BỘ GIÁO DỤC VÀ ĐÀO TẠO

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

KHOA CÔNG NGHỆ THÔNG TIN
BỘ MÔN: BLOCKCHAIN VÀ ỨNG DỤNG

-----📖-----

BÁO CÁO LAB 2

- **Nhóm thực hiện: Nhóm 9**

Thành phố Hồ Chí Minh, 25 tháng 12 năm 2022

I. Danh sách thành viên:

MSSV	Họ và Tên
19120478	Nguyễn Quang Định
19120562	Lê Thành Lộc
19120534	Phạm Đức Huy
19120521	Lê Nhật Khánh Hưng
19120625	Nguyễn Hữu Phương

II. Bài làm:

1. Theory & Business:

Exercise 1: What is the advantage of the Blockchain being a decentralized, Peerto-Peer system instead of a centralized one?

Một trong những lợi ích của một hệ thống phi tập trung, ngang hàng blockchain là nó có khả năng chống tội phạm và giả mạo cao hơn. Bơi vì blockchain được phân phán trên một mạng lưới các node, không có điểm kiểm soát hay sự cố duy nhất. Điều này có nghĩa là blockchain không thể dễ dàng bị tắt hoặc bị điều khiển bởi một đối tượng duy nhất, giống như một chính phủ hoặc tổ chức.

Một lợi ích khác của hệ thống phi tập trung là nó cho phép độ minh bạch và trách nhiệm cao hơn. Tất cả các giao dịch trên blockchain đều được ghi lại và có thể nhìn thấy bởi tất cả mọi người, làm cho khó khăn cho bất kì một bên nào muốn ẩn hoặc thay đổi thông tin.

Phi tập trung cũng cho phép sự bình đẳng giữa các người tham gia tăng lên. Trong một hệ thống tập trung, quyền lực và kiểm soát thường tập trung vào tay một vài người, nhưng trong một hệ thống phi tập trung, tất cả những người tham gia đều có một tiếng nói bình đẳng và có thể đóng góp vào mạng lưới.

Tổng thể, phi tập trung có thể cung cấp sự an toàn, minh bạch và công bằng hơn trong một hệ thống blockchain

Exercise 2: Consider a private blockchain running three nodes. Given that it is managed from a single organization, how do we assure decentralization?

Để đảm bảo phi tập trung trong một private blockchain chạy trên 3 node quản lý bởi một tổ chức duy nhất, điều quan trọng là đảm bảo rằng không có điểm duy nhất kiểm soát hoặc sự cố trong hệ hệ thống. Điều này có thể đạt được thông qua các biện pháp sau:

- Chắc chắn rằng các node chạy trên những cá nhân hoặc nhóm khác nhau trong cùng một tổ chức, thay vì bị kiểm soát bởi một cá nhân hay một tổ chức duy nhất.
- Sử dụng một thuật toán đồng thuật có thể không phụ thuộc vào một node hoặc một nhóm các node để thực hiện các quyết định, giống như một thuật toán proof-of-work hay proof-of-stake

- Thường xuyên đánh giá và kiểm tra hệ thống để chắc chắn rằng nó hoạt động theo một cách phi tập trung và thực hiện bất cứ thay đổi cần thiết để duy trình phi tập trung.
- Xem xét việc thêm các node vào mạng để phi tập trung hệ thống thêm nữa

Cũng rất quan trọng để chắc chắn rằng một private blockchain là minh bạch và trách nhiệm, để tất cả các bên có thể nhìn thấy và xác nhận các giao dịch và quyết định được được thực hiện trên mạng lưới. Điều này có thể giúp đảm bảo hệ thống phi tập trung và ngăn chặn bất cứ bên nào có quá nhiều sự kiểm soát trên hệ thống.

Exercise 3: What stops a malicious entity from being able to alter blocks in a permissionless blockchain?

Có một vài yếu tố giúp ngăn chặn một đối tượng độc hại từ thay đổi block trong một blockchain không cần sự cho phép:

- *Hàm băm mã hóa*: Mỗi block trong một blockchain chứa một hàm băm mã hóa của block trước đó, nó hoạt động như một dấu vân tay của khối block trước đó và đảm bảo sự toàn vẹn của nó. Nếu bất kì dữ liệu trong một block thay đổi, hàm băm của block cũng sẽ thay đổi, làm cho ngay lập tức rõ ràng rằng block đó đã bị giả mạo.
- *Proof-of-work*: Trong một proof-of-work blockchain, mỗi block sẽ bị khai thác bằng cách giải một câu đố khó. Điều này yêu cầu một sức mạnh tính toán và nguồn lực lớn, điều này giúp ngăn chặn một đối tượng duy nhất có khả năng thay đổi nhiều block trên cùng một hàng.
- *Network consensus*: Để cho một block được chấp nhận vào mạng lưới, nó phải được xác thực bởi một lượng lớn các node trên mạng lưới. Điều này giúp đảm bảo rằng một đối tượng giả mạo không thể thay đổi một block mà không đồng thuận với các node còn lại trên mạng lưới.
- *Khả năng thay đổi mạng lưới*: Một khi một block đã được thêm vào mạng lưới, nó rất khó để thay đổi hoặc xóa đi. Điều này làm cho một đối tượng giả mạo khó khăn trong việc thay đổi lịch sử của blockchain.

