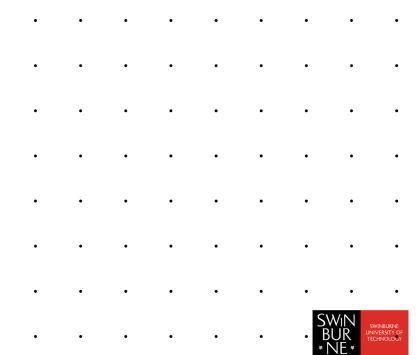


# **DNS Attacks**



#### **DNS** and Hosts

- Before the internet got big, people typed in the IP address of each site they wanted to visit.
- This became tedious, so a scheme was devised whereby the host and domain name of frequently visited web sites was added to a file (HOSTS) which is used to look up the IP address which corresponds to a *host.domain* name. Users could download and install updated versions of *HOSTS* files from popular web sites.
- DNS automated the domain name lookup process using dedicated DNS servers which could communicate with each other and any interested PC through UDP port 53.



#### DNS and Hosts...

- The contents of a HOSTS file take precedence over DNS queries. HOSTS should be empty or just contain the loopback address (127.0.0.1) on modern PCs.
- A common strategy used by spy/adware, browser hijackers and pharmers is to add incorrect entries to the HOSTS file so that correctly entered URIs direct you to the wrong web site.
- A common strategy used for web filtering is to add the domain names of forbidden sites to the hosts file with the loopback address.
- On Linux: see /etc/hosts
- On Win: see \Windows\System32\drivers\etc\hosts



## **DNS Attacks**

#### DNS cache poisoning

- > Attacker sends many DNS queries and replies at the same time. The replies spoof the IP of the authoritative name server.
- ➤ Need to guess the DNS query ID (0-65535)
  - may be predictable
  - birthday paradox increases chances of a correct guess to ~50%

#### Kaminsky attack

- Query non-existent hosts on the target domain.
- Easier to guess the DNS query ID
- DNS client caches ADDITIONAL record (including IP of target host)



# **DNS Cache Poisoning Prevention**

- Use random identifiers for queries
- Always check identifiers
- Port randomization for DNS requests
- Deploy DNSSEC
  - Challenging because it is still being deployed and requires reciprocity



# Kaminsky Defences

- 2008 patches which increases randomness of DNS IDs
- Local DNS servers only accept queries from inside the network
- Source-port randomisation
- UDP packet ID combined with DNS ID to enlarge ID range to 32bits (3 billion)
- DNSSec



#### **DNSSec**

- DNSSec adds two important features to the DNS protocol
  - > Data origin authentication (verify where data came from) & Data integrity protection (verify data didn't been modified during transit)
- DNS queries, replies are digitally signed.
  - prevents replies from being spoofed (fake source address, ID)
  - > requires public key crypto, chain of trust
    - But certificates, private keys get stolen
    - DNSSec adoption very small (1%)
    - https://www.eweek.com/security/dnssec-adoption-needs-to-grow-to-secure-core-internet-

protocols/

# NSLookup (Win/Linux/Mac)

- NSLookup queries the domain name system to find the IP address of a domain name.
- NSLookup can
  - > Resolve DNS Issues
  - Search for optimal mail servers



## **NSLookup**

- NSLookup works for different levels of the DNS system.
- For the details of the .edu.au name servers, try

```
c:\temp>nslookup -type=mx edu.au
Server: venus.it.swin.edu.au
Address: 136.186.5.30
edu.au
     primary name server = ns1.ausregistry.net
     responsible mail addr = dns.ausregistry.net.au
     serial = 2003071563
     refresh = 14400 (4 hours)
     retry = 3600 (1 \text{ hour})
     expire = 3600000 (41 \text{ days } 16 \text{ hours})
     default TTL = 86400 (1 day)
```



# Reverse DNS Lookup

- Use this to verify that an IP points to a domain name.
  - "The reverse DNS entry for an IP is found by reversing the IP, adding it to "in-addr.arpa", and looking up the PTR record. So, the reverse DNS entry for 66.249.72.14 is found by looking up the PTR record for 14.72.249.66.in-addr.arpa." DNSStuff.com

```
c:\temp>nslookup -type=ptr 14.72.249.66.in-addr.arpa
```

Server: venus.it.swin.edu.au

Address: 136.186.5.30

Non-authoritative answer:

14.72.249.66.in-addr.arpa name = crawl-66-249-72-14.googlebot.com

72.249.66.in-addr.arpa nameserver = ns4.google.com 72.249.66.in-addr.arpa nameserver = ns1.google.com 72.249.66.in-addr.arpa nameserver = ns2.google.com 72.249.66.in-addr.arpa nameserver = ns3.google.com ns1.google.com internet address = 216.239.32.10 ns2.google.com internet address = 216.239.34.10 ns3.google.com internet address = 216.239.36.10 ns4.google.com internet address = 216.239.38.10



## Whois

- Gets name server info + details of system administrators.
- Difficult to use (have to get the right whois server).

[jhamlynharris@mercury jhamlynharris]\$
whois -h whois.melbourneit.com
telstra.COM
[whois.melbourneit.com]

Domain Name........ telstra.com
Creation Date....... 1995-09-14
Registration Date..... 2001-08-28
Expiry Date....... 2009-09-13
Organisation Name.... Telstra Corporation
Organisation Address. 18/300
Organisation Address. MELBOURNE
Organisation Address. 3000
Organisation Address. VIC

Admin Name........... Domains Administrator Admin Address........ 18/300 Admin Address....... MELBOURNE

Organisation Address. AUSTRALIA

Admin Address...... 3000 Admin Address...... VIC Admin Address.....AUSTRALIA Admin Email.......cdp@tppinternet.com Admin Phone...... +61.883084046 Admin Fax..... Tech Name...... Domains Administrator Tech Address....... 18/300 Tech Address..... Tech Address...... MELBOURNE Tech Address...... 3000 Tech Address...... VIC Tech Address...... AUSTRALIA Tech Email..... corpdomains@team.telstra.com Tech Phone.....+61.883084046 Tech Fax..... Name Server...... dns0.telstra.net Name Server...... dns1.telstra.net Name Server..... sec1.apnic.net

[jhamlynharris@mercury jhamlynharris]\$

Name Server..... sec3.apnic.net

- On the web at <u>http://www.whois.net/</u>
- and on Linux computers.
  - try whois –h whois.aunic.net swin.edu.au

