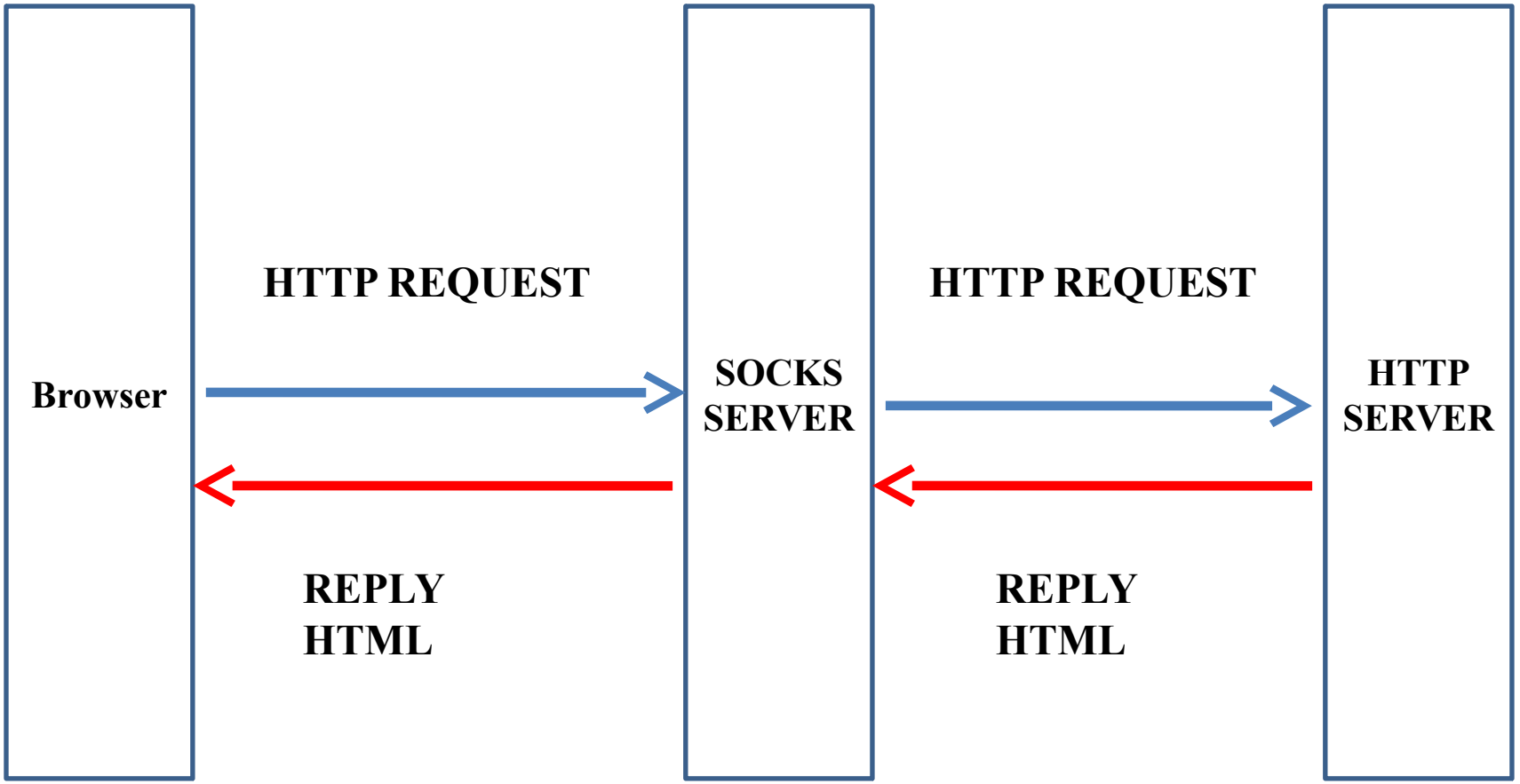
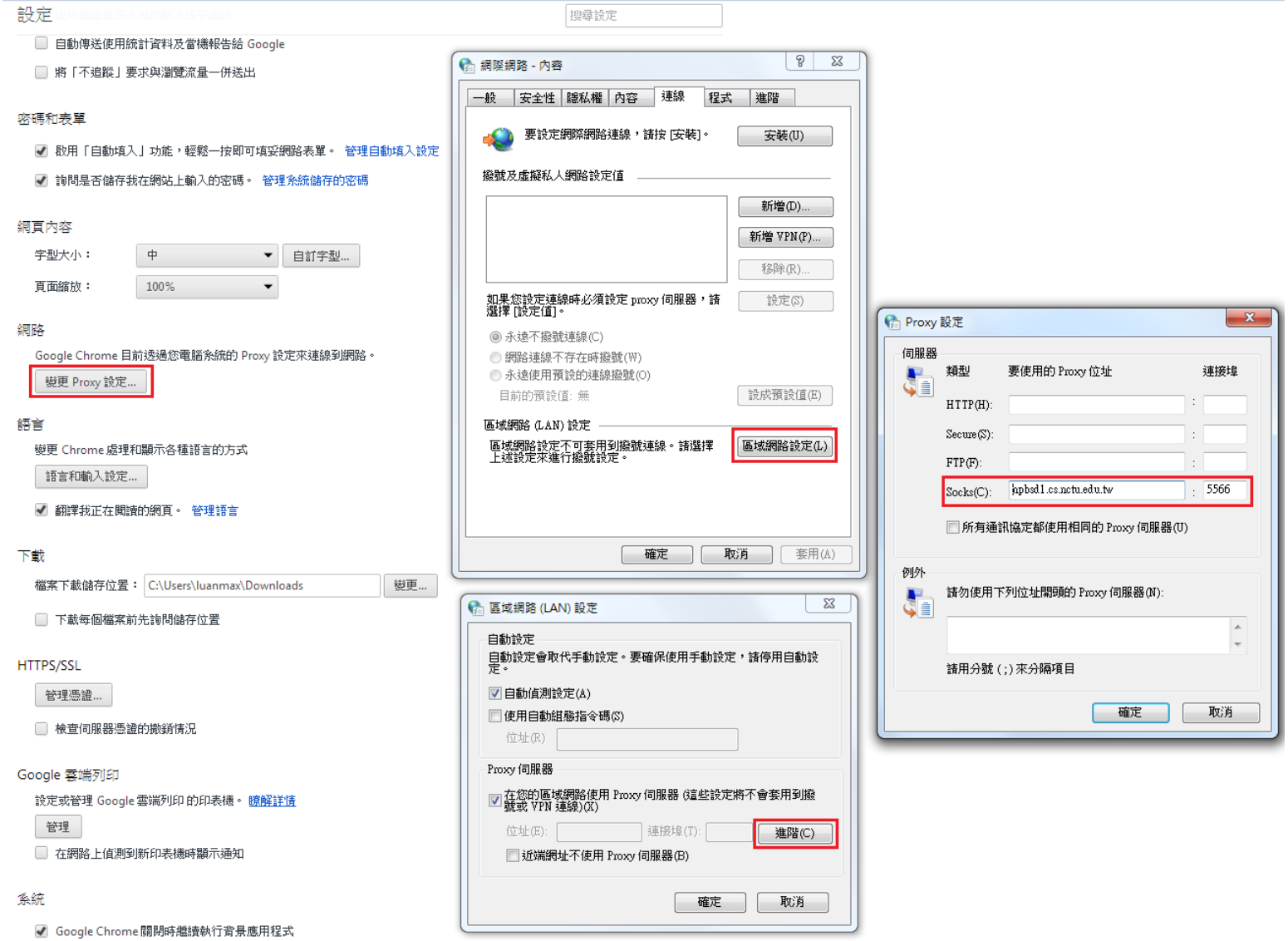
Project IV:
SOCKS4 Server

