LPI Certification

Exam 102-500
LPI Level 1 Certification Part 2

Exam Questions (2026-07-15 Updated)

102-500 Linux Professional Institute Level 1 (Jul 2026)

QUESTION 1
Which of the following is a valid IPv6 address?
A. **`2001:db8:0g21::1`**
B. **`2001::db8:4581::1`**
C. **`2001:db8:3241::1`**
D. **`2001%db8%9990%%1`**
E. **`2001.db8.819f..1`**

Answer: C
IPv6 位址的標準語法和格式規範（由 IETF 定義）包含以下核心規則：

- 十六進位字元：IPv6 使用十六進位（Hexadecimal）表示，合法的字元僅包含數字 0-9 以及英文字母 **`a-f`**（或大寫 **`A-F`**）。
- 冒號分隔：位址結構由 8 個區段（Hextets）組成，區段與區段之間固定使用**冒號（`:`）**來分隔。雙
- 冒號縮寫：允許使用唯一的**雙冒號（`::`）**來省略連續出現的多個全零區段，但全體位址中最多隻能出現一次 **`::`**，以防解析時產生長度歧義。

選項 C 符合所有規範：`2001:db8:3241::1` 的所有字元皆為合法的十六進位，使用冒號分隔，且僅包含一組 `::` 縮寫。

為什麼其他選項是錯誤的？
A. `2001:db8:0g21::1` 錯誤原因：字元不合法。第三個區段中包含了英文字母 **`g`**，而十六進位合法的字母範圍僅到 **`f`（`15`）**，因此 **`g`** 是無效字元。
B. `2001::db8:4581::1` 錯誤原因：出現了兩組雙冒號。此位址同時包含了**兩組 `::`**（前方一組、後方一組），這會導致核心無法依據數學邏輯推算出各自究竟代表省略了多少個全零區段，屬於嚴格禁止的錯誤語法。
D. `2001%db8%9990%%1` 錯誤原因：分隔符號錯誤。該位址完全錯誤地使用了百分比符號（`%`）來當作區段分隔線，不符合 IPv6 標準的**冒號（`:`）**規範。
E. `2001.db8.819f..1` 錯誤原因：分隔符號與縮寫錯誤。此選項混淆了 IPv4 的**點號（`.`）**與 IPv6 的雙點縮寫，IPv6 規定必須使用冒號與雙冒號，使用點號是完全無效的。

QUESTION 2
Which of the following statements is valid in the file **`/etc/nsswitch.conf`** ?
A. **`multi on`**
B. **`192.168.168.4 dns-server`**
C. **`namespaces: net mount procs`**
D. **`include /etc/nsswitch.d/`**
E. **`hosts: files dns`**

Answer: E

/etc/nsswitch.conf（Name Service Switch 組態檔） 用於定義系統查詢各種行政資料庫（如使用者密碼、主機名稱、網路協定等）的優先順序。

- 它的標準語法格式為：資料庫: 來源 1 來源 2 ...。
- **hosts: files dns** 表示當系統需要解析主機名稱（hosts）時，會先查詢本地的靜態檔案（files，即 **/etc/hosts**），如果找不到，再透過 DNS（Domain Name System） 服務進行解析。這完全符合該檔案的語法和功能。

為什麼其他選項是錯誤的？

A. multi on 錯誤原因： 這是 **/etc/host.conf** 檔案（舊版解析器組態檔）中的合法設定，用來指示是否允許單一主機名稱對應多個 IP 地址，而不是放在 **/etc/nsswitch.conf** 中。

B. 192.168.168.4 dns-server 錯誤原因： 這是 **/etc/hosts** 檔案的格式（IP 位址 後面接 主機名稱對應），不能寫在 **/etc/nsswitch.conf** 內。

C. namespaces: net mount procs 錯誤原因： 語法不正確。**namespaces** 不是 **/etc/nsswitch.conf** 所支援的標準行政資料庫名稱。

D. include /etc/nsswitch.d/ 錯誤原因： 雖然較新版本的 GNU C 函式庫（**glibc**）引進了 **include** 功能（通常配合特定的檔名），但在標準的 **/etc/nsswitch.conf** 中，直接指定一個目錄而沒有完整的格式、或當作常規配置項目在 Linux 認證考試（如 LPIC-1）的標準設計中均被視為錯誤選項。

QUESTION 3

Which of the following connection types, as seen in **nmcli connection show**, may exist in NetworkManager? (Choose **THREE** correct answers.)