Tổng thể, các yếu tố đó sẽ làm cho một blockchain không có sự cho phép chống lại được các đối tượng giả mạng và đảm bảo được sự toàn vẹn của dữ liệu mà nó chứa đựng.

Exercise 4: What about permissioned blockchains? Can a malicious entity change in these as well?

Các blockchain được phép là các mạng riêng tư, tập trung, nơi chỉ một số bên nhất định được cấp quyền tham gia và truy cập mạng. Trong một blockchain được phép, mạng thường được kiểm soát bởi một tổ chức hoặc tập đoàn duy nhất, tổ chức này có khả năng cấp và thu hồi quyền truy cập mạng.

Mặc dù blockchain được phép có thể có một số biện pháp bảo mật giống như blockchain không được phép, chẳng hạn như hàm băm mật mã và sự đồng thuận của mạng, nhưng việc tập trung hóa mạng khiến nó dễ bị giả mạo và kiểm duyệt hơn. Bởi vì mạng được kiểm soát bởi một tổ

chức hoặc tập đoàn duy nhất, nên một thực thể độc hại trong tổ chức đó có thể thay đổi các block hoặc kiểm duyệt các giao dịch trên mạng.

Để ngăn chặn giả mạo và đảm bảo tính toàn vẹn của dữ liệu trên blockchain được phép, điều quan trọng là phải có các biện pháp bảo mật mạnh mẽ, chẳng hạn như kiểm soát truy cập mạnh mẽ và kiểm toán thường xuyên, đồng thời thường xuyên xem xét và cập nhật các giao thức bảo mật của mạng. Điều quan trọng đối với tổ chức hoặc tập đoàn kiểm soát mạng là phải minh bạch và chịu trách nhiệm trong các quy trình ra quyết định của mình.

Exercise 5: How does a client insert a transaction in the system?

Trong hệ thống blockchain, khách hàng có thể chèn một giao dịch bằng cách tạo một giao dịch mới và phát nó lên mạng.

Dưới đây là phác thảo chung về các bước liên quan đến việc chèn một giao dịch:

- Khách hàng tạo một giao dịch mới bằng cách chỉ định các chi tiết của giao dịch, chẳng hạn như người gửi, người nhận và số tiền được chuyển.
- Khách hàng ký giao dịch bằng khóa riêng của họ để chứng minh danh tính của họ và ủy quyền cho giao dịch.
- Máy khách phát giao dịch lên mạng. Điều này có thể liên quan đến việc gửi giao dịch tới một node cụ thể hoặc tới nhiều node trong mạng.
- Các node trên mạng nhận giao dịch và xác thực giao dịch đó bằng cách kiểm tra xem nó đã được ký hợp lệ chưa và người gửi có đủ tiền để hoàn tất giao dịch hay không.
- Nếu giao dịch hợp lệ, các node sẽ thêm nó vào bản sao blockchain cục bộ của chúng và cũng có thể chuyển tiếp giao dịch tới các nút khác trong mạng.
- Khi giao dịch đã được thêm vào blockchain, nó được coi là một phần của bản ghi vĩnh viễn của blockchain và không thể bị thay đổi hoặc xóa.

Nhìn chung, quá trình chèn một giao dịch liên quan đến việc tạo giao dịch, phát nó lên mạng và để nó được xác thực và thêm vào blockchain bởi các node trên mạng.

Exercise 6: How do we know if a transaction was created by a client and not by a malicious node?

Có một số cách để đảm bảo rằng một giao dịch được tạo bởi một khách hàng hợp pháp chứ không phải bởi một node độc hại:

- *Chữ ký mật mã*: Các giao dịch thường được ký bằng khóa riêng của khách hàng để chứng minh danh tính của họ và ủy quyền cho giao dịch. Các node trên mạng có thể xác minh chữ ký bằng khóa công khai của khách hàng, điều này đảm bảo rằng chỉ khách hàng có private key tương ứng mới có thể tạo giao dịch.
- *Sự đồng thuận của mạng*: Để một giao dịch được mạng chấp nhận, nó phải được xác thực bởi phần lớn các node trên mạng. Điều này giúp đảm bảo rằng giao dịch là hợp pháp và không được tạo bởi một node độc hại.

- *Phí giao dịch*: Trong một số hệ thống blockchain, khách hàng phải trả phí giao dịch để mạng xử lý giao dịch của họ. Điều này giúp ngăn cản các node độc hại tạo giao dịch giả mạo, vì chúng sẽ phải trả phí để mạng lưới chấp nhận giao dịch.
- *Hệ thống danh tiếng*: Một số hệ thống blockchain sử dụng hệ thống danh tiếng để theo dõi lịch sử của các node trên mạng và xác định những node có lịch sử tạo giao dịch không hợp lệ hoặc độc hại. Điều này có thể giúp ngăn chặn các nút độc hại có thể tạo giao dịch giả mạo.

