殭屍網路簡介與偵測

國立中山大學 資管系
陳嘉玫

Botnet是什麼?

- 'bot'一詞取自於'robot'.
- 'bot' 指的是不需要使用者操作便能夠自動的執行任務的程式
- Blogbots, e.g., wikipedia, xanga Note: http://en.wikipedia.org/wiki/Wikipedia:Bots
- 其它例子: xdcc, fserve bots for IRC
- 惡意 bots :
  - *偷竊資訊的惡意軟體*
  - 特色: 複制網路與檔案存取(file access)的process，並悄悄的散播

Botnet是什麼?

- Botnet 俗稱殭屍網路(Zombie Network)
- 透過 email、通訊軟體或利用電腦系統漏洞等方式，將 Bot(殭屍程式) 隱藏於軟體中，植入受害主機
- Bot 會隱藏於受感染主機上，會主動對外連接至指定的 Server，接收並執行 Botmaster(駭客) 定義的命令，常見於將Bot 主機(受感染的主機) 指定連線至 IRC Server(聊天室) 上。

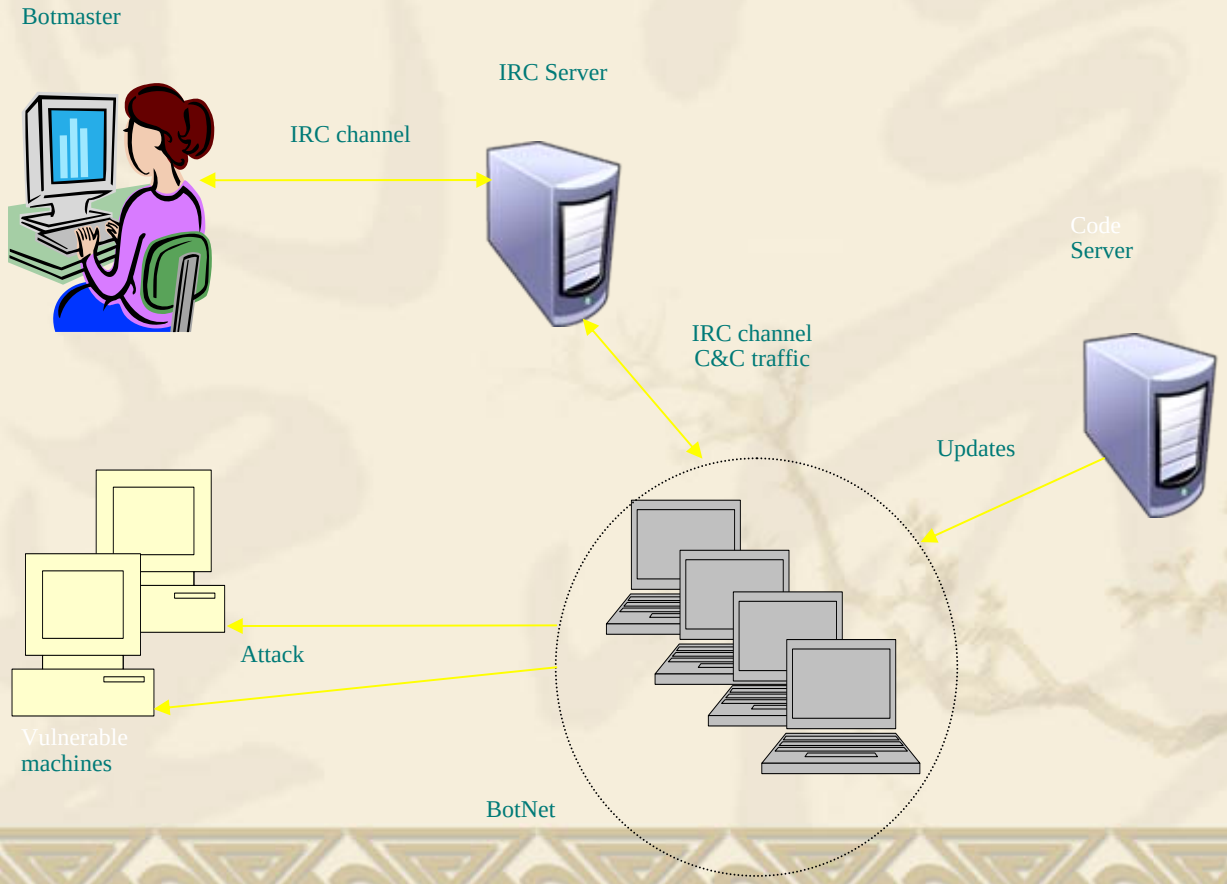
威脅嚴重性

- 2008年4月30日中國駭客利用 Botnet 發動 DDoS 攻擊，癱瘓巴哈姆特網站首頁與遊戲網站「遊戲基地」

- 驚人的成長速度
  - 半年的時間，受感染的主機就增加了 3 萬台
  - 各種不同的惡意軟體幾乎都用殭屍網路形式控制

Botnets

- 由受感染的機器/bot所組成的一種網路 - 一旦機器受到感染，從此便受控於駭客(botmaster)



