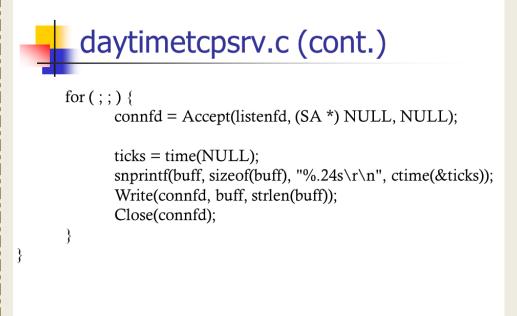
# Socket

Final Homework

110816032 謝妤婕 OuO

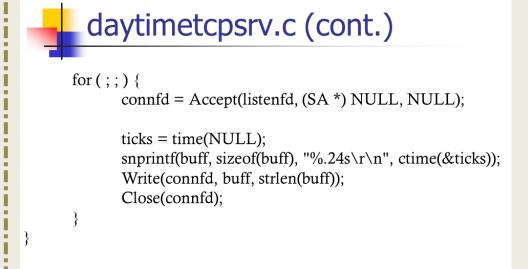


#### 一般格式化

```
sprint(char* str, const char* format, ...)
int t = 10;
char STR[20];
sprintf(STR, "%d*%d=%d", t, t, t*t);
■ 可能發生 buffer overflow

安全的格式化
snprintf(char* str, size_t size, const char* restrict format, ...)
snprintf()比 sprinf()多了一個參數 size
限制最大的寫入資料
```

14



### snprintf()會把過長的部分切掉

利用 asprintf()

```
char* cstr;
int c = asprintf( &cstr, "%d*%d=%d", t, t, t*t);
```

#### 但 asprintf()不是標準函式

■用 snprintf()改寫

14

```
char* cstr;
int c = snprintf( NULL, 0, "%d*%d=%d", t, t, t*t);
cstr = new char[c+1];
snprintf( cstr, c+1, "%d*%d=%d", t, t, t*t);
```



# daytimetcpsrv.c (cont.)

```
for (;;) {
     connfd = Accept(listenfd, (SA *) NULL, NULL);

     ticks = time(NULL);
     snprintf(buff, sizeof(buff), "%.24s\r\n", ctime(&ticks));
     Write(connfd, buff, strlen(buff));
     Close(connfd);
}
```

for(;;)

就是 while(1)不然 while(true)也可以

所以不要讓分號空空的 O III O !!!!!

真的是吼 氣氣氣氣氣

npch2 page0005、page0006 也是啦



#### **Details of Your Local Network**

- To find out interfaces
  - netstat -ni
- To find out routing table
  - netstat -rn
- To find out details of an interface
  - ifconfig
- To discover hosts on a LAN
  - ping

CMD 網路相關指令

顯示或修改目前 Routing Table。

route

print 顯示目前 Routing Table

add 加入路由資料

delete 删除路由資料

change 更改路由資料

20

偵測主機與遠端主機間的連線是否暢通, 並顯示封包傳送往返的時間。

# ping [主機的網址或 IP 位址]

- † 持續發出測試封包直到手動 Break
- -n 設定測試封包的個數,預設為 4 個
- → 設定測試封包的長度,預設為 64-byte

顯示主機網路介面的 IP 位址、網路遮罩 (mask)、預設 Gateway 位址、主機名稱及 DNS 伺服器位址等資訊。Physical Address 是指 MAC 位址(網卡號碼)。

### ipconfig

/aⅡ 顯示網路介面詳細的設定情形

顯示主機至遠端主機間封包傳送所經之路 逕、Gateway 位址及往返時間。 顯示遠端主機的某一個用戶資訊。

finger [使用者名稱]@ [主機網址或 IP 位址]

# tracert [主機的網址或 IP 位址]

- -d 不須將 IP 位址解析為網址
- -h 設定封包傳送所經的最大 hop 數
- -W 設定封包傳送逾時秒數,超過則重送

連到網路上遠端主機並模擬其終端機。

telnet [主機的網址或 IP 位址][port(預設 23)]