指導教授 ： 吳毅成

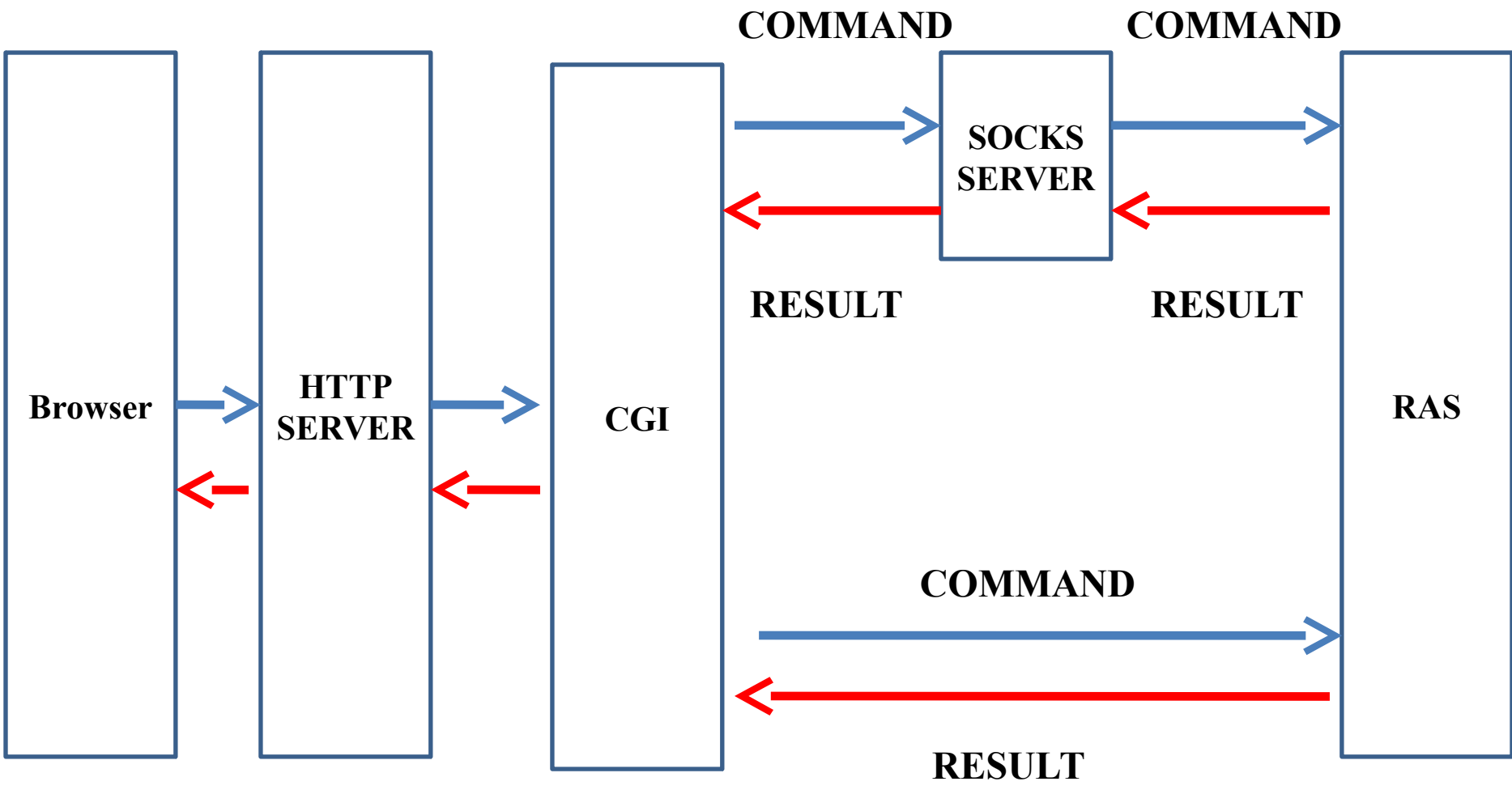
Architecture (1/3)