A. **tcp**

B. **ethernet**

C. **wifi**

D. **ipv6**

E. **bridge**

Answer: B, C, E

當執行 nmcli connection show 指令時，NetworkManager 會列出所有已配置的網路連線設定檔。其中的 TYPE 欄位代表該連線所使用的物理或邏輯接口類型。

- **B. ethernet**：用於常見的有線區域網路（Wired Ethernet）連線。
- **C. wifi**：用於 IEEE 802.11 無線網路連線。
- **E. bridge**：用於建立網路橋接器（Network Bridge），將多個網路介面組合在同一個交換網段中。

除了這三個選項外，NetworkManager 支援的其他合法連線類型還包括 **loopback**、**tun**、**pppoe**、**gsm** 和 **infiniband** 等。

為什麼其他選項是錯誤的？

A. tcp 錯誤原因： TCP（傳輸控制協定）是傳輸層的網路協定。它負責資料包的可靠傳輸，並不是 NetworkManager 管理的底層硬體或虛擬接口連線類型。

D. ipv6 錯誤原因： IPv6 是網路層的網際網路協定（IP Protocol）。在 NetworkManager 中，IPv6 屬於連線設定檔內部的「屬性（Property）」或「配置區段」，例如 **ipv6.method** 或 **ipv6.addresses**，而非獨立的連線類型。

QUESTION 4

What is true about the Hop Limit field in the IPv6 header?

A. The field is not changed during the transport of a package.

B. The field is transmitted within a hop-by-hop extension header.

C. Each router forwarding the packet increases the field's value.

D. Each router forwarding the packet decreases the field's value.

E. For multicast packages, the field's value is always **1**.

Answer: D

IPv6 標頭中的 **Hop Limit（跳數限制）** 欄位長度為 8 位元，其作用與 IPv4 中的 **TTL（Time to Live，存活時間）** 欄位完全相同，主要用於**防止網路封包陷入死循環（Routing Loops）**。

- 當來源主機發送 IPv6 封包時，會為該欄位設定一個初始值（例如 **64** 或 **128**）。
- **每當封包經過一台路由器並被轉發（Forwarding）時，該路由器就會將 Hop Limit 的值減 1。**
- 如果 Hop Limit 的值減到 **0**，路由器就會丟棄該封包，並向來源主機發送一個 **ICMPv6 "Time Exceeded"**（逾時）的訊息。

為什麼其他選項是錯誤的？

A. The field is not changed during the transport of a package. 錯誤原因： 如上所述，封包在傳輸過程中，每經過一台路由器轉發，Hop Limit 的值就會被修改（減 1），並不是一成不變的。

B. The field is transmitted within a hop-by-hop extension header. 錯誤原因： Hop Limit 是位於 IPv6 基本標頭（Base Header）中的固定欄位，並不是寫在 逐跳擴充標頭（Hop-by-Hop Options Extension Header）裡面。

C. Each router forwarding the packet increases the field's value. 錯誤原因： 路由器是減少（Decrease）該數值，而不是增加（Increase）。如果增加會導致封包永遠無法被消除。

E. For multicast packages, the field's value is always 1.錯誤原因： 雖然群播（Multicast）封包有時會將 1 作為預設的 Hop Limit 以限制傳播範圍（例如只留在本地網段），但這並非絕對（not always）。群播封包的 Hop Limit 可以根據需要設定為大於 1 的任何合法數值，以便跨越路由器傳播。

