình chứa một thử thách trong cuộc thi CTF (Capture-The-Flag, một dạng cuộc thi phổ biến trong ngành An Toàn Thông Tin). Trong thực tế, việc tổ chức cuộc thi CTF cũng gặp rất nhiều trường hợp bị DDOS dẫn đến việc chất lượng của cuộc thi bị giảm đi rất nhiều, ảnh hưởng lớn tới thí sinh và BTC).

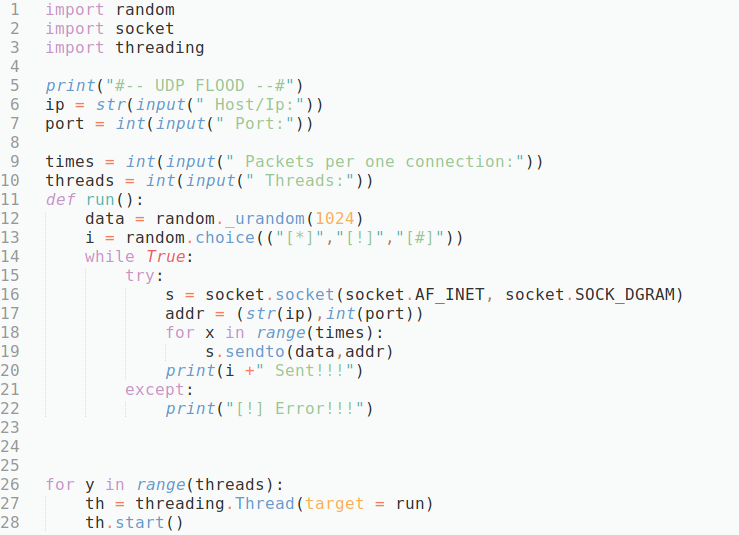
*Mã nguồn Python của thử thách CTF đơn giản như sau*:



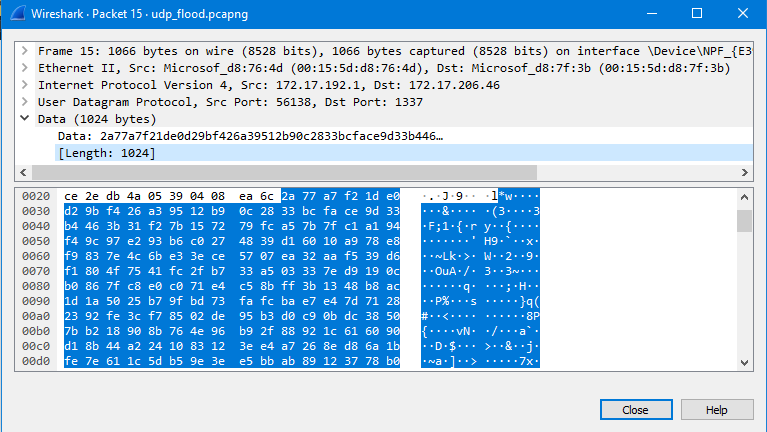
Theo lý thuyết, việc tấn công DOS UDP Flood khá đơn giản, gửi càng nhiều gói tin chứa càng nhiều dữ liệu càng tốt tới một cổng, ở đây là cổng **31337** , nên việc thiết kế thuật toán bằng Python sẽ không mất quá nhiều thời gian.