歷史

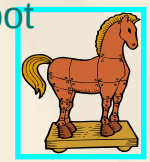
- 最初，並不存在惡意的bots
  - ex: google bot, game bot etc.
- 隨後，心懷不軌的人想到利用bots進行：
  - 散佈垃圾郵件與釣魚信件
  - 取得其它電腦的控制權
  - 針對servers進行攻擊(如DDOS)
- 許多惡意bots因而產生
  - SDBot/Agobot/Phatbot etc.
- Botnets隨後開始出現

TimeLine

GM (by Greg, Operator)
recognized as first IRC bot.
Entertained clients with games

GT bots combined mIRC client, hacking scripts & tools (port - scanning, DDos)

W32/Agobot bot family added modular design and significant functionality



W32/Mytob hybrid bot major e-mail outbreak

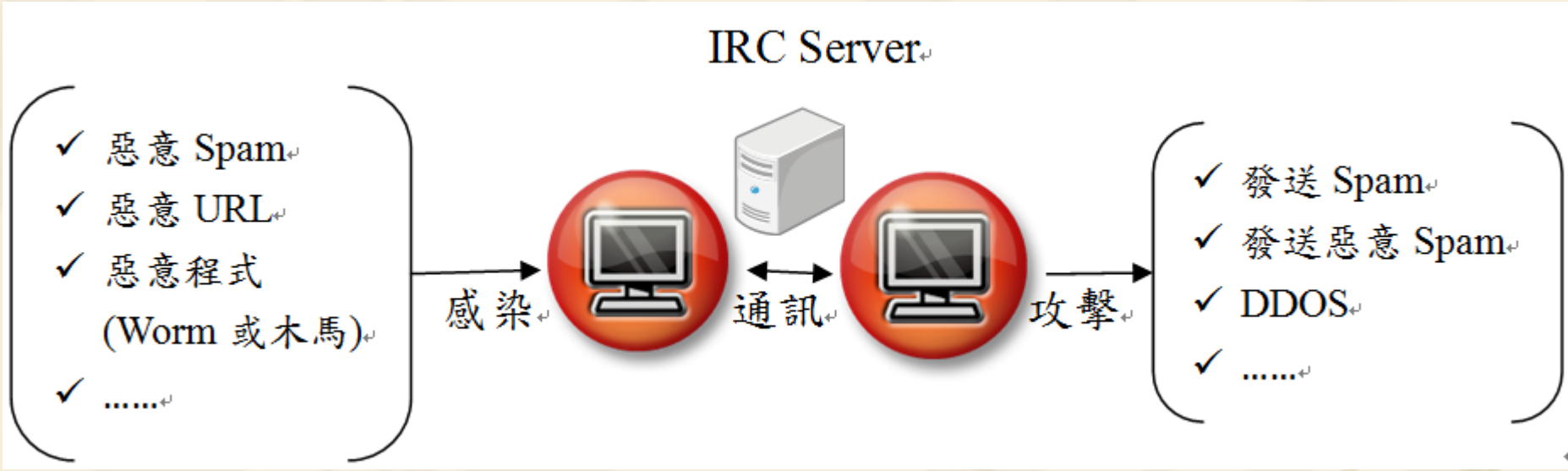
1989 1999 2000 2001 2002 2003 2004 2005 2006 Present

