**BÀI TẬP TRÊN LỚP**

**MÔN HỌC: HỆ PHÂN TÁN**

**CHƯƠNG 3: ĐỊNH DANH TRONG HỆ PHÂN TÁN**

**HỌ TÊN SV:** Nguyễn Trọng Hùng  **MSSV:** 20194290

**MÃ LỚP:** 136836  **MÃ HỌC PHẦN:** IT4611

**Câu hỏi 1: Tại sao không thể lấy địa chỉ của Access Point để sử dụng như địa chỉ của thực thể?**

Vì thông tin có khả năng chuyển từ Access point này sang Access point khác

**Câu hỏi 2: Với việc sử dụng Định Danh, các vấn đề gì cần phải xem xét?**

Vấn đề cần xem xét là cần cải thiện không gian tên

Giải pháp: tái sử dụng định danh sau 1 khoảng thời gian nhất định nếu định danh đó không còn được sử dụng

**Câu hỏi 3: Xét một thực thể di chuyển từ vị trí A sang vị trí B. Trong quá trình di chuyển thực thể đó có đi qua các nút trung gian nhưng chỉ dừng lại ở đó khoảng thời gian ngắn. Khi đến B, thực thể đó dừng lại. Chúng ta biết rằng việc thay đổi địa chỉ trong một dịch vụ tổ chức vị trí phân cấp (hierarchical location service) là rất mất thời gian để hoàn thành, vì vậy cần tránh làm việc này khi thực thể tạm dừng ở các nút trung gian. Hãy đề xuất một mô hình kết hợp cả dịch vụ tổ chức vị trí phân cấp và cơ chế chuyển tiếp con trỏ (forwarding pointers) để có thể xác định được vị trí của thực thể khi nó ở các nút trung gian.**

Khi thông tin di chuyển từ vị trí A sang vị trí B, khi đến B, nó để lại con trỏ và ghi thêm địa chỉ mới đó -> mô hình phân cấp. Khi chuỗi con trỏ trung gian bị xóa đi thì địa chỉ ở A cũng bị xóa.

**Câu hỏi 4: Trình bày một số phương pháp ARP Spoofing để thấy được điểm yếu của phương pháp định danh sử dụng cơ chế quảng bá.**

Có hai phương pháp:

+ Phương pháp 1: Hacker gửi đi bản tin ARP reply với địa chỉ MAC là địa chỉ broadcast ở lớp hai FF:FF:FF:FF:FF:FF. Kết quả là thông tin máy nạn nhân khi gửi unicast đến một host nào đó cũng bị coi là gửi đến tất cả các host trong mạng.

+ Phương pháp 2: Hacke gửi ra toàn mạng bản tin ARP reply giả mạo địa chỉ MAC của một server nào đó trong mạng bằng một địa chỉ MAC không tồn tại. Kết quả là khi các host yêu cầu một dịch vụ nào đó trên server thì sẽ không được đáp ứng. Gói tin yêu cầu có gửi đi nhưng sẽ bị huỷ vì không có host nào nhận cả -> tấn công DoS.

**Câu hỏi 5: Vấn đề còn tồn tại đối với cơ chế chuyển tiếp con trỏ (Forwarding Pointer) là gì?**

Không có phạm vi cụ thể để thực hiện, tất cả các vị trí trung gian trong 1 chuỗi sẽ phải duy trì một phần của nó trong chuỗi Forwarding pointer, liên kết có thể bị phá vỡ

**Câu hỏi 6: Nhược điểm của giải pháp Home-based là gì? Giải pháp nào để giải quyết nhược điểm đó?**

Nhược điểm: có nhiều trường hợp thực thể ở rất gần client nhưng home-agent lại ở quá xa client -> tăng thời gian truyền tin

Giải pháp: đưa ra vị trí gốc, giúp theo dõi vị trí hiện tại của 1 thực thể, nếu thực thể ở quá xa Homeagent trong 1 thời gian đủ lâu, thì nó sẽ được gắn 1 Home-agent mới ở gần hơn.

