# 113 年特種考試地方政府公務人員及離島地區公務人員考試試題

考試別：地方政府公務人員、離島地區公務人員考試　　曹勝老師解題

等　別：三等考試

類　科：資訊處理

科　目：資通網路與安全

一、針對資安攻擊與防禦，請回答下列問題：

㈠請說明防火牆（Firewall）與入侵防禦系統（Intrusion Prevention Systems, IPS）如何進行合作以防禦阻斷服務攻擊（Denial-of-Service,DoS）。若二者的合作仍無法完全防禦阻斷服務攻擊，請說明其可能之原因。（15 分）

㈡請說明何謂分散式阻斷服務攻擊（DistributedDenial-of-Service,DDoS）；並說明系統管理者應如何加強防範分散式阻斷服務攻擊的事件發生。（10 分）

1.《考題難易》★★★☆☆
2.《解題關鍵》防火牆與 IPS 透過分層防禦與規則更新協作應對 DoS 攻擊，但對大規模 DDoS 攻擊需採用高效能設備、多層防禦機制、流量監控及驗證措施等綜合手段應對。
3.《命中特區》AF24 資通安全講義 P3-16~P3-22

【擬答】

㈠防火牆與入侵防禦系統（IPS）如何合作防禦 DoS 攻擊

1. 防火牆的作用

⑴封鎖不必要的流量：防火牆可根據預設規則（如 IP 位址、埠號或協定）封鎖異常流量。

⑵限制流量頻率：設置速率限制（Rate Limiting），避免單一來源流量過高。

⑶防止未經授權的連接：防火牆主要防止未經授權的存取，確保系統僅處理合法流量。

2. 入侵防禦系統（IPS）的作用

⑴深度封包檢測（Deep Packet Inspection, DPI）：IPS 能分析流量內容，識別惡意模式（如重複封包、異常請求）。

⑵實時阻斷攻擊：IPS 能在發現異常行為後自動封鎖特定攻擊源或流量。

⑶行為分析：IPS 透過異常偵測機制，發現偏離正常行為的攻擊活動。

3. 防火牆與 IPS 的合作機制

⑴分層防禦：防火牆負責過濾基礎規則層面的流量（如來源 IP 限制）。IPS 則深入分析通過防火牆的流量，檢測更複雜的攻擊模式。

⑵即時更新規則：IPS 可將識別到的新攻擊特徵（如惡意 IP）通報給防火牆，更新封鎖規則。

4. 防火牆與 IPS 可能無法完全防禦 DoS 攻擊的原因

⑴攻擊規模過大：攻擊者可能發送大量流量，超出系統或網路帶寬的處理能力。

⑵分散來源的特性：如攻擊流量來自大量合法但受感染的設備，防火牆與 IPS 難以區分合法與惡意流量。

⑶資源耗盡攻擊：某些 DoS 攻擊（如慢速 HTTP 攻擊）可能針對應用層，模擬合法流量，繞過基礎防護措施。

㈡分散式阻斷服務攻擊（DDoS）與防範措施

1.何謂分散式阻斷服務攻擊（DDoS）

DDoS 是指攻擊者利用大量受感染的設備（如殭屍網路）同時向目標系統發送大量請求或流量，導致資源耗盡，系統無法為正常使用者提供服務。特徵：

⑴來源分散：攻擊流量來自全球多個 IP 位址，增加防禦難度。

⑵流量規模巨大：可能達到 Tbps 級別。

⑶多層次攻擊：可涵蓋網路層（如 SYN Flood）、傳輸層（如 UDP Flood）及應用層（如 HTTP Flood）。

2.防範 DDoS 攻擊的措施

⑴強化網路基礎架構

①部署高效能設備：使用高性能路由器、防火牆及 IPS，提升流量處理能力。

②擴展網路帶寬：確保帶寬足以承受高流量，減少單點故障。

⑵建立多層防禦機制

①內容交付網路（CDN）：將流量分散至多個伺服器節點，降低單一伺服器負載。

②DDoS 防護服務：使用專業防護服務（如 Cloudflare、AWS Shield），自動檢測並緩解攻擊。