W32/PrettyPark 1st worm to use IRC as C&C.
DDoS capable

W32/Sdbot First family of bots developed as a single binary Russian named sd

W32/Spybot family emerged

Botnet發展三階段



Botnet 生命週期

Botmaster

1. Botmaster infects victim with bot (worm, social engineering, etc)

Victim

C&C Server

Botnet 生命週期

Botmaster

Victim

C&C Server

2. Bot connects to C&C server. This could be done using HTTP, IRC or any other protocol.

Botnet 生命週期

Botmaster

3. Botmaster sends commands through C&C server to bot

C&C Server

Victim

Botnet 生命週期

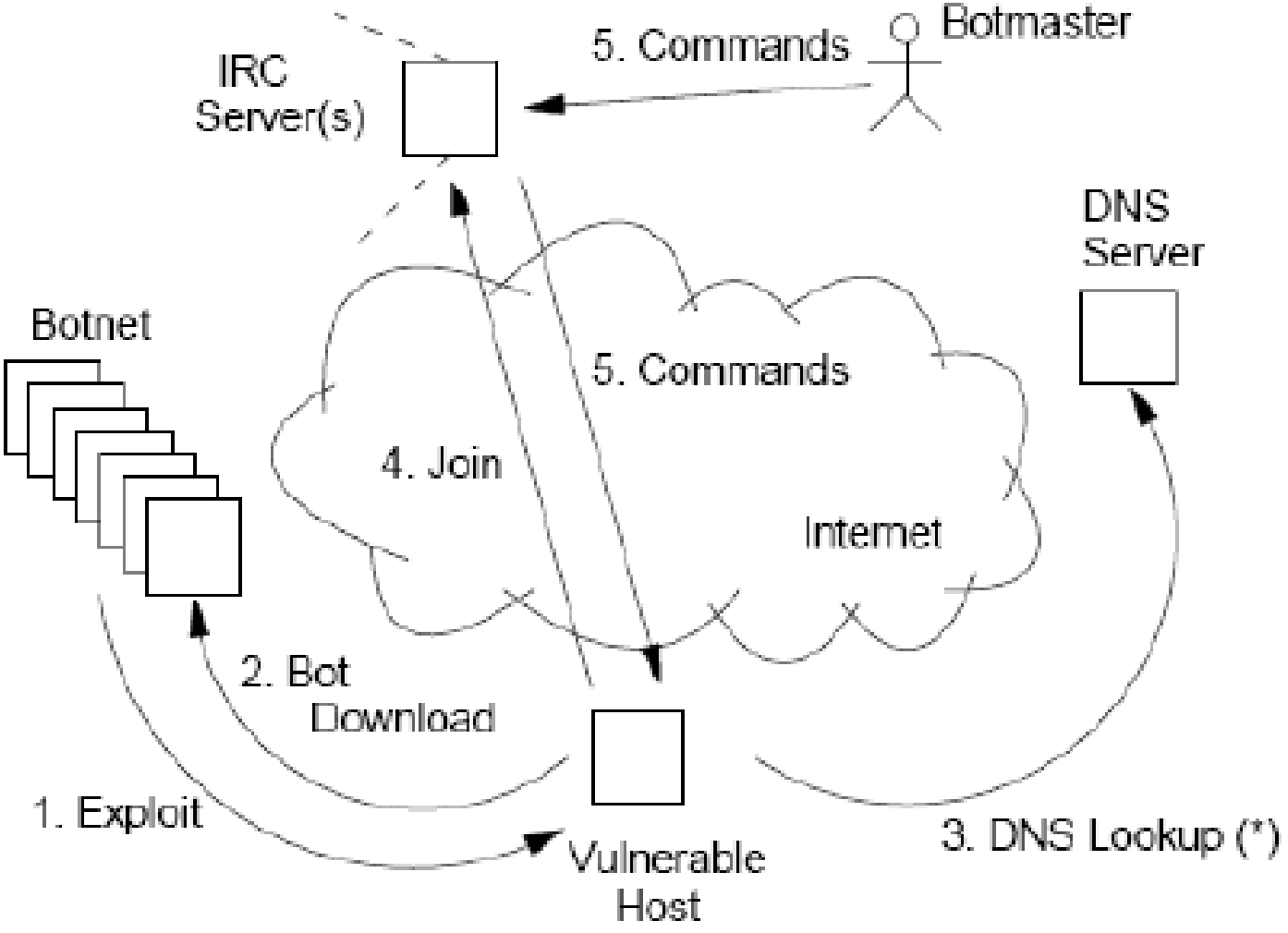
Botmaster

Victim

4. Repeat. Soon the botmaster has an army of bots to control from a single point

C&C Server

形成bot的過程


Botmaster
5. Commands
IRC
Server(s)
DNS
Server
Botnet
5. Commands
4. Join
Internet
2. Bot
Download
1. Exploit
Vulnerable
Host
3. DNS Lookup (*)

特性

- 可操控的主機常多達數萬台之多
- 會執行一系列的攻擊行為，從散播 Spam、垃圾廣告、病毒到發動阻斷服務攻擊。
- 自動搜尋擁有弱點的電腦主機，具有自動複製散佈功能

惡意程式特性比較表

類型／特點	傳播性	可控性	竊密性	危害等級
Bot	可控傳播	高度可控	有	完全控制
Trojan horse	無	可控	有	完全控制
Worm	主動傳播	無／弱	無／弱	主機和網路資源
Spyware	無	無	嚴重竊密	資料外洩
Virus	干預傳播	無	無	感染文件

From CNCERT/CC 技術文章「關於殭屍網路」*CNCERTCC_TR_2005-001*

常見的惡意行為

- 分散式阻斷服務攻擊(DDOS)
- 寄送垃圾郵件
- 釣魚(偽造網頁)
- 廣告軟體 (Trojan horse)
- 間碟軟體 (keylogging, information harvesting)
- 儲存盜版內容

Botnet統計資料

- Botnets
  - 大量的bots所組成
  - 透過IRC channel控制與下達命令給bots
  - 擁有1,000台bots的botnet，規模仍不算大
  - 根據觀察，每小時有8,000台bot與C&C進行連線
  - 規模高達20萬到30萬台bots的botnet是可能存在的
  - Channels with < 100 bots are rare
  - Example: One random Undernet observation
    - 52,000 visible
    - 67,000 invisible
    - 30,000 are bots

分類

- 依控制方式
  - IRC Botnet：控制和通信方式為利用 IRC 協定的Botnet
    - spyBot、GTBot 和 SDBot
  - AOL Botnet：與 IRC Bot 類似，AOL 為美國線上提供的一種即時通信服務
    - AIM-CanBot 和 Fizzer
  - P2P Botnet：包含了 P2P 的用戶端，可以連入採用了 Gnutella 技術(一種開放源碼的檔共用技術）的服務器，利用 WASTE 檔共用協議進行相互通信。
    - AgoBot 和 PhatBot

P2P Botnets

- 由於Bots 處於分散式管理的網路環境下，因此偵測十分的困難，而且P2P botnet比集中控管式的botnet更具恢復能力
- 比較新的安全議題，為研究的主要對象

真實的威脅：P2P Botnets

- P2P botnets 是現今botnets進一步進化的結果
- 較佳的復原能力 (分散式管理)
- 難以追蹤與阻止C&C
- 已露端倪的特性 (例如，分散式架構，高頻寬但規模小)
- 自動新增bots，因此botnet規模不會隨著時間而減少
- 變種速度快