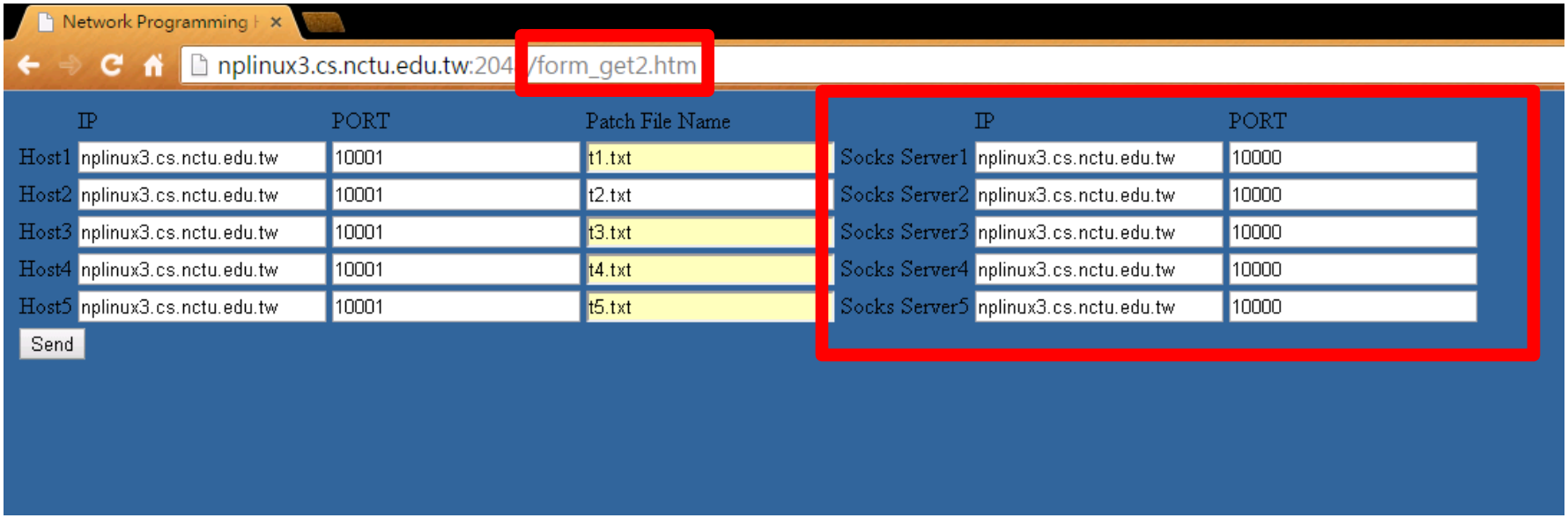
Architecture (1/3): Browser Setting



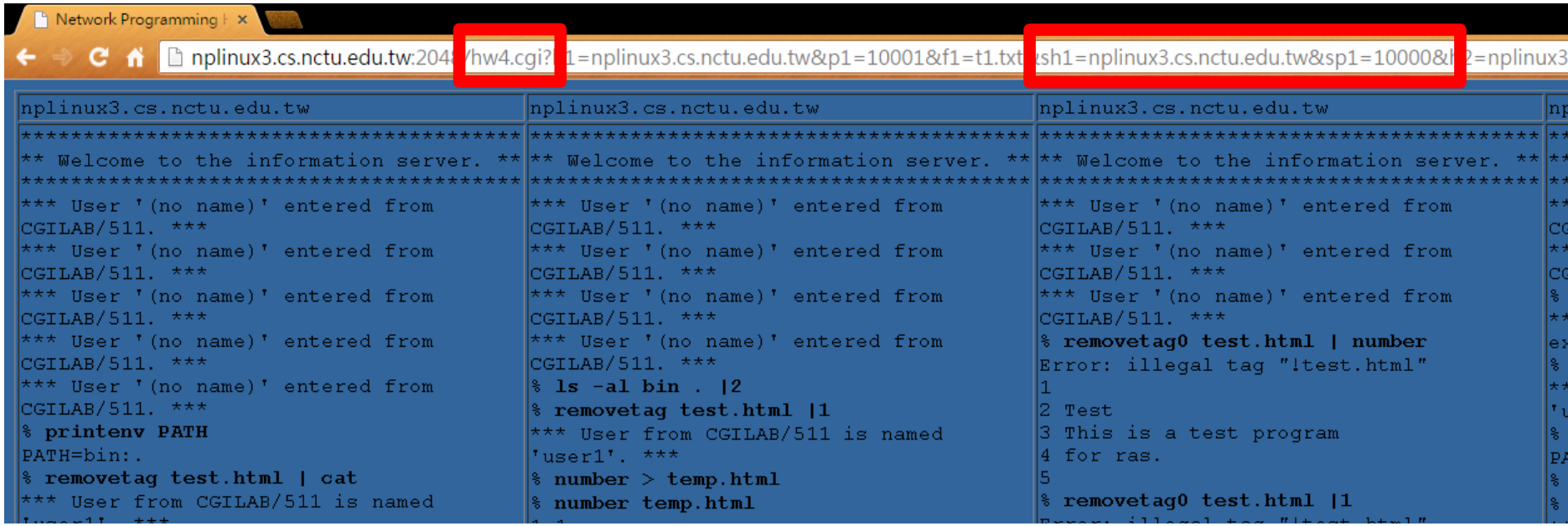
Architecture (2/3)