③負載平衡（Load Balancing）：分散流量至多個伺服器，避免過載。

⑶流量監控與異常偵測

①實時流量分析：部署網路監控系統，快速發現異常流量模式。

②啟用自動化封鎖：對異常行為 IP 實施臨時封鎖（Rate Limiting 或 Blackhole Routing）。

⑷防禦應用層攻擊

①驗證機制：實施 CAPTCHA 或多因素驗證，確保請求來自合法使用者。

②應用程式優化：增強伺服器應用程式的效能，減少惡意流量對資源的消耗。

⑸減少殭屍網路的影響

①教育用戶：提高用戶安全意識，減少受感染的設備數量。

②ISP 合作：與網際網路服務供應商協作，過濾惡意流量。

二、請回答下列負載平衡（ Load Balancing）與網路架構（ Network Architecture）的問題：

㈠在實現負載平衡過程中，請說明循環法（Round Robin）和加權循環法（Weighted Round Robin）的差異。（15 分）

㈡請說明當使用者請求頻繁切換至不同的伺服器時，可能會遇到的問題；並說明透過會話黏著（Session Stickiness）解決此問題的可行方法。（10 分）

1.《考題難易》★★★★☆

2.《解題關鍵》循環法平均分配請求，加權循環法根據伺服器性能調整分配比例；會話黏著透過 Cookie、IP 或內部追蹤解決多伺服器切換導致的會話狀態丟失問題，提高用戶體驗但需管理負載不均風險。

【擬答】

㈠循環法（Round Robin）與加權循環法（Weighted Round Robin）的差異

1.循環法（Round Robin）

工作原理是將請求按順序輪流分配給伺服器，每台伺服器都獲得相等數量的請求。例如，有三台伺服器 S1、S2、S3，請求分配依序為：S1→S2→S3→S1→S2→S3...。適合伺服器性能均等且負載壓力類似的環境。

2.加權循環法（Weighted Round Robin）

工作原理是在循環法基礎上，根據伺服器性能或容量設置權重（Weight），請求分配比例依據權重調整。例如，若 S1、S2、S3 的權重分別為 3:2:1，請求分配將為：S1→S1→S1→S2→S2→S3→S1→S1→S1→S2→S2→S3...。當伺服器性能不均時，可根據伺服器的處理能力進行更合理的負載分配。

3.主要差異

| 特徵 | 循環法 | 加權循環法 |
|---|---|---|
| 分配方式 | 請求平均分配 | 根據伺服器權重分配 |
| 適用環境 | 伺服器性能均等 | 伺服器性能不均 |
| 配置靈活性 | 配置簡單 | 需設定伺服器權重 |

㈡頻繁切換伺服器的問題與會話黏著（Session Stickiness）的解決方法

1.頻繁切換伺服器的問題

當使用者的請求在多個伺服器間切換時，可能導致以下問題：

⑴會話狀態丟失：使用者在伺服器 S1 發起的請求產生了一個會話狀態（如登入資訊、購物車內容）。下一個請求被分配到伺服器 S2，但 S2 無法獲取 S1 的會話資訊，導致用戶需重新登入或狀態重置。

⑵使用體驗不一致：不同伺服器的緩存或配置可能不同，頻繁切換會導致不一致的體驗。

⑶資源浪費：當伺服器無法共享會話狀態時，伺服器需要重新生成用戶會話，增加資源消耗。

2.透過會話黏著（Session Stickiness）解決問題

會話黏著（Session Stickiness）確保同一使用者的請求在一段時間內始終分配到同一伺服器。實現方法包括基於 Cookie、IP 地址或負載平衡器內部的會話追蹤。會話黏著的實現方式