QUESTION 5

Which of the following **nmcli** subcommands exist? (Choose **TWO** correct answers)

A. **nmcli ethernet**

B. **nmcli device**

C. **nmcli wifi**

D. **nmcli address**

E. **nmcli connection**

Answer: B, E

在 Linux 系統中，**nmcli** 是用來控制 NetworkManager 並報告網路狀態的命令列工具。它的指令結構通常是 **nmcli [選項] 對象 [指令]**，其中「對象（Object）」就是**主子命令（Subcommand）**。

- **B. nmcli device：device**（可簡寫為 **d**）是合法的子命令，主要用來管理和查看硬體網路介面（如 **eth0**、**wlan0** 等）的狀態。常用指令如 **nmcli device status** 或 **nmcli device**

show。

- **E. nmcli connection：connection**（可簡寫為 **c**）是合法的子命令，主要用來管理、建立、修改和刪除網路連線設定檔。常用指令如 **nmcli connection show** 或 **nmcli connection up <連線名稱>**。

為什麼其他選項是錯誤的？

A. nmcli ethernet 錯誤原因： **ethernet** 並不是 **nmcli** 的頂層子命令。它是某個連線在建立或查看時所屬的「連線類型（Type）」，而非獨立的子命令。

C. nmcli wifi 錯誤原因： 雖然我們經常用到 **Wi-Fi** 功能，但 wifi 在 **nmcli** 中並不是第一層的子命令。若要管理 Wi-Fi，正確的完整指令架構是將 **wifi** 作為 **device** 的下層操作，例如：**nmcli device wifi list** 或 **nmcli device wifi connect**。

D. nmcli address 錯誤原因： **nmcli** 沒有名為 **address** 的子命令。如果需要配置或檢視 IP 地址，通常是在 **nmcli connection modify** 中設定 **ipv4.addresses** 屬性，或者直接使用系統自帶的 **ip address** 指令。

QUESTION 6

How many IP addresses can be used for unique hosts inside the IPv4 subnet **192.168.2.128/26** ?

A. 6

B. 14

C. 30

D. 62

E. 126

Answer: D

核心計算步驟計

1. **算主機位元數（Host Bits）**

   IPv4 地址總共有 32 位元。遮罩長度為 /26，代表前 26 位元是網路部分，剩下的位元留給主機。

   (32 – 26 = 6 個主機位元)

2. **計算總 IP 位址數量**

   利用 6 個主機位元，總共可以組合出 2^6 個不同的 IP 位址：

   (2^6 = 64 個 IP)

3. **扣除無法分配給主機的保留位址**

   在任何常規的 IPv4 子網路中，都必須扣除 2 個 具有特殊用途的位址，無法指派給實體主機：

   - **網路位址（Network Address）**：主機位元全為 0 的位址，此例中為 192.168.2.128。
   - **廣播位址（Broadcast Address）**：主機位元全為 1 的位址，此例中為 192.168.2.191。

4. **計算可用主機 IP 數量（Valid Hosts）**

   將總數減去 2 即為答案：

   (64 - 2 = 62 個可用主機 IP)

QUESTION 7
Which of the following IPv4 networks are reserved by IANA for private address assignment and private routing? (Choose **THREE** correct answers)

A. `10.0.0.0/8`
B. `127.0.0.0/8`
C. `169.255.0.0/16`
D. `172.16.0.0/12`
E. `192.168.0.0/16`

Answer: A, D, E

根據網際網路號碼分配局（IANA）以及 RFC 1918 規範，共有三個 IPv4 位址區段被保留給 私有網路（Private Network）內部指派與路由使用。這些位址不會在全球網際網路（Public Internet）上被公開路由：

- **A. `10.0.0.0/8`（Class A Private address）**：範圍從 **10.0.0.0** 到 **10.255.255.255**，常用於大型企業內部網路。

- **D. `172.16.0.0/12`（Class B Private address）**：範圍從 **172.16.0.0** 到 **172.31.255.255**，包含 16 個連續的 B 類子網路。

