專題報告

—在 windows 上實作 clientAgent—

組員：魏廷芸 徐曼妮 張智諺 曾思維

一、 研究目的

在原先的實驗室環境中，已有 linux 系統的 client agent，現在想透過 pydivert 將原本 linux 程式由 iptable 負責處理的部分改寫，使其可在 windows 系統上執行。

二、 研究方向

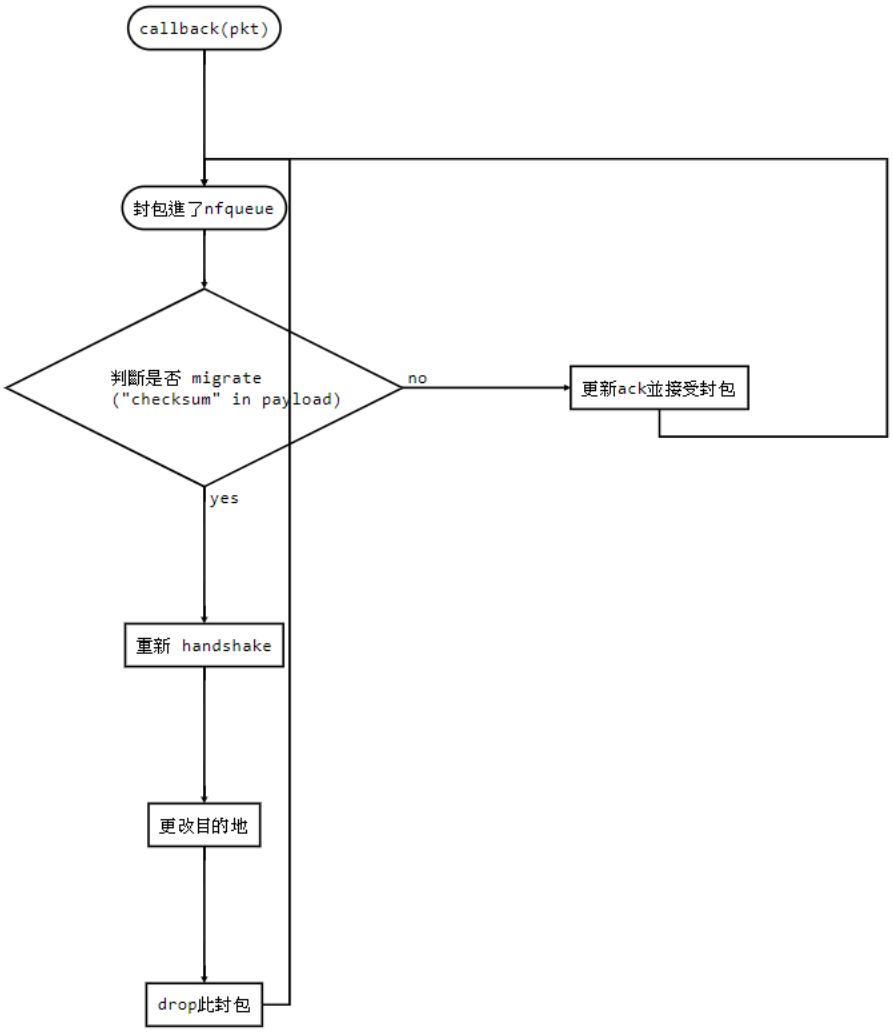
原程式中，使用 nfqueue 負責處理封包，iptable 負責 forwarding。Pydivert 將兩者結合一起使用，因此只要將程式碼中 iptable 跟 shell 的部分對應 pydivert 的語法修改，其他部分保持原本 code 即可。

三、 系統架構

使用者連上已知 ip 後，透過 clientAgent 將 ip 修改為實際的伺服器 ip，達成伺服器遭 DDoS 攻擊轉移至另一主機後，使用者能在不需重新連線的情況下直接轉移到新主機上繼續使用服務。

四、 程式架構

- Linux version：



Linux version 文字版本(紅字為需要修改為 windows 版本的部分)：

Arg_override()	設定真正的目標 port(ssh/http/https/openvpn)

init()	初始化設定: 1. 設定 logger 2. 設定 destination a、 取得 ip b、 取得 port c、 取得 routing table 3. 設定網卡**(略過此步驟不影響程式執行) a、 擷取網卡內容 4. 設定 source