⑴基於 Cookie 的黏著：負載平衡器在使用者首次請求時設置 Cookie，標記該用戶應分配的伺服器。後續請求根據 Cookie 將請求定向到同一伺服器。

⑵基於 IP 地址的黏著：根據用戶的來源 IP，將所有來自同一 IP 的請求分配到特定伺服器。

⑶負載平衡器內部追蹤：負載平衡器內部記錄會話與伺服器的映射關係，根據該映射將請求定向到適當的伺服器。

3.會話黏著的優點

⑴確保會話狀態一致性，提高使用者體驗。

⑵減少伺服器重新生成會話的資源浪費。

4.會話黏著的限制

⑴負載不均風險：某些伺服器可能因大量用戶黏著而過載。

⑵伺服器故障問題：若伺服器崩潰，用戶需重新建立會話或轉移到其他伺服器。

三、針對 SQL 注入攻擊（SQL Injection），請回答下列問題：

㈠請說明何謂 SQL 注入攻擊；並說明此攻擊的常見實務案例。（15 分）

㈡請說明使用預備語句（Prepared Statements）可以有效防止 SQL 注入攻擊的原因以及其運作原理。（10 分）

1.《考題難易》★★☆☆☆
2.《解題關鍵》識別 SQL 注入的攻擊方式及其危害，並透過預備語句的參數化查詢與自動轉義機制，有效防止惡意輸入操控 SQL 語法，提升安全性與效率。
3.《命中特區》AF24 資通安全講義 P2-16~P2-18

【擬答】

㈠SQL 注入攻擊的說明與實務案例

1. SQL 注入攻擊的定義

SQL 注入攻擊（SQL Injection）是一種透過將惡意的 SQL 指令嵌入到應用程式的輸入字段（如網頁表單或 URL）來操作後端數據庫的攻擊方式。攻擊者藉由操控查詢語句，實現以下目的：

⑴獲取未授權的數據訪問。

⑵修改或刪除數據庫中的資料。

⑶破壞數據庫結構或進行其他惡意行為。

2. 常見實務案例

未處理用戶輸入的查詢，

範例代碼：

```
SELECT *
FROM users
WHERE username = 'user_input' AND password = 'password_input';
```

如果攻擊者輸入以下值：

username: admin' OR '1'='1

password: any_password

SQL 查詢將被解析為：

```
SELECT *
FROM users
WHERE username = 'admin' OR '1'='1' AND password = 'any_password';
```

由於 '1'='1' 始終為真，攻擊者可繞過驗證進入系統。

3. 批量數據操控

在查詢中插入結束語句的符號 ;，添加額外的 SQL 指令：

```
SELECT *
FROM users
WHERE username = 'admin';
DROP TABLE users; --';
```

上述語句在執行後將刪除 users 表。

4. 資料泄露

攻擊者通過構造 UNION 查詢合併敏感資料：

```
SELECT username, password
FROM users
WHERE id = 1 UNION
  SELECT credit_card_number, cvv
  FROM payments;
```

㈡預備語句（Prepared Statements）的防護機制與運作原理

1. 預備語句的作用

預備語句是 SQL 查詢的一種編程方式，允許開發者事先定義查詢結構，並將變數值作為參數單獨傳遞給查詢執行。這種方法能防止惡意輸入被直接嵌入查詢語句。

2. 防止 SQL 注入的原因

⑴參數化查詢：預備語句將用戶輸入視為查詢的參數，而不是語法的一部分，從而避免了用戶輸入直接影響 SQL 語法結構。

⑵自動處理特殊字符：預備語句會對輸入數據中的特殊字符（如單引號 '）進行適當轉義，防止其被解釋為 SQL 語法。

3. 預備語句的運作原理

⑴預編譯查詢：預備語句在數據庫中首先進行語法編譯，形成固定的查詢模板。範例：

```
SELECT * FROM users WHERE username = ? AND password = ?;
```

此模板中的 ? 是占位符，用於表示查詢參數。

⑵綁定參數值：使用者的輸入值會作為參數綁定到模板的占位符，而不是直接嵌入查詢語句中。

⑶執行查詢：查詢執行時，數據庫根據編譯後的模板執行查詢，而不會重新解析參數內容。

4. 預備語句的範例

使用預備語句的代碼示例（以 PHP 為例）：

```
$stmt = $conn->prepare("SELECT * FROM users WHERE username = ? AND password = ?");
$stmt->bind_param("ss", $username, $password); // ss 表示兩個字符串參數
$stmt->execute();
```