Torpig Botnet

Based on UCSB lab study from 10 days data

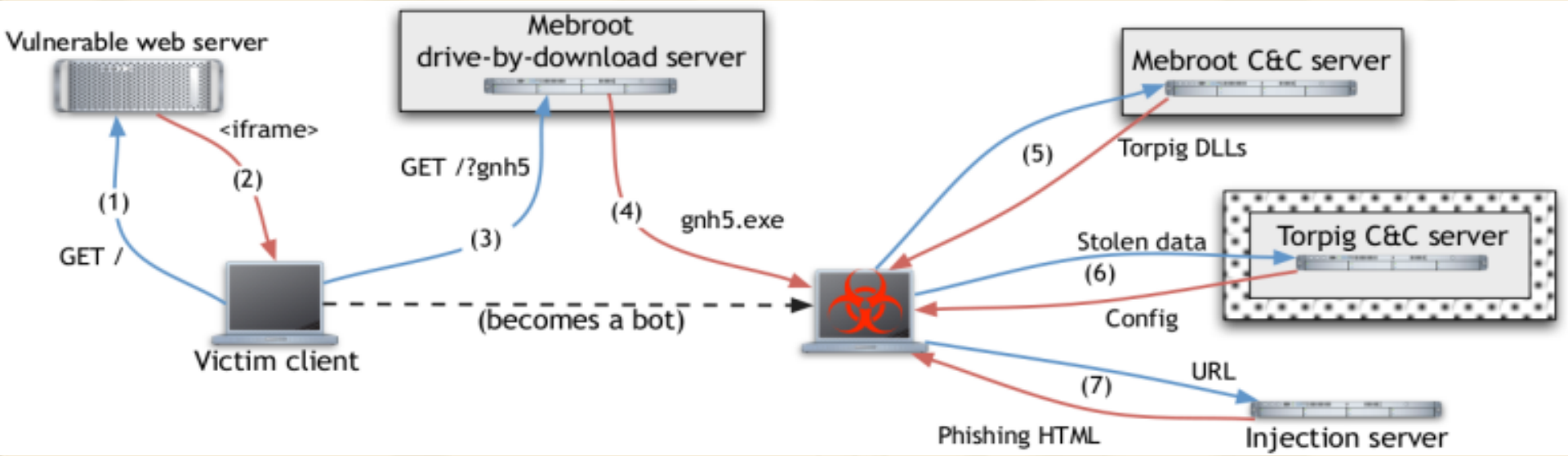
Study Torpig Behavior

- 研究Botnet的方式
  - 被動式分析(Passive analysis)
  - DNS query
  - 分析網路流量
  - 滲入C&C server
    - 觀察與C&C server互動的封包流量
      - 事先取得惡意軟體
      - 在特定環境下執行惡意軟體，成為受感染的機器(bot)
    - 或是加入botnet，成為botnet的一份子

研究Botnet的方式

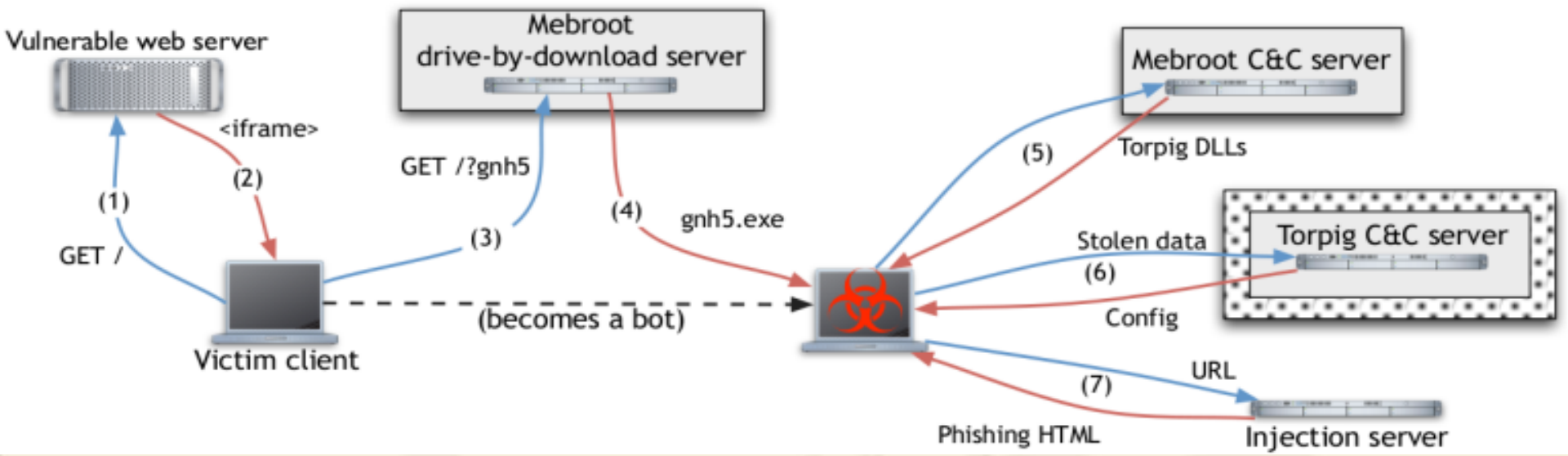
- 劫持(Hijack)
  - 取得實體機器
    - Rustock botnet
  - 竄改DNS所回應的資訊
    - 使DNS所回應的並不是C&C server所對應的IP位址，而是受到我們控制的主機的IP位址
    - Domain flux
      - 本研究中，用來劫持Torpig botnet的技術
      - 此技術存在弱點