Nhìn chung, các biện pháp này giúp đảm bảo rằng các giao dịch trên blockchain được tạo bởi các khách hàng hợp pháp chứ không phải bởi các node độc hại.

Exercise 7: Why can't permissionless blockchains use the classical Consensus algorithms and must instead rely on other forms of Consensus like Proof-OfWork?

Các **blockchain không được phép**, còn được gọi là **public blockchain**, là các mạng phi tập trung mở cho bất kỳ ai tham gia và tham gia. Các chuỗi khối này sử dụng các thuật toán đồng thuận để đạt được thỏa thuận về trạng thái của chuỗi khối và đảm bảo tính toàn vẹn của dữ liệu chứa trong đó.

Các **thuật toán đồng thuận cổ điển**, chẳng hạn như Paxos và Raft, được thiết kế để sử dụng trong các hệ thống tập trung nơi một thực thể duy nhất chịu trách nhiệm điều phối hành động của những người tham gia khác. Các thuật toán này có thể không phù hợp để sử dụng trong các chuỗi khối không được phép, bởi vì chúng dựa trên một điểm kiểm soát hoặc lỗi duy nhất và không mở rộng tốt cho các mạng lớn, phi tập trung.

Thay vào đó, các blockchain không được phép thường sử dụng các **thuật toán đồng thuận thay thế**, chẳng hạn như *proof-of-work* (PoW) hoặc *proof-of-stake* (PoS), được thiết kế đặc biệt để sử dụng trong các mạng phi tập trung. Các thuật toán này cho phép ra quyết định phi tập trung và giúp đảm bảo tính toàn vẹn và bảo mật của chuỗi khối.

Nhìn chung, các blockchain không được phép dựa vào các thuật toán đồng thuận thay thế vì các thuật toán này phù hợp hơn để sử dụng trong các mạng mở, phi tập trung và cung cấp các đảm bảo về tính toàn vẹn và bảo mật mạnh mẽ hơn.

Exercise 8: Why can't a node fake a result from Proof-of-Work?

Rất khó để một node giả mạo kết quả từ Proof-of-Work (PoW) do cách thuật toán PoW được thiết kế. Trong chuỗi khối PoW, mỗi khối phải được "khai thác" bằng cách giải một câu đố khó tính toán. Điều này yêu cầu node thực hiện một số lượng lớn các phép tính và yêu cầu một lượng tài nguyên và sức mạnh tính toán đáng kể.

Bởi vì việc giải câu đố đòi hỏi rất nhiều nỗ lực, nên không chắc rằng một node có thể giả mạo kết quả từ PoW. Ngoài ra, giải pháp cho câu đố rất dễ xác minh, vì vậy nếu một nút cố gắng làm giả kết quả, phần còn lại của mạng sẽ ngay lập tức nhận ra điều đó.

Nhìn chung, nỗ lực tính toán cần thiết để giải câu đố và việc dễ dàng xác minh khiến nút khó giả mạo kết quả từ PoW. Điều này giúp đảm bảo tính toàn vẹn và bảo mật của chuỗi khối.

Exercise 9: Why do we want to add difficulty to the Proof-of-Work?

Trong blockchain Proof-of-Work (PoW), độ khó của câu đố phải được giải để "khai thác" một khối mới là một yếu tố quan trọng giúp đảm bảo tính bảo mật và tính toàn vẹn của chuỗi khối.

Bằng cách thêm độ khó cho PoW, việc một thực thể đơn lẻ kiểm soát chuỗi khối và thao tác dữ liệu chứa trong đó trở nên khó khăn hơn. Điều này là do độ khó tăng lên đòi hỏi nhiều tài nguyên và sức mạnh tính toán hơn để giải câu đố, điều này khiến việc một thực thể đơn lẻ khai thác các khối mới trở nên tốn kém và tốn thời gian hơn.

Ngoài ra, độ khó của PoW giúp điều chỉnh tốc độ các khối mới được thêm vào chuỗi khối. Bằng cách tăng độ khó, tốc độ tạo khối bị chậm lại, giúp ngăn chặn sự tắc nghẽn của chuỗi khối và đảm bảo rằng mạng có thể xử lý các giao dịch một cách hiệu quả.

Nhìn chung, việc thêm độ khó cho PoW giúp đảm bảo tính bảo mật và tính toàn vẹn của chuỗi khối, đồng thời điều chỉnh tốc độ tạo khối.

Exercise 10: How does the difficulty value work in Proof-of-Work?

