1. DLL
2. DLL是什麼 : 是一種包含可供多個程式同時使用的程式碼和資料的函式庫，當程式有需求時函式才進行連結
3. DLL Hijacking : 攻擊者能夠利用Windows載入DLL的規則，將需要劫持程式目錄下的合法DLL替換成惡意DLL，並壤程式連結到此惡意DLL造成損害
4. 降低其風險 : 載入記憶體中驗證其數位簽章或是雜湊值判斷是否為合法DLL檔
5. Cache
6. 是什麼?功用?

→ 介於主記憶體與處理器間的小容量、小型、高速儲存裝置

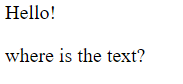
→ 為了讓資料存取的速度適應CPU的處理速度，讓cpu可以先藉由讀取速度較快的cache中存取，減少從主記憶體中讀取的速度

→ 原理 : 主要是因為程式運行時對記憶體的存取呈現局部性（Locality）特徵

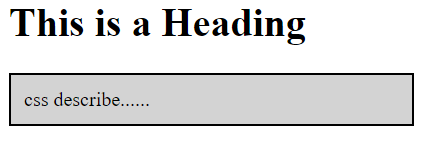
1. 比較write-through write-back優缺點

|  | 優 | 缺 |
| --- | --- | --- |
| write-through | 同時寫入，確保cache與memory資料的一致性 | 效能較差，需同時寫入cache與memory |
| write-back | 執行效能較高，只須更新cache，等cache被替換時才寫回記憶體 | 較高資料遺失、不一致的風險 |

1. 電腦網路
2. mesh : 可以保持每個節點間的連線完整，當網路拓蹼中有某節點失效或無法服務時，這種架構允許使用「跳躍」的方式形成新的路由後將訊息送達傳輸目的地
3. hash - 完整性 : 由於hash fuction的特性，輸入的數值若有不同的話則會生雪崩效應導致輸出的值不同，因此可以藉由判斷原有訊息的hash值與接收到的資訊產生的hash值是否一致判斷其完整性
4. FHSS : 將可用頻帶切割成多個狹窄的 channel，發送者會在各頻道中跳躍，送出資料。接收端要以當初約定好的跳躍樣式(Pattern) 去各相對頻道接收訊息，非送出與接收方很難得知傳輸的通道以達成防止竊聽的目的
5. Data Link Layer、Transport Layer
6. 將訊號的頻譜搬移到任意位置，從而有利於訊號的傳送，並且使頻譜資源得到充分利用
7. html



1. html



1. CPM中的 float/slack

→ 某項活動的寬裕(Slack)時間亦稱為自由浮時 (Free Float )係指該項活動可以待工或延後之時間量，而不影響其接續活動之最早開始日期

→ 專案間的執行先後順序可以根據slack排出後費時間最短的先後順序路徑

1. Gantt Chat (不確定)

→ 時間延誤

→ 需求修改

1. SQL

→ select 學生.姓名, 學生.系名, 課程.課程名稱 From 學生, 修課, 課程 Where學生.學號 = 課程.學號 AND 修課.課程代號 = 課程.課程代號

1. ERD 圖



1. Downcast
2. 欲將以向上轉型物件作為子類別使用時
3. 原有物件可能不具子類別的功能因此導致執行失敗；採用泛型或是編譯前確認使用物件是否符合規範
4. OOP程設