Torpig 運作方式



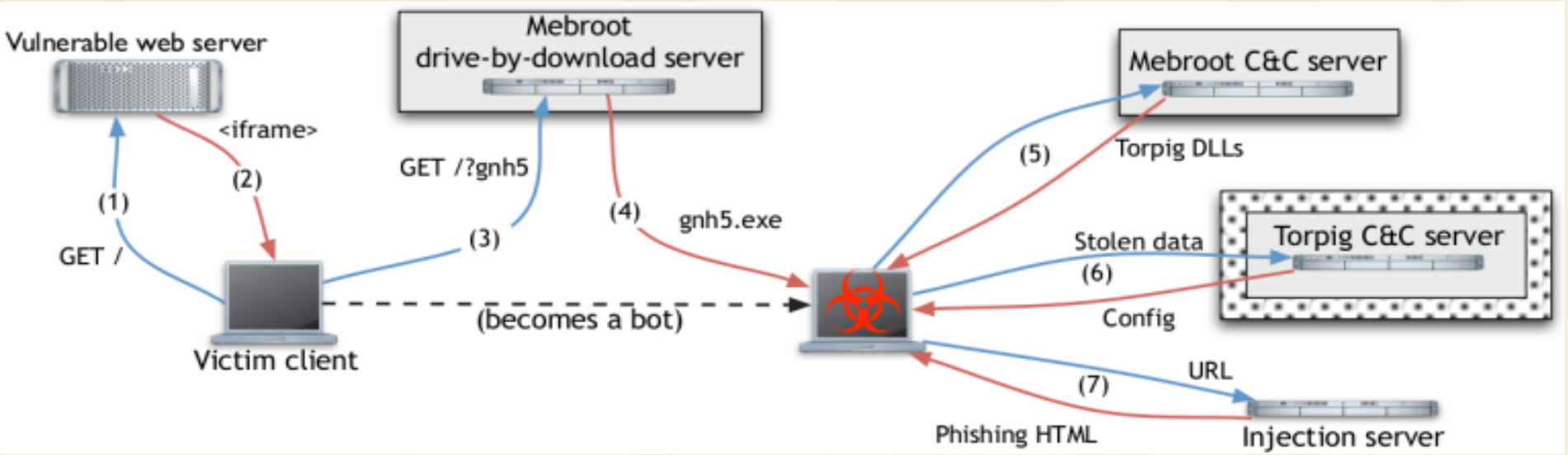
1. 拜訪一個合法但被駭客修改過的網站
2. 此網站會要求瀏覽器拜訪drive-by-download server
3. 瀏覽器拜訪drive-by-download server，向其要求一段JavaScript code並執行，目的在尋找瀏覽器或附加元件是否存在弱點