- Trong *chuỗi khối bằng chứng công việc* (**PoW Blockchain** - viết tắt của **proof-of-work blockchain**), *giá trị độ khó* (**difficult value)** xác định mức độ khó để giải câu đố phải được hoàn thành để "khai thác" một *block* mới. *Giá trị độ khó* thường được biểu thị dưới dạng một số đại diện cho số lượng các số 0 đứng đầu phải được đưa vào hàm băm của *block* để *block* được coi là hợp lệ.
- *Ví dụ:* nếu giá trị độ khó được đặt thành 4, điều này có nghĩa là hàm băm của *block* phải có ít nhất 4 số 0 đứng đầu thì *block* đó mới được coi là hợp lệ.
- Điều này đòi hỏi rất nhiều nỗ lực tính toán để đạt được, vì hàm băm của *block* phải được tính toán và điều chỉnh nhiều lần cho đến khi đáp ứng được độ khó yêu cầu. *Giá trị độ khó* được điều chỉnh định kỳ để đảm bảo rằng tốc độ tạo *block* vẫn nhất quán, bất kể thay đổi về số lượng người khai thác hoặc lượng điện toán được sử dụng trên mạng. **Nếu** tốc độ tạo *block* trở nên quá nhanh, giá trị độ khó sẽ tăng lên để khiến việc khai thác *block* mới trở nên khó khăn hơn. **Nếu** tốc độ tạo *block* trở nên quá chậm, giá trị độ khó sẽ giảm xuống để giúp khai thác *block* mới dễ dàng hơn. Nhìn chung, giá trị độ khó trong chuỗi khối PoW xác định mức độ khó để giải câu đố phải hoàn thành để khai thác một khối mới và giúp điều chỉnh tốc độ tạo khối trên mạng.

Exercise 11: How does a node share its new block?

- Trong một mạng *blockchain,* các *node* là máy tính hoặc thiết bị duy trì một bản sao của *blockchain* và giao tiếp với các *node* khác trong mạng. Khi một *node* tạo một *block* giao dịch mới, nó thường phát *block* đó tới phần còn lại của mạng để các *node* khác có thể xác thực và thêm nó vào bản sao *blockchain* của chúng.

- Có một số cách mà một *node* có thể chia sẻ *block* mới của nó với phần còn lại của mạng, tùy thuộc vào giao thức và kiến trúc cụ thể của *blockchain*. Đây là vài ví dụ:

- **Flooding**: Theo cách tiếp cận này, *node* sẽ phát *block* mới tới tất cả các *node* khác trong mạng. Mỗi *node* nhận được *block* sẽ xác minh nó và nếu nó hợp lệ, nó sẽ chuyển nó cho tất cả các *node* lân cận của nó. Quá trình này tiếp tục cho đến khi tất cả các *node* trong mạng đã nhận được *block*.
- **Gossip protocol**: Theo cách tiếp cận này, *node* sẽ gửi *block* mới đến một số lượng nhỏ các *node* được chọn ngẫu nhiên. Các *node* này sau đó chuyển tiếp khối tới một vài *node* khác, v.v..., cho đến khi *block* được phân phối khắp mạng.
- **Direct communication**: Theo cách tiếp cận này, *node* gửi *block* mới trực tiếp đến các *node* cụ thể trong mạng, chẳng hạn như các *node* lân cận hoặc các *node* mà nó có kết nối trực tiếp.

- Có nhiều cách tiếp cận khác để chia sẻ các *block* mới trong mạng *blockchain* và cách tiếp cận cụ thể được sử dụng có thể khác nhau tùy thuộc vào nhu cầu và mục tiêu của mạng.

Exercise 12: What are the advantages and disadvantages of sharing the chain by broadcasting the block or by verifying other chains from peer nodes?

- Có một số ưu điểm và nhược điểm khi sử dụng các phương pháp khác nhau để chia sẻ *block* và xác minh chuỗi trong mạng *blockchain*. Dưới đây là một số điểm cần xem xét:

- Khối phát sóng:

- *Ưu điểm:*
  - **Nhanh chóng và hiệu quả**: Việc phát một *block* mới cho toàn bộ mạng cho phép tất cả các *node* nhận *block* cùng một lúc, có thể nhanh hơn các phương pháp khác.
  - **Linh hoạt với các phân vùng mạng**: Nếu mạng được phân vùng, các *node* ở một bên của phân vùng vẫn có thể nhận các *block* mới từ phía bên kia bằng cách phát chúng.
- *Nhược điểm:*
  - **Sử dụng nhiều tài nguyên**: Các *block* phát sóng yêu cầu các *node* gửi và nhận một số lượng lớn tin nhắn, điều này có thể tiêu tốn nhiều băng thông và các tài nguyên khác.
  - **Dễ bị spam**: Các tác nhân độc hại có thể cố gắng làm tràn ngập mạng bằng các *block* giả mạo hoặc thư rác khác, điều này có thể làm quá tải hệ thống và gây khó khăn cho việc truyền bá các *block* hợp pháp.