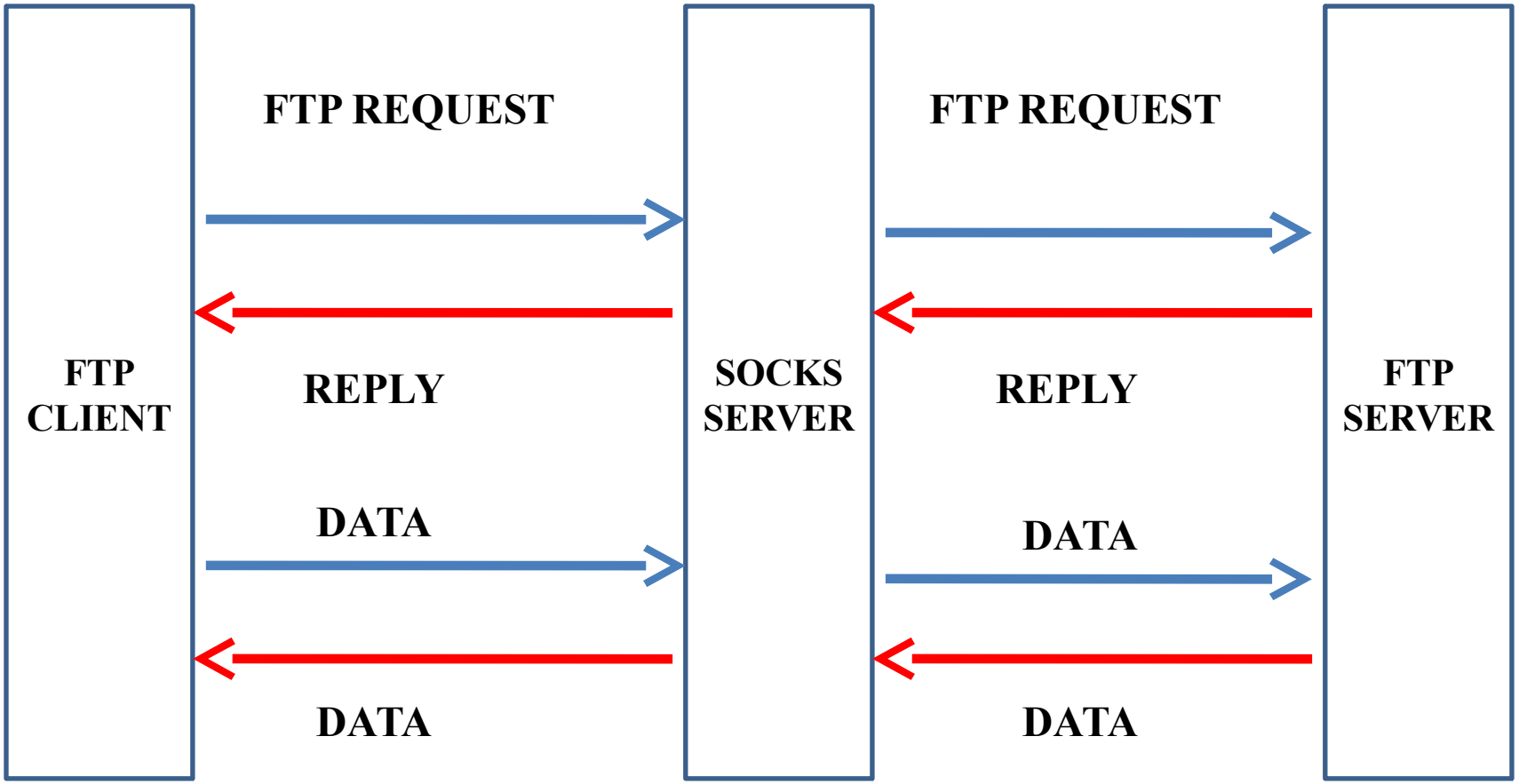
Architecture (2/3): Demo



Architecture (2/3): Demo



Architecture (3/3)