Torpig 運作方式



4. 若成功找到弱點，便從該server下載惡意程式(Mebroot)並執行，此時該主機便成為botnet的一分子

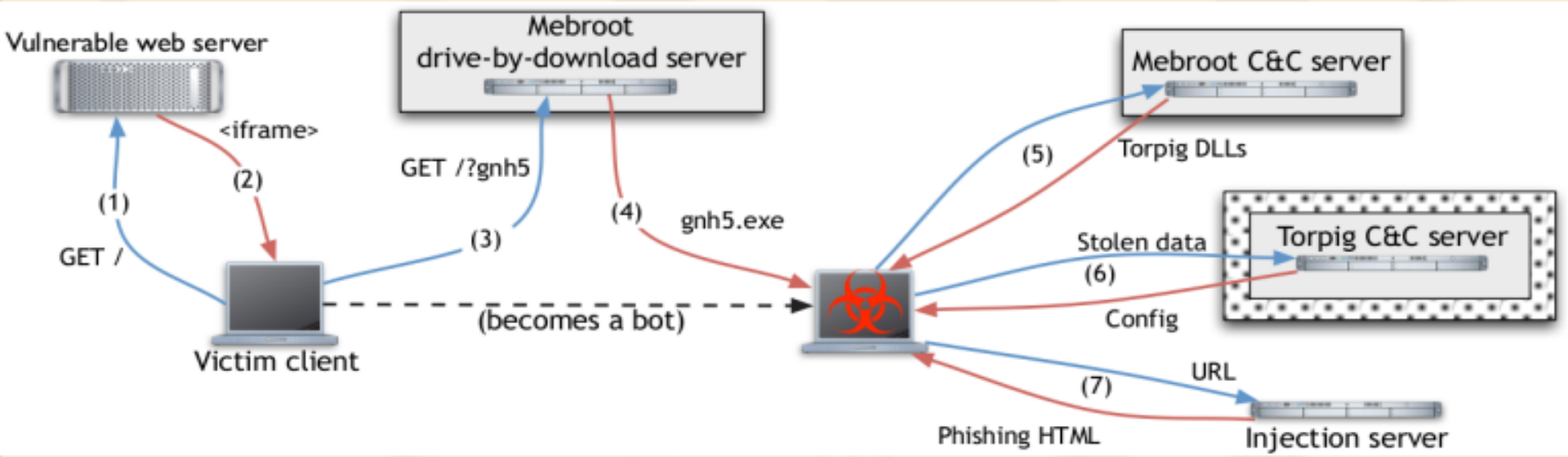
Torpig 運作方式



5. 安裝Mebroot的過程

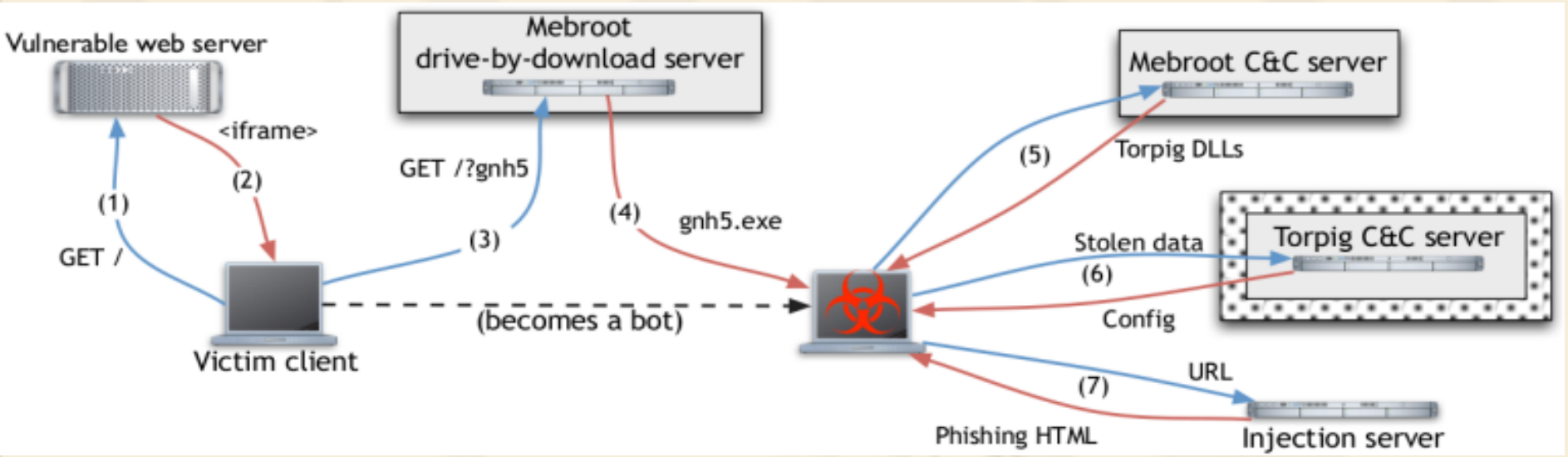
- 注入DLL到explorer.exe，使之後的一切行為看似合法
- 載入driver
- 使用Mebroot覆寫(overwrite)MBR的內容，讓之後每次開機時，Mebroot能先於OS被載入記憶體並執行。

Torpig 運作方式



5. Mebroot本身沒有任何惡意的行為能力，因此在安裝完成後，Mebroot會連線到Mebroot C&C server取得相關惡意模組

Torpig 運作方式



6. 在完成初次更新後，Mebroot會每隔二小時與C&C server建立連線以回傳竊取的資料並更新設定檔

Domain flux

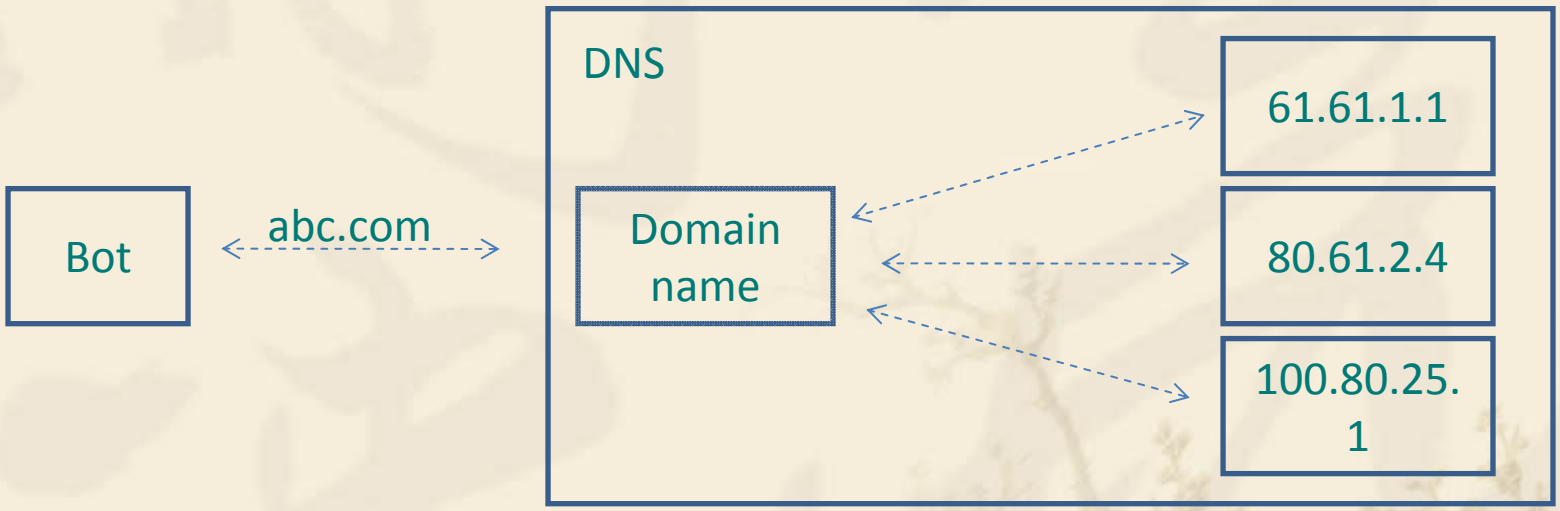
- Bot用於追蹤C&C server的技術
  - 使用事先指定好的IP位址或網域(domain)
    - 此方式易被滲入
      - 收集botnet資料
      - 取得其它bot的IP位址
  - IP fast-flux
  - Domain flux

