

進階作業系統 - 作業2

系統保護與分散式同步

動機

您已經在第4章和第5章學習了系統保護。本作業要求您通過在計算機之間建立TCP/IP連接來實現這些章節中的一些概念。

作業描述

檔案系統權限基礎

在UNIX檔案系統中,每個檔案都具有讀取、寫入和執行的存取權限,分別針對檔案擁有者、群組成員和其他人。例如,假設碩士生Ken建立了一個名為"homework2.c"的檔案,具有讀取和寫入權限。他允許同一個"AOS"群組中的群組成員Barbie讀取該檔案,但不允許其他人存取該檔案。因此,您可以在Ken的目錄中看到檔案描述:

```
-rw-r----- Ken AOS 87664 Nov 20 2025 homework2.c
```

特別地,"homework2.c"有87664位元組,它由AOS群組中的Ken在2025年11月20日建立。Barbie可以讀取該檔案,因為第二個'r'是開啟的。其他人無法存取該檔案,因為第三組的r/w/x位元全部關閉。

第一部分:檔案管理伺服器

在本作業的第一部分,您需要建立一個伺服器來為客戶端管理檔案。應該建立兩組客戶端,即"AOS-group"和"CSE-group",每組至少包含三個客戶端。遵循UNIX檔案系統,您應該為每個檔案指定讀取和寫入權限,針對檔案擁有者、群組成員和其他人。當允許讀取(或寫入)檔案時,客戶端能夠下載(或上傳)檔案。如果客戶端請求對檔案進行無權限的操作,伺服器應該禁止它並列印出訊息以顯示原因。

每個客戶端都能夠動態建立檔案,但應該指定所有存取權限。例如,Ken可以執行以下命令:

命令範例:

- 1) new homework2.c rwrnnn
- 2) read homework2.c
- 3) write homework2.c o/a
- 4) change homework2.c rwnnnn

命令說明

- **new命令:** 幫助Ken在伺服器上建立新檔案,第三個參數給出檔案的權限 ('r'和'w'分別代表讀取和寫入權限,'n'表示沒有權限)。
- **read命令:** 允許Ken從伺服器下載檔案(只有當他有權限且檔案確實存在時)。
- **write命令:** 允許Ken上傳(並修改)現有檔案,第三個參數可以是'o'或'a',分別允許Ken覆蓋原始檔案或將其資料附加到檔案末尾。同樣,只有當Ken有相應權限且檔案確實存在時,他才能寫入檔案。
- **change命令:** 更改檔案的權限。修改後的權限在change命令之後的後續操作中生效。

重要提醒: 所有客戶端都在伺服器端的同一目錄中操作檔案。伺服器必須使用**能力列表(CAPABILITY LISTS)**來管理檔案的權限。您必須向助教展示伺服器端每次操作時能力列表如何變化。

第二部分:並發存取控制

當客戶端正在寫入檔案時,其他客戶端不能讀取或寫入同一檔案。此外,當客戶端正在讀取檔案時,其他客戶端不能寫入該檔案。然而,多個客戶端同時讀取同一檔案是安全的。

因此,在本作業的第二部分,您將被要求在伺服器-客戶端架構中應用上述規則。您需要向助教展示所有上述行為。更具體地說,您的伺服器必須能夠連接多個客戶端,並允許多個客戶端"並發地"讀取/寫入檔案(因此,您的檔案應該足夠大以展示上述行為)。

要求

您必須在UNIX相容系統中使用標準C socket程式庫並提供makefile。如果您不提供正確的makefile,助教將扣分。此外,不要忘記向助教展示您的程式(否則,您將獲得零分)。

評分政策

- 本作業不需要提交任何報告
- 第一部分和第二部分分別佔**60%**和**40%**
- 本作業的截止日期為**2025年12月11日**