**Câu hỏi 7: Khi áp dụng giải pháp sử dụng hàm băm phân tán vào hệ thống Chord thì nó đã tối ưu cơ chế định danh như thế nào?**

**Câu hỏi 8: Trong giải pháp phân cấp, sử dụng cơ chế bộ đệm có tác dụng cải thiện hiệu năng như thế nào? Cho ví dụ.**

Xây dựng một cây tìm kiếm phân cấp và thực hiện phân miền ở các mức khác nhau. Cụ thể ta có thể xây dựng như sau:

+ Một mạng được chia thành một tập hợp các miền . Trong đó một miền ở tầng cao nhất phủ toàn bộ mạng. Mỗi miền được chia tiếp thành các miền con, mỗi miền ở tầng thấp nhất gọi là miền lá, thường ứng với 1 mạng nội bộ trong mạng máy tính hoặc 1 cell trong một mạng điện thoại di động. Tập hợp các miền nói trên được biểu diễn bởi cây tìm kiếm, quy mô lớn trong đó, mỗi miền được đại diện 1 nút trong thư mục miền tầng cao nhất được đại diện 1 nút thư mục gốc.

+ Để lưu dữ thông tin về địa điểm của một thực thể E hiện đang nằm tại miền D, địa chỉ của E được lưu trong một location record đặt tại nút lá D. Các nút tổ tiên của D cũng ghi 1 location record về E trong đó con trỏ tới một nút con mà cây con đó có gốc tại nút con đó lưu trữ địa chỉ của thực thể E. Nút gốc có thông tin về tất cả các thực thể.

**Câu hỏi 9: So sánh liên kết vật lý và liên kết biểu tượng trong hệ thống quản lý tệp của UNIX.**

* Liên kết vật lý:
  + liên kết trực tiếp hai tệp trong cùng một hệ thống tệp và để nhận dạng, nó sử dụng số inode của tệp. Liên kết vật lý không thể được thực hiện trên các thư mục (vì chúng trỏ đến inode).
  + Cả tệp gốc và tệp được tạo đều có cùng một nút và nội dung; do đó họ sẽ có cùng quyền và cùng chủ sở hữu.
  + Việc loại bỏ tệp gốc không ảnh hưởng đến tệp được liên kết và tệp liên kết sẽ vẫn còn. Bất cứ khi nào bạn thực hiện bất kỳ thay đổi nào đối với liên kết cứng, nó sẽ bắt chước trong tệp gốc.
* Liên kết biểu tượng:
  + Thường là một đường dẫn thay thế (hoặc bí danh) cho tệp gốc. Khi một tệp được truy cập, liên kết mềm sẽ chuyển hướng nó đến tệp đích thông qua đường dẫn được viết trong chủ đề của liên kết mềm.
  + Liên kết mềm không sử dụng số inode, không giống như liên kết cứng. Một đường dẫn tuyệt đối hoặc tương đối có thể là một phần của các liên kết tượng trưng.
  + Tạo và xóa các liên kết mềm không ảnh hưởng đến tệp gốc. Nếu tệp được liên kết bị xóa, các liên kết mềm sẽ bị treo, điều đó có nghĩa là nó sẽ không xuất hiện và tạo ra một thông báo lỗi khi tệp đích được truy cập.

**Câu hỏi 10: Khi chúng ta thêm 1 node mới vào hệ thống Chord, chúng ta có cần phải cập nhật toàn bộ các bảng finger?**

Không cần cập nhật toàn bộ các bảng finger, chỉ cần cập nhật cho nút sau và nút trước của node mới, vì trong hệ thống Chord, mỗi nút sẽ quản lý khóa của nút trước đó

**Câu hỏi 11: Phân giải tên đệ qui có ưu điểm gì so với phân giải tên không đệ qui?**

Ưu điểm:

* Phân giải tên có đệ qui xử lý nhiều ở bên server, phân giải tên không đệ quy xử lý nhiều ở bên client. Khi client ở xa, phân giải tên không đệ quy phải gửi đi xa dẫn tới độ trễ cao.
* Phân giải tên có đệ qui các name server tổ chức lưu trữ bộ đệm, khi server lưu trữ kết quả có thể trả lời luôn và nhanh cho các client. - Tốn ít chi phí thông tin