- **E. `192.168.0.0/16`（Class C Private address）**：範圍從 **192.168.0.0** 到 **192.168.255.255**，廣泛應用於家用路由器和小型辦公室（SOHO）網路。

為什麼其他選項是錯誤的？

B. `127.0.0.0/8` 錯誤原因： 這是保留給環回測試（Loopback）使用的網段。其中最著名的位址是 127.0.0.1（localhost），用於指向本機作業系統自身進行網路軟體測試，並非用於私有網路的主機配置與路由。

C. `169.255.0.0/16` 錯誤原因： 數值錯誤。IANA 保留給 APIPA（自動登錄私有 IP 尋址） 或稱為本地連結位址（Link-Local Addresses）的正確網段是 **169.254.0.0/16**。當 DHCP 伺服器失效時，作業系統會在此範圍內隨機指派一個位址給網路卡，而選項中的 **169.255.0.0/16** 是一般的公網（Public IP）網段。

QUESTION 8
Which of the following commands configure network interfaces based on the system's existing distribution-specific configuration files? (Choose **TWO** correct answers.)

A. `ifconfig`
B. `ifdown`
C. `ifpause`
D. `ifstart`
E. `ifup`

Answer: B, E

在 Linux 系統中，**ifup** 和 **ifdown** 是高階的網路介面管理指令。它們與底層直接修改核心狀態的指令（如 **ip** 或 **ifconfig**）不同，這兩個指令完全依賴發行版特定的文字組態檔來讀取 IP、閘道和 DNS 設定。

E. ifup：用於啟動（Enable）指定的網路介面。它會去讀取組態檔並自動套用設定。

B. ifdown：用於關閉（Disable）指定的網路介面，並釋放相關的網路資源（例如透過 DHCP 取得的租約）。

各 Linux 發行版對應的組態檔位置範例：

Debian / Ubuntu 系：讀取 **/etc/network/interfaces** 檔案。

RHEL / CentOS 系 (早期與相容模式)：讀取 **/etc/sysconfig/network-scripts/ifcfg-<介面名稱>** 檔案。

為什麼其他選項是錯誤的？

A. ifconfig 錯誤原因： 這是用於直接檢視或手動配置網路介面的工具。如果您使用 **ifconfig eth0 192.168.1.10**，它是直接在核心（Kernel）即時生效，完全不會去參考或讀取系統內建的發行版特定組態檔。

C. ifpause 錯誤原因： Linux 系統中並不存在名為 **ifpause** 的網路管理指令。

D. ifstart 錯誤原因： Linux 標準網路指令集中沒有 **ifstart**。雖然名字聽起來像啟動，但正確用於啟動發行版設定的指令是 **ifup**。

QUESTION 9

Which of the following information is stored in **/etc/shadow** for each user?

A. The timestamp of the user's last login

B. The user's private SSH keys

C. The hashed password of the user

D. The numerical user ID (UID)

E. The path to the user's home directory

Answer: C

/etc/shadow（影子密碼檔）是一個專門用來安全存放 Linux 系統使用者認證資訊的文字檔案。

- 為了解決 **/etc/passwd** 必須開放給所有使用者讀取（World-Readable）的安全漏洞，Linux 系統將加密處理後的密碼雜湊值（Hashed Password）單獨抽離出來，存放在只有 **root** 系統管理員有權限讀取的 **/etc/shadow** 檔案中。
- 除了密碼雜湊值，該檔案還紀錄了密碼過期時間、帳號失效時間等與密碼安全週期相關的欄位。

為什麼其他選項是錯誤的？

A. The timestamp of the user's last login 錯誤原因： 使用者最後一次登入的時間戳記通常存放在二進位日誌檔案中（例如 **/var/log/wtmp** 或 **/var/log/lastlog**），可以透過 **last** 或 **lastlog** 指令查詢，而不是放在 **/etc/shadow**。