5. 預備語句的優勢

⑴增強安全性：有效防止 SQL 注入攻擊。

⑵提高效率：重複執行相同查詢時，數據庫僅需編譯一次。

⑶易於維護：查詢與數據分離，提高代碼可讀性與可擴展性。

四、零信任（Zero Trust）架構被視為現代網路安全的新標準，它強調「永不信任，始終驗證」的安全策略。請說明在實施零信任架構時企業可能面臨的挑戰，以及相應的解決方案。（25 分）

1.《考題難易》★★☆☆☆

2.《解題關鍵》識別零信任架構實施過程中的五大挑戰，並針對每個挑戰提供逐步實施、技術支持、身份管理、效能優化、成本控制等解決方案。

3.《命中特區》113 高考、113 司法特考類似題。

【擬答】

㈠零信任架構（Zero Trust Architecture）的挑戰與解決方案

零信任架構是一種針對網路安全的現代策略，其核心理念是「永不信任，始終驗證」（Never trust, always verify）。雖然零信任架構能夠顯著提升安全性，但企業在實施過程中可能面臨多方面的挑戰。

㈡挑戰與解決方案

1. 基礎架構複雜性

實施零信任需要重新設計網路架構，例如微分段（micro-segmentation）和應用動態訪問控制。傳統系統可能缺乏零信任所需的技術支持。解決方案：

⑴逐步實施：先選擇高風險資產或關鍵系統作為試點，逐步擴展零信任範圍。

⑵採用混合模式：保留傳統架構的同時，將零信任功能分階段整合進現有系統。

⑶使用雲原生工具：透過雲服務供應商的零信任解決方案（如微分段服務），降低實施成本與複雜性。

2. 資料與用戶身份管理

實現零信任需要對所有用戶和設備進行身份驗證和授權，而許多企業缺乏統一的身份管理系統。大量的用戶、設備和應用程序的動態行為使得管理變得困難。解決方案：

⑴採用統一身份管理：引入身份和存取管理（Identity and Access Management, IAM）平台。

⑵多因素驗證（MFA）：強化身份驗證，減少單點故障風險。

⑶實施自適應訪問控制：基於用戶行為、設備狀態和風險等上下文動態調整授權策略。

3. 效能與用戶體驗

每次訪問都需要驗證和授權，可能導致延遲，影響用戶體驗。實施細粒度的控制（如基於地理位置、設備健康狀況等）可能進一步增加延遲。解決方案：

⑴優化網路性能：引入邊緣計算（Edge Computing）或內容分發網絡（CDN）以減少延遲。

⑵實施緩存和快速驗證機制：對低風險的請求使用快取機制減少重複驗證次數。

⑶引入用戶行為分析（UBA）：針對正常行為模式設置低延遲驗證，針對異常行為加強驗證。

4.成本與資源限制

部署零信任需要額外的基礎設施、工具和專業知識，可能增加初期投資。中小型企業可能缺乏實施零信任的資源和技術支持。解決方案：

⑴採用 SaaS 工具：選擇軟體即服務（SaaS）形式的零信任解決方案以降低資本支出。

⑵引入托管安全服務（MSS）：將零信任的實施與運行外包給專業服務供應商。

⑶根據風險優先分配資源：針對最高風險的資產或應用優先部署零信任。

5.員工和管理層的接受度

員工可能對新政策（如多因素驗證或限制存取權）感到不便，導致抵觸。管理層可能對零信任的效益和實施成本心存疑慮。解決方案：

⑴員工教育與培訓：向員工解釋零信任的重要性和運作原理，提升接受度。

⑵透明的效益展示：透過數據展示零信任在減少安全事件上的成效，爭取管理層支持。

⑶提供簡化工具：讓員工使用更友好的身份驗證工具（如生物辨識）。