Domain flux

- IP fast-flux
  - 一個網域(domain)對應到多個IP位址
  - 網域所對應的IP位址會快速的變動

Domain flux

- **IP fast-flux**

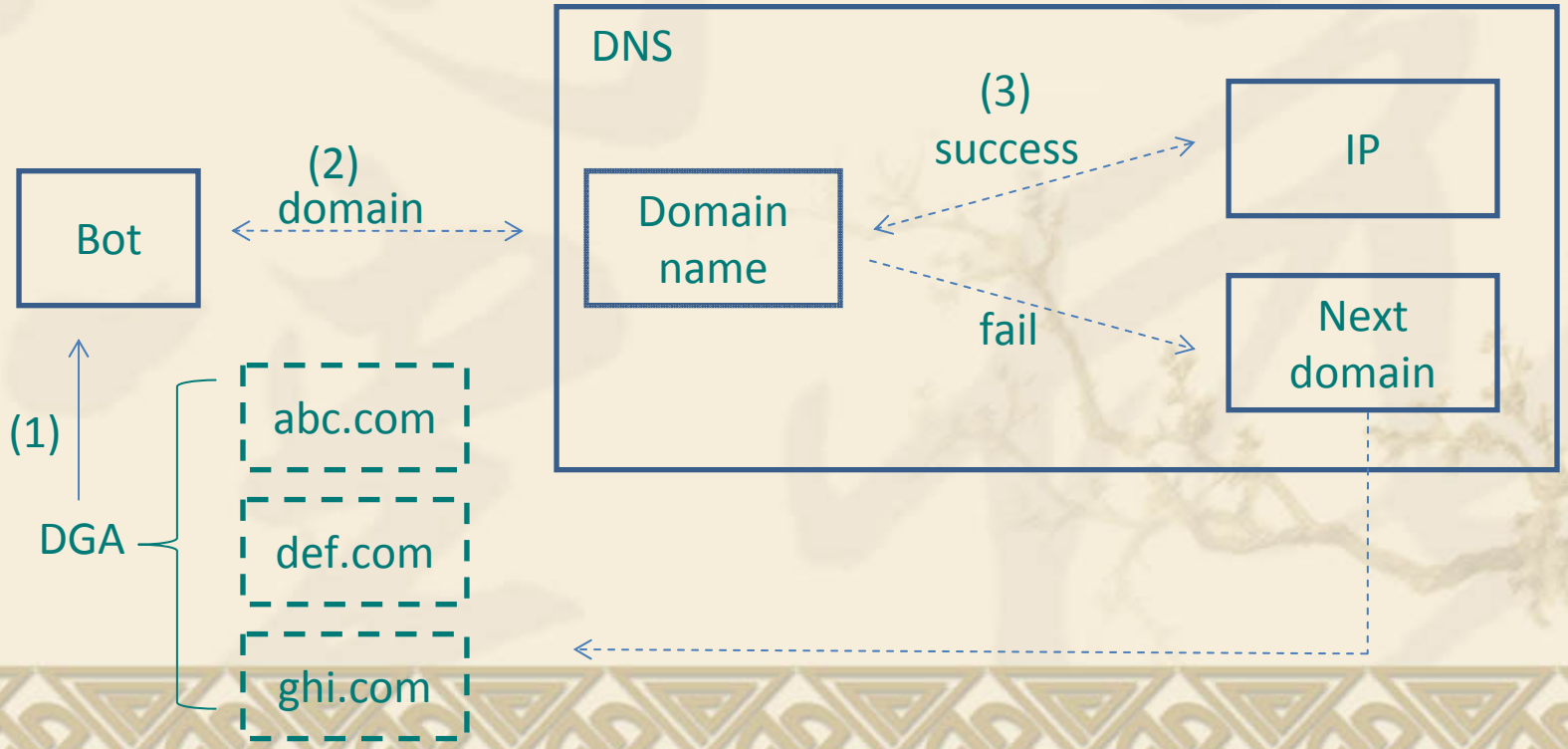


Domain flux

- Domain flux
  - 與IP fast-flux的概念相反
  - 多個網域(domain)對應到一個IP位址
  - bot使用DGA取得一組網域(domain)
    - DGA：Domain generation algorithm
    - 網域(domain)的產生是隨機的

