

# Hệ thống tập tin Giao diện



## Bài tập thực hành

- 10.1** Một số hệ thống tự động xóa tất cả các tệp người dùng khi người dùng đăng xuất hoặc công việc kết thúc, trừ khi người dùng yêu cầu rõ ràng rằng chúng được giữ lại; các hệ thống khác giữ tất cả các tệp trừ khi người dùng xóa chúng một cách rõ ràng. Thảo luận về giá trị tương đối của mỗi cách tiếp cận.
- Trả lời:** Xóa tất cả các tệp không được người dùng lưu cụ thể có ưu điểm là giảm thiểu dung lượng tệp cần thiết cho mỗi người dùng bằng cách không lưu các tệp không mong muốn hoặc không cần thiết. Lưu tất cả các tệp trừ khi bị xóa cụ thể sẽ an toàn hơn cho người dùng ở chỗ không thể vô tình làm mất tệp do quên lưu chúng.
- 10.2** Tại sao một số hệ thống theo dõi loại tệp, trong khi những hệ thống khác lại để cho người dùng hoặc đơn giản là không triển khai nhiều loại tệp? Hệ thống nào là "tốt hơn?"
- Trả lời:** Một số hệ thống cho phép các hoạt động tệp khác nhau dựa trên loại tệp (ví dụ: tệp ascii có thể được đọc dưới dạng luồng trong khi tệp cơ sở dữ liệu có thể được đọc qua chỉ mục cho một khối). Các hệ thống khác để lại cách giải thích dữ liệu của tệp như vậy cho quá trình và không cung cấp trợ giúp nào trong việc truy cập dữ liệu. Phương pháp "tốt hơn" phụ thuộc vào nhu cầu của các quy trình trên hệ thống và nhu cầu của người dùng đối với hệ điều hành. Nếu một hệ thống chạy hầu hết các ứng dụng cơ sở dữ liệu, hệ điều hành có thể triển khai tệp kiểu dữ liệu và cung cấp các hoạt động hiệu quả hơn thay vì khiến mỗi chương trình thực hiện cùng một thứ (có thể theo những cách khác nhau). Đối với các hệ thống đa năng, có thể tốt hơn là chỉ triển khai các loại tệp cơ bản để giữ cho kích thước hệ điều hành nhỏ hơn và cho phép tự do tối đa đối với các quy trình trên hệ thống.

**10.3** Tương tự, một số hệ thống hỗ trợ nhiều kiểu cấu trúc cho dữ liệu của tệp, trong khi những hệ thống khác chỉ hỗ trợ một luồng byte. Ưu nhược điểm là gì?

**Trả lời:** Một lợi thế của việc hệ thống hỗ trợ các cấu trúc tệp khác nhau là sự hỗ trợ đến từ hệ thống; các ứng dụng riêng lẻ không bắt buộc phải cung cấp hỗ trợ. Ngoài ra, nếu hệ thống cung cấp hỗ trợ cho các cấu trúc tệp khác nhau, nó có thể triển khai hỗ trợ có lẽ hiệu quả hơn một ứng dụng. Nhược điểm của việc hệ thống cung cấp hỗ trợ cho các loại tệp xác định là nó làm tăng kích thước của hệ thống. Ngoài ra, các ứng dụng có thể yêu cầu các loại tệp khác với những gì được cung cấp bởi hệ thống có thể không chạy trên các hệ thống đó.

Một chiến lược thay thế là hệ điều hành xác định không hỗ trợ cấu trúc tệp và thay vào đó coi tất cả các tệp là một chuỗi byte. Đây là cách tiếp cận được thực hiện bởi các hệ thống UNIX. Ưu điểm của cách tiếp cận này là nó đơn giản hóa việc hỗ trợ hệ điều hành cho các hệ thống tệp, vì hệ thống không còn phải cung cấp cấu trúc cho các loại tệp khác nhau. Hơn nữa, nó cho phép các ứng dụng xác định cấu trúc tệp, do đó giảm bớt tình huống hệ thống có thể không cung cấp định nghĩa tệp cần thiết cho một ứng dụng cụ thể.

**10.4** Bạn có thể mô phỏng cấu trúc thư mục đa cấp với cấu trúc thư mục một cấp trong đó các tên dài tùy ý có thể được sử dụng không? Nếu câu trả lời của bạn là có, hãy giải thích cách bạn có thể làm như vậy và đối chiếu lược đồ này với lược đồ thư mục đa cấp. Nếu câu trả lời của bạn là không, hãy giải thích điều gì ngăn cản sự thành công của mô phỏng. Câu trả lời của bạn sẽ thay đổi như thế nào nếu tên tệp được giới hạn trong bảy ký tự?