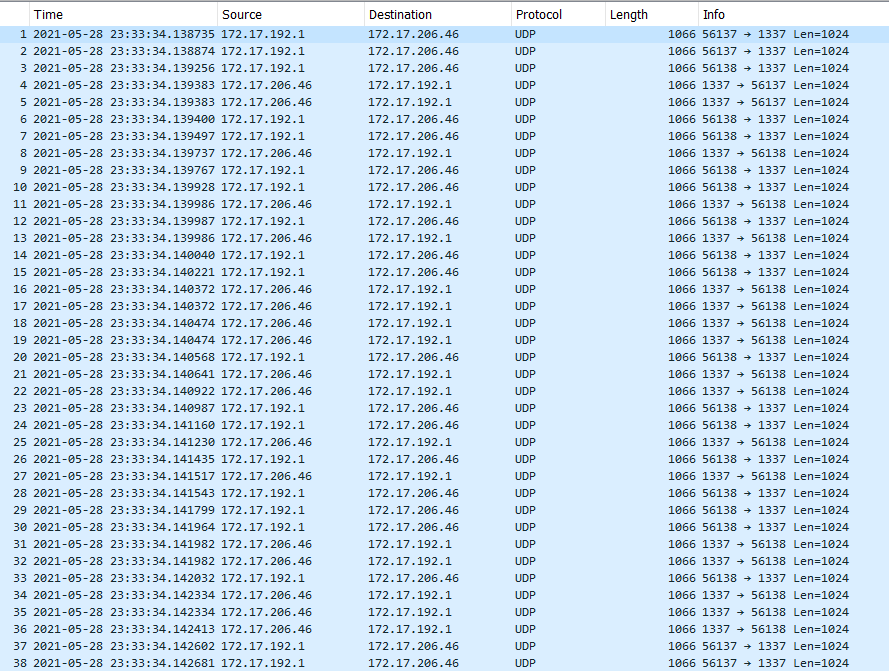
Trong thực tế, để tấn công các máy chủ lớn, các Attacker sẽ sử dụng nhiều máy tính để thực hiện tấn công từ chối dịch vụ DOS cùng lúc, đây còn gọi là các máy zombie trong một hệ thống bot-net, các máy zombie này hầu hết đều đã bị nhiễm mã độc thông qua một hoặc nhiều lỗ hổng bảo mật, và Attacker có quyền điều khiển các máy tính này từ xa.

*Mã nguồn Python demo tấn công UDP Flood DOS*:



Sau khi khởi động chương trình cho thử thách CTF trên, và thực hiện tấn công UDP, do máy chủ và bản thân chương trình UDP server không có bất cứ phương pháp bảo vệ nào, nên máy chủ nhanh chóng bị quá tải và không thể xử lý những tác vụ tưởng chừng như rất đơn giản. Thu thập gói tin bằng chương trình *Wireshark* và thực hiện phân tích:





*Hình ảnh gói tin được truyền qua lại giữa máy chủ CTF và máy attacker*

Dựa vào thông tin khá trực quan từ Wireshark, dễ nhận thấy máy chủ phải thực hiện truyền nhận dữ liệu đồng thời phải xử lý rất nhiều dữ liệu ngẫu nhiên cùng lúc, từ đó dẫn tới quá tải.

1. **Phòng chống tấn công DoS UDP Flood bằng cách tính Entropy của gói tin.**
2. **Entropy là gì?**

Trong khoa học máy tính, entropy là độ ngẫu nhiên được hệ điều hành hoặc ứng dụng thu thập để sử dụng trong mật mã hoặc các mục đích sử dụng khác yêu cầu dữ liệu ngẫu nhiên. Tính ngẫu nhiên này thường được thu thập từ các nguồn phần cứng (phương sai về tiếng ồn của quạt hoặc ổ cứng), hoặc là những nguồn có sẵn từ trước như chuyển động của chuột hoặc bộ tạo ngẫu nhiên được cung cấp đặc biệt. Việc thiếu entropy có thể có tác động tiêu cực đến hiệu suất và bảo mật.