顯示 IP 位址與 MAC 位址的對應關係,用來檢查 IP 位址相衝等問題。

arp

用於二台主機間複製檔案,遠端主機須提供 rcp 服務。

### rcp [主機名稱:來源檔] [主機名稱:目的檔]

- 文字模式傳輸方式
- -b 二進位模式傳輸方式
- -r 連同子目錄一起複製

查詢一台主機的 IP 位址及其對應的網域名稱。(有二種工作模式,若有指定欲查詢的主機網址或 IP 位址則立即顯示結果;若沒加任何參數則進入指令模式,可鍵入主機網址、IP 位址或其它命令進行查詢。)

nslookup [主機的網址或 IP 位址]

顯示目前 TCP/IP、UPD、ICMP 的狀態,建立連線之雙方及相關流量統計資料。

顯示 NetBIOS 名稱列表、節點名稱對應的 IP 位址及 NetBIOS 進行通訊流量統計。

#### netstat

- -a 顯示所有 TCP、UPD 當時連線狀態
- -e 顯示 Ethernet 介面之流量統計資料
- -p 顯示指定協定的連線及統計資料
- -r 顯示路由表(Routing Table)
- -S 顯示 TCP/IP、UPD、ICMP 的流量統計資料

#### nbtstat

- -a [遠端主機名稱]顯示遠端主機的名稱
- -A [遠端主機 IP 位址]顯示遠端主機 NetBIOS 名稱列表
- -n 顯示主機本身 NetBIOS 名稱列表
- -r 顯示主機執行過 NetBIOS 名稱解析次數的統計
- -s 顯示主機 NetBIOS (Connection Table)



# Types of Networking APIs

- TCP socket
- UDP socket
- raw socket over IP (bypass TCP/UDP)
- datalink (bypass IP)
  - BPF (BSD Packet Filter)
  - DLPI (SVR4 Data Link Provider Interface)

#### 1. Stream sockets(connection-oriented)

資料在 stream sockets 間是利用 TCP 封包來傳送,因此接收端可以收到順序無誤、無重覆、正確的資料。此外 TCP 傳送時是採資料流的方式,因在傳送時會所有資料會視情況被分割在數個 TCP 封包中。



# Types of Networking APIs

- TCP socket
- UDP socket
- raw socket over IP (bypass TCP/UDP)
- datalink (bypass IP)
  - BPF (BSD Packet Filter)
  - DLPI (SVR4 Data Link Provider Interface)

#### 2. Datagram sockets(connectionless)

資料在 datagram sockets 間是利用 UDP 傳送,因此接收端可能會收到次序錯誤的資料,且其中部分資料亦可能會遺失。

# npch3 page0003

#### Comparison of Various Socket **Address Structures** IPv6 UNIX Datalink socketaddr in{} socketaddr in6{} socketaddr un{} socketaddr dl{} length AF\_LOCAL length AF\_INET6 AF LINK AF INET 1ength 1ength interface index 16-bit port# 16-bit port# 32-bit 32-bit name len type IP address flow label addr len sel len interface name (unused) and link-layer address 128-bit fixed length IPv6 address pathname (16 bytes) (up to 104 bytes) variable length fixed length (24 bytes) variable length

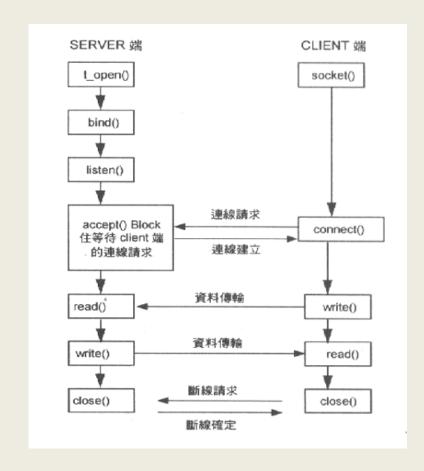
#### TCP Header



# npch2 page0006/ npch4 page0002



- socket()
- connect()
- bind()
- listen()
- accept()
- fork() and exec functions
- Concurrent servers
- close()
- getsockname() and getpeername()



npch2 page0006/ npch4 page0002 (簡易中文版)