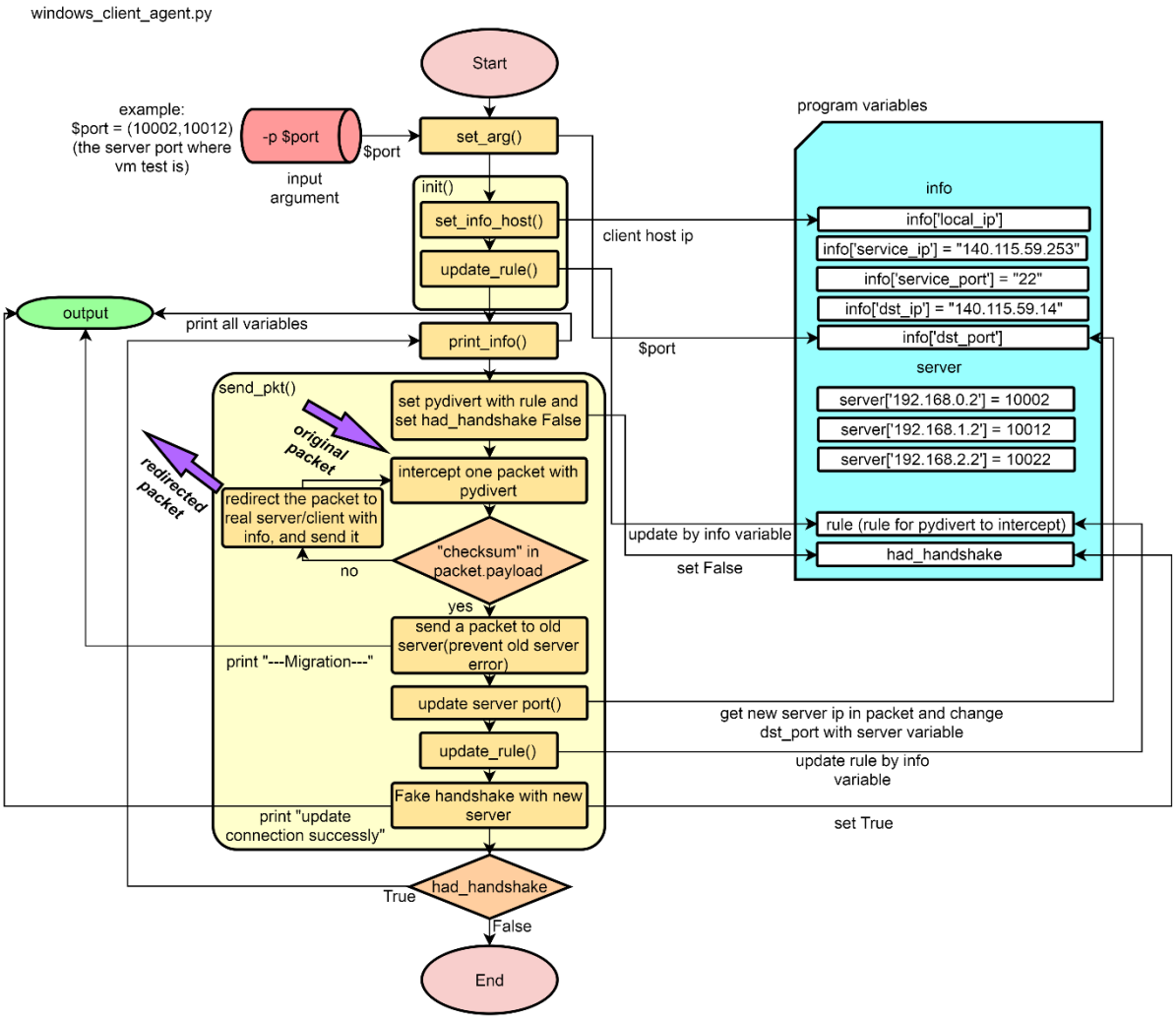
Setufw()	設定 iptable: 1. 列舉所有 host 的 port 2. 在 raw 的 prerouting 插入 rule(將 server 封包放入 nfqueue)

建立 nfqueue

Callback()	處理封包： 1. 分辨封包來源，把封包轉成 scapy 物件 2. 若 packet[RAW]有特定 checksum-> live migration 3. 創 ip/tcp ack 並送出

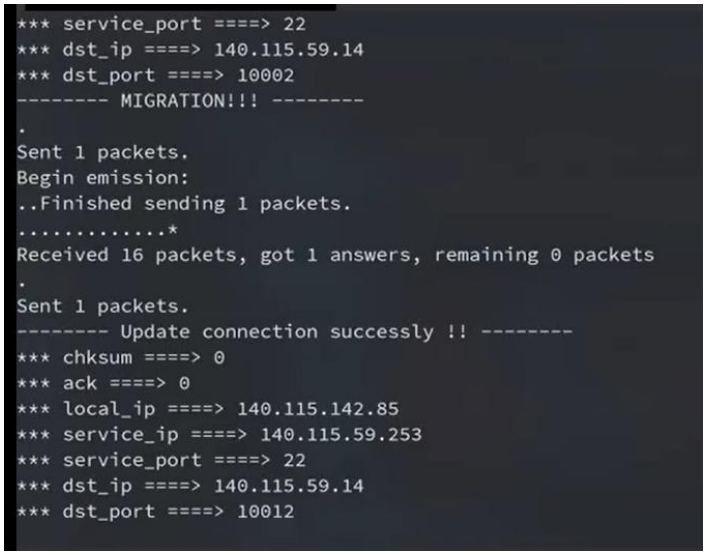
chdst()	改變 destination：重設 IP Table 規則 1. Rule num** 2. 刪除連線紀錄 3. 重新設 IP Table，dest 改為新的目標(新 ip，新 port)
handshake()	偷偷 handshake 1. 改變 iptable: 不傳 RST、不收 SYN/ACK 2. 建立 scapy 物件，傳 ip/tcp/syn/r 給新 host 3. 若收到回傳，用 sendip 回傳 ACK 4. 刪除被更改的規則