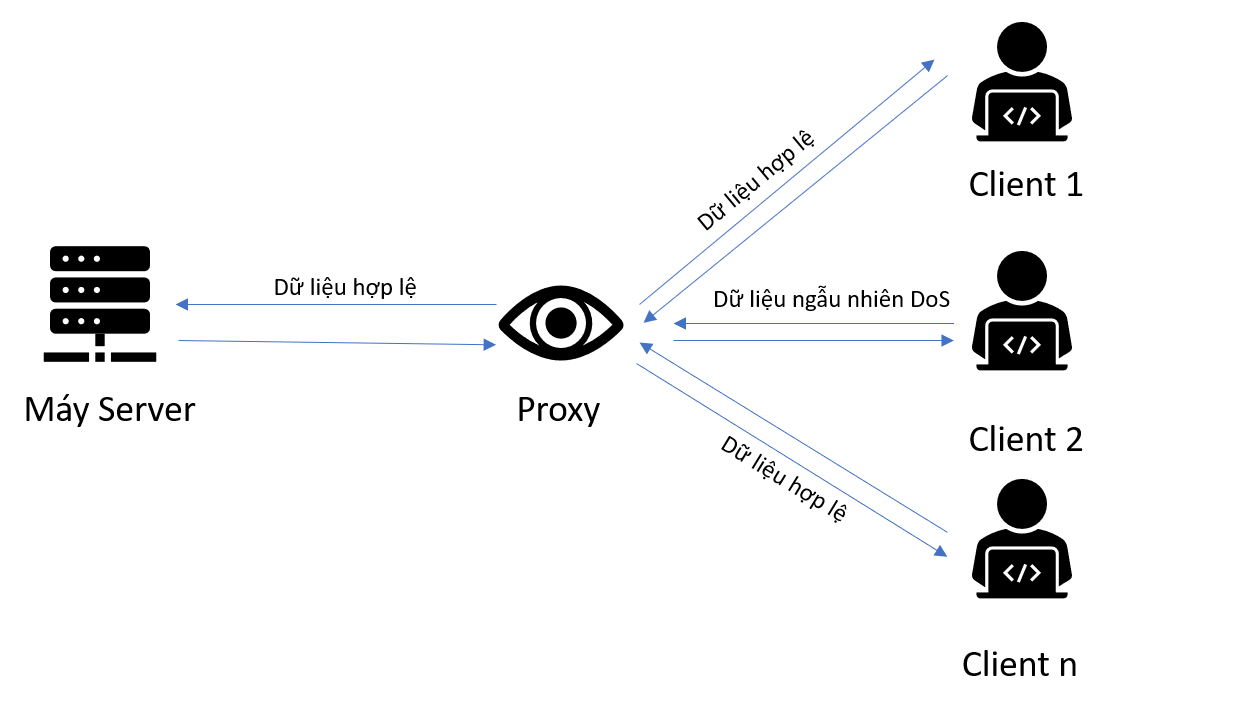
1. **Việc sử dụng Entropy liên quan gì đến việc phòng chống DoS UDP Flood?**

Như đã phân tích gói tin trong phần trước, chúng ta nhận thấy rằng việc một máy chủ dẫn đến bị quá tải trong quá trình tấn công DoS bằng UDP Flood là do phải xử lý quá nhiều gói tin có dữ liệu ngẫu nhiên cùng lúc. Bằng cách tính toán Entropy của các gói tin này, chúng ta có thể phân biệt đâu là gói tin bình thường, đâu là gói tin gây nhiễu loạn hệ thống.

1. **Thiết kế hệ thống phát hiện DoS bằng Entropy như thế nào?**

Trong thực tế, việc mở rộng chương trình service (ở đây là chương trình chứa thử thách CTF) để có khả năng tính toán Entropy sẽ khá phức tạp, và không đạt hiệu quả phòng chống DoS. Vì thế nên, chúng ta sẽ phát triển một chương trình Proxy đóng vai trò là một tường lửa giúp phát hiện cuộc tấn công DoS UDP Flood bằng cách nằm ở giữa client và server, thực hiện tính toán Entropy cho các gói tin được gửi từ client lên server.

Mô hình thiết kế :



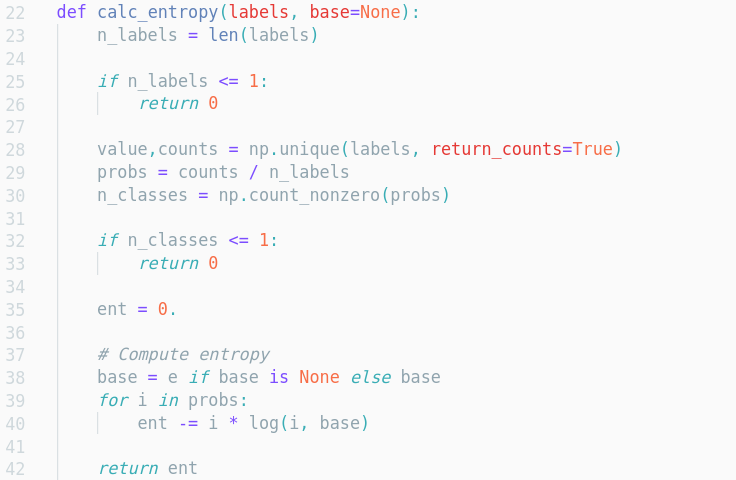
1. **Thiết kế thuật toán tính toán Entropy cho gói tin như thế nào?**