- Xác minh chuỗi từ các nút ngang hàng:

- *Ưu điểm:*
  - **Hiệu quả hơn:** Việc xác minh chuỗi từ các *node* ngang hàng cho phép các *node* chỉ yêu cầu dữ liệu cụ thể mà chúng cần, thay vì nhận và xử lý một số lượng lớn các *block* không cần thiết.

    - **Giảm thư rác:** Bằng cách yêu cầu dữ liệu từ các đồng nghiệp cụ thể thay vì chấp nhận tất cả các khối đến, các nút có thể giảm nguy cơ chấp nhận các khối giả mạo hoặc thư rác khác.
- Nhược điểm:
    - **Chậm hơn:** Việc xác minh chuỗi từ các *node* ngang hàng yêu cầu các *node* yêu cầu và đợi dữ liệu từ các *node* khác, điều này có thể chậm hơn so với nhận dữ liệu trực tiếp.
    - **Ít linh hoạt hơn đối với các phân vùng mạng:** Nếu mạng bị phân vùng, các *node* ở một bên của phân vùng có thể không yêu cầu hoặc nhận dữ liệu từ phía bên kia, điều này có thể gây ra sự chậm trễ hoặc các sự cố khác.

- **Kết luận:** cuối cùng, phương pháp tốt nhất để chia sẻ các *block* và xác minh chuỗi sẽ phụ thuộc vào các yêu cầu và mục tiêu cụ thể của mạng *blockchain.*

Exercise 13: What happens when multiple nodes mine a new block at the same time?

Điều gì xảy ra khi nhiều node khai thác một block mới cùng một lúc

- Trong một blockchain network, multiple node có thể khai thác một khối mới cùng một lúc. điều này xảy ra khi hai hoặc nhiều nút độc lập giải câu đố cần thiết để tạo một block mới và thêm nó vào chuỗi.
- Khi điều này xảy ra, mạng phải xác định khối nào sẽ được chấp nhận làm khối hợp lệ tiếp theo trong chuỗi. Quá trình này gọi là Fork. Có hai loại fork chính có thể xảy ra trong blockchain: soft fork và hard fork.
    - Soft là một một thay đổi đối với giao thức chuỗi khối tương thích ngược, nghĩa là các Node chạy phiên bản cũ của giao thức có thể xác thực và xử lý các khối được tạo bởi các nút chạy phiên bản mới. Trong một soft fork mạng thường sẽ đi theo chuỗi dài nhất, bất kể nó được tạo bằng phiên bản nào của giao thức.
    - Mặt khác, một hard fork là một thay đổi đối với giao thức chuỗi khối không tương thích ngược. Điều này có nghĩa là các nút chạy phiên bản cũ của giao thức sẽ không thể xác thực hoặc xử lý các khối được tạo bởi các nút chạy phiên bản mới. Trong một hard fork, mạng thường sẽ chia thành hai chuỗi riêng biệt, với một số nút theo chuỗi cũ và các nút khác theo chuỗi mới.

    - Trong cả hai trường hợp, mạng cuối cùng sẽ giải quyết phân nhánh bằng cách chọn một chuỗi làm chuỗi hợp lệ và loại bỏ chuỗi kia. Quá trình này có thể mất một chút thời gian và có thể dẫn đến một số gián đoạn hoặc nhầm lẫn tạm thời trong mạng.

Exercise 14: Why do the miners choose the longest chain to work and not a forked, smaller chain?

Tại sao những người khai thác chọn chuỗi dài nhất để làm việc mà không phải là một chuỗi nhỏ hơn, rẽ nhánh?

Trong mạng chuỗi khối, những người khai thác thường chọn làm việc trên chuỗi dài nhất vì đó là chuỗi có nhiều khả năng được phần còn lại của mạng chấp nhận là chuỗi hợp lệ.

Có một số lý do tại sao chuỗi dài nhất thường được coi là hợp lệ nhất:

- *Bảo mật*: Chuỗi càng dài thì càng có nhiều công việc được thực hiện để bảo mật. Điều này có nghĩa là kẻ tấn công sẽ khó thay đổi hoặc đảo ngược các giao dịch trên một chuỗi dài hơn.
- *Sự đồng thuận*: Bằng cách làm việc trên chuỗi dài nhất, những người khai thác báo hiệu cho phần còn lại của mạng rằng họ đồng ý với trạng thái hiện tại của chuỗi khối. Điều này giúp duy trì sự đồng thuận trong mạng và đảm bảo rằng tất cả các nút đang hoạt động hướng tới cùng một mục tiêu.
- *Hiệu quả*: Trong hầu hết các trường hợp, các công cụ khai thác sẽ hiệu quả hơn khi làm việc trên chuỗi dài nhất vì cần ít nỗ lực hơn để xây dựng trên chuỗi hiện có so với bắt đầu một chuỗi mới từ đầu.