B. The user's private SSH keys 錯誤原因： 使用者的 SSH 私鑰是高機密性的個人憑證，固定存放在使用者家目錄下的隱藏資料夾中，路徑為 **~/.ssh/id_rsa**（或其他演算法名稱），絕不會寫在系統的全

域組態檔中。

D. The numerical user ID (UID)錯誤原因： 使用者的數值識別碼（UID）以及主要群組識別碼（GID）是存放在 **`/etc/passwd`** 檔案的第三和第四欄位中。**`/etc/shadow`** 是透過「使用者名稱（Username）」來與 **`/etc/passwd`** 進行關聯，檔案內並不包含 UID。

E. The path to the user's home directory 錯誤原因： 使用者家目錄的絕對路徑（Home Directory）同樣是記錄在 **`/etc/passwd`** 的第六欄位中，例如 **`/home/username`**。

QUESTION 10

Which of the following commands shows all active systemd timers?

A. **`systemctl-timer show`**

B. **`timectl list`**

C. **`systemctl -t`**

D. **`systemctl list-timers`**

E. **`timeq`**

Answer: D

在 Linux 系統中，**`systemd`** 使用定時器單元（Timer Units，副檔名為 **`.timer`**）來處理排程任務（類似於傳統的 **`cron`**）。

- 檢視這些定時器的標準管理工具是 **`systemctl`**。執行 **`systemctl list-timers`** 會列出目前系統中所有處於啟用/作用狀態（**active**）的定時器。
- 該指令輸出的資訊非常直觀，會清楚顯示定時器下一次執行的時間（**NEXT**）、距離下一次執行還有多久（**LEFT**）、上一次執行的時間（**LAST**）、距離上一次執行過了多久（**PASSED**），以及定時器對應觸發的服務名稱（**UNIT**）與啟用它的檔案（**ACTIVATED**）。
- 提示：若要連同「未啟用（**inactive**）」的定時器一併列出，可以加上參數 **`systemctl list-timers --all`**。

為什麼其他選項是錯誤的？

A. `systemctl-timer show` 錯誤原因： 系統中沒有 **`systemctl-timer`** 這個獨立指令。**`systemctl`** 本身不使用連字號（**`-`**）來連接子命令。

B. `timectl list` 錯誤原因： Linux 中沒有 **`timectl`** 這個指令。與系統時間、時區控制相關的正確 **`systemd`** 工具名稱為 timedatectl，而它也沒有 **`list`** 這種列出定時器的語法。

C. `systemctl -t` 錯誤原因： **`-t`**（或 **`--type`**）是 **`systemctl`** 的一個「篩選參數」，後面必須指定單元類型（例如 **`systemctl -t timer`** 才能列出定時器類型，或 **`systemctl -t service`** 篩選服務），直接單獨輸入 **`systemctl -t`** 屬於錯誤或不完整的語法。

E. `timeq` 錯誤原因： Linux 系統與 **`systemd`** 套件中完全不存在名為 **`timeq`** 的指令。

QUESTION 11

Which of the following tasks can the **date** command accomplish? (Choose two.)

A. Set the system's date and time.

B. Set the system's date but not the time.

C. Calculate the time span between two dates.

D. Print a calendar for a month or a year.

E. Display time in a specific format.

Answer: A, E

在 Linux 系統中，**date** 指令是一個非常基礎且多功能的系統時間工具，它主要負責兩大核心任務：

- **Set the system's date and time.（設定系統的日期與時間）**
  - 說明： 系統管理員（**root**）可以使用 **date** 指令配合 **-s**（或 **--set**）參數來修改系統當前的時鐘。
  - 範例： 執行 **date -s "2026-06-08 07:15:00"** 可以同時把系統的日期改為 2026 年 6 月 8 日，並把時間設定為早上 7 點 15 分。
- **Display time in a specific format.（以特定格式顯示時間）**
  - 說明： 常規使用者執行 **date** 時預設會顯示目前時間。如果加上 + 號和格式化控制碼，就能讓時間依據自訂格式輸出，這在編寫指令碼（Shell Script）命名備份檔案時非常實用。
  - 範例： 執行 **date "+%Y-%m-%d %H:%M:%S"** 會輸出形如 **2026-06-08 07:15:00** 的標準格式文字。