Entropy của một gói tin ngẫu nhiên với k ở trạng thái rời rạc K bằng công thức như sau:

**H(X) = -sum(each k in K p(k) \* log(p(k)))**

Entropy thấp nhất được tính cho một biến ngẫu nhiên có một sự kiện duy nhất với xác suất là 1,0. Entropy lớn nhất cho một biến ngẫu nhiên sẽ là nếu tất cả các sự kiện đều có khả năng xảy ra như nhau.

Sử dụng thư viện **numpy** và **scipy** của python, mã nguồn hàm tính entropy của gói tin như sau:



1. **Dữ liệu thu thập được từ một cuộc tấn công DoS UDP Flood**

Sau khi khởi động chương trình thử thách CTF đồng thời cùng với chương trình proxy để thực hiện sniff gói tin. Ta tiến hành tấn công DoS UDP Flood, kết quả entropy của một vài gói tin thu được kết quả như sau:

|  |  |
| --- | --- |
| STT | Entropy |
| 1 | 5.374658652987844 |
| 2 | 5.4250226975966775 |
| 3 | 5.396595316074639 |
| 4 | 5.4114481440191735 |
| 5 | 5.4114481440191735 |
| … | … |

Dựa vào bảng trên, chúng ta nhận thấy rằng entropy của một gói tin gồm những byte dữ liệu ngẫu nhiên sẽ cho ra entropy cao hơn nhiều so với những gói tin thông thường. Từ đây, chúng ta có thể chọn hạn mức entropy cho một gói tin hợp lệ là **5** và phát triển tiếp được thuật toán cảnh báo phát hiện tấn công DoS bằng phương thức UDP Flood.

1. **Thuật toán cảnh báo phát hiện tấn công DoS bằng UDP Flood.**

Như đã trình bày ở phần trên, với hạn mức entropy cho gói tin tối đa là 5, ta sẽ cho chương trình proxy sniff gói tin và tính toán entropy. Nếu một địa chỉ IP cố tình gửi nhiều gói tin chứa nhiều byte ngẫu nhiên quá một số lần nhất định, chương trình proxy sẽ gửi cảnh báo tới người dùng, và thực hiện chặn địa chỉ IP này trong một khoảng thời gian ngắn, hoặc có thể chặn vĩnh viễn, từ chối mọi gói tin tới từ địa chỉ IP đó.

Mã nguồn phát hiện tấn công UDP Flood:



1. **Ưu nhược điểm của cách phòng thủ bằng Entropy Proxy:**

***\* Ưu điểm:***

- Gọn nhẹ, mã nguồn mở, có thể cài đặt ở bất kỳ nơi đâu.

- Phát hiện nhanh chóng, có thể phát triển tối ưu riêng dành cho kiểu tấn công DoS UDP Flood.

***\* Nhược điểm:***

- Vẫn tồn tại khả năng bị qua mặt. Đặt trường hợp attacker không sử dụng những gói tin có các bytes ngẫu nhiên, thay vào đó sử dụng một số lượng lớn dữ liệu có một byte duy nhất (vd: **‘a’ \* 1024**). Lúc này, entropy của gói tin sẽ không cao nhưng vẫn đạt được mục đích tấn công DoS bằng UDP Flood, vì vẫn có rất nhiều dữ liệu mà máy chủ phải xử lý.

1. **Phòng chống tấn công DoS UDP Flood bằng Snort và tường lửa Pfsense:**
2. **Tường lửa Pfsense:**