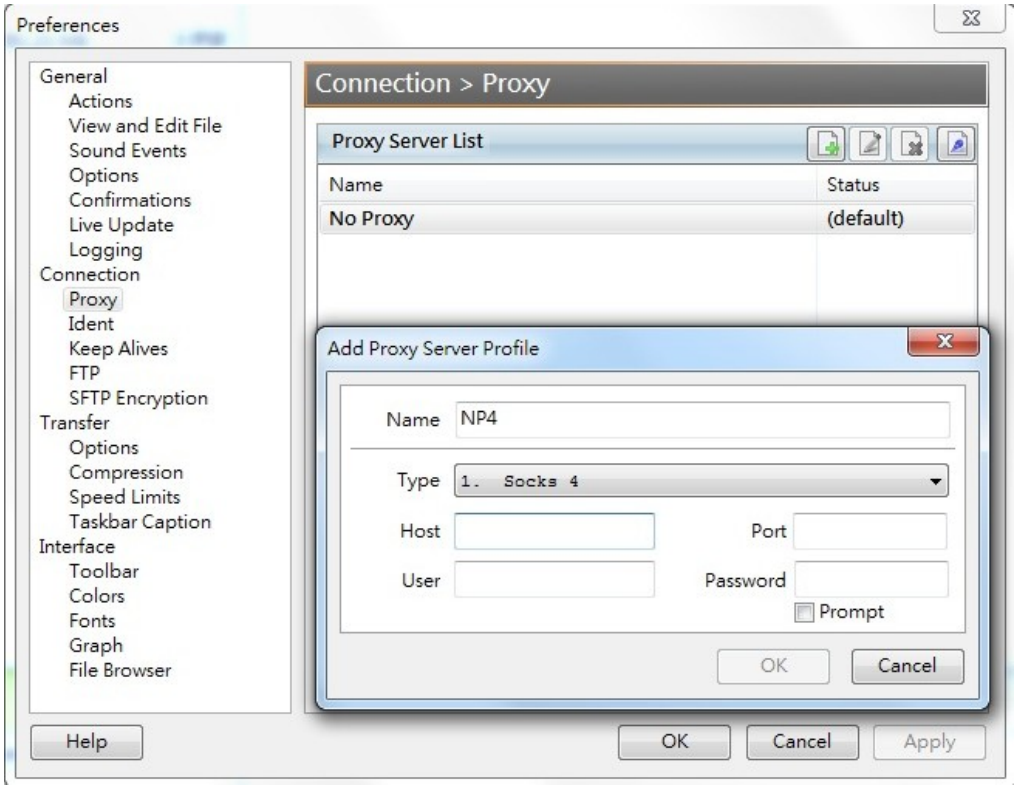


Architecture (3/3): FlashFXP

- 測試請不要用 FileZilla
- 建議使用 FlashFXP

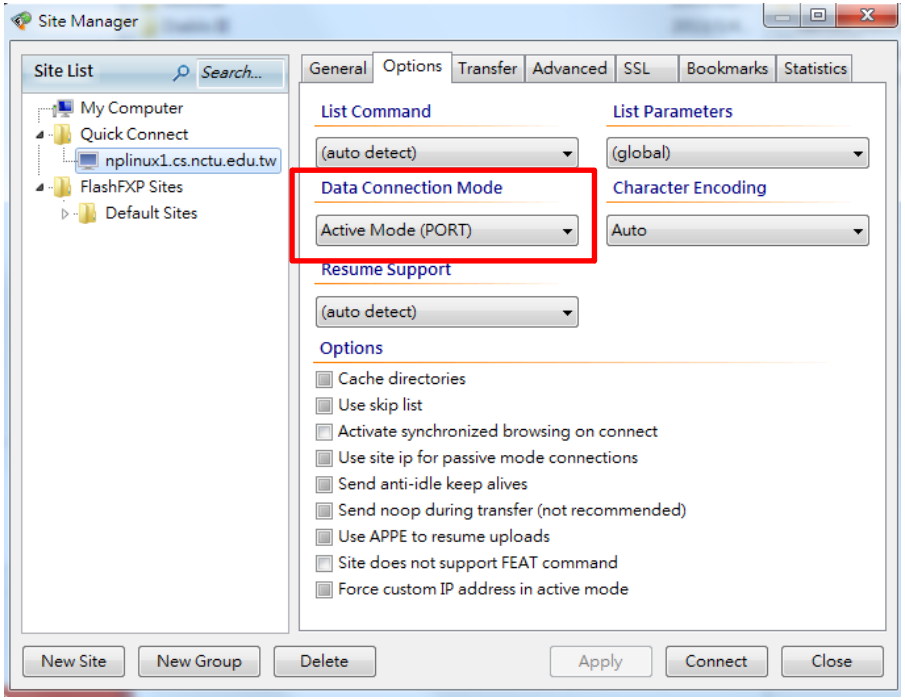
Architecture (3/3): FlashFXP

- Options Preferences Connection Proxy
  - Add entry
    - Socks4, Host/Port



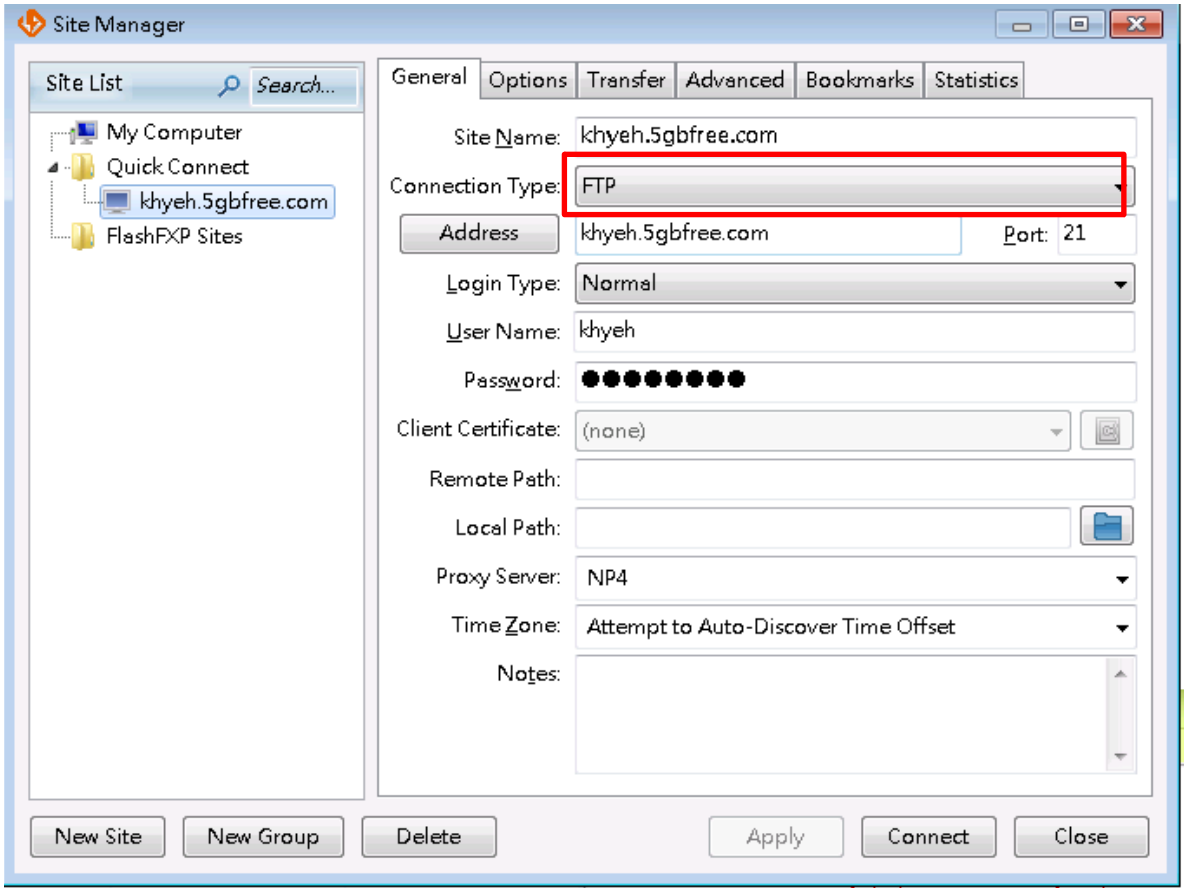
Architecture (3/3): FlashFXP

- Sites Site Manager Option Data Connection Mode
  - Change to Active Mode(PORT)



Architecture (3/3): FlashFXP

- Sites Site Manager General Apply
  - Connect



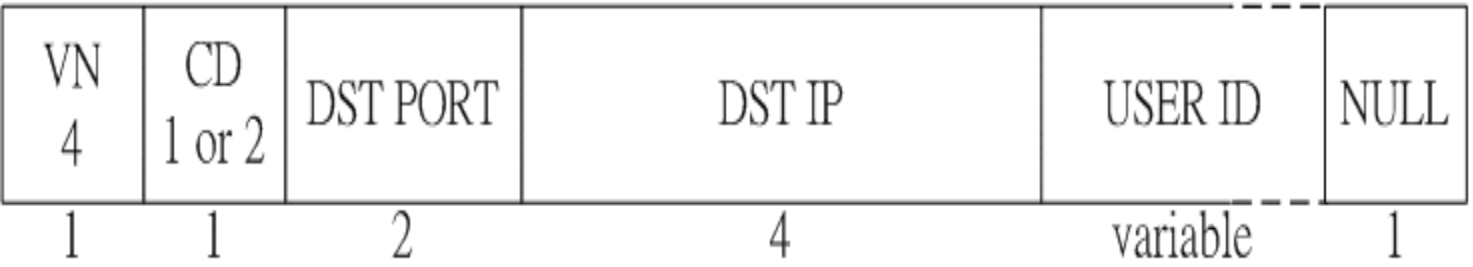
Architecture (3/3): FlashFXP

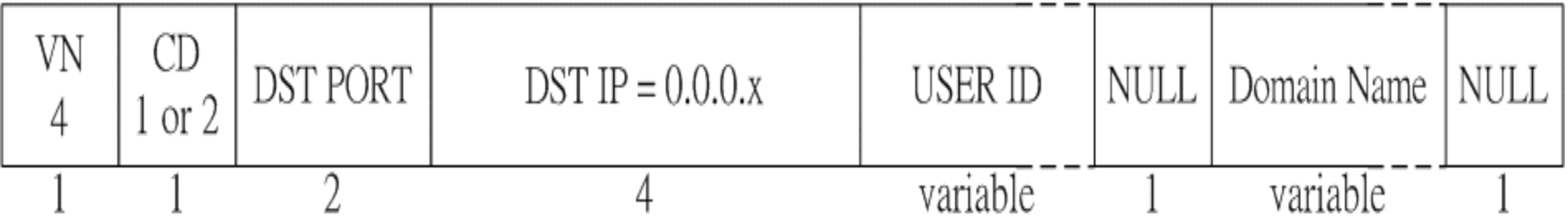
- 需要一個 FTP Server 來測試 Bind mode
  - 網路上申請可當 FTP server 上傳 / 下載的空間
    - 可以在 http://5gbfree.com/ 申請一個 5GB 的空間
      - 假設你申請的帳戶名為 : UserName
        - » Server: **ftp.UserName.5gbfree.com:21**
        - » Account: UserName
  - 或是自行在自己電腦架設 FTP server
    - 可參考 : http://goo.gl/UjrFwy