為什麼其他選項是錯誤的？

B. Set the system's date but not the time. 錯誤原因： 當使用 date 設定日期時（例如 **date -s "2026-06-08"**），如果沒有指定時間，系統會自動把時間歸零重設為子夜 **00:00:00**，因此它不具備「唯獨修改日期、卻能完全保留原有時間不變」的功能。

C. Calculate the time span between two dates. 錯誤原因： **date** 指令雖然支援讀取特定時間（如 **date -d "next Friday"**），但它本身不包含直接計算並輸出兩個特定日期之間相差幾天、幾小時的「時間跨度計算」內建功能（通常需要結合算術運算 **$[ ]** 與時間戳記相減來實作）。

D. Print a calendar for a month or a year. 錯誤原因： 在 Linux 中，用於列印月曆或年曆的專屬指令是 **cal**（或 **ncal**），並不是 **date** 指令的功能。

QUESTION 12

What is true about the file **/etc/localtime**?

A. It is a plain text file containing a string such as **Europe/Berlin**.

B. It is created and maintained by the NTP service based on the location of the system's IP address.

C. It is a symlink to **/sys/device/clock/ltime** and always contains the current local time.

D. After changing this file, **newtzconfig** has to be run to make the changes effective

E. It is either a symlink to or a copy of a timezone information file such as **/usr/share/zoneinfo/Europe/Berlin.**

Answer: E

`/etc/localtime` 是 Linux 系統用來定義目前本地時區（Timezone）的核心組態檔。

- 系統中所有的通用時區數據檔案（編譯後的二進位 TZif 格式檔案）都存放在 **`/usr/share/zoneinfo/`** 目錄下。
- 當管理員要變更系統時區時，最標準的做法就是將 **`/etc/localtime`** 建立成一個指向對應時區檔案的符號連結（Symlink）。
  - 現代方法範例： 執行 **`timedatectl set-timezone Europe/Berlin`**，系統會自動在背景將 **`/etc/localtime`** 重新鏈接到 **`/usr/share/zoneinfo/Europe/Berlin`**。
  - 傳統手動範例： **`ln -sf /usr/share/zoneinfo/Europe/Berlin /etc/localtime`**。
- 在某些特殊環境或不支援軟連結的系統中，它也可能直接是該時區檔案的一份完整複製本（Copy）。

為什麼其他選項是錯誤的？

A. It is a plain text file containing a string such as Europe/Berlin. 錯誤原因： /etc/localtime 是二進位格式的數據檔案（含有時區規則、日光節約時間變更點等），並不是純文字檔。包含時區字串（如 **Europe/Berlin**）的純文字檔通常是 **`/etc/timezone`**（**Debian 系**）或 **`/etc/sysconfig/clock`**（舊版 **RHEL** 系）。

B. It is created and maintained by the NTP service based on the location of the system's IP address. 錯誤原因： 時區必須由系統管理員手動指定或透過安裝程式設定。NTP（網路時間協定）服務只負責將系統的「世界協調時間（UTC）」與遠端時間伺服器同步，完全不會根據 IP 地理位置去自動修改本地時區檔案。

C. It is a symlink to `/sys/device/clock/ltime` and always contains the current local time. 錯誤原因： **`/sys/`** 是核心虛擬檔案系統，並不存在 **`/sys/device/clock/ltime`** 這樣的時間路徑。此外，**`/etc/localtime`** 儲存的是「時區規則與定義」，而不是用來動態刷新並存放當前幾點幾分的檔案。

D. After changing this file, newtzconfig has to be run to make the changes effective 錯誤原因： 變更 **`/etc/localtime`** 的軟連結後，變更會立即生效（所有新啟動的進程都會讀取新時區），完全不需要執行任何名為 **`newtzconfig`** 的指令。