#### socket()

根據指定的 socket type 建立一個新的 socket,為 socket 分配系統資源,並回傳一個 file descriptor。

#### bind()

一般用於 server,用來將 socket file descriptor 和 sockaddr structure 相關聯, sockaddr 結構中會指出這個 socket(sockfd)所要監聽的 address, port number 等。

#### listen()

用於 server,使一個 socket(sockfd)進入監聽狀態。

# npch2 page0006/ npch4 page0002 (簡易中文版)

#### connect()

用於 client 端,透過 sockfd 和 sockaddr structure 向指定的 server 進行通訊,如果是TCP,則 connect function 會建立起連線。

#### accept()

用於 server 端,接受一個從 remote client 來的 TCP 連線請求,和 remote client 建立 TCP 連線並將建立的 socket 關連到 sockfd 與 sockaddr。

#### close()

呼叫系統關閉分配的 sockfd, TCP 則連線會中斷。

# 其他補充

- 1. WebSocket
- 2. Agent Builder 使用 Socket 監視資料
- 3. OSI Model

# 1. WebSocket

一種網路傳輸協定,可在單個 TCP 連接上進行全雙工通訊,位於 OSI 模型的應用層。

WebSocket 協定在 2011 年由 IETF 標準化為 RFC 6455,後由 RFC 7936 補充規範。

Web IDL 中的 WebSocket API 由 W3C 標準化。

WebSocket 協定支援 Web 瀏覽器與 Web 伺服器之間的互動,便於實現客戶端與伺服器的即時資料傳輸。伺服器可以通過標準化的方式來實現,而無需客戶端首先請求內容,並允許訊息在保持連接打開的同時來回傳遞。通訊通過 80port 或 443port 完成,這在防火牆阻止非 Web 網路連接的環境下是有益的。

from <a href="https://zh.wikipedia.org/wiki/WebSocket">https://zh.wikipedia.org/wiki/WebSocket</a>

# 2. Agent Builder 使用 Socket 監視資料

使用 Socket 資料來源會將資料從與代理程式在相同系統上執行的外部應用程式提供至該代理程式。 外部應用程式可以在需要的任何時間,將資料傳送至代理程式。代理程式不會啟動或停止正在將資料傳送至 Socket 的應用程式(使用者控制)。

Socket 資料來源有以下限制:

依預設,只能存在與本端主機 (127.0.0.1) 的連線。

如需配置代理程式以接受遠端主機連線的相關資訊(必須在具有遠端系統網路介面連線的系統上)。

from <a href="https://www.ibm.com/docs/zh-tw/capm?topic=sources-monitoring-data-by-using-socket">https://www.ibm.com/docs/zh-tw/capm?topic=sources-monitoring-data-by-using-socket</a>

# 3. OSI Model

- 7 Application Layer: HTTP > FTP > POP3
- 6 Presentation Layer
- 5 Session Layer
- 4 Transport Layer: TCP > UDP
- 3 Network Layer: IP、ICMP、ARP、RARP 路由器(Router)
- 2 Data Link Layer: MAC、LLC 橋接器(Bridge)、交換器(Switch)
- 1 Pysical Layer: 實體連線、基頻編碼、寬頻調變、多工
- 中繼器(Repeater)、集線器(Hub)

# Reference

- ■ip 相關指令
  - > https://swangs.pixnet.net/blog/post/19501057
- Socket 流程圖
  - > https://www.geeksforgeeks.org/socket-programming-cc/
  - > http://wmnlab.ee.ntu.edu.tw/nmlab/exp1\_socket.html
  - > https://www.kshuang.xyz/doku.php/programming:c:socket