- 整體流程 :
  - 下載 & 設定好 FlashFXP
    - 透過 Socks server 使用 active mode 連線
  - 連線至 FTP server
  - 開始上傳和下載資料

Implementation

SOCKS4_REQUEST





[CD]
1: CONNECT command
2: BIND command

Implementation

Request

```
read(sock, buffer, size);
unsigned char VN = buffer[0] ;
unsigned char CD = buffer[1] ;
unsigned int DST_PORT = buffer[2] << 8 |
                                    buffer[3] ;
unsigned int DST_IP = buffer[4] << 24 |
                                  buffer[5] << 16 |
                                  buffer[6] << 8 |
                                  buffer[7] ;
char* USER_ID = buffer + 8 ;
```

Implementation

SOCKS4_REPLY

[CD]
90: request granted
91: request rejected or failed

Implementation

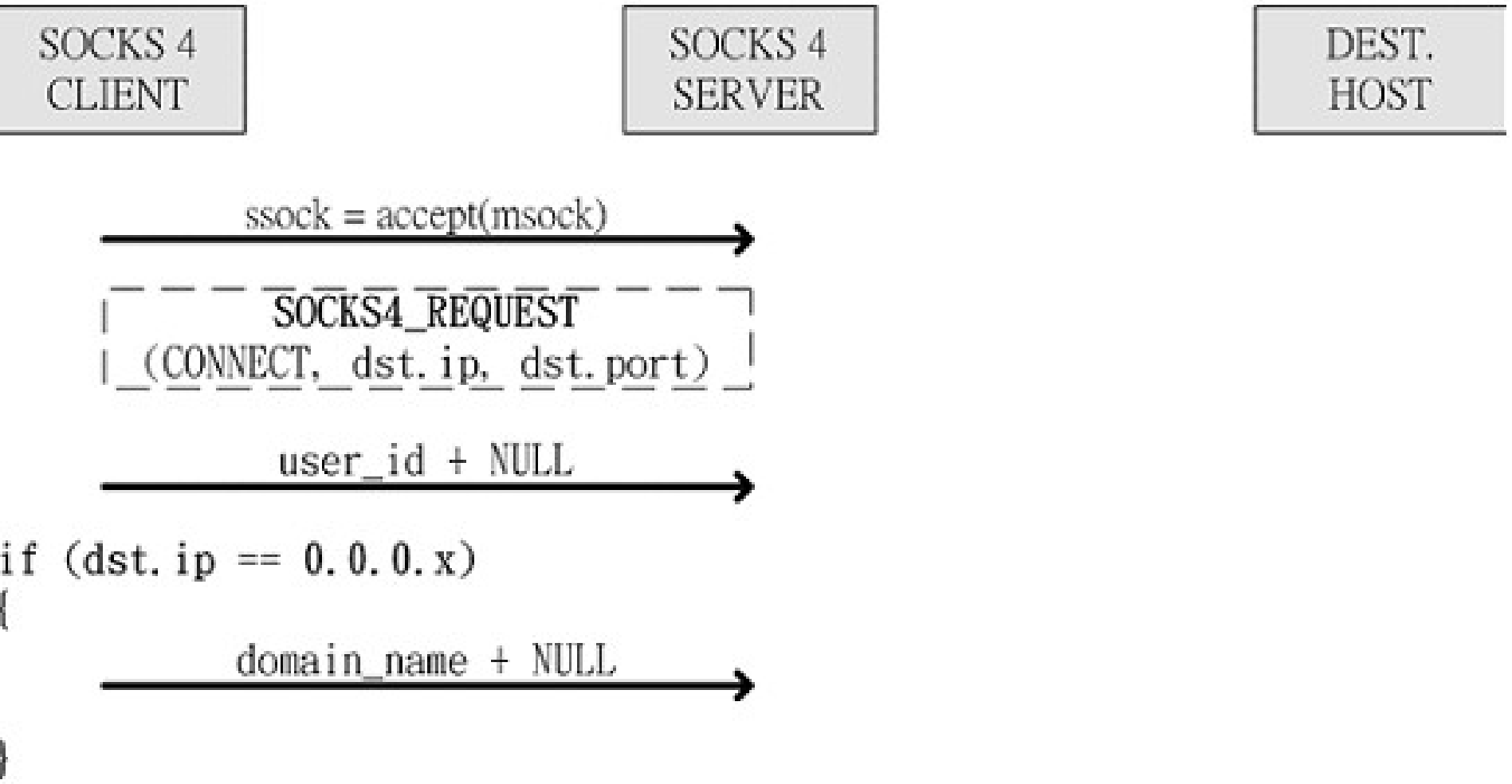
Reply

```
package[0] = 0;
package[1] = (unsigned char) CD ; // 90 or 91
package[2] = port / 256;
package[3] = port % 256;
package[4] = ip >> 24;
 // ip = ip in SOCKS4_REQUEST for connect mode
 // ip = 0 for bind mode
package[5] = (ip >> 16) & 0xFF;
package[6] = (ip >> 8)  & 0xFF;
package[7] = ip & 0xFF;
write(sock, package, 8);
```