Mặc dù chuỗi nhỏ hơn cuối cùng có thể bắt kịp và vượt qua chuỗi dài nhất, nhưng điều này thường khó xảy ra trừ khi chuỗi nhỏ hơn có một số lợi thế đáng kể, chẳng hạn như phần thưởng khối cao hơn hoặc độ khó thấp hơn. Trong hầu hết cái nó mang lại sự bảo mật cao nhất và tiềm năng nhận phần thưởng lớn nhất.c trường hợp, những người khai thác sẽ tiếp tục làm việc trên chuỗi dài nhất.

Exercise 15: What happens to the other block and transactions that were created in the forked chain that was discontinued?

Khi một nhánh rẽ được giải quyết trong mạng chuỗi khối, một chuỗi thường được chọn làm chuỗi hợp lệ và chuỗi còn lại bị loại bỏ. Điều này có nghĩa là bất kỳ khối và giao dịch nào được tạo trên chuỗi bị loại bỏ sẽ không được đưa vào chuỗi hợp lệ và sẽ không được coi là hợp lệ bởi mạng.

Tuy nhiên, điều này không nhất thiết có nghĩa là các giao dịch trên chuỗi bị loại bỏ sẽ bị mất vĩnh viễn. Trong một số trường hợp, có thể "phát lại" các giao dịch trên chuỗi hợp lệ, theo cách thủ công hoặc sử dụng phần mềm chuyên dụng. Điều này có thể cho phép người dùng khôi phục giá trị hoặc tài sản được liên kết với các giao dịch của họ, ngay cả khi bản thân các giao dịch đó không được đưa vào chuỗi hợp lệ.

Tuy nhiên, điều quan trọng cần lưu ý là việc phát lại các giao dịch không phải lúc nào cũng khả thi và có thể không phải lúc nào cũng là cách hành động tốt nhất. Ví dụ: nếu chuỗi bị loại bỏ do kẻ tấn công tạo ra đang cố gắng chi tiêu gấp đôi hoặc lừa đảo người dùng, thì việc phát lại các giao dịch trên chuỗi hợp lệ có thể không an toàn hoặc không được khuyến khích.

Nói chung, người dùng nên xem xét cẩn thận các rủi ro và hậu quả tiềm ẩn của các nhánh và thực hiện các biện pháp phòng ngừa thích hợp để bảo vệ tài sản và giao dịch của họ. Điều này có thể bao gồm theo dõi các giao dịch của họ và theo dõi trạng thái của mạng chuỗi khối để đảm bảo rằng các giao dịch của họ được đưa vào chuỗi hợp lệ.

Exercise 16: What if the service was already made and the payment (in the form of a transaction) disappears?

- Trong mạng *blockchain*, các giao dịch thường được coi là không thể đảo ngược sau khi chúng được thêm vào chuỗi. Điều này có nghĩa là nếu một giao dịch biến mất khỏi chuỗi, thì có thể khó hoặc không thể khôi phục giao dịch đó hoặc tài sản hoặc giá trị liên quan đến giao dịch đó.

- Nếu một dịch vụ đã được cung cấp và khoản thanh toán cho dịch vụ đó được thực hiện dưới hình thức giao dịch mà sau đó biến mất khỏi *block*, các bên liên quan có thể cần tìm một cách khác để giải quyết vấn đề. Điều này có thể liên quan đến việc đạt được thỏa thuận mới hoặc sử dụng phương thức thanh toán khác để giải quyết khoản nợ.

- Điều quan trọng đối với người dùng mạng *blockchain* là phải xem xét cẩn thận các rủi ro và hậu quả tiềm ẩn của việc sử dụng công nghệ *blockchain* cho các giao dịch và thực hiện các biện pháp phòng ngừa thích hợp để bảo vệ tài sản và giao dịch của họ. Điều này có thể bao gồm xem xét cẩn thận các điều khoản dịch vụ cho nền tảng hoặc dịch vụ đang được sử dụng, theo dõi các giao dịch và theo dõi trạng thái của mạng *blockchain* để đảm bảo rằng các giao dịch được ghi lại chính xác.

Exercise 17: Being a Peer-to-Peer service, how does a node discover which nodes to connect to in order to join the system?

Trong dịch vụ ngang hàng (P2P), các nút khám phá các nút khác để kết nối nhằm tham gia hệ thống thông qua một quy trình gọi là khởi động. Quá trình này liên quan đến việc kết nối với một nút đã biết, thường được gọi là "node gốc" hoặc "node khám phá", duy trì danh sách các nút khác trong mạng. Sau đó, nút mới có thể sử dụng danh sách này để kết nối với các nút khác và tham gia mạng. Trong một số trường hợp, các nút cũng có thể sử dụng bảng băm phân tán (DHT) hoặc các cơ chế khác để khám phá và kết nối với các nút khác trong mạng. Các nút cũng có thể khám phá các nút khác thông qua việc sử dụng các thông báo quảng bá hoặc bằng cách lắng nghe các kết nối đến.

2. Blockchain4Students:

Exercise 1: Consider the next code fragment. What type is the Blockchain4Students blockchain?