Domain flux

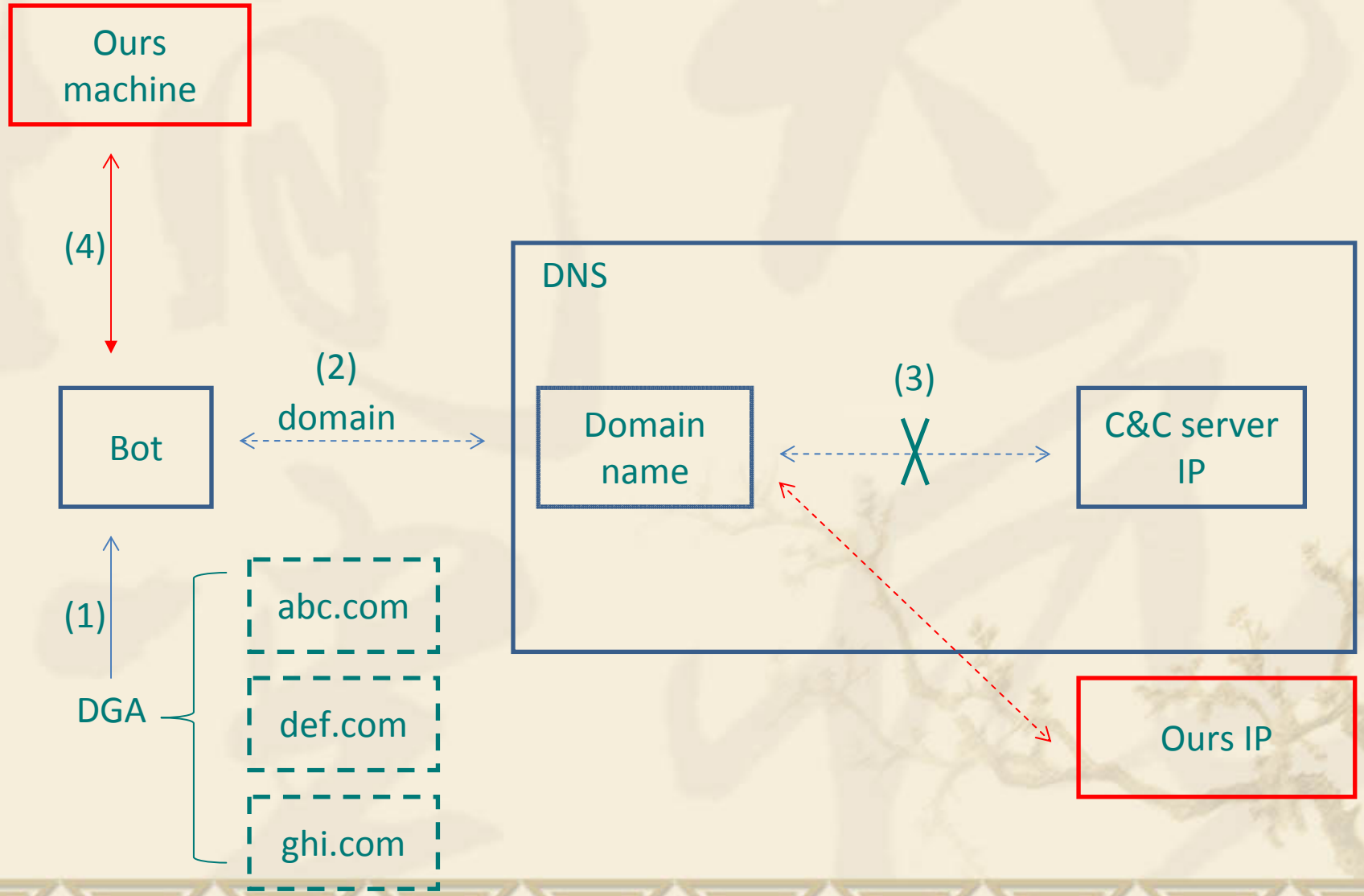
- **Domain flux**
  - bot會週期性的產生一組網域(domain)
  - bot試著解析所產生的網域。若確實有與該網域相對應的IP位址，則與此IP位址建立連線；若無，則解析下一個網域。
  - 若對方能回給bot一個有效的回應，則代表找到C&C；若否，則持續尋找下去。



如何取得Torpig 控制權

- Domain flux的弱點
  - 可透過反組譯事先算出可能產生的網域
  - 駭客不可能對所有的網域事先都進行註冊
- 方法：
  - 先行對那些會被bots所聯繫的網域進行註冊
  - 偽造有效的回應給bots

如何取得Torpig 控制權



結論

- Botnet是現今網際網路最大的威脅
  - 許多攻擊的來源
- 只有在網路層才能成功的偵測與遏制
  - 最理想的情況是在攻擊之前就被偵測出

參考資料

- www.cc.gatech.edu/classes/AY2006/cs6262_spring/**botnets**.ppt
- www.hudakville.com/infosec/**botnets**.ppt
- www.utd.edu/~bxt043000/cs4398_f08/Lecture17.ppt
- UCSB, Your Botnet is My Botnet: Analysis of a Botnet Takeover