**Trả lời:** Nếu các tên dài tùy ý có thể được sử dụng thì có thể mô phỏng cấu trúc thư mục đa cấp. Điều này có thể được thực hiện, chẳng hạn, bằng cách sử dụng ký tự "." để chỉ ra phần cuối của một thư mục con. Vì vậy, ví dụ, tên *jim.java.F1* chỉ định rằng *F1* là một tệp trong thư mục con *java* đến lượt nó nằm trong thư mục gốc *jim*.

Nếu tên tệp được giới hạn trong bảy ký tự, thì lược đồ trên không thể được sử dụng và do đó, nói chung, câu trả lời là *không*. Cách tiếp cận tốt nhất tiếp theo trong tình huống này là sử dụng một tệp cụ thể làm bảng ký hiệu (thư mục) để ánh xạ các tên dài tùy ý (chẳng hạn như *jim.java.F1*) thành các tên tùy ý ngắn hơn (chẳng hạn như *XX00743*), sau đó được sử dụng để truy cập tệp thực tế.

**10,5** Giải thích mục đích của *mở()* và *gần()* các hoạt động. **Trả lời:**

- Các *mở()* hoạt động thông báo cho hệ thống rằng tệp được đặt tên sắp hoạt động.
- Các *gần()* hoạt động thông báo cho hệ thống rằng tệp được đặt tên không còn được sử dụng tích cực bởi người dùng đã phát hành hoạt động đóng.

**10,6** Đưa ra một ví dụ về một ứng dụng trong đó dữ liệu trong tệp sẽ được truy cập theo thứ tự sau:

Một. Tuần tự

b. Ngẫu nhiên

**Trả lời:**

Một. In nội dung của tệp.

b. In nội dung bản ghi nhớ. Bản ghi này có thể được tìm thấy bằng cách sử dụng kỹ thuật bấm hoặc chỉ mục.

**10,7** Trong một số hệ thống, một thư mục con có thể được đọc và ghi bởi người dùng được ủy quyền, giống như các tệp thông thường.

Một. Mô tả các vấn đề bảo vệ có thể phát sinh.

b. Đề xuất phương án giải quyết từng vấn đề bảo vệ mà bạn đã nêu trong phần a.

**Trả lời:**

Một. Một phần thông tin được lưu giữ trong mục nhập thư mục là vị trí tệp. Nếu người dùng có thể sửa đổi vị trí này, thì anh ta có thể truy cập các tệp khác đánh bại kế hoạch bảo vệ quyền truy cập.

b. Không cho phép người dùng ghi trực tiếp vào thư mục con. Thay vào đó, cung cấp các hoạt động hệ thống để làm như vậy.

**10,8** Hãy xem xét một hệ thống hỗ trợ 5000 người dùng. Giả sử rằng bạn muốn cho phép 4990 người dùng này có thể truy cập vào một tệp.

Một. Bạn sẽ chỉ định sơ đồ bảo vệ này như thế nào trong UNIX?

b. Bạn có thể đề xuất một sơ đồ bảo vệ khác có thể được sử dụng hiệu quả hơn cho mục đích này so với sơ đồ được cung cấp bởi UNIX?

**Trả lời:**

Một. Có hai phương pháp để đạt được điều này:

1. Tạo danh sách kiểm soát truy cập với tên của tất cả 4990 người dùng.
2. Đặt 4990 người dùng này vào một nhóm và đặt quyền truy cập nhóm cho phù hợp. Không phải lúc nào cũng có thể triển khai lược đồ này vì các nhóm người dùng bị hệ thống hạn chế.

b. Quyền truy cập chung vào tệp áp dụng cho tất cả người dùng trừ khi tên của họ xuất hiện trong danh sách kiểm soát truy cập với quyền truy cập khác. Với lược đồ này, bạn chỉ cần đưa tên của mười người dùng còn lại vào danh sách kiểm soát truy cập nhưng không có đặc quyền truy cập nào được phép.

**10,9** Các nhà nghiên cứu đã gợi ý rằng, thay vì có một danh sách truy cập được liên kết với mỗi tệp (chỉ định người dùng nào có thể truy cập tệp và cách thức), chúng ta nên có *danh sách kiểm soát người dùng* được liên kết với từng người dùng (chỉ định tệp người dùng có thể truy cập và cách thức). Thảo luận về giá trị tương đối của hai chương trình này.

**Trả lời:**

## 38 Chương 10 **Giao diện hệ thống tệp**

- *Danh sách kiểm soát tệp.* Vì thông tin kiểm soát truy cập được tập trung ở một nơi duy nhất, nên việc thay đổi thông tin kiểm soát truy cập sẽ dễ dàng hơn và điều này đòi hỏi ít không gian hơn.
- *Danh sách kiểm soát người dùng.* Điều này yêu cầu ít chi phí hơn khi mở tệp.