Kiểu blockchain của Blockchain4Students là blockchain liên kết là một mạng được quản lý chung bởi một nhóm các tổ chức và có thể là công khai hoặc riêng tư. Blockchain công khai là một mạng phi tập trung mở cho bất kỳ ai và các giao dịch được ghi lại trên một sổ cái có sẵn công khai. Mặt khác, một blockchain riêng tư là một mạng chỉ có thể truy cập được đối với một nhóm hoặc tổ chức cụ thể và thường được sử dụng cho các quy trình nội bộ.

Exercise 2: We give rewards right after a block is mined. What is the problem with this approach?

Có một vài vấn đề tiềm ẩn với việc thưởng cho người khai thác ngay sau khi một khối được khai thác:

- *Chi tiêu gấp đôi*: Nếu những người khai thác có thể nhận được phần thưởng trước khi block được thêm vào chain, thì họ có khả năng tạo ra nhiều giao dịch với cùng số tiền, được gọi là chi tiêu gấp đôi. Điều này có thể dẫn đến hoạt động gian lận và làm suy yếu tính toàn vẹn của blockchain.
- *Kích thước blockchain*: Nếu phần thưởng được phân phối ngay sau khi một block được khai thác, điều đó có thể dẫn đến sự gia tăng kích thước của blockchain khi nhiều phần thưởng được thêm vào chain. Điều này có thể làm cho blockchain khó lưu trữ và bảo trì hơn, đồng thời có thể dẫn đến các vấn đề về khả năng mở rộng.
- *Rủi ro bảo mật*: Nếu phần thưởng được phân phối ngay sau khi một block được khai thác, nó có thể khuyến khích những người khai thác ưu tiên tốc độ hơn bảo mật. Điều này có thể dẫn đến các lỗ hổng bảo mật trong blockchain khi các công ty khai thác vội vàng hoàn thành các block càng nhanh càng tốt.

Nhìn chung, tốt hơn hết là phân phối phần thưởng cho những người khai thác sau khi một block đã được thêm vào chain, vì điều này giúp đảm bảo tính toàn vẹn và bảo mật của blockchain.

Exercise 3: Each Transaction includes the ID from the node that created it. Why is this information necessary?

ID node được bao gồm trong mỗi giao dịch đóng vai trò như một cách để xác định nguồn gốc của giao dịch và xác minh rằng nó được tạo bởi một nút hợp pháp trên mạng. Thông tin này là cần thiết vì nó giúp đảm bảo tính toàn vẹn và bảo mật của chuỗi khối.

Bằng cách bao gồm ID node trong mỗi giao dịch, có thể xác minh tính xác thực của giao dịch và đảm bảo rằng nó không được tạo bởi một node độc hại hoặc trái phép. Điều này đặc biệt quan trọng trong một mạng phi tập trung, nơi không có cơ quan trung ương nào xác minh tính xác thực của các giao dịch.

Ngoài ra, ID node có thể được sử dụng để xác minh chữ ký số của giao dịch, giúp đảm bảo rằng giao dịch không bị giả mạo hoặc sửa đổi theo bất kỳ cách nào. Bằng cách xác minh chữ ký và ID nút, có thể xác nhận rằng giao dịch hợp lệ và không bị thay đổi. Điều này giúp duy trì tính toàn vẹn và bảo mật của chuỗi khối.

Exercise 4: When a transaction request arrives at the node, it signs it with its own private key. What is the possible attack with this approach?

Một cuộc tấn công có thể được thực hiện bằng cách sử dụng quy trình ký và yêu cầu giao dịch được mô tả trong mã bạn cung cấp là cuộc tấn công "man-in-the-middle". Trong kiểu tấn công này, kẻ tấn công chặn một yêu cầu giao dịch và sửa đổi nó trước khi chuyển tiếp nó đến node để xử lý. Kẻ tấn công có thể thay đổi địa chỉ đích, số lượng giao dịch hoặc các chi tiết khác để thao túng giao dịch theo một cách nào đó.

Nếu node không cẩn thận xác minh tính xác thực của yêu cầu giao dịch, nó có thể ký vào phiên bản đã sửa đổi của yêu cầu và phát nó lên mạng như một giao dịch hợp pháp. Điều này có thể dẫn đến tổn thất tài chính hoặc hậu quả tiêu cực khác cho nạn nhân của cuộc tấn công.

Để ngăn chặn kiểu tấn công này, điều quan trọng là nút phải xác minh tính xác thực của yêu cầu giao dịch trước khi ký. Điều này có thể được thực hiện bằng cách sử dụng chữ ký điện tử để xác minh tính toàn vẹn của yêu cầu hoặc bằng cách kiểm tra yêu cầu đối với danh sách các giao dịch hợp pháp đã biết. Bằng cách thực hiện các biện pháp phòng ngừa này, có thể giúp ngăn chặn các cuộc tấn công trung gian và duy trì tính bảo mật và tính toàn vẹn của chuỗi khối.