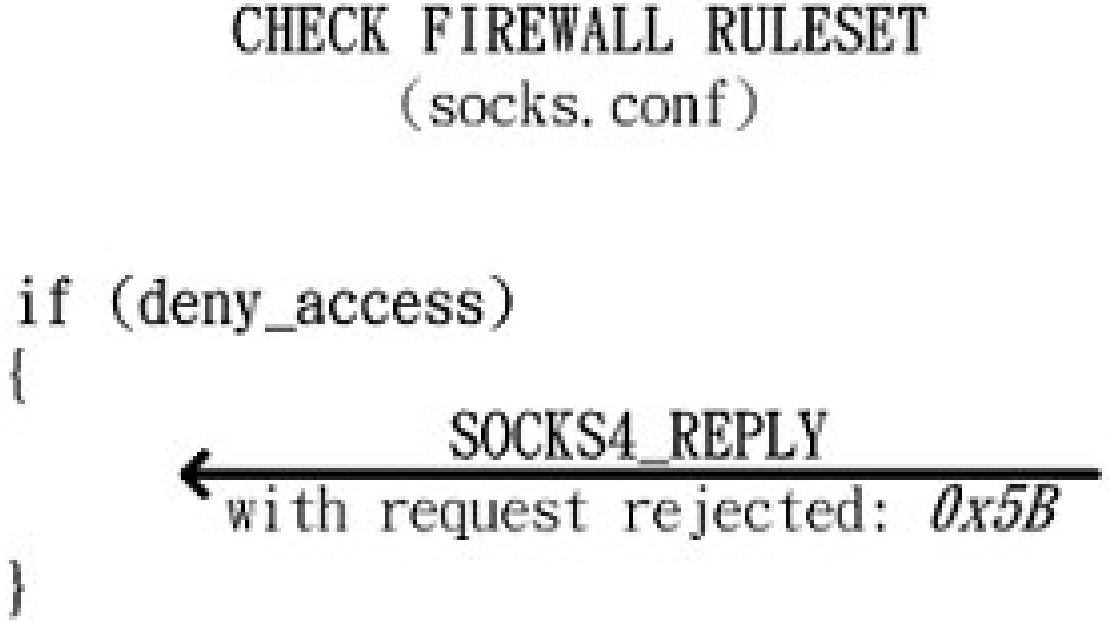
Protocol

- Connect mode
- Bind mode

Connect mode



Connect mode



Connect mode

```
else
{
```

rsock=connectTCP(dst. ip, dst. port)

→

s4_rep.vn = 0x00;
s4_rep.cd = (rsock > -1) ? 0x5A : 0x5B;
s4_rep.dst_ipv4 = s4_req.dst_ipv4;
s4_rep.dst_port = s4_req.dst_port;