- Windows version：

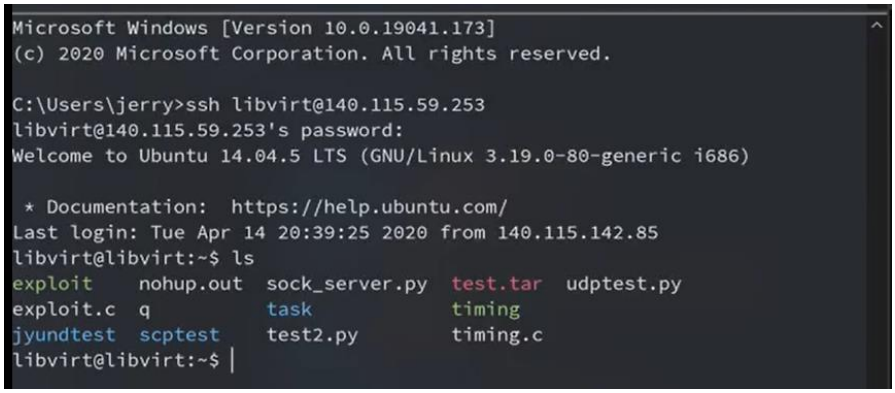


五、 測試結果

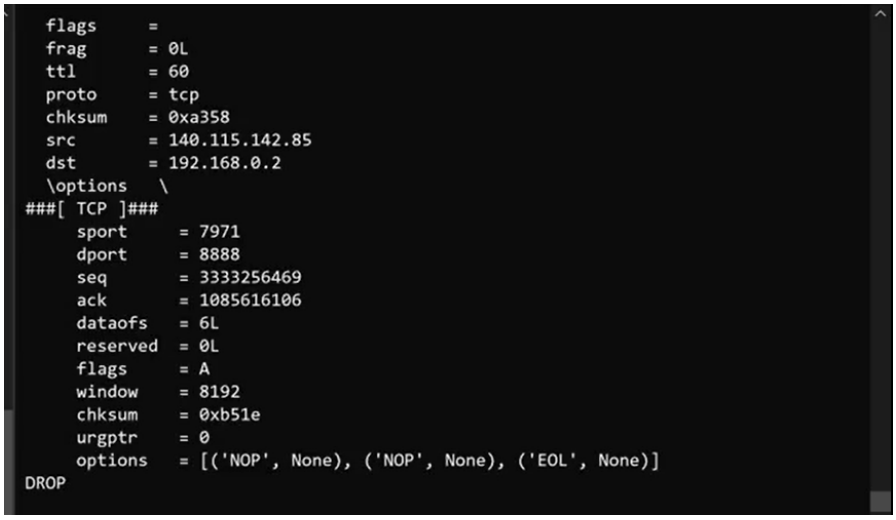
Windows client agent 顯示 migration 完成：



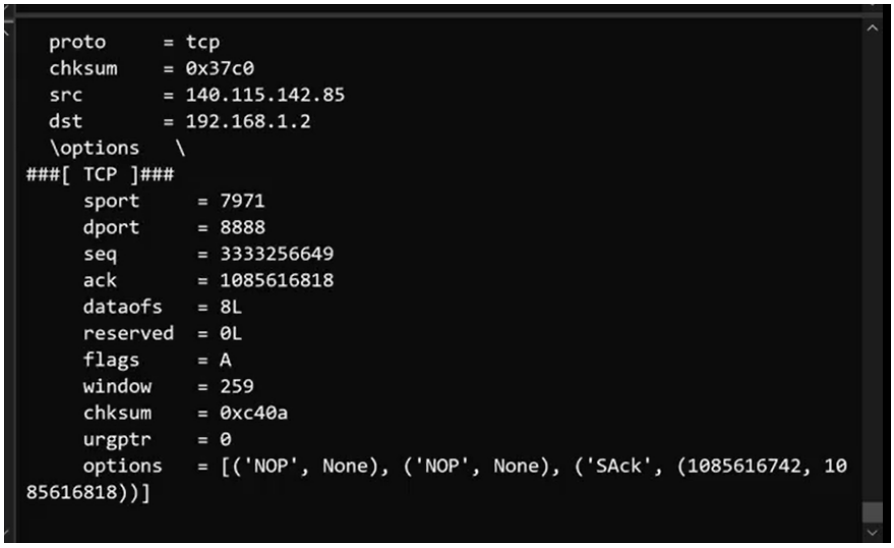
此時，在 libvirt 內下 ls 指令，成功獲得回應：



原本的 host 停止接收到封包：



新的 host 開始接收封包：



根據圖片結果顯示，在經過 live migration 後，windows client 仍與 server 保持連線。