Exercise 5: When a node first connects, it contacts a discovery node to find other peers in the Blockchain. Although this method is possible, why is it not realistic to use on broader scale blockchains?

Việc sử dụng discovery Node để tìm các node khác trong chuỗi khối có thể không thực tế trên quy mô rộng hơn vì một vài lý do. Đầu tiên, một discovery Node có thể trở thành một bottleneck cho mạng, vì nó sẽ cần xử lý các yêu cầu từ tất cả các nút đang cố gắng khám phá các đồng nghiệp khác. Điều này có thể dẫn đến sự chậm trễ và tăng độ trễ cho các nút đang cố gắng kết nối. Thứ hai, một discovery Node có khả năng là một điểm lỗi duy nhất đối với mạng, vì nếu node này bị hỏng hoặc không khả dụng, các nút trong mạng có thể không khám phá được các đồng nghiệp mới. Cuối cùng, việc dựa vào một discovery Node duy nhất cũng có thể khiến mạng dễ bị tấn công hơn, vì kẻ tấn công có thể nhắm mục tiêu vào discovery Node để phá vỡ mạng.

Exercise 6: When the node obtains a valid chain bigger than its own, it discards its own chain for the bigger one. Why?

Node loại bỏ chuỗi của chính nó để lấy chuỗi lớn hơn vì blockchain được thiết kế để trở thành một sổ cái phân tán được phân cấp và bảo mật bằng mật mã. Nó dựa trên khái niệm đồng thuận, có nghĩa là tất cả các nút trong mạng đều đồng ý về trạng thái của blockchain.

Trong một blockchain, các khối được thêm vào chuỗi thông qua một quy trình gọi là khai thác, trong đó những người khai thác cạnh tranh để giải một câu đố mật mã. Người khai thác đầu tiên giải được câu đố sẽ thêm khối tiếp theo vào chuỗi và được thưởng cho công việc của họ. Kết quả là, chuỗi khối không ngừng phát triển khi các khối mới được thêm vào.

Để một node có một chuỗi hợp lệ, nó phải chứa tất cả các khối đã được thêm vào chuỗi khối. Nếu chuỗi của nút ngắn hơn chuỗi mà nó nhận được từ một node ngang hàng, điều đó có nghĩa là chuỗi của node ngang hàng chứa nhiều khối hơn, nghĩa là nó cập nhật hơn và do đó có nhiều khả năng là chuỗi chính xác hơn. Trong trường hợp này, lợi ích tốt nhất của node là loại bỏ chuỗi của chính nó và sử dụng chuỗi ngang hàng để duy trì đồng bộ với phần còn lại của mạng và duy trì tính toàn vẹn của blockchain.

Exercise 7: With our current implementation, when a node joins, he is not aware of any state of its peers. Add the functionality of when a node joins the system to ask and update its own Blockchain.

Đây là một cách tiếp cận khả thi để thêm chức năng này vào mã:

1. Thêm một phương thức mới vào class Blockchain cho phép một nút yêu cầu và cập nhật blockchain của chính nó. Phương thức này có thể được gọi là *update_blockchain()* và nó có thể nhận một đối số duy nhất: ID của nút.
2. Bên trong phương thức *update_blockchain()*, hãy sử dụng thư viện yêu cầu để gửi yêu cầu GET tới nút khám phá, yêu cầu danh sách tất cả các nút khác trong blockchain.
3. Sau khi nhận được danh sách các node, hãy lặp lại danh sách và gửi yêu cầu GET tới từng node, yêu cầu bản sao blockchain mới nhất của chúng.
4. Khi một bản sao của blockchain được nhận từ một node, hãy so sánh độ dài của nó với độ dài của blockchain của node hiện tại. Nếu nó dài hơn, hãy loại bỏ blockchain của node hiện tại và thay thế nó bằng blockchain đã nhận.
5. Sau khi tất cả các node đã được liên hệ và blockchain đã được cập nhật, phương thức *update_blockchain()* có thể trả lại chuỗi khối đã cập nhật cho người gọi.

Đây là một ví dụ về lớp Blockchain được cập nhật trông như thế này:

```
class Blockchain:
    def __init__(self, blocks=[], node_id=None):
        self.chain = blocks
        self.node_id = node_id
        self.discovery_node = "http://localhost:5000"

    def update_blockchain(self):
        # Send a GET request to the discovery node to get a list of all the
nodes in the blockchain
        r = requests.get(f"{self.discovery_node}/nodes")
        nodes = r.json()

        # Iterate through the list of nodes and send a GET request to each
one, asking for their latest copy of the blockchain
        for node in nodes:
            r = requests.get(f"{node}/chain")
            received_chain = r.json()
            if len(received_chain) > len(self.chain):
                # Replace the current node's blockchain with the received
blockchain
                self.chain = received_chain
        # Return the updated blockchain
        return self.chain
```

Để sử dụng lớp Blockchain được cập nhật này, bạn có thể gọi phương thức *update_blockchain()* bất cứ khi nào một nút mới tham gia hệ thống