Searching for DNS Cache Poisoners

Duane Wessels
The Measurement Factory/ISC/CAIDA
wessels@measurement-factory.com

July 25, 2005

Motivation

- During March/April 2005, SANS Internet Storm Center reports a number of DNS cache poisoning "attacks" are occurring.
- http://isc.sans.org/diary.php?date=2005-04-03
  http://isc.sans.org/presentations/dnspoisoning.php
- Poisoned nameservers have bogus NS records for the *com* zone.
- May have been a vector for spyware propagation.
- Microsoft Windows NT, 2000, 2003 are affected.

The Poisoning Attack

- An authoritative nameserver is configured to return a bogus and out-of-bailiwick NS authority record. See example next slide.
- A caching resolver trusts and caches the bogus referral.
- Future queries for names in the poisoned zone go to the bogus nameserver.
- The bogus nameserver returns incorrect answers to queries that it should not be receiving.

Poison

```
; <<>> DiG 9.3.1 <<>> +trace none.cc
;; global options:  printcmd
.                       30321   IN      NS      A.ROOT-SERVERS.NET.
.                       30321   IN      NS      H.ROOT-SERVERS.NET.
.                       30321   IN      NS      C.ROOT-SERVERS.NET.
.                       30321   IN      NS      G.ROOT-SERVERS.NET.
.                       30321   IN      NS      F.ROOT-SERVERS.NET.
.                       30321   IN      NS      B.ROOT-SERVERS.NET.
.                       30321   IN      NS      J.ROOT-SERVERS.NET.
.                       30321   IN      NS      K.ROOT-SERVERS.NET.
.                       30321   IN      NS      L.ROOT-SERVERS.NET.
.                       30321   IN      NS      M.ROOT-SERVERS.NET.
.                       30321   IN      NS      I.ROOT-SERVERS.NET.
.                       30321   IN      NS      E.ROOT-SERVERS.NET.
.                       30321   IN      NS      D.ROOT-SERVERS.NET.
;; Received 436 bytes from 206.168.0.2#53(206.168.0.2) in 3 ms
```

Poison, cont

```
cc.                     172800  IN      NS      L3.NSTLD.COM.
cc.                     172800  IN      NS      D3.NSTLD.COM.
cc.                     172800  IN      NS      A3.NSTLD.COM.
cc.                     172800  IN      NS      E3.NSTLD.COM.
cc.                     172800  IN      NS      C3.NSTLD.COM.
cc.                     172800  IN      NS      G3.NSTLD.COM.
cc.                     172800  IN      NS      M3.NSTLD.COM.
cc.                     172800  IN      NS      H3.NSTLD.COM.
;; Received 298 bytes from 198.41.0.4#53(A.ROOT-SERVERS.NET) in 52 ms
```

Poison, cont

```
none.cc.                172800  IN      NS      NS1.FRAKES.NET.
none.cc.                172800  IN      NS      NS2.FRAKES.NET.
;; Received 71 bytes from 192.41.162.32#53(L3.NSTLD.COM) in 56 ms

none.cc.                86400   IN      A       64.202.173.35
cc.                     86400   IN      NS      ns3.cc.
cc.                     86400   IN      NS      ns1.cc.
cc.                     86400   IN      NS      ns2.cc.
;; Received 143 bytes from 66.249.7.25#53(NS2.FRAKES.NET) in 51 ms
```

Poison, cont

```
; <<>> DiG 9.3.1 <<>> @ns2.frakes.net boogaboogabooga.cc

;; QUESTION SECTION:
;boogaboogabooga.cc.            IN      A

;; ANSWER SECTION:
boogaboogabooga.cc.     86400   IN      A       64.202.173.35

;; AUTHORITY SECTION:
cc.                     86400   IN      NS      ns3.cc.
cc.                     86400   IN      NS      ns1.cc.
cc.                     86400   IN      NS      ns2.cc.

;; ADDITIONAL SECTION:
ns1.cc.                 86400   IN      A       66.249.1.244
ns2.cc.                 86400   IN      A       66.249.7.25
ns3.cc.                 86400   IN      A       66.249.1.100
```

Wildcard?

Vulnerable Implementations

- Windows NT
  - vulnerable by default
  - SP4 and later can become not-vulnerable after editing registry
- Windows 2000
  - SP1, SP2 vulnerable by default
  - SP3 and later not-vulnerable by default
- Windows 2003
  - not-vulnerable by default
- Symantec gateway/firewall products

Searching for Poisoners

- Start with a (large) list of DNS names or zones.
- Discover the set of authoritative nameservers for a zone by following referrals starting at the root (or at least TLD).
- Query each authoritative nameserver.
- Compare the NS RR set in each reply to the previously-learned referrals for parent zones.
- This technique only finds parent-zone poisoning. Furthermore, we are limiting our search to TLD poisoning at this point.

Survey 2005-06-03

- Input is 12,521,883 names from tcpdump on F-root.
- Found 172 "poisoning" nameservers — these return bogus referrals to a TLD.
- The following zones are poisoned:

zone	count
.	150
com	13
net	10
cc	2
info	2
cn	1
org	1

- Some nameservers poison multiple zones

Is the Sky Falling?

- With so many poisoners out there, why don't we hear about more problems?
- Fortunately, it seems that most implementations do not allow the root zone to be poisoned.
- Maybe nobody ever uses the names for which they are authoritative.
- Maybe the bogus nameservers return "NXDOMAIN" or some other non-answer.
  - yes, some do
- Maybe they answer and proxy the (web) traffic so the user doesn't even realize it.
  - yes, some do

Absence of Malice?

Never attribute to malice what can adequately be explained by stupidity

- Many of the poisoners are companies that provide DNS-related services
  - registrars
  - resellers
  - speculators
  - brokers
- Others appear to be legitimate companies.
- They should know better.
- Many of the names leading to poisoners are within expired zones. That is, put the name in your browser and you see a page telling you "This domain name has expired. Click here to pay."

Stupidity

- We suspect that many of these potential poisoners are just being lazy.
- For example, the BIND nameserver requires one file per zone, which becomes a problem when you have many zones.
- So they are probably creating a zone file for the parent and listing all their zones, or worse, using wildcards.

Stupidity, For Example

```
$ORIGIN com.
@               IN      SOA     ns1.goober.com root.goober.com
                ( 100 200 300 400 500 )
                IN      NS      ns1.goober.com.
                IN      NS      ns2.goober.com.
expired1        IN      A       192.168.0.1
expired2        IN      A       192.168.0.1
expired3        IN      A       192.168.0.1
expired4        IN      A       192.168.0.1
...
```

It's also likely that they would use a wildcard, rather than list the domains individually.

Next Steps

- Continue scanning for poisoners and nameservers with bogus referrals.
- Automate the procedure
- Make weekly "shame list" reports available to OARC members and network operators.
- Try to categorize poisoners as malicious, lazy, etc.
- Consider other ways to poison a DNS cache.

The End