← SOCKS4_REPLY
granted: *0x5A*, failed: *0x5B*

REDIRECT SOCKET DATA

WRITE to ssock ←
← READ from rsock

ssock

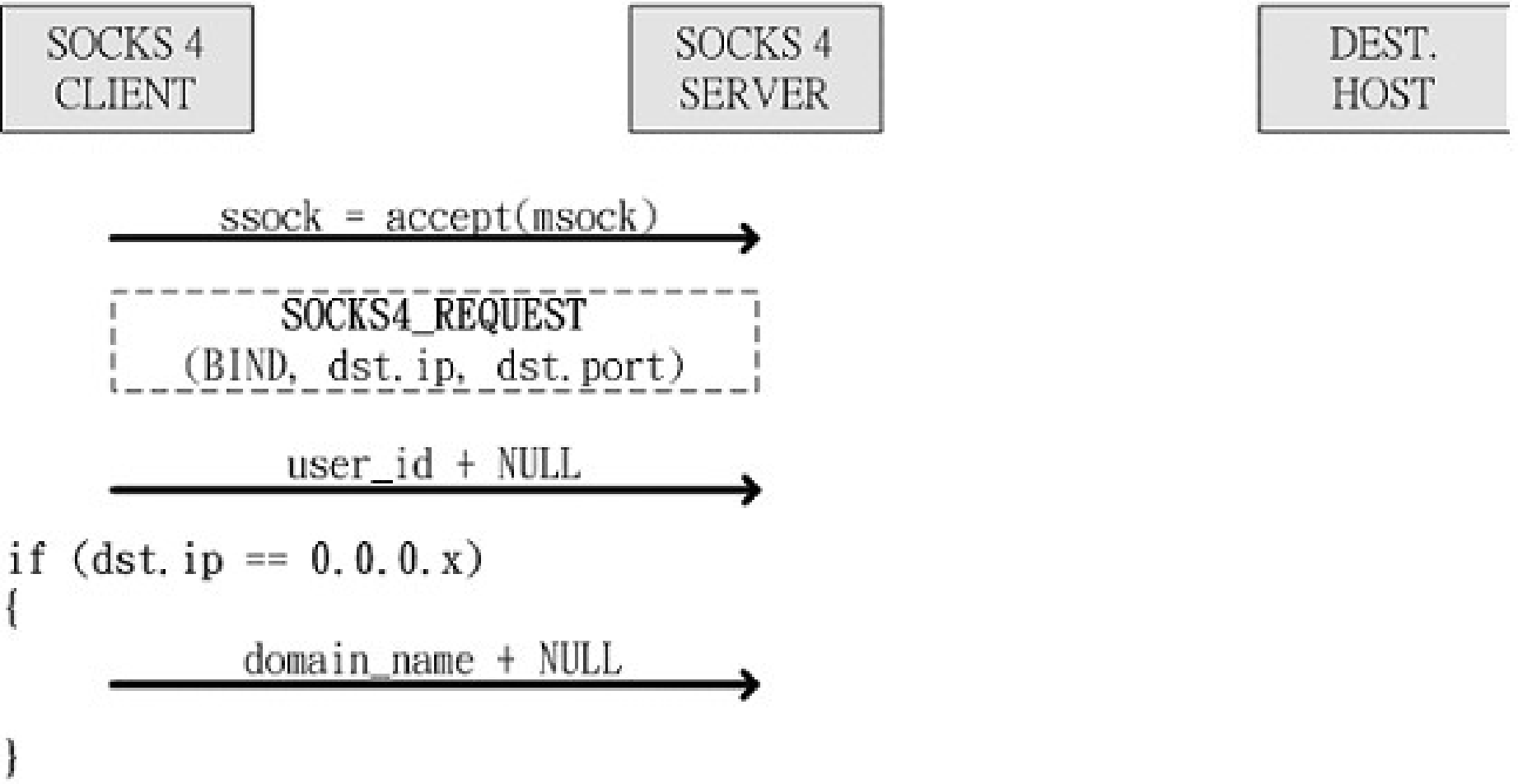
READ from ssock →

rsock

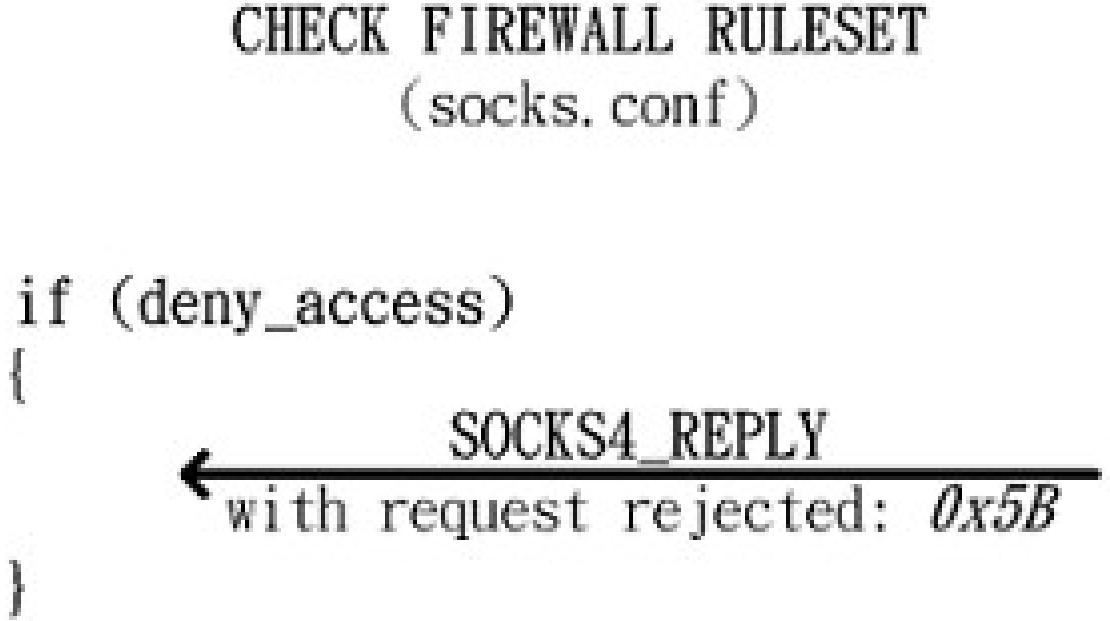
→ WRITE to rsock

```
}
```

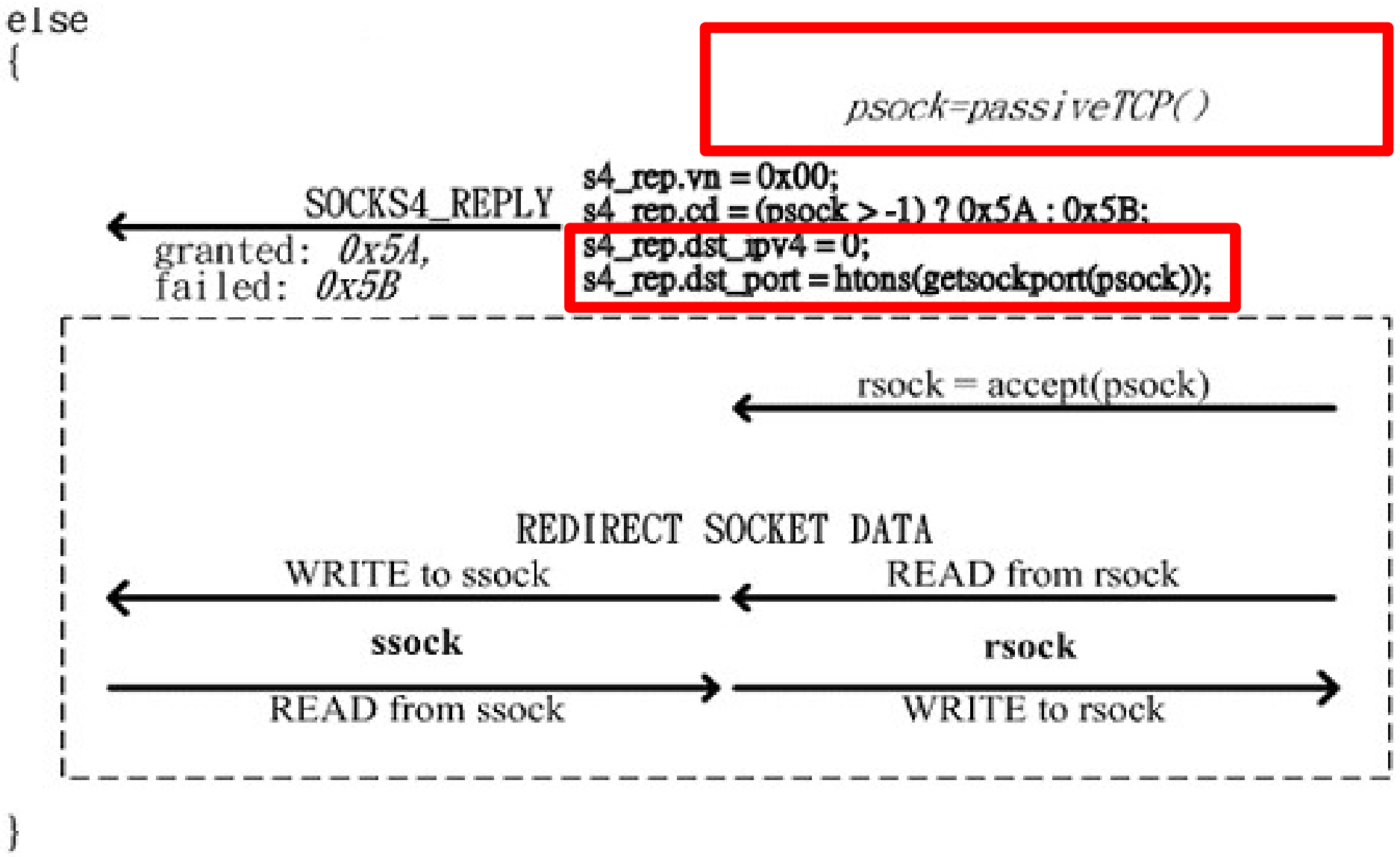
Bind mode


SOCKS 4
CLIENT
SOCKS 4
SERVER
DEST.
HOST
sock = accept(msock)
SOCKS4_REQUEST
(BIND, dst.ip, dst.port)
user_id + NULL
if (dst.ip == 0.0.0.x)
{
domain_name + NULL
}

Bind mode


CHECK FIREWALL RULESET
(socks.conf)
if (deny_access)
{
SOCKS4_REPLY
with request rejected: 0x5B
}

Bind mode


else
{
psock=passiveTCP()
SOCKS4_REPLY
granted: 0x5A,
failed: 0x5B
s4_rep.vn = 0x00;
s4_rep.cd = (psock > -1) ? 0x5A : 0x5B;
s4_rep.dst_ipv4 = 0;
s4_rep.dst_port = htons(getsockport(psock));
rsock = accept(psock)
REDIRECT SOCKET DATA
WRITE to ssock
READ from rsock
ssock
rsock
READ from ssock
WRITE to rsock
}

END